DOCTOR OF PHILOSOPHY

Quantitative decision making in reverse logistics networks with uncertainty and quality of returns considerations

Niknejad, Ali

Award date:
2014

Awarding institution:
Coventry University

Quantitative Decision Making in Reverse Logistics Networks with Uncertainty and Quality of Returns Considerations

Ali Niknejad

Faculty of Engineering and Computing

Coventry University

A thesis submitted in partial fulfilment of the University's requirements for the Degree of

Doctor of Philosophy

September, 2014

To Yeganeh, my cheerful lovely wife

To my patient loving mother

To the memory of my father

To Jila, Nora and Albert

Declaration

I declare that the work described in this PhD thesis, unless otherwise stated in the text, is my own work and has not been previously submitted for any academic degree.

Ali Niknejad

September, 2014

Acknowledgements

I would like to express my sincere gratitude to my Director of Studies, Prof. Dobrila Petrovic, who has been the constant source of support and guidance throughout this research. Her supervision of this project was essential and, without doubt, this would not be possible without her.

I would like to gratefully acknowledge Prof. Keith Burnham, my supervisor and the director of Control Theory and Applications Centre (CTAC), for his invaluable help during this research. I also wish to thank my fellow students and colleagues at Coventry University for the friendly and pleasant working atmosphere.

Finally, I wish to acknowledge my friends and family for their support and encouragement during this research which have helped me greatly to remain motivated and focused.

Abstract

Quantitative modelling of reverse logistics networks and product recovery have been the focus of many research activities in the past few decades. Interest to these models are mostly due to the complexity of reverse logistics networks that necessitates further analysis with the help of mathematical models. In comparison to the traditional forward logistics networks, reverse logistics networks have to deal with the quality of returns issues as well as a high degree of uncertainty in return flow. Additionally, a variety of recovery routes, such as reuse, repair, remanufacturing and recycling, exist. The decision making for utilising these routes requires the quality of returns and uncertainty of return flow to be considered.

In this research, integrated forward and reverse logistics networks with repair, remanufacturing and disposal routes are considered. Returns are assumed to be classified based on their quality in ordinal quality levels and quality thresholds are used to split the returned products into repairable, remanufacturable and disposable returns. Fuzzy numbers are used to model the uncertainty in demand and return quantities of different quality levels. Setup costs, non-stationary demand and return quantities, and different lead times have been considered.

To facilitate decision making in such networks, a two phase optimisation model is proposed. Given quality thresholds as parameters, the decision variables including the quantities of products being sent to repair, disassembly and disposal, components to be procured and products to be repaired, disassembled or produced for each time period within the time horizon are determined using a fuzzy optimisation model. A sensitivity analysis of the fuzzy optimisation model is carried out on the network parameters including quantity of returned products, unit repair and disassembly costs and procurement, production, disassembly and repair setup costs. A fuzzy controller is proposed to determine quality thresholds based on some ratios of the reverse logistics network parameters including repair to new unit cost, disassembly to new unit cost, repair to disassembly setup, disassembly to procurement setup and return to demand ratios. Fuzzy controller's sensitivity is also examined in relation to parameters such as average repair and disassembly costs, repair, disassembly, production and procurement setup costs and return to demand ratio. Finally, a genetic fuzzy method is developed to tune the fuzzy controller and improve its rule base. The rule base obtained and the results of sensitivity analyses are utilised to gain better managerial insights into these reverse logistics networks.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Within the past few decades, environmental concerns have raised attention to product recovery and sustainability of supply chains and logistics networks. Consumer's inclination toward environmental responsibility and legal pressure for sustainable products along with economic benefits of product recovery are among the main reasons which led manufacturers to integrate recovery activities into their processes (Brito and Dekker, 2004; Ilgin and Gupta, 2010).

Reverse logistics (RL) refers to the essential activities for product recovery and disposal. Proliferation of RL networks leads to a variety of unique challenges for decisions makers in logistics networks. Uncertainty of return flow, the diverse quality of returned products and quality dependent routing of returns are some of the main challenges that are unique to RL networks (Galbreth and Blackburn, 2006). These issues are the inspiration behind this research.

This research provides solutions to facilitate decision making in integrated RL networks with uncertainty in demand, return quantity and quality of returns. Such networks exist in a variety of industries, from white goods and electronics to tyres (Lebreton and Tuma, 2006; Srivastava, 2008). In some cases, Original Equipment Manufacturers (OEM) decide to enter the recovery market of their own products, either because of take-back laws or voluntarily, but encounter a considerable increase in complexity of managing the network (Krikke et al., 1998b). Although significant capital is at stake in these networks, little has been done to address efficient decision making. To this end, this research will help to improve the decision making in such networks.

In this chapter, we first investigate some of the background concepts, paying a special attention to the definition of RL. Uncertainty in RL networks and quality of returned products and its effect on routing the returns, as the two main unique issues in decision making for RL, are briefly discussed afterwards. The objectives of this research are presented in the following section. Next, the overall outline of the thesis is presented. Publications generated during this research are mentioned at the end, including the literary contributions of a journal paper, a book chapter, conference papers and presentations.

1.1 Background concepts of reverse logistics

> *Reverse logistics is the process of planning, implementing and controlling backward flows of raw materials, in process inventory, packaging and finished goods, from a manufacturing, distribution or use point, to a point of recovery or point of proper disposal.*
>
> *European Working Group on Reverse Logistics, RevLog (1998-) (Brito and Dekker, 2004)*

As the RevLog's definition suggests, RL deals with a variety of necessary activities to handle unwanted and returned products with the aim of recovering the potential value from those products while also safely and properly disposing what is not valuable. Some authors have stressed the importance of value recovery in the RL, which can be a crucial feature that sets apart RL from the traditional waste management activities (Brito and Dekker, 2004).

RL is different from the traditional logistics activities, i.e. *Forward Logistics*, which considers production from raw materials and the flow from the point of origin to the consumer. However, the RL activities are sometimes combined with the forward production activities which leads to *integrated forward/reverse logistics networks* (Fleischmann et al., 1997). Especially, in the recent years, 'producer responsibility laws', among other reasons, have encouraged Original Equipment Manufacturers (OEMs) to integrate reverse logistics activities within their networks (Dowlatshahi, 2000). It is also worth mentioning that *closed loop supply chains* refer to the supply chains that rely on product recovery to supply the customers. Integrated logistics networks and closed loop supply chains are similar concepts which are often used interchangeably in the literature. However, they can be distinguished by the coordinated flow of forward and return materials in a closed loop supply chain: products are returned to the same supply chain and also, only products produced by the same supply chain are used in recovery. These assumptions are not necessarily true for an integrated forward and reverse logistics networks (Bei and Linyan, 2005).

RL has a close relationship with the environmental concerns in the logistics networks which are collectively addressed as *Green logistics*. Green logistics considers the environmental impact of various logistics activities, including product design, inventory, production and transportation, and the efforts to minimise this impact. However, RL is different from green logistics as green logistics is concerned about the environmental impact of all logistic activities, especially those of the forward chain while RL considers product recovery and does not directly consider environmental issues (Brito and Dekker, 2004).

RL includes product recovery activities which are crucial to sustainability, such as repair, remanufacturing and recycling. While recycling typically refers only to the reuse of materials used for a product without preserving its structure, repair usually involves activities necessary to restore a damaged product into the working order, while preserving its integrity. In contrast, remanufacturing comprises disassembly, replacement of components where necessary and assembly of a product to bring it back into as-good-as-new condition (Thierry and Salomon, 1995). This variety of alternative recovery options are one of the main reasons that RL networks are inherently more complicated than the forward networks.

Quantitative modelling of logistics networks has been used extensively to improve decision making in forward logistics networks (Dekker and Fleischmann, 2004). One such example is *inventory control* models which help determining the right quantity and time to stock inventories. Also, *production planning* is concerned with the production environment and managing the necessary resources, including human resources, raw materials and time that is required to produce the right amount of final products in the right time. The decision making in inventory control and production planning is usually interconnected and it can be difficult to distinguish between the two types of models.

Inventory control and production planning models have also been applied to RL networks including integrated forward-reverse logistics networks. However, the relationships in RL networks are inherently more complicated and traditional inventory control and production planning models designed for forward networks cannot provide a sufficient analysis of RL networks. This issue has led to development of comprehensive models specific to RL networks (Dekker and Fleischmann, 2004).

1.2 Uncertainty in reverse logistics networks

One of the most important features of the reverse flow is the presence of uncertainty in quantity, quality and timing of returned products which needs to be considered when developing quantitative models of reverse flows (Fleischmann et al., 1997; Inderfurth, 2005). This uncertainty is considerably higher than the uncertainty present in forward networks, such as the uncertainty in demand. Hence, in comparison with the forward networks, it is even more important to consider uncertainty in modelling the RL networks.

Uncertainty has adverse effects on performance of logistics networks. In decision making for logistics networks, it is necessary to consider these effects in order to mitigate the negative consequences as much as possible. For example, higher uncertainty in return quantities should be compensated by an increase in safety stocks of either returned or final products to avoid stock outs. However, in this case, an increase in stock will increase the cost of recovery and, in extreme cases, can even make recovery activities uneconomical. Therefore, uncertainty has considerable impact in decision making, especially in RL networks where its presence is eminent.

Two general types of quantitative modelling of uncertainties are typically used: stochastic/probabilistic models and fuzzy logic models. In contrast with probabilistic models that consider the precise likelihood of an event as the source of uncertainty, fuzzy logic deals with vagueness and imprecision of values. Fuzzy logic is particularly advantageous in the absence of historical data where precise statistical models cannot be created (Qin and Ji, 2010). In such situations, experts' judgements can be employed by using fuzzy linguistic variables, for example to roughly estimate quantity and quality of returned products.

However, applications of fuzzy logic to reverse logistics quantitative modelling are very limited. This is surprising as it is very difficult to estimate return quantity and quality with accuracy (de Brito and van der Laan, 2009). Hence, the application of fuzzy logic to model the uncertainty in product returns is among the main objectives of this research.

1.3 Quality of returned products in RL

Quality of returned products is arguably the most difficult issue to handle in RL networks. Products are returned for a variety of reasons and their quality is of high variability. Also, it is reasonable to say that quality significantly affects the cost of the recovery for a product and the value that can be potentially recovered from that product. This fact can influence most decisions in the RL network from scheduling to inventory control (Guide and Wassenhove, 2001).

Inspection is one of the crucial stages of the RL networks. The opportunities to acquire knowledge about quality of returns before their arrival to the RL network are very limited or even non-existent. Also, variability in quality requires knowledge about the qualities as it is simply impossible to treat all returns in the same manner; one might be reusable as is, the other is damaged beyond any hope of recovery. Hence, to make proper decisions for the product recovery, it is necessary for the returns to go through inspection (Nenes et al., 2010). It is in this stage that the fate of a particular product will be determined and the appropriate recovery route is chosen. Also, to facilitate and standardise inspection, *quality grading* systems are used in some industries to classify the returns into quality levels characterised by predefined criteria (Guide and Wassenhove, 2001).

Quite obviously, quality affects product recovery. Depending on the quality of the returned product, different recovery activities such as reuse, repair, remanufacturing, recycling or even disposal can be applicable to a particular product; very low quality returns should be recycled or disposed while a very high quality returned product can be even directly reused. This necessitates alternative recovery routes and quality dependent routing of returns. However, the relationship between quality of returns and the best recovery route for the products is very sparsely researched. One of the main objectives of this research is to better understand this relationship and provide more insights into decision making in presence of quality of returns.

Finally, it is worth mentioning that recovery networks are not always passive recipients of product returns. It is possible to control the quality of product return by using product acquisition management methods such as financial incentives to encourage higher quality of returns (Guide and Wassenhove, 2001). This however is not always possible. Contractual or legal constraints, for example, may prohibit the firm from rejecting lower quality returns.

1.4 Aim and objectives of this research

The main aim of this research is to understand the relationship between quality of returns and the inherent uncertainty of returned products with the decision making in the RL networks, particularly with regard to quality dependent routing. By analysing these relationships, it is possible to ultimately extract and summarise knowledge about the RL networks and the effects of uncertainty and quality of returns on their performance. This extracted knowledge can be of help to managers of such networks to improve the decision making in the presence of product returns. With this aim in mind, the following objectives for this research are considered:

1. To identify sources of uncertainties in an integrated RL network and investigate the possibility of using fuzzy logic for representing the identified uncertainties.

2. To deal with the quality of returns in a realistic way, considering the influence of quality on the decision making especially with regard to the optimal recovery route, as a result of varying recovery costs.

3. To develop an optimisation model for decision making in an integrated RL network with simultaneous repair and remanufacturing activities and also disposal.

4. To develop a fuzzy controller for determining the proper quality levels for each recovery channel based on some of the main network parameters such as unit recovery, purchasing and setup costs.

5. To investigate fine tuning the proposed fuzzy controller based on optimisation results, in order to enrich the proposed fuzzy rule base.

6. To develop a new software that can accommodate the mentioned optimisation model, the fuzzy controller and the tuning algorithm.

7. To employ sensitivity analysis to better understand the influence of individual parameters on the optimisation and controller's results.

8. To interpret the improved fuzzy controller to gain more insights into the relationships within the integrated RL networks to enhance our understanding of the topic.

1.5 Structure of the thesis

This thesis consists of seven chapters. Each of the following chapters are briefly introduced as follows:

Chapter 2: Literature Review.

With focus on Reverse Logistics (RL) literature, relevant published papers are discussed in this chapter. Especially, quality of returns and uncertainty in return flow, which are the centre points of this thesis, are analysed and relevant literature is examined. Research areas such as inventory control and production planning, product design, network design and secondary markets are considered.

Chapter 3: Background of Relevant Methods.

The aim of this chapter is to provide a brief introduction to the methods that are used throughout this thesis. Mainly, fuzzy logic, mathematical optimisation and genetic algorithm are discussed. More specifically, fuzzy arithmetic, fuzzy control, fuzzy optimisation and genetic fuzzy methods are essential to the following chapters and, hence, they are introduced in Chapter 3.

Chapter 4: Reverse Logistics Fuzzy Optimisation.

The RL network structure discussed in the thesis is introduced in this chapter. Decision making for this network is split into two phases: Phase 1 where the decisions regarding the acceptable quality of returns is generated, by using quality thresholds, and Phase 2 where the remaining decisions such as quantity of repair, disassembly, procurement and production are decided. In the Phase 2, a fuzzy optimisation method is applied to determine the variables, where the quality threshold values determined in Phase 1 are used as input into Phase 2. Sensitivity analysis is carried out to understand the relationship between the performance of the RL networks and some of the network parameters.

Chapter 5: Development of a Fuzzy Controller to Determine Quality Thresholds.

The optimisation model developed in Chapter 4 assumes that the quality thresholds are given as parameters. In this chapter, fuzzy control is applied to determine the quality thresholds based on a few network parameter ratios including repair to production unit cost ratio, disassembly to production unit cost ratio, repair to disassembly setup ratio, disassembly to procurement setup ratio and return to demand ratio. These parameter ratios are individually examined and appropriate fuzzy rules and membership functions are suggested for each ratio. The quality threshold policy based on the developed fuzzy controller are compared with proposed benchmark policies and sensitivity analysis is used to understand the relationship between the parameter ratios and performance of the policies.

Chapter 6: Genetic-Fuzzy Tuning of Fuzzy Rules.

In Chapter 5, a fuzzy controller has been developed manually. However, many complex relationships cannot be easily identified using a manual approach. In such situations, rule learning techniques can be utilised to improve the controller. In Chapter 6, a genetic fuzzy method is proposed and applied to the problem to fine tune the manually defined controller.

Chapter 7: Conclusions and Further Work.

In this final chapter, outcomes of the research are summarised. The results obtained in the preceding chapters are compared and the managerial implications of them are discussed. At the end, a few directions for future research in this area are suggested.

1.6 Publications generated during the PhD

1.6.1 Journal papers

A. Niknejad and D. Petrovic. Optimisation of Integrated Reverse Logistics Networks with Different Product Recovery Routes. *European Journal of Operational Research*, 2014 [Accepted]

A. Niknejad and D. Petrovic. Development and Tuning of a Fuzzy Controller for Quality Dependent Routing of Product Returns. [To be submitted to an OR journal]

1.6.2 Book chapter

A. Niknejad and D. Petrovic. Introduction to Computational Intelligence Techniques and Areas of Their Applications in Medicine. In A. Agah, editor, *Medical Applications of Artificial Intelligence*, chapter 4, pages 51–70. CRC Press, 2013

1.6.3 Conference papers and proceedings

D. Petrovic and A. Niknejad. Fuzzy logic based model for determining recovery routes for returned products in reverse logistics networks. In *Conference Handbook of OR55 Annual Conference*, 3-5 September, Exeter, UK, 2013 [Abstract]

D. Petrovic and A. Niknejad. Two Phase Optimisation of Reverse Logistics Networks Considering the Quality of Returned Products. In *Abstract Book of 26th European Conference on Operational Research*, 1-4 July, Rome, Italy, 2013 [Abstract]

A. Niknejad and D. Petrovic. Two Phase Optimization of Reverse Logistics Networks. In *Proceedings of 6th International Conference of Iranian Operations Research Society*, 8-9 May, Tehran, Iran, 2013 [Extended Abstract]

A. Niknejad and D. Petrovic. Two Phase Optimization of the Reverse Logistics Network with Multiple Recovery Routes and Quality Inspection. In *Conference Handbook of OR54 Annual Conference*, 4-6 September, Edinburgh, UK, 2012 [Abstract]

A. Niknejad and D. Petrovic. A Genetic-Fuzzy Framework for Optimization of Reverse Logistics Networks with Multiple Recovery Routes. In *Keynotes and Short Papers of OR53 Annual Conference*, pages 53-58, 6-8 September, Nottingham, UK, 2011

Chapter 2

Literature Review

2.1 Introduction

Reverse Logistics (RL) networks have gain more importance in the past two decades due to an increase in awareness of environmental sustainability and also the potential economic benefits of recovery operations (Ilgin and Gupta, 2010). Although recovery provides economic and ecological benefits, it introduces certain challenges to management of logistics networks which is mainly due to the inherent uncertainty of reverse flow, both in term of quantity and quality of returned products (Fleischmann et al., 1997; Inderfurth, 2005).

In this research, we are focused on inventory control and production planning decision making in a RL networks with alternative recovery routes, uncertainty in demand and return and varying quality of returned products. Hence, in the literature review, covering the issues related to this area has been the first priority. To begin, we discuss some of the basic concepts of RL and product recovery including various recovery routes present in RL systems, typical RL structures presented in the literature and environmental concerns. Next, inventory control and production planning problems in RL are examined in more detail. Then, the quality of returns and its effect on some of the main decisions in the RL network are discussed. We aimed for this discussion to be thorough and to provide analysis of various relevant RL features and problems. At the end, the chapter will be concluded by discussing the research topics and gaps in this area.

2.2 Reverse logistics and product recovery

RL is a broad term used to collectively identify product return with all the relevant activities such as returns collection, inventories management of, information flow, network design and planning, product recovery and disposal. It is important to note that RL is different from waste management as RL emphasises on recovery of value from the returned products (Brito and Dekker, 2004). Hence, product recovery plays a significant role in RL as the proper recovery can make the difference between a success and a failure of RL networks.

Several parameters of the RL networks such as the quality of returned products, demand for recovered products, cost of recovery, etc., can influence the appropriate action taken with regard to the returned products. Thierry and Salomon (1995) propose three options for returned products: direct resell, recovery and disposal; recovery is categorised into: *repair*, *refurbishing*, *remanufacturing*, *cannibalization* and *recycling*; disposal can be either *incineration* or *land-filling*. The main difference between these activities is the share of recovery operations in the final product value which means that repair requires the least amount of recovery operations while recycling requires the most.

Thierry and Salomon (1995) define repair as returning product to 'working order' by fixing or replacing broken parts. Refurbishing involves more operations, to bring the product to a specific quality, which is still, usually, less than the quality of a new product. Remanufacturing demands even more, which means bringing the quality to the same level as of the new product. In remanufacturing, the product is always disassembled into its components and after inspection, fixing and replacement, the product will be assembled again into as-good-as-new product. Cannibalization assumes recovery of some of reusable components of the product and their reuse in other recovery activities. Finally, recycling includes material reuse only, so that the product and its components lose their identity and only the materials are recovered. An overview of these alternatives is illustrated in Figure 2.1.

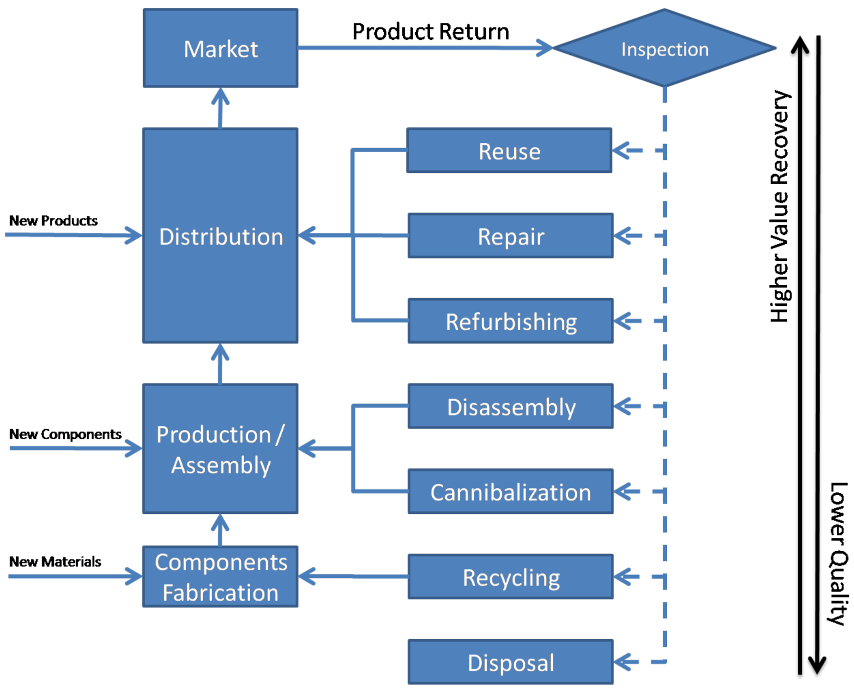

Figure 2.1 An overview of RL recovery alternatives and their relationship with the forward logistics.

Figure 2.1 summarises all configurations of integrated RL networks that is available in the literature, where each model in the literature considers only part of the presented overview. It is worth noting that higher value recovery options are not necessarily the most economical or ecological options. For example, in case of papers, it can be debated that landfill is a better option than recycling as disposing biodegradable paper has less of an adverse effect on the environment than bleaching it (Brito and Dekker, 2004). Also, a refurbished product might have no demand, and hence value, in the market, but it can be valuable as a set of spare parts (Brito and Dekker, 2004). We will see in the following chapters how the quality of the returned products can influence the optimal choice of recovery.

Furthermore, *integration* of forward and reverse logistic networks is also a prominent topic in RL. Integration of the two networks, usually leading to a *Closed Loop Supply Chain (CLSC)*, provides a strategic advantage to Original Equipment Manufacturers (OEMs) by allowing them to recover their own products as opposed to let 3rd party remanufacturers handle the recovery (Brito and Dekker, 2004). It is also a very challenging topic as the existence of both forward and reverse sources in the supply chain aggregates the uncertainty, both in demand and a return flow (Cardoso et al., 2012). General schema of an integrated RL network is presented in Figure 2.2.

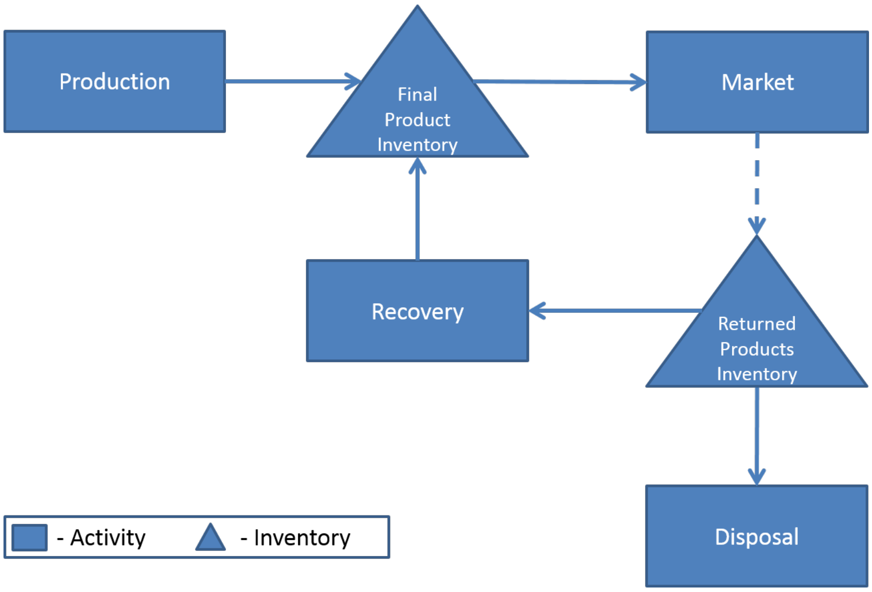

Figure 2.2 An integrated RL network model.

Many papers have examined different RL structures and recovery routes. For example, de la Fuente et al. (2008) provide a case study for integration of forward and reverse networks with repair and recycling in a company from the metal mechanic sector, analysing business processes before and after the integration. Furthermore, Jorjani et al. (2004) propose a piece-wise linear programming model for disassembly decision making considering Equal-To-New (ETN), resell, reuse, recycle and landfill options. Among others, Jayaraman (2006) considers forward production, reuse and remanufacturing, Guide et al. (2005) examine refurbishing and repair, Mitra (2007) examines remanufacturing and refurbishing and Yoo et al. (2012) consider forward production, rework (repair) and salvage (remanufacturing). While many others assume a single recovery route with or without forward production. For example, recycling with production (Dobos and Richter, 2004, 2006), multiple remanufacturing options (Inderfurth et al., 2001), forward production and remanufacturing (Inderfurth, 2004, 2005; Jaber and El Saadany, 2009) and remanufacturing only (Zikopoulos and Tagaras, 2008).

Legal obligations are obviously very important to the industry and their effect on the recovery has been the subject of research as well. With this regard, Biehl et al. (2007) examine the case of US remanufacturers and their upcoming legal obligation to recover 40% of end-of-line carpet. A simulation model for RL network design is used and it has been concluded that the carpet industry will struggle to meet the target goal.

2.3 Inventory control and production planning in RL

Fleischmann et al. (1997) categorise quantitative models in RL into *inventory control*, *production planning* and *distribution planning*. Distribution planning is related to collection and transportation of products and materials throughout the network. Distribution planning often involves designing the network as well as determining the flows between network's nodes. Inventory control in RL is an extension to the traditional inventory control which includes not only the new products and components inventory, but also covers returned products and intermediate recovery materials and outputs. Finally, production planning mainly includes decisions for selecting the right recovery option and scheduling of product recovery through various routes. As quality of returns is uncertain and can vary extensively and also recovery activities are very much dependant on the quality of returns, planning can become significantly harder than the traditional forward-only networks.

Because of the complexity of the reverse flow and its integration with forward flow, dividing inventory control and production planning decisions is not straightforward and will yield to sub-optimal decisions. Also, the approach adapted in our research integrates all inventory control decisions with some of the production planning decisions, including Aggregate Production Planning (APP) and recovery route selection. This approach is popular in the literature for RL decision making (Ilgin and Gupta, 2010) and hence, in this section, we examine the RL literature considering both problems together. For a more detailed analysis of literature in this area, we refer interested readers to Ilgin and Gupta (2010).

Quantitative models for inventory control and production planning can be categorised into three classes: 1) *EOQ (Economic Order Quantity) and lot sizing models*: These models use principles of EOQ to determine the size of procurement, production and recovery batches which are optimal to satisfy demand, which is usually considered to be constant. 2) *Heuristic Policies*: There are custom policies (such as 'pull' or 'push' policies) which are used to heuristically control the network. 3) *Optimisation based models*: These models mathematically formulate logistics networks and then use optimisation methods (such as linear programming, integer programming, etc.) to find the optimal values for decision variables.

These categories are valid for forward logistics as well as reverse logistics. However, in the following sections, we will focus on literature that use these models in RL networks.

2.3.1 EOQ and lot-sizing in RL networks

A popular approach to inventory management in networks is lot sizing. Lot sizing simplifies decision making into determining the size and number of batches of components procurement, recovery, production, etc., for a particular time horizon. One of the limitation of lot sizing methods is treating variations in demand and return flows which might require readjusting its demand and return parameters (Teunter, 2004).

Schrady (1967) provides the earliest work regarding lot sizing in a production/ recovery environment which considers constant demand and return, single production and recovery cycles and zero lead time and uses Economic Order Quantity (EOQ) to determine optimal order size. As an extension, Teunter (2001) uses EOQ models for production/recovery lot sizing, also with zero lead time, but with the disposal done at the beginning of a production cycle. Different holding costs for manufactured and recovered items are considered. They argue that either one production lot with multiple recovery lots or one recovery lot with multiple production lots, referred to from here on as *'two main classes'*, are usually optimal and hence, only those policies are worth considering.

Teunter (2004) proposes closed form formulas to determine optimal parameters for the two main classes in a production/recovery environment. In a similar work, Dobos and Richter (2004) investigate lot-sizing decisions in a production/recycling model with a disposal option, arbitrary number of lots, stationary demand and return rates. It is concluded that when disposal is considered as a constant percentage of return, the extreme policies (all disposal or all recycling) dominate the mixed policies. Also, Teunter et al. (2006) investigate non-stationary demand/return problem, known as dynamic lot-sizing problem, and provide a polynomial time dynamic programming solution for joint setup cost case, where a single line is shared between production and recovery. They also propose heuristics for both joint setup cost case and separate setup cost case, where production and recovery are separate. Following the Teunter et al. work, Konstantaras and Skouri (2010) determine sufficient conditions for optimality of the two main classes, with either single production or recovery lot but with variable number of total lots, with and without shortage. Also, Konstantaras et al. (2010) add inspection and sorting activities and possibility of selling part of the recoverable items in a secondary market, at a lower cost. Optimality of the results are proven, for both the two main classes of single production or recovery lot.

Mitra (2012) develops a two-echelon model for a closed-loop supply chain with correlated demand and return and discusses the relationship between the rate of product return with network's cost. Both deterministic and stochastic cases are analysed and it is concluded that although a higher return rate and higher correlation with demand reduces the net demand, i.e. the demand that should be satisfied through new products, it might not necessarily reduce the costs.

Other authors used EOQ based and lot sizing methods in inventory control and production planning in RL networks while considering quality of returns. This literature will be considered in more details in Section 2.4.3.

2.3.2 Heuristic policies for RL networks

One of the common approaches to the RL inventory control and production planning problem, is to use certain policies to control the material flow. This approach has the benefit of being simple and this advantage has made these policies the preferred choice of managers in practice (Inderfurth and Laan, 2001). But, these policies are often either introduced as heuristics only or their optimality is only proven within certain limiting and/or unrealistic assumptions.

Laan and Salomon (1997) consider push and pull policies to coordinate production, recovery and disposal in a stochastic inventory network and investigate the economic value of planned disposal in such network. van der Laan et al. (1999) examine the case when both manufacturing and remanufacturing can happen simultaneously with push and pull policies.

Inderfurth and Laan (2001) propose using a four parameter policy with a modified inventory position to control inventory in a stochastic manufacturing/remanufacturing environment with disposal setting. Parameters include the size of the safety stock, lot size of production, lot size of remanufacturing and maximum inventory position. Inventory position is modified by including extra past orders, compared to the traditional definition of the inventory position, to compensate for the effects of different lead times on the policy.

Mahadevan et al. (2003) provide heuristics to determine parameters for periodic review, push inventory policies for product recovery with uncertain demand. Also, Nenes et al. (2010) investigate using custom policies to improve a network with stochastic quantity and quality of returns which we will discuss in Section 2.4.3.

Furthermore, Tang and Naim (2004) apply control engineering approach to integrated RL networks and analyse the effect of information transparency on system's robustness by comparing three types of systems which differ in the amount of inventory and production pipeline information is used.

2.3.3 Optimisation models of RL networks

Optimisation methods such as *Linear Programming (LP)* and *Mixed Integer Programming (MIP)* are mathematically proven techniques to be used to find optimal solutions of mathematical models formulated in a specific way. Compared to EOQ and lot sizing based models, optimisation models are more flexible; for example, in terms of analysing multiple periods, multiple products, etc. Also, the solution of an optimisation model is known to be optimal, unlike the case of heuristic policies, where there is no guarantee about the quality of their results. Unfortunately, implementing optimisation methods require more knowledge of problems and considerably better information collection, which may make them impractical. Hence, they are unlikely to be used in practice (Teunter et al., 2006).

A variety of optimisation techniques are used in the literature. For example, Kim et al. (2006) propose a general MIP model for integrated logistics networks which have the option of ordering components from suppliers or remanufacturing them either through a subcontractor or in-house. Model is multi-periodic, multi-product and multi-component. Wei et al. (2011) apply a robust optimisation technique for inventory control and production planning in an integrated RL network with uncertain demand and return. Also, Shi et al. (2011) examine a production planning problem with multiple products and uncertain demand and return in a closed loop network. A non-linear programming model is proposed and a Lagrangian relaxation based approach is utilised to solve the problem. Furthermore, Guo et al. (2014) propose a dynamic programming model for a network with two recovery routes: disassembly and repair where each route satisfied a separate demand. Uncertainty was taken into account by using stochastic parameters. However, quality of returns, variations in demand and return, setup costs and lead times were not considered.

Fuzzy optimisation has been used extensively in traditional inventory control and production planning problems. For example, Torabi and Hassini (2008) consider *Supply Chain Master Planning (SCMP)*, integrating procurement, production and distribution planning, using fuzzy optimisation techniques. Also, Wang and Fang (2001) consider *Aggregate Production Planning (APP)* with fuzzy demand and two objectives, including maximising profit and minimising changes in workforce and propose an interactive Fuzzy Linear Programming (FLP) model. In a similar work, Wang and Liang (2004) examine APP problem with multiple products, three fuzzy objectives (minimising total production costs, minimising carrying and back ordering costs and minimising rate of change in labour levels) and fuzzy variables, including demand forecast, labour levels and machine capacity. In a RL environment, Cardoso et al. (2012) consider uncertainty in demand in an integrated RL network with both production/recovery and distribution decisions using MILP formulation and scenario tree approach. Each node in the scenario tree represents a state of the network and each arc, that has an associated probability, stands for a situation that a new piece of information becomes available and one state evolves into another one.

Please note that the case of inventory control and production planning with quality considerations have also been examined in the literature. We will discuss these papers in Section 2.4.3.

2.4 Quality of returned products

Quality of returns plays an important role in the product recovery. Unlike the forward route where the quality is expected to be almost always of a certain standard [1], such strict quality conditions are not feasible at all in the return flow. The mere fact that a product is being returned (i.e., rejected by one party) makes it obvious that we cannot expect a homogeneous quality. Arguably, this is the most important aspect that separates the modelling of return flow from modelling of the forward flow. For this reason, we have treated quality of returns as the centre point of this research, too. In this section, we will discuss the literature which consider the quality of returns in more details.

Table 2.1 summarises the papers relevant to the quality of returns. Several key features of these papers have been shown in the table to give a clear picture of the variety of research carried out in this area. "Recovery yield" and "Quality levels" are the two main categories of modelling quality of returns which will be discussed in more details in the following sections. "Forward logistics" column show that the paper has considered the integration with forward route while "Inventories" show that the holding costs and/or the remaining stock at the end of the period is modelled.

Other columns represent a logistics model or feature in the model that have been considered. "Distribution" refers to distribution and transportation of products to/from point of sale. "Network Design" considers location allocation models for production, distribution and/or recovery facilities. "Value of Information" is about the study of effects of presence or absence of knowledge about certain parameters of the RL networks. "Product Design" refer to models that include product design decisions. Also, "Secondary Market" pinpoints the models that have consider the option of selling returned products in secondary markets with lower quality requirements.

Furthermore, "Fixed Cost" refers to the setup costs and/or ordering costs which

[1] For example, in the widely used six sigma method, less than 0.002 defective parts in a million is expected.

are independent from the quantity of order, production or recovery and might have been included in the respective model. "Period" is either "S" (for Single) or "M" (for Multiple) which shows the way demand/return data is handled by the model. Hence, lot sizing models which assume a constant demand and return are usually considered single periodic. Please note that a single periodic model can also be applied to a multi-periodic scenario by using average demand/return quantities but the solution is unlikely to be optimal. "Part Types" displays the number of parts or components of a product modelled in the network and is also either "S" (for Single) or "M" (for Multiple). "Lead time" indicates that if the model has used lead time information. The three columns for "Demand", including "Uncertain", "Non-stationary" and "Unknown/Unlimited" exhibit the main characteristic of the modelled demand. "Stochastic Uncertainty" and "Fuzzy Uncertainty" point out the type of uncertainty modelling used in the paper. Also, "Types of Recovery Routes" indicates the number of different recovery routes modelled in the paper. Finally, the "Methods" column gives a brief description of the mathematical method used to solve the problem. Some of the used methods include Analytical (including lot sizing), MILP (Mixed Integer Linear Programming), MINLP (Mixed Integer Non Linear Programming), MDP (Markov Decisions Processes), SMIP (Stochastic Mixed Integer Programming), Queuing Theory and custom (heuristic) policies.

In this section, we focus on the literatures which consider the quality of returned products only. First we examine the two main ways of modelling the quality of returns, yield rate and quality levels. Then, application of quality of returns in inventory control/production planning problems are considered. Also, quality in disassembly tree and product design problems, relationship of quality with purchasing price, how quality affects product supply in secondary markets and quality in network design and location allocation problems are all discussed. At the end, some case studies relevant to the quality of returns are mentioned.

Table 2.1 Papers related to quality of returns

Paper	Quality		Model Features														Demand			Uncertainty		Types of Recovery Routes	Methods
	Recovery Yield	Quality Levels	Forward Logsitics	Inventories	Case Study?	Distribution	Network Deisgn	Value of Information	Disassembly	Purchasing	Product Design	Fixed Cost	Secondary Market	Period	Part Types	Lead Time	Uncertain	Non-stationary	Unknown/Unlimited	Stochastic	Fuzzy		
OUR WORK	-	X	X	X	-	-	-	-	-	-	-	X	-	M[2]	S	X	X	X	-	-	X	2	FMILP+Fuzzy Control
Krikke et al. (1998a)	-	X	-	-	X	-	-	-	X	-	-	-	-	S	M	-	-	-	X	X	-	1	SDP
Guide and Wassenhove (2001)	-	X	-	-	X[3]	-	-	-	-	X	-	-	-	-	-	-	-	-	X	-	-	1	-
Souza et al. (2002)	-	X	-	-	X	-	-	-	-	-	-	-	X	S	S	X	X	-	-	X	-	1	Q.Theory + Despatching Policies
Guide et al. (2003)	-	X	-	-	X	-	-	-	-	X	-	-	-	S	-	-	-	-	X	-	-	1	Analytical
Ferrer (2003)	X	-	X	X	-	-	-	X	-	-	-	X	-	S	S	-	-	-	-	X	-	1	Analytical
Ferrer and Ketzenberg (2004)	X	-	X	X	-	-	-	X	-	-	-	-	-	M	M	X	-	-	-	X	-	1	MDPs
Aras et al. (2004)	-	X	X	X	-	-	-	-	-	-	-	-	-	M	S	X	X	-	-	X	-	1	Markov Chains
Guide et al. (2005)	X	-	-	X	X	-	-	-	-	-	-	-	X	M	M	X	-	-	X	-	-	2	LP
Inderfurth (2005)	X	-	X	X	-	-	-	-	-	-	-	-	-	S	-	-	X	-	X	X	-	1	Analytical
Dobos and Richter (2006)	X	-	X	X	-	-	-	-	-	-	-	X	-	S	-	-	-	-	-	-	-	1	Analytical
Jayaraman (2006)	-	X	X	X	X	-	-	-	-	-	X	-	-	M	M	-	-	X	-	-	-	2	LP
Galbreth and Blackburn (2006)	-	X	-	-	-	-	-	-	-	-	-	-	-	S	S	-	X[4]	-	-	X	-	1	Analytical
Teunter (2006)	-	X	-	-	-	-	-	-	X	-	-	-	-	S	M	-	-	-	X	X	-	1	SDP
Zikopoulos and Tagaras (2007)	X	-	-	-	-	X	-	-	-	-	-	-	-	S	-	-	X	-	-	X	-	1	Analytical
Mitra (2007)	-	X	-	X	X	-	-	-	-	-	-	-	X	S	S	-	-	-	-	-	-	2	Analytical

Continued on next page

[2] S: Single and M: Multiple
[3] Case study only
[4] Both deterministic and non deterministic cases are analysed

Table 2.1 – *Continued from previous page*

Zikopoulos and Tagaras (2008)	X	-	-	X	-	-	-	-	-	-	-	X	-	S+M	S	-	X	-	-	X	-	1	Analytical
Tagaras and Zikopoulos (2008)	X	-	-	X	-	-	-	-	-	-	-	X	-	M	S	X	X	-	-	X	-	1	Analytical
Aras and Aksen (2008)	-	X	-	-	-	-	X	-	-	X	-	X	-	S	-	-	-	-	X	-	-	1	MINLP + Tabu Search
Rubio and Corominas (2008)	X	-	X	-	-	-	-	-	-	-	-	X	-	S	S	-	-	-	-	-	-	1	Analytical
Mukhopadhyay and Ma (2009)	X	-	X	X	-	-	-	X	-	-	-	-	-	M	S	-	X	-	-	X	-	1	Analytial
Ketzenberg et al. (2009)	X	-	X	X	-	-	-	X	-	-	-	-	-	M	S	-	X	-	-	X	-	1	Analytical + MDPs
Pishvaee et al. (2009)	X	-	X	X	-	X	X	-	-	-	-	X	-	S	S	-	X	-	-	X	X	1	SMIP
Behret and Korugan (2009)	-	X	X	X	-	-	-	X	-	X[5]	-	-	-	M	S	X	X	-	-	X	-	1	Simulation + Combination of Tabu & Scatter Search
Xanthopoulos and Iakovou (2009)	X	-	X	X	X	-	-	-	X	X[6]	-	X	-	M	M	X	-	X	-	X	-	1	GP + MILP + Simulation
Jaber and El Saadany (2009)	X	-	X	X	-	-	-	-	-	-	-	X	X	S	S	-	-	-	-	-	-	1	Analytical
Nenes et al. (2010)	X	-	X	X	X	-	-	-	-	-	-	-	-	M	S	-	X	-	-	X	-	1	Custom Policies
Nakashima and Gupta (2010)	-	X	X	X	-	-	-	-	-	X[7]	-	-	-	M	S	X	X	-	-	X	-	1	MDPs
El Saadany and Jaber (2010)	-	X[8]	X	X	-	-	-	-	-	X	-	X	-	M	S	-	-	-	-	-	-	1	Analytical + DP
Lieckens and Vandaele (2012)	X	-	X	X	-	X	X	-	-	-	-	X	X	S	S	X	-	-	-	X	-	1	MINLP + Differential Evolution (DE)
Mahapatra et al. (2012)	-	X	X	X	X	-	-	-	-	-	X	X[9]	X	M	S	-	-	X	-	-	-	1	MILP
Das and Chowdhury (2012)	-	X	X	-	-	-	-	-	-	-	X	X	X	S	M	-	-	-	-	-	-	1	MIP

Continued on next page

[5] Only as parameters
[6] Only as parameters
[7] Only as parameters
[8] Continuous levels
[9] Only for capacity readjustments

Table 2.1 – *Continued from previous page*

Nenes and Nikolaidis (2012)	-	X	-	X	-	-	-	-	-	-	-	X	-	M	S	-	-	X	-	-	-	1	MILP
Özkır and Başlıgıl (2012)	X	-	X	X	-	X	X	-	-	-	-	X	-	M	M	-	-	X	-	X	-	2	MILP
Pokharel and Liang (2012)	-	X	-	X	-	-	-	-	-	X	-	-	-	S	S	-	-	-	-	X	-	1	Analytical
Yoo et al. (2012)	X	-	-	X	-	-	-	X	-	-	-	X	-	S	S	-	-	-	-	-	-	2	Analytical
Marshall (2012)	-	X	X	X	-	-	-	-	-	-	-	X	X	M	S	X	X	-	-	X	-	2	Analytical + MDPs
Korugan et al. (2013)	X	-	X	X	-	-	-	-	-	-	-	-	-	M	S	-	-	-	X	X	-	1	MDPs
Jin et al. (2013)	-	X	X	X	-	-	-	-	-	-	-	-	X	S	M	-	X	-	-	X	-	1	MDPs + Custom Policies
Xiong et al. (2013)	-	X[10]	-	X	-	-	-	-	-	X	-	-	-	M	S	-	-	-	-	X	-	1	MDPs
Das and Dutta (2013)	X	-	X	X	-	-	-	-	-	-	-	-	X	M	M	-	-	-[11]	-	-	-	2	System Dynamics

[10] Continious levels

[11] The effects of product exchange on demand are considered

2.4.1 Modelling quality

Various approaches have been proposed to deal with the quality of returned products and inherent uncertainty. One of the common approaches is to model the quality by a probabilistic *yield rate* which specifies the probability of a single product to be successfully recovered. In this approach, only two outcomes are considered: either a returned product is recoverable or it is not (Dobos and Richter, 2006).

Another proposed approach has been to assume a set of predefined *quality levels* that have different acquisition costs and remanufacturing costs. Depending on these parameters, a particular quality level might be more or less desirable for certain recovery activities. Also, more general approaches which assume a continuous quality range has been studied as well; although these studies are few and far between and further investigation of this issue is necessary.

In this section, modelling quality as yield rate and also quality levels are investigated separately and their applications to different problems are analysed.

Yield rate

Quality of returned products has often been simplified to a yield rate, which determines the percentage of returned products suitable for recovery. This simplification has the advantage of dealing with a single unit remanufacturing cost, regardless of its quality, and hence, helps simplify the mathematical model.

Dobos and Richter (2006) analysed the case of lot-sizing in a production and recovery environment with two options: either to buyback all returned items from the supplier and use the ones which are recoverable or to buyback the recoverable products only. It has been found that for the total cost, outsourcing the inspection is optimal. Inderfurth (2005) developed an optimisation model to minimise cost of an integrated RL network with stationary demand, equal lead times and stochastic uncertainty in both return quantity and quality. Variables of the model include number of items that are produced, remanufactured and disposed and also the inventory positions of remanufacturable and final products. Zikopoulos and Tagaras (2007) considered a case of two alternative collection points with different, but probabilistically correlated yield rates considering a single time period. These alternative collection points can represent alternative return streams such as a few customers with large quantities of return versus a large number of customers with few returns. The conditions where using a single collection point is optimal are identified. Additionally, they suggested multiple quality levels and multiple periods as interesting possible future extensions. Also, Rubio and Corominas (2008) analytically determine economically optimal policies for integrated recovery and production networks with or without variable capacity setup costs. Inventory costs are not considered and quality of returns is modelled yield rates. Furthermore, Lieckens and Vandaele (2012) assume a fixed percentage of outputs in each layer of the network is of acceptable quality to be used in the following layer, while the rest is disposed. Layers represent different stages of the reverse or forward supply chain such as retailer, inspection, remanufacturing and distribution where each layer can consist of multiple facilities. In addition, Nenes et al. (2010) compared several alternative policies for production planning in the presence of returned products with either as-good-as-new or remanufacturable quality.

Zikopoulos and Tagaras (2008) analyse the economic attractiveness of sorting returned products to determine quality, when there is an uncertainty such as error in the perceived quality in the acquired information. It has been concluded that sorting is economical when the cost of sorting, disposal, transportation and also average quality are low and disassembly cost and sorting accuracy is high. Furthermore, Tagaras and Zikopoulos (2008) extend Zikopoulos and Tagaras (2008) to deterministic yield and examine various sorting options including no sorting, central sorting (in a remanufacturing facility) or local sorting (in collection sites). Also, Korugan et al. (2013) use Markov chains to examine operational failures along quality failures in a two machine, one buffer setting in a shared production/remanufacturing environment.

Ferrer (2003) examines an optimal lot sizing model in different scenarios of managerial decisions to address the lack of proper yield information. The scenarios include a situation where yield information comes very late, where yield is realised just before the disassembly by the operator, where lead time of the supplier is short and, if necessary, new products can fill the shortage and finally, where the actual yield is known in advance. Also, Ferrer and Ketzenberg (2004) compare the trade-off between the value of returned products yield information and lead-time of a 3rd party supplier of new products. Specifically, the influence of yield rate realization, i.e. when the information about quality of returns are available, is studied. Through four models, all cases of a long or short lead time and an early or late realization of yield rate are compared and concluded that, depending on the number of parts in the product, it is economically worthwhile to invest on early realization of yield information. Similarly, Mukhopadhyay and Ma (2009) investigate yield rate of returned products in relation with production/recovery activities. Different scenarios were investigated regarding when and how much information about yield rate is available. A deterministic yield rate model, where yield is known is advanced is compared with random yield rate model where yield rate is uncertain. In the random case, both possibilities of short or long delivery lead time of new parts, where yield rate is realised early enough or too late to order replacement new parts, are studied. Additionally, Ketzenberg et al. (2009) analyse the value of information in a production/recovery network with stochastic demand, return quantity and quality (yield rate) and conclude that all three types of information (about demand, return quantity and yield) are useful and the conditions that each is the most valuable are identified. Also, Pishvaee et al. (2009) consider demand, return quantity and also quality as yield rate to be uncertain and use scenario based optimisation to tackle the uncertainties.

Quality levels

Quality levels, also known as *types*, *categories* or *classifications*, are predefined, disjoint and ordinal quality classes of returned products which are used to distinguish the amount of work and cost needed to recover a particular returned product. In contrast with yield rates, application of quality levels allows for a fine grained analysis of return quality and has been considered in the literature.

Souza et al. (2002) use queuing theory to find the optimal production plan for recovery of products with three quality levels, including superior, average and inferior. The firm considered has three separate recovery facilities, each specialised for remanufacturing of one particular quality level but also able to remanufacture other levels at extra cost and time. The company also has the option of selling the returned product as is, at lower price.

Aras et al. (2004) use a Markov chain based model to show the advantage of prioritising returned products for recovery based on their quality. It is shown that in certain circumstances, such as low demand, relatively high return, low quality of returns and high difference between quality of returns, it is more cost effective to prioritise than to recover all returned products without discrimination. Similarly, Nakashima and Gupta (2010) use Markov decision process with stochastic demand to analyse the effects of prioritising either of the two available classes of returned products, with different lead time, remanufacturing, acquisition and holding costs. Behret and Korugan (2009) analyse an integrated manufacturing/remanufacturing system in which returned products are inspected and then classified into three quality levels (bad, average and good), where each level can be recovered using its own recovery facility, with respectable recovery cost and time, or disposed. Using simulation, it has been concluded that quality based classification can provide a cost improvement for high quantities of return. Additionally, Das and Dutta (2013) used system dynamics in an integrated reverse network with three recovery options: repair, remanufacturing and recycling. Quality of returns was modelled as fixed percentages of products which could go to each recovery route. However, simulation of network behaviour using a custom policy without setup costs was the focus of this work. Also, Marshall (2012) proposes a deterministic lot-sizing model and a stochastic model to analyse integrated manufacturing/remanufacturing networks with the possibility of recovery for high quality returns and low quality returns.

Jayaraman (2006) propose a linear programming model for production planning in a closed-loop RL network with predefined quality levels and zero lead times. In a similar line, Mahapatra et al. (2012) analyse the problem of heterogeneity of returned products quality in integrated manufacturing-remanufacturing networks and propose a MIP model to find optimal production plan for a multi-product and multi-period case, with capacity constraints and zero lead times. The case of an office equipment manufacturer is also discussed.

Alternatively, Galbreth and Blackburn (2006) explore the possibility of using a threshold quality level to determine products which are acceptable for the recovery activity. Remanufacturing costs is assumed to be a continuous function of quality and both the acceptable quality threshold and the acceptable return rate are determined in such a way as to minimise procurement and remanufacturing costs in a single period setting.

Aras and Aksen (2008) focus on facility location problem in the presence of different quality levels. The remanufacturing firm considered provides different incentives for each quality level returned. Customer's willingness to return depends on the incentive as well as the distance to the recovery centre. Other authors have also used quality levels in determining the purchasing price (El Saadany and Jaber, 2010; Guide and Wassenhove, 2001; Guide et al., 2003; Pokharel and Liang, 2012; Xiong et al., 2013). We will discuss these works in Section 2.4.5.

A few areas exist for improvement in the use of quality levels. For example, uncertainty modelling has been limited mainly to stochastic methods while fuzzy applications can bring more insight, especially when the definition of quality levels are inherently vague. Also, use of quality thresholds as decision variables can provide a more practical model of the sorting process and needs more attention. Additionally, suitability of different recovery routes for returns with different quality levels is an important area that should be investigated further. Furthermore, classification error is an important issue relevant to quality levels which should be examined further.

2.4.2 Modelling uncertainty

Uncertainty in supply chains is usually modelled using either stochastic (probabilistic) methods or fuzzy (possibilistic) methods. In this section, some of the applications of each of these approaches in supply chain modelling, particularly in RL which concerns quality of returns, is discussed.

Stochastic models

Stochastic models have been widely applied to RL networks for the purpose of modelling uncertainty. Some of these applications, especially those which consider quality of returns, are discussed here.

Regarding quality of returns, as mentioned earlier, many authors have used yield rate (a simple probability) to model quality. These models are arguably classed as the stochastic models. However, in this section, we are more interested in the part of literature which explicitly considers the stochastic nature of the RL networks and not just a yield rate.

Markov Decisions Processes (MDPs) are among popular methods to model the uncertainty in the RL networks. Particularly, MDPs have been used to model quality failures with yield rates (Ferrer and Ketzenberg, 2004; Ketzenberg et al., 2009; Korugan et al., 2013) where stochastic failures in recovery can happen because of quality issues, and quality levels (Jin et al., 2013; Nakashima and Gupta, 2010; Xiong et al.,

2013) where they are determined stochastically in the model.

Scenario based stochastic optimisation models have been used in integrated RL network design problem. Pishvaee et al. (2009) consider quality of returns in network design, which we will discuss in more details later. Also, Lieckens and Vandaele (2007) examine stochastic lead times in the recovery networks design problem with a single product and return locations, recovery facilities and customer zones; the resulting MINLP model is solved using differential evolution technique. Additionally, Kara and Onut (2010) use two stage stochastic mixed integer and robust programming to find optimal recycling centre locations and network flows while analysing a case study in paper recycling. Additionally, Salema et al. (2007) introduce a multi-product, generic network design model with capacity limits and scenario based uncertainty in demand and return flows.

Simulation modelling of RL networks is another popular venue of applications. Among many, Kara et al. (2007) examine an RL network with multiple collection points, disassembly centre, multiple recycling and disposal options and remanufacturing plant using simulation. As mentioned, Behret and Korugan (2009) simulate integrated RL networks with quality classification and uncertainty in return timing, quantity and quality. Furthermore, Gharbi et al. (2008) provide a control policy for remanufacturing rate in a closed loop network with both planned (end of life) demand and stochastic unplanned demand.

Application of queuing theory has also been proposed in the literature. As mentioned, Souza et al. (2002) consider a recovery network with stochastic return quality. Additionally, Vahdani et al. (2012) propose a robust queuing model for the design of an integrated RL network with uncertainty in setup costs, transportation costs, production rate, operational costs and storage capacities. Although quality of returns is not considered. The proposed method is a combination of robust optimisation, queuing theory and fuzzy multi-objective programming.

Stochastic demand and return parameters are used analytically by many authors.

For example, Inderfurth and Laan (2001) analyse the effect of lead time in optimal policy for integrated inventory control with stochastic demand and return. As mentioned, Nenes et al. (2010) investigate the inventory control with recovery problem when demand and return quantity and quality are of stochastic nature. (Inderfurth and Langella, 2005) propose heuristics for disassembly tree problem where the yield is stochastic.

Fuzzy logic based models

While fuzzy logic has been applied extensively to many aspects of supply chain, surprisingly its applications to modelling the RL networks which concern quality of returns are quite sparse. In fact, apart from our work presented in this thesis and Pishvaee et al. (2009), which only use fuzzy clustering to limit the total number of scenarios, to the best of our knowledge, there has not been any other applications. In this section, some of the notable and relevant research in supply chain using fuzzy logic are considered, although it is by no mean a complete survey. For a comprehensive review of soft computing techniques' applications in the supply chain domain, we refer interested readers to Ko et al. (2010). Please note that some of the relevant concepts in fuzzy logic will be discussed in the next chapter.

Petrovic et al. (2008) use fuzzy sets to model customer demand and inventory positions in a Distribution Supply Chain (DSC) including several retailers and a warehouse. The solution is found by decomposing the problem of determining parameters for inventory control policies into several simpler sub problems, solving them and then using an iterative coordinating procedure to determine satisfactory results for the overall chain by introducing new constraints to the sub problems. Miller and John (2010) use genetic algorithm to find suitable inventory levels in a multi-echelon supply chain where some of the parameters, including demand, inventory level, transportation distances and costs, stock out level and cost, carry over and holding cost, are described as Interval Type-2 fuzzy numbers. Furthermore, Baltacioğlu et al. (2011) propose Fuzzy Wagner Whithin algorithm for inventory control, applied to stocking of Turkish Armed Forces' Class I products.

Qin and Ji (2010) utilise soft computing techniques, such as fuzzy simulation and genetic algorithm, to determine the location of collection centres in a product recovery network using fuzzy sets to model return volume, setup and unit cost of collection centres.

Torabi and Hassini (2008) propose a Multi Objective Possibilistic Mixed Integer Linear Programming (MOPMILP) model to assist planning for procurement, production and distribution, and then provide a novel interactive approach to find a solution for this model. Also, in a different, but similarly interesting study, Peidro et al. (2009) use the fuzzy triangular numbers to represents uncertainty in demand quantity in the supply chain (SC) and propose a fuzzy mixed integer optimisation model to minimise the SC costs.

Wadhwa et al. (2009) use *Fuzzy TOPSIS* Multi Criteria Decision Making (MCDM) technique to determine the most suitable recovery option incorporating many factors including economical, ecological, market, quality and legislative impact. Fuzzy TOPSIS is also applied to a supplier selection problem (Awasthi et al., 2010), while Fuzzy Analytical Hierarchical Process (AHP) is applied to selection of 3rd party reverse logistics providers (Efendigil et al., 2008).

Fuzzy is very useful in absence of historical data and lack of precise distribution functions, where stochastic and probabilistic methods cannot be applicable. This is usually the case when dealing with the quantity and quality of returns for which arguably more is unknown than known. For such problems, it is for example possible to integrate managerial opinion using fuzzy linguistic variables.

As mentioned earlier, fuzzy logic applications to RL with quality of returns seems non-existent. This is in contrast with the fuzzy's main benefits that are generally applicable to these problems.

2.4.3 Quality in inventory control and production planning

Nenes et al. (2010) discuss the case study of Dutch Railways' relays procurement from either as-good-as-new returns, remanufacturing or 3rd party suppliers while considering quality of returns. Both demand quantity and return quantity and quality are assumed to be stochastic and several policies were compared. Currently implemented policy of using safety stock is found to be suboptimal compared to proposed alternatives.

As mentioned, Inderfurth (2005) and Dobos and Richter (2006) examine joint lot sizing decisions for both remanufacturing and manufacturing systems with yield information. Additionally, Zikopoulos and Tagaras (2008) and Tagaras and Zikopoulos (2008) consider a recovery planning problem with stochastic return quantity and quality for a two level recovery network. They compare the option of sorting or not sorting the return before recovery. While Zikopoulos and Tagaras (2008) consider networks compromised of collection site and remanufacturing facility in a single periodic setting, Tagaras and Zikopoulos (2008) assume networks with a central remanufacturing facility and a number of collection facilities. Also, Ferrer and Ketzenberg (2004), Ferrer (2003), Mukhopadhyay and Ma (2009) and Ketzenberg et al. (2009) investigate the value of yield information in multi-periodic production/recovery systems and how that can affect the optimal decision making for inventories, forward procurement and recovery. Although, Ferrer (2003) is limited to the single period case only. In a more recent study, Yoo et al. (2012) consider jointly value of information with lot sizing decisions for a single periodic production/recovery network. Two recovery options are available and the inspection process is imperfect which can be improved at a cost. Also, Marshall (2012) consider a production/recovery network with distinction between high quality returns which would be repaired and low quality return that can be disassembled into components. Four different models are introduced, including a deterministic lot sizing model with single market, a discrete-time MDP model also with a single market and a discrete-time and a continuous time MDP models with a secondary market.

As mentioned before, other researchers examine models with multiple quality levels (Aras et al., 2004; Behret and Korugan, 2009; Galbreth and Blackburn, 2006; Nakashima and Gupta, 2010; Souza et al., 2002). All of these models have considered only one route of recovery with stochastic uncertainty in demand, return quantity and/or return quality. Also, in the similar line of research, Nenes and Nikolaidis (2012) propose a MILP based multi-periodic model with deterministic demand and return. Nenes and Nikolaidis (2012) assume that 3rd party collection sites have several batches of returned products available which the recovery facility may choose to acquire or ignore, while it also has the option of using a certain part of acquired batches. In their model, the quantity of products which belong to a certain quality level for each particular batch is known.

Some of the literature considers integrating inventory and production planning problems with other RL problems. For example, Das and Chowdhury (2012) utilise an MIP model for RL production planning with product design decisions which effect product recovery and quality considerations. Additionally, Xanthopoulos and Iakovou (2009) combine the decision making in the disassembly of returned products with the production planning and inventory control problem and introduce a two-phase algorithm as the solution procedure, using goal-programming in the first phase and MIP in the second phase. In the first phase, a multi criteria goal programming analysis is used to select a desirable disassembly tree, while in the second phase, a multi-product multi-period MIP model of the recovery network is proposed to maximise profit. Also, Jayaraman (2006) considers Remanufacturing Aggregate Production Planning (RAPP) and introduces an LP model to make production, remanufacturing, procurement and disposal decisions in multi-period, multi-product, multi-component and multi quality level closed loop networks. Mahapatra et al. (2012) also examine the effect of heterogeneous quality of returns and non-uniform quantity of returns on integrated RL networks using a MILP model.

Other authors examine the case of recovery network's inventory control and production planning when more than one markets are available for the products. We will discuss these papers in Section 2.4.6.

Regarding the gap in the literature, there are some areas that have not been explored or are mentioned very sparsely in the literature. One that is of interest to this research is the use of multiple recovery routes. Although some researchers have covered alternative options for the same type of recovery, for example, different remanufacturing options which can represent different recovery facilities (Behret and Korugan, 2009; Souza et al., 2002), different recovery routes are not often considered simultaneously. Different recovery routes such as repair and remanufacturing have fundamental differences which should be included in a network model and, hence, can have significant impact on results.

Among those who have considered different routes, Jayaraman (2006) considers remanufacturing and reuse. He assumes a zero lead time and no uncertainty both in return and demand. Also, Guide et al. (2005) consider a recovery only network with either in-house refurbishing or outsourced repair. They assume a deterministic demand and return and a yield rate based quality model. Similarly, Mitra (2007) consider recovery only network with remanufacturing and refurbishing, without uncertainty, with zero lead time and a single period. Marshall (2012) on the other hand, consider repair, disassembly and production. However, only a fixed definition for high and low quality returns are used and the demand and return quantities are probabilistically uncertain but stationary. Research presented in this thesis is unique as it develops a multi-period, multi quality level, multi recovery route model with different lead times, non-stationary demand and return quantity, and fuzzy uncertainty in return quantity and quality and demand quantity.

2.4.4 Quality in disassembly tree and product design

Subjects of product design for remanufacturing and disassembly tree have significant importance in decision making in RL networks. This area has been studied in literature, albeit briefly. One notable example is Ferrer (2001), which investigates the effects of design on recovery and introduces measures of 'recyclability', 'disassmblability' and 'reusability' to compare different designs.

The quality of returns in RL models clearly has an important influence on product design decisions as there are many direct relationships between the products design, the expected end-of-life product quality and the recovery process. Especially, product design determines the way the product is assembled and, henceforth, can be disassembled. The disassembly decisions, including to disassemble the product into components and disassemble components into sub-components, are collectively examined as *disassembly tree* decisions. There have been a few attempts at addressing the effect of quality of returned products and their components on the optimal decision making for disassembly tree. Notably, Krikke et al. (1998a) proposed a method to find the optimal recovery option of either disassembly, disposal or refurbishing, i.e. direct reuse in each stage of the disassembly tree based on the quality level of the product or component. Also, Teunter (2006) extends this model by adding multiple reuse options and allowing partial disassembly. Additionally, Xanthopoulos and Iakovou (2009) consider a similar model using a two phase approach which consists of goal programming and MILP but includes lead time of the disassembly and recovery processes. The uncertainty of the system is captured using a simulation approach with stochastic parameters.

Furthermore, among literature which consider quality of returns, others have analysed product design directly and measured its influence on the optimal decisions in RL networks (Das and Chowdhury, 2012; Jayaraman, 2006; Mahapatra et al., 2012). For example, Das and Chowdhury (2012) propose an MIP model for a RL network with multiple products, components and design options. Although, to the best of our knowledge, these studies are few and far between and more attention is needed to the issue of product design and disassembly tree, especially in relation with quality of returns.

2.4.5 Relationship between quality of returns and purchasing price

Effects of different purchasing prices with respect to the quality of returned products have been investigated as well. Guide and Wassenhove (2001) propose a framework based on the *Economic Value Added* concept to evaluate economic attractiveness of recovery options considering the effect of acquisition price depending on the quality of returns. As a follow up, Guide et al. (2003) examined the influence of acquisition price on the quality, quantity and timing of return. As mentioned, return is assumed to be graded in one of the predefined quality levels with different corresponding remanufacturing costs. The remanufacturer is free to decide how much to procure from each quality level, but the return rate of each level is a function of acquisition price. Profit is optimised in a single period setting, assuming demand to be a function of sale price.

Moreover, El Saadany and Jaber (2010) extended Dobos and Richter (2006) model by including the return rate as a function of purchasing price and acceptance quality level. Pokharel and Liang (2012) determine the optimal acquisition prices and quantities in a multi collection centres and one consolidation centre model where the consolidation centre is responsible for meeting the demand, avoiding surplus and consolidating return with necessary spare parts for the remanufacturing site. Recently, Xiong et al. (2013) has proposed dynamic pricing for return in a stochastic and continuous time environment using Markov decision processes.

Other researchers have included acquisition price as part of various RL problems. For example, as mentioned before, Aras and Aksen (2008) consider the return incentive available to customers as part of a network design problem where closeness to the collection centre also affects customer willingness to return. Also, different acquisition prices based on quality of returns have been modelled as part of production planning problems in RL (Behret and Korugan, 2009; Nakashima and Gupta, 2010).

For future research in this area, multi-period analysis of the relationship between acquisition price, quality and quantity has been suggested (Pokharel and Liang, 2012) and it is worth more attention. Also, a scenario in which alternative products are available with different yields and different customer willingness to return might be worth considering too. Furthermore, the inventory control and production planning models in this area often have a simplistic approach to the acquisition price e.g. only as a fixed parameter. This can be extended by integrating more detailed inventory control and production planning decisions with purchasing price decisions.

2.4.6 Secondary markets and selling prices

Difference in the quality of remanufactured products and dissimilar selling prices are also investigated in the literature. One instance is Mitra (2007) which assumes two categories of recovered products, including remanufactured and refurbished, where each product has its own availability, acquisition price and sell price. Profit is optimised by determining sell prices subject to a variable demand for each category. Jaber and El Saadany (2009) distinguish between two classes of demand, corresponding to primary market and secondary market, which can be satisfied by either manufactured or remanufactured products. Lot-sizing variables are determined by optimising total cost assuming demand in one class to be substitutable from the other class at a cost. Marshall (2012) extends the proposed single market model into a dual market model and provides an analysis using both a discrete-time and a continuous time MDP models. Furthermore, Guide et al. (2005) investigate the remanufacturing of HP electronics to be sold in secondary market.

As mentioned earlier, Lieckens and Vandaele (2012) propose a multi-layer multi routing RL network design model, where products can be sold, based on their quality, on either primary or secondary market. Souza et al. (2002) propose a production planning model in which the company has the option of recovering the product to as-good-as-new condition or to sell it as is, in the secondary market at a lower price. Additionally, Mahapatra et al. (2012) simply assume different market sale price for newly manufactured and remanufactured products while they satisfy a shared sale target.

Furthermore, some models have considered more than one quality of final products. For example, Das and Chowdhury (2012) propose an MIP model for RL network planning which allows for multiple quality levels of final products. Also, Jin et al. (2013) examine the case of product reassembly from components of different quality which can be sold in different markets, in the presence of classes of demand. Components of better quality can substitute for those of worst quality to form a product that is considered to have the quality equivalent to that of the worst of its components.

Various qualities of sales and hence, classes of demand is an interesting topic which goes hand in hand with the quality of returned products. Linking the selling prices with the acquisition prices, storage and production costs has been suggested as one of the future research directions (Mahapatra et al., 2012). Also, uncertainty in the market and selling price can also be a topic worth investigating.

2.4.7 Network design, location allocation and distribution

One of the extensively researched topics regarding integration of reverse and forward flows is network design and location allocation problem (Du and Evans, 2008; Easwaran and Üster, 2010; Lu and Bostel, 2007; Pishvaee et al., 2010). Notably, Srivastava (2008) provides a conceptual multi-product and multi-echelon recovery network design using MILP. Also, Lee et al. (2013) uses a bi-objective Mixed Integer Non-Linear Programming (MINLP) model with total shipping cost and time as objectives for dynamic distribution network design in Integrated RL for Third Party Logistics (3PLs).

Distribution and vehicle routing problem in RL has also been studied to a great degree (Lee et al., 2008; Zak, 1999; Zhuan et al., 2008). Ilgin and Gupta (2010) provide a comprehensive study. One example of this line of research is presented in Alshamrani et al. (2007). They consider a vehicle routing problem with RL, inspired by the blood distribution by American Red Cross, and propose a heuristic method to determine route and pick-up strategy.

Research into RL network design which also consider quality of returns has been carried out as well. For example, Aras and Aksen (2008) investigate a location-allocation problem for collection facilities while the consumer willingness to return is modelled as a function of proximity of the facility and the acquisition price for each quality level. Among the papers which consider quality of returns in the distribution area is Zikopoulos and Tagaras (2007) which analyse the case of multiple return collection points. Their model determines the transportation quantities from these locations to the point of recovery.

Some relevant papers have combined both strategic network design problem and tactical distribution problem. For example, as mentioned earlier, Lieckens and Vandaele (2012) analyse the effect of stochastic uncertainty, especially in lead-times, on the network design with multiple layers and quality dependant routing. Also, Pishvaee et al. (2009) use stochastic optimisation and consider many uncertain parameters including transportation costs in an integrated multi-level RL network with production, distribution, collection, recovery and disposal centres. Also, Özkır and Başlıgıl (2012) analyse a multiple recovery network design and offers a MILP model to determine the optimal locations and flows for plants, distribution centres, reverse centres and collection points.

As the subject of considering the quality of returns in network design and distribution problems is still new, these works are still insufficient and the area requires more attention to be implementable in practice (Özkır and Başlıgıl, 2012). Some suggestions for future research would be to include more sophisticated models of multiple levels and uncertainty in quality and analyse the effect of specific customer zones on the expected quality of returns, i.e. varying quality depending on a collection point.

2.4.8 Case studies

Guide et al. (2005) analyse the case study of refurbishing Hewlett-Packard (HP) electronics with a short life cycle to be sold in the secondary market. Refurbishing is either done by the Outsourced Design and Manufacturing supplier (ODM), when the recovery is 'High Touch', or in-house, when the product requires little refurbishing operations or 'Low Touch'. Analysis has been done through many tools such as LP models and final suggestions have proven to improve HP's refurbishing operations. Also, Krikke et al. (1998a) uses a TV remanufacturing case to showcase the introduced model for product recovery.

The case of ReCellular, a consumer electronic remanufacturer, has been investigated in the literature thoroughly (Guide and Wassenhove, 2001; Guide et al., 2003; Jayaraman, 2006; Souza et al., 2002). ReCellular is the largest mobile phone remanufacturer in the US and collects returned phones from mobile operators as well as third party collectors. After remanufacturing, phones are sold back either to US phone operators to be sold as part of postpaid or prepaid plans or they might even be exported to secondary markets. One interesting aspect of ReCellular's network is the grading of returned phones. Phones are assigned a nominal quality level depending on the metrics defined for each of the quality levels. These metrics involve functional, electrical and cosmetic aspects of the phone (Guide and Wassenhove, 2001).

Similar to the studies about ReCellular, Mitra (2007) examines the case of mobile remanufacturing in India with different quality, cost and availabilities of returned and remanufactured products. Also, Mahapatra et al. (2012) investigate an integrated production planning and inventory control problem for a printer tuner cartridge manufacturing and remanufacturing case. Additionally, Nenes et al. (2010) consider the case of procuring ’relays’ for Dutch Railways from recovered or new products with stochastic demand and return.

2.5 Conclusions

In this chapter, published literature in the area of RL, relevant to this research is briefly analysed. After an introduction to the discussion, various RL structures and main product recovery issues, such as recovery routes and green logistics, have been discussed. Further on, problems of inventory control and production planning, which are the focus of this research, are categorised into lot sizing models, heuristic policies and optimisation models. Furthermore, quality of returned products has been investigated in more details. Among the issues related to the quality of returns, modelling of quality, using yield rates and nominal quality levels, has been considered. Also, modelling of uncertainty, using stochastic and fuzzy approaches, is examined. Additionally, treating quality in inventory control and production planning and also network design problems are discussed. Finally, impact of quality in disassembly tree decisions and product design, consideration of secondary markets and relationship of quality with purchasing price, and case studies in this area are among the other issues investigated.

Throughout this review we identified a few gaps in the literature. Importantly, the relationship between quality of returns and multiple recovery routes is very sparsely discussed in the literature. Further on, fuzzy logic has not been applied in inventory control and production planning problems in RL networks. Although, fuzzy logic can be particularly useful for uncertainty modelling in RL networks as precise information, especially regarding the returns, are not usually available for them. This research will investigate these issues by using fuzzy logic to model uncertainties in demand and return of RL networks, considering an inventory control and production planning problem with different recovery routes.

Chapter 3

Background of Relevant Methods

3.1 Introduction

Throughout the thesis, several concepts from Operational Research and Computational Intelligence are used. This chapter will provide a description of these concepts and methods to familiarise the reader with the area. Particularly, fuzzy logic and fuzzy arithmetic and their applications in control, optimisation and genetic fuzzy methods are influential to this research and are discussed in this chapter. Also, optimisation methods such as linear programming and mixed integer programming are briefly discussed. Additionally, genetic algorithm and its multi objective variants, are also the other relevant area which is considered.

This chapter is arranged as follows. First, basic concepts of fuzzy logic are presented in Section 3.2, followed by a description of fuzzy arithmetic and fuzzy control. In Section 3.3 the basic methods of linear and integer optimisation are examined. Section 3.4 is dedicated to optimisation models which consider fuzzy constraints and objectives, known as fuzzy optimisation. Next, Section 3.5 provides a brief introduction to genetic algorithm and multi objective genetic algorithm. Genetic fuzzy methods are introduced in Section 3.6 which are used in automatic learning of fuzzy rules. Finally, Section 3.7 provides a brief summary of methods and a discussion about their application in the research.

3.2 Fuzzy logic

Fuzzy logic is an extension of the traditional logic that addresses the uncertainty and imprecision which exist in real world but is not considered in the traditional logic. While in the traditional logic every proposition has either a zero (false) or one (true) value, in fuzzy logic any value between zero and one is acceptable as the *degree of truth* of a particular proposition. These degrees of truth represent ambiguity and vagueness in real world information (Pedrycz and Gomide, 1998). As an example, fuzzy logic can provide a more realistic model of the truth by describing propositions to be “half true”, “nearly false” or “quite true” as opposed to a crisp and binary description of either true or false.

Also, based on the same principle, fuzzy sets are extensions of classical sets in which the elements can have any *membership degree* between zero and one, as opposed to the classical set theory in which the element can either be a member (with membership degree of one) or a non-member (with membership degree of zero). The function that assigns membership degrees to elements in the set’s domain is referred to as the *membership function*. For example, when dealing with propositions such as “temperature is high” or “glucose level is low”, classical set theory provides little flexibility in definition of vague terms such as *low*, *medium* and *high*. In the temperature example, using the classical set theory, each term should be defined as a range of temperatures that belong to the particular term; as a partial membership is not possible in the classical set theory, there is no choice other than to use some thresholds to define each term, e.g. any temperature below 5°C is low. But in reality there is no particular “hard edge” between these terms and any number associated to be the definition will be artificial and in contradiction to the common sense. The concept of a partial membership, which is the core of fuzzy sets theory, helps in addressing this problem by providing soft edges in the definition of the sets; e.g. any temperature below 5°C is definitely low, but the degree of membership of temperatures above 5°C decreases gradually from one until it reaches zero for 15°C, as can be seen in Figure 3.1.

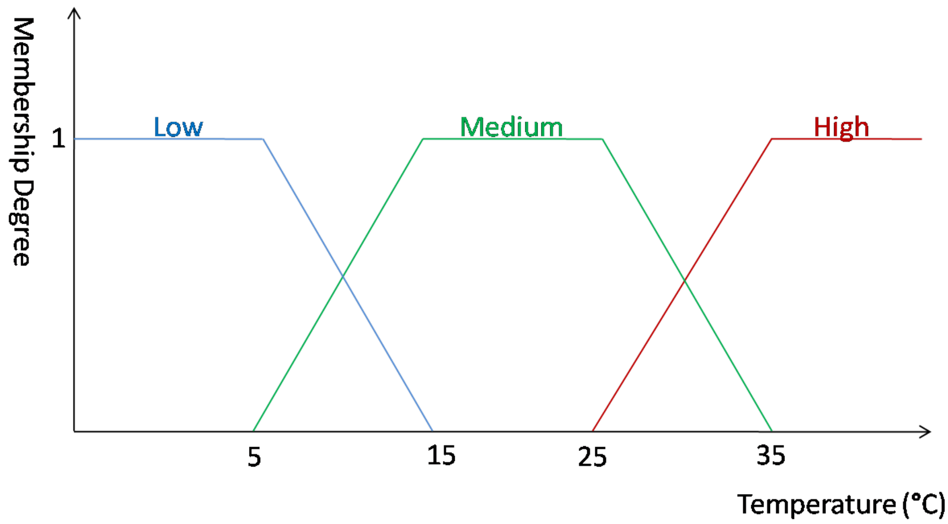

Figure 3.1 Membership functions of fuzzy sets Low, Medium and High with regard to temperature.

The definition of fuzzy sets gives rise to the concept of *fuzzy linguistic variables*. In conventional mathematics, variables are typically numeric, but in fuzzy logic, variables can get linguistic terms as values; these terms describe vague concepts such as *cold*, *mild* and *hot*. This proves to be particularly valuable when a human expert needs to understand and define a problem model since humans are generally more familiar with linguistic descriptions than with numerical values.

Fuzzy logic has various applications in control, data clustering and classification, image processing, arithmetic, optimisation and so on. In this section, fuzzy arithmetic and fuzzy control will be discussed further. Fuzzy optimisation will also be introduced later.

3.2.1 Fuzzy arithmetic

In real life applications, very often we have to deal with imprecise quantities. One such example relevant to this research would be the demand and return quantities which are often impossible to be determined precisely. Using the concepts of fuzzy numbers, *fuzzy arithmetic* provides a solution to model the uncertainty that exists in such quantities and to carry on calculations using this imprecise data.

Using the *fuzzy extension principle*, it is possible to define different types of fuzzy numbers and arithmetic operators (Pedrycz and Gomide, 1998). However, the details of this principle is beyond the scope of this work and we only consider a simple but popular type of fuzzy numbers, known as *fuzzy trapezoidal numbers*.

Fuzzy trapezoidal numbers are one of the intuitive and flexible ways to represent uncertainty in vague quantities. For example, definitions such as 'most probably between 9 and 10 but definitely higher than 7 and lower than 12' are valid statements to define these numbers. Fuzzy trapezoidal number $\tilde{a}$ is represented by a 4-tuple $(\underline{a}, a_L, a_U, \overline{a})$ with a membership function $\mu_{\tilde{a}}(x)$, as follows:

$$\mu_{\tilde{a}}(x) = \begin{cases} \dfrac{x-\underline{a}}{a_L-\underline{a}} & \underline{a} \leq x < a_L \\ 1 & a_L \leq x \leq a_U \\ \dfrac{\overline{a}-x}{\overline{a}-a_U} & a_U < x \leq \overline{a} \\ 0 & otherwise \end{cases}$$

The trapezoidal membership function of fuzzy number $\tilde{a}$ is graphically presented in Figure 3.2.

Table 3.1 shows a typical set of fuzzy operators on trapezoidal fuzzy numbers which are used in this research, where $\tilde{a}$=$(\underline{a}, a_L, a_U, \overline{a})$ and $\tilde{b}$=$(\underline{b}, b_L, b_U, \overline{b})$.

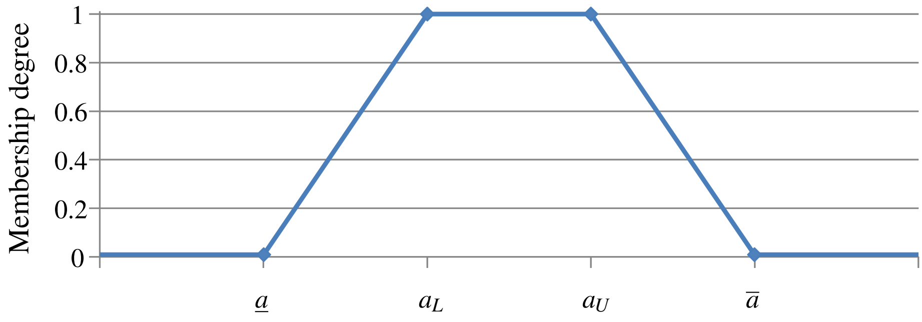

Figure 3.2 Membership function of a trapezoidal fuzzy number

Table 3.1 Fuzzy operators

Operator	Syntax	Formula
Fuzzy Addition	$\tilde{a}+\tilde{b}$	$(\underline{a}+\underline{b}, a_L+b_L, a_U+b_U, \overline{a}+\overline{b})$
Fuzzy Subtraction	$\tilde{a}-\tilde{b}$	$(\underline{a}-\overline{b}, a_L-b_U, a_U-b_L, \overline{a}-\underline{b})$
Fuzzy Multiplication	$\tilde{a}*\tilde{b}$	$(\underline{a}*\underline{b}, a_L*b_L, a_U*b_U, \overline{a}*\overline{b})$
Fuzzy Division	$\tilde{a}/\tilde{b}$	$(\underline{a}/\overline{b}, a_L/b_U, a_U/b_L, \overline{a}/\underline{b})$
Scalar Multiplication	$m\tilde{a}$	$(m\underline{a}, ma_L, ma_U, m\overline{a})$
Defuzzification	$Defuzz(\tilde{a})$	$(\underline{a}+2a_L+2a_U+\overline{a})/6$

It is important to understand that the application of fuzzy extension principle gives the same fuzzy addition, subtraction and scalar multiplication operators as in Table 3.1. However, conventional operations defined by the extension principle for multiplication and division do not necessarily lead to trapezoidal numbers. Hence, the fuzzy multiplication and division operators given in Table 3.1 are trapezoidal approximations of the respective operation (Fodor and Bede, 2006). Also, the defuzzification operator used is based on Detyniecki and Yager (2000).

Another type of often used fuzzy numbers are *fuzzy triangular numbers*. In fact they can be considered as a special case of fuzzy trapezoidal numbers where for fuzzy number $\tilde{a}$, it is assumed that $a_L = a_U$.

3.2.2 Fuzzy control

Fuzzy control refers to the application of fuzzy logic to control theory. A fuzzy controller contains a set of fuzzy if-then rules, defined using fuzzy linguistic variables, to control a system by utilizing analogue input data and producing analogue outputs. Each fuzzy rule consists of a *premise* which is the 'if' condition of the rule and also a *consequence* that determines the outputs of the rule. As the fuzzy if-then rules that constitute the core of the controller are specified in the form of natural language expressions, the main advantages of fuzzy controllers over traditional controllers is the ease of specifying, understanding and manipulating the controller by human experts. This is particularly useful when the system is not or cannot be identified mathematically, but a linguistic description of the system can be obtained from the expert based on the expert's experience, intuition or heuristics.

Fuzzy controllers comprises four main activities: *fuzzification*, *fuzzy inference*, *fuzzy rule base* and *defuzzification*. In the fuzzification component, input is translated into its equivalent fuzzy descriptions which essentially are the membership degrees of the input to the corresponding fuzzy linguistic terms. Fuzzy inference is responsible for determining the outputs based on the fuzzy descriptions of inputs by using the fuzzy if-then rules which are stored in the fuzzy rule base. To accomplish this task, it is necessary to calculate the *firing strength* for each rule which is usually calculated as the minimum of membership degrees of the inputs in the premise of the rule to their corresponding linguistic values. The firing strength is then used to calculate the fuzzy output of the rule for which the membership function is the minimum of the firing strength with the membership function of the linguistic value in the consequence. Finally, in the defuzzification component, the fuzzy outputs of the rules in the inference engine are aggregated for each output and then converted to the representative scalar outputs used to control the system. It is worth mentioning that *scaling functions* are optionally used to scale the inputs and outputs from their corresponding domain into the domain of the membership functions. This is useful for simplifying the definition of membership functions. The overall schema is represented in Figure 3.3.

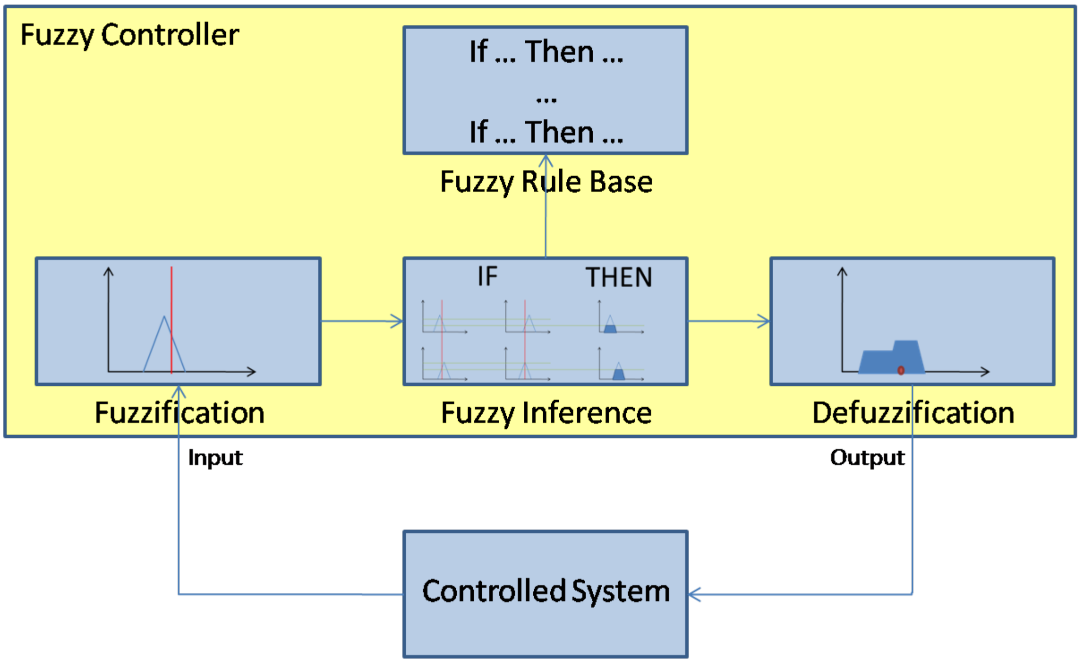

Figure 3.3 Schema of a typical fuzzy controller

Two main types of fuzzy inference are proposed: *Mamdani-type of inference* and *Takagi-Sugeno (TS)-type of inference*. The main difference between the two types of inference is that TS-type has crisp outputs typically modelled as functions of crisp input variables, while Mamdani-type has fuzzy outputs (Zeng et al., 2000). For this reason, the defuzzification stage is only relevant for Mamdani-type inference.

It is worth noting that the information used in the fuzzification and defuzzification components, such as membership function definitions and type of defuzzification are considered to constitute *fuzzy controller's data base*. The fuzzy controller's data base and rule base, which together describe a controller, are collectively known as *fuzzy controller's knowledge base*.

Various methods have been suggested for defuzzification. Most often applied are centre of gravity method and weighted average method. In this research, we apply the *weighted average method*.

Essentially, weighted average method is based on defuzzifying the output of each rule separately (by using the centre of gravity value) and then calculating the weighted average of these defuzzified outputs to have an aggregated value for the corresponding output. The weights used are the firing strengths of the corresponding rules. This method can be formulated as follows:

$$y = \frac{\sum_{i=1}^{n} f_i c_i}{\sum_{i=1}^{n} f_i}$$

where y is the output value, n is the number of rules, f_i is the firing strength of the i-th rule and c_i is the centre of the gravity of corresponding fuzzy output value, for rule i.

While fuzzy controllers allow for extracting knowledge represented by natural language expressions from experts, this approach might not be feasible or desirable in many cases; for example, in complicated systems where experts cannot understand the whole system or where the system properties change frequently and the rule base needs to be updated dynamically. To address these issues, several methods have been introduced with the purpose of *automatic identification* of the fuzzy controller‘s structure and parameters. Some of these methods use various AI techniques such as Fuzzy Clustering, Genetic Algorithm, Artificial Neural Networks (ANNs) and reinforcement learning methods to identify a fuzzy controller (Berenji and Khedkar, 1992; Halgamuge and Glesner, 1994; Herrera et al., 1995; Jang and Sun, 1993; Pham and Karaboga, 2000; Takagi and Sugeno, 1985). The method based on genetic algorithm will be discussed later in this chapter.

3.3 Optimisation methods

Optimisation methods are used to find the optimal values of decision variables for a certain type of models. These models can be either linear or non-linear.

Linear Programming (LP), is a classic example of optimisation techniques which deals with linear models. LP models have been used extensively in the literature for a variety of problems. Also, Mixed Integer Programming (MIP) is another example which deals with models that consist of linear variables along with integer variables. In this section, we will discuss these two main types of optimisation methods.

3.3.1 Linear programming

LP provides a mathematical formulation for determining the best decisions in a model which comprises linear constraints and an objective. LP consists of *decisions variables*, *objective function* and *constraints*. Decision variables are real valued, non-negative variables and their values should be determined. The objective or the goal of the model is given as a linear function of the decision variables, called a 'linear objective function' which needs to be either minimized (like cost or environmental impact) or maximized (like profit, income, or customer satisfaction). Also, constraints are linear non-equalities which should be satisfied by the decision variables.

A maximization LP model can be presented in *canonical form* as follows:

$$\begin{aligned}
&\textit{maximize} \quad && c^T x \\
&\text{subject to}: && \\
& && Ax \leq b \\
& && x \geq 0
\end{aligned} \tag{3.1}$$

where x is a vector of linear variables, c is a vector of coefficients of the objective function, A is a matrix that represents the coefficients of the constraints and b is the vector that represents the right hand sides of those constraints.

Linear constraints of a LP model define a multi-dimensional space called *feasible region*. The decision variables are bound to this region and the objective value should be optimised within it. If this region is empty, the problem will be *infeasible* - without a solution. It can be proven that this region specifies a convex polytope and also, the optimal decisions, if they exist, can be found at one of the vertices of this convex polytope (Bazaraa et al., 2011, p. 91). One of the well-known algorithms to solve LP problems is *Simplex Algorithm* which uses this feature to find the optimal decisions by traversing the edges of the polytope.

Any problem with real variables, linear objective function and linear constraints can be represented as a LP model. Usually LP problems need to be rewritten into the canonical form by making some mathematical manipulations (Bazaraa et al., 2011, p. 4). Also, several commercial and non-commercial *solvers* exist that can solve LP problems, such as CPLEX (IBM), lp_solve (lps, 2010) and Gurobi Optimiser (Gurobi Optimization, 2014).

3.3.2 Mixed integer programming

Mixed integer programming (MIP) considers the type of programming models which have *integer variables* in addition to linear variables. More precisely, it is assumed that some of the variables can only have integer values. Also, some of the variables can be limited to a certain range. For example, in many location allocation problems, *zero-one variables* which are integer variables that can only accept zero or one, are used to decide which potential locations for the facilities are going to be used. It is worth mentioning that the complexity of general mixed integer problems are NP-Complete (Nemhauser and Wolsey, 1999, p. 131).

One of the main algorithms to solve MIP problems is *Branch and Bound Algorithm*. In essence, Branch and Bound algorithm gradually divides the problem into sub spaces with the aim of finding an optimal integer solution using the Simplex Algorithm. Any problem that has an LP solution with non-integer value for an integer variable can be *branched* into two sub problems by splitting the feasible region to two sub spaces. In the first problem, the variable with non-integer value is limited to values which are less or equal than the largest preceding integer value. In the second problem, this variable is limited to values that are more or equal than the smallest following integer. The solution of the initial integer problem should be a solution of either of those sub-problems which might be branched again into other smaller problems. To solve the main problem, it might be necessary to solve all the sub problems. However, using a procedure called *pruning*, some of the sub problems might prove to be futile if any integer solution found proves to be better than the upper bound of the sub problem, for which the LP result can be used as an upper bound. In this case, the sub problem will simply be removed. An example of Branch and Bound method is shown in Figure 3.4.

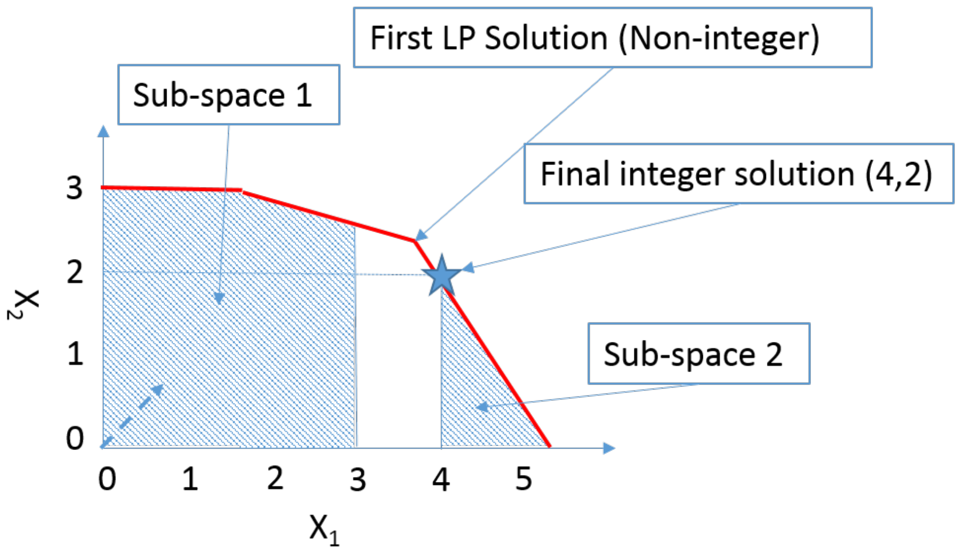

Figure 3.4 An illustrative example of Branch and Bound Method

3.4 Fuzzy programming

In many optimisation applications, gathering precise values of parameters for a model is not possible. As a result, fuzzy sets can be used to model uncertainty of model's parameters. This type of models is collectively known as *fuzzy programming* and can have several variations, depending on the type of uncertainties that exist in the model and also the treatment of those uncertainties in the model (Zimmermann, 2001, p. 337). It is worth mentioning that because of the uncertainty inherent in the model, it is not possible to guarantee an optimal solution and instead, a good solution is desired. To find a good solution, apart from considering the main objective of the model such as cost or profit, it is also necessary to consider the uncertainty of the solution that should be reduced.

In 1970, Bellman and Zadeh proposed an approach to encounter the uncertainty of the objective function and constraints by interpreting both in a symmetrical way. It is assumed that both the fuzzy constraints and the objective(s) are defined by their membership functions. To solve the fuzzy programming model, both constraints and objective(s) need to be satisfied which effectively means that they are treated symmetrically and there is no difference between the two (Zimmermann, 2001, p. 331). It is worth pointing out that symmetric treatment of the constraints and objective(s) is not always desirable and many authors have adapted to use non symmetric approaches (Tang and Wang, 1997; Zimmermann, 2001). We focus only on symmetric approaches in this chapter.

The symmetrical modelling concept by Bellman and Zadeh has been applied to linear programming (Zimmermann, 2001, p. 337). This is done by introducing the concepts of a *tolerance interval* and a *satisfaction degree*. In this method, a tolerance interval is defined as the maximum violation of a constraint or objective function. In the case of objective function, aspirational levels are defined to determine the desirable range for the objective. A satisfaction degree is assigned to each constraint and objective function; the satisfaction degree of 1 means a complete satisfaction of the relevant constraint or objective, while a satisfaction degree of 0 shows a maximum violation. These individual satisfaction degrees are aggregated into a single satisfaction degree of the solution. The satisfaction degree of the solution is often calculated as the minimum of all satisfaction degrees of the objective function and constraints. By maximising this minimum satisfaction degree, essentially, the suitability of all possible solutions is maximised and the most desirable solution can be achieved. Although, as mentioned, the details are different between various approaches.

Interactivity feature of fuzzy programming models is also worth taking into consideration. One of the advantage of fuzzy optimisation approaches is to give the decision maker the ability to judge the results and alter them through the fuzzy parameters, if the results are not satisfactory. To implement this ability, *interactive* models for fuzzy optimisation are introduced (Karsak and Kuzgunkaya, 2002; Stanley Lee and Li, 1993; Wang and Fang, 2001).

Fuzziness can exist in any part of the optimisation model, including the objective function, constraints and variables. Depending on the source of fuzziness, different fuzzy programming models exist. For example, Baykasoglu and Göçken (2008) classified these models into 15 types and reviewed the approaches available in the literature for each type. As the model proposed in this research only contains fuzziness in constraints and on objective function, this type is considered in this chapter only.

3.4.1 Optimisation with fuzzy constraints and fuzzy objective

Uncertainty of optimisation models with a fuzzy objective means that a unique optimal solution is not available. As a result, these models can be translated into a variety of crisp equivalents. So naturally, different approaches to handling fuzzy mathematical programming problems with a fuzzy objective function and fuzzy constraints have been investigated and proposed in the literature (e.g. Cadenas and Verdegay (2006);

Herrera and Verdegay (1995); Inuiguchi and Ramik (2000); Karsak and Kuzgunkaya (2002); Stanley Lee and Li (1993); Wang and Fang (2001)).

One common approach is to define an optimal fuzzy cost set. In this approach, for each value of cost, the maximum satisfaction degree is calculated. This is done by fixing the objective function to the desired cost by adding a new constraint and then, maximising the minimum satisfaction degree of the fuzzy constraints to find the maximum satisfaction degree. Then, the maximum satisfaction degree obtained is assigned as the membership degree of that cost to the optimal cost set. In this way, the decision maker can understand the relationship between the uncertainty and the cost, and ultimately choose the desirable compromise between the two. This approach is used by Liu and Kao (2004) and Cadenas and Verdegay (2006). However, this approach is not applied in this research as this it is very computationally expensive (i.e. requires several runs of MIP models) and also, it provides a membership function as the solution instead of a single crisp value, which is easier to summarise.

In this work, we apply an approach based on Zimmermann (2001) which is also similar to Karsak and Kuzgunkaya (2002). The approach is mainly chosen because it allows for uncertainty in constraints and on objective to be modelled by fuzzy trapezoidal numbers in mixed integer programming models, which can be conveniently solved by the available solvers, and the results are easy to interpret. This approach provides a solution for a mixed integer fuzzy programming model with uncertainty in constraints and the objective converted into a crisp mixed integer optimisation model. It is assumed that fuzzy values are represented by fuzzy trapezoidal numbers and both the less or equal constrains and the greater or equal constraints exist in the model. The use of the trapezoidal membership functions are quite beneficial as calculating the relevant satisfaction degrees is relatively easy.

A fuzzy programming model with fuzzy constraints and a fuzzy objective function is presented as follows:

$$
\begin{aligned}
&\text{Find}\, X = [x_1, x_2, ..., x_n]\, \text{which} \\
&\textit{minimise} \qquad \tilde{f}(X) \\
&s.t. \\
&\qquad g_i(X) \le \tilde{b}_i \qquad\qquad i = 1,2,...,m \\
&\qquad g'_j(X) \ge \tilde{b}'_j \qquad\qquad j = 1,2,...,m'
\end{aligned}
\tag{3.2}
$$

where X is the vector of variables, $\tilde{f}(.)$ is a fuzzy objective function, $g_i(.)$ and $g'_i(.)$ are crisp functions and $\tilde{b}_i$ and $\tilde{b}'_j$ are fuzzy parameters. As we focus on mixed integer linear problems, it is assumed that $\tilde{f}(X)$, $g_i(X)$ and $g'_j(X)$ are all linear functions and vector X consists of non-negative real or integer variables. It is important to note that $\tilde{f}(X)$ cannot include fuzzy coefficients of variables as it will lead to non-linearity in the conversion process, but can include constant fuzzy terms. Additionally, as mentioned, fuzzy parameters $\tilde{b}_i$ and $\tilde{b}'_j$ are modelled using trapezoidal membership functions as illustrated in Figure 3.2.

The resulting fuzzy linear programming is converted into a crisp linear programming model as follows:

$$
\begin{aligned}
&\textit{maximise} \qquad \alpha \\
&s.t. \\
&\qquad \overline{f}(X) + (1-\alpha)\left(f_U(X) - \overline{f}(X)\right) \le f_{\min} + (1-\alpha)\left(f_{\max} - f_{\min}\right) \\
&\qquad g_i(X) \le \underline{b}_i + (1-\alpha)\left(b_{iL} - \underline{b}_i\right) + (1-\alpha)p_i \qquad\qquad i = 1,2,...,m \\
&\qquad g'_j(X) \ge \overline{b'}_j + (1-\alpha)\left(b'_{jU} - \overline{b'}_j\right) - (1-\alpha)p'_j \qquad\qquad j = 1,2,...,m' \\
&\qquad \alpha \in [0,1]
\end{aligned}
$$

where α represents the satisfaction degree and p_i and p'_j are tolerances introduced for fuzzy right hand sides $\tilde{b}_i$ and $\tilde{b}'_j$, respectively. As mentioned before, fuzzy tolerance values represent the maximum extent to which the constraint can be relaxed and represent the decision maker's intuition about the flexibility of parameters and constraints. In extreme cases, the constraints are not relaxed at all, $\alpha = 1$, while for $\alpha = 0$ constraints are relaxed up to their tolerance values as follows:

When $\alpha = 1:$

$$\overline{f}(X) \leq f_{\min}$$
$$g_i(X) \leq \underline{b}_i \qquad i = 1,2,...,m$$
$$g'_j(X) \geq \overline{b}'_j \qquad j = 1,2,...,m'$$

when $\alpha = 0:$

$$f_U(X) \leq f_{\max}$$
$$g_i(X) \leq b_{iL} + p_i \qquad i = 1,2,...,m$$
$$g'_j(X) \geq b'_{jU} - p'_j \qquad j = 1,2,...,m'$$

An example of fuzzy constraints modelled in this way is presented in Figure 3.5. The satisfaction degree of less or equal constraint $g_i(X) \leq \tilde{b}_i$ is shown, with and without the tolerance value.

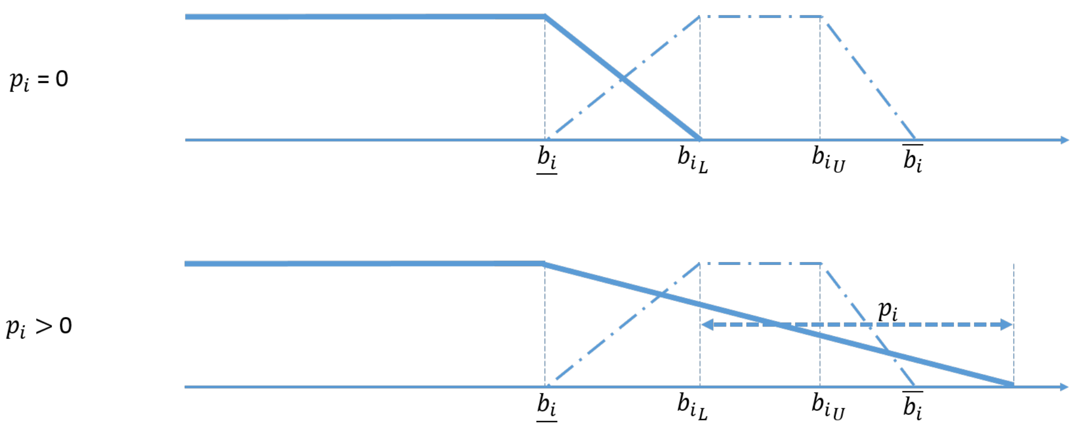

Figure 3.5 Satisfaction degree of $g_i(X) \leq \tilde{b}_i$ with and without the tolerance value

The fuzzy objective function is transformed into a crisp constraint which limits the fuzzy values of the objective function with respect to the worst and best possible objective function values, f_{max} and f_{min}, respectively. These values determine the aspirational level of the objective. In extreme cases, when the satisfaction degree reaches its maximum, $\alpha = 1$, the objective function should be lower than or equal to f_{min}, i.e. $\overline{f}(X) \leq f_{min}$; while for $\alpha = 0$, $f_U(X) \leq f_{max}$. Values f_{min} and f_{max} are determined depending on the problem under consideration.

When a constraint with fuzzy parameters is expressed using equal relation, it is divided into two constraints: one less or equal constraint and one greater or equal constraint, and then converted into equivalent crisp constraints as described above. This feature is used in the fuzzy optimisation model in Chapter 4.

However, while the first priority is to maximise satisfaction degree, different solutions with different objective function values can have the same maximum satisfaction degree. Hence, among these available solutions with the maximum satisfaction degree, it makes sense to choose the one with the better objective function value. This is achieved by running a similar model in a second step. In this step, the satisfaction degree α is fixed by its optimal value α^* and the cost function at this optimal satisfaction degree is optimised. In this way, the solution with the least objective value will be chosen among solutions with equal satisfaction degree. This technique has been applied in Guua and Wu (1999) and Karsak and Kuzgunkaya (2002). The model for the second step is as follows:

$$
\begin{aligned}
&\textit{minimise} \quad \left(\overline{f}(X)+(1-\alpha^*)\left(f_U(X)-\overline{f}(X)\right)\right)\\
&s.t.\\
&\quad \overline{f}(X)+(1-\alpha^*)\left(f_U(X)-\overline{f}(X)\right) \le f_{\min}+(1-\alpha^*)\left(f_{\max}-f_{\min}\right)\\
&\quad g_i(X) \le \underline{b}_i+(1-\alpha^*)\left(b_{iL}-\underline{b}_i\right)+(1-\alpha^*)p_i \qquad\qquad i=1,2,...,m\\
&\quad g'_j(X) \ge \overline{b}_j+(1-\alpha^*)\left(b_{jU}-\overline{b}_j\right)-(1-\alpha^*)p'_j \qquad\qquad j=1,2,...,m'\\
&\quad \alpha^* \in [0,1]
\end{aligned}
$$

Finally, it is trivial to extend this model into multi objective models. As the fuzzy objective function is transformed into a constraint, all objective functions in a multi objective model can also be translated into similar constraints. The objective function in the converted crisp model will require additional terms in the new objectives.

3.5 Genetic algorithm

Genetic Algorithm (GA) belongs to a general class of methods called *Evolutionary Computing (EC)* which are based on the concept of natural evolution. EC methods often simulate biological phenomena such as population, natural selection, genetic inheritance, mutation, survival of the fittest, etc, usually in a stochastic manner, to perform a guided random search (Eiben and Smith, 2008, p. 1-13). Particularly, GA is a *metaheuristic algorithm* which is used to provide useful solutions for optimisation problems. Unlike optimisation techniques, metaheuristics such as GA cannot guarantee an optimal or even a near optimal solution, but metaheuristics are often more flexible in the type of problem they can handle, usually perform faster than the traditional optimisation techniques and can often provide an acceptable solution. One particular benefit of GA is the flexibility in modelling the problem which allows non-linearity, unlike the optimisation techniques which are usually limited to linear or integer problems.

GA is an iterative algorithm which keeps a list of good solutions - the *population*. In each iteration, new solutions are generated from the current population and evaluated using a *fitness function*. New solutions alongside the current population are compared with each other to select the most suitable solutions for the new population. The selection process needs to keep the best solutions (*elite solutions*) which are found so far for the next iteration, while it should also include other (though even worse) solutions to avoid getting trapped in local optima. GA iterations continue until one of the termination criteria such as a maximum number of iterations reached or lack of a change in the best solution recorded in a certain number of iterations, is satisfied.

A problem is described in GA by an appropriate *fitness function* and *chromosome*. The purpose of the fitness function is to evaluate desirability of each solution. This is important in selecting solutions that should be kept or used to generate new solutions. Fitness functions can be of non-linear form, but since the function evaluation happens frequently, they typically need to be easy to calculate. A solution in the solution space is represented as an encoding of the problem decision variables, which is usually done as an array of binary, integer or real variables. By analogy to the genetics, the structure of this encoding is called the *chromosome*, while each particular solution (i.e. a member of the population) is usually called an *individual*. An element of a chromosome is called a *gene* while the value which is represented by a gene is known as *allele*. It is worth pointing out that the encoding of chromosomes has a direct impact on the definition of genetic operations used in the GA and how the GA can create the next generation's population.

An important part of GA is to create new solutions from the previous generation's population. The new solutions should ideally preserve the good features of previous solutions, while new areas in the solution space must be explored as well. *Genetic operations* are used to generate new solutions from the previous ones. They combine some solutions from the previous generation (usually selected randomly) to create *off-springs* for the next generation. Two most important operations are *cross-over* and *mutation*. Cross-over combines two individuals to create two off-springs which both take parts of their chromosome from either of the two parents. In contrast, mutation uses one individual and randomly changes one or more genes in the selected individual to create a new one. Cross-over makes sure that good characteristics are carried over to the next generation, while mutation is necessary to create high diversity among chromosomes. Examples of the two operations are shown in Figure 3.6 and Figure 3.7, respectively.

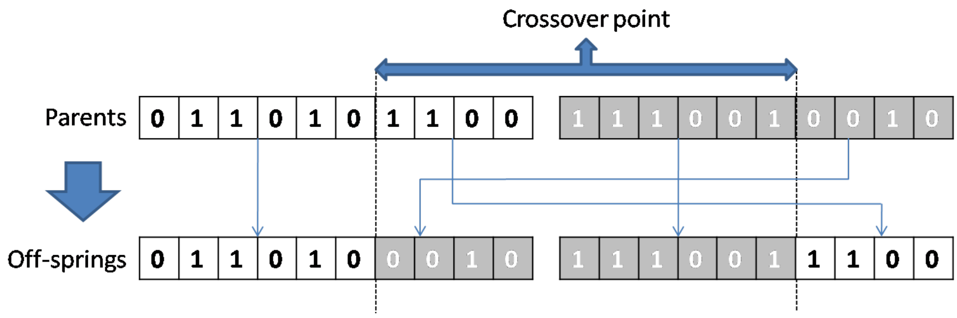

Figure 3.6 An example of a single point cross over operation.

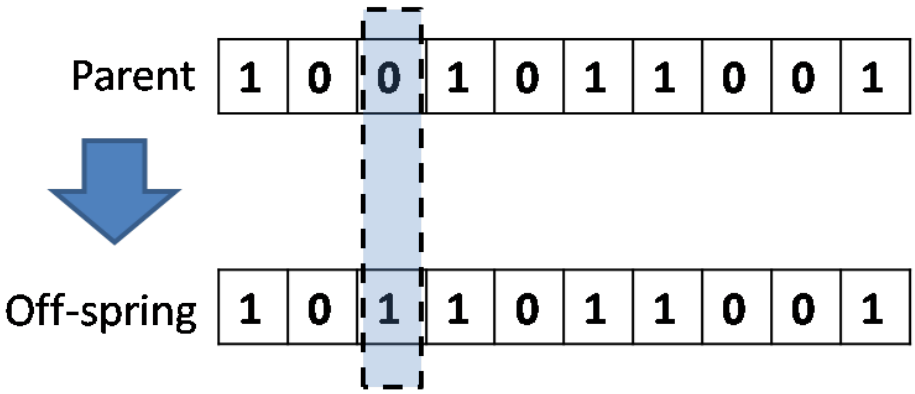

Figure 3.7 An example of a mutation operation.

In summary, GA comprises six stages: initialization, evaluation, selection, genetic operations (including cross over and mutation), replacement and termination. Initialization generates a few random solutions to create the initial population. In each iteration, evaluation is carried out to calculate the value of fitness function for each solution. It is used in the selection stage to determine if the solution will influence the next generation and how frequently it will be used. Different genetic operations are sequentially applied to the selected solutions to generate off-springs and evaluation is repeated for the newly obtained solutions. In the replacement stage, some of the solutions in the current population are replaced with the off-springs based on their fitness values. To terminate the algorithm, either a set number of iterations are used or small changes in the best chromosomes fitness function values between generations indicates the end of algorithm.

3.5.1 Multi-objective GA

Typical optimisation problems involve one objective function only such as economic cost, ecological effect, etc. Even in cases when more than one objective functions are necessary to consider, they might be combined to form a single objective function. Often, it is the sum of weighted objective functions, where weights represent objectives preferences. However, in some situations, a priori knowledge about preferences between objectives might not be available. Further on, a single objective function does not preserve the multi objective nature of a problem and can, for example, enable compensation among objectives. In such situations, a set of *non-dominated solutions* (or *Pareto optimal solutions*) is desirable to obtain. A solution is non-dominated if there is no other solution which is better or equal for all objective functions and is at least better for one objective function. This means that no improvement can be gained in any objective function without sacrificing another one. Otherwise, the solution is considered to be *dominated*, which means that another solution exists that is at least better for one objective function, while it is not worse for any other objective function. Figure 3.8 shows a two objectives solution space with non-dominated solutions and a dominated solution sub space.

Multi Objective Genetic Algorithms (MOGAs) are a variant of GAs which evaluate solutions in a multi objective manner and provide a set of non-dominated solutions

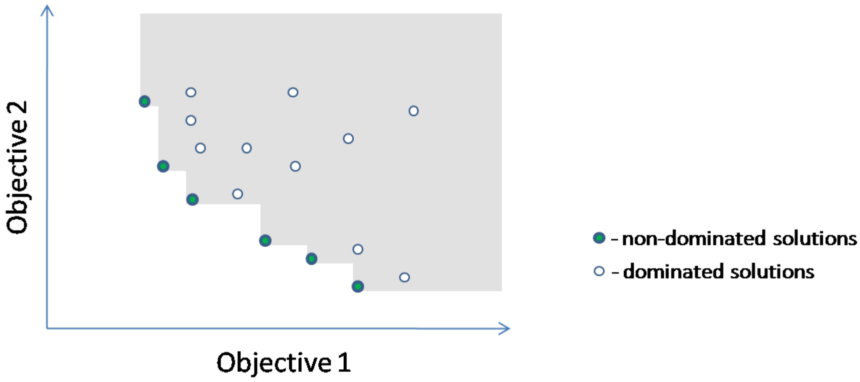

Figure 3.8 Pareto optimal (non-dominated) solutions in a 2 objective functions solution space.

as the result. The fitness functions of MOGAs are usually based on *Pareto ranking* method in which the solution's rank is based on the number of solutions which dominate that particular solution; for non-dominated solutions the rank is one (only the solution dominates itself). Higher ranks are given to worse solutions. One of the most popular MOGAs is *Non-dominated Sorting Genetic Algorithm II (NSGA-II)* (Deb et al., 2002). NSGA-II performs a non-dominated sorting (like many other MOGAs), but the difference is in the use of a complementary crowding distance sorting. Solutions are first chosen by their non-dominated rank, and crowding distance is used when a few solutions need to be chosen among equally ranked solutions. In NSGA-II, the crowding distance is calculated as the sum of the solution's nearest neighbours distances with regard to each objective function divided by the maximum range of corresponding objective function. This feature leads to the higher diversity among the solutions which helps the GA in the discovery of the solution space.

3.6 Genetic fuzzy methods

Fuzzy controllers have several parameters in its knowledge base which need to be determined. On the other hand, finding the right values manually can sometimes be difficult. Different methods have been proposed to help automatically acquiring fuzzy controller parameters. One popular solution is to use GA to determine the parameters. In this way, parameters of the fuzzy controller, such as membership function definitions, scaling functions and fuzzy rules or a part of these parameters are encoded into the genetic representation. The GA is used to find a good fuzzy controller definition for a specific fitness function. The fitness function determines the suitability of the controller for the specific problem. For example, it can be calculated as an output of a simulation of the model which is being controlled by the fuzzy controller which is defined by the genetic chromosome.

In this section, the general classes of genetic fuzzy methods are described briefly. Some of the ways fuzzy rules can be encoded into the genetic representation is also considered. It is worth pointing out that genetic fuzzy methods are used in Chapter 6.

3.6.1 General approaches

As mentioned, genetic fuzzy approaches can be used to determine parameters of any or all parts of the fuzzy knowledge base, including the data base (membership functions, scaling functions, etc) or rule base.

Regarding the data base, there are several ways to encode the membership functions' definitions in a genetic representation. For example, it is possible to have different function types (triangular, trapezoidal, etc.), centre point of the membership function, the scaling function definition and the defuzzification method in the genetic representation (Cordón, 2001, p. 92). However, these methods have not been applied in this research and they are beyond the discussion of this chapter.

Several methods have been proposed for deriving fuzzy rules using GA. Classical methods include 1) Pittsburgh learning approach 2) Michigan learning approach 3) Iterative rule learning approach (Cordón, 2001, p. 127). These methods differ in the way the fuzzy rules discovery is structured. Each of the methods will be discussed briefly.

In Pittsburgh learning approach, each chromosome includes definitions of all fuzzy rules that need to be determined. The fitness function generates a controller based on the chromosome and applies it to the model of the controlled system to measure the suitability of the genetic individual. In this approach, the solution would be the best chromosome of the last population. Main problem of this method is the relatively large search space (Cordón, 2001, p. 179).

Michigan approach, on the other hand, defines a single rule in each chromosome and uses the whole population as the rule base. In each generation, the rules that are worst performing will be removed and all chromosomes of the last population are used to create the solution. This approach is best suited for on-line or reinforcement learning scenarios, where the learning model consists of a set of actions, transitions, rules and rewards and the result of a decision is not explicitly and completely defined (Cordón, 2001, p. 153). The issue of this method is that the rules do not have an opportunity to collaborate and only individual merits are considered.

Iterative rule learning method determines one rule at a time by applying the GA and then repeats this process until enough rules are acquired. When a rule is determined, it is appended to the rule base and the GA is used again until the desired number of rules are attained. After finding these rules, a post processing stage is used to select the rules which are the most cooperating. This approach does not have the problems mentioned for the previous two approaches (Cordón, 2001, p. 219). However, it is more complicated to implement.

3.6.2 Rule encoding methods

A few ways to code fuzzy rules in a genetic representation are proposed in the literature. Three approaches are discussed briefly: 1) Decision table 2) Relational matrix 3) List of rules.

The decision table approach uses a single gene for each combination of inputs where the gene value represents the output linguistic variable. Hence, each gene corresponds to a fuzzy rule while the value shows the output of that rule. A value of zero is often reserved for expressing the lack of a rule. This method is useful when the input space is small, otherwise the matrix will become very large (Cordón, 2001, p. 181). An example of this method is shown in Figure 3.9.

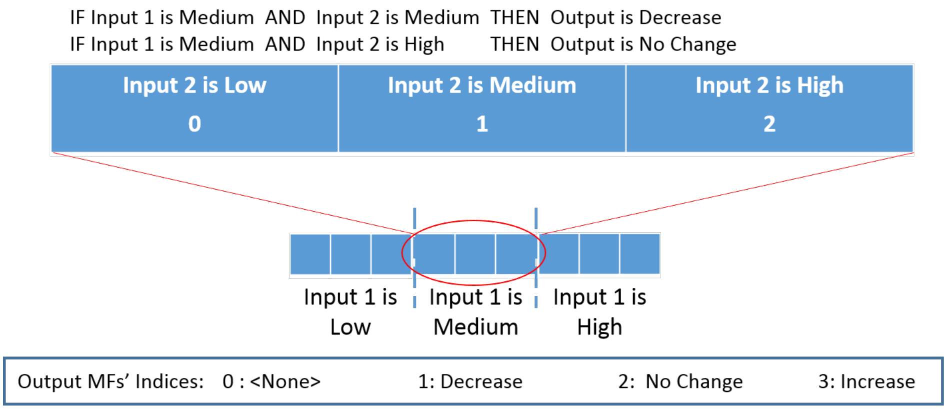

Figure 3.9 An example of decision table genetic fuzzy encoding as an integer array.

It is possible to consider the fuzzy rules as fuzzy themselves. In this way, each rule is assigned a membership degree that is used to determine its membership to the rule base and effectively weight its influence in the fuzzy inference. Please note that this is in addition to the firing strength and should not be confused with it. A relational matrix is a matrix of these membership degrees for all possible rules. The matrix is defined between the combination of fuzzy inputs and combination of fuzzy outputs, to include a cell for each possible rule. This usually require a large real valued genetic representation and is only useful for small number of inputs and outputs (Cordón, 2001, p. 183). An example is shown in Figure 3.10.

The list of rules approach encodes a fuzzy rule separately in the chromosome and does not include all potential rules in the encoding. Hence, it has smaller chromosomes for larger input/output spaces and, as a result, it is used exclusively in such applications.

One approach to encode a rule in the list of rules is by encodings of its terms

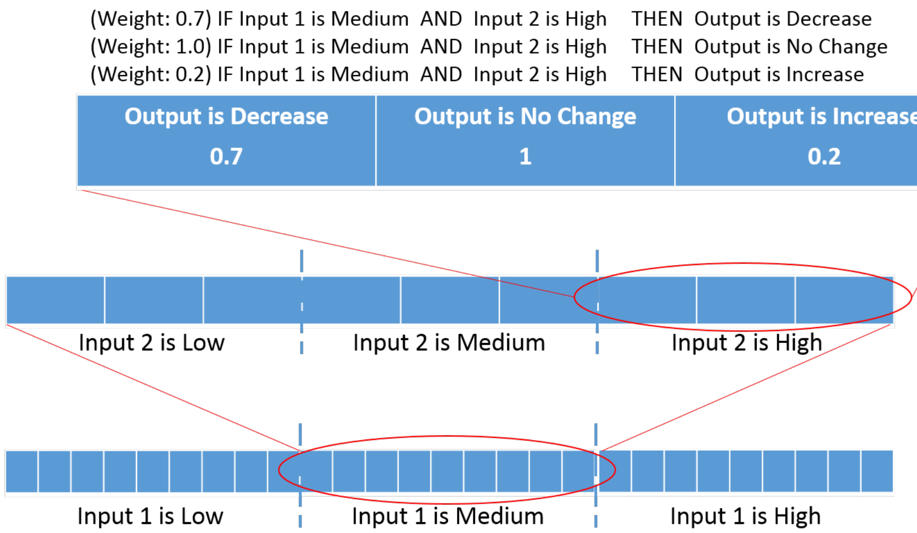

Figure 3.10 An example of a relational matrix genetic fuzzy real-valued encoding.

individually. Each term is defined by a pair of values, the first value showing the input or output being referenced and the second value pointing to the linguistic variable that is being used. In this method, the number of terms in each rule is limited. However, it is possible to reference more than one linguistic variable for an input or output.

Another approach, which is used in this research, is to encode each new rule as an array of integer values with the length equal to the number of input variables plus the number of output variables. Each of these integers represents the index of the linguistic variable used for the relevant input/output variable with zero denoting lack of that variable in the rule. Using this representation, any combination of variables are possible for the rule. An example of this representation is shown in Figure 3.11.

3.7 Discussion and summary

In this chapter, main background concepts used in this research have been introduced and discussed. These concepts are fuzzy logic, including fuzzy sets, fuzzy arithmetic and control; optimisation, including linear programming and integer programming; GA including Multi Objective GA; and also combinations of these methods such as

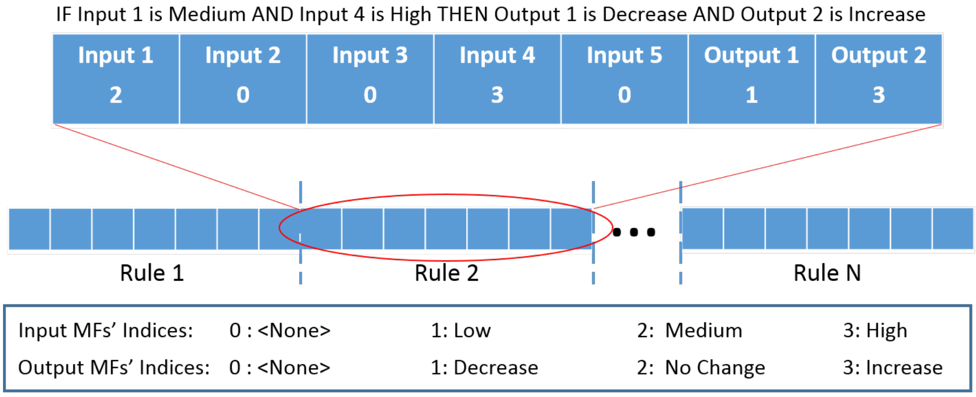

Figure 3.11 An example of the genetic representation of fuzzy rules encoded as an integer array.

fuzzy optimisation and genetic fuzzy approaches. Further on, for fuzzy optimisation, a conversion method has been introduced.

Obviously fuzzy logic plays an important role in this research, as it is necessary to model uncertainty within the reverse logistics networks and fuzzy logic provides a very good method for uncertainty modelling in such networks. Also, several concepts of fuzzy sets and applications such as fuzzy arithmetic, fuzzy rule based control, fuzzy optimisation and genetic fuzzy methods are all applied throughout the research. Furthermore, fuzzy optimisation methods, especially the fuzzy mixed integer optimisation is used in Chapter 4 and it has been briefly introduced. Genetic algorithm, being one of the popular methods of soft computing, is used as part of the genetic fuzzy methods in Chapter 6 and it is concisely examined. However, while multi objective genetic algorithm is not directly applied in the research, it is briefly presented in this chapter since it is commonly used in the literature.

Chapter 4

Reverse Logistics Fuzzy Optimisation

4.1 Introduction

In this chapter, the focus is placed on RL networks integrated with a traditional forward production route, two alternative recovery routes, including repair and remanufacturing, and also a disposal option. Return products are inspected to determine their quality. They are separated into repair, remanufacturing and disposal routes based on repair and remanufacturing quality thresholds. The effects of different repair and remanufacturing thresholds on RL network performance are examined.

It is assumed that product in the RL network includes a single component. An example of such networks exists in tyre industry and especially, recovery of tyres (Lebreton and Tuma, 2006), where multiple recovery options such as re-grooving and rethreading are available. Burning the tyres is often used when the quality of returned tyres are not satisfactory for recovery. Returned tyres are assigned to predefined quality levels which determine the cost of recovery in each recovery route. In this case, tyre casing can be considered as the single recoverable component.

In this chapter, fuzzy sets are used to describe uncertainty in both demand and quantity of returned products of a specific quality level. One of the main advantages that fuzzy sets provide is the possibility of describing parameters as linguistic variables (Zadeh, 1975). In this approach, in the absence of statistical data, the expert can give linguistic descriptions of the quantity values which are modelled using fuzzy numbers, for example, returned quantity is 'considerably more than x', 'about x', 'more than x but less than y', etc. (Petrovic et al., 2008).

This chapter proposes a fuzzy optimisation model to determine network decisions variables. The fuzzy optimisation model helps in dealing with the fuzzy values and finding the decisions which can both reduce the risk of decision making on that non-precise data and also the cost of the network. The fuzzy optimisation model is converted to an MIP model through the procedure described in the previous chapter. Also, to examine the relationship with the quality thresholds, this model is run with all the combinations of these thresholds and the effects they have on the outcome of the optimisation model are analysed. Additionally, the impact of some RL network parameters, including quantity of returned products, unit repair and disassembly costs, and setup costs of all activities, on the selection of optimal quality thresholds is analysed.

This chapter is arranged as follows. In Section 4.2, the problem statement is presented by describing RL networks under consideration and the main assumptions made. In Section 4.3, the fuzzy mixed integer optimisation model is presented. Using the model described in Section 4.3, a set of numerical experiments are conducted and the results are reported in Section 4.4. Finally, in Section 4.5 the chapter is concluded by discussing the summary and outcomes.

4.2 Problem statement

An RL network with two possible recovery routes, including repair and remanufacturing, disposal route integrated with a main production/forward logistics route is considered. Remanufacturing route comprises disassembly of returned products, their stock in the component inventory and subsequent production. Both repaired and remanufactured products are stored in the final products inventory assuming their as-good-as-new condition. Quality inspection, carried out for each returned product, determines the appropriate route that the return should take. In addition, the final products inventory is replenished by the standard forward production route which utilises new components purchase. The RL network is presented in Figure 4.1.

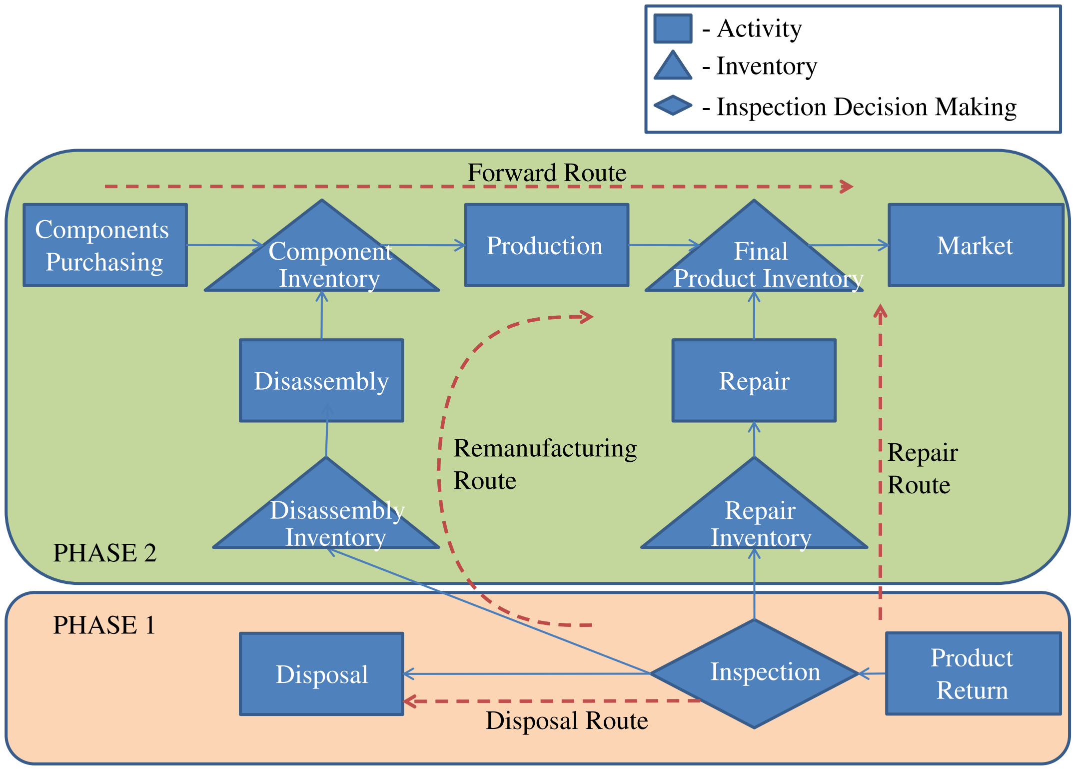

Figure 4.1 Diagram of the integrated RL network

Economic efficiencies of recovery routes are dependent on the quality of returned products. Typically, repair is more efficient for relatively good quality products, while remanufacturing is more appropriate for relatively more defective/damaged products. Quality of a returned product is assigned a nominal quality level which leads to different repair and remanufacturing costs; a higher quality level incurs cheaper repair and remanufacturing costs.

The following assumptions are made:

- The RL network is evaluated using the cost function only. The cost includes inventory holding costs, production and recovery (i.e. repair and disassembly) variable unit costs, production and recovery (i.e. repair and disassembly) setup costs and lost sale costs.

- The RL network is considered within a time horizon.
- The network is dynamic; production and recovery activities have different lead times.
- A single product type consisting of a single component is considered.
- Recovered products, both repaired and remanufactured, are considered as-good-as-new.
- Returned products are inspected on a first come first served basis (i.e. it is not possible to prioritise inspection of some products over the others).
- Demand and return quantities are not precisely known, and they are specified using fuzzy numbers.
- The appropriate recovery path for each returned product is assigned based on quality thresholds.

As mentioned, tyre recovery can be an example of such networks. The tyre can be considered as a single product, while the single component is the tyre casing. Quality grading of returned tyres is also very useful for tyre recovery (Radhi, 2012). While tyre manufacturer do not usually integrate their forward production with the remanufacturing, tyre casings are often procured from third parties when there is a shortage of recovered casings from the internal tyre buffing (disassembly) process. Hence, these other sources of tyre casings can be considered as part of the forward production route while retreading can be treated as remanufacturing, re-grooving and other repair activities can be categorised collectively as repair and burning the tyres is the disposal option. Also, in some RL networks it is possible that the final recovered product is as-good-as-new e.g. when contractual obligations require a specific number of products of acceptable, but not necessarily equal, quality to be supplied. However, even if it is not the case, it is possible to include in the model the difference between the price of the new and the recovered product as a fictitious cost in the repair and/or remanufacturing costs.

4.3 RL optimisation model

The complex structure of the RL network considered, along with different lead times of different routes, setup costs, impact of quality of returned products on the economic efficiency and uncertainty in demand and returned products quantities of different qualities make the optimisation of the whole network a difficult task. In the model proposed, the RL network is split into two sub-networks which are considered in two phases. Phase 1 considers inspection and disposal route, while Phase 2 considers the rest of the network including repair and disassembly inventories and their respective activities, as well as the forward route including procurement, components inventory, production and final products inventory.

Fuzzy return quantities of different qualities are inputs into Phase 1 which calculates the fuzzy quantity of products to be sent to the repair, remanufacturing and disposal routes. Based on these inputs from Phase 1, and fuzzy demand, the optimisation model of Phase 2 determines quantities to be repaired, quantities to be disassembled, quantities of new components to be procured and quantities of final products to be produced in each period of time within the time horizon under consideration.

Apart from reducing the complexity of the problem by dividing it into two sub problems, using two phases has an extra benefit of allowing a more thorough investigation of the quality thresholds in Phase 1. In this way, it is possible to experiment with different methods to determine quality thresholds in Phase 1 while leaving the rest of the decisions to Phase 2.

4.3.1 Phase 1

In this model, the basic policy of inspection of returned products is considered where they are inspected promptly upon arrival. In order to determine the recovery or disposal route for each returned product, Phase 1 uses quality thresholds to separate the returned products into disposable, remanufacturable and repairable products.

The following notations are used:

Table 4.1 Notations used in Phase 1

T	Number of time periods within the time horizon under consideration.
Q	Number of quality levels.
$t \in \{1,2,...,T\}$	Index of time period.
$q \in \{1,2,...,Q\}$	Quality level.
$\widetilde{BI}(t,q)$	Fuzzy quantity of returned products at period t of quality level q, represented as trapezoidal membership function $(\underline{BI}(t,q), BI_L(t,q), BI_U(t,q), \overline{BI}(t,q))$.
$c_R(q)$	Unit cost of repair of product of quality level q.
$c_M(q)$	Unit cost of disassembly of product of quality level q.
c_G	Unit cost of disposal.
QT_R	Quality threshold for returned products acceptable for repair (Repair Quality Threshold).
QT_M	Quality threshold for returned products acceptable for remanufacturing (Remanufacturing Quality Threshold).

Continued on next page

Table 4.1 – *Continued from previous page*

$\widetilde{B'_R}(t,q)$	Fuzzy quantity of inspected products of quality level q to be sent to the repair route at period t, represented as trapezoidal membership function $(\underline{B'_R}(t,q), B'_{RL}(t,q), B'_{RU}(t,q), \overline{B'_R}(t,q))$.
$\widetilde{B'_M}(t,q)$	Fuzzy quantity of inspected products of quality level q to be sent to the remanufacturing route at period t, represented as trapezoidal membership function $(\underline{B'_M}(t,q), B'_{ML}(t,q), B'_{MU}(t,q), \overline{B'_M}(t,q))$.
$\widetilde{B_R}(t)$	Total fuzzy quantity of inspected products to be sent to the repair route at period t, represented as trapezoidal membership function $(\underline{B_R}(t), B_{RL}(t), B_{RU}(t), \overline{B_R}(t))$.
$\widetilde{B_M}(t)$	Total fuzzy quantity of inspected products to be sent to the remanufacturing route at period t, represented as trapezoidal membership function $(\underline{B_M}(t), B_{ML}(t), B_{MU}(t), \overline{B_M}(t))$.
$\widetilde{B_G}(t)$	Fuzzy quantity of inspected products to be sent to the disposal route at period t, represented as trapezoidal membership function $(\underline{B_G}(t), B_{GL}(t), B_{GU}(t), \overline{B_G}(t))$.
$c_{avg,R}$	Average cost of repair per product with respect to different returned products qualities.
$c_{avg,M}$	Average cost of disassembly per product with respect to different returned products qualities.

Quality levels assigned to the returned products after inspection are discrete and crisp values from 1 to Q, where 1 represents the lowest, while Q represents the highest quality levels. Two thresholds, remanufacturing and repair thresholds, divide the quality range into three quality groups: repairable, remanufacturable and disposable products, as shown in Figure 4.2. In the model presented in this chapter, it is assumed that the thresholds are determined in advance.

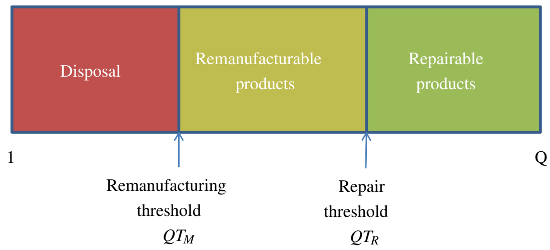

Figure 4.2 Quality groups determined by two quality thresholds

As the quantities of returned products with different quality levels are fuzzy, quantities to be sent to the repair, remanufacturing and disposal routes during the time horizon under consideration become fuzzy too. The following formulas are used to determine these fuzzy quantities using fuzzy operators given in the previous chapter:

$$\widetilde{B'_R}(t,q) = \begin{cases} \widetilde{BI}(t,q) & QT_R \leq q \leq Q \\ 0 & \text{otherwise} \end{cases} \quad \text{for all } q \in \{1,2,...,Q\}$$

$$\widetilde{B'_M}(t,q) = \begin{cases} \widetilde{BI}(t,q) & QT_M \leq q < QT_R \\ 0 & \text{otherwise} \end{cases} \quad \text{for all } q \in \{1,2,...,Q\}$$

$$\widetilde{B_R}(t) = \sum_{q=1}^{Q} \widetilde{B'_R}(t,q)$$

$$\widetilde{B_M}(t) = \sum_{q=1}^{Q} \widetilde{B'_M}(t,q)$$

$$\widetilde{B_G}(t) = \sum_{q=1}^{QT_M-1} \widetilde{BI}(t,q)$$

The returned fuzzy quantities of products to be repaired or remanufactured incur the following costs:

$$c_{avg,R} = \frac{\sum\limits_{q=1}^{Q} c_R(q) \sum\limits_{t=1}^{T} Defuzz(\widetilde{B'_R}(t,q))}{\sum\limits_{q=1}^{Q} \sum\limits_{t=1}^{T} Defuzz(\widetilde{B'_R}(t,q))}$$

$$c_{avg,M} = \frac{\sum\limits_{q=1}^{Q} c_M(q) \sum\limits_{t=1}^{T} Defuzz(\widetilde{B'_M}(t,q))}{\sum\limits_{q=1}^{Q} \sum\limits_{t=1}^{T} Defuzz(\widetilde{B'_M}(t,q))}$$

where the operator $Defuzz$ represents a defuzzified fuzzy value.

4.3.2 Phase 2

A fuzzy mixed integer programming model which accommodates uncertainty in demand and quantity of products sent for repair and remanufacturing is proposed.

The following notations are used:

Table 4.2 Notations used in Phase 2

T	Number of time periods within the time horizon under consideration.
$t \in \{1,2,...,T\}$	Index of time period.
$\widetilde{D}(t)$	Fuzzy quantity of demand at period t, represented as trapezoidal membership function $(\underline{D}(t), D_L(t), D_U(t), \overline{D}(t))$.
$\widetilde{B_R}(t)$	Fuzzy quantity of products sent to repair at period t (calculated in Phase 1).
$\widetilde{B_M}(t)$	Fuzzy quantity of products sent to remanufacturing at period t (calculated in Phase 1).

Continued on next page

Table 4.2 – *Continued from previous page*

LT_C	Lead time of procurement.
LT_P	Production lead time.
LT_R	Repair lead time.
LT_M	Disassembly lead time.
h_S	Unit holding costs of final products inventory.
h_C	Unit holding costs of components inventory.
h_R	Unit holding costs of repair inventory.
h_M	Unit holding costs of disassembly inventory.
c_C	Unit cost of procurement.
c_P	Unit cost of production.
c_L	Unit cost of lost sale.
$c_{avg,R}$	Average cost of repair per product with respect to different qualities (calculated in Phase 1).
$c_{avg,M}$	Average cost of disassembly per product with respect to different qualities (calculated in Phase 1).
f_C	Setup cost of procurement (order cost).
f_P	Setup cost of production.
f_R	Setup cost of repair.
f_M	Setup cost of disassembly.
$H_C(0)$	Initial stock level of components inventory.
$H_S(0)$	Initial stock level of final products inventory.
$H_R(0)$	Initial stock level of repair inventory.
$H_M(0)$	Initial stock level of disassembly inventory.

Continued on next page

Table 4.2 – *Continued from previous page*

$H_C(t)$	Stock level of components inventory at period t.
$H_S(t)$	Stock level of final products inventory at period t.
$H_R(t)$	Stock level of repair inventory at period t.
$H_M(t)$	Stock level of disassembly inventory at period t.
$S(t)$	Quantity of final products to be sent to the market at period t.

Table 4.3 Decision Variables in Phase 2

$CP(t)$	Number of components to be procured at period t.
$C(t)$	Number of components to be used in production at period t.
$R(t)$	Number of products from the repair inventory to be used in repair activity at period t.
$M(t)$	Number of products from the disassembly inventory to be used in disassembly activity at period t.
$\lambda_P(t)$	Zero-one variable to determine if production will occur at period t or not.
$\lambda_C(t)$	Zero-one variable to determine if procurement will occur at period t or not.
$\lambda_R(t)$	Zero-one variable to determine if repair will occur at period t or not.
$\lambda_M(t)$	Zero-one variable to determine if disassembly will occur at period t or not.

Fuzzy programming model

Model-1 represents a fuzzy mixed-integer programming model for optimisation of the RL network under consideration. The objective function includes 5 parts: (I) holding costs for the four inventories in the RL network, including repair, disassembly, final product and component inventories, (II) component procurement, production, repair and disassembly costs, (III) setup costs for respective activities, (IV) lost sale cost and, (V) disposal cost.

Constraint (1) is used to balance the repair inventory level at each period with the previous period. Repair inventory at period t considers the repair inventory at period $t-1$, number of products inspected in period t and sent for repair and number of products to be used for repair in period t. Since the number of inspected products to be sent for repair is uncertain, it is represented as a fuzzy number, and, consequently the constraint is fuzzy, too. Constraints (2) to (4) are similar to constraint (1) but for the disassembly, components and final product inventories. Constraint (5) restricts quantity of products to be sent to the market to be equal to or less than fuzzy demand. Additionally, constraints (6) to (9) are used to make sure that zero-one decision variables for procurement, production, repair and disassembly are set to one when there is any product being procured, produced, repaired or disassembled, respectively at each time period, where Y represents a large number. Furthermore, constraints (10) restrict λ decision variables to be either zero or one, while constraints (11) show that all other variables are non-negative. Finally, constraints (12) set the quantity of procurement, production, repair and disassembly at time period 0 or before to be zero.

Model-1

$$Minimise \sum_{t=1}^{T} [h_R H_R(t) + h_M H_M(t) + h_S H_S(t) + h_C H_C(t)] + \quad (I)$$
$$\sum_{t=1}^{T} [c_C CP(t) + c_P C(t) + c_{avg,R} R(t) + c_{avg,M} M(t)] + \quad (II)$$
$$\sum_{t=1}^{T} [f_C \lambda_C(t) + f_P \lambda_P(t) + f_R \lambda_R(t) + f_M \lambda_M(t)] + \quad (III)$$
$$\sum_{t=1}^{T} c_L(\tilde{D}(t) - S(t)) + \quad (IV)$$
$$\sum_{t=1}^{T} c_G \widetilde{B_G}(t) \quad (V)$$

$Subject\, to:$

$$H_R(t) - H_R(t-1) + R(t) = \tilde{B}_R(t) \qquad 1 \le t \le T \quad (1)$$
$$H_M(t) - H_M(t-1) + M(t) = \tilde{B}_M(t) \qquad 1 \le t \le T \quad (2)$$
$$H_C(t) = H_C(t-1) + CP(t - LT_C) + M(t - LT_M) - C(t) \qquad 1 \le t \le T \quad (3)$$
$$H_S(t) = H_S(t-1) + C(t - LT_P) + R(t - LT_R) - S(t) \qquad 1 \le t \le T \quad (4)$$
$$S(t) \le \tilde{D}(t) \qquad 1 \le t \le T \quad (5)$$
$$Y\lambda_P(t) \ge C(t) \qquad 1 \le t \le T \quad (6)$$
$$Y\lambda_C(t) \ge CP(t) \qquad 1 \le t \le T \quad (7)$$
$$Y\lambda_R(t) \ge R(t) \qquad 1 \le t \le T \quad (8)$$
$$Y\lambda_M(t) \ge M(t) \qquad 1 \le t \le T \quad (9)$$
$$\lambda_P(t), \lambda_C(t), \lambda_R(t), \lambda_M(t) \in \{0,1\} \qquad 1 \le t \le T \quad (10)$$
$$R(t), M(t), CP(t), C(t), H_R(t), H_M(t), H_C(t), H_S(t) \ge 0 \qquad 1 \le t \le T \quad (11)$$
$$C(t) = 0, CP(t) = 0, R(t) = 0, M(t) = 0 \qquad t \le 0 \quad (12)$$

$CP(t)$ where $LT_C < t \le 0, M(t)$ where $LT_M < t \le 0,$

$C(t)$ where $LT_P < t \le 0, R(t)$ where $LT_R < t \le 0,$

$H_R(0), H_M(0), H_C(0)$ and $H_S(0)$ are inputs into the model.

Conversion to a crisp integer programming model

The fuzzy integer programming model, Model-1, needs to be converted into a crisp integer programming model to be solved using available solvers. Constraints (1-2) and (5) have fuzzy right hand sides and part IV of the objective function includes a fuzzy term as well and, hence the objective function is also fuzzy. Different approaches to handling fuzzy mathematical programming problems with a fuzzy objective function, fuzzy coefficients and fuzzy constraints have been investigated and proposed as reviewed in Chapter 3. A modified approach based on the concept of symmetric fuzzy linear programming (Zimmermann, 2001) is proposed in this thesis which is described in Chapter 3. It is applied to Model-1, generating Model-2 as follows:

Model-2

$$
\begin{aligned}
&\mathit{Maximise}\ \ \alpha-\varepsilon\Bigg[\sum_{t=1}^{T}\left[h_R H_R(t)+h_M H_M(t)+h_S H_S(t)+h_C H_C(t)\right]+\\
&\quad\sum_{t=1}^{T}\left[c_C CP(t)+c_P C(t)+c_{avg,R}R(t)+c_{avg,M}M(t)\right]+\\
&\quad\sum_{t=1}^{T}\left[f_C\lambda_C(t)+f_P\lambda_P(t)+f_R\lambda_R(t)+f_M\lambda_M(t)\right]+\\
&\quad\sum_{t=1}^{T}c_L\left[\overline{D}(t)+(1-\alpha)\left(D_U(t)-\overline{D}(t)\right)-S(t)\right]+\\
&\quad\sum_{t=1}^{T}c_G\left[\overline{B_G}(t)+(1-\alpha)(B_{GU}(t)-\overline{B_G}(t))\right]\Bigg]
\end{aligned}
$$

$Subject\,to:$

$$
\begin{aligned}
&\sum_{t=1}^{T}\left[h_R H_R(t)+h_M H_M(t)+h_S H_S(t)+h_C H_C(t)\right]+\\
&\quad\sum_{t=1}^{T}\left[c_C CP(t)+c_P C(t)+c_{avg,R}R(t)+c_{avg,M}M(t)\right]+\\
&\quad\sum_{t=1}^{T}\left[f_C\lambda_C(t)+f_P\lambda_P(t)+f_R\lambda_R(t)+f_M\lambda_M(t)\right]+\\
&\quad\sum_{t=1}^{T}c_L\left[\overline{D}(t)+(1-\alpha)\left(D_U(t)-\overline{D}(t)\right)-S(t)\right]+\\
&\quad\sum_{t=1}^{T}c_G\left[\overline{B_G}(t)+(1-\alpha)(B_{GU}(t)-\overline{B_G}(t))\right]\\
&\quad\leq f_{\min}+(1-\alpha)(f_{\max}-f_{\min})
\end{aligned}
\qquad (0)
$$

$$
\begin{array}{lll}
H_R(t)-H_R(t-1)+R(t)\leq \underline{B_R}(t)+(1-\alpha)\left(B_{R_L}(t)-\underline{B_R}(t)\right)+(1-\alpha)p_R & 1\leq t\leq T & (1)\\
H_R(t)-H_R(t-1)+R(t)\geq \overline{B_R}(t)+(1-\alpha)\left(B_{R_U}(t)-\overline{B_R}(t)\right)-(1-\alpha)p'_R & 1\leq t\leq T & (1')\\
H_M(t)-H_M(t-1)+M(t)\leq \underline{B_M}(t)+(1-\alpha)\left(B_{M_L}(t)-\underline{B_M}(t)\right)+(1-\alpha)p_M & 1\leq t\leq T & (2)\\
H_M(t)-H_M(t-1)+M(t)\geq \overline{B_M}(t)+(1-\alpha)\left(B_{M_U}(t)-\overline{B_M}(t)\right)-(1-\alpha)p'_M & 1\leq t\leq T & (2')\\
H_C(t)=H_C(t-1)+CP(t-LT_C)+M(t-LT_M)-C(t) & 1\leq t\leq T & (3)\\
H_S(t)=H_S(t-1)+C(t-LT_P)+R(t-LT_R)-S(t) & 1\leq t\leq T & (4)\\
S(t)\leq \underline{D}(t)+(1-\alpha)\left(D_L(t)-\underline{D}(t)\right)+(1-\alpha)p_D & 1\leq t\leq T & (5)\\
Y\lambda_P(t)\geq C(t) & 1\leq t\leq T & (6)\\
Y\lambda_C(t)\geq CP(t) & 1\leq t\leq T & (7)\\
Y\lambda_R(t)\geq R(t) & 1\leq t\leq T & (8)\\
Y\lambda_M(t)\geq M(t) & 1\leq t\leq T & (9)\\
\lambda_P(t),\lambda_C(t),\lambda_R(t),\lambda_M(t)\in\{0,1\} & 1\leq t\leq T & (10)\\
R(t),U(t),CP(t),C(t),H_R(t),H_M(t),H_C(t),H_S(t)\geq 0 & 1\leq t\leq T & (11)\\
C(t)=0,CP(t)=0,R(t)=0,M(t)=0 & t\leq 0 & (12)
\end{array}
$$

where f_{min} and f_{max} represent approximations of the best and the worst total cost, which determine the *aspirational level* of the objective. To determine f_{min}, the same model, with $\alpha=0$ and without constraint (0) is optimised and the minimum cost value is used as f_{min}. But, to determine the f_{max}, we assume a worst case scenario of losing all the sale and holding the returned products in the inventories. The cost of the network in this scenario is used as f_{max}, as shown in the following formula:

$$f_{max} = \sum_{t=1}^{T} c_L \overline{D}(t) + \sum_{t=1}^{T} h_R(T-t)\overline{B_R}(t) + \sum_{t=1}^{T} h_M(T-t)\overline{B_M}(t)$$

4.4 Numerical experiments

In order to gain an insight into the behaviour of the RL networks under consideration, different experiments are carried out. First, performances of different policies with different remanufacturing and repair thresholds are obtained. The performance of each policy is calculated by determining outputs of the Phase 1 and inputting them into Phase 2, i.e. the fuzzy optimisation model. Further on, the impact of various RL network parameters on the RL cost incurred is analysed in the corresponding numerical experiments.

The parameters chosen for sensitivity analysis include quantity of returned products, unit repair and disassembly costs, unit production cost and setup costs. Quantity of returned products has been chosen as it has a significant impact on the level of recovery and its influence on the quality dependent routing should be investigated. Unit costs for repair, disassembly and production are very important in determining the average cost of recovery per unit which affects the optimal recovery route. Also, setup costs are included as each route has a different setup cost structure and the effect of changes in these costs are interesting to be analysed.

4.4.1 RL network parameters and inputs

Parameters of a RL network used in the experiments are detailed in Table 4.4.

Please note that inspection costs are assumed to be negligible. Parameters relevant to the conversion of the fuzzy optimisation model into a crisp model are tolerance

Table 4.4 Main RL network parameters

Activities	Unit Cost	Setup Cost	Lead Time
Production	30	1000	3
Repair	Table Below	1000	2
Disassembly	Table Below	1000	4
Components Procurement	100	1000	5
Disposal	5		
Lost Sale	150		

Unit Recovery Cost	Repair	Remanufacturing
Quality Level 1	160	130
Quality Level 2	160	130
Quality Level 3	160	110
Quality Level 4	160	90
Quality Level 5	135	70
Quality Level 6	110	50
Quality Level 7	85	30
Quality Level 8	60	30
Quality Level 9	35	30
Quality Level 10	10	30

Inventories	Unit Holding Cost
Repair inventory	4
Disassembly inventory	3
Component inventory	5
Final product inventory	6

values p_R,p'_R,p_M and p'_M which are set as 30% of the average quantity of returned products in the respective routes, $p_D = 4$, while f_{min} and f_{max} are determined as described in section 4.3.2.

A time horizon of 25 unit time periods is considered. Fuzzy demand and return quantities of different quality levels for each period are presented in Table 4.5. It is worth noting that the total demand in the time horizon is represented by the trapezoidal membership function [1642,1679,1721,1758], while the total quantity of returned products is [1112,1224,1248,1360]. Quantity of return for each quality and time period is a fuzzy number with trapezoidal membership function. Hence, their sum should also have trapezoidal membership function. Defuzzified values of total demand and returned products are 1700 and 1236, respectively. In addition, demand is zero for the first 8 periods to allow the RL network to prepare for supplying the demand (the lead time for the forward production route which is the longest lead time is 8).

Table 4.5 Fuzzy demand and fuzzy quantities of returned products with different quality levels

Time Period	Demand	Quantity of returned products with quality level									
		1	2	3	4	5	6	7	8	9	10
1	[0,0,0,0]	[5,6,6,7]	[5,6,6,7]	[4,4,4,4]	[4,4,4,4]	[5,5,5,5]	[5,5,5,5]	[5,5,5,5]	[3,4,4,5]	[5,5,5,5]	[5,6,6,7]
2	[0,0,0,0]	[3,4,4,5]	[3,3,3,3]	[6,7,7,8]	[5,6,6,7]	[6,6,6,6]	[5,6,6,7]	[5,6,6,7]	[4,4,4,4]	[3,3,3,3]	[4,4,4,4]
3	[0,0,0,0]	[4,5,5,6]	[4,4,4,4]	[5,6,6,7]	[4,4,4,4]	[5,5,5,5]	[6,7,7,8]	[3,3,3,3]	[3,3,3,3]	[4,4,4,4]	[5,5,5,5]
4	[0,0,0,0]	[5,5,5,5]	[6,6,6,6]	[4,5,5,6]	[4,4,4,4]	[3,4,4,5]	[6,6,6,6]	[5,6,6,7]	[4,5,5,6]	[5,5,5,5]	[3,3,3,3]
5	[0,0,0,0]	[4,4,4,4]	[6,6,6,6]	[4,4,4,4]	[4,5,5,6]	[6,7,7,8]	[6,7,7,8]	[2,2,2,2]	[5,6,6,7]	[5,6,6,7]	[4,5,5,6]
6	[0,0,0,0]	[4,5,5,6]	[4,5,5,6]	[4,5,5,6]	[5,5,5,5]	[4,4,4,4]	[5,5,5,5]	[3,3,3,3]	[5,5,7,7]	[6,6,6,6]	[4,4,4,4]
7	[0,0,0,0]	[4,5,5,6]	[4,5,5,6]	[5,6,6,7]	[3,3,3,3]	[6,6,6,6]	[4,4,4,4]	[5,6,6,7]	[6,7,7,8]	[3,4,4,5]	[5,5,5,5]
8	[0,0,0,0]	[4,4,4,4]	[4,5,5,6]	[3,4,4,5]	[4,5,5,6]	[3,4,4,5]	[4,4,4,4]	[3,3,3,3]	[5,5,5,5]	[4,4,4,4]	[5,6,6,7]

Continued on next page

Table 4.5 – *Continued from previous page*

9	[95,98,102,105]	[8,8,8,8]	[4,4,4,4]	[5,6,6,7]	[4,4,4,4]	[5,6,6,7]	[3,4,4,5]	[4,4,4,4]	[4,5,5,6]	[5,5,5,5]	[4,5,5,6]
10	[96,100,100,104]	[4,4,4,4]	[5,5,5,5]	[5,5,7,7]	[4,5,5,6]	[4,4,4,4]	[4,4,4,4]	[4,5,5,6]	[3,3,3,3]	[4,5,5,6]	[5,5,5,5]
11	[95,99,101,105]	[4,5,5,6]	[6,6,6,6]	[5,5,5,5]	[4,5,5,6]	[5,5,5,5]	[3,4,4,5]	[4,4,4,4]	[4,4,4,4]	[5,5,7,7]	[3,3,3,3]
12	[95,98,102,105]	[6,7,7,8]	[3,3,3,3]	[5,5,7,7]	[4,4,4,4]	[4,4,4,4]	[6,6,6,6]	[4,5,5,6]	[4,5,5,6]	[4,4,4,4]	[5,6,6,7]
13	[98,99,101,102]	[4,4,4,4]	[4,4,4,4]	[5,6,6,7]	[3,4,4,5]	[4,5,5,6]	[5,5,7,7]	[4,5,5,6]	[5,5,5,5]	[3,4,4,5]	[3,4,4,5]
14	[96,98,102,104]	[5,5,7,7]	[5,6,6,7]	[4,5,5,6]	[6,6,6,6]	[5,5,7,7]	[6,6,6,6]	[6,6,6,6]	[10,10,10,10]	[5,5,5,5]	[5,5,5,5]
15	[98,98,102,102]	[4,5,5,6]	[6,7,7,8]	[4,4,4,4]	[5,6,6,7]	[4,4,4,4]	[6,7,7,8]	[4,5,5,6]	[4,5,5,6]	[5,6,6,7]	[5,5,5,5]
16	[99,100,100,101]	[3,4,4,5]	[4,5,5,6]	[4,5,5,6]	[3,4,4,5]	[5,5,5,5]	[5,5,5,5]	[3,3,3,3]	[5,5,5,5]	[5,5,5,5]	[4,5,5,6]
17	[98,98,102,102]	[5,5,7,7]	[4,5,5,6]	[6,7,7,8]	[5,5,5,5]	[3,4,4,5]	[4,5,5,6]	[6,6,6,6]	[3,4,4,5]	[4,5,5,6]	[6,6,8,8]
18	[97,99,101,103]	[4,4,4,4]	[5,5,7,7]	[4,5,5,6]	[6,7,7,8]	[5,6,6,7]	[5,5,5,5]	[4,5,5,6]	[4,4,4,4]	[3,4,4,5]	[4,4,4,4]
19	[96,98,102,104]	[4,4,4,4]	[4,4,4,4]	[5,5,5,5]	[3,4,4,5]	[4,4,4,4]	[2,3,3,4]	[3,3,3,3]	[4,4,4,4]	[4,4,4,4]	[6,6,6,6]

Continued on next page

Table 4.5 – *Continued from previous page*

20	[97,99,101,103]	[5,6,6,7]	[5,5,5,5]	[5,6,6,7]	[5,5,5,5]	[5,5,5,5]	[4,4,4,4]	[5,6,6,7]	[4,5,5,6]	[4,5,5,6]	[4,5,5,6]
21	[99,100,100,101]	[3,3,3,3]	[6,6,6,6]	[5,5,5,5]	[4,5,5,6]	[4,4,4,4]	[2,3,3,4]	[3,3,3,3]	[5,5,5,5]	[6,6,6,6]	[5,6,6,7]
22	[95,99,101,105]	[5,5,5,5]	[5,6,6,7]	[4,5,5,6]	[5,5,5,5]	[3,4,4,5]	[4,4,4,4]	[6,6,6,6]	[6,6,6,6]	[5,5,5,5]	[5,5,5,5]
23	[95,99,101,105]	[6,6,6,6]	[3,4,4,5]	[5,6,6,7]	[6,6,6,6]	[3,4,4,5]	[4,5,5,6]	[4,5,5,6]	[5,6,6,7]	[3,4,4,5]	[5,5,5,5]
24	[97,99,101,103]	[3,3,3,3]	[5,6,6,7]	[5,6,6,7]	[5,6,6,7]	[4,4,4,4]	[5,5,5,5]	[4,4,4,4]	[5,5,5,5]	[4,4,4,4]	[5,5,7,7]
25	[96,98,102,104]	[4,5,5,6]	[6,7,7,8]	[5,5,5,5]	[4,4,4,4]	[5,5,7,7]	[5,6,6,7]	[5,5,5,5]	[3,4,4,5]	[4,5,5,6]	[3,4,4,5]

4.4.2 Implementation

Experiments are carried out using the software written in C# language (.NET Framework 4.0). The software includes an implementation for the conversion of fuzzy optimisation model into the MIP model and uses Gurobi Mixed Integer Programming Solver (version 5.5) to solve the MIP problem. The software implemented was run on a computer with Intel Core i7-3612QM CPU @ 2.10GHz and 6GB of RAM.

4.4.3 Basic policies

To better understand the behaviour of the RL network, the impact of different quality thresholds for repair and remanufacturing of returned products are examined. Quality thresholds have a great influence on the RL network performance as they influence the overall cost of recovery activities, inventories and the lost sale costs. However, these relationships are quite complex, because in addition to quality thresholds, other parameters also affect the network performance.

In the following experiments, 66 different basic policies $P(QT_R, QT_M)$ are used, consisting of all possible combinations of repair and remanufacturing quality thresholds, where it is assumed that the quality is described as an integer in the interval [1..11], where quality threshold 11 represent no recovery. The quality threshold for repair is always greater or equal to the disassembly threshold. In the case when they are equal, the returned products of that or higher quality are repaired, while the rest is disposed. Furthermore, when the repair quality threshold is 11, the repair route is not used at all, and, in the case when both thresholds are 11, neither repair nor remanufacturing routes are used and all returned products are disposed. The total quantity of repaired and remanufactured products and the average costs of these recovery activities per product are shown in Table 4.6. It is evident from the table that as the repair threshold increases from 1 to 11, the quantity of returned products sent to the repair route decreases and the average repair cost per unit time period decreases as well. The same applies when the disassembly threshold is increasing.

Table 4.6 Performance of the recovery routes with different recovery thresholds

Policy	Total recovery cost	Average cost of repair per product ($c_{avg,R}$)	Total repair quantity	Crisp total repair quantity	Average cost of disassembly per product ($c_{avg,M}$)	Total disassembly quantity	Crisp total disassembly quantity
P(1,1)	133908	108.34	[1112,1224,1248,1360]	1236	—	[0,0,0,0]	0
P(2,1)	130217	102.63	[1002,1103,1123,1224]	1113	130.00	[110,121,125,136]	123
P(2,2)	114227	102.63	[1002,1103,1123,1224]	1113	—	[0,0,0,0]	0

Continued on next page

Table 4.6 – *Continued from previous page*

P(3,1)	126348	95.11	[886,975,993,1082]	984	130.00	[226,249,255,278]	252
P(3,2)	110358	95.11	[886,975,993,1082]	984	130.00	[116,128,130,142]	129
P(3,3)	93588	95.11	[886,975,993,1082]	984	—	[0,0,0,0]	0
P(4,1)	119649	84.88	[770,843,857,930]	850	123.06	[342,381,391,430]	386
P(4,2)	103658	84.88	[770,843,857,930]	850	119.81	[232,260,266,294]	263
P(4,3)	86888	84.88	[770,843,857,930]	850	110.00	[116,132,136,152]	134
P(4,4)	72148	84.88	[770,843,857,930]	850	—	[0,0,0,0]	0
P(5,1)	111178	72.41	[661,722,736,797]	729	115.17	[451,502,512,563]	507
P(5,2)	95188	72.41	[661,722,736,797]	729	110.42	[341,381,387,427]	384
P(5,3)	78417	72.41	[661,722,736,797]	729	100.51	[225,253,257,285]	255
P(5,4)	63677	72.41	[661,722,736,797]	729	90.00	[109,121,121,133]	121
P(5,5)	52787	72.41	[661,722,736,797]	729	—	[0,0,0,0]	0
P(6,1)	103313	59.96	[551,603,613,665]	608	106.46	[561,621,635,695]	628
P(6,2)	87324	59.96	[551,603,613,665]	608	100.73	[451,500,510,559]	505
P(6,3)	70555	59.96	[551,603,613,665]	608	90.69	[335,372,380,417]	376
P(6,4)	55816	59.96	[551,603,613,665]	608	80.00	[219,240,244,265]	242
P(6,5)	44926	59.96	[551,603,613,665]	608	70.00	[110,119,123,132]	121
P(6,6)	36456	59.96	[551,603,613,665]	608	—	[0,0,0,0]	0
P(7,1)	95757	46.88	[437,478,486,527]	482	97.03	[675,746,762,833]	754
P(7,2)	79765	46.88	[437,478,486,527]	482	90.60	[565,625,637,697]	631
P(7,3)	62997	46.88	[437,478,486,527]	482	80.48	[449,497,507,555]	502
P(7,4)	48257	46.88	[437,478,486,527]	482	69.73	[333,365,371,403]	368
P(7,5)	37367	46.88	[437,478,486,527]	482	59.80	[224,244,250,270]	247
P(7,6)	28896	46.88	[437,478,486,527]	482	50.00	[114,125,127,138]	126
P(7,7)	22596	46.88	[437,478,486,527]	482	—	[0,0,0,0]	0
P(8,1)	89489	35.07	[333,364,372,403]	368	88.23	[779,860,876,957]	868
P(8,2)	73497	35.07	[333,364,372,403]	368	81.33	[669,739,751,821]	745
P(8,3)	56728	35.07	[333,364,372,403]	368	71.14	[553,611,621,679]	616
P(8,4)	41985	35.07	[333,364,372,403]	368	60.33	[437,479,485,527]	482
P(8,5)	31097	35.07	[333,364,372,403]	368	50.39	[328,358,364,394]	361
P(8,6)	22626	35.07	[333,364,372,403]	368	40.50	[218,239,241,262]	240
P(8,7)	16326	35.07	[333,364,372,403]	368	30.00	[104,114,114,124]	114
P(8,8)	12906	35.07	[333,364,372,403]	368	—	[0,0,0,0]	0
P(9,1)	85738	22.24	[220,240,246,266]	243	80.90	[892,984,1002,1094]	993
P(9,2)	69741	22.24	[220,240,246,266]	243	73.95	[782,863,877,958]	870
P(9,3)	52977	22.24	[220,240,246,266]	243	64.20	[666,735,747,816]	741
P(9,4)	38237	22.24	[220,240,246,266]	243	54.09	[550,603,611,664]	607
P(9,5)	27342	22.24	[220,240,246,266]	243	45.14	[441,482,490,531]	486
P(9,6)	18873	22.24	[220,240,246,266]	243	36.90	[331,363,367,399]	365
P(9,7)	12574	22.24	[220,240,246,266]	243	30.00	[217,238,240,261]	239
P(9,8)	9154	22.24	[220,240,246,266]	243	30.00	[113,124,126,137]	125
P(9,9)	5404	22.24	[220,240,246,266]	243	—	[0,0,0,0]	0

Continued on next page

Table 4.6 – *Continued from previous page*

P(10,1)	85140	10.00	[112,122,126,136]	124	75.45	[1000,1102,1122,1224]	1112
P(10,2)	69155	10.00	[112,122,126,136]	124	68.67	[890,981,997,1088]	989
P(10,3)	52384	10.00	[112,122,126,136]	124	59.47	[774,853,867,946]	860
P(10,4)	37642	10.00	[112,122,126,136]	124	50.14	[658,721,731,794]	726
P(10,5)	26753	10.00	[112,122,126,136]	124	42.17	[549,600,610,661]	605
P(10,6)	18282	10.00	[112,122,126,136]	124	35.21	[439,481,487,529]	484
P(10,7)	11980	10.00	[112,122,126,136]	124	30.00	[325,356,360,391]	358
P(10,8)	8560	10.00	[112,122,126,136]	124	30.00	[221,242,246,267]	244
P(10,9)	4810	10.00	[112,122,126,136]	124	30.00	[108,118,120,130]	119
P(10,10)	1240	10.00	[112,122,126,136]	124	—	[0,0,0,0]	0
P(11,1)	87620	—	[0,0,0,0]	0	70.89	[1112,1224,1248,1360]	1236
P(11,2)	71633	—	[0,0,0,0]	0	64.36	[1002,1103,1123,1224]	1113
P(11,3)	54858	—	[0,0,0,0]	0	55.75	[886,975,993,1082]	984
P(11,4)	40120	—	[0,0,0,0]	0	47.20	[770,843,857,930]	850
P(11,5)	29233	—	[0,0,0,0]	0	40.10	[661,722,736,797]	729
P(11,6)	20757	—	[0,0,0,0]	0	34.14	[551,603,613,665]	608
P(11,7)	14460	—	[0,0,0,0]	0	30.00	[437,478,486,527]	482
P(11,8)	11040	—	[0,0,0,0]	0	30.00	[333,364,372,403]	368
P(11,9)	7290	—	[0,0,0,0]	0	30.00	[220,240,246,266]	243
P(11,10)	3720	—	[0,0,0,0]	0	30.00	[112,122,126,136]	124
P(11,11)	—	—	[0,0,0,0]	0	—	[0,0,0,0]	0

The RL network performance achieved under different recovery policies are reported in Table 4.7, including α (the satisfaction degree achieved in fuzzy optimisation), the average cost (which is calculated as the total cost incurred in the network divided by total demand during the time horizon), percentage of demand satisfied through each route, including repair, remanufacturing and forward routes, lost sale, the total setup costs for all recovery and production activities, costs per unit for all recovery and production activities, holding costs of all inventories and lost sale costs.

Table 4.7 Performance of the RL network under different recovery policies

Policy	Satisfaction degree α	Average cost	% of supply				Total costs			
			Repair	Remanufacturing	Forward route	Lost sale	Setup	Activity	Inventory holding	Lost sale
P(1,1)	0.71	134.23	65%	0%	35%	0%	11000	202553	14633	0
P(2,1)	0.58	133.15	62%	0%	38%	0%	13000	196958	16398	0
P(2,2)	0.70	132.57	59%	0%	41%	0%	13000	198710	12991	0
P(3,1)	0.63	133.53	54%	5%	41%	0%	15000	196114	15890	0
P(3,2)	0.58	131.19	55%	0%	45%	0%	13000	192949	16417	0
P(3,3)	0.68	130.46	53%	0%	47%	0%	15000	194539	10889	0
P(4,1)	0.65	132.78	47%	8%	44%	2%	14000	189044	17801	4873
P(4,2)	0.63	130.88	47%	6%	47%	0%	17000	191695	13144	0
P(4,3)	0.58	128.48	48%	4%	49%	0%	16000	189487	11592	0
P(4,4)	0.67	128.02	46%	0%	54%	0%	14000	190285	11048	208
P(5,1)	0.64	130.65	40%	18%	42%	0%	16000	191113	14989	0
P(5,2)	0.65	129.43	40%	13%	46%	0%	16000	189053	14322	0
P(5,3)	0.63	127.59	40%	9%	51%	0%	17000	186261	12300	0
P(5,4)	0.62	126.48	41%	4%	55%	1%	15000	182973	12364	2602
P(5,5)	0.68	126.31	40%	0%	60%	0%	16000	187161	8831	0
P(6,1)	0.68	130.13	33%	23%	43%	1%	17000	188015	14813	1400
P(6,2)	0.64	128.09	34%	18%	48%	0%	19000	186068	12030	0
P(6,3)	0.63	126.35	34%	13%	53%	0%	17000	183512	12935	0
P(6,4)	0.64	125.74	34%	10%	57%	0%	16000	182429	13253	0
P(6,5)	0.58	125.00	34%	5%	61%	0%	15000	182111	12677	0
P(6,6)	0.68	126.27	33%	0%	67%	0%	13000	186629	11651	0
P(7,1)	0.68	129.36	26%	30%	44%	0%	17000	187262	15648	0
P(7,2)	0.68	127.91	26%	25%	49%	0%	19000	184744	13041	0
P(7,3)	0.65	125.75	27%	20%	53%	0%	19000	181308	12123	0
P(7,4)	0.63	124.58	27%	16%	57%	0%	19000	179144	11566	0
P(7,5)	0.63	124.97	27%	11%	63%	0%	16000	180490	13244	0
P(7,6)	0.55	124.99	28%	6%	67%	0%	15000	181363	12784	0
P(7,7)	0.68	128.06	26%	0%	73%	0%	16000	188158	8742	750
P(8,1)	0.63	128.44	20%	35%	44%	0%	18000	185658	14695	0
P(8,2)	0.63	126.84	20%	32%	48%	0%	19000	182664	13301	0
P(8,3)	0.63	125.28	20%	25%	54%	0%	18000	180243	13392	0
P(8,4)	0.63	124.62	20%	21%	58%	0%	19000	178207	12574	0
P(8,5)	0.63	124.94	20%	15%	64%	0%	16000	179785	13892	0
P(8,6)	0.63	126.31	20%	10%	69%	0%	17000	182331	12030	0

Continued on next page

Table 4.7 – *Continued from previous page*

P(8,7)	0.63	128.87	20%	5%	75%	0%	16000	187260	11783	0
P(8,8)	0.63	130.60	20%	0%	80%	0%	16000	192931	8455	0
P(9,1)	0.62	129.54	13%	42%	45%	0%	17000	188057	15161	0
P(9,2)	0.62	127.92	14%	38%	49%	0%	18000	183974	14832	0
P(9,3)	0.62	126.45	14%	32%	54%	0%	20000	180915	12697	0
P(9,4)	0.62	125.79	14%	27%	60%	0%	20000	179678	12096	0
P(9,5)	0.62	126.22	14%	21%	65%	0%	20000	180840	11010	0
P(9,6)	0.62	127.84	14%	16%	70%	0%	20000	183335	10634	0
P(9,7)	0.62	130.75	14%	10%	77%	0%	16000	189166	13075	0
P(9,8)	0.58	133.40	14%	5%	81%	0%	17000	194352	10804	0
P(9,9)	0.62	135.89	14%	0%	86%	0%	15000	201176	9551	0
P(10,1)	0.58	131.89	7%	49%	44%	0%	19000	190980	14238	0
P(10,2)	0.58	130.29	7%	43%	50%	0%	21000	188229	11610	0
P(10,3)	0.58	128.47	7%	38%	55%	0%	18000	185666	13395	0
P(10,4)	0.58	127.95	7%	32%	61%	0%	19000	183703	12745	0
P(10,5)	0.58	128.50	7%	27%	66%	0%	21000	184497	10245	0
P(10,6)	0.58	130.21	7%	22%	71%	0%	19000	187420	11590	0
P(10,7)	0.58	133.34	7%	16%	77%	0%	19000	192658	11008	0
P(10,8)	0.58	136.72	7%	11%	82%	0%	17000	198800	12013	0
P(10,9)	0.58	140.33	7%	5%	88%	0%	16000	205753	11534	0
P(10,10)	0.58	142.30	7%	0%	93%	0%	14000	211896	10111	0
P(11,1)	0.71	135.93	0%	52%	48%	0%	18000	200813	12275	0
P(11,2)	0.70	134.67	0%	47%	53%	0%	18000	197940	12341	0
P(11,3)	0.68	132.91	0%	42%	58%	0%	16000	194619	13971	0
P(11,4)	0.67	132.25	0%	37%	63%	0%	18000	193295	11449	0
P(11,5)	0.68	132.94	0%	31%	69%	0%	18000	194221	11044	0
P(11,6)	0.68	134.44	0%	26%	74%	0%	16000	197131	12034	0
P(11,7)	0.68	137.59	0%	20%	80%	0%	18000	202226	9627	0
P(11,8)	0.63	140.71	0%	16%	84%	0%	14000	207795	12767	0
P(11,9)	0.62	144.12	0%	10%	90%	0%	16000	213926	9790	0
P(11,10)	0.58	147.36	0%	5%	95%	0%	14000	220040	10570	0
P(11,11)	0.62	149.65	0%	0%	100%	0%	12000	226631	9112	83

Table 4.7 shows that the optimal policy with regard to the satisfaction degree is different from the optimal policy from the cost point of view. In terms of average cost, the $P(7,4)$ is the best policy for the network under consideration while in terms of satisfaction degree, both $P(11,1)$ and $P(1,1)$ perform better. This can be justified by the fact that both $P(11,1)$ and $P(1,1)$ eliminate one of the recovery routes and hence, reduce the overall uncertainty by lowering the complexity. However, when the

economic cost is concerned, a combination of both routes can provide better value as the recovery cost differs depending on the quality of return and route of recovery.

Studying Table 4.7 can provide a number of insights. Among many, for example it is interesting to observe that for the policy $P(2,1)$ returned products of quality level 1 are available for remanufacturing but they are not in fact used for this purpose, as that is economically prohibitive. In this case, it is obvious that $P(2,2)$ should have a better performance. This behaviour is also present for the repair route, as can be seen in policies $P(3,3)$ and $P(3,2)$. Regarding the lost sale, as the total forward production unit costs is less than the lost sale, it makes sense to avoid lost sale as much as possible. However, in some situations where a small number of products demand cannot be produced or recovered, it might be preferable to lost the sale as opposed to pay for a new setup of procurement and production to make those products. This behaviour is observable in the table, as a low percentage of lost sale happens infrequently.

In the next sections, the sensitivity of this outcome to different network parameter values is examined. It is worth noting that we will focus on economic efficiency (average cost) aspect of the outcomes within the following experiments.

4.4.4 Quantity of returned products

Quantity of returned products has a considerable effect on the performance of RL networks. In extreme cases, a small return quantity can make recovery uneconomical because forward production is necessary to satisfy demand and setup cost will increase the total cost, while a high return quantity may make forward route unnecessary by providing enough quantity of high quality returned products to satisfy demand fully. In order to understand the effect of quantity of return on the RL network, different quantities of return are tested and the results are presented in Figure 4.3 and Figure 4.4. The percentages refer to the percentage of demand that is returned while average cost refers to the total RL network cost divided by total demand. Fuzzy quantities of returned products in each unit time period are generated in such a way as to make the total quantity of returned products equal to the corresponding percentage of total demand. It is worth noting that, in the main experiment, presented in section 4.4.1, this value was roughly equal to 70%.

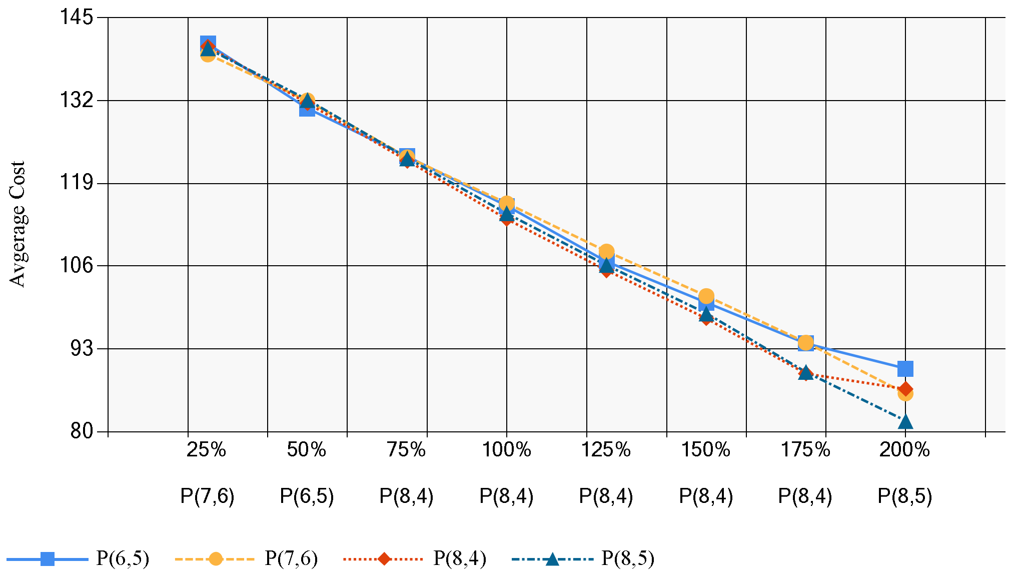

Figure 4.3 The average cost of the best policies for different ratios of return to demand

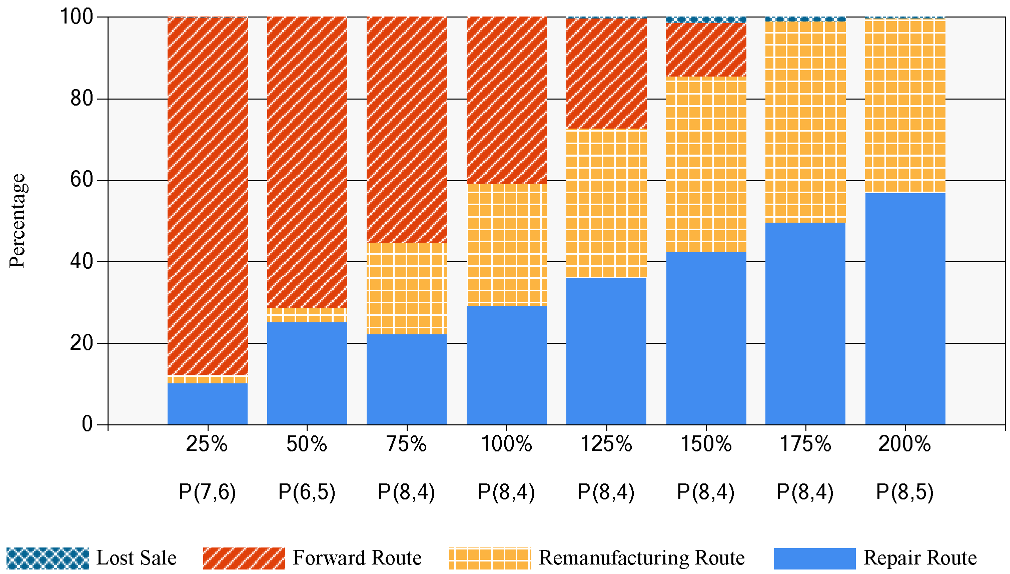

Figure 4.4 Percentage of the products supply of each route and the lost sale for different ratios of return to demand

Figure 4.3 shows the average cost of the RL network achieved for different ratios of return quantity to demand under different recovery policies. Percentages higher than 100 refer to cases when quantities of retuned products are higher than demand satisfied by the RL network under consideration (e.g., products are manufactured by other networks). The chart presents the best average cost incurred and the corresponding recovery policy. The best policies are $P(6,5)$, $P(7,6)$, $P(8,4)$ and $P(8,5)$. As it can be seen in Figure 4.3, policies perform similarly for lower return quantities, because demand is mainly satisfied from the forward route and not from the recovery routes and, hence, the average incurred costs under different recovery policies are similar. For higher return quantities, the differences in average cost are more noticeable.

Figure 4.4 represents the breakdown of different routes used to satisfy demand, expressed by percentages of demand satisfied through repair, remanufacturing and forward routes and the unsatisfied demand i.e., lost sale. The best recovery policy for each case of quantities of returns is printed below the x-axis labels. It can be concluded that for lower and higher quantities of returns the share of repair route is increased compared to the disassembly route as this leads to fewer setups and consequently lower setup costs. In the case of lower returns' quantities demand is mainly satisfied from the forward route. The small number of available returned products are more economical to be directed to one of the routes than to be split between the two routes. Additionally, repair route is preferred as the cheap unit cost of repair, compared to the respective disassembly unit cost plus unit cost of production, for high quality products make repair the more attractive option in comparison to remanufacturing route. In the case of higher returns' quantities, more demand is satisfied using the recovery routes than the forward route. In this case, both routes are utilised until the quantity of good quality returned products surpass the demand, in which, repair route will be preferred as it has a lower setup cost as well as having a cheaper repair unit cost for high quality returns. In this case, repair of higher quality returns is effectively substituting remanufacturing of relatively lower quality returns which is possible because of the abundance of returned products.

4.4.5 Unit repair and disassembly costs

Repair and disassembly costs have a great effect on the RL network decisions on which routes to use for product supply; a change in unit costs could make the alternative recovery option more or less attractive for recovery of products of a particular quality level, but, also, it can affect the ways in which products are supplied. In this experiment, various changes in unit repair and disassembly costs are considered, expressed as percentages of the initial repair and disassembly costs for different quality levels. The results are shown in Figure 4.5 and Figure 4.6 for repair costs; and Figure 4.7 and Figure 4.8 for disassembly costs. The best recovery policy is mentioned under the x-axis for each change in the unit recovery costs.

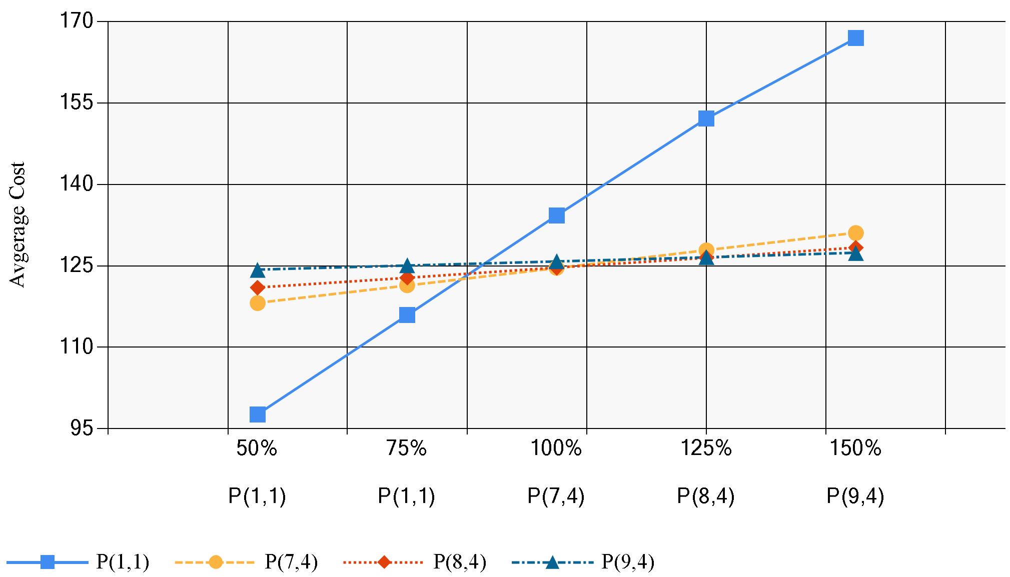

Figure 4.5 Comparison of the best policies average cost for different unit repair costs

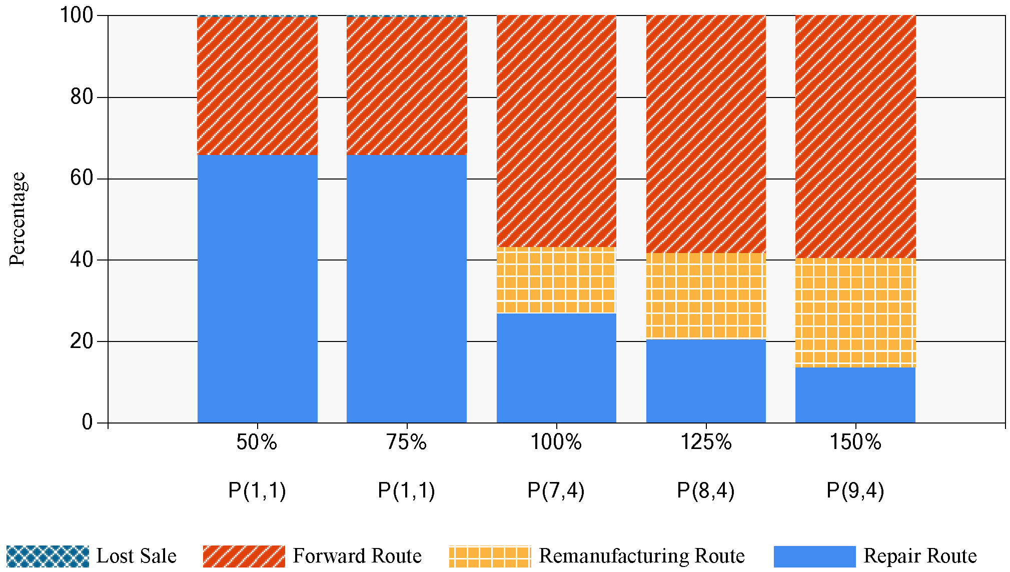

Figure 4.6 Percentage of the products' supply of each route and the lost sale for different unit repair costs

As one can see, in Figure 4.5, in the case of lower unit repair costs, the policy $P(1,1)$ which has a low quality threshold for repair and uses the repair route for recovery only, outperforms other policies. Furthermore, this policy is more sensitive to the increases in unit repair costs, and, as the repair costs increase, the average cost incurred increases rapidly. In contrast, average costs under recovery policies with higher repair thresholds (such as $P(9,4)$ and $P(8,4)$) are less sensitive to increases in the unit repair costs because less quantities of returned products are repaired. It is evident from Figure 4.6, that for lower unit repair costs, the repair route is the main source of supplies with the help of the forward route, while for higher unit repair costs, the remanufacturing route as well as the forward route are used more as these are the cheaper alternatives to the repair.

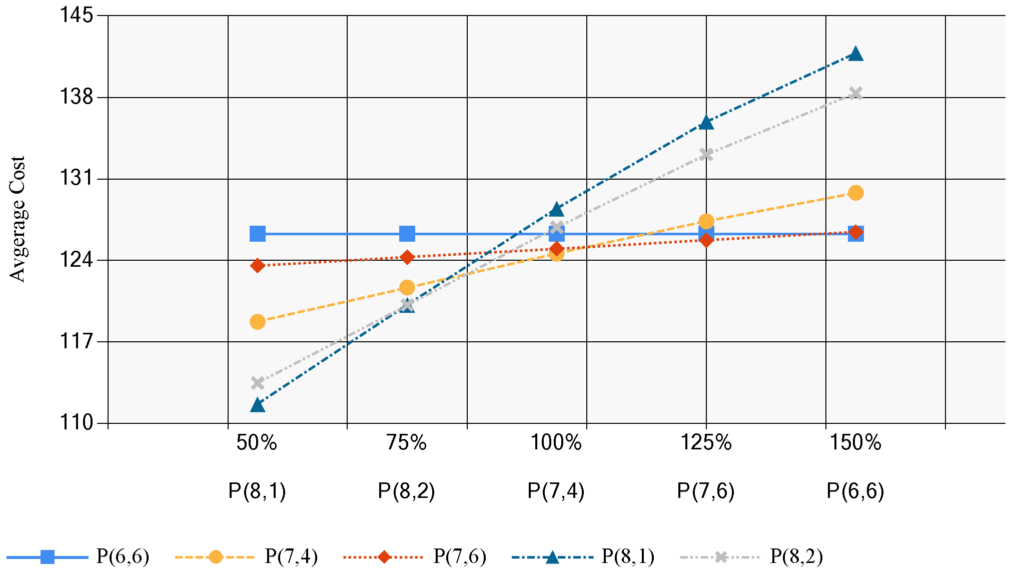

Figure 4.7 Comparison of the best policies average cost for different unit disassembly costs

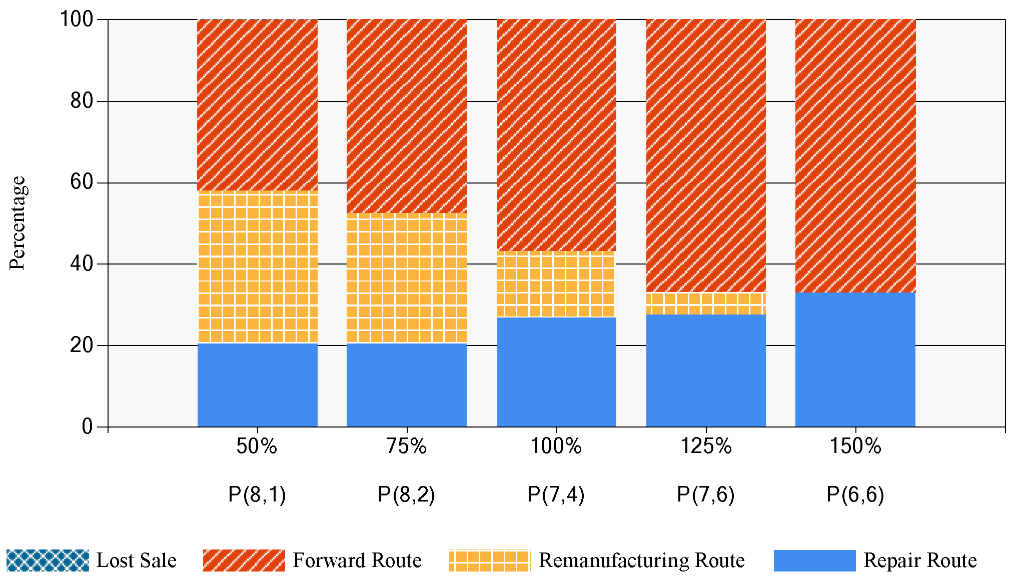

Figure 4.8 Percentage of the products' supply of each route and the lost sale for different unit disassembly costs

Similarly for different disassembly costs, in Figure 4.7 the lower the disassembly cost is, more remanufacturing is expected and the higher it is, less remanufacturing will be used. This leads to policies with very low remanufacturing threshold and high repair threshold (such as $P(8,1)$ and $P(8,2)$) for low disassembly costs and repair only policies such as $P(6,6)$ for high disassembly costs.

Obviously, the same trend can be seen in Figure 4.8 where for lower disassembly costs we see more remanufacturing and for higher costs, less remanufacturing share in the supply. It is important to note that, as remanufacturing route "competes" with both repair and forward routes, we also see an increase in repair route's share, as the disassembly costs increases.

4.4.6 Setup costs

To understand the effect of setup costs on the average cost incurred in an RL network different values of setup costs for each of the four activities, including repair, disassembly, production and procurement, are examined. Setup costs for repair, disassembly, production and component procurement are set to the values which changes from 100 to 10000. The results are shown in Figure 4.9 and Figure 4.10 for repair setup cost; Figure 4.11 and Figure 4.12 for disassembly setup cost; Figure 4.13 and Figure 4.14 for production setup cost and Figure 4.15 and Figure 4.16 for procurement setup cost.

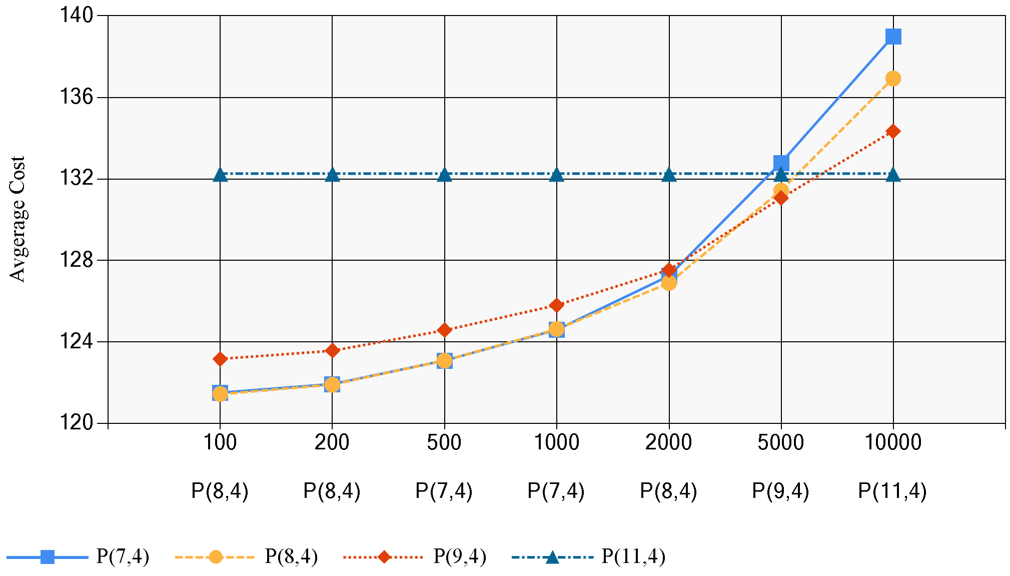

Figure 4.9 Average cost incurred under the recovery policies for repair setup cost

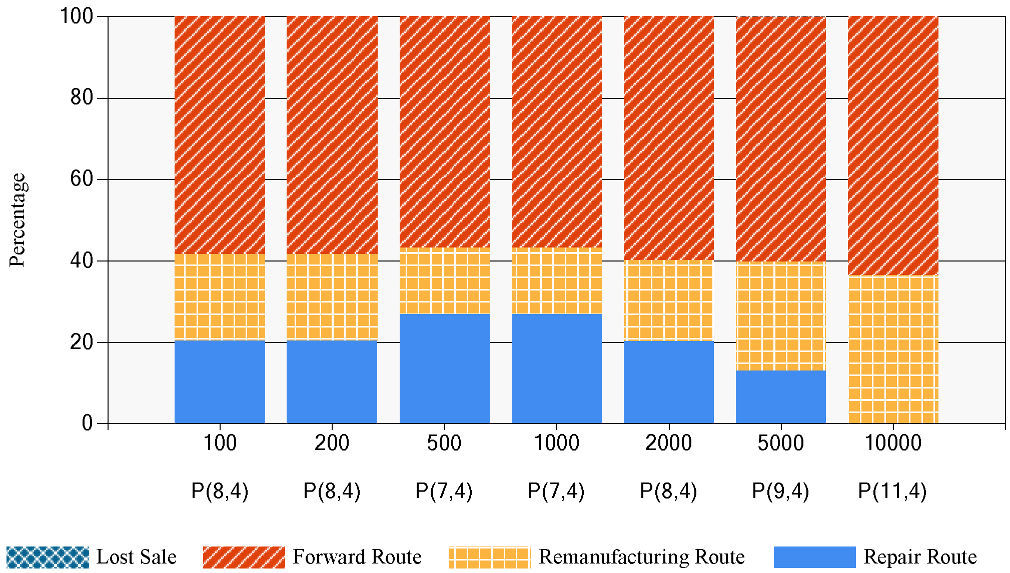

Figure 4.10 Percentage of the products supply of each route and the lost sale for different repair setup cost

It is evident from Figure 4.9 that as the repair setup cost increases, the average costs for all policies which include repair increase as well. Hence, for very high repair setup costs, a remanufacturing only policy such as $P(11,4)$ is more economical. Also, one can see in Figure 4.10 that the share of remanufacturing increases when the repair setup cost reaches its highest values while repair activities are reduced.

What is surprising is the slight increase in repair quality threshold when the repair setup costs is reduced to 200 from 500. First, it is important to observe that the average cost of the best policy still continues to reduce by a reduction in setup costs. This phenomena is possibly because repair cost at 1000 is expensive enough to limit the number of setups for repair but, because of difference in unit costs, repair of high quality returns is still more economical than disassembly and repair setups are unavoidable. So, few repair setups with higher quantities are utilised to limit the effect of repair setup costs while eliminating some of the disassembly setups in favour of larger repair setups. However, for repair setup cost of 500, the network has a higher degree of freedom to have repair setups with smaller quantities. Hence, a higher repair quality threshold is used while disassembly activities are increased.

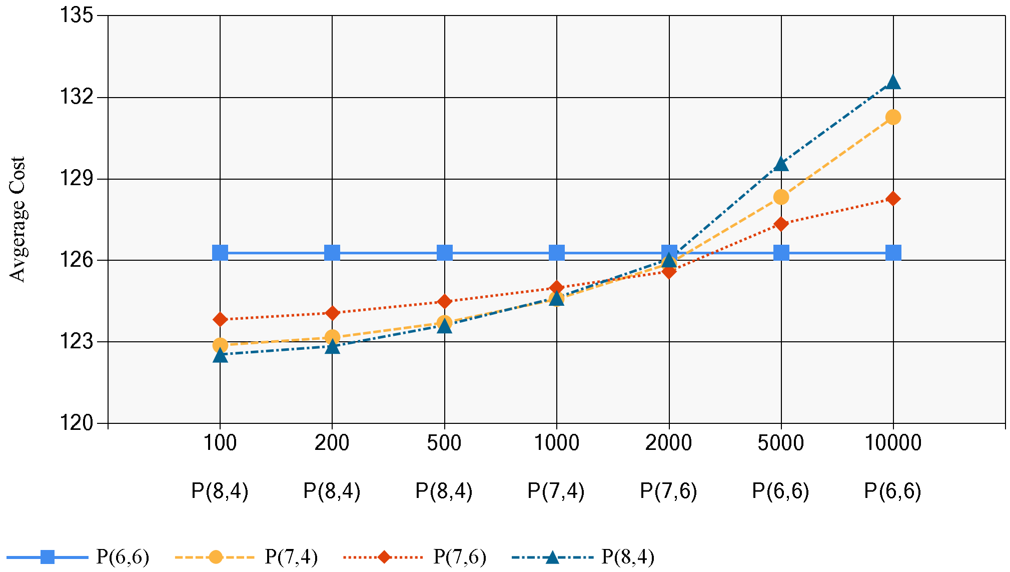

Figure 4.11 Average cost incurred under the recovery policies for different disassembly setup cost

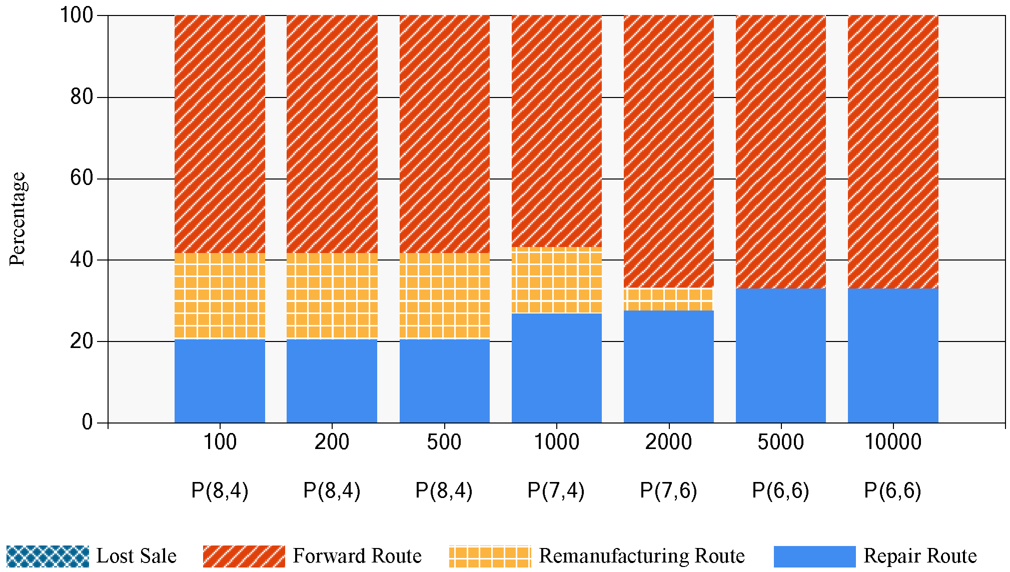

Figure 4.12 Percentage of the products supply of each route and the lost sale for different disassembly setup cost

In Figure 4.11, as the disassembly setup cost increases, we see an increase in the average cost of all policies which include remanufacturing activities. The increase obviously has a direct relationship with the quantity of products sent to this route, which is also dependant on the difference between the repair and remanufacturing quality thresholds. Based on the same rule, repair only policies such as $P(6,6)$ are independent from the disassembly setup cost and perform better in high disassembly setup scenarios. Also, it can be noticed in Figure 4.12 that the share of remanufacturing decreases with an increase in the disassembly setup cost. This share will be proportionately replaced by the forward and repair routes.

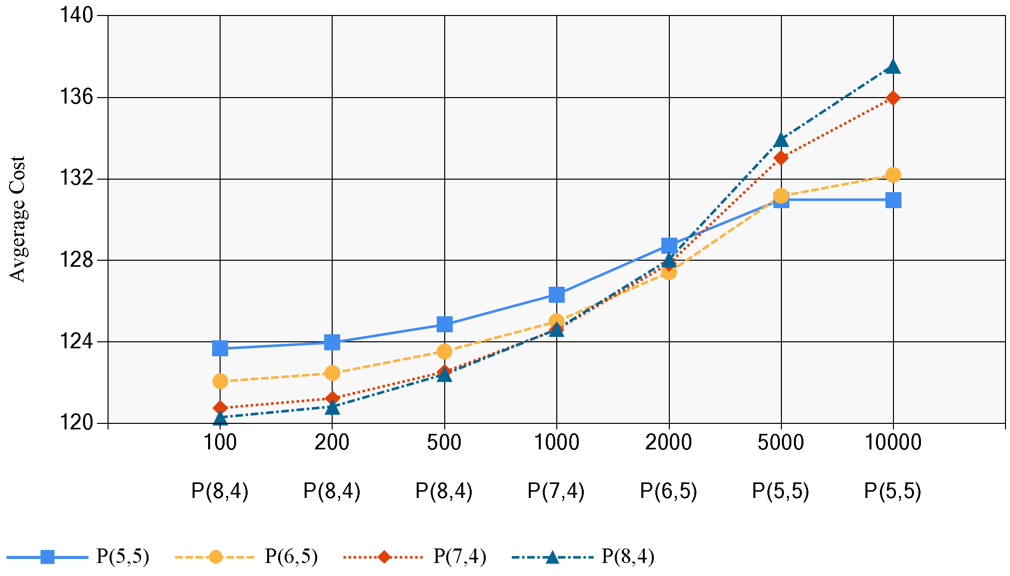

Figure 4.13 Average cost incurred under the recovery policies for different production setup cost

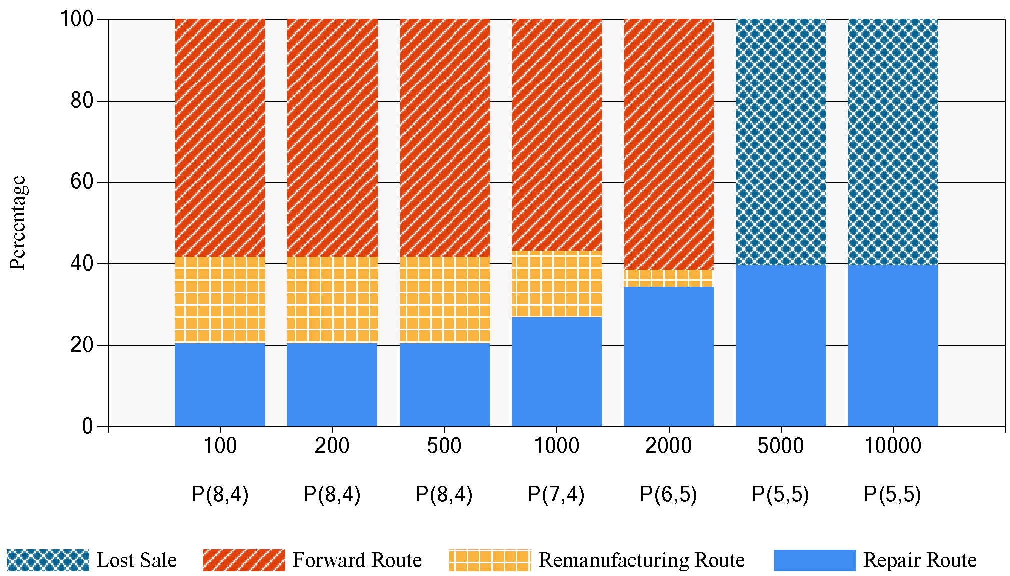

Figure 4.14 Percentage of the products supply of each route and the lost sale for different production setup cost

As it can be seen in both Figure 4.13 and Figure 4.14, changes in production setup cost interestingly leads to results similar to that of the changes in disassembly setup cost. This is because of the role of production activity in remanufacturing which means that it will affect the remanufacturing route but not the repair route. Although, it also affects the forward route which means that for higher production setup costs, lost sale increases while repair also substitutes the remanufacturing route.

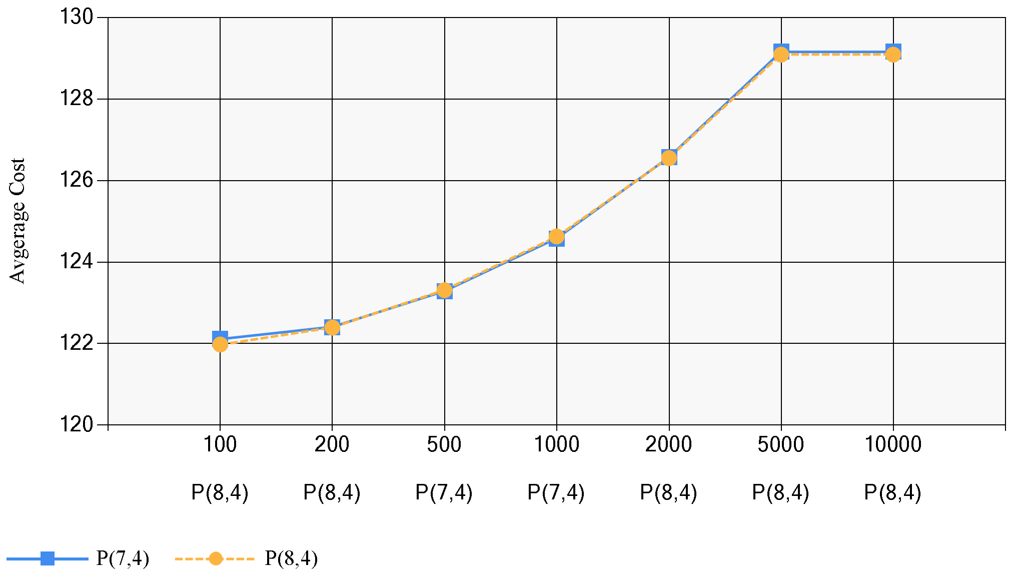

Figure 4.15 Average cost incurred under the recovery policies for different procurement setup cost

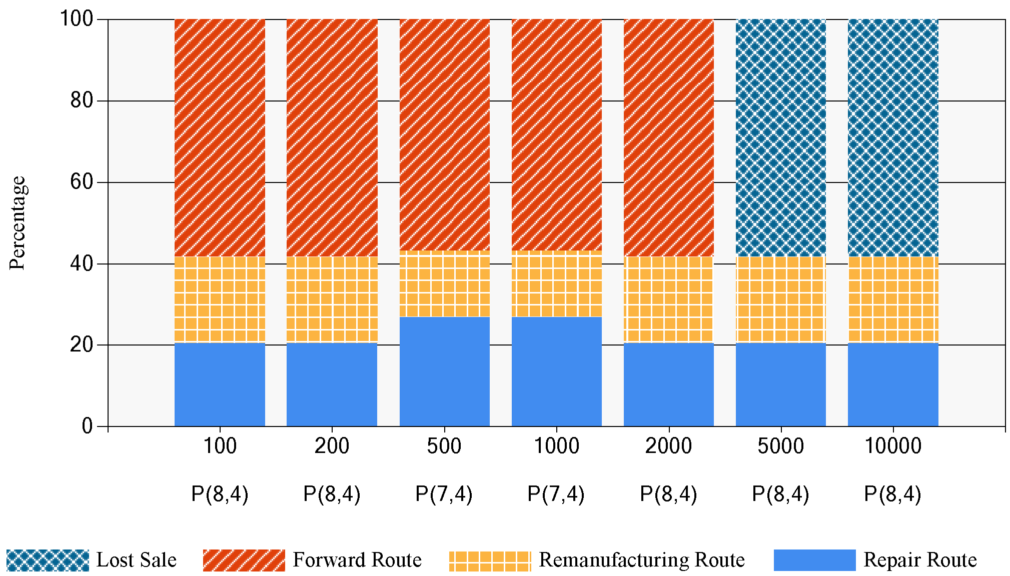

Figure 4.16 Percentage of the products supply of each route and the lost sale for different procurement setup cost

Considering both Figure 4.15 and Figure 4.16, procurement setup cost has very little effect on the determination of optimal quality thresholds. The alternation between two close policies ($P(8,4)$ and $P(7,4)$) has very little effect on the network cost. The only important behaviour to notice is the switch to the lost sale instead of the forward route for high procurement setup costs which is due to forward production becoming unprofitable for high procurement costs.

4.5 Conclusions

This chapter is focused on RL networks with a forward route, two recovery routes, including repair and remanufacturing, and a disposal option. A fuzzy mixed integer programming model which facilitates decision making in presence of uncertainty in demand and quantity of returned products of different quality levels is developed. Quality of products is described as a scalar value which is utilized to separate the returned products into the two recovery, including repair and remanufacturing routes, and a disposal route.

The RL network performances, including satisfaction degree (α), average cost of recovery, share of supply for each route and a breakdown of total costs, under different recovery policies are compared. It is concluded that recovery policies have a considerable impact on the RL network cost. Also, by carrying out numerical experiments, it is concluded that return quantity, unit repair and disassembly costs and some of the setup costs have impacts on the optimal recovery policy. Hence, a simple approach which assumes all returned products to be either recoverable or non-recoverable is not always realistic and can lead to inferior solutions.

For quantity of returned products, it is observable that mixed production and recovery policies are only desirable for medium values of return, between 50% and 150% of demand. This is because for values lower than 50%, the economic savings from recovery does not justify the setup costs necessary for operating two alternative recovery routes. Hence, one of them should be closed down, in which case remanufacturing is chosen as it depends on setup of both production and disassembly as opposed to repair which only requires the repair setup. Also, for values higher than 150%, enough products are available through recovery which removes the need for production altogether. Additionally, as the quantity of returns increases even further, eventually remanufacturing route will also be redundant as the repair of higher quality products can eventually satisfy demand completely.

For unit repair and disassembly costs, the results are hardly surprising as a lower (higher) cost of repair (disassembly) will lead to an increase (decrease) of repair (disassembly) activity. Although to achieve this, sometimes both quality threshold needs to be adjusted. For example, a decrease in disassembly unit costs leads to an increase in repair quality threshold and decrease in remanufacturing quality threshold.

Regarding the setup costs, a significant increase in either setup cost will lead to the suspension of the respective route. So, for high repair setup cost, a remanufacturing only policy; for high disassembly setup cost, a repair only policy; for high production setup cost, a repair only policy without forward production; and for high procurement setup cost, a mixed policy without forward production are preferred. Also, in all cases, it can be seen that a reduction of setup costs, to less than 1000, usually makes the policy $P(8,4)$ the preferred option over $P(7,4)$. This is because the two policies have a very close performance and minor changes can lead to a switch between the two.

Chapter 5

Development of a Fuzzy Controller to Determine Quality Thresholds

5.1 Introduction

Quality of returned products has a significant impact on the cost of recovery activities. This impact implies the importance of choosing the right quality levels to be used for each type of recovery activity. However, it is challenging to find the optimal quality levels for a multitude of reasons. Firstly, many parameters of the logistics network can affect the optimal values and, because of their usually non-linear relationship, sometimes even the slightest changes in one parameter can have drastic consequences. Secondly, assuming that an exact-optimisation method can be used to determine a better solution, because of its mathematical complexity, the solution would be hard to interpret by the managerial experts. These issues can be addressed by using fuzzy control which has the advantage of being easily interpretable by human experts, as linguistic terms and concept are used to construct the knowledge base of the fuzzy controller.

In this chapter, five main network parameters, including repair to new (forward production) unit cost ratio, remanufacturing to new (forward production) unit cost ratio, repair to disassembly setup ratio, disassembly to procurement setup ratio and return to demand ratio, are considered. First four parameters are selected to represent most of the influential parameters relevant the cost of the alternative routes which obviously have a significant effect on the quality threshold. The last parameter, return to demand ratio, is also important as the relative quantity of return can lead to different scenarios for recovery operations.

A fuzzy controller is proposed to heuristically determine the 'right' decision about returned products quality thresholds. The network parameters are used as the controller's inputs and quality thresholds (for repair and remanufacturing) are determined as the outputs. The controller's performance is compared with benchmark policies, including fixed threshold policies and a policy based on cost comparison estimation.

Other decision making tools, from optimisation methods to different artificial intelligence techniques can be applied to determine the quality thresholds. However, fuzzy control has been chosen as it allows the utilisation of expert's knowledge in the form of rules written in natural language as well as allowing humans to directly understand the knowledge collected as part of the rule base. In this way fuzzy control sets apart from many other methods, such as neural networks, that often work simply as a 'black-box' (Casillas, 2003).

This chapter is structured as follows. In Section 5.2, a dataset used throughout this chapter to test the results is presented. Section 5.3 introduces the fuzzy controller which is proposed here, detailing its inputs, outputs, membership functions and rules. Afterwards, Section 5.4 discusses benchmark policies that can be used instead of policies generated by the controller. In Section 5.5 the results of the proposed controller as well as the benchmark policies on the test dataset are presented and analysed. The sensitivity of these policies are analysed and compared in Section 5.6 and finally in Section 5.7 a summary of the outcomes is discussed and some future directions are given.

5.2 Test dataset

The proposed controller needs to be general enough to be applicable to any RL network with a similar structure. To examine this, values of various parameters of the network are randomly generated. A set of these random values, called data points, are used as a test dataset. To evaluate a particular quality thresholds policy, it is tested on each data point and compared with the best policy available for that data point. An error is calculated for each data point and each policy as the network cost of that policy for the data point minus the network cost of the best policy for the data point which is then divided by the network cost of the best policy for the data point. Finally, using Mean Percentage Error (MPE), the error rate over the whole dataset (all data points) is calculated, using the individual error rates.

The network parameters used are presented in Table 5.1. It is worth noting that the size of the dataset used for the purpose of experiments in this chapter is 1000 data points.

As shown on Table 5.1, setup costs for production, repair, disassembly and components procurement are set randomly from a range of available values {100, 200, 500, 1000, 2000, 5000, 10000}. It is worth noting that each of these setup costs are chosen separately and can be different from each other.

Table 5.1 represents 10 quality levels for returned products and the corresponding unit costs. The unit costs for repair and remanufacturing are set randomly too. For these unit costs, two ratios P_R and P_M are chosen randomly, from a predefined set, and unit costs are calculated as the corresponding chosen ratios of the initial costs. Unit repair costs for all quality levels are calculated using the chosen repair cost ratio P_R and similarly all unit remanufacturing costs are determined using the independently chosen remanufacturing cost ratio P_M. The initial costs are the same as those defined in Chapter 4.

All other parameters shown in this table stay constant for the purpose of this test.

Table 5.1 RL networks parameters

Activities	Unit Cost	Setup Cost	Lead Time
Production	30	f_P	3
Repair	Table Below	f_R	2
Disassembly	Table Below	f_M	4
Components Procurement	100	f_C	5
Disposal	5		
Lost Sale	150		

where f_P, f_R, f_M, $f_C \in \{100, 200, 500, 1000, 2000, 5000, 10000\}$

Unit Recovery Cost	Repair	Remanufacturing
Quality Level 1	$160 * P_R$	$130 * P_M$
Quality Level 2	$160 * P_R$	$130 * P_M$
Quality Level 3	$160 * P_R$	$110 * P_M$
Quality Level 4	$160 * P_R$	$90 * P_M$
Quality Level 5	$135 * P_R$	$70 * P_M$
Quality Level 6	$110 * P_R$	$50 * P_M$
Quality Level 7	$85 * P_R$	$30 * P_M$
Quality Level 8	$60 * P_R$	$30 * P_M$
Quality Level 9	$35 * P_R$	$30 * P_M$
Quality Level 10	$10 * P_R$	$30 * P_M$

where P_R, $P_M \in \{0.5, 0.75, 1, 1.25, 1.5\}$

Inventories	Unit Holding Cost
Repair inventory	4
Disassembly inventory	3
Component inventory	5
Final product inventory	6

Another changing parameter which is not shown in Table 5.1 is the return quantity. To understand the effects of various return to demand ratios, the quantity of return is also changed. In order to do this, the fuzzy quantities of returns with different quality levels for all periods within the considered time horizon are scaled in a way that the overall defuzzified sum of them will match a specific percentage of the overall defuzzified sum of demands of all periods. This percentage is chosen randomly from $\{0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2\}$. An example of the return quantities, for the value 0.5 is shown in Table 5.2.

Table 5.2 Fuzzy demand and fuzzy quantities of returned products at 50% of demand with different quality levels

Time Period	Demand	Quantity of returned products with quality level									
		1	2	3	4	5	6	7	8	9	10
1	[0,0,0,0]	[4,4,4,4]	[4,5,5,6]	[3,3,3,3]	[2,2,2,2]	[3,3,3,3]	[4,4,4,4]	[4,4,4,4]	[2,3,3,4]	[4,4,4,4]	[4,5,5,6]
2	[0,0,0,0]	[3,3,3,3]	[2,2,2,2]	[4,5,5,6]	[3,4,4,5]	[5,5,5,5]	[3,4,4,5]	[3,4,4,5]	[3,3,3,3]	[2,2,2,2]	[3,3,3,3]
3	[0,0,0,0]	[3,4,4,5]	[3,3,3,3]	[5,5,5,5]	[3,3,3,3]	[3,3,3,3]	[4,5,5,6]	[2,2,2,2]	[2,2,2,2]	[3,3,3,3]	[4,4,4,4]
4	[0,0,0,0]	[4,4,4,4]	[5,5,5,5]	[2,3,3,4]	[2,2,2,2]	[2,2,2,2]	[4,4,4,4]	[3,4,4,5]	[2,3,3,4]	[4,4,4,4]	[2,2,2,2]
5	[0,0,0,0]	[2,2,2,2]	[5,5,5,5]	[2,2,2,2]	[3,4,4,5]	[4,5,5,6]	[4,5,5,6]	[1,1,1,1]	[5,5,5,5]	[4,5,5,6]	[3,4,4,5]
6	[0,0,0,0]	[3,4,4,5]	[2,3,3,4]	[3,4,4,5]	[3,3,3,3]	[3,3,3,3]	[3,3,3,3]	[2,2,2,2]	[4,4,4,4]	[5,5,5,5]	[3,3,3,3]
7	[0,0,0,0]	[2,3,3,4]	[3,4,4,5]	[5,5,5,5]	[2,2,2,2]	[5,5,5,5]	[3,3,3,3]	[3,4,4,5]	[5,6,6,7]	[3,3,3,3]	[4,4,4,4]
8	[0,0,0,0]	[3,3,3,3]	[3,4,4,5]	[2,2,2,2]	[3,4,4,5]	[2,3,3,4]	[3,3,3,3]	[2,2,2,2]	[3,3,3,3]	[3,3,3,3]	[4,4,4,4]
9	[95,98,102,105]	[6,6,6,6]	[3,3,3,3]	[4,4,4,4]	[3,3,3,3]	[4,5,5,6]	[2,2,2,2]	[3,3,3,3]	[2,3,3,4]	[3,3,3,3]	[3,3,3,3]

Continued on next page

Table 5.2 – *Continued from previous page*

10	[96,100,100,104]	[3,3,3,3]	[4,4,4,4]	[3,4,4,5]	[3,3,3,3]	[3,3,3,3]	[3,3,3,3]	[3,4,4,5]	[2,2,2,2]	[3,4,4,5]	[4,4,4,4]
11	[95,99,101,105]	[3,4,4,5]	[5,5,5,5]	[3,3,3,3]	[3,4,4,5]	[4,4,4,4]	[2,2,2,2]	[3,3,3,3]	[3,3,3,3]	[5,5,5,5]	[2,2,2,2]
12	[95,98,102,105]	[5,6,6,7]	[2,2,2,2]	[3,4,4,5]	[3,3,3,3]	[3,3,3,3]	[4,4,4,4]	[3,4,4,5]	[3,3,3,3]	[2,2,2,2]	[4,5,5,6]
13	[98,99,101,102]	[3,3,3,3]	[3,3,3,3]	[5,5,5,5]	[3,3,3,3]	[3,4,4,5]	[4,5,5,6]	[3,3,3,3]	[4,4,4,4]	[2,2,2,2]	[3,3,3,3]
14	[96,98,102,104]	[4,4,4,4]	[4,5,5,6]	[3,4,4,5]	[4,4,4,4]	[4,4,4,4]	[4,4,4,4]	[4,4,4,4]	[8,8,8,8]	[3,3,3,3]	[4,4,4,4]
15	[98,98,102,102]	[3,3,3,3]	[4,5,5,6]	[3,3,3,3]	[5,5,5,5]	[2,2,2,2]	[4,5,5,6]	[3,3,3,3]	[3,4,4,5]	[3,4,4,5]	[4,4,4,4]
16	[99,100,100,101]	[3,3,3,3]	[3,3,3,3]	[3,4,4,5]	[3,3,3,3]	[4,4,4,4]	[4,4,4,4]	[1,1,1,1]	[4,4,4,4]	[4,4,4,4]	[3,3,3,3]
17	[98,98,102,102]	[4,5,5,6]	[4,4,4,4]	[5,5,5,5]	[4,4,4,4]	[2,2,2,2]	[3,4,4,5]	[5,5,5,5]	[2,2,2,2]	[4,4,4,4]	[4,5,5,6]
18	[97,99,101,103]	[3,3,3,3]	[4,4,4,4]	[2,3,3,4]	[4,5,5,6]	[3,4,4,5]	[4,4,4,4]	[3,3,3,3]	[3,3,3,3]	[2,3,3,4]	[2,2,2,2]
19	[96,98,102,104]	[3,3,3,3]	[2,2,2,2]	[4,4,4,4]	[2,2,2,2]	[3,3,3,3]	[2,2,2,2]	[2,2,2,2]	[3,3,3,3]	[3,3,3,3]	[4,4,4,4]
20	[97,99,101,103]	[4,5,5,6]	[3,3,3,3]	[3,4,4,5]	[4,4,4,4]	[4,4,4,4]	[3,3,3,3]	[3,4,4,5]	[3,3,3,3]	[3,4,4,5]	[3,3,3,3]

Continued on next page

Table 5.2 – *Continued from previous page*

21	[99,100,100,101]	[2,2,2,2]	[5,5,5,5]	[4,4,4,4]	[3,4,4,5]	[3,3,3,3]	[1,1,1,1]	[2,2,2,2]	[4,4,4,4]	[5,5,5,5]	[3,4,4,5]
22	[95,99,101,105]	[4,4,4,4]	[3,4,4,5]	[2,3,3,4]	[3,3,3,3]	[2,3,3,4]	[2,2,2,2]	[4,4,4,4]	[5,5,5,5]	[4,4,4,4]	[4,4,4,4]
23	[95,99,101,105]	[4,4,4,4]	[3,3,3,3]	[5,5,5,5]	[4,4,4,4]	[3,3,3,3]	[3,4,4,5]	[3,4,4,5]	[3,4,4,5]	[2,2,2,2]	[4,4,4,4]
24	[97,99,101,103]	[2,2,2,2]	[4,5,5,6]	[4,5,5,6]	[3,4,4,5]	[3,3,3,3]	[4,4,4,4]	[3,3,3,3]	[3,3,3,3]	[3,3,3,3]	[5,5,5,5]
25	[96,98,102,104]	[3,4,4,5]	[5,6,6,7]	[3,3,3,3]	[3,3,3,3]	[5,5,5,5]	[3,4,4,5]	[4,4,4,4]	[2,3,3,4]	[3,3,3,3]	[2,2,2,2]

The total demand in Table 5.2 is [1642,1679,1721,1758] and the total quantity of returns is [818,891,891,964] which is roughly around 50% of the total demand. The defuzzified values are 1700 for total demand and 891 for total quantity of returns.

5.3 Fuzzy controller

A fuzzy controller with two outputs, including Repair Quality Threshold and Remanufacturing Quality Threshold, is proposed. The objective of this controller is to determine suitable quality thresholds, based on certain network parameters, to be used as the parameters for the Phase 1 of the model described in the previous chapter. The controller is to be designed in such a way as to determine quality thresholds for a variety of network parameters. This will allow the insight gathered through the design of the controller to be applicable to all networks with similar network structure. In this section different elements of the controller, including its outputs, inputs, rules and its implementation procedure will be discussed.

5.3.1 Quality threshold bases and outputs

Controller has two outputs including Repair Quality Threshold QT_R, and Remanufacturing Quality Threshold QT_M. Repair quality threshold determines the quality levels of returned products which are adequate for repair (i.e. the quality levels are higher or equal to the repair quality threshold), while remanufacturing quality threshold is used to find the suitable quality levels for remanufacturing (i.e. returned products with quality of equal or higher than the remanufacturing quality threshold but lower than the repair quality threshold will be remanufactured).

The fuzzy controller uses bases for the quality thresholds which act as the starting points. These bases include Repair Quality Threshold Basis and Remanufacturing Quality Threshold Basis. Fuzzy rules will determine if the actual quality thresholds should be higher or lower than the bases.

Each of these two thresholds has three membership functions, including *Increase*, *AsIs* and *Decrease* defined in the same way. For each of these outputs, the controller needs to decide whatever to use the same value as the basis (AsIs), use a higher value (Increase) or use a lower value (Decrease). The definitions of the membership functions are as illustrated in Figures 5.1 and 5.2.

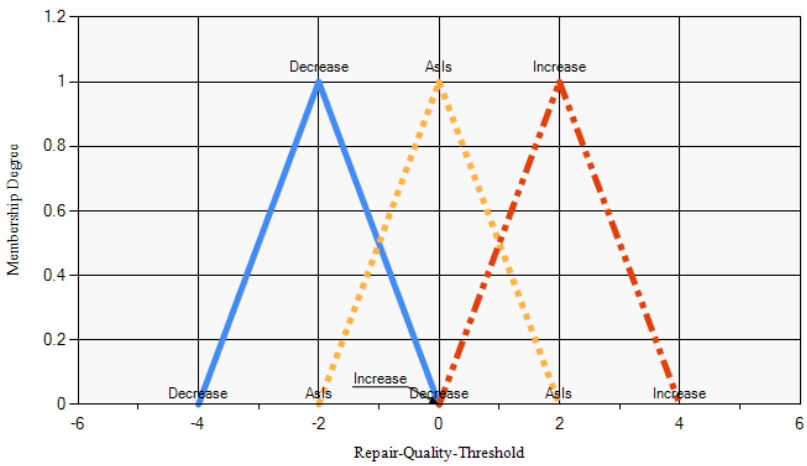

Figure 5.1 Membership functions of Repair Quality Threshold output.

Please note that if in the result of controller, the remanufacturing quality threshold

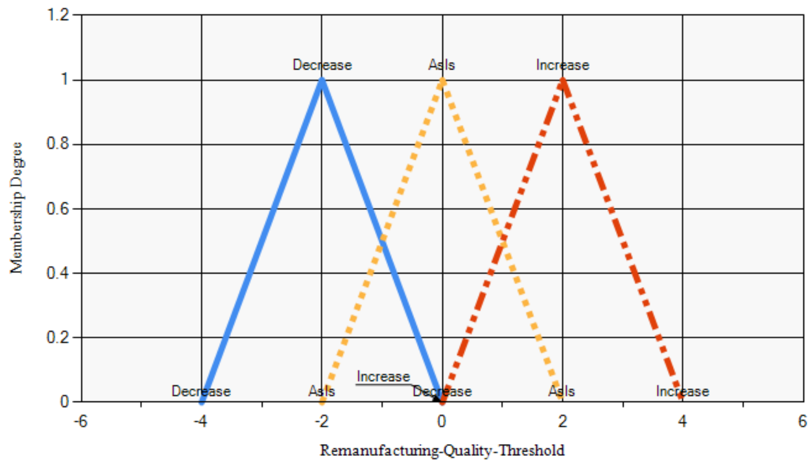

Figure 5.2 Membership functions of Remanufacturing Quality Threshold output.

is higher than the repair quality threshold, it can be interpreted as the controller's decision to stop remanufacturing. So, in this case, the repair quality threshold remains the same (as that is the level controller sees fit for repair), but the remanufacturing quality threshold is set to be equal to the repair quality threshold which means that there is no remanufacturing.

It is worth mentioning that during the following experiments, the repair quality threshold 6 and remanufacturing quality threshold 5 have been used as the starting point for the bases. As it will be discussed in Section 5.3.4, these bases will be updated as necessary.

5.3.2 Input ratios

Each logistics network has its own unique set of parameters which can vary substantially from other networks. Since a general method is needed which can be applied to various networks, instead of using absolute values of parameters which are not easy to interpret without knowing the full context, ratios of related parameters are used.

In this analysis, we are mostly focusing on recovery and production costs which are either variable (per unit) or fixed (setup) costs. This attention is justified as recovery/production costs affect the suitability of a particular recovery route more than other parameters, such as inventory costs and lead times. To analyse production/recovery costs, five ratios are utilized:

1) Repair to New Unit Cost Ratio: this is the ratio between the unit cost of repair, at the Repair Quality Threshold Basis and the unit cost of new production, including the procurement of a new component and the production unit cost. This ratio determines, just from the unit cost perspective, is the repair more economical than the new production or not.

2) Remanufacturing to New Unit Cost Ratio: Similar to the above, this is the ratio between the unit cost of remanufacturing at the Remanufacturing Quality Threshold Basis and the unit cost of new production.

3) Repair to Disassembly Setup Ratio: This is the ratio between the setup cost of the repair and the disassembly setup cost. This ratio helps in comparing the two routes in terms of the setup costs.

4) Disassembly to Procurement Setup Ratio: Similar to the above, this is the ratio between the disassembly setup cost and the component procurement setup cost. This ratio also compares remanufacturing and forward setup routes.

5) Return to Demand Ratio: This is the ratio between the total defuzzified recoverable returns and the total defuzzified demand. It is worth noting that recoverable returns are those returns with quality equal or higher than the Remanufacturing Quality Threshold Basis.

5.3.3 Input membership functions and fuzzy rules

In this section, the inputs of the controller are discussed further, especially by introducing the membership functions and rules used in the controller. Scatter plots are used to summarise the test dataset results for each input which will give the justification for the proposed rules. It is worth noting that each of the proposed rules has only one input; hence, they have been grouped based on the input used and analysed accordingly.

Scatter plots

The relationships between the five ratios defined in Section 5.3.2 and the desirable quality thresholds is analysed using scatter plots. For each of the ratios, two scatter plots are presented, one for the repair quality threshold and another for the remanufacturing quality threshold. They show the most desirable values for these thresholds. The more desirable a value of the quality threshold is for a specific value of the ratio, the bigger the relevant point on the scatter plot[1] should be. Therefore, the size of each scatter point needs to be determined to demonstrate its desirability.

Before determining the sizes, the desirability of a threshold policy for data point n is defined, as follows:

$$DS\left(n, QT_R, QT_M\right) = \frac{C_n^*}{C_n^{P(QT_R,QT_M)} - C_n^*} \tag{5.1}$$

where $DS\left(n, QT_R, QT_M\right)$ is the desirability of the policy $P(QT_R, QT_M)$ for the data point n, $C_n^{P(QT_R,QT_M)}$ is the cost incurred using the policy $P(QT_R, QT_M)$ for data point n and C_n^* is the cost incurred using the best threshold policy for the data point n. The higher (lower) the difference between the costs, the lower (higher) is the desirability. Please note that for policies with optimal cost, the denominator of the formula is zero. However, this will not cause a problem as the desirability formula is replaced within the harmonic average formulation in formulas 5.3, 5.4 and 5.7.

Also, data points which have the same value for the ratio under consideration need

[1] We will call these points on the scatter plot simply *scatter points*. These are different from 'data points' which refer to the test dataset.

to be separated into subsets. These subsets are defined as follows:

$$N'(v) = \{n | n \in 1..N, g(n) = v\} \tag{5.2}$$

where $N'(v)$ is the subset of data points which have the value v of the ratio under consideration, while function $g(n)$ returns the value of the ratio for data point n at the current quality threshold bases, N is the number of data points. Please note that the quality threshold bases used for determining the ratios are respectively 6 and 5 for repair and remanufacturing quality threshold bases.

As mentioned, for each ratio two scatter plots will be drawn; one for the repair quality threshold and one for the remanufacturing quality threshold. Regarding the former, the size of each scatter point for the repair quality threshold QT_R with ratio value v, defined as $SR(v, QT_R)$, is calculated as:

$$SR(v, QT_R) = \frac{|N'(v)|\, QT_R}{\sum\limits_{n \in N'(v)} \sum\limits_{QT_M=1}^{QT_R} \frac{1}{DS(n, QT_R, QT_M)}} \tag{5.3}$$

The provided formula calculates the average desirability of any policy with the repair quality threshold QT_R for all data points that have the value v of the ratio under consideration using the harmonic average formula. For this purpose, all remanufacturing thresholds that are less than or equal to the repair quality threshold need to be considered. The denominator is simply the total number of desirability terms in the nominator, used to calculate the average of all desirability values. $|N'(v)|$ is the number of relevant data points and QT_R is the number of threshold policies with repair quality threshold QT_R. Harmonic average is chosen as it puts more emphasis on low desirability (high error) policies, compared to the arithmetic mean that focus more on the high desirability (low error) policies.

Similarly, the size of each scatter point for the remanufacturing quality threshold QT_M with ratio value v, represented as $SM(v, QT_M)$, is determined as follows:

$$SM(v, QT_M) = \frac{|N'(v)|\,(Q - QT_M + 1)}{\sum\limits_{n \in N'(v)} \sum\limits_{QT_R = QT_M}^{Q} \frac{1}{DS(n, QT_R, QT_M)}} \tag{5.4}$$

The provided formula is also calculating the average desirability value for all relevant policies and data points. It is worth noting that $(Q - QT_M + 1)$ is the number of policies with remanufacturing quality threshold QT_M.

A series of lines are drawn within each scatter plot that represent the most desirable quality threshold values over the range of the ratio values. To show the most desirable ratio values, these lines connect the weighted average of the respective quality threshold that is weighted by the corresponding point size. We will call these lines *weighted average lines*. The weighted averages for repair and remanufacturing quality thresholds, $Avg_R(v)$ and $Avg_M(v)$ respectively, are calculated as follows:

$$Avg_R(v) = \frac{\sum\limits_{QT_R=1}^{Q} QT_R * SR(v, QT_R)}{\sum\limits_{QT_R=1}^{Q} SR(v, QT_R)} \tag{5.5}$$

$$Avg_M(v) = \frac{\sum\limits_{QT_M=1}^{Q} QT_M * SM(v, QT_M)}{\sum\limits_{QT_M=1}^{Q} SM(v, QT_M)} \tag{5.6}$$

Also, a series of points at the top of the scatter plots are presented. These points show the density of data points for certain ratio values and called *density points*. Essentially, they are presenting the number of data points with the respective ratio value of v, or $|N'(v)|$.

It is worth noting that in the case when there is a repair only policy, both quality thresholds should be the same and therefore, the choice of remanufacturing quality threshold depends on the repair quality threshold. To determine that, the policies in the $SM(v, QT_M)$ formula is limited to the repair only ones (i.e. $QT_R = QT_M$) and a point size for those policies is calculated as follows:

$$SM^*(v, QT_M) = \frac{|N'(v)|}{\sum\limits_{n \in N'(v)} \frac{1}{DS(n, QT_M, QT_M)}} \tag{5.7}$$

The formula provides an average value of desirability for repair only threshold policies. As for each ratio value v, only one threshold policy is considered, the total number of terms in the nominator is $|N'(v)|$, for which the formula provides an average.

If the new size $SM^*(v, QT_M)$ is larger than the size $SM(v, QT_M)$, the scatter point will be marked with a star symbol to represent the fact that a repair only policy is more desirable.

Repair to new unit cost ratio

Repair to new unit cost ratio provides a good estimate of the suitability of the Repair Quality Threshold Bases for the repair route. In this context, 'new' refers to forward route production unit cost, which is procurement plus production unit costs. A low ratio shows that repairing is more economical while a high value means that the forward production is more attractive. Also, rules for this input are only concerned about the repair quality threshold. This means that any increase in this ratio should be followed by an increase in the repair quality threshold and, similarly, any decrease should be treated by an adequate decrease in the repair quality threshold.

This can be confirmed in the scatter plots shown in Figure 5.3.

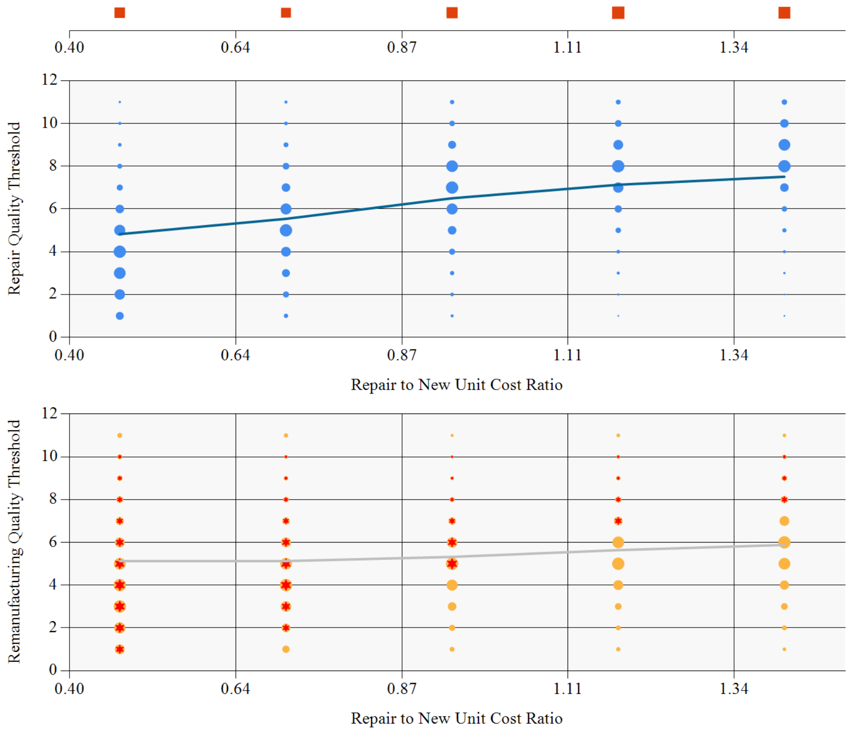

Figure 5.3 Scatter plots of quality thresholds for Repair to New Unit Cost Ratio input.

Based on the provided scatter plots in Figure 5.3, value of 0.8 has been chosen as the medium value for the membership functions, because the weighted average lines on the scatter plot for repair quality threshold is near the basis value of 6 which means that the controller should keep the repair quality threshold 'as is' around this value. Overall, a steady increase in repair quality threshold from 0.50 to 1.10 can be observed while the change stagnates outside this range. This is expected as, the more expensive repair is in comparison to the forward production, it is less desired and hence, the repair quality threshold should be increased. It is worth mentioning that the density points are almost uniform and, hence, all ratios are of the same importance.

In the case of repair to new unit cost ratio, changes in remanufacturing quality threshold are mostly dependant on changes in repair quality threshold. This is mainly because when repair quality threshold is low, networks typically adapt a repair only policy which has the same remanufacturing quality threshold as the repair quality threshold. When the repair quality threshold is increased, greater prevalence of mixed policies is expected which can have higher remanufacturing quality thresholds as the remanufacturing quality threshold is not limited to the respective repair quality threshold any more. Hence, for the repair to new unit cost ratio, it has been decided not to include remanufacturing quality threshold in the fuzzy rules. Because the changes in repair quality thresholds can already create the desirable effect on the remanufacturing quality threshold.

Membership functions for this ratio are shown in Figure 5.4.

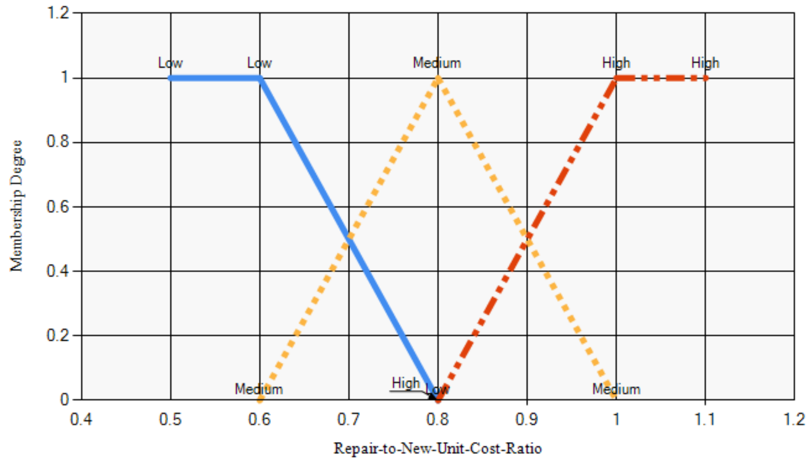

Figure 5.4 Membership functions of Repair to New Unit Cost Ratio input.

Here are the proposed rules:

```
1) IF Repair-to-New-Unit-Cost-Ratio is High THEN
     Repair-Quality-Threshold is Increase
```

When the repair to new ratio is high, the repair quality threshold needs to be increased to reduce the quantity of repair.

```
2) IF Repair-to-New-Unit-Cost-Ratio is Medium THEN
     Repair-Quality-Threshold is AsIs
```

When the repair to new ratio is medium, repair quality threshold should stay the same.

```
3) IF Repair-to-New-Unit-Cost-Ratio is Low THEN
     Repair-Quality-Threshold is Decrease
```

If the repair to new ratio is low, the repair quality threshold should be decreased to increase the quantity of repair.

Remanufacturing to new unit cost ratio

Remanufacturing to new unit cost ratio controls the remanufacturing quality threshold in the same way that repair to new unit cost ratio provides for the repair quality threshold. To be more precise, an increase (decrease) in this ratio should be responded to by an decrease (increase) in the remanufacturing volume. However, changing the remanufacturing quantity can be done by changing both quality thresholds. An increase in the remanufacturing volume can be achieved by increasing the repair quality threshold while decreasing the remanufacturing quality threshold. On the other hand, decrease in the remanufacturing volume can be achieved by decreasing the repair quality threshold and increasing the remanufacturing quality threshold. This hypothesis can be validated using the scatter plots represented in Figure 5.5.

From Figure 5.5 it is observed that repair quality threshold has a decreasing trend while remanufacturing quality threshold has an increasing trend which is expected, as described. Ratio value of 1 is used as the medium value as it represents the scenario in which remanufacturing cost is the same as the forward production cost and also, for this value, both quality thresholds are close to their bases. It is worth mentioning that the remanufacturing quality thresholds is increasing within the range of 0.5 to 1.1 while it stagnates for greater values. This is simply because at around the value of 1.1 the repair only policies perform considerably better than the other policies and any increase in the ratio will not have any effect. The density points are again of similar size, so the ratio values are also of homogeneous importance.

Membership functions for this input are defined in Figure 5.6.

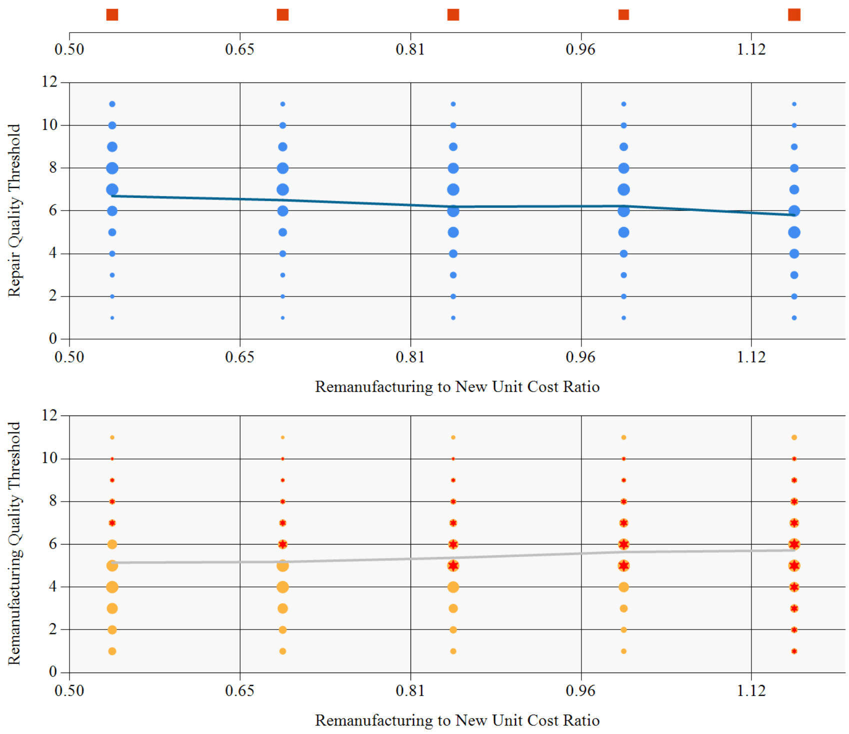

Figure 5.5 Scatter plots of the quality thresholds for Remanufacturing to New Unit Cost Ratio input.

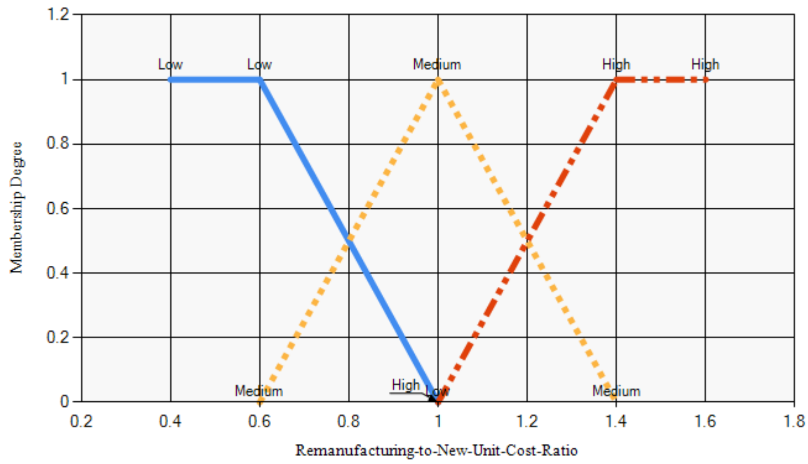

Figure 5.6 Membership functions of Remanufacturing to New Unit Cost Ratio input.

The rules for this input is as follows:

```
4) IF Remanufacturing-to-New-Unit-Cost-Ratio is High THEN
     Remanufacturing-Quality-Threshold is Increase AND
     Repair-Quality-Threshold is Decrease
```

When remanufacturing to new unit cost ratio is high, remanufacturing is not desirable any more. In order to reduce the remanufacturing quantity which possibly leads to a repair only policy, the remanufacturing quality threshold is increased and the repair quality threshold is decreased.

```
5) IF Remanufacturing-to-New-Unit-Cost-Ratio is Medium THEN
     Remanufacturing-Quality-Threshold is AsIs AND
     Repair-Quality-Threshold is AsIs
```

If the ratio is medium then both the repair and remanufacturing quality thresholds should be as they are.

```
6) IF Remanufacturing-to-New-Unit-Cost-Ratio is Low THEN
     Remanufacturing-Quality-Threshold is Decrease AND
     Repair-Quality-Threshold is Increase
```

When remanufacturing to new unit cost ratio is low, more remanufacturing activity is preferred which means that the remanufacturing quality threshold must decrease and repair quality threshold increase.

Repair to disassembly setup ratio

Repair to disassembly setup ratio compares the setup cost of the repair route with the disassembly setup cost. The higher the ratio is, repair is less attractive in comparison with the disassembly while the lower it is, repair becomes more economical. This can be seen in the scatter plots represented in Figure 5.7.

Since repair to disassembly setup ratio has a large range of values, the scatter plot is presented in logarithmic scale. This will bring the range of values to a more manageable scale and also it is easier to compare the values.

From Figure 5.7 an increasing trend in repair quality threshold is seen while remanufacturing quality threshold is unaffected. The density points show a concentration of data points in the range of 0.50 to 5. Hence, these ratios seem to be more

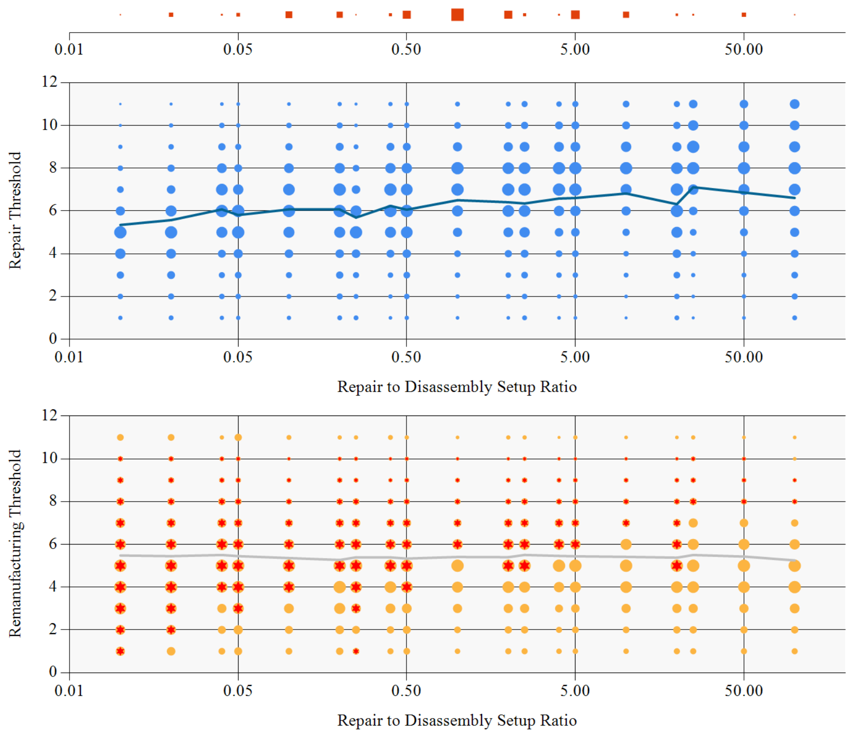

Figure 5.7 Scatter plots of the quality thresholds for Repair to Average Setup Ratio input.

important than the ratios outside this range. For the membership functions, a medium about the ratio 3 and a spread of 5 is chosen which puts the medium membership function in the range of -2 to 8 and also determines low and high membership functions. Although a negative ratio is not meaningful, the membership functions are assumed to be symmetrical and, as a result, the low end of the membership function is dependent on the centre and spread of the membership function. This definition however is compatible with the observable increasing trend in repair quality threshold from the beginning of the range that stagnates starting around the ratio 8.

The Figure 5.8 shows the definitions of membership functions for this input.

Here are the rules:

```
7) IF Repair-to-Disassembly-Setup-Ratio is High THEN
     Repair-Quality-Threshold is Increase
```

If the ratio is high, repair route is less economical and hence, the repair quality

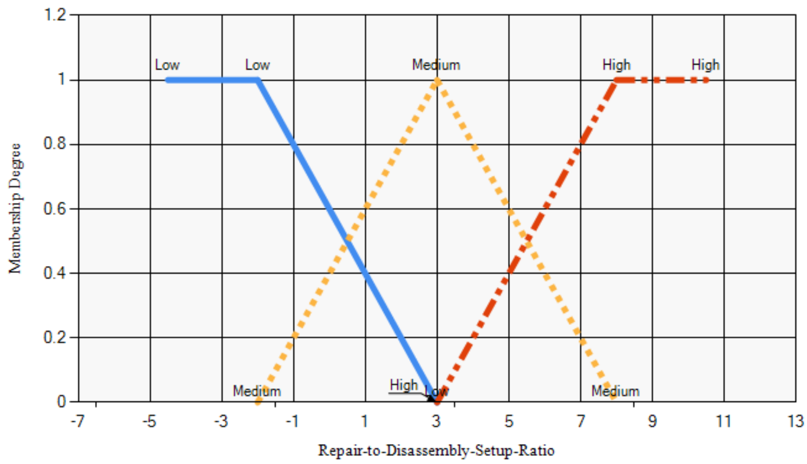

Figure 5.8 Membership functions of Repair to Average Setup Ratio input.

threshold should be increased.

```
8) IF Repair-to-Disassembly-Setup-Ratio is Medium THEN
     Repair-Quality-Threshold is AsIs
```

When the ratio is medium, the repair quality threshold stays as is.

```
9) IF Repair-to-Disassembly-Setup-Ratio is Low THEN
     Repair-Quality-Threshold is Decrease
```

When the ratio is low, repair becomes more cost effective and the repair quality threshold should be decreased.

Disassembly to procurement setup ratio

Similarly, disassembly to procurement setup ratio compares the disassembly setup cost to the procurement setup cost. This is important in deciding between the remanufacturing and forward routes, as the production setup cost incurs in both routes. Particularly this ratio is relevant to determining the remanufacturing quality threshold. It is expected that the higher (lower) the ratio is, the higher (lower) the remanufacturing quality threshold should be. The scatter plot is shown in Figure 5.9. Similar to the repair to disassembly setup ratio, disassembly to procurement setup ratio is also presented in logarithmic scale.

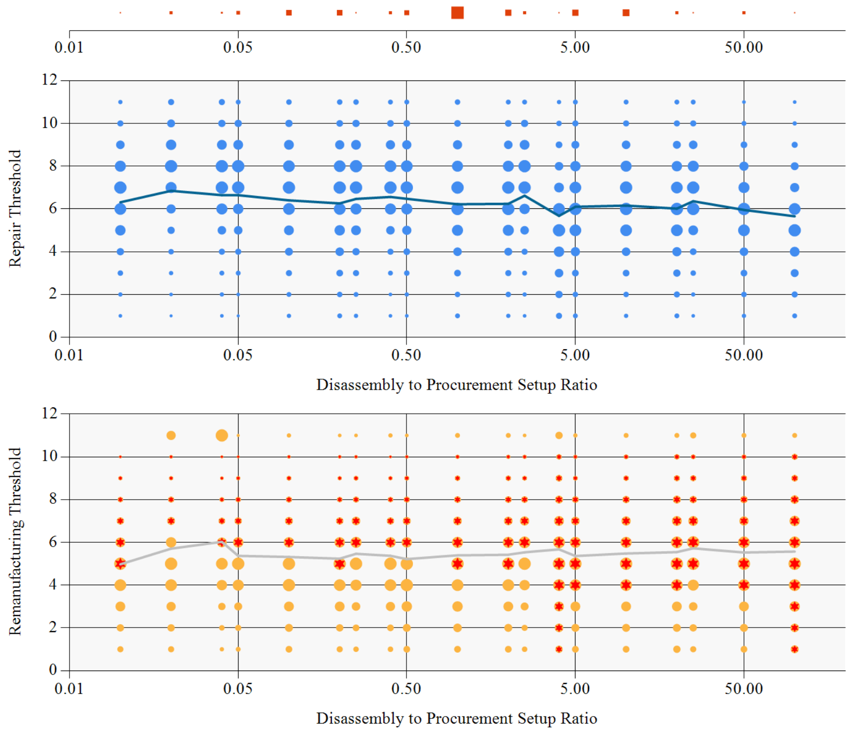

Figure 5.9 Scatter plots of the quality thresholds for Disassembly to Procurement Setup Ratio input.

The trends in Figure 5.9 are less obvious than the similar trends for other ratios and both thresholds seem to stay around the same value. Also, the density points has a higher concentration around the ratio 1. The fluctuations that can be seen in both plots can be attributed to the lower number of data points outside the ratio 1 which increases the effect of randomness of the data on the results of scatter plots.

A higher prevalent of repair only policies starting from the ratio 3 can be observed. Also, an increasing trend in the remanufacturing quality threshold starting from ratio 0.5 to ratio 3 exists. For the membership functions, medium is defined to be around 0.5 with its high end reaching the ratio 3 which puts the low end to be at ratio -2. This also determines the other membership functions.

Definitions of the membership functions are illustrated in Figure 5.10.

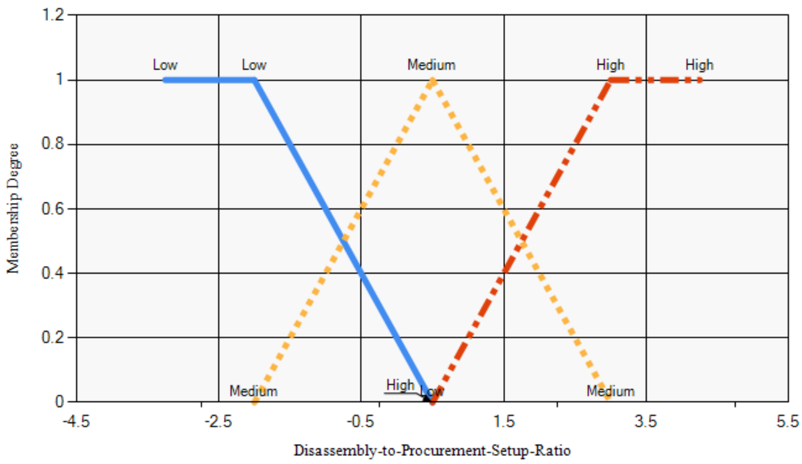

Figure 5.10 Membership functions of Disassembly to Procurement Setup Ratio input.

The relevant rules are as follows:

```
10) IF Disassembly-to-Procurement-Setup-Ratio is High THEN
      Remanufacturing-Quality-Threshold is Increase
```

As observed, the desirable result for high disassembly to procurement setup ratio is to increase the remanufacturing quality threshold and to have a higher prevalent of repair only policies. By using this rule, when the ratio is high, the remanufacturing quality threshold should be increased to reduce the quantity available for remanufacturing or even eliminate this route altogether.

```
11) IF Disassembly-to-Procurement-Setup-Ratio is Medium THEN
      Remanufacturing-Quality-Threshold is AsIs
```

When the ratio is medium, the remanufacturing quality threshold stays the same.

```
12) IF Disassembly-to-Procurement-Setup-Ratio is Low THEN
      Remanufacturing-Quality-Threshold is Decrease
```

If the ratio is low, the remanufacturing quality threshold is decreased which will increase the use of remanufacturing route. However, this rule is rarely used, as the membership definition of low is placed mostly in the negative range. This is however in line with the observation from the scatter plot, as barely any decrease in remanufacturing quality threshold is observed in the lower values of the ratio.

Return to demand ratio

Return quantity affects the volume of products available for recovery which is very influential in the network performance. The scatter plot is shown in Figure 5.11.

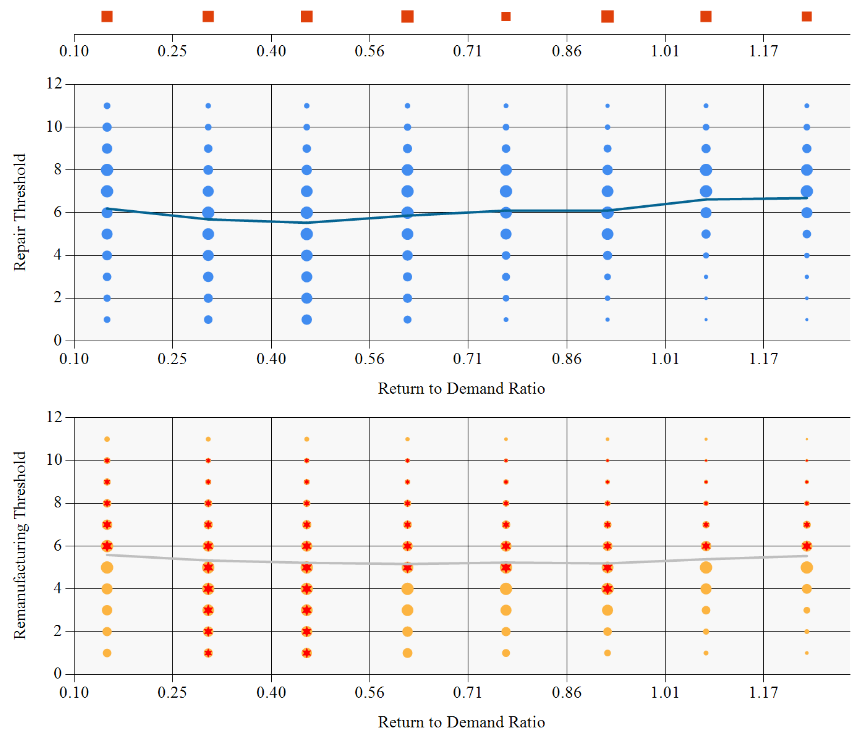

Figure 5.11 Scatter plots of the quality thresholds for Remanufacturing to Average Setup Ratio input.

Medium is chosen to be around ratio 1.1 as, at this ratio, the demand is fully satisfied through return. The extra 10% of returned products helps alleviate the adverse effects of uncertainty and timing of returns on supply. Above the value 1.1, from 1.10 to 1.25, an increase in both the repair and remanufacturing quality thresholds can be seen in Figure 5.11. This is to avoid excess amount of returns to be recovered. For the lower values, from 0.1 to 0.9, we see a slight decrease in remanufacturing quality threshold while repair quality threshold fluctuates but overall stays the same. A decrease in remanufacturing quality threshold will increase the volume of products to be recovered which can, to some extent, restore the volume of recovery to the desirable higher level. It is worth mentioning that the density points are of similar size and, hence, the ratios are of equal importance.

For the very low value of return to demand ratio, around 0.15, a slight increase in both quality thresholds is observed. This is due to the fact that the highest possible volume of recovery is so low that it is more efficient to rely more on the forward route than the recovery routes.

Membership functions are defined in Figure 5.12.

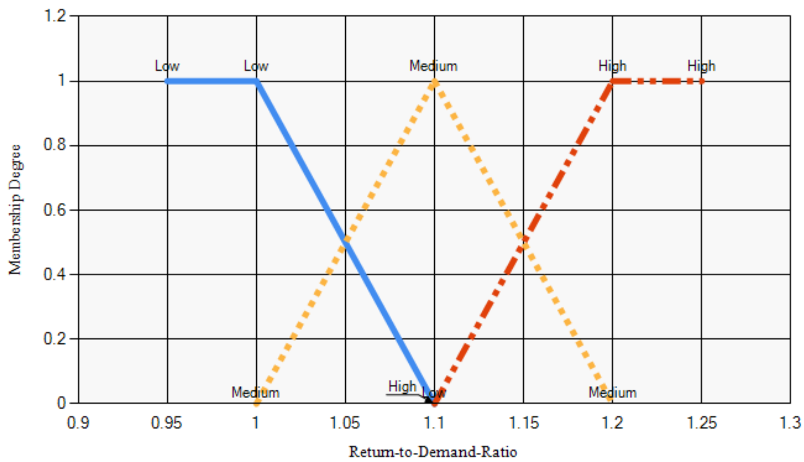

Figure 5.12 Membership functions of Return to Demand Ratio input.

Here are the proposed rules:

```
13) IF Return-to-Demand-Ratio is High THEN
 Remanufacturing-Quality-Threshold is Increase AND
 Repair-Quality-Threshold is Increase
```

When very high quantity of return, compared to demand, is available, we can rely solely on repairing the highest quality of returned products which is the cheapest possible option. Therefore, the remanufacturing quality threshold is increased to stop remanufacturing. Also, the repair quality threshold is increased to reduce unwanted returned products.

```
14) IF Return-to-Demand-Ratio is Medium THEN
Remanufacturing-Quality-Threshold is AsIs AND
Repair-Quality-Threshold is AsIs
```

For medium ratio, the repair and remanufacturing quality thresholds stay the same.

```
15) IF Return-to-Demand-Ratio is Low THEN
Remanufacturing-Quality-Threshold is Decrease
```

If the return quantity is lower compared to demand, it is reasonable to decrease the remanufacturing quality threshold to try to keep the recovery volume at an acceptable level that can justify the setup costs for recovery facilities.

5.3.4 Calculating outputs

The proposed controller needs to determine two parameters: 1) Repair Quality Threshold QT_R and, 2) Remanufacturing Quality Threshold QT_M. The controller has two corresponding fuzzy outputs which are determined as follows. First step is to defuzzify the outputs, the Weighted Average Defuzzification Technique is used here (as described in Chapter 3). The resulting value is relative to the quality threshold bases and determine how much the bases need to be changed. Therefore, after defuzzification, the relative outputs of the controller need to be added to the quality threshold bases. Then, the defuzzified outputs are rounded to the nearest integer (as threshold should always be integer). However, after this step, it is possible for the outputs to be unacceptable; this is the case when the repair quality threshold is lower than the remanufacturing quality threshold. In this case, the remanufacturing quality threshold is set to the repair quality threshold. This will guarantee that the repair route will receive the products deemed to be suitable for repair while the remanufacturing route will not be used.

Each run of the proposed controller decides how much the two quality thresholds based need to changed. Therefore, in order to reduce the impact of the initial quality threshold bases, it can be justified to run the controller more than once, and using updated bases as new bases. Ideally, the controller should converge to certain quality threshold bases, but it is not always feasible. So, in addition to the first termination criteria which refers to the convergence of the threshold bases, we have to have a second termination criteria which is the maximum number of iterations the controller is ran consequently, called *Maximum Iterations (MI)*. The lower the MI is, the effects of the initial bases are more prominent. Also, the higher the MI is, it will take longer to calculate the results. In section 5.5, we will compare the results of this controller with different values for the MI.

5.4 Benchmark policies

The proposed controller is compared to other policies for determining thresholds in order to evaluate its performance. Two types of policies are used for comparison: 1) Fixed Threshold Policies: these are the policies which are using fixed repair and remanufacturing quality thresholds, independent from the network parameters. 2) Policy based on the Cost Estimate Comparison: This policy uses a comparison of the estimated total cost of the network, for all quality thresholds policies, to find the most suitable quality thresholds policy. Each of these types of policies is discussed here.

5.4.1 Fixed threshold policies

Fixed threshold policies are the policies where both the repair and remanufacturing quality thresholds are fixed in advance, independent of the RL parameter values. These policies represent a scenario in which the recovery facilities use the same criteria for separating the returned products into repairable, remanufacturable and disposable products, regardless of the network and the market.

A fixed threshold policy represents a scenario in which the criteria for suitability of returned products for recovery has been traditionally the same in the recovery process and the human experts are reluctant to change the policy in the face of variations to the network parameters. While a fixed policy is, as it will be shown in the next sections, very economically inefficient, it is likely to be used in a RL network as the most convenient option. Such policies have been described in the literature (Dobos and Richter, 2006; Lieckens and Vandaele, 2012; Mukhopadhyay and Ma, 2009; Nenes et al., 2010; Zikopoulos and Tagaras, 2007).

5.4.2 Policy based on the Cost Estimate Comparison

To validate the results of the controller, it is necessary to compare it with different policies. The fixed threshold policies use the same quality thresholds irrespective to network parameters. For example, a fixed threshold policy will have the same thresholds for very cheap and very expensive repair while this is clearly not appropriate. The policy based on the Cost Estimate Comparison (CEC) is introduced here that determines the quality thresholds from network parameters.

The CEC policy is designed as a heuristic for comparing different fixed policies by estimating their corresponding cost incurred. The basis of this policy is to estimate the total cost of repair, remanufacturing and forward-production and then use this cost to calculate, for each fixed policy, the total estimated cost. Based on these results, the policy with the least estimated cost will be chosen, using an exhaustive search of all incurred costs.

The CEC policy requires an exhaustive search over all policies which is more complex than the fuzzy controller. More specifically, this policy requires to be run over all policies while the controller is only needed to be run once. On the other hand, fixed threshold policies are very simple and require almost not additional computation. As a result, it is expected that the CEC policy perform better than the controller while the controller should perform better than the fixed threshold policies.

Estimating the total cost

Calculating the total cost by solving the mixed-integer programming model described in Chapter 4 is a complicated task. Therefore, instead of solving the optimisation model, an estimation is calculated based on some simplifying assumptions.

To reduce the complexity of estimating total costs, we chose not to include the total inventory holding costs. We consider this to be an acceptable compromise, as the total inventory holding costs are usually smaller than other cost elements, as the results in Table 4.7 suggest. However, to estimate the share of setup costs in the estimated total costs, the size of batches for each activity are estimated using the Economic Order Quantity (EOQ) formula. The EOQ formula considers the inventory holding costs to determine the optimal batch sizes. Hence, the holdings costs are indirectly included in the estimation of the total costs.

The algorithm uses an exhaustive search to find the best combination of quality thresholds which minimise the estimated total cost. The quality thresholds in the following formulas act as parameters and each formula is applied separately for each combination of the quality thresholds.

One important step to estimate the total cost is to approximate the number of setups needed for each of the production and recovery activities including repair, disassembly, production and procurement. Here, we apply the EOQ formula which is used widely to determine the optimal batch size in a simple production environment

(Harris, 1990). The formula for determining the optimal batch size is as follows:

$$Q^*(D,K,h) = \sqrt{\frac{2DK}{h}} \tag{5.8}$$

where Q^* is the optimal batch size, D is the total demand for the time horizon, K is the setup cost and h is the unit holding cost for the same time horizon.

Also, the number of setups NS in a multi-period environment can be approximated using the following formula:

$$NS(D,K,h) = \begin{cases} 0 & \text{if } D = 0 \\ min\left(T, max(1, \lceil \frac{D}{Q^*(D,K,h)} \rceil)\right) & \text{if } D > 0 \end{cases} \tag{5.9}$$

where T is the number of periods.

It is assumed that each new batch will incur a setup cost. Next, the number of setups for the repair activity should be estimated. To do this, the following values for h, D and K are replaced in Equations 5.8 and 5.9 to calculate NS_R:

$$h_1 = T(h_S - h_R)$$
$$D_1 = \sum_{q=QT_R}^{Q} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right)$$
$$K_1 = f_R$$
$$NS_R = NS(D_1, K_1, h_1)$$

where h_S and h_R are respectively the per period unit holding costs of final and repair inventories, $\widetilde{BI}(t,q)$ represents the quantity of return of quality q at time period t and f_R is the setup cost for repair. Please note that h_1 represents the difference in holding cost between the repair and the final inventory, for the whole time horizon, as this is the effective holding cost payable for the repair activity. Also, D_1 is the part of demand which is satisfied through repair which we assume here to be the same as quantity of all the repairable products.

Using the calculated NS_R above, the estimated total cost of repair includes the total unit cost of repair, which is dependent on quality, and an estimation of the setup costs for repair. This is formulated as follows:

$$TC_R = \sum_{q=QT_R}^{Q} \left(c_R(q) \sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) + f_R NS_R$$

where TC_R is the estimated total cost incurred with the threshold policy (QT_R, QT_M), $c_R(q)$ is the unit cost of repair for quality level q and NS_R is the estimated number of setups.

Also, similar to the repair activity, the number of disassembly setups (NS_D) is estimated using Equations 5.8 and 5.9 and the following values:

$$h_2 = T(h_C - h_D)$$
$$D_2 = \sum_{q=QT_M}^{QT_R - 1} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right)$$
$$K_2 = f_D$$
$$NS_D = NS(D_2, K_2, h_2)$$

where h_C and h_D are respectively the per period unit holding costs of components and disassembly inventories, and f_R is the setup cost for disassembly. Similar to h_1, h_2 is also the difference between the holding cost for components inventory and the disassembly inventory, as that is the effective holding cost that is going to be paid as a result of the disassembly setup.

Similarly, the estimated total cost of remanufacturing can be calculated as follows:

$$TC_M = \sum_{q=QT_M}^{QT_R-1} \left(c_M(q) \sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) + f_D NS_D$$

where TC_M is the estimated total cost of remanufacturing incurred with the threshold policy (QT_R, QT_M), $c_M(q)$ is the unit cost of disassembly for quality level q and f_D is the setup cost for disassembly. Please note that the unit production cost and setup cost of production activity will be included later.

The total cost of disposal can be calculated as follows:

$$TC_G = \sum_{q=1}^{QT_M-1} \left(c_G \sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right)$$

where TC_G is the estimated total cost of disposal with the threshold policy (QT_R, QT_M) and c_G is the unit cost of disposal.

Next, we need to consider two scenarios: 1) to produce the remaining demand from new components and 2) to consider the rest of the demand not satisfied by recovery as lost sale. The total cost of production and lost sale which are incurred in these scenarios are calculated. Finally, a scenario is chosen that minimises the total cost of the network.

For the first scenario, when all the remaining demand will be satisfied by forward production which uses newly procured components, the number of setups for procurement and production needs to be estimated. The following estimates of the number of setups of procurement (NS_C) are used:

$$
\begin{aligned}
&h_3 = Th_C \\
&D_3 = \sum_{t=1}^{T} Defuzz(\widetilde{D}(t)) - \sum_{q=QT_M}^{Q} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) \\
&K_3 = f_C \\
&NS_C = NS(D_3, K_3, h_3)
\end{aligned}
$$

where f_C is the setup cost for procurement and D_3 is the remaining demand. The following estimates are used to calculate the number of setups for production in this scenario (NS_{P_1}):

$$
\begin{aligned}
&h_4 = T(h_S - h_C) \\
&D_4 = \sum_{t=1}^{T} Defuzz(\widetilde{D}(t)) - \sum_{q=QT_R}^{Q} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) \\
&K_4 = f_P \\
&NS_{P_1} = NS(D_4, K_4, h_4)
\end{aligned}
$$

where f_P is the setup cost for production. Please note that D_4 is the total demand reduced by the part of demand that has been satisfied through repair. This recognises the fact that both new production and remanufacturing route use the production activity.

Using the calculated number of setups for procurement and production, the total cost of production in the first scenario is determined:

$$
\begin{aligned}
TC_P = \quad & c_C \left[\sum_{t=1}^{T} Defuzz(\widetilde{D}(t)) - \sum_{q=QT_M}^{Q} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) \right] + f_C NS_C + \\
& c_P \left[\sum_{t=1}^{T} Defuzz(\widetilde{D}(t)) - \sum_{q=QT_R}^{Q} \left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q)) \right) \right] + f_P NS_{P_1}
\end{aligned}
$$

where TC_P is the estimated total cost of production and procurement and c_C and

c_P are respectively the unit cost of the new component procurement and production.

For the second scenario, when all the remaining demand not satisfied by recovery is considered lost, the number of setups for production (NS_{P_2}) which is incurred by remanufacturing activities is calculated. Following estimates are used:

$$h_5 = T(h_S - h_C)$$
$$D_5 = \sum_{q=QT_M}^{QT_R-1}\left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q))\right)$$
$$K_5 = f_P$$
$$NS_{P_2} = NS(D_5, K_5, h_5)$$

where D_5 is the demand that is satisfied through remanufacturing. h_5 is also the difference between the holding costs of final and components inventories.

Using the estimated number of setups for production, the total cost of production and lost sale in the lost sale scenario is calculated as follows:

$$\begin{aligned} TC_L = \quad & c_P\left[\sum_{t=1}^{T} Defuzz(\sum_{q=QT_M}^{QT_R-1}\left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q))\right)\right] + f_P NS_{P_2} + \\ & c_L\left[\sum_{t=1}^{T} Defuzz(\widetilde{D}(t)) - \sum_{q=QT_M}^{Q}\left(\sum_{t=1}^{T} Defuzz(\widetilde{BI}(t,q))\right)\right] \end{aligned}$$

where TC_L is the total cost of production and lost sale.

Finally, the total cost of the network can be calculated using the following formula:

$$TC = TC_R + TC_M + TC_G + min(TC_P, TC_L)$$

5.5 Analysis of results

In this section, performance of the fuzzy controller will be compared with the performance of two benchmark policies defined in Section 5.4. For this comparison, the 1000 random networks in the test dataset described in Section 5.2 are used and each threshold policy, including the threshold policy generated by the fuzzy controller and two benchmark threshold policies, is applied to all networks. Finally, the total cost incurred under each threshold policy is compared with the best policy for each of the networks under consideration and a measure of accuracy, called Mean Percentage Error (MPE) is calculated for each threshold policy.

5.5.1 Measure of accuracy of threshold policy: Mean Percentage Error (MPE)

To measure the performance of each threshold policy, the Mean Percentage Error (MPE) is used. The main advantage of using the MPE is that the resulting error is not calculated in terms of the actual value of the cost function and it allows us to compare network costs for networks with different parameter values. It might be interesting to note that, since the cost is compared to the cost achieved under the best threshold policy, the error can never become negative. It means that there is no difference between the MPE and the Mean Absolute Percentage Error (MAPE).

MPE is the average percentage of error of the threshold policy, in comparison with the best threshold policy for each of the networks under consideration. The best policy is found by an exhaustive search of all quality thresholds for each network (data point).

First the average cost of the network C_n is calculated by dividing the total cost of the network incurred during the time horizon by the defuzzified total quantity of demand as follows:

$$C_n = TC_n/Defuzz(\sum_{t=1}^{T} \widetilde{D}(t))$$

where C_n is the average cost of the network incurred by using the threshold policy under consideration, TC_n is the total cost incurred in the network and $\widetilde{D}(t)$ is the fuzzy demand in period t.

The average cost is calculated for a set of test data $X_1, ..., X_N$; where in the experiments carried out $N = 1000$. Based on the average cost, the MPE for the threshold policy, is calculated as follows:

$$MPE = \frac{100\%}{N} \sum_{n=1}^{N} \frac{C_n(X_n) - C_n^*(X_n)}{C_n^*(X_n)}$$

where $C_n^*(X_n)$ is the best average cost for the network (best among all the combinations of quality thresholds) for the test data X_n.

5.5.2 Implementation

The software and hardware environment used is the same as described in Chapter 4 which also includes an implementation of the Mamdani-type fuzzy controller proposed.

To give an idea about the computational performance of the algorithms, the optimisation algorithm which is described in the previous chapter took 6269 minutes ($\sim$104 Hours) to calculate the optimal results for the test dataset with 1000 data points. For the same data set and using the optimal results from the previous step, the controller (with MI=5) takes 12.13 seconds, while the CEC policy takes 12.35 seconds to calculate their results which is to find the threshold policy for each data point. The fixed policies are instantaneous as the threshold policy is fixed in advance.

5.5.3 Comparison of threshold policies' performances

To have a better understanding of the fuzzy controller's performance, it will be compared with the fixed thresholds policies and the CEC policy. In this comparison, the fuzzy controller with various maximum iterations (MI) values will be used.

In order to compare all the threshold policies, 1000 data points presented in Section 5.2 are used. For each data point, the average costs of the network incurred under all of the threshold policies are determined by using the fuzzy optimisation model, described in the Chapter 4. The cost incurred under the threshold policy is determined for all data points and it is compared with cost incurred under the best policy. This gives the error rate of the threshold policy for all the data point. The MPE measure is used to calculate the overall error of each policy.

The results are presented in Table 5.3.

Table 5.3 Comparison of the accuracy of the fuzzy controller and the benchmark threshold policies using the MPE measure

Policy	MPE	Policy	MPE
CEC	2.12%	P(10,3)	19.91%
FC[MI=5]	4.32%	P(9,2)	20.24%
FC[MI=10]	4.33%	P(10,6)	20.37%
FC[MI=20]	4.33%	P(7,1)	21.32%
FC[MI=1]	4.61%	P(6,1)	21.50%
P(7,5)	10.47%	P(8,1)	21.67%
P(6,5)	10.75%	P(5,1)	22.50%
P(8,5)	10.81%	P(4,2)	23.12%
P(8,4)	11.03%	P(10,2)	24.05%
P(7,4)	11.20%	P(9,8)	24.20%
P(7,6)	11.42%	P(9,1)	24.50%

Continued on next page

Table 5.3 – *Continued from previous page*

P(6,6)	11.66%	P(10,7)	24.73%
P(6,4)	12.16%	P(11,4)	24.85%
P(8,6)	12.74%	P(11,5)	25.43%
P(5,5)	13.08%	P(3,3)	26.47%
P(5,4)	13.54%	P(4,1)	26.65%
P(9,4)	13.58%	P(11,3)	26.96%
P(9,5)	13.60%	P(3,2)	26.96%
P(8,3)	13.60%	P(11,6)	27.50%
P(7,3)	13.78%	P(9,9)	28.27%
P(7,7)	14.24%	P(10,1)	28.32%
P(6,3)	14.54%	P(10,8)	30.29%
P(8,7)	15.80%	P(3,1)	30.62%
P(9,6)	15.83%	P(11,2)	31.68%
P(5,3)	15.94%	P(11,7)	31.76%
P(9,3)	16.03%	P(2,1)	33.36%
P(7,2)	17.65%	P(2,2)	34.14%
P(10,4)	17.67%	P(10,9)	35.71%
P(8,2)	17.71%	P(11,1)	36.32%
P(10,5)	17.89%	P(11,8)	37.10%
P(6,2)	18.07%	P(10,10)	39.11%
P(4,4)	18.95%	P(11,11)	40.58%
P(5,2)	19.25%	P(1,1)	40.87%
P(8,8)	19.43%	P(11,9)	43.03%
P(4,3)	19.50%	P(11,10)	47.62%
P(9,7)	19.70%		

As it is clear from Table 5.3, $P(7,5)$ is the best performing among all the fixed policies considering all data points, followed by $P(6,5)$, $P(8,5)$, $P(8,4)$ and so on. The fuzzy controller performs considerably better than all the fixed threshold policies but the $MI = 5$ seems to be the best number of iterations for the fuzzy controller, followed very closely by $MI = 10$ and $MI = 20$.

The CEC policy performs the best among all quality threshold policies, even the fuzzy controller. However, this policy uses an exhaustive search among all possible quality thresholds. We will use a learning method, based on Genetic Algorithm (GA), to improve the performance of the fuzzy controller with the aim of surpassing the CEC policy's performance. The developed GA will be described in Chapter 6.

5.6 Sensitivity analysis

In this section, sensitivity of the quality thresholds policies' performance to the main parameters is presented. Several network parameters with a wide range of possible values, that will be considered, represent a huge multi-dimensional space which is not possible to analyse completely and thoroughly. Therefore, the results are generated and compared for each of the parameters separately. The parameters considered include the production, repair, disassembly and components procurement setup costs; average repair and disassembly unit costs; and the return to demand ratio.

The main test dataset, introduced in Section 5.2, is used which includes a combination of changes in all parameters simultaneously. The average performance for each value of the relevant parameter under consideration is reported using the MPE. These analyses give us a general picture of how each of the respective parameters influences the network and its performance.

It is worth noting that in each of the following charts, the percentage of error is reported for the fuzzy controller with different MI values, the CEC policy and also for those fixed thresholds policies that, at least for one value, are producing the best MPE among all fixed threshold policies. The non-optimal fixed quality thresholds policies are removed to simplify the charts and remove irrelevant details.

5.6.1 Average repair cost

In Figure 5.13, sensitivity of average repair costs, with respect to different quality levels, is presented. The average repair cost for values defined in Table 5.1 is $107.5 * P_R$ where $P_R \in \{0.5, 0.75, 1, 1.25, 1.5\}$. Hence, the range of values for average repair cost is from $107.5 * 0.5 = 53.75$ to $107.5 * 1.5 = 161.25$.

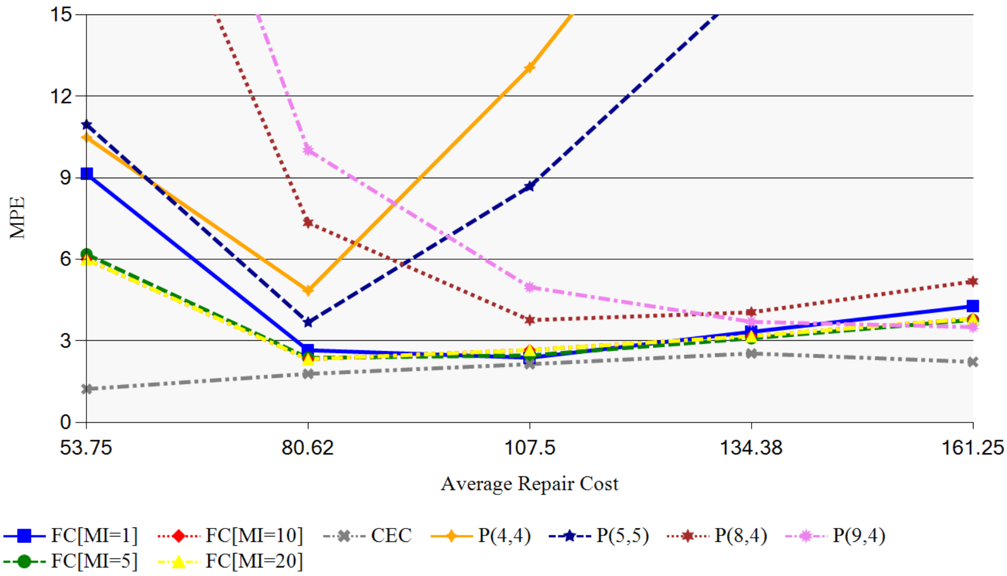

Figure 5.13 Sensitivity analysis of the average repair cost on the test dataset.

From Figure 5.13 it can be concluded that the controller usually performs better than the fixed quality threshold policies and slightly worse than the CEC policy while its performance usually decreases on both ends of the range of average repair cost values. To address this, more membership functions for repair cost can be useful, for example, using 'High' and 'Very High' instead of just 'High'. Especially, fixed policies almost always perform worse than the fuzzy controller but, for higher repair cost, their performance starts to improve and occasionally surpasses that of the fuzzy controller. Please note, the performance of the fixed quality thresholds, $P(4,4)$, $P(5,5)$, $P(8,4)$ and $P(9,4)$ are presented because they are the fixed policies that are performing the best among all fixed policies, at least for one value within the range of average repair cost values.

5.6.2 Average disassembly cost

Sensitivity of the networks performance to average disassembly cost, with respect to different quality levels, is shown in Figure 5.14. Please note that the range of average disassembly cost for values defined in Table 5.1 is from $70 * 0.5 = 35$ to $70 * 1.5 = 105$. Figure 5.14 clearly illustrate the effectiveness of the fuzzy controller approach. It provides lower error rates than all the fixed quality threshold policies for all cases; while, the controller is only surpassed by the CEC policy. Also, as it is obvious from the chart, the repair only policy $P(6,6)$ performs the best among all fixed policies for the highest average value, while $P(9,3)$, which implies a high number of returned products to be remanufactured, is the most suitable fixed policy for the lowest value of the average disassembly cost. Although, the fuzzy controller's performance is considerably more homogeneous with a slight increase in error in both extremes of the average disassembly cost which, similarly to the average repair cost, can be alleviated by a finer definition of the relevant membership functions and rules.

5.6.3 Repair setup cost

Sensitivity of the networks' performance to the repair setup cost is shown in Figure 5.15. The range of repair setup cost, based on values in Table 5.1, is from 100 to 10000. Superiority of the fuzzy controller to the fixed policies is demonstrated. As it can be seen from the chart, the lower the repair quality threshold is, the more the fixed policy is susceptible to increase in the repair setup cost. This behaviour can be observed in the difference between $P(7,5)$ and $P(8,5)$. On the other hand, the remanufacturing only policy $P(11,4)$ decreases in MPE as a result of increases in the repair setup cost because, in this scenario, the remanufacturing becomes more

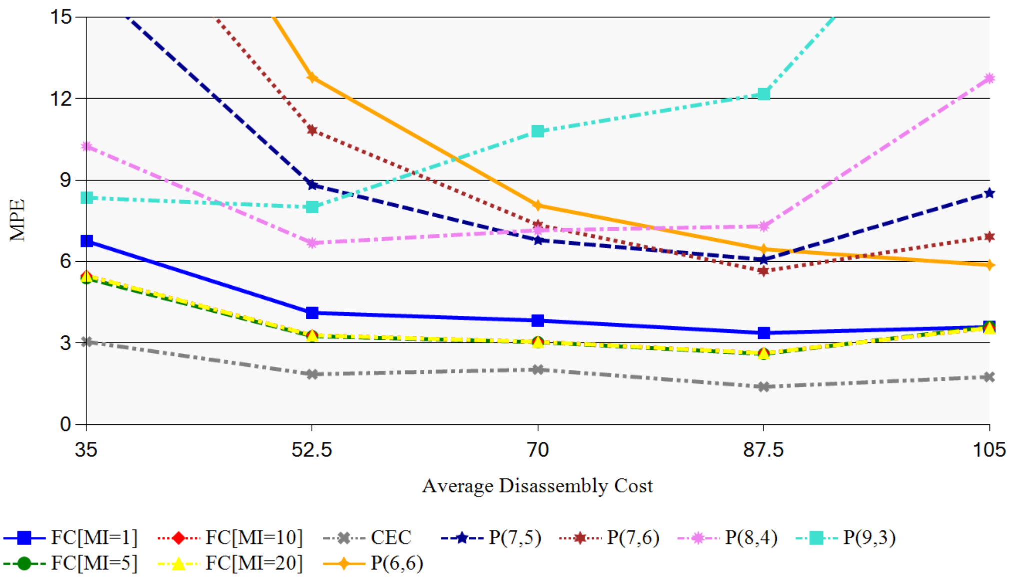

Figure 5.14 Sensitivity analysis for the average disassembly cost on the test dataset.

attractive compared to the repair. However, the fuzzy controller's performances have little variations with regard to the incurred repair setup cost. Interestingly it is hard to improve the fuzzy controller by changing the rules or membership functions relevant to repair setup cost.

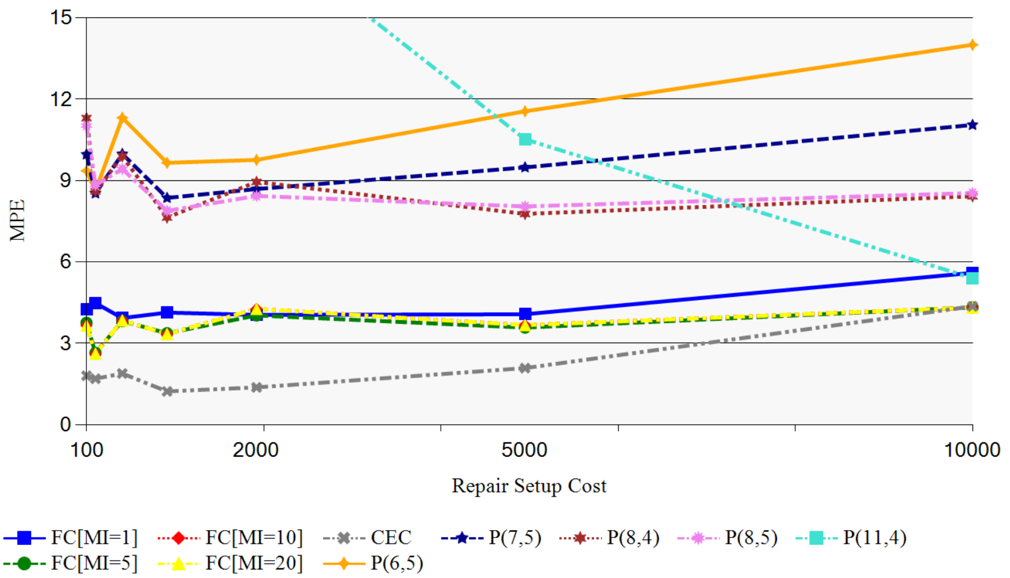

Figure 5.15 Sensitivity analysis for the repair setup cost on the test dataset.

5.6.4 Disassembly setup cost

Sensitivity of the networks' performances to the disassembly setup cost is shown in Figure 5.16. The range of disassembly setup cost values is from 100 to 10000. Like in the cost of repair setup cost, the fuzzy controller's performances rarely vary. In the case of the fixed quality threshold policies, repair only policies such as $P(6,6)$ improve its performance as a result of increases in the disassembly setup cost, while policies which imply higher remanufacturing activities, such as $P(8,4)$ and $P(7,5)$, worsen.

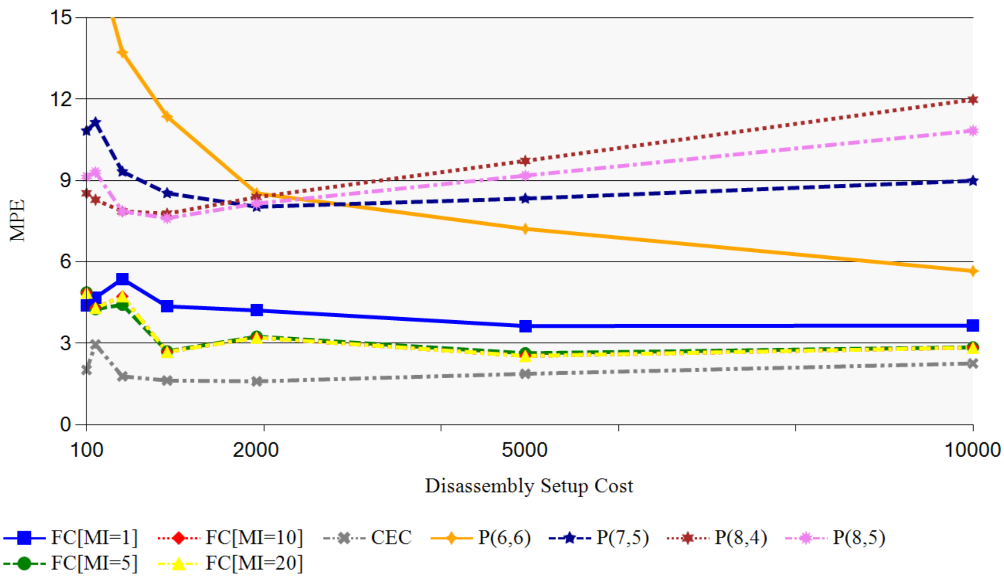

Figure 5.16 Sensitivity analysis of the disassembly setup cost on the test dataset.

5.6.5 Production setup cost

Sensitivity of the network performance to production setup cost is shown in Figure 5.17. Similar to other setup costs, the range of values is from 100 to 10000. High values of production setup cost, in the range of 5000 to 10000, reduce the difference in performance of the policies. This can be attributed to an increase in total network costs for all policies, as a result of higher setup costs, and lower relative difference among the total networks costs of the policies. To clarify, production setup cost affects all policies because it also affects the forward route, unlike repair and disassembly setup costs that can only affect policies that use their corresponding routes. Regarding the fixed policies, repair only policy $P(6,6)$ is the best option for high values of production setup cost in the range of 5000 to 10000, while those with high remanufacturing activities, such as $P(8,4)$ and $P(8,5)$, are the best for low values of production setup cost in the range of 100 to 500.

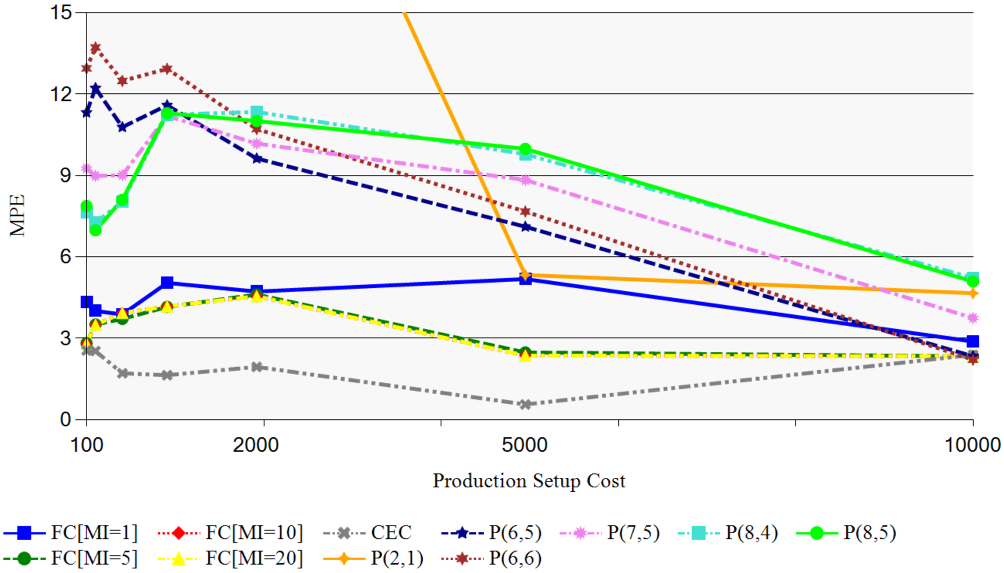

Figure 5.17 Sensitivity analysis of the production setup cost on the test dataset.

5.6.6 Components procurement setup cost

Figure 5.18 represents the sensitivity of the different quality threshold policies to the component procurement setup costs, in the presence of variations of the component procurement setup cost from 100 to 10000. For high values of components procurement setup cost, in the range of 5000 to 10000, the overall cost of the networks is increased for all policies, as the increase in procurement setup cost affects the forward route. Hence, a relatively lower difference between the policies is observed. Also, when the procurement setup cost is high, at the value of 10000, $P(4,3)$ provides high level of recovery, which reduces the need for forward production and procurement. So, it performs better for higher values of procurement setup cost. In the case of low procurement setup cost, in the range of 100 to 500, policy $P(8,5)$ is the best, as there is an increase in the production activity which is shared with remanufacturing route. Therefore, the remanufacturing route is used more often.

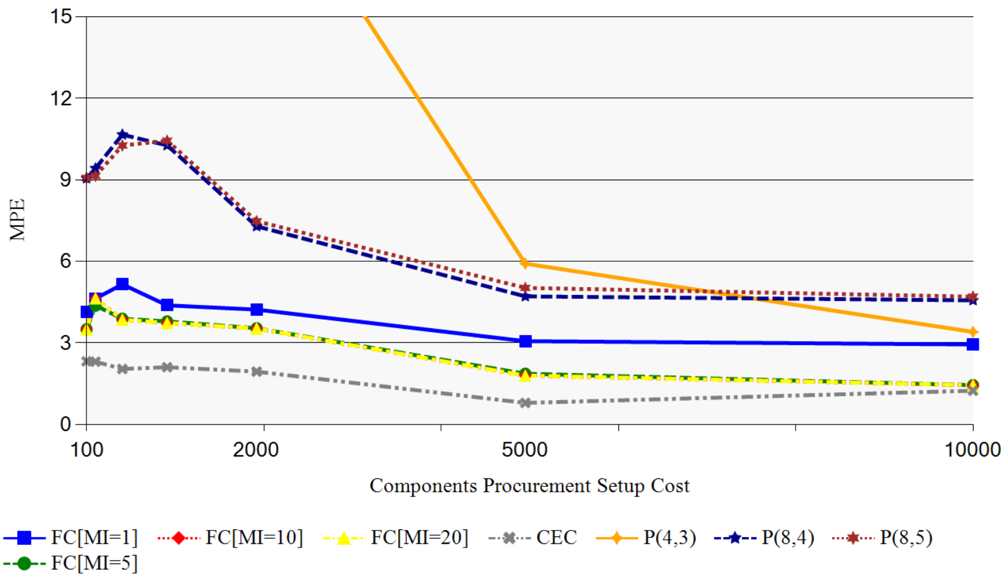

Figure 5.18 Sensitivity analysis of the components procurement setup cost on the test dataset.

5.6.7 Return to demand ratio

Figure 5.19 shows the sensitivity of quality threshold policies' performance to the return to demand ratio in the range of 0.25 to 2.00. For all the policies, higher variations in the MPE can be seen for the higher return ratios than the lower ratio values, which is due to the increase in the impact of recovery policies when there are higher return quantities. The fuzzy controller performs better than all fixed threshold policies throughout the range of return to demand ratio values. However, for the higher values in the range of 1 to 1.75, its performances degrade. Fixed policies perform similarly for low ratios, but as the ratio increases from 1 to 1.75, i.e. there is a high quantity of returned products, policies with a high number of remanufacturing activity and low remanufacturing quality threshold in particular, such as $P(8,4)$ and $P(8,5)$ gain an advantage over other fixed policies, because, by an increase in remanufacturing, recovery only policies are possible and the forward route can be eliminated. However, at return to demand ratio of 2 it can be seen that $P(8,5)$ gains an advantage over $P(8,4)$, as it is now possible to rely on higher quality returns and dispose the returned products with quality level 4.

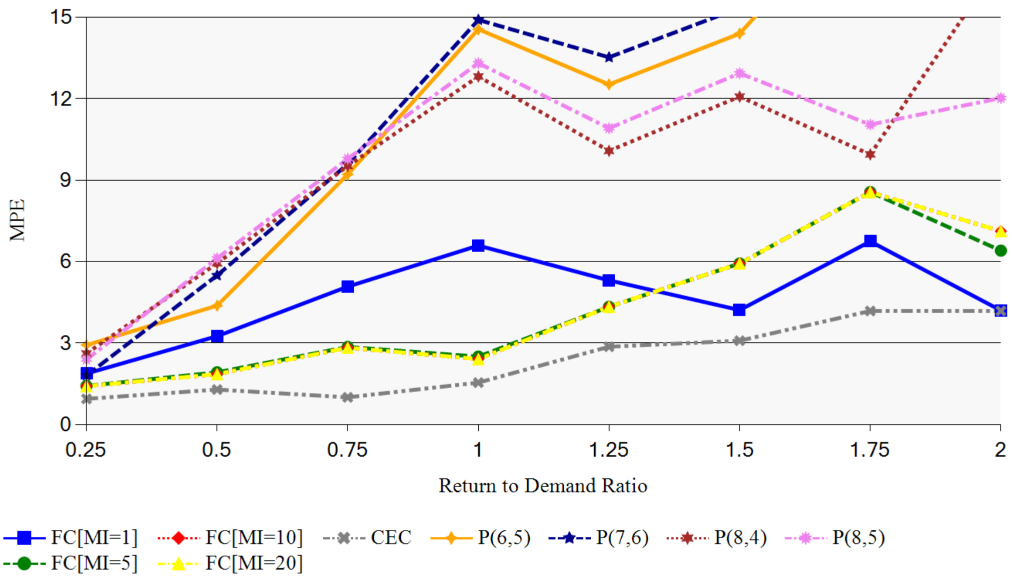

Figure 5.19 Sensitivity analysis of the return ratio on the test dataset.

5.7 Summary and conclusions

A fuzzy controller has been introduced in this chapter to determine the quality thresholds that are used in the RL optimisation model. The proposed controller uses some of the important ratios of network parameters as inputs, such as repair to new unit cost ratio, remanufacturing to new unit cost ratio, repair to disassembly setup ratio, disassembly to procurement setup ratio and return to demand ratio. Fuzzy rules and membership functions are defined for each of these inputs. The fuzzy controller performance is tested and compared with the benchmark policies, including fixed threshold policies and the CEC policy. Test dataset is generated with random variations of network parameters and used in the analysis of the different quality threshold policies. Finally, sensitivity analyses are carried out to better understand the effectiveness of the fuzzy controller and benchmark policies, depending on some of the relevant network parameters.

Using fuzzy sets theory in this thesis has one general advantage. With a lack of statistical data, fuzzy numbers can be used to model input parameters, providing the experts' knowledge of the uncertainty in the parameter's values. Furthermore, use of fuzzy control in this chapter is especially beneficial in gaining an insight into the relationship between various RL parameters and the quality thresholds and their impact on RL performance. This insight can be used to implement efficient recovery policies in RL networks.

The results obtained show that the fuzzy controller provide significant improvements over any fixed quality threshold policies. However, the more complex CEC policy performs better than the fuzzy controller while it requires an exhaustive search of all possible combinations of quality thresholds which is not needed for the fuzzy controller. In the next chapter, we will look into improving the fuzzy controller using Genetic Algorithm with the aim of having a more effective approach to determine the quality thresholds.

Chapter 6

Genetic-Fuzzy Tuning of Fuzzy Rules

6.1 Introduction

One of the main advantages of fuzzy controllers is the ability to capture the experts' knowledge as natural language IF-THEN expressions which constitute the fuzzy rule base. However, there is always a potential to use available data to improve the fuzzy controller in an automatic way with the aim of not only increasing the performance of the controller but also to improve our own understanding of the system that is being controlled. This creates an opportunity for many machine learning methods to either tune the currently established fuzzy controllers or even to go a step further and determine all the controller's parameters from scratch.

In Chapter 5, a fuzzy controller has been defined to determine quality thresholds which set the appropriate quality levels for repair and remanufacturing. The aim of this chapter is to present an approach to tuning this initial controller with the goal of reducing the MPE measure on the same dataset. In order to do so, we utilise genetic fuzzy methods for learning additional rules. Particularly, unlike the initial rules defined in Chapter 5 which are very simple and are only constituted of one term in the premise, we are looking to determine the more complex rules that can have a conjugate of terms in the premise and consequently are more difficult to conclude by human experts. These rules will complement the currently established initial rules in the controller.

As previously mentioned, one of the main advantages of fuzzy control is that it provides a deep insight of its coded knowledge that is directly understandable and/or modifiable by the human experts. Moreover, genetic algorithm is a widely used meta-heuristic method that can provide near optimal solutions in complex search spaces by requiring a performance measure only. Genetic fuzzy methods, that are a combination of the fuzzy control and genetic algorithm, provide both advantages of these methods as well as providing a flexible environment where priori information, such as previously determined fuzzy rules and membership functions, to be utilised in the search (Herrera, 2008). This flexibility has been utilised here in determining additional fuzzy rules for the controller without changing any of the other properties of the controller or initial rules. This allows us to compare the two controllers easily and improve our understanding of the relationships through the fuzzy rules that are readable in natural language.

This chapter is arranged as follows. First, some remarks about the dataset being used are made in Section 6.2. A genetic fuzzy method being used is discussed in details in Section 6.3 and specifics of the genetic operations and parameters are presented. Furthermore, Section 6.4 presents the results of the genetic fuzzy method and the comparison of the results obtained by the fuzzy controller described in Chapter 5. Also, the best rules determined using this method are noted in Section 6.5 and their possible interpretation is discussed. Finally, the chapter is concluded with a summary and a discussion of the outcomes in Section 6.6.

6.2 Dataset

In machine learning methods, a common issue that can invalidate the results is known as *overfitting*. Overfitting happens when the model is only capable of representing the specific data used for learning and not the underlying relationship. To make sure that this issue is not present in an obtained model, two separate datasets are utilised, namely a *training dataset* and *test dataset*. The training dataset is used during the learning process while the test dataset is used after obtaining the model to determine if the model is overfitted. Overfitting can be detected when the error for the test data is considerably higher than for the training data.

In genetic fuzzy tuning of the fuzzy controller, the training dataset is used in calculating the fitness function of the GA. Essentially, the tuned controller is evaluated against the training dataset and the MPE measure achieved using the training dataset is used as the fitness value. Test dataset on the other hand is utilised to validate the resulting controller generated by the GA.

The same dataset, introduced in Chapter 5 is going to be used. Where, the 1000 data points are divided into a training and a test dataset. For this purpose, 700 data points are randomly selected from the 1000 data points to be used as the training data set while the remaining 300 data points are used as the test data set.

Having divided the main dataset into the training dataset and test dataset, the MPE measure can be calculated in three ways: on the training dataset, on the test dataset and on the main dataset which includes both training and test datasets. From now on, these MPE measures will be called *Training MPE*, *Test MPE* and *Overall MPE*, respectively.

6.3 Genetic fuzzy method

In this section, a genetic fuzzy method is introduced in order to tune and possibly improve the fuzzy controller defined in Chapter 5. Specifically, the genetic fuzzy method introduces new rules to the rule base of the initial controller in order to improve the MPE measure of the controller on the training dataset. These new rules differ from the initial rules in that they can have more than one term in the premise

(IF) part. Hence, they can represent more complex relationships between two or more input variable and their influence on the output variables. These rules are inherently more complicated than the simple single input variables defined in Chapter 5 and, therefore they are an example of machine learning methods helping human experts.

The genetic fuzzy technique used is based on Pittsburgh method that is introduced in Chapter 3; the genetic encoding is essentially a list of rules with fixed number of rules. The main advantage of Pittsburgh method is the fact that it can analyse the rules in combination with each other and the benefit of their collaboration is considered. This feature is vital in tuning the fuzzy controller, as the new rules will cover the same input space as the initial rules. In this way the new rules can improve the initial rules in certain parts of the input space.

In this section, first the genetic fuzzy algorithm and its parameters are discussed. Then, the encoding of the new fuzzy rules in the genetic chromosome and the fitness function used are introduced. Finally, genetic operators applied are defined.

6.3.1 Genetic algorithm and parameters

A typical GA, as introduced in Chapter 3, but with minor variations is being used in modifying the fuzzy controller. General features of this GA including its selection method, termination and population are discussed here while problem specific features such as encoding, fitness function and genetic operators are introduced later on.

A population size of 2000 individuals is used in the experiment. In each iteration, genetic operators are randomly processed on the parent generation to create a set of off-springs. The parent population and the new off-springs will collectively create the candidate individuals from which the next generation will be selected.

The selection method is based on a combination of elitism and diversity. To provide a combination of both, a percentage of the new generation is selected from the elite individuals that are performing the best among all candidates, while the rest of the total population is selected using a random process. After some experimentation, it has been decided to select one fifth from the elites while the remaining four fifth are selected randomly. However, the probability of selection for the remaining individuals is not uniform and each individual is assigned a probability of being selected based on its fitness value. The probability formula is as follows:

$$p(x_i) = \frac{(f_{max} - f(x_i)) \, / \, (f_{max} - f_{min})}{\sum_{j=1}^{N} \left(f_{max} - f(x_j)\right) / \, (f_{max} - f_{min})}$$

where x_i is the i-th candidate, $p(x_i)$ is the selection probability of this candidate, f_{max} and f_{min} are the maximum and minimum fitness values of all candidates respectively, $f(x_i)$ is the fitness value for the i-th candidate and N is the number of candidates.

Please note that the described random selection of the rest of the population is without replacement and the same individual, including those in the elite group, cannot be selected multiple times. To avoid replacement, any selected individual is simply removed from the candidates and the probability values described above are recalculated. It is worth mentioning that the comparison method used between individuals considers the fact that the order of fuzzy rules is not important and different orderings of the same rules are considered to be equal.

Regarding the termination of the GA, two criteria are being used, a fixed number of iterations (200 iterations for this experiment) or a certain number of iterations without a change in the best chromosome found (in this experiment, 50 iterations).

6.3.2 Encoding of the rules

In the GA chromosome, 10 fuzzy rules are encoded. These rules will be added to the initial 15 rules which have been defined in Chapter 5. The Pittsburgh approach is used and rules are of fixed length. As described in Chapter 3, each rule is encoded as a list of integer values for all fuzzy inputs and outputs. Each value represents a linguistic variable that corresponds to the respective variable in the rule. A value of zero is also reserved to express the absence of the corresponding variable. An illustration of this encoding is presented in Figure 3.11.

Using this encoding, for the controller introduced in Chapter 5 with five inputs and two outputs, each rule is encoded as 7 integer variables. As each of the fuzzy inputs/outputs have three linguistic variables, these variables can be assigned any value between 0 and 3, virtually carrying two bits of information. Therefore, the genetic representation of each rule carries 14 bits of information and the overall chromosome with 10 encoded rules is equivalent to 140 bits of information.

6.3.3 Fitness function

The main goal of the developed GA is to improve the performance of the controller in determining the quality thresholds. Hence, the fitness function of the GA is based on the MPE measure which needs to be minimised. Essentially, for each set of new rules, a MPE measure is calculated using the results of the fuzzy controller, which includes the initial rules and the new rules, on the training dataset. This MPE measure will be used by GA to select the desirable fuzzy rules.

However, duplicate rules are not desirable. To avoid duplications, a penalty is given to the rules that have the same premise part to any other rule in the controller. The penalty chosen is a prohibitive value of 1 for each duplicate rule in the resulting controller, which is added to the calculated MPE measure.

It is worth mentioning that many genetic fuzzy methods assign a penalty for an incomplete covering of the input space. This penalty should ensure that the defined rules cover the input space completely and no data point is without a relevant rule. However, this is not necessary in our work as the initial rules defined in Chapter 5 already cover the input space completely and, as none of those rules will be removed from the final rules, complete covering is guaranteed.

6.3.4 Genetic operators

Genetic operators play a crucial role in GA and their design can greatly influence the effectiveness of the algorithm. Hence, several criteria need to be considered for choosing these operators. First, these operators should be suitable for the genetic encoding used. For example, in this GA they need to accommodate integer encoding as opposed to the more widely used binary encoding. Also, the specifics of the genetic representation such as location dependence or independence of certain variables of the encoding are important and should be exploited by the genetic operators. For example, for the proposed representation, location of the rules in the encoding does not affect the outcome and therefore several representations i.e., encodings can exist for the same controller. To allow permutations of the same chromosome to be considered, a reordering mechanism described later will be used.

Application of genetic operators is done first by ordering the selected individuals randomly, and then, a random number is compared by the cross over probability to determine if the cross over operator is going to be applied to the individual, one by one. In the case when the cross over operator is applied, the current and the next individuals are going to be used as parents. Similarly, this process is repeated for the mutation operator with the mutation probability. Please note that the mutation operator and a cross over operator can be applied to the same individual.

Genetic operators, including a mutation operator, a classical cross over operator and a rule cross over operator with reordering are introduced in the subsequent sections.

Mutation operator

A very simple mutation operator, based on the one defined in Chapter 3 and shown in Figure 3.7 is utilised. Every time the mutation operator is applied, one integer gene within the chromosome is randomly chosen and it is changed to a random integer value in the respective range between 0 and 3, relevant to this encoding. This operator is similar to the classic mutation operator with the difference that the classic operator is defined for binary encodings while this operators is suitable for integer encodings.

Please note that in this experiment, a mutation probability of 0.3 per individual is used. This means that for each selected individual there is a 30% chance for the mutation operator to be applied. Cordón (2001) considers the typical range of mutation probability to be from 0.001 to 0.02 per bit, which considering that each individual in this experiment is made out of 140 bits, the proposed probability belongs to the typically used range.

Classic cross over operator

A single point cross over operator, as defined in Chapter 3 and presented in Figure 3.6, is used. In this operator, one random cross over point within the chromosome is chosen with uniform probability. Then, one off-spring is created by joining the first part of the first parent and the second part of the second parent while the other off-spring is made of the first part of the second parent and second part of the first parent. In some cases, this operator can be useful; for example, joining the premise part of one rule with the consequence part of another rule.

Rule cross over operator with reordering

The classic cross over operator does not utilise any understanding of the underlying encoding of the chromosome and is solely based on the position (Cordón, 2001). A new rule cross over operator is introduced to provide a more problem specific operator that can help boost the GA performance for this problem. This operator deals with the encoded fuzzy rules as a whole and transfers each rule in the encoding to one of the two off-springs. More specifically, the rules in the two selected parents are joined to have a total of 20 rules. Then, half of these rules are randomly selected to be placed in the first off-spring while the rest will be placed in the second off-spring. An example of this operator is shown in Figure 6.1.

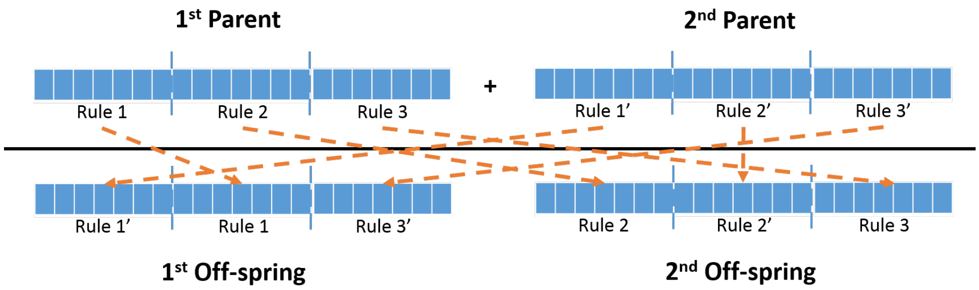

Figure 6.1 An example of the rule cross over operator with reordering.

This operator has two advantages. First, it allows for combining the best rules of the two good controllers to create a better controller without changing the definition of the rules. The second advantage is the ability to reorder the rules which introduces different permutations of the rules to the population.

The cross over probability used in the GA is 0.7, well within the typical range of 0.6 to 0.95 (Cordón, 2001). Although, this probability is shared equally between the two cross over operators. This means that if the cross over is selected, one of the two operators is chosen randomly with a 50%-50% chance.

6.4 Results

As the GA is a heuristic algorithm, it is not guaranteed to converge to the optimal value. Also, because of its stochastic property, often multiple runs of GA is used to increase the likelihood of convergence to the optimal or even a good solution. In this experiment, 10 separate runs of GA is used and the outcomes are reported. The final results of these 10 runs are shown in Table 6.1. For each test, the number of GA iterations that has happened until termination, the Training MPE, the Test MPE and the Overall MPE for the individual with best training MPE are reported.

For the sake of comparison, it is worth reminding that the fuzzy controller proposed in Chapter 5 had Overall MPE of 4.32 while the CEC benchmark policy's Overall MPE was 2.12.

Table 6.1 Results of the 10 runs of the GA and the results of the chromosome with the best Training MPE.

	Number of generations	Test MPE	Training MPE	Overall MPE
Run 1	195	2.11	2.10	2.10
Run 2	199	2.28	2.12	2.17
Run 3	199	2.17	2.10	2.12
Run 4	199	2.22	2.11	2.14
Run 5	130	2.29	2.14	2.18
Run 6	199	2.08	2.12	2.11
Run 7	199	2.16	2.07	2.10
Run 8	199	2.12	2.06	2.08
Run 9	199	2.02	2.04	2.03
Run 10	199	2.12	2.07	2.09

Also, the progress of Test, Training and Overall MPEs throughout iterations for all 10 runs are plotted in Figure 6.2.

One important observation from Table 6.1 and Figure 6.2 is that the test and training results are similar. This suggests that overfitting is not an issue and the outcomes are likely to be applicable to the underlying system.

Furthermore, from both Table 6.1 and Figure 6.2 it can be concluded that while the method is relatively stable and converges to the similar outcomes, the search space is large and, since the GA does not converge to exactly the same results, it is highly unlikely that the optimal value is achieved. However, the final outcomes of most runs are better than all the policies introduced in Chapter 5; GA achieved a fuzzy controller that even performs better than the CEC benchmark policy that is based on an exhaustive search of cost estimations.

Among the controllers in the final populations, the best overall MPE is achieved in Run 9 which has an Overall MPE of 2.01, a Training MPE of 2.04 and a Test MPE of 1.94. This particular controller shows a 53% reduction in the overall MPE, compared to the fuzzy controller proposed in Chapter 5. We applied a pair-wise t-test on the results of these two controllers using the dataset that is introduced in Chapter 5 to check if this improvement is statistically significant. Pairwise t-test utilises the fact that the error for each data point have been measured twice, once on each of the two controllers. Using these measurements, it determines the statistical significance of the difference between the two MPEs. The null hypothesis is that the change in the MPE is coincidental, with $\alpha = 0.05$. The result of the test showed that the null hypothesis is rejected with a probability of $1.96 * 10^{-26}$, hence the improvement is significant. At this level, the confidence interval of percentage of reduction in MPE is between 34.7% and 72.3%.

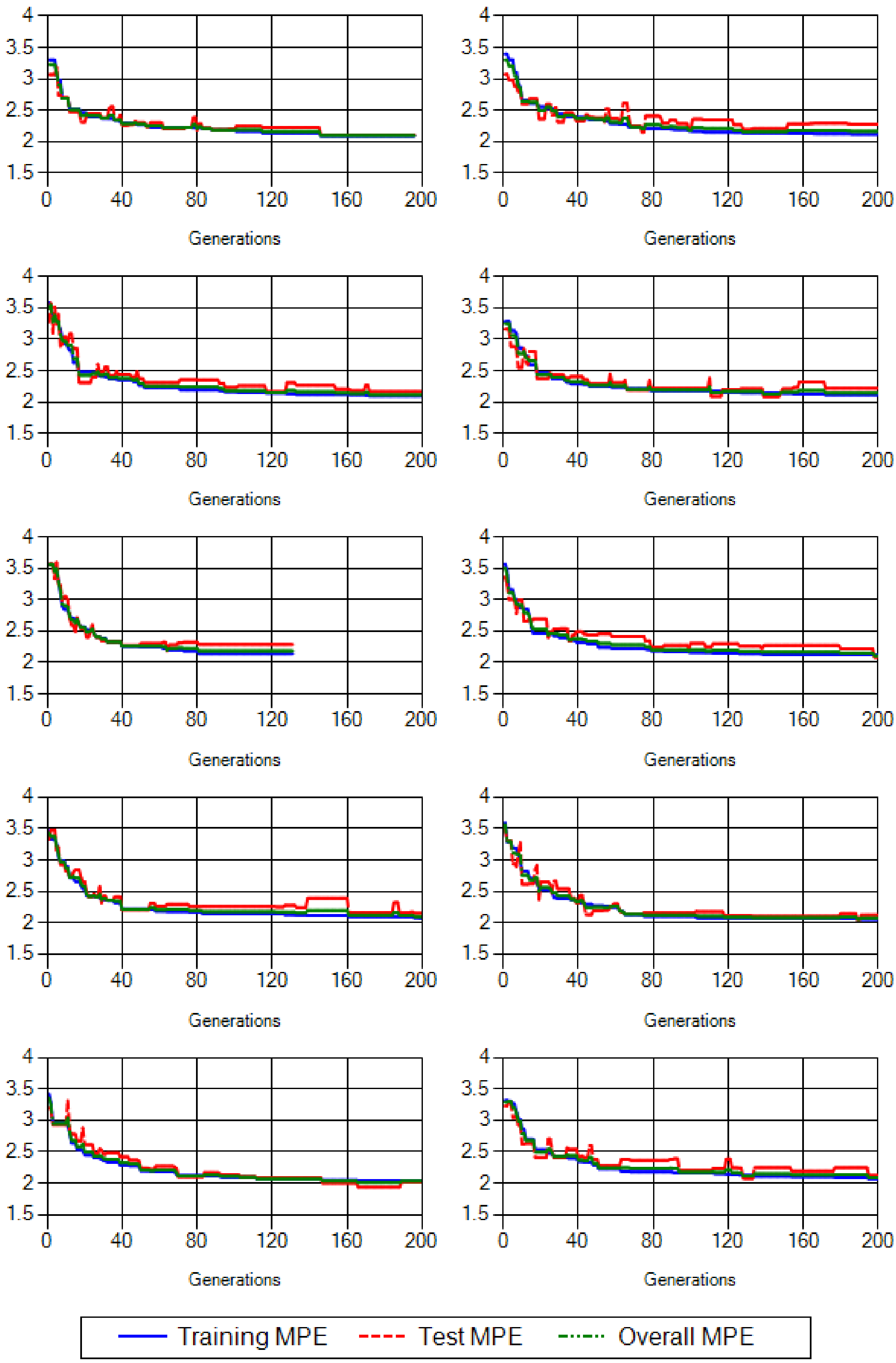

Figure 6.2 Progress of Test, Training and Overall MPEs in GA generations for 10 separate tests.

6.5 Discussion on the final rules

In this section, the details of the best final rules of the fuzzy controller are discussed. Specifically, by looking into the obtained rules, we can gain some insight into the complex relationships between the RL network parameters and the quality thresholds. The insight gathered from this experiment can be useful to experts.

As each run of the GA has a population size of 2000, there are 20000 fuzzy controllers just in the last generations of these runs. Therefore, obviously it is impossible to report the details of all of them. Instead, the controller with the best Overall MPE in the final populations has been chosen to be discussed. This controller is the result of Run 9 which has a Training MPE of 2.04, Test MPE of 1.94 and Overall MPE of 2.01.

The new fuzzy rules in the chosen controller are as follows. Please note that all the rules introduced in Chapter 5 are also in the controller and the new rules are working alongside the initial rules.

```
1) IF Remanufacturing-to-New-Unit-Cost-Ratio is High AND
Disassembly-to-Procurement-Setup-Ratio is Medium AND
Repair-to-New-Unit-Cost-Ratio is High THEN
Remanufacturing-Quality-Threshold is Increase AND
Repair-Quality-Threshold is Increase
```

When both remanufacturing to new and repair to new unit ratios are high, especially when the disassembly setup cost also does not provide any incentive, recovery activities are generally expensive. Therefore, it can be economically undesirable to recover and increasing both quality thresholds to eliminate recovery can be the preferred outcome. This possibility is not covered by the initial rules, instead one of the initial rules suggests an increase in the repair quality threshold, other considers a decrease in the repair quality threshold and an increase in remanufacturing quality threshold and one only propose to keep the remanufacturing quality threshold as is. Therefore, the overall result of initial rules is to keep the repair quality threshold as is while the remanufacturing quality threshold is slightly increased. This new rule can correct the controller's output in this situation.

```
2) IF Remanufacturing-to-New-Unit-Cost-Ratio is High AND
Return-to-Demand-Ratio is High THEN
Remanufacturing-Quality-Threshold is Increase AND
Repair-Quality-Threshold is Increase
```

In case of abundance of returned products, we observed a need to increase the quality thresholds to reduce unnecessary recovery activity. This is even more important in the case of high remanufacturing to new unit cost as it is stressed in this rule. Please note that it is also necessary to increase the repair quality threshold to counteract the rule for high remanufacturing-to-new-unit-cost ratio which suggests decreasing the repair quality threshold.

```
3) IF Return-to-Demand-Ratio is Low AND
Repair-to-New-Unit-Cost-Ratio is Low THEN
Remanufacturing-Quality-Threshold is AsIs
AND Repair-Quality-Threshold is Decrease
```

When the return to demand ratio is low, fewer products are available to recover and, as the setup costs can dominate the overall costs, it makes an economic sense to limit the recovery to one of the two options. In the situation where repair to new unit cost is low, it is clear that a repair only policy with low repair quality threshold is preferred. However, the rule defined in Chapter 5 for low return-to-demand ratio is just reducing the remanufacturing quality threshold, trying to keep the overall recovery volume. Hence, this rule will correct controller's behaviour when a repair only policy is preferred.

```
4) IF Remanufacturing-to-New-Unit-Cost-Ratio is Low AND
Disassembly-to-Procurement-Setup-Ratio is High THEN
Remanufacturing-Quality-Threshold is Decrease
```

Low remanufacturing-to-new-unit-cost ratio and high disassembly-to-procurement-setup-cost imply a low unit cost of disassembly and a high setup cost of disassembly. In this scenario, high quantities of remanufacturing can be desirable as increasing the volume of disassembly would decrease the effect of disassembly setup costs and, combined with low unit disassembly cost, it can lead to cost savings. However, in the case of initial rules, one rule suggests a decrease in remanufacturing quality threshold while the other propose the opposite. Hence, this rule can sway the output into the right direction by decreasing the remanufacturing quality threshold.

```
5) IF Remanufacturing-to-New-Unit-Cost-Ratio is Medium AND
Repair-to-Disassembly-Setup-Ratio is Medium AND
Return-to-Demand-Ratio is Low AND
Repair-to-New-Unit-Cost-Ratio is Low THEN
Remanufacturing-Quality-Threshold is Increase AND
Repair-Quality-Threshold is Decrease
```

Similar to rule 3 when the return-to-demand ratio is low but repair-to-new-unit-cost is also low, increasing repair volume, possibly leading to a repair only policy, is preferred. More specifically, when both disassembly setup cost and unit disassembly cost are not superior to respective repair costs, it becomes more important to use a repair only policy. This rule provides the desired output by increasing the remanufacturing quality threshold simultaneously with decreasing the repair quality threshold.

```
6) IF Remanufacturing-to-New-Unit-Cost-Ratio is Low AND
Disassembly-to-Procurement-Setup-Ratio is High AND
Return-to-Demand-Ratio is Low AND
```

`Repair-to-New-Unit-Cost-Ratio is Low THEN`

`Remanufacturing-Quality-Threshold is AsIs AND`

`Repair-Quality-Threshold is Decrease`

This rule covers the scenario when both repair and remanufacturing unit costs are low but the quantity of returned products is also low and disassembly setup cost is high. In contrast with rule 4, the repair route is preferred to the remanufacturing route and, as the rule suggests, the repair quality threshold is decreased while remanufacturing quality threshold should stay as is.

`7) IF Repair-to-Disassembly-Setup-Ratio is Low AND`

`Return-to-Demand-Ratio is High AND`

`Repair-to-New-Unit-Cost-Ratio is Low THEN`

`Remanufacturing-Quality-Threshold is Decrease AND`

`Repair-Quality-Threshold is Increase`

When both the repair unit cost and the setup cost are relatively low, there is a great tendency for reducing the repair quality threshold in the initial rules. However, in the case of high quantities of return, the quantity of repair should be reduced instead of being increased. This is because an excess of returned products are available and such a quantity of repaired products is simply unnecessary. This fact is being reflected in this rule. Also, as observed, when return to demand ratio is high and there is a high quantity of return, the recovery can be more selective; some products which are not suitable for repair can provide better cost savings by being remanufactured and hence, allowing some remanufacturing activity to be beneficial. This is why the rule decreased the remanufacturing quality threshold.

`8) IF Remanufacturing-to-New-Unit-Cost-Ratio is Medium AND`

`Repair-to-New-Unit-Cost-Ratio is High THEN`

`Remanufacturing-Quality-Threshold is Increase AND`

```
Repair-Quality-Threshold is Increase
```

When the repair unit cost is relatively high and the remanufacturing to new unit cost ratio is medium, one of the initial rules suggests increasing the repair quality threshold while the other suggests keeping both quality thresholds as is. In this rule, both recovery activities are suggested to be reduced by increasing both quality thresholds. In this way, the repair volume will be reduced while remanufacturing activity can be more selective and utilise better quality products only.

```
9) IF Remanufacturing-to-New-Unit-Cost-Ratio is Medium AND
Repair-to-Disassembly-Setup-Ratio is Low AND
Disassembly-to-Procurement-Setup-Ratio is Medium AND
Repair-to-New-Unit-Cost-Ratio is High THEN
Remanufacturing-Quality-Threshold is Increase AND
Repair-Quality-Threshold is Increase
```

This rule is similar to rule 8, and the general comments for that rule applies here as well. Beyond the preconditions of rule 8, in this rule repair to disassembly setup ratio is low and disassembly to procurement setup ratio is considered medium. In this situation, the other initial rules defined in Chapter 5 will reduce the repair quality threshold and keep the remanufacturing quality threshold as is. This rule, along with rule 8, counteracts them to reach the desirable outcome which is to increase both quality thresholds.

```
10) IF Repair-to-Disassembly-Setup-Ratio is High AND
Disassembly-to-Procurement-Setup-Ratio is Medium AND
Repair-to-New-Unit-Cost-Ratio is Low THEN
Repair-Quality-Threshold is Decrease
```

When the repair to disassembly setup ratio is high, the initial rule introduced for this case suggests an increase in repair quality threshold. However, when repair to new unit cost ratio is low, a higher quantity of repair is still desirable. This is particularly true when disassembly setup cost does not have a significant advantage to the forward production and hence, is also unlikely to be more cost effective than repair.

6.6 Conclusions

In this chapter, the goal was to improve the performance of the fuzzy controller introduced in Chapter 5 by finding more specific fuzzy rules using a GA. The dataset introduced in Chapter 5 was split between a Training dataset and a Test dataset, the former to be used in learning the rules and the latter in validating the results. A single objective GA was used which is based on Pittsburgh learning approach with a list of rule encoding. A simple mutation operator and two cross-over operators were utilised.

The results showed a significant improvement to the initial controller with more than 50% reduction in the MPE measure of the controller. The statistical significance of the improvement was proved using a pairwise t-test. Also, the modified controller with the best Overall MPE surpassed the performance of the CEC policy, reaching a 5.2% reduction.

Determined rules show that treating the input parameters in rules separately is not enough and in many situations, combinations of parameters can lead to improved outcomes. These additional rules can equip experts with better understanding about some of the interrelationships between the parameters that are the most influential on the RL network performance.

Chapter 7

Conclusions and Further Work

7.1 Summary and conclusions

In this research, quantitative decision making in RL networks in the presence of uncertainty and variable quality of returns has been examined. Demand and return quality and quantity are among the most important sources of RL networks' uncertainty. Also, quality of returns are identified as being considerably influential on RL decision making. Routing the returned products into alternative recovery routes, including repair and remanufacturing, and disposal are analysed, considering the quality of returns.

Fuzzy numbers provide an appropriate approach to model uncertainty in demand and return flows, which are utilised in this research. Quality thresholds are found to provide a straightforward but effective way to determine desirable quality levels for alternative recovery routes. Additionally, fuzzy control and genetic fuzzy systems are utilised in order to find satisfactory quality thresholds.

A literature review of RL concepts and quantitative models with a special attention to uncertainty and quality of returns has been provided in Chapter 2. Literature concerning quality of returns in RL networks is comprehensively discussed and summarised in tabular form, identifying type of model of quality of returns, model features, considerations of demand, type of uncertainty, type of recovery routes and methods applied. It has been concluded that the quality dependent routing of products is still a new area which requires more research. Also, fuzzy logic applications in RL networks are still very sparse and, due to the natural advantage of fuzzy logic in uncertainty modelling with limited statistical data, such research can be beneficial.

An overview of relevant methods and techniques that are used throughout this research has been presented in Chapter 3. Concepts from fuzzy logic, optimisation and genetic algorithm are discussed and fuzzy arithmetic, fuzzy control, fuzzy optimisation and genetic fuzzy methods are introduced. As part of this chapter, a fuzzy optimisation model with fuzzy constraints and a fuzzy objective function and the relevant conversion procedure into the equivalent crisp MIP model is proposed.

The main contributions of this research are presented in Chapters 4 to 6. In Chapter 4, a novel two phase optimisation model is proposed which facilitates decision making in integrated RL networks with repair, remanufacturing and disposal options. In this model, repair and remanufacturing quality thresholds are given as recovery policies' parameters and used in Phase 1 to determine fuzzy quantities of products sent to repair, remanufacturing and disposal routes as well as the average costs of repair and disassembly with respect to different returned products qualities. They are used in Phase 2 in the proposed fuzzy optimisation model to determine the decisions such as the quantity to be repaired, disassembled, procured and produced in each time period within the time horizon under consideration. Experiments with different recovery policies have been carried out and their effect on performance measures such as satisfaction degree (α), recovery and production costs and share of lost sale, repair, remanufacturing and forward routes in supply have been analysed. To understand the relationship between the performance of recovery policies and other main parameters of the RL network, sensitivity analyses have been carried out considering the main network parameters such as quantity of returned products, repair and disassembly unit costs and setup costs.

Based on the results of the sensitivity analyses, it is concluded that performance of recovery policies depends on the following RL network parameters, including the return quantity, repair and disassembly unit costs and the production, repair and disassembly setup costs. Hence, recovery policies with fixed quality thresholds are not desirable and the RL network's recovery policy needs to be adjusted depending on identified parameters of the RL network. Regarding the quantity of returned products, it has been concluded that mixed policies, which include both repair and remanufacturing routes, are prevalent for the medium values of returns, for both very low and very high returns, repair only policies are preferred. In the case of the low return quantities, this is mostly to reduce the number of setups while for the high return quantities, where an abundance of returned products are available, repair only policies are preferred as they are use the highest quality of returns and incur the lowest cost. For unit repair and disassembly costs, as expected, it is observed that the preferred repair quality threshold increases (decreases) with an increase (decrease) in the repair unit costs while the remanufacturing quality threshold has a similar relationship with the unit remanufacturing costs. However, the preferred repair quality threshold also increases (decreases) with a decrease (increase) in the remanufacturing unit cost. For setup costs, it is concluded that high values of repair setup cost lead to remanufacturing only policies, while high values of either disassembly or production setup costs lead to repair only policies. The best recovery policy is relatively unaffected by changes in the procurement setup costs.

A fuzzy controller is proposed in Chapter 5 to determine quality thresholds based on the following network parameter ratios: repair to new cost ratio, remanufacturing to new cost ratio, repair to disassembly setup ratio, disassembly to procurement setup ratio and return quantity to demand ratio. Based on the generated scatter plots and the sensitivity analyses carried out in Chapter 4, fuzzy rules and membership functions' definitions are suggested. Additionally, two benchmark policies, one based on fixed thresholds and another based on the comparison of cost estimates, are proposed to give a baseline for comparing the performance of the fuzzy controller. Comparison has shown that the proposed controller performs better than all fixed threshold policies while the cost estimate comparison policy outperforms the controller. Sensitivity analyses were carried out considering the main network parameters, including average repair and disassembly costs, repair, disassembly, production and procurement setup costs and return quantity to demand ratio, to better understand the advantages and disadvantages of the fuzzy controller and the benchmark policies.

From the results of sensitivity analyses on the fuzzy controller, a multitude of conclusions have been made. First and foremost, fixed threshold policies' performance is often very sensitive to all identified parameters. Hence, a certain fixed threshold policy that is desirable for a particular value of a parameter, is possibly not desirable for another value. Both the fuzzy controller and the cost estimate comparison policy suffer, to a considerably lesser extent, from this issue. However, for both average repair and disassembly costs, fuzzy controller showed a slightly worst performance on both ends of the parameters' ranges compared to the medium values. While the performance of the fuzzy controller remained relatively constant for all setup costs, increase in return to demand ratio is shown to have an adverse effect on both the fuzzy controller and the cost estimate comparison policy.

A genetic fuzzy method was utilised to improve the fuzzy controller that is intuitively determined. By applying the genetic fuzzy methods proposed, new rules were introduced to improve the fuzzy controller's performance. The manually proposed rules in Chpater 5 were straightforward with a single term in the rule's precedent. However, newly introduced rules were of a more complicated nature with multiple terms in the rule's precedent which allows consideration of the relationships between the RL network parameters. The results showed that the modified fuzzy controller performs significantly better than the original controller as well as both benchmark policies.

The result obtained by applying the genetic fuzzy model showed that considering the ratios separately in the fuzzy controller is not sufficient and the effects of their interdependencies need to be considered also. Particularly, the combined effect of unit and setup costs can lead to different desirable outcomes than those suggested by individual analysis. For example, one rule that has been suggested by the method refers to the situation when the disassembly unit cost is low but the disassembly setup cost is high. Analysing this scenario on an individual basis leads to conflicting results with one suggesting an increase in remanufacturing quality and the other suggesting a decrease. In this scenario, it is beneficial to increase disassembly activities to utilise the cost savings on unit disassembly cost while reducing the effect of disassembly setup cost by having larger setups.

The models described in this research are all implemented in software developed in C# language and Microsoft .NET framework. Gurobi MIP Solver is utilised for determining the solutions for the fuzzy RL optimisation problem which is converted into a crisp MIP problem. The software allows the RL network parameters to be entered as an Excel file and outputs the results also in the form of Excel files.

7.2 Suggestions for future work

Among possible extensions of the developed optimisation model, incorporating multi products and multi components are of relative importance. The presence of multi components can lead to a complex decision making problem with respect to corresponding quality levels. Also, the proposed model assumes constant quality thresholds for the time horizon. Allowing dynamically changing quality thresholds can prove to be difficult, however, it allows for more accurate models of the RL networks and better optimisation of the networks' performance.

This research was limited to a particular structure of RL networks with repair, remanufacturing and disposal. However, this can be extended into a generic model that can accommodate multiple options for each recovery route, disposal and perhaps multiple forward chains. Accompanied by other features such as presence of uncertainty, multi products, multi components and multi quality levels, such a model is likely to be able to realistically represent a great majority of RL networks.

There are also several avenues for technical improvements. For example, genetic fuzzy method applied can be improved. Iterative rule learning methods are promising for improving the convergence time of the algorithm, although they need to be adapted for this application. Also, tuning of membership functions and the existing rules are another possible extension. In the fuzzy controller, extra input parameters can be included, for example to allow for comparison of lead times. The fuzzy optimisation method assumes a crisp average cost of repair and remanufacturing, with respect to different qualities, which can be extended by allowing fuzzy cost values.

Multi objective analysis of RL networks is another area that has been sparsely researched so far. Inclusion of other objectives such as environmental effects and customer service levels will each have their own challenges and merits. Analysis of environmental effects are especially interesting as this can provide a better and more realistic understanding of RL networks environmental advantages while considering quality of returns and uncertainty.

Heuristic methods to improve the performance of RL networks is another area to be investigated. The fuzzy optimisation model proposed in Chapter 4 is highly complex and time-consuming to solve which limits its applications. Simpler heuristic models can be both useful for practical use and also for further academic research. Hence, a study of heuristic methods in this area is very desirable. Especially, it would be interesting to carry out a comparison between heuristic methods with the proposed optimisation method.

This research, although inspired by the structure and complexity of real world logistics networks, is of a theoretical nature. It is necessary to validate the results by applying the proposed models in practice. The study of these results in practice can provide both a better understanding of the RL behaviours and its performance for the practitioners and also help in correcting and improving the proposed theoretical model.

References

Introduction to lp_solve 5.5.2.0, 2010.

A. Alshamrani, K. Mathur, and R. H. Ballou. Reverse logistics: simultaneous design of delivery routes and returns strategies. *Computers & Operations Research*, 34 (2):595–619, Feb. 2007. ISSN 03050548. doi: 10.1016/j.cor.2005.03.015.

N. Aras and D. Aksen. Locating collection centers for distance- and incentive-dependent returns. *International Journal of Production Economics*, 111(2):316–333, 2008.

N. Aras, T. Boyaci, and V. Verter. The effect of categorizing returned products in remanufacturing. *IIE Transactions*, 36(4):319–331, Apr. 2004. ISSN 0740-817X. doi: 10.1080/07408170490279561.

A. Awasthi, S. S. Chauhan, and S. Goyal. A fuzzy multicriteria approach for evaluating environmental performance of suppliers. *International Journal of Production Economics*, 126(2):370–378, Aug. 2010. ISSN 09255273. doi: 10.1016/j.ijpe.2010.04.029.

G. Baltacioğlu, z. Temiz, and S. Erol. Fuzzy Wagner Whitin Algorithm and an Application of Class I Supplies. *Gazi University Journal of . . .*, 24(I):125–134, 2011.

A. Baykasoglu and T. Göçken. A review and classification of fuzzy mathematical programs. *Journal of Intelligent and Fuzzy Systems*, 19:205–229, 2008.

M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali. *Linear programming and network flows*. Wiley. com, 2011.

H. Behret and A. Korugan. Performance analysis of a hybrid system under quality impact of returns. *Computers & Industrial Engineering*, 56(2):507–520, Mar. 2009. ISSN 03608352. doi: 10.1016/j.cie.2007.11.001.

W. Bei and S. Linyan. A review of reverse logistics. *Applied Sciences (APPS)*, 7: 16–29, 2005.

H. Berenji and P. Khedkar. Learning and tuning fuzzy logic controllers through reinforcements. *Neural Networks, IEEE Transactions . . .*, 1992.

M. Biehl, E. Prater, and M. J. Realff. Assessing performance and uncertainty in developing carpet reverse logistics systems. *Computers & Operations Research*, 34(2):443–463, Feb. 2007. ISSN 03050548. doi: 10.1016/j.cor.2005.03.008.

M. D. Brito and R. Dekker. A framework for reverse logistics. In *Reverse Logistics Quantitative Models for Closed-Loop Supply Chains*. Springer Berlin Heidelberg, 2004.

J. M. Cadenas and J. L. Verdegay. A primer on fuzzy optimization models and methods. *Iranian Journal of Fuzzy Systems*, 3(1):1–21, 2006.

S. R. Cardoso, A. P. F. Barbosa-Póvoa, and S. Relvas. Design and Planning of Supply Chains with Integration of Reverse Logistics Activities under Demand Uncertainty. *European Journal of Operational Research*, Dec. 2012. ISSN 03772217. doi: 10.1016/j.ejor.2012.11.035.

J. Casillas. *Interpretability issues in fuzzy modeling*, volume 128. Springer, 2003. ISBN 354002932X.

O. Cordón. *Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases*, volume 19. World Scientific, 2001.

D. Das and P. Dutta. A system dynamics framework for integrated reverse supply chain with three way recovery and product exchange policy. *Computers & Industrial Engineering*, 66(4):720–733, Dec. 2013. ISSN 03608352. doi: 10.1016/j.cie.2013.09.016.

K. Das and A. H. Chowdhury. Designing a reverse logistics network for optimal collection, recovery and quality-based product-mix planning. *International Journal of Production Economics*, 135(1):209–221, Jan. 2012. ISSN 09255273. doi: 10.1016/j.ijpe.2011.07.010.

M. P. de Brito and E. a. van der Laan. Inventory control with product returns: The impact of imperfect information. *European Journal of Operational Research*, 194 (1):85–101, Apr. 2009. ISSN 03772217. doi: 10.1016/j.ejor.2007.11.063.

M. V. de la Fuente, L. Ros, and M. Cardós. Integrating Forward and Reverse Supply Chains: Application to a metal-mechanic company. *International Journal of Production Economics*, 111(2):782–792, Feb. 2008. ISSN 09255273. doi: 10.1016/j.ijpe.2007.03.019.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on*, 6 (2):182–197, 2002.

R. Dekker and M. Fleischmann. Quantitative models for reverse logistics decision making. *Reverse Logistics*, 2004.

M. Detyniecki and R. Yager. Ranking fuzzy numbers using α-weighted valuations. *International Journal of Uncertainty, . . .* , 8(5), 2000.

I. Dobos and K. Richter. An extended production/recycling model with stationary demand and return rates. *International Journal of Production Economics*, 90(3): 311–323, 2004.

I. Dobos and K. Richter. A production/recycling model with quality consideration. *International Journal of Production Economics*, 104(2):571–579, 2006.

S. Dowlatshahi. Developing a theory of reverse logistics. *Interfaces*, 30(June):143–155, 2000.

F. Du and G. W. Evans. A bi-objective reverse logistics network analysis for post-sale service. *Computers & Operations Research*, 35(8):2617–2634, Aug. 2008. ISSN 03050548. doi: 10.1016/j.cor.2006.12.020.

G. Easwaran and H. Üster. A closed-loop supply chain network design problem with integrated forward and reverse channel decisions. *IIE Transactions*, 42(11):779–792, Aug. 2010. ISSN 0740-817X. doi: 10.1080/0740817X.2010.504689.

T. Efendigil, S. Önüt, and E. Kongar. A holistic approach for selecting a third-party reverse logistics provider in the presence of vagueness. *Computers & Industrial Engineering*, 54(2):269–287, Mar. 2008. ISSN 03608352. doi: 10.1016/j.cie.2007.07.009.

A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing (Natural Computing Series)*. Springer, Oct. 2008. ISBN 3540401849.

A. M. A. El Saadany and M. Y. Jaber. A production/remanufacturing inventory model with price and quality dependant return rate. *Computers and Industrial Engineering*, 58(3):352–362, 2010.

G. Ferrer. On the widget remanufacturing operation. *European Journal of Operational Research*, 135(2):373–393, Dec. 2001. ISSN 03772217. doi: 10.1016/S0377-2217(00)00318-0.

G. Ferrer. Yield information and supplier responsiveness in remanufacturing operations. *European Journal of Operational Research*, 149(3):540–556, Sept. 2003. ISSN 03772217. doi: 10.1016/S0377-2217(02)00454-X.

G. Ferrer and M. E. Ketzenberg. Value of information in remanufacturing complex products. *IIE Transactions*, 36(3):265–277, Mar. 2004. ISSN 0740-817X. doi: 10.1080/07408170490274223.

M. Fleischmann, J. M. Bloemhof-Ruwaard, R. Dekker, E. van der Laan, J. a.E.E. van Nunen, and L. N. Van Wassenhove. Quantitative models for reverse logistics: A review. *European Journal of Operational Research*, 103(1):1–17, Nov. 1997. ISSN 03772217. doi: 10.1016/S0377-2217(97)00230-0.

J. Fodor and B. Bede. Arithmetics with fuzzy numbers: a comparative overview. *Proceeding of 4th Slovakian-Hungarian Joint . . .* , 2006.

M. R. Galbreth and J. D. Blackburn. Optimal acquisition and sorting policies for remanufacturing. *Production and Operations Management*, 15(3):384–392, 2006.

a. Gharbi, R. Pellerin, and J. Sadr. Production rate control for stochastic remanufacturing systems. *International Journal of Production Economics*, 112(1):37–47, Mar. 2008. ISSN 09255273. doi: 10.1016/j.ijpe.2006.08.015.

V. Guide and L. Wassenhove. Managing product returns for remanufacturing. *Production and Operations . . .* , 2001.

V. Guide, R. H. Teunter, and L. N. Van Wassenhove. Matching demand and supply to maximize profits from remanufacturing. *. . . & Service Operations . . .* , 5(4):303–316, 2003.

V. D. R. Guide, L. Muyldermans, and L. N. Van Wassenhove. Hewlett-Packard Company Unlocks the Value Potential from Time-Sensitive Returns. *Interfaces*, 35(4): 281–293, July 2005. ISSN 0092-2102. doi: 10.1287/inte.1050.0145.

S. Guo, G. Aydin, and G. C. Souza. Dismantle or remanufacture? *European Journal of Operational Research*, 233(3):580–583, Mar. 2014. ISSN 03772217. doi: 10.1016/j.ejor.2013.09.042.

Gurobi Optimization. Gurobi Optimizer 5.6 Overview, 2014.

S. Guua and Y. Wu. Two-phase approach for solving the fuzzy linear programming problems. *Fuzzy Sets and Systems*, 107:191–195, 1999.

S. K. Halgamuge and M. Glesner. Neural networks in designing fuzzy systems for real world applications. *Fuzzy sets and systems*, 65(1):1–12, 1994.

F. Harris. How many parts to make at once. *Operations Research*, 38(2):135–136, 1990.

F. Herrera. Genetic fuzzy systems: taxonomy, current research trends and prospects. *Evolutionary Intelligence*, 1(1):27–46, Jan. 2008. ISSN 1864-5909. doi: 10.1007/s12065-007-0001-5.

F. Herrera and J. L. Verdegay. Three models of fuzzy integer linear programming. *European Journal of Operational Research*, 83(3):581–593, 1995.

F. Herrera, M. Lozano, and J. Verdegay. Tuning fuzzy logic controllers by genetic algorithms. *International Journal of Approximate Reasoning*, 12(3):299–315, 1995.

IBM. IBM CPLEX Optimizer.

M. A. Ilgin and S. M. Gupta. Environmentally conscious manufacturing and product recovery (ECMPRO): A review of the state of the art. *Journal of environmental management*, 91(3):563–91, 2010. ISSN 1095-8630. doi: 10.1016/j.jenvman.2009.09.037.

K. Inderfurth. Optimal policies in hybrid manufacturing/remanufacturing systems with product substitution. *International Journal of Production Economics*, 90(3): 325–343, Aug. 2004. ISSN 09255273. doi: 10.1016/S0925-5273(02)00470-X.

K. Inderfurth. Impact of uncertainties on recovery behavior in a remanufacturing environment. A numerical analysis. *International Journal of Physical Distribution and Logistics Management*, 35(5):318–336, 2005.

K. Inderfurth and E. V. D. Laan. Leadtime effects and policy improvement for stochastic inventory control with remanufacturing. *International Journal of Production ...*, 71:381–390, 2001.

K. Inderfurth and I. M. Langella. Heuristics for solving disassemble-to-order problems with stochastic yields. *OR Spectrum*, 28(1):73–99, Oct. 2005. ISSN 0171-6468. doi: 10.1007/s00291-005-0007-2.

K. Inderfurth, A. de Kok, and S. Flapper. Product recovery in stochastic remanufacturing systems with multiple reuse options. *European Journal of Operational Research*, 133(1):130–152, Aug. 2001. ISSN 03772217. doi: 10.1016/S0377-2217(00)00188-0.

M. Inuiguchi and J. Ramik. Possibilistic linear programming: A brief review of fuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem. *Fuzzy Sets and Systems*, 111(1):3–28, 2000.

M. Y. Jaber and A. M. El Saadany. The production, remanufacture and waste disposal model with lost sales. *International Journal of Production Economics*, 120(1):115–124, July 2009. ISSN 09255273. doi: 10.1016/j.ijpe.2008.07.016.

J. Jang and C. Sun. Functional equivalence between radial basis function networks and fuzzy inference systems. *Neural Networks, IEEE Transactions on*, 4, Jan. 1993. ISSN 1045-9227. doi: 10.1109/72.728403.

V. Jayaraman. Production planning for closed-loop supply chains with product recovery and reuse: An analytical approach. *International Journal of Production Research*, 44(5):981–998, 2006.

X. Jin, S. J. Hu, J. Ni, and G. Xiao. Assembly Strategies for Remanufacturing Systems With Variable Quality Returns. *IEEE Transactions on Automation Science and Engineering*, 10(1):76–85, Jan. 2013. ISSN 1545-5955. doi: 10.1109/TASE.2012.2217741.

S. Jorjani, J. Leu, and C. Scott. Model for the allocation of electronics components to reuse options. *International Journal of Production Research*, 42(6):1131–1145, Mar. 2004. ISSN 0020-7543. doi: 10.1080/00207540310001632466.

S. Kara, F. Rugrungruang, and H. Kaebernick. Simulation modelling of reverse logistics networks. *International Journal of Production Economics*, 106(1):61–69, Mar. 2007. ISSN 09255273. doi: 10.1016/j.ijpe.2006.04.009.

S. S. Kara and S. Onut. A two-stage stochastic and robust programming approach to strategic planning of a reverse supply network: The case of paper recycling. *Expert Systems with Applications*, 37(9):6129–6137, Sept. 2010. ISSN 09574174. doi: 10.1016/j.eswa.2010.02.116.

E. Karsak and O. Kuzgunkaya. A fuzzy multiple objective programming approach for the selection of a flexible manufacturing system. *International journal of production economics*, 79:101–111, 2002.

M. E. Ketzenberg, E. Laan, and R. H. Teunter. Value of Information in Closed Loop Supply Chains. *Production and Operations Management*, 15(3):393–406, Jan. 2009. ISSN 10591478. doi: 10.1111/j.1937-5956.2006.tb00253.x.

K. Kim, I. Song, J. Kim, and B. Jeong. Supply planning model for remanufacturing system in reverse logistics environment. *Computers & Industrial Engineering*, 51 (2):279–287, Oct. 2006. ISSN 03608352. doi: 10.1016/j.cie.2006.02.008.

M. Ko, A. Tiwari, and J. Mehnen. A review of soft computing applications in supply chain management. *Applied Soft Computing*, 10(3):661–674, June 2010. ISSN 15684946. doi: 10.1016/j.asoc.2009.09.004.

I. Konstantaras and K. Skouri. Lot sizing for a single product recovery system with variable setup numbers. *European Journal of Operational Research*, 203(2):326–335, June 2010. ISSN 03772217. doi: 10.1016/j.ejor.2009.07.018.

I. Konstantaras, K. Skouri, and M. Jaber. Lot sizing for a recoverable product with inspection and sorting. *Computers & Industrial Engineering*, 58(3):452–462, Apr. 2010. ISSN 03608352. doi: 10.1016/j.cie.2009.11.004.

A. Korugan, K. D. Dingeç, T. Önen, and N. Y. Ateş. On the quality variation impact of returns in remanufacturing. *Computers & Industrial Engineering*, 64(4):929–936, Apr. 2013. ISSN 03608352. doi: 10.1016/j.cie.2013.01.003.

H. R. Krikke, A. Van Harten, and P. C. Schuur. On a medium term product recovery and disposal strategy for durable assembly products. *International Journal of Production Research*, 36(1):111–139, 1998a.

H. R. Krikke, A. Van Harten, and P. C. Schuur. On a medium term product recovery and disposal strategy for durable assembly products. *International Journal of Production Research*, 36(1):111–139, 1998b.

E. V. D. Laan and M. Salomon. Production planning and inventory control with remanufacturing and disposal. *European Journal of Operational Research*, 00(97), 1997.

B. Lebreton and A. Tuma. A quantitative approach to assessing the profitability of car and truck tire remanufacturing. *International Journal of production economics*, 104(2):639–652, Dec. 2006. ISSN 09255273. doi: 10.1016/j.ijpe.2004.11.010.

B. K. Lee, K. H. Kang, and Y. H. Lee. Decomposition heuristic to minimize total cost in a multi-level supply chain network. *Computers & Industrial Engineering*, 54(4):945–959, May 2008. ISSN 03608352. doi: 10.1016/j.cie.2007.11.005.

H. Lee, T. Zhang, M. Boile, S. Theofanis, and S. Choo. Designing an integrated logistics network in a supply chain system. *KSCE Journal of Civil Engineering*, 17 (4):806–814, May 2013. ISSN 1226-7988. doi: 10.1007/s12205-013-0087-5.

K. Lieckens and N. Vandaele. Reverse logistics network design with stochastic lead times. *Computers & Operations Research*, 34(2):395–416, Feb. 2007. ISSN 03050548. doi: 10.1016/j.cor.2005.03.006.

K. Lieckens and N. Vandaele. Multi-level reverse logistics network design under uncertainty. *International Journal of Production ...*, 2012.

S.-T. Liu and C. Kao. Solving fuzzy transportation problems based on extension principle. *European Journal of Operational Research*, 153(3):661–674, Mar. 2004. ISSN 03772217. doi: 10.1016/S0377-2217(02)00731-2.

Z. Lu and N. Bostel. A facility location model for logistics systems including reverse flows: The case of remanufacturing activities. *Computers & Operations Research*, 34(2):299–323, Feb. 2007. ISSN 03050548. doi: 10.1016/j.cor.2005.03.002.

B. Mahadevan, D. F. Pyke, and M. Fleischmann. Periodic review, push inventory policies for remanufacturing. *European Journal of Operational Research*, 151(3): 536–551, Dec. 2003. ISSN 03772217. doi: 10.1016/S0377-2217(02)00627-6.

S. Mahapatra, R. Pal, and R. Narasimhan. Hybrid (re)manufacturing: manufacturing and operational implications. *International Journal of Production Research*, 50(14):3786–3808, July 2012. ISSN 0020-7543. doi: 10.1080/00207543.2011.588615.

S. E. Marshall. *Refuse or Reuse: Managing the Quality of Returns in Product Recovery Systems*. Phd thesis, University of Edinburgh, 2012.

S. Miller and R. John. An Interval Type-2 Fuzzy multiple echelon supply chain model. *Knowledge-Based Systems*, 23(4):363–368, May 2010. ISSN 09507051. doi: 10.1016/j.knosys.2009.11.016.

S. Mitra. Revenue management for remanufactured products. *Omega*, 35(5):553–562, 2007.

S. Mitra. Inventory management in a two-echelon closed-loop supply chain with correlated demands and returns. *Computers & Industrial Engineering*, 62(4):870–879, May 2012. ISSN 03608352. doi: 10.1016/j.cie.2011.12.008.

S. K. Mukhopadhyay and H. Ma. Joint procurement and production decisions in remanufacturing under quality and demand uncertainty. *International Journal of Production Economics*, 120(1):5–17, July 2009. ISSN 09255273. doi: 10.1016/j.ijpe.2008.07.032.

K. Nakashima and S. Gupta. Analysis of remanufacturing policy with consideration for returned products quality. *Mechanical and Industrial Engineering Faculty Publications*, 2010.

G. Nemhauser and L. Wolsey. Integer and combinatorial optimization. *Wiley-Interscience*, 1999.

G. Nenes and Y. Nikolaidis. A multi-period model for managing used product returns. *International Journal of Production Research*, 50(5):1360–1376, Mar. 2012. ISSN 0020-7543. doi: 10.1080/00207543.2011.609650.

G. Nenes, S. Panagiotidou, and R. Dekker. Inventory control policies for inspection and remanufacturing of returns: A case study. *International Journal of Production Economics*, 125(2):300–312, 2010.

V. Özkır and H. Başlıgıl. Modelling product-recovery processes in closed-loop supply-chain network design. *International Journal of Production Research*, 50 (8):2218–2233, 2012.

W. Pedrycz and F. Gomide. *An introduction to fuzzy sets: analysis and design*. the MIT Press, 1998.

D. Peidro, J. Mula, R. Poler, and J.-L. Verdegay. Fuzzy optimization for supply chain planning under supply, demand and process uncertainties. *Fuzzy Sets and Systems*, 160(18):2640–2657, 2009.

D. Petrovic, Y. Xie, K. Burnham, and R. Petrovic. Coordinated control of distribution supply chains in the presence of fuzzy customer demand. *European Journal of Operational Research*, 185(1):146–158, 2008.

D. T. Pham and D. Karaboga. Intelligent Optimisation Techniques: Genetic Algorithms. *Tabu Search, Simulated Annealing and Neural Networks*, 2000.

M. S. Pishvaee, F. Jolai, and J. Razmi. A stochastic optimization model for integrated forward/reverse logistics network design. *Journal of Manufacturing Systems*, 28 (4):107–114, Dec. 2009. ISSN 02786125. doi: 10.1016/j.jmsy.2010.05.001.

M. S. Pishvaee, R. Z. Farahani, and W. Dullaert. A memetic algorithm for bi-objective integrated forward/reverse logistics network design. *Computers & Operations Research*, 37(6):1100–1112, June 2010. ISSN 03050548. doi: 10.1016/j.cor.2009.09.018.

S. Pokharel and Y. Liang. A model to evaluate acquisition price and quantity of used products for remanufacturing. *International Journal of Production Economics*, 138 (1):170–176, July 2012. ISSN 09255273. doi: 10.1016/j.ijpe.2012.03.019.

Z. Qin and X. Ji. Logistics network design for product recovery in fuzzy environment. *European Journal of Operational Research*, 202(2):479–490, Apr. 2010. ISSN 03772217. doi: 10.1016/j.ejor.2009.05.036.

M. Radhi. *Impact of Quality Grading and Uncertainty on Recovery Behaviour in a Remanufacturing Environment Impact of Quality Grading and Uncertainty on Recovery Behaviour in a Remanufacturing Environment*. PhD thesis, University of Windsor, 2012.

S. Rubio and A. Corominas. Optimal manufacturing–remanufacturing policies in a lean production environment. *Computers & Industrial Engineering*, 55(1):234–242, Aug. 2008. ISSN 03608352. doi: 10.1016/j.cie.2007.12.009.

M. I. G. Salema, A. P. Barbosa-Povoa, and A. Q. Novais. An optimization model for the design of a capacitated multi-product reverse logistics network with uncertainty. *European Journal of Operational Research*, 179(3):1063–1077, June 2007. ISSN 03772217. doi: 10.1016/j.ejor.2005.05.032.

D. A. Schrady. A deterministic inventory model for reparable items. *Naval Research Logistics Quarterly*, 14(3):391–398, 1967. ISSN 00281441. doi: 10.1002/nav.3800140310.

J. Shi, G. Zhang, and J. Sha. Optimal production planning for a multi-product closed loop system with uncertain demand and return. *Computers & Operations Research*, 38(3):641–650, Mar. 2011. ISSN 03050548. doi: 10.1016/j.cor.2010.08.008.

G. Souza, M. E. Ketzenberg, and V. D. R. Guide Jr. Capacitated remanufacturing with service level constraints*. *Production and ...*, 11(September 2000), 2002.

S. Srivastava. Network design for reverse logistics. *Omega*, 36(4):535–548, Aug. 2008. ISSN 03050483. doi: 10.1016/j.omega.2006.11.012.

E. Stanley Lee and R. Li. Fuzzy multiple objective programming and compromise programming with Pareto optimum. *Fuzzy Sets and Systems*, 53(3):275–288, Feb. 1993. ISSN 01650114. doi: 10.1016/0165-0114(93)90399-3.

G. Tagaras and C. Zikopoulos. Optimal location and value of timely sorting of used items in a remanufacturing supply chain with multiple collection sites. *International Journal of Production Economics*, 115(2):424–432, Oct. 2008. ISSN 09255273. doi: 10.1016/j.ijpe.2007.10.026.

T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE ...*, 1985.

J. Tang and D. Wang. A nonsymmetric model for fuzzy nonlinear programming problems with penalty coefficients. *Computers & operations research*, 24(8):717–725, 1997.

O. Tang and M. Naim. The impact of information transparency on the dynamic behaviour of a hybrid manufacturing/remanufacturing system. *International Journal of Production Research*, 42(19):4135–4152, 2004.

R. Teunter. Lot-sizing for inventory systems with product recovery. *Computers & Industrial Engineering*, 46(3):431–441, June 2004. ISSN 03608352. doi: 10.1016/j.cie.2004.01.006.

R. Teunter. Determining optimal disassembly and recovery strategies. *Omega*, 34(6): 533–537, 2006.

R. H. Teunter. Economic ordering quantities for recoverable item inventory systems. *Naval Research Logistics*, 48(6):484–495, Sept. 2001. ISSN 0894-069X. doi: 10.1002/nav.1030.

R. H. Teunter, Z. P. Bayindir, and W. Van Den Heuvel. Dynamic lot sizing with product returns and remanufacturing. *International Journal of Production Research*, 44 (20):4377–4400, 2006.

M. Thierry and M. Salomon. Strategic issues in product recovery management. *California management . . .* , 37(2), 1995.

S. Torabi and E. Hassini. An interactive possibilistic programming approach for multiple objective supply chain master planning. *Fuzzy Sets and Systems*, 159(2): 193–214, Jan. 2008. ISSN 01650114. doi: 10.1016/j.fss.2007.08.010.

B. Vahdani, R. Tavakkoli-Moghaddam, M. Modarres, and A. Baboli. Reliable design of a forward/reverse logistics network under uncertainty: A robust-M/M/c queuing model. *Transportation Research Part E: Logistics and Transportation Review*, 48 (6):1152–1168, 2012.

E. van der Laan, M. Salomon, R. Dekker, and L. Van Wassenhove. Inventory Control in Hybrid Systems with Remanufacturing. *Management Science*, 45(5):733–747, May 1999. ISSN 0025-1909. doi: 10.1287/mnsc.45.5.733.

S. Wadhwa, J. Madaan, and F. T. S. Chan. Flexible decision modeling of reverse logistics system: A value adding MCDM approach for alternative selection. *Robotics and Computer-Integrated Manufacturing*, 25(2):460–469, 2009.

R.-C. Wang and H.-H. Fang. Aggregate production planning with multiple objectives in a fuzzy environment. *European Journal of Operational Research*, 133(3):521–536, Sept. 2001. ISSN 03772217. doi: 10.1016/S0377-2217(00)00196-X.

R.-C. Wang and T.-F. Liang. Application of fuzzy multi-objective linear programming to aggregate production planning. *Computers & Industrial Engineering*, 46(1):17–41, Mar. 2004. ISSN 03608352. doi: 10.1016/j.cie.2003.09.009.

C. Wei, Y. Li, and X. Cai. Robust optimal policies of production and inventory with uncertain returns and demand. *International Journal of Production Economics*, 134(2):357–367, Dec. 2011. ISSN 09255273. doi: 10.1016/j.ijpe.2009.11.008.

A. Xanthopoulos and E. Iakovou. On the optimal design of the disassembly and recovery processes. *Waste Management*, 29(5):1702–1711, 2009.

Y. Xiong, G. Li, Y. Zhou, K. Fernandes, R. Harrison, and Z. Xiong. Dynamic pricing models for used products in remanufacturing with lost-sales and uncertain quality. *International Journal of Production Economics*, pages 1–11, Apr. 2013. ISSN 09255273. doi: 10.1016/j.ijpe.2013.04.025.

S. H. Yoo, D. Kim, and M.-S. Park. Lot sizing and quality investment with quality cost analyses for imperfect production and inspection processes with commercial return. *International Journal of Production Economics*, 140(2):922–933, Dec. 2012. ISSN 09255273. doi: 10.1016/j.ijpe.2012.07.014.

L. a. Zadeh. The concept of a linguistic variable and its application to approximate reasoning-I. *Information Sciences*, 8(3):199–249, 1975.

J. Zak. The methodology of multiple-criteria decision making in the optimization of an urban transportation system: case study of Poznan City in Poland. *International Transactions in Operational Research*, 6, 1999.

K. Zeng, N.-y. Zhang, and W.-l. Xu. A comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators. *IEEE Transactions on Fuzzy Systems*, 8(6):773–780, 2000. ISSN 10636706. doi: 10.1109/91.890337.

W. Zhuan, Z. Qinghua, Y. Bo, and H. Wenwen. 4/R/I/T distribution logistics network 0-1 programming model and application. *Computers & Industrial Engineering*, 55 (2):365–378, Sept. 2008. ISSN 03608352. doi: 10.1016/j.cie.2008.01.001.

C. Zikopoulos and G. Tagaras. Impact of uncertainty in the quality of returns on the profitability of a single-period refurbishing operation. *European Journal of Operational Research*, 182(1):205–225, 2007.

C. Zikopoulos and G. Tagaras. On the attractiveness of sorting before disassembly in remanufacturing. *IIE Transactions*, 40(3):313–323, Jan. 2008. ISSN 0740-817X. doi: 10.1080/07408170701488078.

H. Zimmermann. Decision Making in Fuzzy Environments. In *Fuzzy set theory-and its applications*, volume 14, page 329. Springer, 2001.