UNIVERSITY OF KWAZULU-NATAL
SCHOOL OF ENGINEERING
MECHANICAL ENGINEERING

DEVELOPMENT OF A DECISION SUPPORT SYSTEM FOR DECISION-BASED PART/FIXTURE ASSIGNMENT AND FIXTURE FLOW CONTROL

Fentahun Moges Kasie
214584670

A thesis submitted in fulfilment of the degree of
Doctor of Philosophy in Engineering at the College of Agriculture,
Engineering and Science, University of KwaZulu-Natal.

January 2018

Supervisor: Professor Glen Bright

Co-Supervisor: Dr. Anthony Walker

Declaration by Supervisor

As the candidate's Supervisor I agree/do not agree to the submission of this thesis.

Signed:____________________________ Date:________________________

Professor Glen Bright

Signed:____________________________ Date:________________________

Dr. Anthony Walker

Declaration 1 - Plagiarism

I, Fentahun Moges Kasie, declare that

1. The research reported in this thesis, except where otherwise indicated, is my original research.

2. This thesis has not been submitted for any degree or examination at any other university.

3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

   a. Their words have been re-written but the general information attributed to them has been referenced.
   b. Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.

5. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

Signed:___ _________ Date:__08 January 2018_____

Fentahun Moges Kasie

Declaration 2 - Publications

DETAILS OF CONTRIBUTION TO PUBLICATIONS that form part and/or include research presented in this thesis (include publications in preparation, submitted, *in press* and published and give details of the contributions of each author to the experimental work and writing of each publication**).**

Publication 1

Kasie, F.M., Bright, G., and Walker, A, An intelligent decision support system for on-demand fixture retrieval, adaptation and manufacture, Journal of Manufacturing Technology Management (published), Vol. 28 No. 2, 2017, pp. 189 -211.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker - supervision.

Publication 2

Kasie, F.M., Bright, G., and Walker, A, Decision support systems in manufacturing: A survey and future trends, Journal of Modelling in Management (published), Vol. 12 No. 3, 2017, pp. 432-454.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker – supervision.

Publication 3

Kasie, F.M., Bright, G., and Walker, A, Developing an intelligent decision support system to determine a stable flow of fixtures, Proceedings of the 23rd ISPE Inc. International Conference on Transdisciplinary Engineering, Parana, Brazil, October 3-7, 2016, IOS Press, pp. 431 - 440.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker – supervision.

Publication 4

Kasie, F.M., Bright, G., and Walker, A, Integrating artificial intelligence and simulation for controlling steady flow of fixtures, Proceedings of the 28th International Conference on CARs & FoF 2016, West Bengal, India, January 6-8, 2016, Springer, pp. 137 - 147.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker – supervision.

Publication 5

Kasie, F.M., Bright, G., and Walker, A, Stabilizing the flow of fixtures using fuzzy case-based reasoning and discrete-event simulation, 27th International Conference on Flexible

Automation and Intelligent Manufacturing, FAIM2017 (accepted for oral presentation), Modena, Italy, June 27-30, 2017.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker – supervision.

Publication 6

Kasie, F.M., Bright, G., and Walker, A, Estimating cost of new products using fuzzy case-based reasoning and fuzzy analytic hierarchy process, Proceedings of the 24th ISPE Inc. International Conference on Transdisciplinary Engineering, Singapore, July 10-14, 2017, IOS Press, pp. 969 - 976.

Contribution: F.M. Kasie - research and writing; Prof. G. Bright and Dr. A. Walker – supervision.

Signed:___ ____________ Date:___08 January 2018___

Fentahun Moges Kasie

Acknowledgements

Firstly, I would like to express my deep sense of gratitude to my supervisor, Professor Glen Bright, for providing me his precious advices, recommendations and guidance to complete this work within the planned time. His encouragement and support was very valuable throughout the course of this research and during the publication of papers in international journals and conference proceedings.

I would like to thank my co-supervisor, Dr. Anthony Walker, for his frequent comments starting from the proposal preparation. His support was outstanding during the selection of the research methods. Frequent meetings and discussions with him helped me to synthesise a new methodological approach in this study.

I am thankful to the examiners of the thesis for their constructive comments and feedback. Their comments were highly valuable for the improvement of this thesis.

I am grateful to my wife, Elfnesh Girma, for her support and understanding and allowing me to work in extra hours during the study period. In addition, I am thankful to my daughter, Arsema, and my son, Michael, for providing me invaluable joys and amusements.

Finally yet importantly, I thank the Almighty God for granting me the power, courage and wisdom to finish this study.

Abstract

An intense competition in a dynamic situation has increased the requirements that must be considered in the current manufacturing systems. Among those factors, fixtures are one of the major problematic components. The cost of fixture design and manufacture contributes to 10-20% of production costs. Manufacturing firms usually use traditional methods for part/fixture assignment works. These methods are highly resource consuming and cumbersome to enumerate the available fixtures and stabilise the number of fixtures required in a system.

The aim of this study was to research and develop a Decision Support System (DSS), which was useful to perform a decision-based part/fixture assignment and fixture flow control during planned production periods. The DSS was designed to assist its users to reuse/adapt the retrieved fixtures or manufacture new fixtures depending upon the state of the retrieved fixtures and the similarities between the current and retrieved cases. This DSS combined Case-Based Reasoning (CBR), fuzzy set theory, the Analytic Hierarchy Process (AHP) and Discrete-Event Simulation (DES) techniques.

The Artificial Intelligence (AI) component of the DSS immensely used a fuzzy CBR system combined with the fuzzy AHP and guiding rules from general domain knowledge. The fuzzy CBR was used to represent the uncertain and imprecise values of case attributes. The fuzzy AHP was applied to elicit domain knowledge from experts to prioritise case attributes. New part orders and training samples were represented as new and prior cases respectively using an Object-Oriented (OO) method for case retrieval and decision proposal. Popular fuzzy ranking and similarity measuring approaches were utilised in the case retrieval process.

A DES model was implemented to analyse the performances of the proposed solutions by the fuzzy CBR subsystem. Three scenarios were generated by this subsystem as solution alternatives that were the proposed numbers of fixtures. The performances of these scenarios were evaluated using the DES model and the best alternative was identified. The novelty of this study employed the combination of fuzzy CBR and DES methods since such kinds of combinations have not been addressed yet. A numerical example was illustrated to present the soundness of the proposed methodological approach.

Keywords: Decision support systems, case-based reasoning, analytic hierarchy process, fuzzy set theory, object-oriented methods, discrete-event simulation, fixtures.

Table of contents

List of figures

List of tables

List of acronyms

AHP	Analytic Hierarchy Process
AI	Artificial Intelligence
ANN	Artificial Neural Network
CAFD	Computer Aided Fixture Design
CBIS	Computer-Based Information System
CBR	Case-Based Reasoning
DBMS	Database Management System
DES	Discrete-Event Simulation
DSS	Decision Support System
EDP	Electronic Data Processing
FCBR	Fuzzy Case-Based Reasoning
FMCDM	Fuzzy Multi-Criteria Decision-Making
FMS	Flexible Manufacturing System
GA	Genetic Algorithm
GDSS	Group Decision Support System
IDSS	Intelligent Decision Support System
IJDSS	Intelligent Just-in-time Decision Support System
KB	Knowledge-Based
KPI	Key Performance Indicator
MADM	Multi-Attribute Decision-Making
MCDM	Multi-Criteria Decision-Making
MIS	Management Information Systems
MLT	Manufacturing Lead Time
NN	Nearest Neighbour
ODSS	Organizational Decision Support System
OLAP	On-Line Analytical Process
RBR	Rule-Based Reasoning
ROC	Rank Order Clustering
SAW	Simple Additive Weighting
SMART	Simple Multi-Attribute Rating Technique
SMARTER	Simple Multi-Attribute Rating Technique Exploiting Ranks
SMARTS	Simple Multi-Attribute Rating Technique with Swings

CHAPTER 1

1. INTRODUCTION

1.1 Problem background

An intense competition in a dynamic and turbulent market has significantly increased the requirements that must be considered in the current manufacturing systems. The major requirements are flexibility, reduced product lifecycle, lower production volume and higher product mix, more responsiveness to customer requirements, better quality of products and an efficient utilisation of resources. Among several activities involved in manufacturing systems, fixtures are one of the limited resources, Suri and Whitney [122] and major problematic components, Boyle *et al.* [23], which need special considerations. Fixtures are required to hold, support and locate workpieces for specific processes or assembly operations. They directly affect the quality of products, the productivity of processes and the cost of products as presented in Kumar and Paulraj [74], Ostojic *et al*. [92], Peng *et al*. [97] and Wang *et al*. [134]. The effort on designing and fabricating fixtures significantly affects the production cycle in improving the current products and developing new products. The costs of fixture design and manufacture contribute to 10-20% of the total cost of manufacturing, Bi and Zhang [20]. These costs of fixtures can rise if the available fixtures are not well managed and utilised. With reference to this problem, it was visible that an appropriate strategy should have been developed for fixture/part assignment and control decisions to reduce operational costs, improve system productivity and enhance on-time delivery.

Traditionally, fixtures are ordered and assigned to their corresponding workpieces through manual and trial-and-error methods. These methods are highly resource consuming and cumbersome to manage the existing and newly manufactured fixtures. They are unable to enumerate the available fixtures and determine the stable number of fixtures required in manufacturing processes during a specified production period. The on-demand availability of fixtures significantly affects the flexibility, responsiveness, throughput rate, resources utilisation and delivery rate of manufacturing firms when product orders are processed. Having a few specialised or general purpose fixtures causes unnecessary machine downtime costs and having too many fixtures results unnecessary holding costs and resources wastage, Stecke [120]. In order to alleviate such kinds of problems, systematic fixture assignment and control techniques must be designed for the current manufacturing processes. Part/fixture assignment and fixture flow control is one of the complex problems in manufacturing, which was not adequately researched in the past because of two major reasons.

a) Traditional manufacturing firms are highly focused on the issues of part planning rather than fixture planning. This conventional approach of planning causes low resources utilisation and poor performances in the manufacturing sector, Özbayrak and Bell [93] and Rahimifard and Newman [104].
b) In the past, research findings on fixture planning were mainly concerned on the problems of fixture design and manufacture with the help of Computer-Aided Fixture Design (CAFD) facilities. Little attention was provided to the management of the available fixtures and their flows, Rahimifard and Newman [105].

Based on the identified research gap, the problem statement of this study was formulated in Figure 1-1. This figure presents a bounded fixture supply, storage and manufacture loop in manufacturing systems. Fixtures usually flow from place to place (e.g. from a work centre to a storage and vice versa) in the loop of manufacturing systems. In order to stabilise the flow of fixtures, a decision-based part/fixture assignment should be done at order arrival times. These decisions can be used to determine the stable number of fixtures required in the system during any planned production period. The decisions should be proposed with the help of an appropriate DSS.

According to the research problem formulated in Figure 1-1, the on-demand fixture retrieval and manufacture system can execute a decision-based fixture/part assignment and control to the current order arrivals using an appropriate DSS. New product orders from the manufacturing system must incorporate the necessary descriptions of part order attributes as problem descriptions. These descriptions should include the required process plan sets and the crucial attributes of product orders that can characterise these order arrivals for decision-based part/fixture assignments. The manufacturing system (e.g. a machining centre, an assembly line, a welding station, etc.) should receive part/fixture collectives based on specific decisions made at the on-demand fixture retrieval and manufacture system. The main decision alternatives were proposed as follows:

- Retrieve a fixture and assign. This decision alternative should recommend the reuse of the retrieved fixtures without any revisions for new part orders.
- Retrieve a fixture, adapt and assign. As this decision alternative is passed, the retrieved fixtures should be modified to adapt them for new order arrivals. The modifications can

be reconfiguring modular fixtures, adding some features into general-purpose or specialised fixtures.

- Manufacture a new fixture and assign. This decision alternative should encourage the manufacture of new fixtures when adaptations are impossible in any ways.
- Remove and manufacture. When the retrieved fixture is in a failed state, this decision alternative has to recommend a removal of the retrieved cases and replacing them with the current cases together with the newly manufactured fixtures.

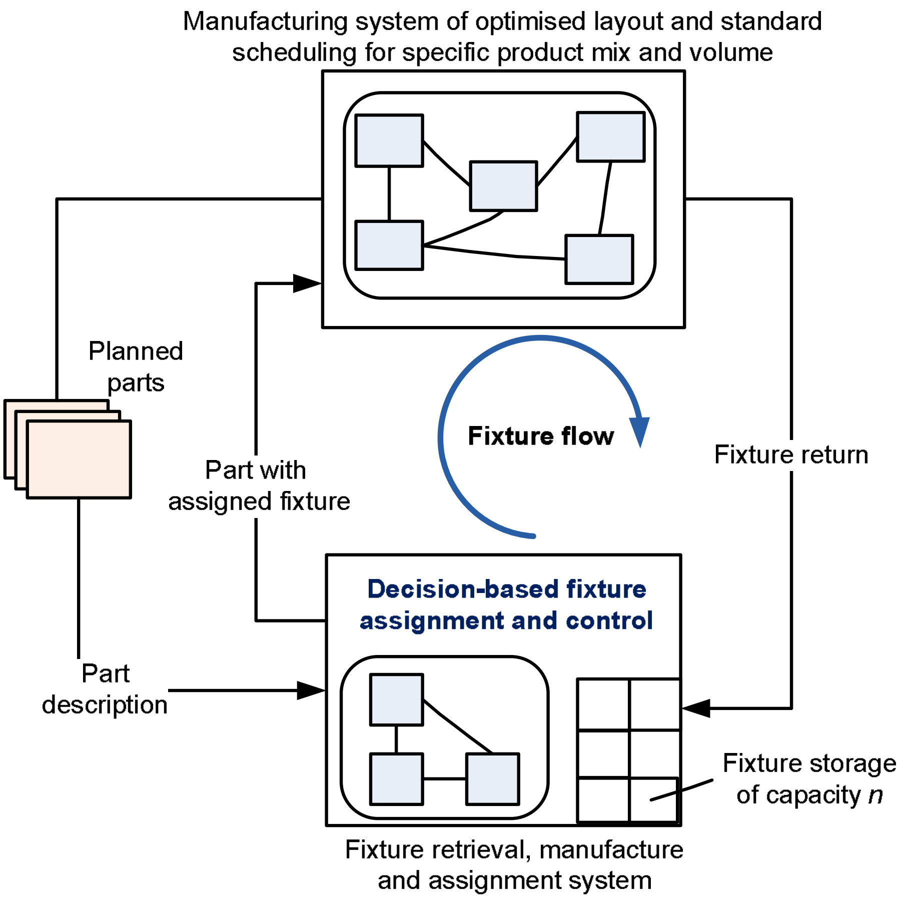


Figure 1-1: Bounded fixture supply, storage and manufacture loop

A fixture assignment strategy using these decision alternatives was called decision-based part/fixture assignment in this study. By implementing these decision-based assignments with the help of the right DSS, the users of this DSS can stabilise the number of fixtures flowing in the system. In other words, they can determine the fixture storage capacity n in Figure 1-1. In manufacturing situations, the number of flowing fixtures rapidly increases in the system at early stages of production periods, and it slowly increases and becomes stable as the system becomes more matured. The number of fixtures in a matured state of the system was named stable (steady state) number of fixtures in this research work.

1.2 Aim and objectives

This study researched and developed a DSS in order to carry out a decision-based part/fixture assignment and fixture flow control. This section briefly describes the major activities that should be performed to articulate the research problem.

1.2.1 Aim

The aim of this study was to research and develop a DSS that operates on simple decision sets in order to ensure *n*-bounded growth in fixture flows. The proposed DSS stabilised the flow of fixtures in manufacturing systems in which it serves, during a planned production period for a specific product mix and volume.

1.2.2 Objectives

In order to attain the aim of this study, the following specific objectives were outlined.

- Research the current state of the arts in DSS and identify areas of original contribution potentials.
- Research and develop a DSS framework that integrates CBR, RBR, fuzzy set theory and Multi-Attribute Decision-Making (MADM) approaches.
- Construct and represent fuzzy cases using an OO method.
- Research and develop the right case retrieval and retaining approach.
- Implement an artificial manufacturing environment in DES software to support the research and development of the DSS.
- Validate and test the DSS model with respect to various decision parameters in DES software.

1.3 Methodological approach

This study used the combination of both quantitative and qualitative methodological approaches in computer based laboratory environments. A numerical example was illustrated in order to show the applicability of the proposed methodological approach. Quantitative approaches were employed in quantitative situations such as case representations, fuzzy rankings, AHP analyses, DES analyses and case similarity measures using the inverse of the

Euclidean distance. Qualitative approaches were utilised in fuzzy situations like case attribute representation and selection, rating the importance of case attributes, evaluating the state of the retrieved fixtures and decisions proposal activities.

CBR and DES were the principal research methods in this study. Fuzzy set theory, RBR and MADM (specifically the AHP) approaches were combined with CBR in order to construct the AI component of the researched DSS. New part order arrivals as new cases and training samples (previous orders) as prior cases were represented using an OO method for case retrieval and decision proposal strategies. Four categories of product attributes such as numerical values, symbolic or descriptive terms, nominal values and linguistic terms were incorporated in the case representation process. The inverse of the Euclidean distance, which is one of the popular pattern recognition and matching functions, was applied to measure the similarities between new and prior cases for decision-based part/fixture assignments.

The fuzzy CBR system was used to represent the vague and uncertain values of case attributes for accommodating the required flexibility in case representations and decision analyses. The fuzzy AHP was implemented to elicit, and represent domain knowledge and judgements of human experts for ranking the weights of case attributes. Fuzziness was assimilated in the proposed DSS to articulate unstructured knowledge in human thoughts and decision-making. Popular fuzzy ranking methods were exploited to defuzzify this vague and imprecise knowledge in the decision-making process.

A DES model was implemented to analyse the performances of the proposed solutions by the fuzzy CBR subsystem of the DSS. This model was useful to minimise the uncertainties and risks of the proposed solutions due to the lack of knowledge and experience in case construction and rating case attributes. The researched DES model was used to predict the near future situations of the proposed solutions instead of using historical data to validate their accuracies. It should be noted that the proposed solutions in this study were the proposed stable numbers of fixtures required in a particular process.

1.4 Research contribution

It was stated that a decision-based part/fixture assignment and fixture flow control was one of the complex problems that have not been adequately studied in the past. Past studies regarding the determination of the stable number of fixtures within manufacturing processes in specific production periods were missed out. This study articulated this problem using a systematic fixture assignment and control strategy. This research gap was identified and treated the area of an original contribution potential.

A new methodological approach was synthesised in this study by combining the existing complex theories in AI and DES. The fuzzy CBR subsystem of the researched DSS performed decision-based part/fixture assignments in parallel to any standard and feasible part plans. Following these assignments, the stable numbers of fixtures required within specific production periods were determined as alternative solutions from the fuzzy CBR component. The performances of these alternative solutions were validated and predicted with the help of a DES model. With reference to the current literature in DSS, the combination of fuzzy CBR and DES methodologies has not been exploited yet to solve such kinds of complex problems. It was implied that the methodological approach presented in this study is a significant contribution to the current DSS research.

The performances of the proposed solution alternatives were simulated in terms of specific key performance indicators such as machine utilisation, average stay time in a process and operational costs of fixtures. These key performance indicators revealed that the DSS could improve the utilisation of the available fixtures and other related resources, manufacturing lead-times and operational costs of processes under investigation. The relationship between the number of fixtures in a simulated machining process and operational costs of fixtures was determined by combining the fuzzy CBR and DES elements of the proposed DSS. This was regarded as a novel approach to determine the stable number of fixtures that could minimise the total operational costs of fixtures in manufacturing processes.

1.5 List of publications

In this section, journal articles and conference papers, which were published during the author's study period as a PhD student at the University of KwaZulu-Natal, are included.

a) KASIE, F.M., BRIGHT, G., AND WALKER, A. An intelligent decision support system for on-demand fixture retrieval, adaptation and manufacture, Journal of Manufacturing Technology Management (published), Vol. 28 No. 2, 2017, pp. 189 - 211.
b) KASIE, F.M., BRIGHT, G., AND WALKER, A. Decision support systems in manufacturing: A survey and future trends, Journal of Modelling in Management (published), Vol. 12 No. 3, 2017, pp. 432-454.

c) KASIE, F.M., BRIGHT, G., AND WALKER, A. Developing an intelligent decision support system to determine a stable flow of fixtures, Proceedings of the 23rd ISPE Inc. International Conference on Transdisciplinary Engineering, Parana, Brazil, October 3-7, 2016, IOS Press, pp. 431 - 440.
d) KASIE, F.M., BRIGHT, G., AND WALKER, A. Integrating artificial intelligence and simulation for controlling steady flow of fixtures, Proceedings of the 28th International Conference on CARs & FoF 2016, West Bengal, India, January 6-8, 2016, Springer, pp. 137 - 147.
e) KASIE, F.M., BRIGHT, G., AND WALKER A. Stabilizing the flow of fixtures using fuzzy case-based reasoning and discrete-event simulation, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017 (accepted for oral presentation), Modena, Italy, June 27-30, 2017.
f) KASIE, F.M., BRIGHT, G., AND WALKER A. Estimating cost of new products using fuzzy case-based reasoning and fuzzy analytic hierarchy process, Proceedings of the 24th ISPE Inc. International Conference on Transdisciplinary Engineering, Singapore, July 10-14, 2017, IOS Press, pp. 969 - 976.

1.6 Thesis layout

This study is organised into six chapters. Each part of the thesis elaborates the different aspects of the research work. The next five chapters are briefly described in this section.

Chapter 2 reviews the literature and theoretical grounds of the study. This chapter starts with a review in DSS theories such as its historical evolutions, frontiers and components. The role of DSS in manufacturing is also reviewed. The relationships between DSS and AI theories are deliberated. Specifically, the relevance of fuzzy CBR in DSS development, with references to their common objectives, is reviewed. The significance of integrating CBR and other knowledge-based approaches such as fuzzy MADM, RBR and OO case representation approaches are discussed. The roles of DES in decision-making and the importance of combining DES and AI approaches in manufacturing are reviewed. Finally, a theoretical framework for this study is synthesised in order to present the contributions of this study to the existing knowledge in DSS and articulate the stated research problem.

Chapter 3 deals with the development of the researched DSS in this study. It elaborates the methods and steps involved to carry out the research. Special attention is provided to the combination of the fuzzy CBR and DES elements of the DSS. It describes the case construction process incorporating case attributes identification and case representation methods. Rating the importance of case attributes using the fuzzy AHP and fuzzy ranking methods are deliberated. The major roles of the fuzzy CBR subsystem such as case retrieval, decision proposal and case retaining activities are presented including several analytical equations and knowledge-based guiding rules. Finally, the need of a DES model in order to validate and test the solutions proposed by the fuzzy CBR subsystem is discussed.

Chapter 4 implements the methods that were explained in Chapter 3 using a numerical example. The numerical example is illustrated by considering an ideal milling operation centre in computerised laboratory environments. Product orders are treated as fuzzy cases in terms of twelve product attributes using an OO case representation approach. The weights of these case attributes are determined using the fuzzy AHP. For case retrieval and decision analyses, the equations and rules presented in Chapter 3 are exploited to demonstrate the numerical example. Several Java library classes and methods are employed to support the case retrieval and retaining processes. Three alternative solutions are proposed by the fuzzy CBR component of the proposed DSS. The performances of these three scenarios are analysed and predicted using a DES model in order to select the best solution alternative.

Chapter 5 discusses the findings presented in Chapter 4. The problem statement and objectives are restated and discussed to answer the research questions, which were outlined as specific research objectives in this introductory chapter. The implications of the research methods with respect to the current theories in Chapter 2 and the research findings in Chapter 4 are explained. The implications of this study in the view of the combination between a fuzzy CBR system and a DES model are discussed. The relationship between the numbers of fixtures flowing in manufacturing processes and operational costs of fixtures are demonstrated. Lastly, the research contributions and limitations are explained with reference to the research findings.

Chapter 6 incorporates the conclusions and suggestions for future research. It summarises the findings and discussions of the overall study as conclusions. It suggests some important ideas, which are beyond the scope of this study, as the directions for future research.

1.7 Summary

This introductory chapter highlighted the problems of fixture planning in the current research. A decision-based part/fixture assignment and fixture flow control was identified as the current research gap in the current manufacturing. Depending upon this gap, this study aimed at researching and developing a DSS in order to address this research problem.

Specific objectives were outlined to meet the aim of the study. A new methodological approach that combined a fuzzy CBR system and a DES model was proposed. This combined approach was regarded as a novel contribution to the current studies in DSS. In addition, journal articles and conference papers, which were published during this study, were outlined as a list of publications. Finally, a summary of each chapter was presented in order to depict the structure of the thesis.

CHAPTER 2

2. REVIEW OF LITERATURE AND THEORIES

2.1 Introduction

This chapter is intended to reinforce the statement of the problem presented in the previous chapter with the help of the existing sources of literature and theories in DSS. Depending upon a review of literature from various related sources of literature, the areas of original contribution potential of this work are identified and discussed. It starts with a review in DSS theories such as its historical evolutions, frontiers and components. The definitions of DSS in different contexts are reviewed. The dimensions of manufacturing systems, which have not been addressed by the current DSS research, are also identified and explained. Next, the relationships between DSS and AI theories are elaborated. Specifically, the roles of fuzzy CBR systems in DSS development are reviewed.

The combination fuzzy CBR and other knowledge-based approaches such fuzzy MADM, RBR and OO case representation approaches in DSS development are discussed. The importance of DES in decision-making and the relevance of combining DES and AI technologies in the development of DSS in manufacturing systems are reviewed. Finally, a theoretical framework of this study is synthesised based on the current research gaps in DSS. Special emphasis is given to the combination between fuzzy CBR and DES subsystems of the proposed framework. This is a significant step in this study in order to contribute into the current knowledge in DSS and articulate the stated research problem.

2.2 Theoretical ground of DSS

The development of DSS is evolutionary and their scope was limited to support individual decision makers with the help of available computer applications during their inception. Their current applications are incredibly vast following the technological advancement in information technology, intense competitions among firms, volatile features of customer needs and regulatory requirement for societal welfare. Due to their versatile applications, presently, it is challenging to define their boundaries and identify their components in explicit ways.

2.2.1 Evolution of DSS

DSS have passed different development stages depending upon the innovation of the driving technologies. The research in DSS has been one of the attractive research topics since their inception at the beginning of 1970s and faced different challenges because of the rapid development and innovation in the field of information technology, Liu *et al*. [83]. The use of computers in organisations was significantly increased during the 1955 to 1971 period; however, a few were successful in the way in which management makes decisions, Gorry and Morton [56]. The cause of the failure was that managerial works were misunderstood by system developers, Arnott and Pervan [12]. Interactive computer tools were applied in decision-making in the 1960s, Eom *et al*. [47]. Their capabilities were limited to solve structured managerial problems alone, Power [99]. In the early 1970s, researchers and practitioners were inspired to use interactive computer-based technologies to solve semi-structured and unstructured problems at different managerial levels rather than handling structured and routine tasks as reviewed in Eom and Lee [46], Shim *et al*. [114], Turban *et al*. [126], Power [98] and Liu *et al.* [83].

The concept of DSS was first coined by Scott Morton in 1971 by merging two major research streams: (a) the theoretical studies of organisational decision-making occurred at the Carnegie Institute of Technology and (b) the technical works carried out on interactive computer systems largely at the Massachusetts Institute of Technology (MIT), in the 1950-60s, Keen and Morton [69]. In 1971, Gorry and Morton [56] developed a prominent management information systems framework, which was regarded as a classical decision support tool for solving semi-structured and unstructured problems. This framework was developed by integrating Anthony's [11] categories of management activities such as strategic planning, management control and operational control and Simon's [115] descriptions of decision types such as programmed decisions and non-programmed decisions. Following Morton's integration concept, several systems were proposed. Gerrity [54] developed an integrated man-machine/computer decision system in 1971 to support investment managers in their day-to-day decisions in administrating clients' portfolio. Little [82] developed a market-mix model for product promotion, pricing and adverting tasks in 1975. Alter [5] studied an 'interactive problem solving' approach, which describes the synergy of man-computer interactions to solve ill-defined problems in 1977.

In the 1970s, the DSS research was largely focused on supporting personal decision-making strategies. In the 1980s, the attention of DSS research was gradually shifted into Group

Decision Support Systems (GDSS) and Organisational Decision Support Systems (ODSS) in order to address more complex situations and enhance the effectiveness of DSS at intra-organisational and inter-organisational levels as stated in Eom and Lee [46], and Sprague and Carlson [118]. AI techniques, specifically expert systems were embedded as important tools into DSS in the middle of 1980s, Eom and Lee [46]. It was described in Keen and Morton [69], AI techniques would be the greatest potential for improving decision-making tasks. Bonczek *et al*. [21] proposed a conceptual framework in order to integrate AI technologies in the DSS development process. Data warehousing, On-Line Analytical Processing (OLAP), data mining, the World Wide Web Technology, OO methodologies, intelligent agents, the Internet and corporate intranet emerged as integral parts for building DSS in the 1990s, Shim *et al*. [114] and Power [98]. At the beginning of the 21st century, complex DSS incorporating knowledge management systems, information portals, business intelligence and communication-driven DSS were introduced in integrated Web environments so as to meet the challenges of competitiveness, Power [98]. For more information regarding a historical overview of DSS, interested readers are referred to Power [100] and Arnott and Pervan [12].

2.2.2 Frontiers of DSS

The term DSS has been contextualised in different ways by several DSS scholars and practitioners. People in different backgrounds and experiences perceive DSS in quite different perspectives. These contextual differences mainly depend upon the tasks that should be done by DSS (structured-unstructured), challenges of defining the boundaries of DSS in relation to Management Information Systems (MIS) and management science models, and conceptual variations related to their evolutionary developments. Sprague [118] and Keen [68] stated that some writers considered DSS advancements of MIS, management science and Electronic Data Processing (EDP) systems and others considered DSS a subset of MIS. Er [48] explained that EDP and DSS are complementary halves of Computer-Based Information Systems (CBIS). In addition, the author described that the term DSS do not have a universally accepted definition. Turban *et al*. [126] argued that DSS is a content-free term, in which the definitions of DSS vary depending upon situations. According to Keen [68], the difficulty to define the term DSS is because of the difficulties to identify the boundaries among DSS, management science and MIS. As stated in Keen and Morton [69], the authors briefly described the differences among MIS, management science and DSS in terms of their areas of impact on, payoffs to the organisation and their relevance to the users. Based on

Thierauf [124], MIS was viewed as a subset of DSS to solve well-structured problems; however, DSS can solve problems with varying degrees of well to ill definition levels.

According to Sprague [118], DSS are powerful weapons in information systems for improving the effectiveness of users in organisations, which draw on transaction processing systems and interact with the entire information systems to support the decision-making process. Alter [6] argued that the purpose of DSS should be to support managers, who are responsible for making and implementing decisions instead of replacing them. The author stated the boundaries of EDP and DSS including their characteristics. According to his explanation, EDP systems are designed for data storage and retrieval, transaction processing, record keeping and business-reporting activities in order to reduce costs, improve the accuracy and efficiency of day-to-day operations; however, DSS are the legitimate uses of interactive computer-based systems to improve individual and organisational effectiveness rather than increasing efficiency in data processing tasks.

According to Gorry and Morton [56], DSS were described as interactive computer-based systems, which aim at supporting managers to solve semi-structured and unstructured problems in organisations. Keen [68] stated the scope of DSS and non-DSS. According to the author, the term DSS is relevant to situations where a final system can be developed only through an adaptive process of learning and evolution. Learning, adaptation and evolution are very essential elements while building DSS. This makes DSS different from management science, MIS and other traditional models. This ideology of DSS strengthens the previous notions of the DSS stated in Gorry and Morton [56]. They stressed that DSS are not intended to involve in routine data processing activities. However, they are useful to support managers to improve the quality of their decisions for solving unstructured and ill-defined problems; must be able to assist the evolution of managers' decision-making abilities through their understandings of the dynamically changing environments; must be educative; and managers can develop insights into the relationships between their decisions and the goals they desire to achieve.

In order to differentiate DSS from other systems, Sprague and Carlson [119] identified the following characteristics. DSS tend to be aimed at semi-structured and unstructured problems that the upper level managers typically face; they should use models or analytic techniques in integration with traditional data access and retrieval functions; and they have to be easy to use by non-computer professional users in an interactive mode. In addition, they should emphasise flexibility and adaptability to accommodate changes in decision-making environments. Er [48] added another characteristic of DSS that they support but do not replace the upper-level managers in decision-making, which is an important aspect to differentiate DSS from expert systems. According to Holsapple and Whinston [61], DSS must be capable to accomplish various tasks depending upon the existing circumstances. They acquire and maintain descriptive knowledge (e.g. record keeping, procedure keeping, rule keeping, etc.); present knowledge on an ad hoc basis in various customised ways and in standard reports; select any desired subsets of stored knowledge for presentation or deriving new knowledge; and interact directly with decision makers in such a way that the users have flexible choices. According to these characteristics, DSS are autonomous systems rather than supporting decision makers. This idea seems to be in contradiction to the roles of DSS proposed by other DSS advocates.

An important demarcation between DSS and MIS was proposed by Power [98]. According to the writer, in its narrow definition, MIS was treated as a management reporting system, which provides periodic, structured and paper-based reports to managers. In its broad definition, MIS was treated as an information system that could provide managers with on-line accesses to the required information. However, DSS were regarded as interactive real-time systems to respond to both unplanned and planned events. They can incorporate various analytical information systems, consult and interact with distributed target group members of decision makers, grow to enterprise wide DSS that could be connected to data warehouse systems and serve many managers within a company. He broadly defined DSS as "interactive computer based-systems that help people to use computer communications, data, documents, knowledge, and models to solve problems and make decisions". Power's definitions were immensely dependent upon the latest developments in the areas of information technology and DSS. In addition, Power strongly insisted that DSS are ancillary systems, which are not intended to replace skilled decision makers and they should be considered when two assumptions seem reasonable: (a) good information is likely to improve the quality of decisions; and (2) managers need computerised support to meet complex problems.

Another important perspective to define the boundaries of DSS is the structure of tasks that must be performed. DSS are designed to solve problems, which are semi-structured, Keen and Morton [69] and Keen [68]; semi-unstructured and unstructured, Gorry and Morton [56], and Sprague and Carlson [119]; structured, semi-structured and unstructured, Thierauf [124]; and unstructured, Bonczek *et al.* [21]. According to Eom and Lee [46], the definitions of DSS were reviewed and summarised as computer-based interactive systems that could support decision makers rather than replacing them; utilise data and models; solve problems with varying degrees of structured, semi-structured and unstructured; and focus on the effectiveness rather than the efficiency of decision processes.

2.2.3 Decision-making, problem solving and DSS

In the past, several frameworks were proposed to describe decision-making and problem solving approaches. A popular three phase decision-making framework was initially developed by Simon [115]. As presented in Sprague and Carlson [119], Forgionne [52] and Turban *et al*. [126], Simon's three-phase paradigm of intelligence, design and choice was modified by incorporating an implementation phase as the fourth element (Figure 2-1).

Figure 2-1 indicates that decision-making is a continuous process, which is a flow of activities within a loop from the intelligence to implementation phases. Successful decision-making results in solving a real problem and failure leads a return to the earlier phase i.e. continuous reviewing of the process is required. After solving the current problem, the decision makers investigate new real problems or opportunities and this never-ending process continues forever.

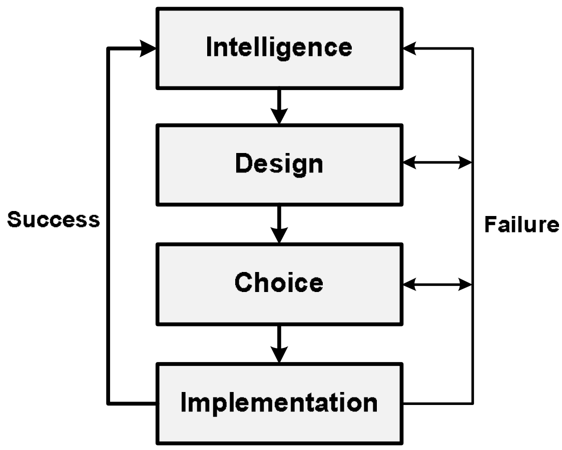


Figure 2-1: Decision-making process, adapted from Simon [115], Sprague and Carlson [119], Forgionne [52] and Turban et al. [126]

The major activities in the intelligence phase are problem or opportunity observation and understanding, data collection, problem identification and defining the statement of the problem. The design phase focuses on an accurate and a precise model formulation for the problem statement, development of decision alternatives, defining controllable and uncontrollable variables, and their relationships, setting criteria for choices, predicting outcomes and validation of models.

During the choice phase, the decision makers use explicitly models to evaluate, verify and test the proposed alternatives, and generate the recommended actions that best meet the decision criteria. The implementation phase incorporates pondering the analyses and recommendations, developing implementation plans and ensuring the required resources to execute the selected alternative(s).

Decision-making is the study of choosing preferable alternative courses of actions among several alternatives in order to meet specified decision objectives. Differentiating the terms decision-making and problem solving is somehow confusing as stated in Turban *et al*. [126], because they are highly interrelated to each other. In most cases, problem solving encompasses all four phases of the process; however, decision-making excludes the fourth phase (implementation).

MIS tools are mostly used in the intelligence phase to enhance information access, retrieval and processing efficiencies; management science tools are commonly applicable in the design and choice phases; however, DSS are utilised in all phases of the process as stated in Turban *et al*. [126], and Sprague and Carlson [119]. This can be seen as one of the ways to define the boundaries of DSS, MIS and management science.

2.2.4 Components of DSS

DSS are composed of different interacting components or subsystems. According to Sprague and Carlson [119], traditional DSS could be built using four major subsystems named dialog (user interface), database, model and DSS architecture network. Turban *et al*. [126] suggested the inclusion of a knowledge-based subsystem as an optional component in addition to Sprague and Carlson’s framework. Power [98] proposed an architecture of DSS construction with the help of his expanded DSS framework that incorporated database, model, communication and user interface components. He stated that the database could be replaced with the knowledge base and/or document base components depending upon the driving technologies of the system. Depending upon the recent state of the arts in DSS, a holistic framework is presented using Figure 2-2 to depict the components of DSS.

In traditional DSS, the database incorporates all the required information concerning the current situation, which can be managed by Database Management System (DBMS) software and it can be interconnected with the corporate data warehouse, Turban *et al*. [126]. The document base can either replace or work in parallel to the traditional database component to increase the effectiveness of the existing DSS. Document-based DSS are recently emerging to overcome the limitations of databased DSS. The latest research findings assured that only about 20% information is circulating in organisations in the form of structured and numerical data; however, the other remainder 80% information is hidden in unstructured documents, Tseng and Chou [125] and Feki *et al.* [50].

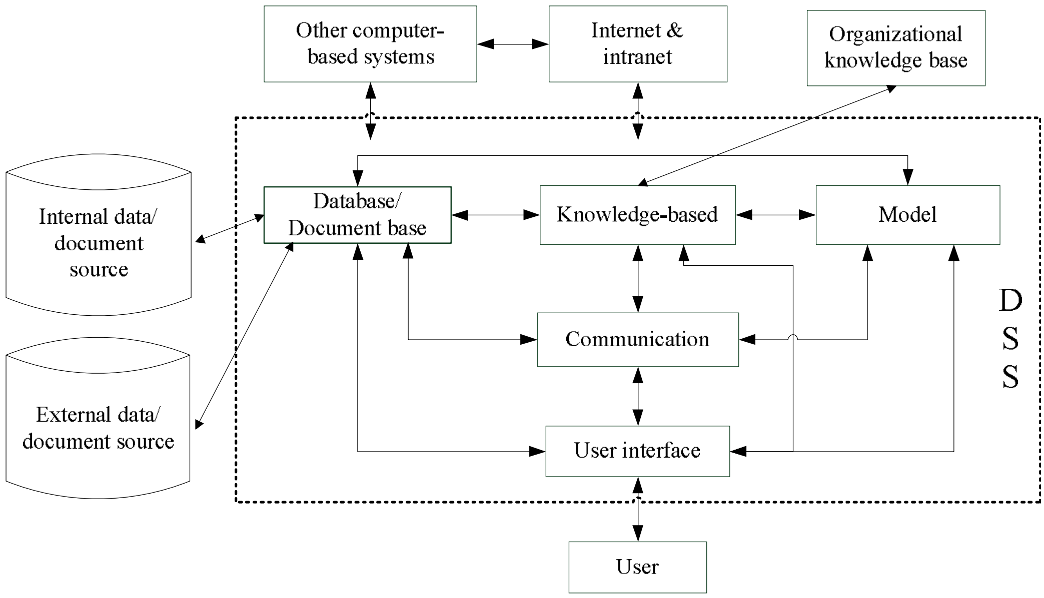


Figure 2-2: Framework for DSS construction, adapted from Turban et al. [126] and Power [98]

The knowledge-based component provides intelligent capabilities to the DSS and can be interconnected with the knowledge warehouse, Turban *et al.* [126] and Nemati *et al.* [88]. This component has a specialised problem-solving expertise in a particular domain of knowledge, Power [99]. The complexities and dynamics of business environments enforce the current DSS to be capable to adapt and accommodate the frequent changing needs of decision makers, Chuang and Yadav [36]. Integrating knowledge-based systems with DSS improves the quality of decision outputs, Turban and Watkins [127]. The knowledge-based subsystem uses the required data/documents from the database/document base. It acts as human consultants to assist decision makers in understanding, expressing, and structuring their problems, Angehrn and Lüthi [10]. Goul *et al.* [57] and Nichols and Goul [89] reviewed the pattern of AI-based research progresses and the influences of integrating AI technologies with DSS at personal, group, inter-organisational and intra-organisational levels.

The model component of the framework can comprise of many quantitative models like optimisation, statistical analyses, simulation, etc. that provide simplified representation capabilities for any DSS. These capabilities are required in order to build partially or fully customised analytical or simulation models depending upon the complexity of problems in consideration. Traditionally, models are used to analyse decision alternatives proposed by the users; however, in the knowledge-based systems, models generate various performance scenarios with reference to decision alternatives recommended by the knowledge-based subsystem of the DSS.

The communication component is an optional subsystem because it is not applicable in single-user DSS. It refers to how hardware is organised; how software and data are distributed in the system; and how components of the system are integrated and connected using a network, Web-server, client/server and mainframe, Power [98]. It can be included in multi-user DSS and excluded in single-user ones.

The user interface is one of the most useful components of any DSS and sometimes called DSS generator, query and reporting tool and front-end development tool. It consists of dialogue, maps, menus, icons, representations, charts, graphs and Web-browser, Power [98]. It allows the interactions between the system and the user to communicate with and commands the DSS, Turban *et al*. [126].

Depending upon the dominance of the required components that drives the DSS, Power [99] classified DSS into five major categories; data-driven, model-driven, document-driven, knowledge-driven and communication-driven systems. In addition, the classifications of DSS were studied in Alter [6], Hackathorn and Keen [58] and Holsapple and Whinston [61]. The importance of such classifications is to identify the types of DSS that must be developed as the potential solution to specific decision problems and to combine the most useful features of each system to a problem in consideration, Alter [6]. This avoids the vagueness regarding the term DSS and differentiates more clearly what types of DSS are being studied and built in a specified problematic situation using an appropriate driving technology, Power [99] and Power and Sharda [101].

2.3 DSS in manufacturing

Manufacturers must quickly respond to changes in customer needs by making efficient adaptations to their internal processes, in line with their customer requirements. This is a substantial challenge for the current manufacturing firms in order to compete in dynamically changing and uncertain environments. Innovation in the design and operation of manufacturing systems is required to meet this challenge. DSS have been implemented as one of the important tools for manufacturing systems to articulate several complex problems and faced various challenges during their implementation processes. Alter [7] was the first to identify the problems of DSS implementation for manufacturing in unstructured situations. Suri and Whitney [122] proposed a DSS on long-term, medium-term and short-term bases to improve the performance of Flexible Manufacturing Systems (FMS). Various research-based strategies have been proposed in DSS studies in order to meet the challenges of manufacturing; as a result, different solutions and frameworks have been presented. Simulation-based DSS were widely accepted because of their capabilities to model complex and dynamic systems, which are beyond the scopes of analytical models accordingly Jahangirian *et al.* [64], AlDurgham and Barghash [3] and Ali and Seifoddini [4]. Recent studies have revealed that purely simulation-based approaches are time-consuming and challenging to their users for further analyses and interpretation of results from several scenarios, Chan and Chan [27] and Pehrsson *et al.* [96].

As stated in the introductory chapter, the current DSS in manufacturing have still lacked to address other important dimensions of manufacturing systems such as integrated fixture, jig and tool management problems. This is because part scheduling problems have been believed as major issues and others like fixture and tool planning issues have been treated as minor problems in today's manufacturing systems, Özbayrak and Bell [93]. This misunderstanding causes low resources utilisation and poor performances as argued in Özbayrak and Bell [93] and Rahimifard and Newman [104]. Another limitation is research findings on fixture planning problems in the past were mostly focused on the problems of fixture design and manufacture using CAFD facilities. Different techniques were proposed to make fixture designs more reconfigurable and modular to accommodate various types of product orders. These techniques were reviewed in Wang *et al.* [134] and Boyle *et al.* [23]. For example, the importance of adapting and utilising previous fixture designs were studied in Sun and Chen [121], Li *et al.* [78], Boyle *et al.* [22], Wang and Rong [133], Peng *et al.* [97] and Zhou *et al.* [143] using CBR approaches. The adaptations of previous fixture designs were addressed in past studies; however, studies in part/fixture assignment and control problems, which can improve the utilisation of the available fixtures, have not been well studied in the past. The existing sources of literature revealed that this research area needs more explorations at present and in the future. This is because a few studies were conducted to assign fixtures to their corresponding part orders and control the stable flows of fixtures in manufacturing processes.

Rahimifard and Newman [104] presented a simulation-based multi-flow scheduling system for the simultaneous planning of workpieces, fixtures and cutting tools in flexible machining cells; namely workpiece dominated, tool dominated and fixture dominated planning strategies. The fixture dominated planning strategies were two types: the first is fixture cluster-based job allocation, which clusters jobs into similar fixture requirements to assign specialised fixtures; and the second is fixture availability-based job allocation, which assigns jobs into the available modular fixtures. In addition, Rahimifard and Newman [105] proposed an integrated planning and control system, which generates short-term schedules for the flows of workpieces, fixtures and cutting tools. These planning strategies are useful to solve part/fixture assignment problems and utilise the available fixtures alone; however, they lack to address demand-driven adaptation and learning aspects of DSS. Their systems do not articulate the situations where the available fixtures should be reused/adapted and new fixtures should be manufactured. These two problems situations are very important to determine the stable number of fixtures required in a given manufacturing process. In other words, DSS must be developed to support people who are required to solve complex problems through the adaptive process of learning and evolutions to accommodate changes in dynamic environments in the near future, Power [99] and Keen [68].

2.4 Artificial intelligence for DSS development

According to Holsapple and Whinston [61], Intelligent Decision Support Systems (IDSS) are DSS that make extensive use of computer-based methods from the field of AI systems. The term knowledge-based DSS was used to name IDSS as presented in Turban *et al*. [126], Özbayrak and Bell [93], and Benz and Mertens [17]. In addition, Turban *et al*. [126] classified IDSS into two major categories: the first were rule-based expert systems and the second were advanced IDSS that could use CBR systems, Artificial Neural Networks (ANNs), Genetic Algorism (GA) and fuzzy set theory. The former has specialised to utilise domain knowledge and expertise in specific subject areas to solve specific problems. The objectives of such systems are to replace human experts in problem solving and important decision-making strategies. The later works as an advisory system in order to propose decisions and solutions for several open-ended problems in unstructured situations. Beemer and Gregg [15] presented the architects of both rule-based expert systems and case-based advisory systems. The researchers classified their recent research reviews into expert systems and advisory systems.

Depending upon the problem statement of this study, this section focuses on the interactions among CBR, fuzzy set theory, RBR and OO case representation approaches.

2.4.1 Fuzzy set theory

In real situations, problems cannot be articulated using crisp data alone. Human thoughts and decisions are usually uncertain and imprecise. They are puzzling to define them within clear boundaries. Knowledge and experience can be reasonably expressed in terms of linguistic terms, fuzzy numbers and fuzzy sets rather than crisp values in real problem situations. Because of these requirements, decision-making models are required to incorporate vagueness and uncertainties inherited to human thoughts. Fuzzy set theory is used to address these important problems and grade the degree of membership of objects in vague and uncertain environments, Zedah [142]. It is usually essential to solve problems in which their descriptions are imprecise and uncertain to define their constraints and consequences of possible actions that are not precisely known, Bellman and Zadeh [16].

The fundamental definitions of fuzzy set theory, which are applicable to this study, are defined with references to Jang *et al.* [65] and Zimmermann [144]. These definitions are summarised in the next six definitions.

Definition 2.1. A fuzzy set A is in a universe U whose elements are generically denoted by x, then A is defined as a set of ordered pairs: $A = \{(x, \mu_A(x)) \,/\, x \in U\}$. The membership function of x to fuzzy set A is denoted by $\mu_A(x)$. The graded membership value $\mu_A(x)$ is a real number within the interval [0, 1].

Definition 2.2. A fuzzy set A of the universe U is normal if it is always possible to find at least an element $x \in U \,/\, \mu_A(x) = 1$.

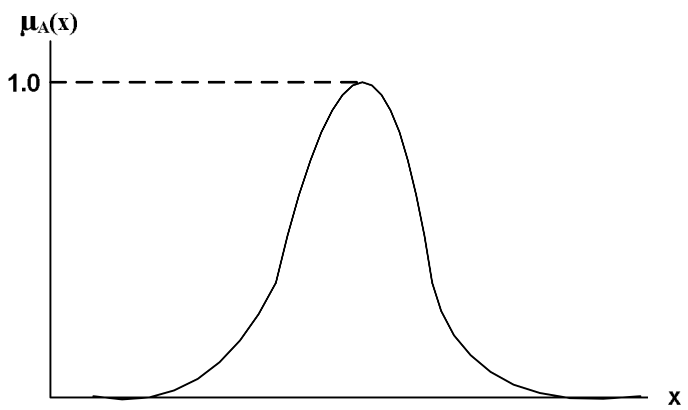


Figure 2-3: A fuzzy number A

Definition 2.3. A fuzzy set A of the universe is convex if and only if for any x_1, x_2 Є U and any λ Є [0, 1]; then $\mu_A(\lambda x_1 + (1 - \lambda) x_2) \geq \min\{\mu_A(x_1), \mu_A(x_2)\}$.

Definition 2.4. A fuzzy number A is a fuzzy set in the universe U, which satisfies the conditions for normality and convexity as presented in Figure 2-3.

In addition, special fuzzy numbers such as triangular and trapezoidal fuzzy numbers, which are employed in this study, are depicted in Figure 2-4. A_1 is a triangular fuzzy number and A_2 is a trapezoidal fuzzy number.

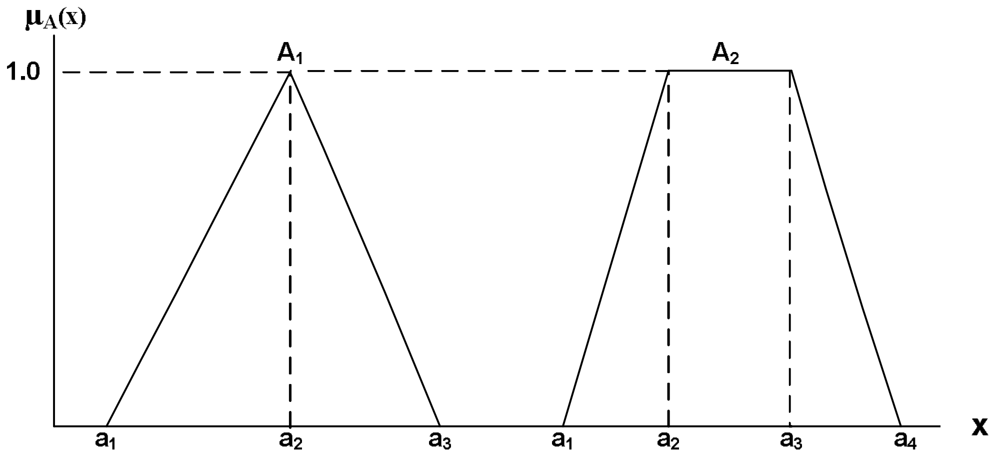


Figure 2-4: Triangular (A_1) and trapezoidal (A_2) fuzzy numbers

Definition 2.5. According to Van Laarhoven and Pedrycz [129], a triangular fuzzy number A together with its membership function $\mu_A(x)$ is defined as:

$$\mu_A(x) = \begin{cases} \frac{x-a_1}{a_2-a_1}, x \in [a_1, a_2] \\ \frac{a_3-x}{a_3-a_2}, x \in [a_2, a_3] \\ 0, \quad otherwise \end{cases} \tag{2.1}$$

Definition 2.6. With reference to Definition 2.5, a trapezoidal fuzzy number A together with its membership function $\mu_A(x)$ is defined as:

$$\mu_A(x) = \begin{cases} \frac{x-a_1}{a_2-a_1}, x \in [a_1, a_2] \\ 1, \quad x \in [a_2, a_3] \\ \frac{a_4-x}{a_4-a_3}, x \in [a_3, a_4] \\ 0, \quad otherwise \end{cases} \tag{2.2}$$

2.4.2 Case-based reasoning system

CBR is one of the popular knowledge representation and reasoning paradigms in the field of an AI. It emerged in the beginning of 1980s from the works of Roger Schank on dynamic memory that focuses on remembering past episodes as cases and scripts as situation patterns for new problem solving and learning strategies, Schank [112]. CBR has been used in a variety of problem solving and interpretive tasks; including design, planning, diagnosis, explanation, justification, classification, predicting, etc., Kolodner [72]. These several applications of CBR were reviewed in Kolodner [72], Kolodner [73], de Mántaras and Plaza [40] and de Mantaras [39]. According to Watson and Marir [136], CBR systems are attracting the attention of researchers and practitioners. The main reasons are: (a) an explicit domain model is not required for knowledge elicitation; (b) identifying the significant features of a case is easier than creating an explicit model; (c) large volumes of information can be managed using database management techniques; (d) case maintenance is easier since CBR systems can learn through acquiring new cases from previous solved cases. CBR is one of the most important methodologies to develop advisory systems in the field of IDSS, Beemer and Gregg [15]. Advisory systems are used to provide recommendations to human users in unstructured situations in which there is no a single solution available to the problem on hand. The final decision is left to the users or human experts in such advisory systems instead of replacing them in important problem solving situations.

CBR is an analogical and inductive reasoning approach, which draws inferences of a new problem depending upon experiences learned from previously solved problems, Chi and Kiang [34] and Kolodner [71]. A new problem is solved by reusing and/or adapting successful experiences to the current similar situations. It is a machine learning paradigm that enables adaptations and sustained learning since a new experience is retained when a problem is solved; and making it immediately available for future retrieval as discussed in Aamodt and Plaza [2], Kolodner [72], de Mantaras [39] and de Mántaras and Plaza [40]. According to Aamodt and Plaza [2] and de Mantaras [39], CBR problem solving approaches are different from other AI approaches in two major aspects. Firstly, they do not solely rely on general domain dependent heuristic knowledge like rule-based expert systems; it uses specific and concrete knowledge from previously experienced problem situations. Secondly, CBR systems are capable to utilise incremental learning from accumulated experiences to solve new problems, which means its effectiveness increases through time as more and more cases are retained in the case library.

Problem solving through retrieving successful experiences is a powerful and frequently applied approach in human thoughts and decision-making. Human reasoners usually prefer to reuse and/or adapt their past similar situations to the current problem instead of starting from scratch every time. Remembering previously solved problems can be difficult to human users; however, computers are best to do these tedious and complex tasks, Kolodner [71]. In this aspect, CBR systems seem more consistent with the natural reasoning process of people, Kolodner [72]. This is the reason that findings from cognitive psychology have approved the psychological plausibility of CBR systems as reviewed in Aamodt and Plaza [2], Kolodner [71] and de Mantaras [39].

As stated in Kolodner [71], in uncertain and dynamically changing environments, where much is unknown and solutions are open-ended, CBR systems are preferred over other AI techniques. This is because they can propose different solution alternatives to their users based on partially available knowledge. In addition, CBR systems regularly update the available cases in the case library and they can be efficiently trained using relatively small amount of data; however, other AI systems like ANNs are unable to do so as stated in Oh and Kim [91]. Moreover, most of the AI techniques have been intended to replace human decision-makers in important decisions, which is against to the objectives of DSS, Arnott and Pervan [12]. However, CBR systems are designed to propose various alternative decisions by reminding previously encountered similar situations. This reveals that CBR is relevant to the objectives of DSS and it can be utilised as one of the major components of DSS.

Aamodt and Plaza [2] described their general CBR cycle in terms of four ‘Re’s in Figure 2-5, which is sometimes called R^4 model.

a) Retrieve. It is searching the most similar or relevant prior case to the current problem. The similarity of each historical case in the case library to the current problem is measured using the right case retrieval methods or similarity matching functions.
b) Reuse. It is reusing the knowledge and experiences stored in the retrieved case. If the retrieved case is nearly identical to the current problem, directly reusing the retrieved case without any modifications can be the best decision alternative.
c) Revise. If some features of the retrieved cases do not match to the current cases, some revisions on the retrieved cases must be performed using the rules of revisions in order to adapt those cases to the current problems. During adaptation, some features can be added to/deleted from the retrieved cases to meet the requirements of the current cases

in consideration. The adapted solutions should be tested and verified in simulated or real-industrial environments.

d) Retain. Finally, the revised solution is retained with its corresponding problem as the learned case for future reuse/adaptations, in case similar situations are encountered in the future. Every learned case should be indexed in its case library. Indexing is used to give an identification label to the current solution with its corresponding problem for future retrieval and adaptations, Kolodner [72].

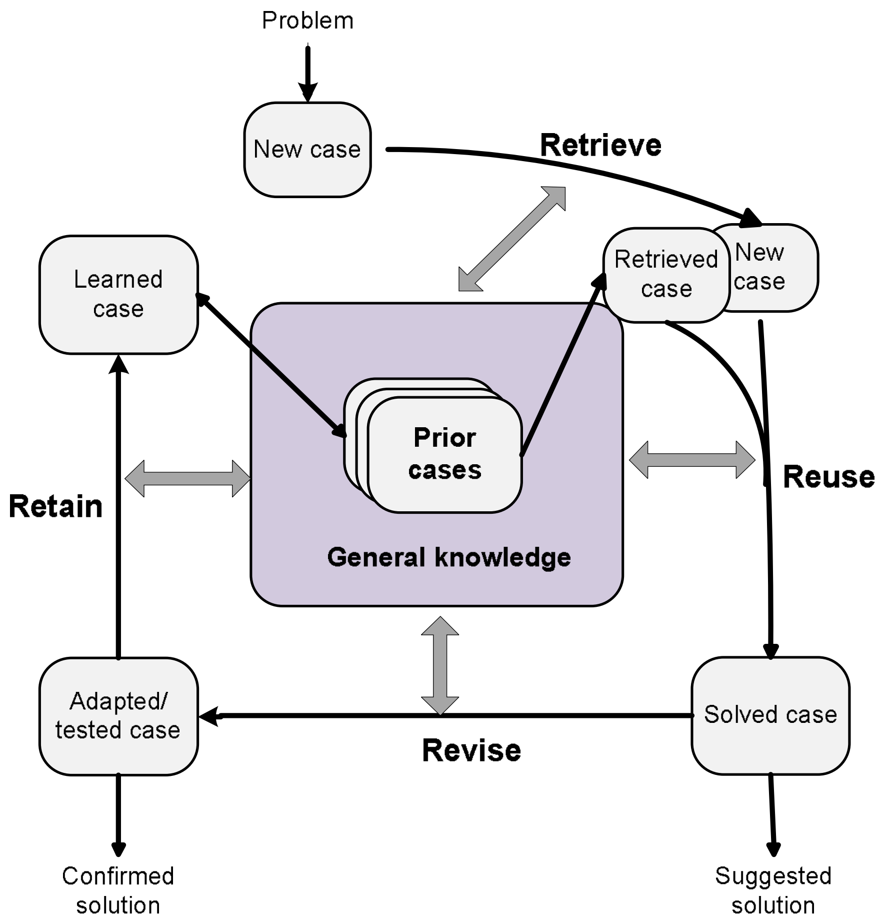


Figure 2-5: CBR cycle adopted from Aamodt and Plaza [2]

Case representation in CBR systems is useful to represent the reasoners' previous experiences contained in the form of cases for reasoning strategies in the future, Bergmann *et al*. [18]. According to Kolodner [73], a case is a contextualised piece of knowledge representing previous experiences. As stated in Watson and Marir [136] and Dubois *et al*. [42], a prior case can be represented in terms of its several features as a problem description and its corresponding solution as a solution description. Case representation is one of the substantial issues in CBR systems, which can strongly influence the effectiveness of the case retrieval and adaptation process, Chen *et al*. [31]. A case is said to be fuzzy if at least one of its features is described in terms of fuzzy linguistic terms, fuzzy numbers or fuzzy sets, Zimmermann [144]. Uncertainty and vagueness are usually inherited in CBR systems because they are mostly utilised in unstructured situations to solve open-ended problems. In order to articulate the problems of fuzziness in CBR systems, the values of case features can be suitably expressed in terms of fuzzy knowledge rather than crisp values. Incorporating fuzzy set theory into the classical CBR approaches enhances the decision-making process since it can include incomplete and imprecise knowledge stored in the form of past cases in the case base, de Mantaras [39], and de Mántaras and Plaza [40]. It was studied that fuzzy set theory increases the flexibility, Chang *et al.* [29] and broadens the applicability of CBR approaches, Li and Ho [77]. de Mántaras and Plaza [40] and de Mantaras [39] discussed the importance of integrating CBR and fuzzy set theory during the case representation, retrieval and adaptation stages of CBR systems development. In addition, de Mantaras [39] reviewed a number of successful applications of fuzzy CBR systems. Dubois *et al*. [42] proposed a fuzzy set framework for CBR and reviewed the previous works related to fuzzy CBR systems. Slonima and Schneider [116] demonstrated a general case representation and similarity-searching framework when cases are represented in terms of fuzzy attributes and originated from different classes.

Fuzzy CBR approaches have been widely researched and applied because of their flexibility and effectiveness in decision-making. Some of the latest applications are: Wu *et al*. [139] proposed a fuzzy CBR system that generates new product ideas from past product database systems. The researchers integrated fuzzy CBR and fuzzy AHP approaches in order to retrieve product ideas that tend to be highly valuable. Faez *et al*. [49] presented a fuzzy CBR model to solve vender selection problems and the AHP was applied to weight case attributes. The authors employed a mixed integer-programming model using the outputs of their fuzzy CBR model. Khanum *et al*. [70] proposed a fuzzy CBR system for recognising facial expressions. Chang *et al.* [29] presented a fuzzy CBR model to forecast sales of a printed circuit board factory. The researchers used a Fuzzy Multi-Criteria Decision-Making (FMCDM) method to select the most useful and relevance prior case to the target problem. Li and Ho [77] proposed a fuzzy CBR system that predicts financial return rates of investment projects using the combination of a fuzzy CBR and GA. For additional information regarding the integration of fuzzy set theory and DSS, interested readers are referred to Metaxiotis *et al*. [87].

2.4.3 Integrating case-based and rule-based reasoning systems

General domain knowledge represented in the form of rules in ruled-based expert systems is usually required to support the CBR process. It may range from weak to strong depending upon the problem type, Aamodt and Plaza [2]. Guiding rules may be developed and integrated into CBR systems in order to improve their performances. RBR is a deductive reasoning approach, which is derived from well-defined theories in the form of rules to infer about new problems. These symbolic rules are used to represent general domain knowledge to solve specific problems from scratch every time. The rule-based expert systems are usually unable to handle problems derived from experiences, unpredictable instances and ill-defined problems when knowledge elicitation is a bottleneck, Prentzas and Hatzilygeroudis [103]. In reality, it is very difficult to represent complex domain knowledge from experts in the form of rules. This makes that rule-based expert systems are unable to solve unstructured problem situations in which the required knowledge is incomplete and imprecise, Beemer and Gregg [15].

CBR systems do not rely on general knowledge theories; and a new problem is solved by remembering experiences of old similar situations and they are not affected by knowledge incompleteness and vagueness as stated before. However, these systems lack to utilise general knowledge and provide adequate explanations to their proposed decisions, Chi and Kiang [34] and Prentzas and Hatzilygeroudis [103]. Many industrial problems exist in the middle of these two extreme ends. Integrating RBR and CBR is regarded as a popular strategy to make systems productive using their synergic effects and avoid the limitations of CBR systems in real situations as described in Aamodt [1], de Mántaras and Plaza [40], de Mantaras [39], Prentzas and Hatzilygeroudis [103], Golding and Rosenbloom [55], Dutta and Bonissone [43], and Chi and Kiang [34]. Some of these authors proposed their own frameworks that reveal how CBR and RBR can be integrated in industrial systems, Chi and Kiang [34], Dutta and Bonissone [43], Aamodt and Plaza [2], and Golding and Rosenbloom [55]. Marling *et al.* [86], and Prentzas and Hatzilygeroudis [103] reviewed several systems that integrate rules and cases in various application domains. In addition, Golding and Rosenbloom [55], and Prentzas and Hatzilygeroudis [103] presented the taxonomy or scheme of AI techniques that combine CBR and RBR approaches.

An integrated application of CBR and RBR is usually required to determine the closeness of the retrieved and new cases. In such situations, the rules of decision-making are vaguely expressed in terms of fuzzy linguistic terms. It is common to say the similarity between the retrieved and new cases is very high, high, medium, low, etc. in practical decision-making situations. For instance, in Figure 2-3, if the similarity between the retrieved case and the new case is very high, the recommended decision may be to reuse the retrieved case. If the level of similarity between these two cases is medium or low, the best decision can be to adapt prior cases according to the requirements of the current problems. The terms “very high”, “medium”, “low”, etc. are linguistic terms, which are vague and imprecise to express the exact numerical similarity values between these two cases. This indicates that fuzziness is a natural phenomenon while rules and cases are participating in decision-making processes.

2.4.4 Object-oriented methods of case representation

As stated in Section 2.4.2, a case can be represented in terms of its several attributes and this representation affects the case retrieval process. Case representation in CBR formalises the use of familiar knowledge and experience representation methods in AI, Bergmann *et al.* [18]. The right case representation approach is required in order to meet the objectives of case reasoners. Several case representation approaches have been proposed in the past as discussed in Bergmann *et al.* [18], and Pal and Shiu [94]. An OO case representation approach is widely accepted by CBR system software developers, Watson and Marir [136]. Its popularity comes from its structured and compact-data representation capability, software reusability and easiness for users to understand, Pal and Shiu [94].

OO case representation methods are particularly useful in complex problem domains in which cases/objects with different structures occur and each object is described by a set of features, Bergmann and Stahl [19]. They provide more flexibility and modularity to the system in consideration through utilising the inheritance principles, Bergmann *et al*. [18]. They can measure case similarities to compare cases of different classes but in the same parent class using the knowledge contained in the class hierarchy, Bergmann and Stahl [19]. According to Raphael and Kumar [106], OO case representation techniques are powerful to incorporate qualitative information and knowledge in the form of messages and methods in addition to attribute-value pairs. Some applications of OO methods in the development of fuzzy CBR systems were presented in de Mantaras [39].

2.5 Multi-attribute decision-making for CBR systems

Cases in case libraries are represented in terms of their multiple attributes/features. The real case retrieval process usually uses MADM approaches. MADM is used to either select the best alternative among a finite set of alternatives or prioritise these alternative using well-defined attributes/criteria. When cases are considered in MADM analyses, prior cases are treated as alternative solutions and case features are treated as multiple attributes. The roles of MADM in CBR systems are to: (a) prioritise the weighs of case attributes; (b) find case similarities using the distance measures between the current and prior cases; and (c) select the most similar prior cases that match to the current problems on hand. According to Chang *et al*. [29], when the case retrieval process is treated using MADM approaches, best cases are selected not only on the basis of similarity of features but also on the degree of preferences over other cases. The preferences of human experts/users can be strongly conformed in MADM approaches in order to select the most relevant cases instead of blindly accepting the preferences of system developers.

The classical MADM approaches treat both the values of attributes and their weights as crisp numbers, Chen and Hwang [32]. In reality, such kinds of approaches are not convincing because the values of attributes and their weights can be expressed in terms of linguistics terms, fuzzy numbers and fuzzy sets rather than those crisp numbers. In order to address such complex situations, the current MADM approaches incorporate fuzziness associated with human decision-making strategies. Bellman and Zadeh [16] initially articulated the concepts of fuzzy set theory into Multi-Criteria Decision-Making (MCDM) problems. Baas and Kwakernaak [13] proposed the first fuzzy MADM approach that widely accepted as the classical fuzzy MADM framework in this research field. The fuzzy versions of MADM studies were reviewed and elaborated in Chen and Hwang [32], Ribeiro [107], Carlsson and Fullér [26], Kahraman *et al*. [67] and Mardani *et al*. [85].

2.5.1 Determining weights of case attributes

In MADM analyses, the determination of the weights of attributes is a crucial part for a multi-attribute value analysis, Weber and Borcherding [137]. Attributes weighting requires domain knowledge elicitation to make the case reasoning meaningful, Park and Han [95]. A key factor in the case retrieval process (similarity measure) is weighting case attributes, An *et al*. [8], and Pal and Shiu [94]. In the past, several multi-attribute weighting methods were proposed. These methods range from ordinary direct weight allocation methods to complex hierarchical approaches. Several researchers examined differences among these approaches. No specific and robust method was found to address all problem situations. It has been recommended that attribute weighting methods should be selected depending upon specific problem situations. The commonly used multi-attribute weighting methods, which are based on multi-attribute value theories can be classified into two broad categories such as the AHP and multi-attribute scoring approaches, Pöyhönen and Hämäläinen [102].

The AHP is a systematic approach to acquire and represent experts' domain knowledge for rating case features, Park and Han [95]. The AHP is an important knowledge and experience elicitation method in order to prioritise decision-making actions or criteria, Saaty [110]. The classical AHP was initially developed by Saaty in 1970, Saaty [109]. Presently, the AHP is one of the widely accepted MADM approaches with vast applications as discussed in Forman and Gass [53], Demirel *et al*. [41], Xu *et al.* [140] and Lee *et al*. [76]. Vaidya and Kumar [128] reviewed its different applications. The AHP has unique capabilities to decompose and structure any complex decision problems hierarchically; determine the relative importance of attributes or sub-attributes using pairwise comparisons; represent human judgements in terms of numerical values; measure the consistency of pairwise comparisons; and hierarchic composition or synthesis as presented in Forman and Gass [53], Wind and Saaty [138], Zahedi [141] and Saaty [111]. According to Ho [60], the popularity of the AHP is because of its easiness to use, flexibility and capability to be integrated with other approaches. The developments of the AHP applications were reviewed in Ishizaka and Labib [63] and its integrated applications with other techniques were reviewed in Ho [60]. Some recent studies revealed that the uses of integrating the AHP and CBR systems to prioritise case attributes, for example in Kuo [75], An *et al*. [8], Changchien and Lin [30], Faez *et al*. [49], Wu *et al*. [139], and Park and Han [95]. Fuzzy set theory was not directly addressed in the classical AHP, Chen and Hwang [32]. The classical AHP was extended into the fuzzy version of the AHP to address uncertainties and vagueness in real decision-making situations, Van Laarhoven and Pedrycz [129], and Buckley [25]. In addition, wide ranges of studies regarding the applications of the fuzzy AHP were reviewed in Demirel *et al*. [41]. The fuzzy version of this approach was utilised in the first publication (see Section 1.5).

Different versions of multi-attribute scoring methods were proposed. Some of the common approaches are Simple Additive Weighting (SAW), Churchman and Ackoff [37]; Simple Multi-Attribute Rating Technique (SMART), Edwards [44]; the extensions of SMART such as SMART with Swings (SMARTS), von Winterfeldt and Edwards [130] and SMART Exploiting Ranks (SMARTER), Edwards and Barron [45] or Rank Order Clustering (ROC) Barron and Barrett [14]. Among these methods, SAW is the most popular and widely used method due to its simplicity and easiness to use as illustrated in Chen and Hwang [32]. The fuzzy version of this method was initially introduced by Baas and Kwakernaak [13]. In addition, Chen and Hwang [32], and Kahraman *et al*. [67] illustrated a variety of numerical examples to compare and contrast fuzzy SAW methods proposed in past studies. The fuzzy version of a SAW method was applied in the third publication listed in Section 1.5.

2.5.2 Similarity measure and case selection

Distance from target method is one of the widely accepted MADM approaches because it is simple, easy to understand and straightforward to describe as stated in Chen and Hwang [32], and Kahraman [66]. In CBR systems, the target is the current problem and solution alternatives are prior cases. Distance-based case retrieval approaches mostly calculate the Euclidean distance between any two cases using feature-value pairs, which constitute the required cases. The most similar case is selected using this calculated distance, Liao *et al.* [80]. A prior case with the shortest distance from the target problem is the most similar case that should be retrieved for reuse or adaptations. This case retrieval approach is known as the Nearest Neighbour (NN) pattern matching function using the Euclidean distance measure. Many case retrieval approaches have been proposed in the past namely NN, inductive learning, knowledge guided and validated approaches as explained in Pal and Shiu [94]. Among these, the NN is the most common and popular pattern recognition function in *n*-dimensional Euclidean space as reviewed in Pal and Shiu [94], Park and Han [95] and Faez *et al*. [49].

When different types of attributes constitute cases, the best way to measure the distance between cases is finding the distance/similarity measures with respect to the individual case attributes and then calculating the cumulative weighted distance/similarity between two cases using the normalised weights of case attributes and the individual distance measures, Kolodner [73] and Watson [135]. Slonima and Schneider [116] presented different equations for measuring the similarities with respect to different types of case attributes such as crisp, range and fuzzy values. Faez *et al*. [49] applied three different approaches to measure similarities for crisp and fuzzy case attributes.

2.5.3 Fuzzy ranking

When fuzzy set theory is integrated with MADM methods, it improves the flexibility of the decision-making process, Chang *et al*. [29]. A number of fuzzy ranking methods were proposed to defuzzify and rank fuzzy values in MADM analyses. Most of these proposed approaches are computationally cumbersome and intractable when the number of alternatives and attributes become larger and larger.

In order to articulate this problem, Chen and Hwang [32] reviewed the pros and cons of the existing fuzzy ranking approaches. In addition, the authors proposed a new fuzzy MADM approach to reduce the computational difficulties of the reviewed approaches. In their new approach, the following three steps are included. (a) any linguistic terms should be projected into their equivalent trapezoidal/triangular fuzzy numbers, which are scaled into any real numbers within the range of [0, 1]; (b) these fuzzy numbers should be converted into their estimated crisp values using the right fuzzy ranking approaches; and (c) an appropriate MADM approach must be applied depending upon the problem type.

This approach avoids some computational difficulties by converting any fuzzy data into crisp values before any MADM operations are undertaken. Although, its inputs are either fuzzy data or a combination of fuzzy and crisp data, its outputs are usually crisp numbers in the range of [0, 1]. Any complex problems with a combination of fuzzy data and crisp data can be easily accommodated with the help of this approach. Their proposed approach favours the right and left scoring technique using maximising and minimising sets. However, several fuzzy scoring techniques have been proposed in different problem domains. For example, recently Chen and Chen [33] highlighted the limitations of previously proposed methods and proposed a new ranking method to address those limitations. Brunelli and Mezei [24] conducted comparative studies on existing fuzzy ranking methods.

2.6 DES in DSS development

Simulation is one of the most widely accepted and utilised interactive modelling techniques. Specifically, DES models have been immensely used in manufacturing and business because of their increased computational power, cost reduction and successful applications as decision support tools, AlDurgham and Barghash [3]. According to Smith [117], DES involves the imitation of descriptive computer models of complex systems and exercising those models in order to predict the operational performances of the underlying systems being modelled. As stated in Jahangirian *et al*. [64], DES is well-recognised in decision-making because of its relevance in real industrial applications in order to accommodate the complexities of the whole enterprise without any productivity-paradoxes. Its interactive capabilities are attractive features for its recognition and supremacy over other modelling techniques. DES models are easier ways to build up models for representing real system scenarios so as to identify bottlenecks, enhance system performances in terms of productivity, queues, resources utilisation, cycle times, etc. as stated in Ali and Seifoddini [4] and Rahimifard and Newman [104].

DES applications are classified into two broad categories such as system design and analysis simulation and system operational simulation, Smith [117] and Andersson and Olsson [9]. In the context of system design and analysis, DES is used in a conventional way in order to analyse, evaluate, test and validate newly designed complex systems prior to their final implementation. It is usually applied in long-term decision-making tasks like facility layout design, manufacturing system design, etc., used for a single design exercise and a runtime of the model is not its major concern during simulation times. System operational simulation is applicable for short-term planning, scheduling and the control of manufacturing systems such as shop floor control, short time scheduling, capacity planning, etc. In this context, a runtime of the model is a very significant factor, Smith [117].

Wide ranges of applications of DES were reviewed in several past publications using various research approaches as reviewed in Jahangirian *et al.* [64], Chan and Chan [27], Smith [117] and Shafer and Smunt [113]. For additional information, the latest research works pertinent to the applications of DES-based DSS in manufacturing were reviewed in the second publication of Section 1.5.

2.7 Combining AI and DES in DSS

A solution proposed by AI methodologies should be usually validated with the help of appropriate modelling techniques to reveal the soundness of the proposed solution. An integration of AI techniques and DES models is essential to develop intelligent simulation models for planning and control of production systems, Rahimifard and Newman [104]. In dynamic and stochastic manufacturing environments, designing a simulation only decision support tool is time-consuming and unrealistic to find optimal solutions, Pehrsson *et al.* [96]. Finding an optimal solution might not be practical because system flexibility is required to accommodate the frequent changes of user needs; and integrating simulation models with AI methods is required to accommodate these changes, Chan and Chan [27]. According to Rogers and Gorden [108], a purely simulation-based approach is time-consuming and it results in significant time delays due to human user interventions for selecting the required candidate actions and interpreting their results.

Hybrid approaches, which can combine AI engines with DES models, are essential to articulate such kinds of problems in the current dynamically changing environments. These approaches make the current DSS more intelligent and flexible in order to emulate human expertise. AI technologies are required in simulation systems to access simulated results, operate as human expertise and explain the consequences of decisions. Similarly, DES models are required in Al research, where automatic planning systems are derived from a set of AI techniques representing recommended solutions to be tested for their feasibility, completeness, efficiency, etc., as discussed in O'Keefe and Roach [90]. Angehrn and Lüthi [10] added that the main goal of DSS is not only to provide information concerning specific problem-solving techniques but also to provide decision makers with tools for interactively exploring, designing, and analysing decision situations; and act as human consultants in order to help decision-makers in understanding, expressing and structuring their problems in dynamic situations. Due to the limitations of purely AI-based and purely DES-based DSS, several researchers presented decision support tools that integrated AI techniques and DES models. The following research works, which are based on the combination of AI and DES technologies, are reviewed in manufacturing.

Benz and Mertens [17] embedded knowledge-based systems into simulation models to enhance the statistical knowledge of the system in consideration. In this research, SIMULEX software was presented as a prototype that combined expert systems and DES models in order to propose a DSS for short-term rescheduling in manufacturing systems. Iassinovski *et al.* [62] presented a unified DSS framework for the purpose of model sharing, reusability and integration of intelligent simulation and optimisation techniques to articulate the problems of dynamic systems. Conteh and Forgionne [38] proposed an Intelligent Just-In-Time Decision-Making Support System (IJDSS), which utilised simulation models to test the efficacy of the IJDSS relative to traditional DSS concepts.

Mahdavi *et al.* [84] developed an interactive simulation-based DSS using an adaptive controller for integrating a real-time simulator and a rule-based DSS for the production control of stochastic flexible job shop manufacturing systems. Chan *et al.* [28] used a simulation approach assisted by a knowledge-based system in the design of flexible manufacturing systems. The AHP was applied to analyse the outputs from FMS simulation models and intelligent tools such as expert systems, fuzzy systems and artificial neural networks were employed for supporting the FMS design process. Ali and Seifoddini [4] suggested a simulation-based intelligent system in order to accommodate uncertainties and risks in high-mix and low-volume manufacturing systems. An intelligent simulation model was designed to represent factory floor dynamics for labour and machine dynamics, and fuzzy rule-based systems were developed for uncertainty representations. Feng *et al.* [51] proposed a simulation-based DSS that integrates manufacturing systems, multi-agent systems, OO techniques and simulation methods in order to form a unified system for evaluating different alternatives in manufacturing systems using simulation scenarios.

Völkner and Werners [131] presented an OO simulation-based DSS known as business-process simulation system, which is specifically applicable to evaluate different alternatives in business process planning and workflow sequencing actions in uncertain situations. According to Völkner and Werners [132], the researchers improved the previous system by incorporating fuzzy set theory and knowledge-based procedures in the previous version to address fuzziness and linguistic uncertainties. Zülch and Becker [145] used an integrated approach of DES and heuristics approaches to develop an optimised man-machine configuration to plan personnel and technical resources. Liraviasl *et al.* [81] presented a DSS framework, which supports the decisions of reconfiguration of manufacturing systems using hybridised agent-based simulation and DES techniques.

2.8 Synthesis of theoretical framework

In the previous sections of this chapter, the fundamental issues of DSS were addressed in order to identify the potential areas of original contribution to the current body of knowledge in DSS. It was found that past studies on the subject matter of DSS integrating AI and DES methodologies were one of the most attractive research topics. These combined approaches were applied in several operations such as shop floor control, parts scheduling and sequencing, FMS design, machine maintenance planning, etc. However, a major problem in such kinds of DSS is that they were unable to articulate other crucial dimensions of manufacturing systems such as integrated fixture management problems due the reasons stated in the introductory chapter.

A research architecture presented in Figure 2-6 was proposed to address this research problem space, which was identified as the current research gap in DSS study. The framework was synthesised using the current knowledge reviewed from various sources of literature presented in the previous sections to address the problem statement in this study. The assumptions and the main components of the proposed DSS framework are presented in the next two subsections.

2.8.1 Assumptions

In order to articulate the proposed research problem, the following important assumptions were considered, which were useful to define the boundaries of the research problem.

- The manufacturing environment was dynamic and deterministic within a specific short production period. Assuming *m* parts were scheduled, the proposed DSS should determine *n* stable number fixtures required to manufacture these *m* products.
- Similar part orders required the same fixture for the required operations.
- The current factory layout was optimal enough to process these *m* products.
- The best part scheduling and sequencing procedure was determined to manufacture *m* parts using unknown number of *n* fixtures.
- During simulation experiments, the effects of tools, machine breakdown and shortage of other resources were not considered to focus on the effects of fixtures alone.
- The attribute values of part orders were static at specific machining operations.
- The costs of reuse, adaptation and manufacture decision sets were suitably estimated using other expert systems or human experts. Cost estimation was beyond the scopes of this study.

2.8.2 Components of theoretical framework

The proposed theoretical framework incorporated four major elements to comply with the requirements of the current DSS literature such as AI or Fuzzy CBR (FCBR), database, model/DES and user interface components (Figure 2-6). Communication technologies were excluded to make the framework simple for understanding and they could be easily incorporated depending upon the nature of the firm under consideration as stated in Section 2.2.4. This study mainly focused on the combination of fuzzy CBR and DES subsystems, which are specifics of knowledge-based (KB) or AI systems and models respectively.

The database component incorporated all the required data that could be used as input variables to run both the fuzzy CBR and DES subsystems. In the case of this study, the database included all resources required at shop floor level to run manufacturing centres (operators, available machines, fixtures, buffers, materials handling equipment and storage); historical data; operational performance targets; planning and scheduling information; new and prior product order descriptions to construct cases; fixture descriptions; and weights of case attributes.

The AI component was the one that was in charge to process product order descriptions, and propose decisions and solutions. This subsystem was intended to utilise immensely a CBR methodology. Product orders were regarded as new cases to represent them using an OO method. This method was selected because of its flexibility to construct cases using a combination of various types of case attributes as problem descriptions. The types of these case attributes could include crisp numerical values, intervals, fuzzy terms, symbolic values, descriptive terms, etc. Historical case in the case library should be expressed in the same way except they incorporated assigned fixtures as solution descriptions. The part descriptions and historical data from the database were substantial to create these cases. In order to weight these case attributes and retrieve the most relevant prior cases in the case base, the right fuzzy MADM methods such as the fuzzy AHP and weighted NN pattern matching functions were proposed respectively. When a new case (problem) arrived at the system, the inference engine was considered to search the most similar previous cases in the case library to the new cases using the proposed fuzzy MADM approaches. To improve the CBR process, the integration of RBR from general domain knowledge and the FCBR subsystem was proposed with reference to the current literature regarding their integration effects (Section 2.4.3).

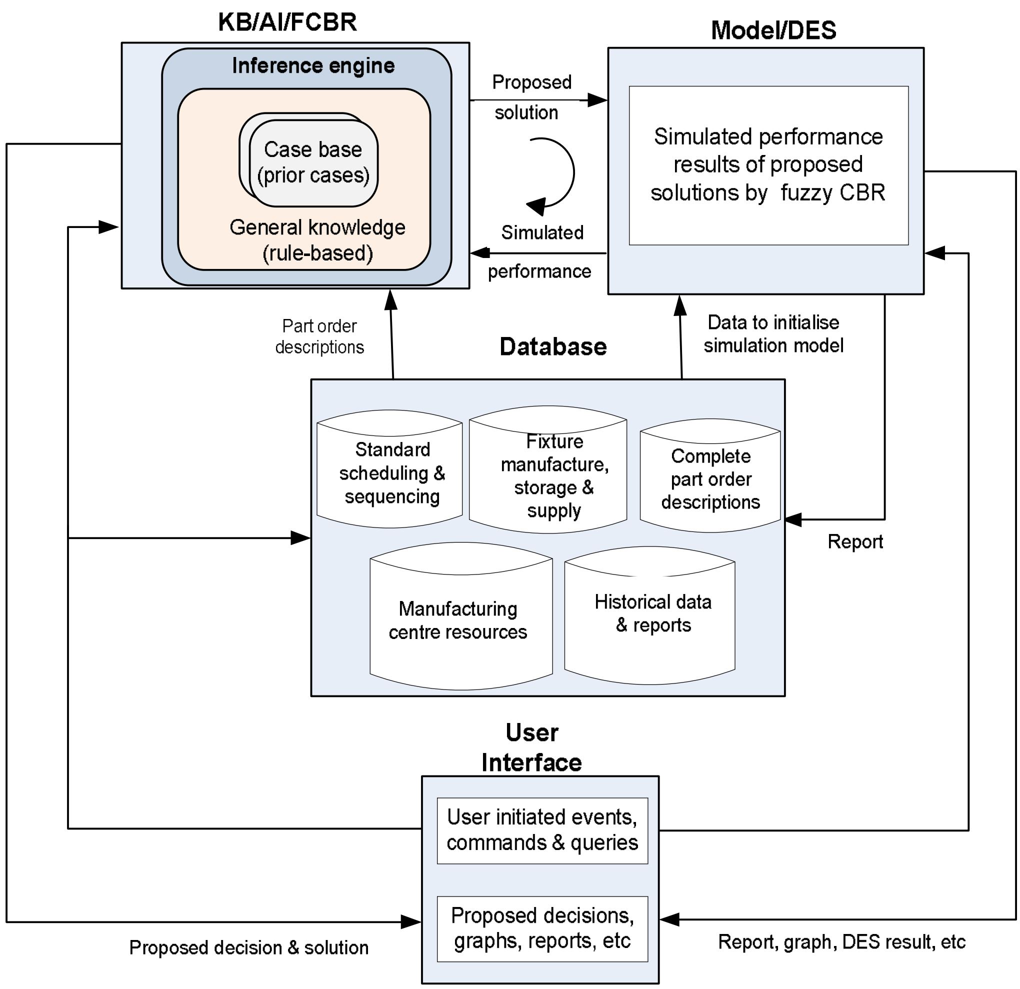


Figure 2-6: Theoretical framework of the research problem

According to this study, a solution proposed by the fuzzy CBR systems should be validated with the help of an appropriate modelling technique in order to examine its usefulness and soundness. DES models were considered in this regard with reference to the current literature in DSS presented in Section 2.7. A solution means the proposed stable number of fixtures required in a specific production period as defined in the introductory chapter. The DES subsystem should receive the recommended stable numbers of fixtures as the proposed solutions from the fuzzy CBR subsystem. It should also receive substantial information from the database for initialising the DES model(s). According the proposed framework, if various solutions are recommended by the AI component, the DES must generate unique performance scenarios for every proposed solution. In an automated manufacturing environment, if the fuzzy CBR subsystem proposes the initial solution, the DES should evaluate its performances using Key Performance Indicators (KPIs); and the CBR in turn can access the simulated performances and propose other better solutions if the target performances are not met by the system. This improvement cycle should continue until the intended operational performances are attained. Finally, the best solution can be selected based on the results of the DES among the recommended solution alternatives.

The user interface subsystem was proposed to enter input data and queries into the system. These inputs can be the values of case attributes in various forms, weights of case attributes, setup times, product processing times, etc. The outputs of the DSS can also be presented such as proposed decision sets, DES performance results, etc.

2.9 Summary

This chapter reviewed the fundamental theories of DSS in line with the statement of the research problem. The theories regarding the evolutions, definitions and fundamental components of DSS were discussed. With reference to the frontiers of DSS, the objectives of the current DSS were determined. The important dimensions of manufacturing systems, which have not been sufficiently articulated in the current DSS such as integrated fixture and tool management strategies, were identified as the current research gaps in DSS.

A CBR methodology was proposed as the main constituent in the AI part of DSS because it was reviewed that the objectives of CBR systems are pertinent to the objectives of DSS. According to the current literature, both DSS and CBR are designed in order to advise or support human experts by proposing alternative solutions in complex and unstructured situations unlike other KB systems, which are intended to replace human experts in decision-making. In addition, CBR systems learn from their successful experiences through time in order to solve effectively and efficiently new problems. In this aspect, CBR is consistent with the natural reasoning process of people. Knowledge adaptation and updating is inherited in CBR systems. CBR systems are highly flexible to accommodate uncertainties and changes in dynamic situations because they can be easily integrated with other KB systems such as fuzzy set theory, RBR, fuzzy MADM and OO case representation methods. The significance of combining CBR and these systems was reviewed in line with the DSS requirements.

The roles of DES in DSS development in the context of system design and analysis, and operational simulation were reviewed. The importance of integrating AI and DES methodologies in decision-making in the context of manufacturing was discussed. Recent research and developments in DSS, which integrate AI and DES techniques, were reviewed. It was noted that these combined approaches in DSS research have not been utilised in fixture planning and management problems. With reference to this research gap in the current DSS studies, a theoretical framework for this study was proposed. This framework was regarded as a significant addition to the existing knowledge in DSS to articulate the research problem in this study.

CHAPTER 3

3. DEVELOPMENT OF DECISION SUPPORT SYSTEM

3.1 Introduction

This chapter is devoted to elaborate the methodological approaches reviewed in the previous chapter. It explains the steps and methods required for developing the researched DSS based on the theoretical framework presented in Section 2.8. It mainly focuses on the combination of fuzzy CBR and DES methodologies in order to solve the stated research problem, which has not been exploited in past studies. This section introduces the interactions among the methodological approaches (Figure 3-1). The second section describes the case construction process incorporating case attribute identification and case representation approaches. In the third section, the evaluation strategy to weight case attributes using the fuzzy AHP and the fuzzy ranking methods applied in this study are explained. The fourth section discusses the major roles of the fuzzy CBR subsystem such as case retrieval, decision proposal and case retaining. The interaction between DES and fuzzy CBR methodologies for validating the proposed solutions from the fuzzy CBR subsystem is discussed in the fifth section.

In this research, the AI component of the DSS immensely utilised a fuzzy CBR methodology. In addition, different rules were developed in order to simplify the case representation and retrieval, and decision-making processes. Cases were represented using an OO approach in the Java programming language. The proposed DSS used a fuzzy MADM approach to weight case attributes and search the most similar prior cases to the current problems to perform a decision-based part/fixture assignment. The fuzzy CBR component proposed the stable number of fixtures required to process product orders planned within specified production periods. A DES model was proposed to evaluate the performances of the proposed solutions. The interactions among the methodological approaches in this study are presented in Figure 3-1. The details are discussed in the next subsequent sections.

As stated in the introductory chapter, the focus of this research was an on-demand fixture retrieval, reuse, adaptation and manufacture according to the requirements of part orders from particular manufacturing processes. From this result, this study was able to determine the stable number of active fixtures required in a specified production period in manufacturing operations. In order to address the proposed problem, the assumptions presented in Section 2.8 were fully applied in this chapter. The proposed DSS advised the users to reuse/adapt or manufacture a new fixture after a case retrieval operation based on the state of the retrieved fixture and the similarity measure between the current and retrieved cases.

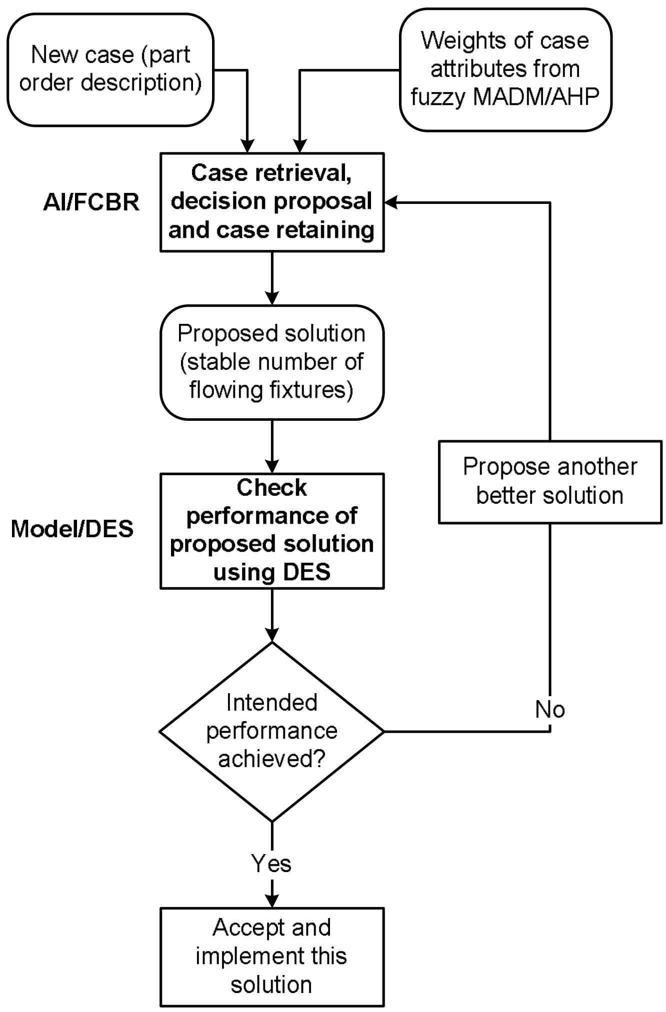


Figure 3-1: A flow diagram of the interaction among the methodological approaches

3.2 Case construction

In this study, product orders were treated as fuzzy cases. The crucial attributes of product orders were identified to construct both prior and new cases. These attributes were used to determine the case similarities between the current and prior cases for decision-based fixture assignment tasks. This implemented the assumption that similar part orders demanded the same fixture during their crucial operations. Because some of product attributes were suitably expressed in terms of fuzzy linguistic terms rather than sharp crisp numbers, the product orders in this research were treated as fuzzy cases.

3.2.1 Case attribute selection and structuring

As the current part orders/cases arrived at the researched system shown in Figure 3-1, the descriptions of their attributes (problem descriptions) were not well organised in terms of their crucial attributes. In addition, these case attributes were not weighted according to their significance. Their attributes can be structured using continuous or discrete numerical values, nominal/categorical values, range values, descriptive/symbolic terms, linguistic terms, etc. It was required to reorganise the current product order arrivals based on their key attributes to improve the productivity of the proposed DSS. In this study, these key attributes were expressed with the combinations of numerical values, nominal values, descriptive/symbolic terms and linguistic terms. These important feature-based descriptions were used to make product orders suitable for case representations, prioritising the weights of case attributes and searching the case similarity values between the current part orders and prior cases under consideration.

The first step to structure the problem descriptions of order arrivals was identifying the key attributes that were required for MADM analyses. This study recommended human experts to select a few critical case attributes, which were substantial to find the similarities among the required product orders for part/fixture assignment strategies. Experts were assumed to use either one or more of attribute rating techniques reviewed in Chapter 2 or their experiences.

3.2.2 Case representation

With reference to Section 2.4.2, cases were described as either new problems or training samples. According to this study, training samples were prior cases together with their corresponding solution descriptions (assigned fixtures). They were a few prior product orders, which were represented and structured, using their identified key attributes as problem descriptions and their assigned fixtures as solution descriptions. Such prior cases can be found from previously solved problems or created by experienced experts when prior cases are not initially available. Specially, when the concerned manufacturing system is newly established, it is very difficult to find previously solved problems. Usually, a CBR system starts with a few training samples and the system updates regularly the number of cases in its case library as new cases enter into the system. This improves the effectiveness of CBR systems through time unlike other AI technologies. This approach was applied to this study.

New problems were the current product orders as new cases, incorporating their problem descriptions alone. Their solutions descriptions can be retrieved and reused/adapted from similar past cases in the case library or newly manufactured fixtures based on the case similarity measures and the state of the retrieved fixtures. This approach was stated as a decision-based part/fixture assignment in Chapter 1. Referring to Figure 3-1, the new problem was described as a product order arrival including its problem descriptions in terms of numerical values, nominal values, symbolic terms and fuzzy linguistics terms.

The problem description included the physical features of workpieces, process requirements and types of operations required at particular workstations. These attributes were used to represent the required cases in a 12-dimensional Euclidean vector space. In case representation stages, linguistic terms were converted into their equivalent fuzzy numbers with the help of the proposed conversion scales indicated in Figure 3-6. These conversion scales are explained in detail in Section 3.4. A case representation scheme for the current product orders and prior cases in the researched and developed DSS is depicted in Figure 3-2. Prior case representations included an additional resource named an assigned fixture as a solution description. The remaining components were used as problem descriptions, which were common to both new and prior cases.

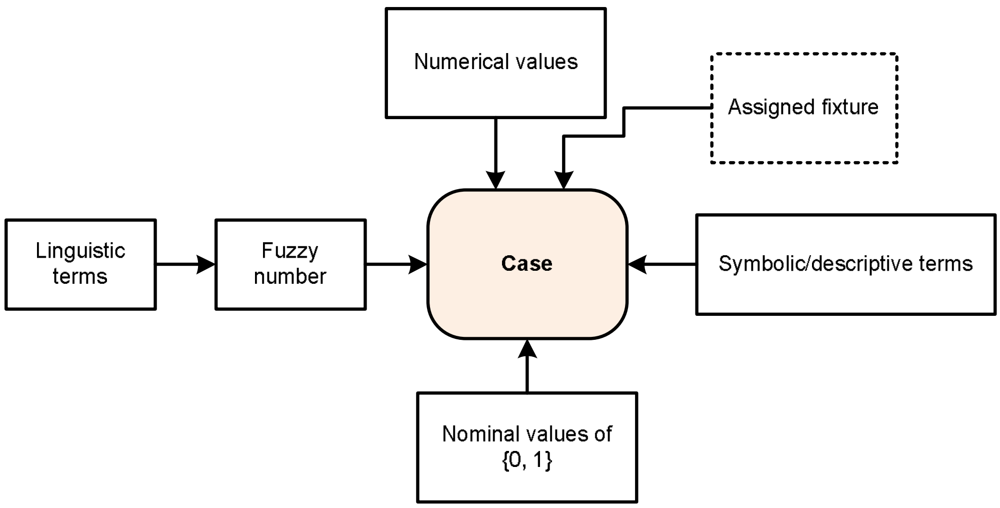


Figure 3-2: Fuzzy case representation scheme of product orders

The proposed DSS used an OO case representation approach in order to create the current and prior cases, because of its advantages reviewed in the previous chapter, using the Java programming language. This programming language was employed for this research project because it is relatively enriched with many in-built library classes and methods, simple and clean for implementation in such complex situations. Some special rules were incorporated into the CBR subsystem to convert symbolic attributes into nominal values of {0, 1} using a few Java in-built methods. The case representation in this DSS was highly comprehensive and flexible because it could incorporate different types of case attributes (Figure 3-2). In addition, if some products were unpredictably ordered and all their numerical attributes found within the maximum and minimum values of the existing representation matrix, these unpredicted orders could be processed without any revisions on the current representation matrix. It was implied that certain orders could be added to/removed from the matrix without affecting the existing matrix values in Table 3-1.

Table 3-1: Case representation matrix

	w_1	w_2	. . .	w_n
	$\boldsymbol{A}_1$	$\boldsymbol{A}_2$	. . .	$\boldsymbol{A}_n$
$\boldsymbol{P}_1$	a_{11}	a_{12}	. . .	a_{1n}
$\boldsymbol{P}_2$	a_{21}	a_{22}	. . .	a_{2n}
.	.	.	. . .	.
.	.	.	. . .	.
.	.	.	. . .	.
$\boldsymbol{P}_m$	a_{m1}	a_{m2}	. . .	a_{mn}

Where:

m is the total number of products planned during a given production period.

n is the total number of case attributes in n-dimensional Euclidean space vector.

$\boldsymbol{P}_1 \ldots \boldsymbol{P}_m$ are m finite product orders planned in a given order arrival sequence.

$\boldsymbol{A}_1 \ldots \boldsymbol{A}_n$ are n finite case attributes to characterise all part orders.

a_{ij} is an attribute value of a product order, $\boldsymbol{P}_i(i=1\ldots m)$ against an attribute $\boldsymbol{A}_j$ ($j= 1\ldots n$) in terms of numerical values, nominal values, symbolic terms or fuzzy terms/numbers.

$w_1 \ldots w_m$ are weights assigned to case attributes.

3.3 Evaluating fuzzy weights of case attributes

After identifying the key product attributes, it was required to prioritise the identified case attributes. This was because not every attribute could be expected to have the same contribution to the case similarity searching process. The major steps for ranking these attribute are presented in Figure 3-3. The details are presented in the next two subsections. This section deals with weighting case attributes using the fuzzy AHP approach and defuzzification of fuzzy numbers with the help of the steps presented in Figure 3-3.

3.3.1 Weighting case attributes

The weights of case attributes can be rated in terms of either crisp numbers or linguistic terms such as “unimportant”, “moderately important”, “important”, etc. The first option can be applied when the uncertainty and vagueness associated with human reasoning is negligible. In practical situations, fuzziness is a natural phenomenon in human decision-making actions, which cannot be neglected as reviewed in Chapter 2. In this research, the second option was preferred to articulate the uncertainty and imprecision of knowledge in production systems. With reference to the previous chapter, it was stated that case retrieval functions usually use MADM approaches and multi-attributes weighting plays significant roles in the MADM processes.

Evaluating the weights of attributes usually requires domain knowledge elicitation to make the case reasoning and decision-making processes more meaningful. In this aspect, the fuzzy AHP is popular and well recognised. This approach was preferred in this study depending upon the nature of the research problem. The AHP was utilised as a supportive expert system to determine the weight of case attributes. The AHP can be usually implemented using either special software packages like Expert Choice or general application software such MS Excel tools. In this research, the second alternative was used because these tools are usually simple and efficient to undertake simple matrix operations and they are easy to integrate with several Java applications.

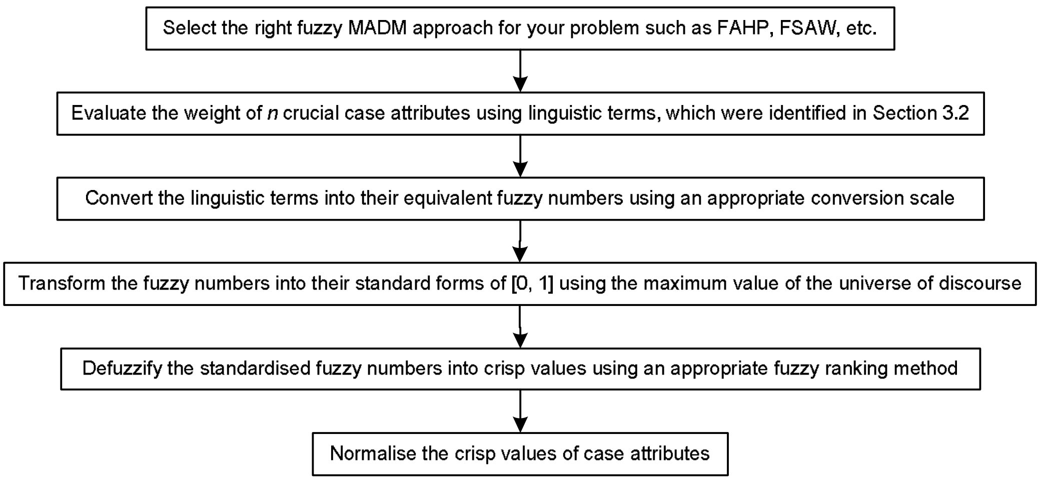


Figure 3-3: Steps for evaluating case attributes

With the help of the fuzzy AHP, the preference of one case attribute over the other was evaluated in terms of linguistic terms like “equally preferred”, “moderately preferred”, "strongly preferred", etc., using pairwise comparisons. These importance ratings were purely subjective and vague to define their boundaries due to human judgments. Table 3-2 presents the relationships among the fuzzy AHP-based linguistic terms, the equivalent fuzzy numbers and the reciprocals of the fuzzy numbers. Similar conversion approaches were applied in other research studies such as Chioua *et al.* [35], Lee *et al.* [76] and Wu *et al.* [139] in different problem domains. The conversion of these linguistic terms into their equivalent trapezoidal/triangular fuzzy numbers is indicated in Figure 3-4, where x is any real number in the range of (0, 10] and $\mu(x)$ is the degree of membership of x to the linguistic terms within the interval [0, 1].

Table 3-2: AHP-based linguistic terms, their equivalent fuzzy numbers and reciprocals of fuzzy numbers

Intensity of preference (Linguistic terms)	Fuzzy number	Reciprocal fuzzy number
Exactly equal	(1, 1, 1)	(1, 1, 1)
Equally preferred	(1, 1, 2)	(1/2, 1, 1)
Intermediate	(1, 2, 3)	(1/3, 1/2, 1)
Moderately preferred	(2, 3, 4)	(1/4, 1/3, 1/2)
Intermediate	(3, 4, 5)	(1/5, 1/4, 1/3)
Strongly preferred	(4, 5, 6)	(1/6, 1/5, 1/4)
Intermediate	(5, 6, 7)	(1/7, 1/6, 1/5)
Very strongly preferred	(6, 7, 8)	(1/8, 1/7, 1/6)
Intermediate	(7, 8, 9)	(1/9, 1/8, 1/7)
Extremely preferred	(8, 9, 10)	(1/10, 1/9, 1/8)

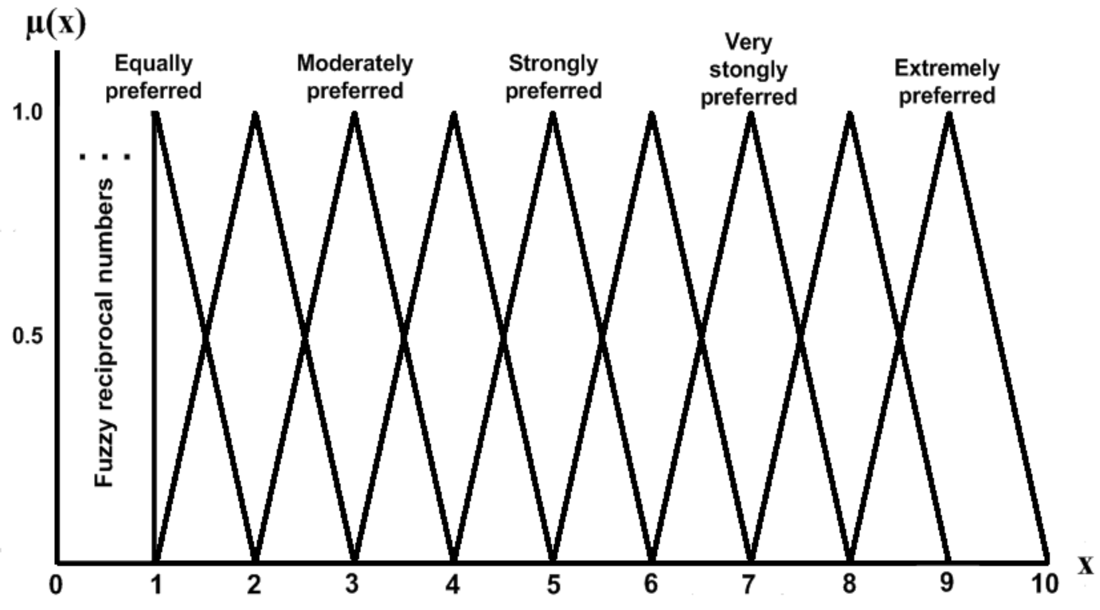


Figure 3-4: Conversion of the AHP-based linguistic terms into fuzzy numbers

3.3.2 Ranking fuzzy numbers

Several fuzzy ranking methods have been proposed in the past to compare and rank fuzzy numbers. This study merged two fuzzy ranking approaches that were proposed by Chen and Hwang [32] and Chen and Chen [33] because of their simplicity, comprehensiveness and flexibility. According to Chen and Hwang [32], any linguistic terms described as input variables should be projected into their corresponding trapezoidal/triangular fuzzy numbers, which are scaled into any real number within [0, 1] using an appropriate scaling method. Then these fuzzy numbers must be transformed into their equivalent crisp numbers with the help of the right fuzzy ranking approaches. This approach avoids computational difficulties in MADM analyses by converting any fuzzy data into crisp values before any MADM operations are undertaken. Although, its inputs are either fuzzy data or a combination of fuzzy and crisp data, its outputs are usually crisp numbers in the range of [0, 1] as mentioned in the previous chapter. Any complex problems with the combination of fuzzy and crisp data can be easily addressed using this approach.

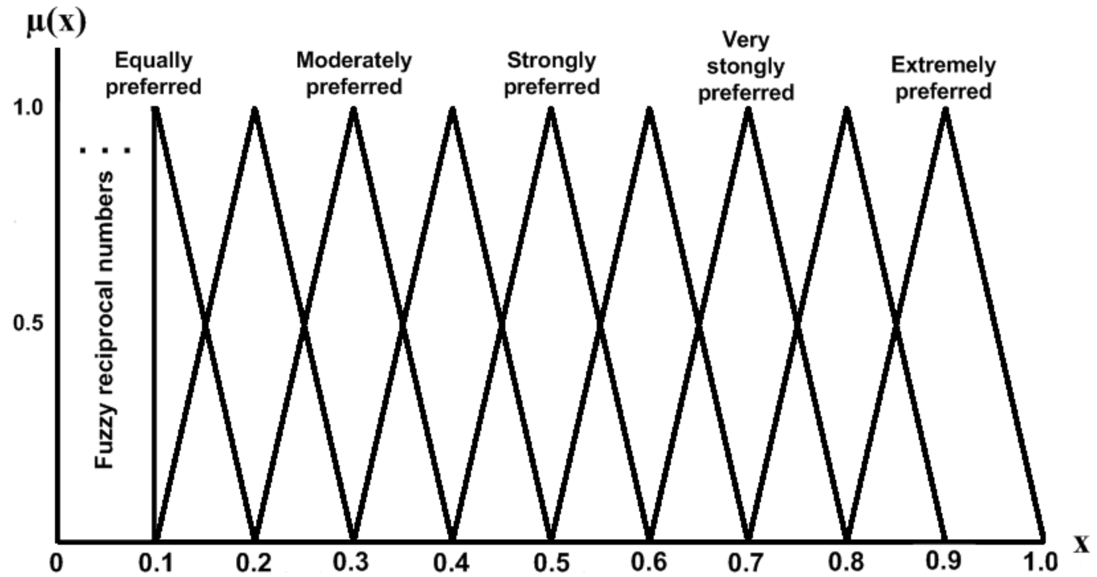


Figure 3-5: Conversion of the AHP-based linguistic terms into standard fuzzy numbers

Chen and Chen [33] argued that any generalised trapezoidal fuzzy numbers could be converted into standard fuzzy numbers in the range of [-1, 1] by dividing them with the magnitude of the maximum value of the universe of discourse. In Figure 3-5, the fuzzy numbers and their reciprocals indicated in Figure 3-4 were scaled into the range of $0 < x \leq 1$ by implementing this approach. It should be noted that the range $0 < x \leq 1$ does not include the number 0. It was excluded in the AHP rating approach because its reciprocal is infinite.

The two stated approaches were integrated in the fuzzy AHP approach in order to defuzzify the linguistic terms that were used to express the weight of case attributes. Referring to Chen and Chen [33], Table 3-3 presents the relationships among the fuzzy AHP-based linguistic terms, their equivalent fuzzy numbers and their standard forms in the range of $0 < x \leq 1$. This conversion was carried out by dividing all the fuzzy numbers and fuzzy reciprocals in Table 3-2 with the maximum value of the universe of discourse, which is the number 10 in this case.

Table 3-3: Linguistic terms, their equivalent fuzzy numbers and standard fuzzy numbers

AHP-based fuzzy linguistic terms	Equivalent		Standard	
	Fuzzy number	Reciprocal	Fuzzy number	Fuzzy reciprocal
Exactly equal	(1, 1, 1)	(1, 1, 1)	(0.1, 0.1, 0.1)	(1/10, 1/10, 1/10)
Equally preferred	(1, 1, 2)	(1/2, 1, 1)	(0.1, 0.1, 0.2)	(1/20, 1/10, 1/10)
Intermediate	(1, 2, 3)	(1/3, 1/2, 1)	(0.1, 0.2, 0.3)	(1/30, 1/20, 1/10)
Moderately preferred	(2, 3, 4)	(1/4, 1/3, 1/2)	(0.2, 0.3, 0.4)	(1/40, 1/30, 1/20)
Intermediate	(3, 4, 5)	(1/5, 1/4, 1/3)	(0.3, 0.4, 0.5)	(1/50, 1/40, 1/30)
Strongly preferred	(4, 5, 6)	(1/6, 1/5, 1/4)	(0.4, 0.5, 0.6)	(1/60, 1/50, 1/40)
Intermediate	(5, 6, 7)	(1/7, 1/6, 1/5)	(0.5, 0.6, 0.7)	(1/70, 1/60, 1/50)
Very strongly preferred	(6, 7, 8)	(1/8, 1/7, 1/6)	(0.6, 0.7, 0.8)	(1/80, 1/70, 1/60)
Intermediate	(7, 8, 9)	(1/9, 1/8, 1/7)	(0.7, 0.8, 0.9)	(1/90, 1/80, 1/70)
Extremely preferred	(8, 9, 10)	(1/10, 1/9, 1/8)	(0.8, 0.9, 1.0)	(1/100, 1/90, 1/80)

The required standard fuzzy numbers were transformed into their estimated crisp values using a fuzzy ranking approach proposed by Chen and Chen [33]. Equation (3.1) was applied to defuzzify the required fuzzy numbers. This approach is simple; it avoids the limitations of other methods; and prefers the most precise fuzzy numbers when different fuzzy numbers have an identical mean value. After determining the crisp score of any trapezoidal fuzzy number, A_{cs}, the classical AHP approach was applied to calculate the normalised weights of case attributes.

$$A_{cs} = \frac{A_{mean}}{1+A_{std}} \tag{3.1}$$

Where A_{mean} and A_{std} are the mean and standard deviation values of a standard fuzzy number respectively.

3.4 Similarity measure for decision analysis

A case representation and weighting the importance of case attributes were two critical preceding tasks in order to measure the case similarities between any new and prior cases. According to this research, fuzzy case attributes, which were described in terms of linguistic terms, were converted into their equivalent fuzzy numbers using eleven conversion scales indicated in Figure 3-6. This framework was proposed by adopting the conversion ideas proposed in Chen and Hwang [32]. Any numbers of conversion scales can be applied based on the precisions required to solve specific problems in consideration. In this figure, the variable x is any real number in the range of [0, 1] and $\mu(x)$ is the degree of membership of x to the linguistic terms within the interval [0, 1]. Eleven verbal terms were proposed to describe triangular fuzzy numbers in the figure; however, the framework was flexible enough to create several trapezoidal fuzzy numbers by merging any two or more neighbouring triangular fuzzy numbers. For example, a trapezoidal fuzzy number (0.6, 0.7, 0.8, 0.9) was created by merging the term “Fairly high” (0.6, 0.7, 0.8) and the term “High” (0.7, 0.8, 0.9). This idea is elaborated in the next chapter using a numerical example.

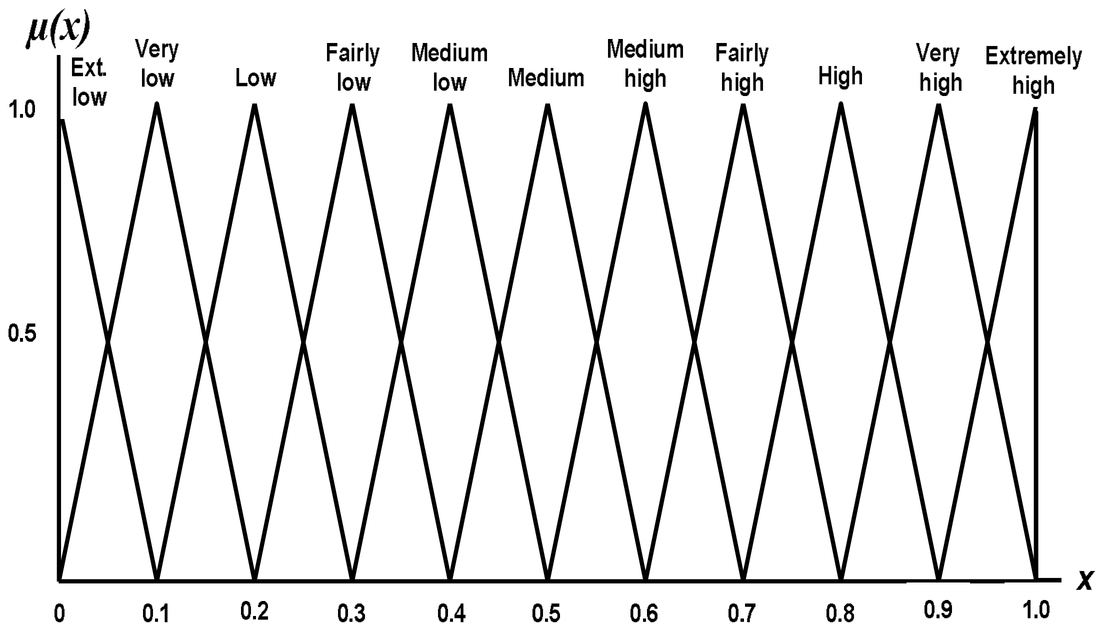


Figure 3-6: Conversion of linguistic case features into fuzzy numbers

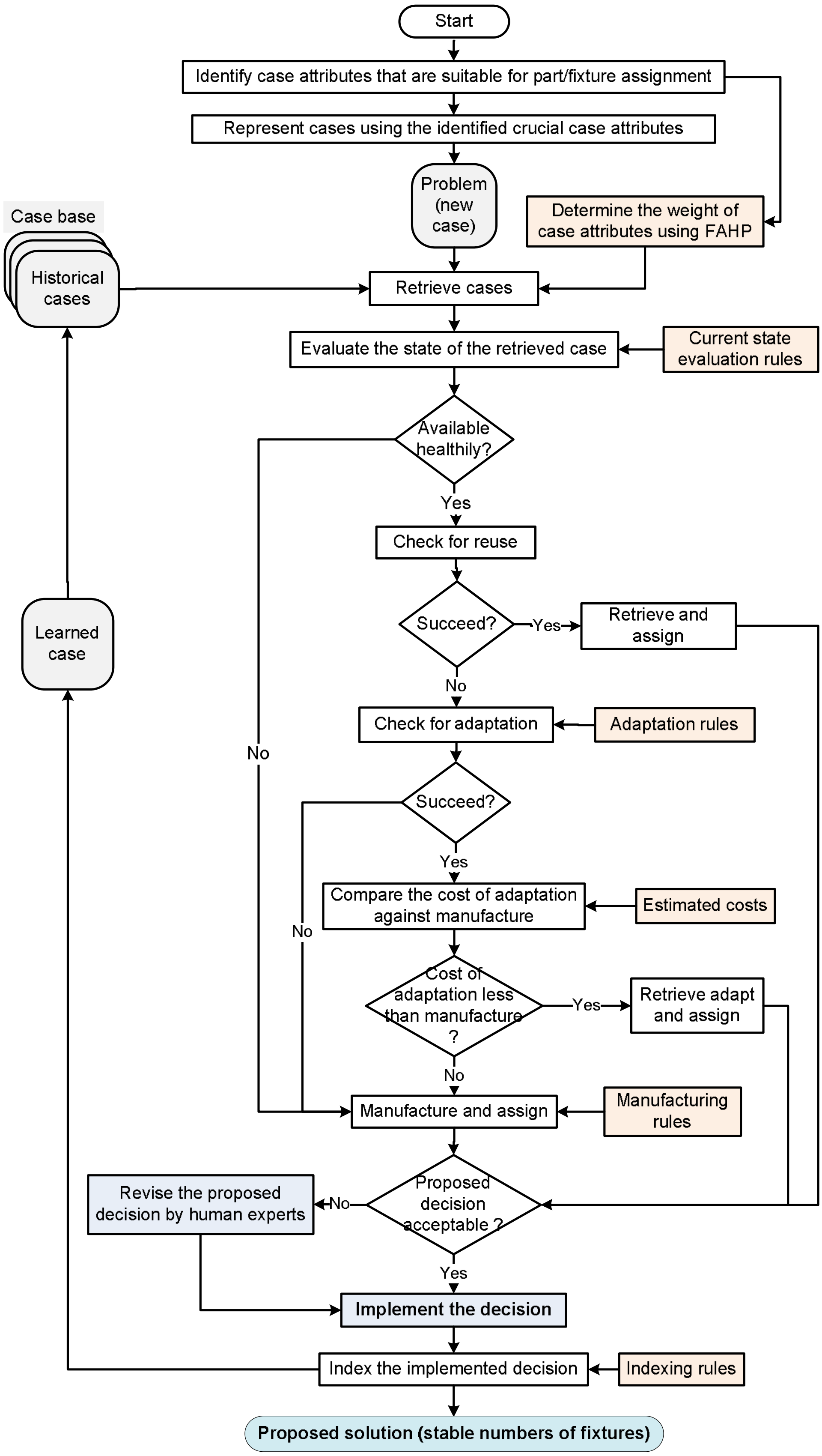


Figure 3-7: Decision logic for on-demand case retrieval, decision proposal, case retaining and solution proposal

This section covers the main tasks of the fuzzy CBR subsystem of the researched DSS such as case retrieval, decision proposal and case retaining as presented in Figure 3-7 with reference to the generalised methodological contexts depicted in Figure 3-1. For the sake of convenience, the major activities discussed in the previous section are included in the upper part of the decision logic.

3.4.1 Case retrieval

With reference to Figure 3-7, the case retrieval process utilised the descriptions of the current problem, descriptions prior cases in the case library and normalised weights of case attributes as its input variables. One of the challenges in CBR subsystems is retrieving the most similar and relevant prior cases that match to the current problems, Faez *et al*. [49]. Case retrieval is one of the most cumbersome tasks to human experts/users in the decision-making process and the reverse is true in computerised environments, Kolodner [71].

A number of case retrieval methods have been proposed to search the similarity between the current and past cases, and identify the most similar prior cases. This study used one of the most popular approaches, which is called the NN pattern matching function in a high dimensional vector space using the inverse of the weighted Euclidean distance. The Euclidean distance measures the distance between objects based on the location of objects in the Euclidean space as stated in Pal and Shiu [94] and Liao *et al*. [80].

This Euclidean distance approach was thought as one of distance from target methods in MADM as reviewed in Section 2.5.2 since it uses the current case as the target and prior cases as alternative solutions. Using the inverse of this weighted Euclidean distance, the similarity between the current case and each prior case in the case library was determined. The weighted Euclidean distance between a new product order p and a prior product order q, $dist(p, q)$ in n-dimensional Euclidean vector space was calculated as follows:

$$dist(p, q) = \sqrt{\sum_{i=1}^{n}[w_i * dist(a_i^p, a_i^q)]^2}, \; dist(a_i^p, a_i^q) \in [0, 1] \tag{3.2}$$

Where:

n is the number of case attributes.

w_i is the normalised weight of the ith case attribute.

$dist(a_i^p, a_i^q)$ is the distance measure between case p and case q with respect to the ith case attribute alone.

a_i^p and a_i^q are the values of the *i*th attribute for cases *p* and *q* respectively.

In this study, $dist(a_i^p, a_i^q)$, the distance between the current and prior cases with respect to every individual *i*th attribute was measured first and the weighted Euclidean distance was calculated using the normalised weights of case attributes and these individual distance measures as indicated in Equation (3.2). The $dist(a_i^p, a_i^q)$ measures were determined depending upon the nature of the individual case attributes.

In the case of numerical attributes:

$$dist(a_i^p, a_i^q) = \frac{|a_i^p - a_i^q|}{a_{i,max} - a_{i,min}}, \ a_i^p \ \& \ a_i^q \in [a_{i,min}, a_{i,max}] \tag{3.3}$$

Where $a_{i,min}$ and $a_{i,max}$ are the minimum and maximum value of the *i*th attribute respectively. They were used to normalise the calculated distance into [0, 1] in order to avoid the effects of measurement unit and scale changes.

For nominal/descriptive attributes:

$$dist(a_i^p, a_i^q) = |a_i^p - a_i^q| = \begin{cases} 1 \ if \ a_i^p \neq a_i^q \\ 0 \ if \ a_i^p = a_i^q \end{cases} \tag{3.4}$$

In the case of fuzzy attributes, trapezoidal fuzzy numbers were considered and Equation (3.5) was adopted from a method of similarity measure of generalised fuzzy numbers, which has been recently proposed by Hejazi *et al.* [59]. Their proposed method combined the concepts of geometric distance, the perimeter, the area and the height of trapezoidal fuzzy numbers. In this case, the value of the height was 1.0 since all fuzzy numbers used in this study were normal and convex (they have equal heights = 1). This method was applied to accommodate situations when the required fuzzy numbers had different sizes and shapes in order to incorporate the effects of their perimeters and areas.

When trapezoidal/triangular fuzzy numbers are in standard forms as stated in Section 3.2.2, $a_i^p = (a_{i,1}^p, a_{i,2}^p, a_{i,3}^p, a_{i,4}^p)$ and $a_i^q = (a_{i,1}^q, a_{i,2}^q, a_{i,3}^q, a_{i,4}^q)$; and $0 \leq a_{i,1}^p \leq a_{i,2}^p \leq a_{i,3}^p \leq a_{i,4}^p \leq 1$ and $0 \leq a_{i,1}^q \leq a_{i,2}^q \leq a_{i,3}^q \leq a_{i,4}^q \leq 1$.

$$dist(a_i^p, a_i^q) = 1 - \left[\left(1 - \sum_{k=1}^{4} \frac{|a_{i,k}^p - a_{i,k}^q|}{4} \right) * \frac{min\left(P(a_i^p), P(a_i^q)\right)}{max\left(P(a_i^p), P(a_i^q)\right)} * \frac{min\left(A(a_i^p), A(a_i^q)\right) + 1}{max\left(A(a_i^p), A(a_i^q)\right) + 1} \right] \tag{3.5}$$

Where:

$P(a_i^p)$ and $P(a_i^q)$ are the perimeters of trapezoidal fuzzy attributes of case p and case q respectively.

$A(a_i^p)$ and $A(a_i^q)$ are the areas of trapezoidal fuzzy attributes of case p and case q respectively.

$$P(a_i^p) = \sqrt{(a_{i,2}^p - a_{i,1}^p)^2 + 1} + \sqrt{(a_{i,4}^p - a_{i,3}^p)^2 + 1} + (a_{i,3}^p - a_{i,2}^p) + (a_{i,4}^p - a_{i,1}^p) \quad (3.6)$$

$$P(a_i^q) = \sqrt{(a_{i,2}^q - a_{i,1}^q)^2 + 1} + \sqrt{(a_{i,4}^q - a_{i,3}^q)^2 + 1} + (a_{i,3}^q - a_{i,2}^q) + (a_{i,4}^q - a_{i,1}^q) \quad (3.7)$$

$$A(a_i^p) = \frac{1}{2}(a_{i,3}^p - a_{i,2}^p + a_{i,4}^p - a_{i,1}^p) \quad (3.8)$$

$$A(a_i^q) = \frac{1}{2}(a_{i,3}^q - a_{i,2}^q + a_{i,4}^q - a_{i,1}^q) \quad (3.9)$$

With reference to Equation (3.2), the calculated values of $dist(a_i^p, a_i^q)$ are always in the range of [0, 1]. The maximum Euclidean distance between any two cases, $dist_{max}(p,q)$, is found when all the values of $dist(a_i^p, a_i^q)$ = 1; and the minimum Euclidean distance between any two cases, $dist_{min}(p,q)$, is found when all the values of $dist(a_i^p, a_i^q)$ = 0 i.e. when p and q are identical items (p = q). Then, the $dist_{max}(p,q)$ and $dist_{min}(p,q)$ values can be simplified and determined by referring to Equation (3.2) as follows:

$$dist_{max}(p,q) = \sqrt{\sum_{i=1}^{n} w_i^2} \quad (3.10)$$

$$dist_{min}(p,q) = 0 \quad (3.11)$$

Because distance and similarity are inversely related, the similarity between two cases p and q, $sim(p,q)$, can be found as follows, Liao *et al.* [80]:

$$sim(p,q) = 1 - dist(p,q) \quad (3.12)$$

The minimum similarity between any two cases, $sim_{min}(q,p)$, was calculated from Equations (3.10) and (3.12) as follows:

$$sim_{min}(p,q) = 1 - dist_{max}(p,q) = 1 - \sqrt{\sum_{i=1}^{n} w_i^2} \quad (3.13)$$

Similarly, the maximum similarity between any two cases, $sim_{max}(p,q)$ was computed from Equations (3.11) and (3.12) as follows:

$$sim_{max}(p,q) = 1 - dist_{min}(p,q) = 1 - 0 = 1 \tag{3.14}$$

Then, $sim(p,q) \in [sim_{min}(p,q)$, 1.0].

All the above equations were coded in the Java programming language and incorporated in the proposed DSS. Using these equations, the DSS generated a list of similarity measures between the current case and prior cases while a new product order was entering into the system. The DSS selected the maximum similarity measure on the similarity list using the Java library method "max(list)", which returns the maximum value from a list, in the "java.util.Collections" class. Depending upon this returned value, any retrieved case q with a higher similarity value to the current problem p, was selected for future retrieval and reuse and/or adaptations.

3.4.2 Decision proposal

Once the most similar prior case to the current problem was retrieved, the next important challenge was recommending a set of decisions to the users based on the current state of the retrieved case and the similarity measure between the current and prior cases. As an intelligent DSS was concerned, it must have advised its users to assess the current state of the retrieved fixture. This was because the retrieved device could be physically damaged or even lost during the retrieval time. In order to address this problem in manufacturing situations, the proposed DSS advised fixture planners to evaluate whether the device was available in a functional state using their evaluation rules and/or opinions. The system recommended the state of the retrieved device should be expressed in terms of fuzzy linguistic terms rather than crisp values. For example, its usefulness can be rated using verbal terms such as "very high", "high", "medium", "low", etc. These terms were converted into their equivalent fuzzy numbers using the conversion scale in Figure 3-6. Next these fuzzy numbers were transformed into their estimated crisp values using Equation (3.1). Finally, a threshold value was proposed to accept or reject the retrieved fixture based on its current state.

Several (If…, Then….) rules were developed and applied in order to support the decision-making process in this study. For example, in the case of the state of the retrieved fixture, the following decision rules were proposed.

- If the retrieved fixture is in a failed state, the proposed DSS recommends a removal of the retrieved case from the case library for permanent revisions/discards and proposes manufacture of a new fixture to replace the removed case.
- If the retrieved device is available in a functional state, the DSS advises the decision makers to reuse/adapt the retrieved case or manufacture of a new fixture depending upon the similarity between the current and retrieved prior cases.

Three important rules were proposed to support decision makers when the retrieved device was in a functional state. Suppose part order p is the target (current) case, which is arriving at the system and q is the retrieved prior case from the case library as stated above; the guiding rules were described as follows:

- If the value of $sim(p,q)$ is close to one i.e. $sim_{max}(p,q)$, then the recommended decision is to reuse directly the retrieved case/fixture.
- If the value of $sim(p,q)$ is medium, then the recommended decision is to adapt the retrieved case to the current problem.
- If the value of $sim(p,q)$ is close to $sim_{min}(p,q)$, then the preferred decision is to manufacture a new fixture.

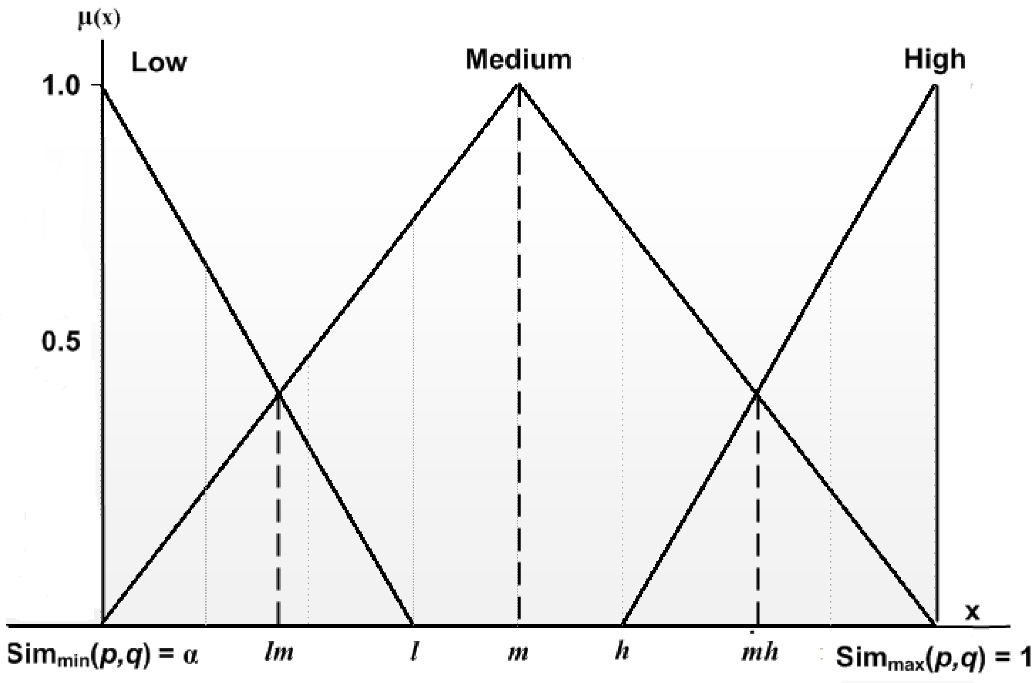


Figure 3-8: Relationship between similarity values and linguistic terms

The proposed rules to implement these decisions were imprecise and vague. In other words, the similarity measures between two cases were expressed using fuzzy verbal terms such as "close to one", "medium" and "close to $sim_{min}(p,q)$". These vague similarity indicators can be expressed with the help of three imprecise linguistic terms namely "high", "medium" and "low" respectively. The relationships between such linguistic terms and similarity measures are presented in Figure 3-8 using the concepts presented in Chen and Hwang [32].

The maximum and the minimum similarity measures were used as the upper and lower bounds of the similarity measure between any two cases, respectively. The variable $x = sim(p,q)$, which is in the range of $[sim_{min}(p,q)$, 1.0] from Equation (3.12) and $\mu(x)$ is the degree of membership of the variable x to the linguistic terms "Low", "Medium" and "High" similarity values, which is in the range of [0, 1]. Finding the intersection of the left-leg of the term "High" and the right-leg of the term "Medium", the threshold similarity measure *mh* to terminate reusing the retrieved cases and start an adaptation of the retrieved cases was determined. In the same way, using the intersection of the left-leg of the term "Medium" and the right-leg of the term "Low", the threshold similarity level *lm* that to terminate adaptations and start manufacture of a new fixture was found. These thresholds were determined by solving simple linear equations indicated next.

Using the left-leg of the term "Medium" and the right-leg of the term "Low" respectively, the threshold *lm* was determined as follows:

$$\mu_{lm}(x) = \begin{cases} \frac{x-\alpha}{m-\alpha}, & x \in [\alpha, m] \\ \frac{l-x}{l-\alpha}, & x \in [\alpha, l] \end{cases} \tag{3.15}$$

Then equating the two sub-equations from Equation (3.15):

$$lm = \frac{m*l-\alpha^2}{m+l-2\alpha} \tag{3.16}$$

Similarly, using the right-leg of the term "Medium" and the left-leg of the term "High" respectively, the threshold *mh* was found.

$$\mu_{mh}(x) = \begin{cases} \frac{1-x}{1-m}, & x \in [m, 1] \\ \frac{x-h}{1-h}, & x \in [h, 1] \end{cases} \tag{3.17}$$

Then equating the two sub-equations from Equation (3.17):

$$mh = \frac{1-h*m}{2-(h+m)} \tag{3.18}$$

Using the thresholds values, the stated fuzzy decision rules were transformed into three ranges of numerical values in order to defuzzify the above imprecise and vague terms in decision rules. They were revised as follows:

- If $mh < sim(p,q) \leq 1$ is fulfilled, then reusing the retrieved fixture is the recommended solution.
- If $lm < sim(p,q) \leq mh$ is fulfilled, then an adaptation of the retrieved case to the current product order is recommended.
- If $sim_{min}(p,q) \leq sim(p,q) \leq lm$ is fulfilled, then manufacture of a new fixture to the current problem is proposed.
- If $sim(p,q) > 1$ or $sim(p,q) < sim_{min}(p,q)$, then the input is invalid.

In addition, the proposed DSS assessed the cost effectiveness of a decision when an adaptation decision was passed. The feasibility and cost effectiveness of the recommended decisions was taken into account in the proposed DSS. Sometimes, the recommended decisions using the calculated similarity measures cannot be efficient and cost effective. As indicated in Figure 3-7, if a decision of an adaptation is passed, the adaptation cost must be compared with the cost of manufacture of a new fixture. Specially, when the required operations are performed on a single machine and parts are subsequently arriving at the machining centre, with demanding the same fixture for adaptations, the machine downtime cost can be significant to reverse previously implemented adaptation decisions. In other words, the cost of fixture adaptations may be higher than the cost of manufacture of a new fixture. In order to articulate this problem, additional rules were recommended as follows:

- If the cost of an adaptation decision is less than the cost of manufacture of a new fixture, then the proposed decision using similarity measures should be accepted and implemented.
- If the cost of an adaptation decision is higher than the cost of manufacture of a new fixture, then the proposed decision using similarity measures should be revised and manufacture of a new fixture is recommended.

The cost of adaptation decisions usually includes machine downtime, process overhead, material and setup costs. Moreover, the cost of manufacture decisions incorporates design, process overhead, material, setup and storage costs. In this study, the DSS designed to read these costs as its input data; however, estimating these costs is beyond scopes of this work. These costs can be estimated by either human experts or other expert systems.

When the decision to reuse or adapt was passed, the DSS checked the availability of the required fixture in its fixture database. If it was available, then the fixture should be retrieved from the database and assigned to the current part order arrival. Otherwise, it should be in the process and the part order should wait the requested fixture from the concerned process. These rules can be especially useful, when two or more process centres are sharing the same fixture.

Finally, as presented in the bottom part of Figure 3-7, the researched DSS provided opportunities to human experts to evaluate the proposed decisions by the system. If the recommended decision was acceptable, it should be directly implemented; otherwise, it should be referred to human experts for correction prior to its final implementation. The importance of intervening human experts in such kinds of situations was studied in Tan *et al*. [123].

3.4.3 Case retaining

Case retaining was one of the substantial tasks in the proposed CBR subsystem. It was useful to retrieve previously implemented decisions for future reuse and adaptations. In this study, two types of case libraries were created and implemented with the help of the "java.util.ArrayList" class in the Java programming language.

a) The first case library retained training samples and new cases that required the use of newly manufactured fixtures. When a new fixture was required, that new case incorporating its assigned new fixture served as a new training sample for future retrieval and usage. This case library was used to determine the total number of active fixtures that were flowing in the system. In other words, the number of active fixtures in the system was identical to the number of cases in the first case library after every part order was processed at specific operation centres.
b) The second case library retained new cases that reused or adapted the retrieved cases. When these new cases reused or adapted the retrieved cases, no need of adding them into the training samples because those retrieved cases were working as members of training samples. This case library was required to propose what activities should be done in the case adaptation process. When the users of the researched DSS encountered

new cases that required similar adaptation tasks to any previous cases, they used the same adaptation procedures to the problem on hand.

In both case libraries, the implemented cases/decisions were indexed using “add (object)” function that is one of the in-built methods of the Java “java.util.ArrayList” class. This method appends a new element at the end of a list. In order to implement this case retaining process, the following indexing rules were proposed.

- If the decision of manufacture a new fixture is passed, add the new case into the first case library.
- If the decision of reuse or adapt is passed, add the new case into the second case library.

3.5 Validating proposed CBR system using DES

The common approach to validate the accuracies of newly designed CBR systems is testing them with the help of historical data. These historical data are unable to predict the near future performances of the proposed systems. They indicate past business situations, which are not much significant to the present and future situations. An intelligent system usually learns from the past and predicts the near future business situations instead of being driven by the current events. As reviewed in the previous chapter, CBR systems are best to learn from the past and DES-based systems are excellent to predict the near future situations through analysing various scenarios or performing “what-if” analysis. With reference to this fact, DES was utilised to validate and predict the performances of the solutions proposed by the fuzzy CBR subsystem in this research.

As shown in Figure 3-1, the DES component of the DSS was intended to receive the proposed stable number of fixtures required within a planned production period as a solution, from the AI (fuzzy CBR) subsystem of the proposed DSS. When this solution was proposed by the fuzzy CBR subsystem, the users of the system were uncertain whether the proposed solution performed according to the intended performances of the system under investigation or the right stable number of fixtures was determined in the system. In order to justify this complex problem situation, DES was done based on process requirements of the planned part orders (e.g. process time, setup time, number of batches, batch size, etc.). If the required performance is achieved, the proposed solution should be accepted and implemented. Otherwise, the fuzzy CBR subsystem should propose another improved solution for the DES subsystem. This operation continues until the target performance is met with the help of the DES results. Various solution alternatives can be generated by changing the number of case attributes in case construction, weights allocated to case attributes, threshold values in Figure 3-8 and combinations of these factors in the AI subsystem of the DSS.

In DES modelling, FlexSim simulation software package (www.flexsim.com/), which is one of the popular, versatile and 3D DES packages in the world, was utilised in this study. Since the proposed numbers of active fixtures flowing in the system were directly affected by implementing a set of decisions (reuse/adapt the retrieved fixture or manufacture a new fixture), these sets of decisions were regarded as discrete events. Other parameters such as setup time and operational costs of fixtures, which were dependent upon these decision sets, were treated as the random variables in a DES model.

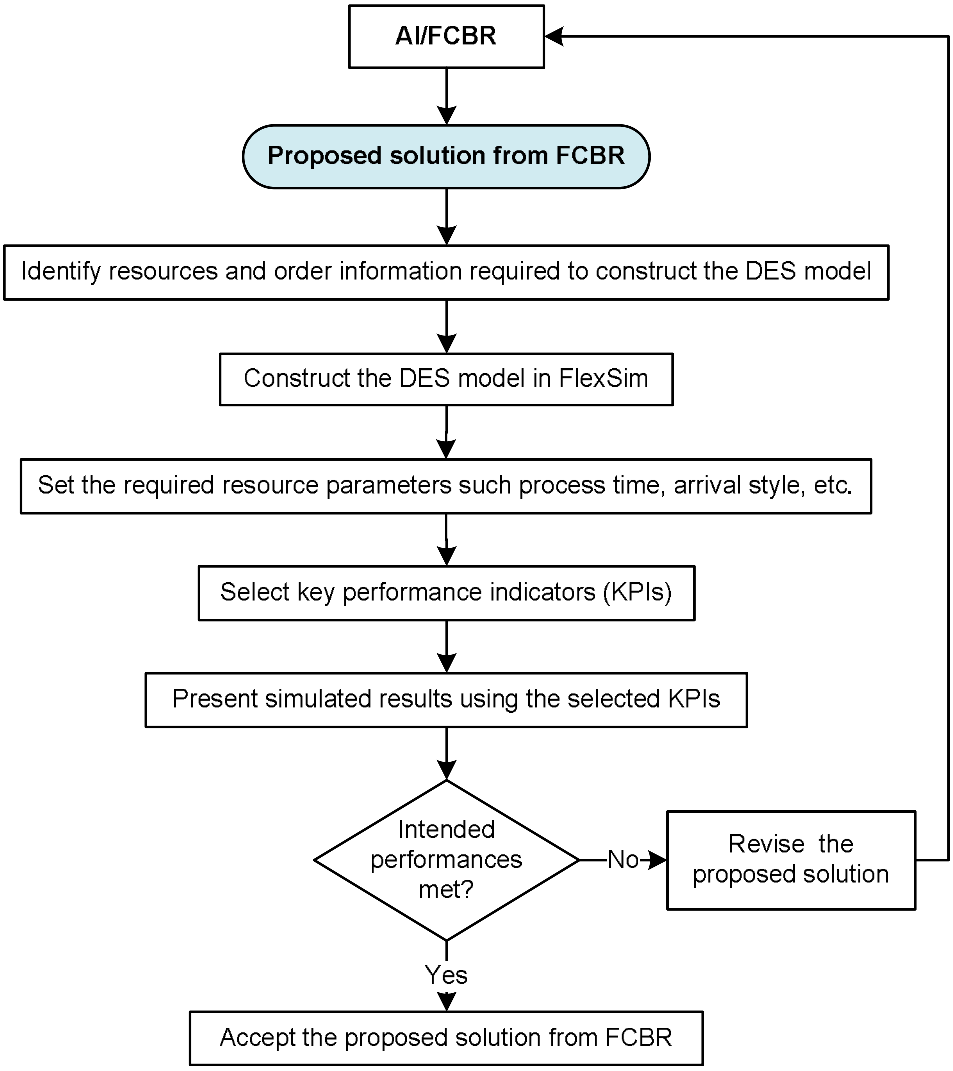


Figure 3-9: Steps in DES modelling

The major activities that were involved in the proposed DES model are presented in Figure 3-9, including the interaction of the DES model with the fuzzy CBR subsystem. The first step in this modelling was identifying the necessary resources required in a DES model construction such sources, queues, processors, operators, transporters, and sinks with reference to the solution received from the fuzzy CBR element of the DSS. In parallel, product order information such as the number of batches, batch size, setup time and process time per unit order, order arrival style, etc. were required to run the proposed DES model. These process requirements were assigned to their relevant resources. For example, batch type, batch size and order arrival style were fed to sources, and process times and setup times were required for processors. Next KPIs such as machine utilisation, manufacturing lead-time, throughput rate, operational costs, etc. were identified to perform “what-if” analysis for several scenarios. The simulated performances of alternative solutions should be presented in terms of these KPIs using appropriate simulation-based charts and graphs. If the simulated results are accordingly the intended performances, the proposed solution by the fuzzy CBR subsystem can be implemented; otherwise, the solution must be revised by changing the parameters in the fuzzy CBR component as stated before.

3.6 Summary

This chapter presented the methods required to conduct this research and how the researched DSS was developed to solve the research problem using the literature and theories reviewed in the previous chapter. The methodological approach synthesised in this chapter combined the existing complex fields such CBR, fuzzy set theory, RBR, OO method, MADM/AHP and NN algorithm and DES to address the problems in fixture assignment and control. The importance of integrating these elements in this problem domain was explained in the dedicated subsections.

The methodology principally focused on the combination of fuzzy CBR and DES techniques, which has not been well addressed in previous research studies. The AI subsystem of the DSS was constructed mainly using a fuzzy CBR methodology to propose the stable number of fixtures in processes. RBR approach was assimilated into the proposed CBR system to improve the case reasoning process. In addition, an OO method was integrated with the fuzzy CBR subsystem to make the case representation more flexible and modular. Fuzzy CBR and fuzzy AHP methods were combined in the case/fixture retrieval process to form the weighted NN function. The fuzzy CBR subsystem was designed to articulate the imprecise values of case attributes in the DSS development process. The fuzzy

AHP was used to elicit experts' domain knowledge for prioritising the weights of case attributes. A fuzzy ranking method was devised to defuzzify linguistic terms into their estimated crisp values to reduce computational difficulties in the AHP.

In order to determine the similarity between new and prior cases for fixture assignment, the NN pattern matching function, specifically the inverse of the weighted Euclidean distance was selected. Different equations were implemented to measure the similarities with respect to the individual types of case attributes. Finally, a DES subsystem was included in the researched DSS in order to analyse the performances of the proposed solutions by the fuzzy CBR subsystem to minimise the risk of the proposed solution due to the lack of knowledge and experience in case construction and weighting case attributes.

In general, this chapter was designed to synthesise a new methodological approach from the current theories and literature in order to articulate the stated research problem in this study.

CHAPTER 4

4. COMPUTATIONAL ANALYSIS

4.1 Introduction

In this chapter, the steps and methodological approaches that were explained in the previous chapter are illustrated using a numerical example. The numerical example is elucidated by taking into consideration relevant machining operations. It is illustrated using milling centre in computerised laboratory environments to reveal the applicability of the researched DSS. Product orders are represented as fuzzy cases in terms of twelve product attributes using an OO case representation approach. Among these case attributes, two of them are described in terms of fuzzy linguistic terms to accommodate the uncertainty and imprecision of knowledge in manufacturing situations. The weights of these twelve case attributes are determined with the help of the fuzzy AHP.

In order to determine the similarities between new and prior cases, the equations presented in the previous chapter are utilised. In addition, the necessary in-built Java library classes and methods are applied to make effective the case retrieval, decision proposal and case retaining processes. For the sake of illustration, sixteen part orders are instanced as new order arrivals and three prior cases are initially treated as training samples. The two case libraries stated in Section 3.4.3 are employed accordingly their intended tasks.

Using the fuzzy CBR subsystem of the proposed DSS, three alternative solutions are proposed by varying the values of the parameters presented in Figure 3-8. The performances of these three scenarios are analysed using a DES model, which is used to model the proposed ideal machining centre.

4.2 Machining centre and case attributes

Fixture selection and assignment problems for specific product orders highly depend upon the types of operations performed, the physical features of the workpieces in consideration and the capability of the process to manufacture a product with the demanded level of quality. These factors vary with reference to the operation centres under investigation. For example, taking into consideration the basic machining operation centres (milling and turning), the operations performed, the required geometric features of workpieces and the outputs from specified operations vary at milling and lathe machining centres. These factors were the bases to identify the crucial product attributes of the part/fixture assignment problems in this study. The performances of specified machining centres can be highly influenced by the types of fixtures selected and assigned. In other words, product attributes required in part/fixture assignment problems vary based on the selected operation centres. For example, the product attributes required to assign fixtures at a milling operation centre cannot be the same as those required at a turning operation centre.

This numerical example was illustrated using a milling operation centre. This machining centre was selected because it is one of the most versatile machining processes to show the applicability of the researched DSS. It can process part orders with various physical geometries and output requirements using its several operations. In order to represent cases using an OO method for this machining centre, in total, it was supposed that twelve key product attributes were adequate to represent product orders for decision-based part/fixture assignments. Assume these attributes were selected and proposed by experienced human experts to meet the objectives of this study. The numerical example was illustrated in computer-based laboratory environments in order to make it easier and more understandable to the readers. It was intended to assign and control milling fixtures in a simulated milling operations centre.

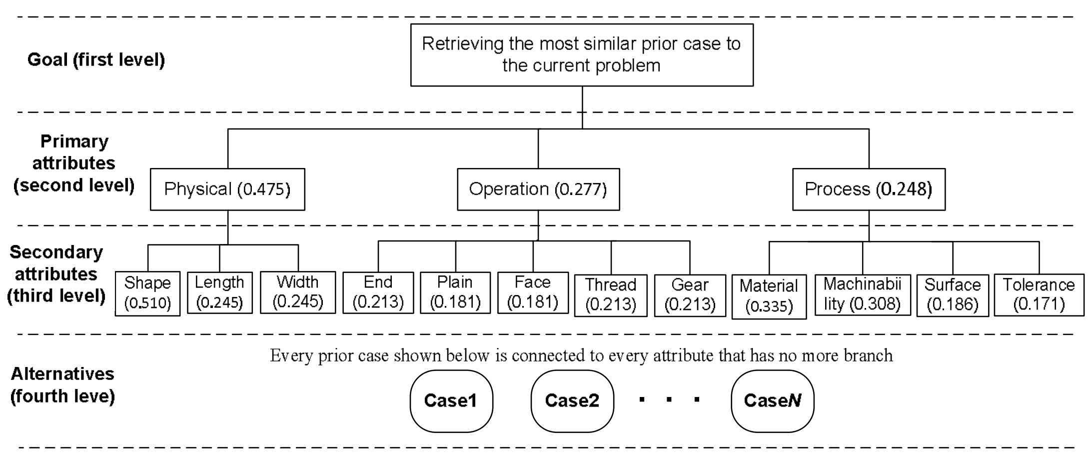


Figure 4-1: Structured case attributes and their weights using the AHP

The proposed twelve attributes were hierarchically structured using four levels presented in Figure 4-1. The first (top) level contains the goal of the MADM process i.e. choosing the most similar or relevant processed part order to the current order arrival. The second level incorporates three major attributes of products for MADM analyses such as: (a) the physical features of workpieces; (b) the types of operations required for milling a specific part order; and (c) the process requirements to carry out the required operations. These three major attributes were subdivided into the secondary sub-attributes at the third level. The fourth (bottom) level consists of all prior cases (*N*-training samples), which were regarded as solution alternatives to the target problems (new part orders). Each prior case was connected to every attribute that has no more further branches, in order to perform MADM analyses as illustrated in Saaty [110].

4.3 Prioritising product attributes

The weight of every primary attribute and secondary attribute at its corresponding level and under its parent attribute was evaluated with the help of the fuzzy AHP. Their normalised weights are indicated in parenthesis in Figure 4-1. The steps presented in Figure 3-3 were implemented in this section. For example, the primary attributes at the second level were compared as presented in Tables 4-1 to 4-5. Using the same approach, the secondary attributes under their preceding primary attributes were compared as depicted in Tables A-1 to A-12 of Appendix A.

a) Pairwise comparisons among the case attributes were carried out using fuzzy verbal terms with reference to the linguistic terms presented in Table 3-2 and Figure 3-4.

Table 4-1: Fuzzy relation matrix

	Physical	Operation	Process
Physical	Exactly equal	Equal to moderate	Equal to moderate
Operation	Reciprocal	Exactly equal	Equally preferred
Process	Reciprocal	Reciprocal	Exactly equal

b) The linguistic terms in Table 4-1 were converted into their equivalent fuzzy numbers and reciprocals. From this, a fuzzy reciprocal matrix was generated using triangular fuzzy numbers (refer to Table 3-2 and Figure 3-4).

Table 4-2: Fuzzy reciprocal matrix

	Physical	Operation	Process
Physical	(1, 1, 1)	(1, 2, 3)	(1, 2, 3)
Operation	(1/3, 1/2, 1)	(1, 1, 1)	(1, 1, 2)
Process	(1/3, 1/2, 1)	(1/2, 1, 1)	(1, 1, 1)

c) The fuzzy numbers in Table 4-2 were converted into standard fuzzy numbers in the range of $0 < x \leq 1$ as presented in Figure 3-5 and Table 3-3 by dividing them with the maximum value of the universe of discourse, which is 10, referring to Table 3-2.

Table 4-3: Fuzzy numbers are converted into standard fuzzy values of $0 < x \leq 1$ (Figure 3-5)

	Physical	Operation	Process
Physical	(0.10, 0.10, 0.10)	(0.10, 0.20, 0.30)	(0.10, 0.20, 0.30)
Operation	(0.033, 0.050, 0.10)	(0.10, 0.10, 0.10)	(0.10, 0.10, 0.20)
Process	(0.033, 0.050, 0.10)	(0.050, 0.10, 0.10)	(0.10, 0.10, 0.10)

d) The fuzzy numbers in Table 4-3 were defuzzified into their estimated crisp numbers with the help of Equation (3.1).

Table 4-4: Defuzzified numbers in the range of $0 < x \leq 1$

	Physical	Operation	Process
Physical	0.100	0.185	0.185
Operation	0.057	0.100	0.119
Process	0.057	0.085	0.100
Sum	0.213	0.370	0.404

e) In order to generate a normalised matrix, Table 4-5, every value in the column was divided by the corresponding column sum. The normalised weight of each attribute, w_i, was determined by calculating the average normalised value of each row. This was the classical approach of the AHP in MADM analysis.

Table 4-5: Normalised matrix

	Physical	Operation	Process	Normalised weight (w_i)
Physical	0.469	0.499	0.458	0.475
Operation	0.266	0.295	0.270	0.277
Process	0.266	0.231	0.248	0.248

f) The same approach was applied to all the secondary attributes that were branched from the same preceding attributes. The results are presented in Appendix A (please see Tables A-1 to A-12).

The normalised weights of the twelve attributes, which had no more further branches into their succeeding attributes, were calculated proportionally using the products of their normalised weights and the normalised weights of their preceding attributes. The results of these calculations from Figure 4-1 are summarised in Table 4-6.

Table 4-6: Hierarchy of case attributes and their normalised weights

Primary attribute	Secondary attribute	Normalised weight calculation	Normalised weight (w_i)
Physical feature (0.475)	Shape (0.584)	0.475x0.510	0.242
	Length (0.208)	0.475x0.245	0.116
	Width (0.208)	0.475x0.245	0.116
Operation types (0.277)	End milling (0.213)	0.277x0.213	0.059
	Plain milling (0.181)	0.277x0.181	0.050
	Face milling (0.181)	0.277x0.181	0.050
	Thread cutting (0.213)	0.277x0.213	0.059
	Gear cutting (0.213)	0.277x0.213	0.059
Process requirements (0.248)	Material type (0.335)	0.248x0.335	0.083
	Machinability (0.308)	0.248x0.308	0.077
	Surface finish (0.186)	0.248x0.186	0.046
	Tolerance (0.171)	0.248x0.171	0.042
Total			1.000

4.4 Product orders as fuzzy cases

This section focuses on the performances of the fuzzy CBR subsystem of the researched DSS. In order to represent the proposed product orders as fuzzy cases using an OO approach, a public class “PartOrder”, which implemented the Cloneable interface in order to create copies of product orders, was defined in the Java programming language. This class incorporated three constructors to create part order instances, part orders in the form of new problems and training samples. In total, this class used forty data fields and twelve of them were used to represent the attributes of part orders. The remainders were applied to represent the weight of attributes, upper and lower limits of numerical attributes, operational costs of fixtures, codes of assigned fixtures and state of the retrieved fixture (refer to Appendix B).

In order to solve this specific problem using the proposed DSS, ninety-eight instance methods, twenty static methods and seventeen in-built Java library methods were employed. In addition, it utilised more than ninety rules that were incorporated to enhance the effectiveness of the researched CBR system. For the basic concepts regarding an OO programming using Java, interested readers are referred to Liang [79].

4.4.1 Case representation using identified product attributes

The identified twelve attributes were expressed in the form of descriptive, crisp (numerical and nominal) and fuzzy data. The shape and material type of workpieces were represented in terms of symbolic/descriptive attributes. The shape was described using short and descriptive terms such as cylindrical, rectangular, hexagonal, I-shaped, etc. The construction material type was also described in terms of its chemical compositions such as carbon steel, aluminium, stainless steel, cast iron, etc. The length, width (diameter to cylindrical shapes) and tolerance limit of product orders were represented using continuous numerical values since these values were easy and simple for users to measure and understand.

The machinability of workpieces and the surface smoothness of finished product orders were described in terms of fuzzy linguistic terms. Machinability is one of the complex features to express using numerical forms. Various factors can affect the machinability of a given material such as material composition, heat treatment, workpiece geometry, grain size, etc. This attribute can be suitably described with the help of linguistic terms rather than crisp numerical values. Considering these factors, it was described using fuzzy verbal terms such as “high”, “medium”, “low”, etc. Similarly, the surface smoothness of a processed product can be expressed in terms of either numerical values or verbal terms. In this numerical example, instead of measuring this attribute in micrometres, it was meaningful and easy to describe the surface smoothness of finished products using linguistic terms in the same way to the machinability. The same approach was applied to this case attribute. For both attributes, the linguistic terms were converted into their equivalent fuzzy numbers using the idea presented in Figure 3-6.

The remaining five attributes, which were regarded as the basic milling operations such as end milling (E), plain milling (P), face milling (F), thread cutting (T) and gear cutting (G) operations indicated in Tables 4-7 and 4-8 were represented in terms of nominal values of {0, 1}. For instance, if a specific product order requires an end milling operation, its value for this attribute is one; otherwise it is zero. The same approach was applied to the remaining four attributes. Table 4-7 indicates structured sixteen product orders ($P1$-$P16$) as new cases and Table 4-8 incorporates three training samples ($TS1$-$TS3$) as prior cases for the retrieval process. All new product orders were represented in terms of their twelve attributes as stated in Section 4.2. Length (L) and width (W) were measured in millimetre [mm]; and tolerance limit (Tl) was measured in 10^{-3} inch. Trapezoidal and triangular fuzzy numbers were assigned to the machinability and surface smoothness attributes with reference to their equivalent linguistic terms indicated in Figure 3-6. The recommended way to create trapezoidal fuzzy numbers from triangular fuzzy numbers was explained in Section 3.4.

Table 4-7: Case representation of new product orders

Part	Shape	Material type	Machinability	Surface finish	*L*	*W*	*Tl*	*E*	*P*	*F*	*T*	*G*
*P*1	Rectangular	Carbon steel	0.7,0.8,0.9,1.0	0.8,0.9,0.9,1.0	585	290	8	1	1	1	0	0
*P*2	Cylindrical	Alloy steel	0.5,0.6,0.6,0.7	0.6,0.7,0.8,0.9	410	155	9	1	0	1	1	0
*P*3	Hexagonal	Cast iron	0.2,0.3,0.4,0.5	0.4,0.5,0.5,0.6	1000	500	8	1	0	0	1	0
*P*4	Rectangular	Stainless steel	0.1,0.2,0.2,0.3	0.8,0.9,0.9,1.0	1190	500	10	0	0	1	1	1
*P*5	I-shaped	Struct. steel	0.6,0.7,0.8,0.9	0.2,0.3,0.4,0.5	960	300	7	0	1	1	0	0
*P*6	Hexagonal	Cast iron	0.1,0.2,0.3,0.4	0.3,0.4,0.4,0.5	950	450	9	1	0	0	1	0
*P*7	Hexagonal	Carbon steel	0.8,0.9,0.9,1.0	0.7,0.8,0.9,1.0	405	160	2	0	1	1	0	1
*P*8	T-shaped	Alloy steel	0.5,0.6,0.6,0.7	0.6,0.7,0.8,0.9	590	295	5	1	1	0	0	1
*P*9	I-shaped	Cast iron	0.2,0.3,0.4,5.0	0.2,0.3,0.4,0.5	1200	175	4	1	0	1	1	0
*P*10	T-shaped	Alloy steel	0.5,0.6,0.6,0.7	0.4,0.5,0.6,0.7	610	300	3	1	1	0	0	0
*P*11	Cylindrical	Alloy steel	0.3,0.4,0.4,0.5	0.5,0.6,0.6,0.7	1150	230	7	1	0	1	1	0
*P*12	Cylindrical	Alloy steel	0.5,0.6,0.6,0.7	0.6,0.7,0.8,0.9	400	160	9	1	0	0	1	0
*P*13	L-shaped	Aluminium	0.7,0.8,0.9,1.0	0.3,0.4,0.5,0.6	760	420	8	1	1	0	0	1
*P*14	C-shaped	Alloy steel	0.5,0.6,0.6,0.7	0.6,0.7,0.8,0.9	580	300	5	1	1	0	0	1
*P*15	I-shaped	Struct. steel	0.4,0.5,0.5,0.6	0.1,0.2,0.2,0.3	1000	340	7	1	1	1	0	0
*P*16	L-shaped	Aluminium	0.7,0.8,0.9,1.0	0.3,0.4,0.5,0.6	750	440	6	0	1	1	0	1

Table 4-8: Case representation for prior product orders

TS	Shape	Material	Machinability	Surface finish	*L*	*W*	*Tl*	*E*	*P*	*F*	*T*	*G*	Fixture
*TS*1	Cylin.	Alloy steel	0.3,0.4,0.4,0.5	0.6,0.7,0.7,0.8	1145	228	8	1	0	1	1	0	Fix101
*TS*2	Rect.	Carbon steel	0.8,0.9,0.9,1.0	0.3,0.4,0.5,0.6	420	350	2	1	1	1	0	0	Fix201
*TS*3	Hex.	Cast iron	0.1,0.2,0.3,0.4	0.3,0.4,0.4,0.5	950	450	9	1	0	0	1	0	Fix302

(Note. The names of fixtures were arbitrarily given for the sake of illustrations).

The three training samples incorporated additional resources as extra attributes, which were "assigned fixtures". Assume this assignment was done using the experiences of similar order arrivals in the past. When such prior cases are not available in the system, the required training samples can be created through manufacturing and assigning few fixtures to a few well-defined product orders in order to use these samples as initial prior cases.

4.4.2 Minimum similarity measure and thresholds

The proposed DSS calculated the similarity between any new part order and any training sample, $sim(p,q)$, using Equation (3.12). The DSS read the normalised weights of product attributes as its input values, which were the outputs of the AHP. In order to propose a set of decisions based on the case similarity measures, the lower bound of the similarity measures and the thresholds of proposed decision sets were determined using the parameters in Figure 3-8. The thresholds were used to define the boundary of every proposed decision.

The researched DSS automatically generated the following essential numerical values when the normalised weights of case attributes from Table 4-6 were fed into the system. These generated numerical values were the minimum similarity value, which was the lower bound of the case similarity measures in the proposed DSS, $sim_{min}(p, q)$ = 0.657 using Equation (3.13); and the maximum similarity value, which was the upper bound of the case similarity measure, $sim_{max}(p, q) = 1.0$ from Equation (3.14), were found. Referring these two values, the value of $sim(p, q)$ was determined to be in the range of [0.657, 1.0] for this particular case. The medium similarity value, which was the average of the lower and upper bound values was found as $sim_{med}(p, q) = m = 0.828$. It should be noted that $sim_{med}(p, q)$ is not necessarily calculated from the average of these two values. This study regarded this value as the proposed value for one of the recommended scenarios (the first scenario). The details of these outputs are presented in Appendix C.

With the help of these calculated values and with reference to Figure 3-8, the linguistic terms "High", "Medium" and "Low", which were useful to describe the case similarities in verbal terms, were converted into their corresponding triangular fuzzy numbers such as (0.828, 1, 1), (0.657, 0.828, 1) and (0.657, 0.657, 0.828) respectively. These fuzzy numbers were created by shifting the right-leg of the term "Low" and the left-leg of the term "High" into the medium similarity value, 0.821. From these fuzzy numbers, the thresholds lm and mh were found as 0.743 and 0.914 respectively (see Figure 4-2). These thresholds were calculated using Equations (3.16) and (3.18) respectively and making $l = m = h$. Equating these parameters is not a rule; the decision should be left to the users of the system, human experts can propose any preferred values from their experiences so as to find the best thresholds. In this case, this approach was employed to simplify the numerical analysis and systematically determine the threshold values.

4.4.3 Distance measure for individual attributes

In order to search the case similarities, distance from target approach (the weighted Euclidean distance), which was regarded as one of the MADM approaches or the NN pattern matching functions, was utilised in this section. As stated in the previous chapter, this method combined the normalised weights of case attributes and the distance measures with respect to the individual case attributes with reference to Equation (3.2). The distance measures with respect to the four categories of case attributes are explained in this subsection.

In order to convert the two descriptive attributes (shape and material type) into nominal values of {0, 1}, the proposed DSS indirectly employed Equation (3.4) with the support of a few proposed rules and a Java in-built method. For example, when the shapes of the current and prior cases were described in terms of identical strings (words or phrases), their distance measure was expressed with the numeric string “0”; otherwise, the distance measure was expressed with the numeric string “1”. The same approach was applied to the material type. The Java in-built library method “Integer.parseInt(numeric string)”, which converts numeric strings into the same integer numbers in the “java.lang.Integer” class, was employed to return the integer values of {0, 1} from their numeric strings. For example, in this case, java.lang.Integer.parseInt(“1”) returns the integer value 1 and java.lang.Integer.parseInt(“0”) returns the integer value 0. Because the two case attributes are in the first and second places in the case representation scheme with reference to Tables 4-7 and 4-8, the individual distance measures $dist(a_1^p, a_1^q)$ and $dist(a_2^p, a_2^q)$ were determined using Equation (3.4).

Regarding the three numerical attributes named length, width and tolerance, the proposed DSS utilised Equation (3.3) to calculate $dist(a_5^p, a_5^q)$, $dist(a_6^p, a_6^q)$ and $dist(a_7^p, a_7^q)$, which were the individual distance measures with respect to the fifth, sixth and seventh attributes by referring to Tables 4-7 and 4-8. In this numerical example, the minimum values were set as 400.0, 150.0 and 2.0; and the maximum values were set as 1200.0, 500.0 and 10.0 to length, width and tolerance limit respectively.

The fuzzy attributes, the third (machinability) and fourth (surface smoothness) attributes in Tables 4-7 and 4-8 were firstly described in terms of linguistic terms. The linguistic terms were converted into their corresponding trapezoidal and triangular fuzzy numbers using the concepts presented in Figure 3-6. Using these fuzzy numbers, Equation (3.5) was utilised in order to determine $dist(a_3^p, a_3^q)$ and $dist(a_4^p, a_4^q)$, which are the distance measures with respect to the third and fourth case attributes.

Concerning the nominal attributes, from the 8th to 12th attributes in Tables 4-7 and 4-8, Equation (3.4) was applied to calculate the individual distance measures from $dist(a_8^p, a_8^q)$ to $dist(a_{12}^p, a_{12}^q)$.

4.4.4 Similarity measure and case selection

After determining the normalised weights of case attributes and the distance measures with respect to the individual case attributes, the weighted Euclidean distance between the current product orders and prior orders stored in the case library was calculated using Equation (3.2). This equation used the normalised weights of case attributes and individual distance measures as its important input variables to measure the weighted Euclidean distance between the concerned cases. Taking into consideration the inverse relationship between the distance and similarity measures, Equation (3.12) was applied to calculate the case similarities between the current case and each prior case in the case library when every new product order entered into the system. From these calculated results, a list of similarity measures in the range of [0, 1] was generated.

In order to select a prior case with the maximum similarity measure, the Java in-built library method "max(list)", which returns the maximum value on a list of objects, was employed in the "java.util.Collections" class. This Java library class incorporates a number of in-built functions to operate a given array list. For example, while the first product order $P1$ was arriving at the system, the proposed DSS calculated the similarities between $P1$ and $TS1$ or $sim(P1, TS1)$ = 0.716, $P1$ and $TS2$ or $sim(P1, TS2)$ = 0.952, P1 and TS3 or $sim(P1, TS1)$ = 0.717 referring to Equation (3.12) and Tables 4-7 and 4-8. This generated an array list of three numerical values. Using the Java in-built function, the maximum value on this list was returned as java.util.Collections.max (0.716, 0.952, 0.717) = 0.952. According to this result, the most similar or relevant previous case to the current part order $P1$ was $TS2$. It was shown that the retrieved prior case was TS2 and the retrieved fixture was the one assigned to the second training sample $TS2$ or Fix201.

In order to access the retrieved fixtures, the researched DSS used the combination of the two Java in-built library methods, specifically "get(integer)" and "indexOf(object)" in the "java.util.ArrayList" class. These methods were implemented to return a case in the case base at a specified index and the index of the first matching case in the case library respectively. In the DSS, the index of the retrieved case in its case library and the index of the maximum similarity measure on its similarity list were identical. This was because the number of elements in the case library and the number of element on the similarity list were the same during the given case retrieval operation. The same computational approach was applied while every new product order was entering into the system except the number of elements on the similarity list increased after *P*5, *P*8 and *P*13 had been processed as presented in Table 4-9.

4.4.5 Decision analysis

According to the proposed DSS, the retrieved fixtures were evaluated by human experts whether they were in functional or failed state. The states of these fixtures were suitably described using verbal terms as stated in Section 3.4.2. The system requested its users to enter the assessment result in the form of a trapezoidal fuzzy number using the right conversion scales presented in Figure 3-6 after every case retrieval stage. The fuzzy number was converted into its equivalent crisp number using Equation (3.1). A threshold was also proposed to accept or reject the retrieved fixture after the evaluation result. This threshold should be decided by experienced human experts. In this numerical example, the threshold was 0.90 for the sake of illustration and a few additional rules were suggested to implement the threshold using (If…, then…) general knowledge dependent rules.

- If the state of the retrieved fixture is below 0.9, then the retrieved case should be removed from it case library for permanent modifications/discard and a new fixture must be manufactured to the current part order arrival to replace the retrieved case.
- If the state of the retrieved fixture is equal and/or above the threshold, then the similarity measure between the current and retrieved cases must be considered to reuse/adapt the retrieved cases or manufacture new fixtures.

The developed DSS was able to remove the retrieved case from the case library at the specified index using the Java library function “remove(integer)” that removes an object on a list at a specified index. In addition, the DSS used the library function “add(integer, object)” in order to add the current case together with its manufactured fixture into the case base in the place of the removed case. This method was useful to add a new object at a specific index on a list. Both of these functions are defined in the “java.util.ArrayList” class. This Java library class also assimilates several in-built methods to manage elements on an array list. These two in-built methods were utilised to remove and replace nonconforming cases for fixture assignment problems as presented in the next paragraph.

In this illustrative example, as *P*3 entered into the system with reference to Table 4-9, the most similar prior case was *TS*3, with the similarity measure of 0.979 and the retrieved fixture of Fix302. In order to test whether the researched DSS could remove nonconforming cases and replace them with new cases, the system was deliberately fed with the functional state of the retrieved fixture below the threshold i.e. a trapezoidal fuzzy number whose equivalent crisp value was less than 0.9. In this situation, the system recommended the removal of TS3 and manufacture of a new fixture (Fix305) instead of reusing the retrieved one (Fix302). The current case *P*3 including its newly manufactured fixture (Fix305) was added into the first case library in the place of the retrieved case *TS*3. Starting from this time, *P*3 served as a new prior case for future retrieval, reuse and adaptations instead of *TS*3. This was proved when *P*6 arrived at the system as the current product order. *P*6 was identical to TS3 in order to elucidate the above argument. The best similarity measure was 0.979, which was the similarity measure between *P*6 and *P*3. This indicated that $sim(P6, P3) = sim(P3, TS3)$. If the system had not removed *TS*3, the best similarity measure would have been 1.0, which was the similarity measure between *P*6 and *TS*3 (*P*6 = *TS*3) (please see Tables 4-7 and 4-8). In addition, the number of cases in the first case library should have been increased by one after *P*3 was processed if the retrieved case had not been replaced.

Table 4-9: Summarised results of the proposed DSS (first scenario)

New part	Batch size	Retrieved case	Similarity measure	Proposed decision	Proposed fixture	No. of cases in 1st library	No. of cases in 2nd library
*P*1	33	*TS*2	0.952	Reuse	Fix201	3	1
*P*2	37	*TS*1	0.888	Adapt	Fix101	3	2
*P*3	44	*TS*3*	0.979	Remove	Fix302/Fix305	3	2
*P*4	38	*TS*2	0.798	Adapt	Fix201	3	3
*P*5	29	*TS*2	0.723	Manufacture	Fix502 (new)	4	3
*P*6	48	*P*3	0.979	Reuse	Fix305	4	4
*P*7	46	*P*3	0.784	Adapt	Fix305	4	5
*P*8	18	*TS*2	0.729	Manufacture	Fix508 (new)	5	5
*P*9	49	*P*5(*TS*4)	0.856	Adapt	Fix502	5	6
*P*10	30	*P*8(TS5)	0.940	Reuse	Fix508	5	7
*P*11	47	*TS*1	0.995	Reuse	Fix101	5	8
*P*12	50	*TS*1	0.877	Adapt	Fix101	5	9
*P*13	32	*P*5(*TS*4)	0.737	Manufacture	Fix703 (new)	6	9
*P*14	45	*P*8(*TS*5)	0.758	Adapt	Fix508	6	10
*P*15	13	*P*5(*TS*4)	0.931	Reuse	Fix502	6	11
*P*16	27	*P*13(*TS*6)	0.922	Reuse	Fix703	6	12

(Note. * Since the retrieved fixture was in a failed state, it was replaced by a newly manufactured fixture).

As the retrieved fixture existed in a functional state, the similarity measures between the current and retrieved cases, the lower and upper bounds of similarity measures, and the thresholds *lm* and *mh* presented in Figure 3-8 were implemented to perform decision-based part/fixture assignments. According to the results from this numerical example, the rules of decision-making, which were proposed in Section 3.4.2, were utilised here using the proposed numerical values in Section 4.4.2. The rules were revised as follows:

- If $0.914 < sim(p,q) \leq 1.0$ is fulfilled, then reusing the retrieved fixture without any revisions is recommended.
- If $0.743 < sim(p,q) \leq 0.914$ is fulfilled, then an adaptation of the retrieved fixture is the recommended solution.
- If $0.657 \leq sim(p,q) \leq 0.743$ is fulfilled, then manufacture of a new fixture is preferred.
- If $sim(p,q) > 1.0$ or $sim(p,q) < 0.657$, then the input is invalid.

The summarised results shown in Table 4-9 were automatically generated using the DSS when every part order arrived at the system according to the cases represented in Table 4-7. When the retrieved fixture or newly manufactured fixture was assigned to the current product order, the proposed system added the copy of this new case into one of the two case libraries for future retrieval and adaptations. In order to perform this crucial task, the system used the “Cloneable” interface by overriding the “clone()” method defined in the Java “Object” class. In addition, after every product order was processed, the previous list of similarity measures was cleared to generate a new list of similarity measures for the next order arrival. The library method “clear()” was implemented to do this action, which is defined in the “java.util.ArrayList” class. This kept the numbers of cases in the first case library the same as to the number of similarity measures included in the similarity list when evey new product order was processed.

As indicated in Table 4-9, the proposed DSS started with three prior cases that were initially treated as training samples incorporating their attribute values and assigned fixtures. Assume that they were previously solved problems. When the first and second product orders ($P1$ and $P2$) were processed, the existing prior cases were adequate and the retrieved fixtures were found in conforming states to process these two orders. As $P3$ arrived at the system, the retrieved fixture was found in a failed state and a new fixture was required to replace this damaged fixture as explained above. For further information, sampled inputs/outputs of the proposed DSS are presented in Appendix C.

While $P5$ was entering into the system, the retrieved fixture was found in a conforming state but the best similarity measure between the current and retrieved cases $sim(P5, TS2)$ = 0.723 < 0.743. According to the stated rules of decision-making, the system recommended manufacture of a new fixture because an adaptation of the retrieved fixture was impossible since the variation between the two cases was very high. Assuming this proposal was accepted by the users; the required fixture was manufactured and assigned to the current product order. The numbers of cases in the first case library increased by one after $P5$ was processed. In other words, $P5$ with its newly manufactured and assigned fixture (Fix502) was included in the first case library to work as a new training sample ($TS4$) for future usages. The same happened when $P8$ and $P13$ were also processed. In the same way as $P5$, these two new cases were regarded as new training samples for future retrieval and uses. This was proved when $P9$, $P10$, $P14$, $P15$ and $P16$ were processed i.e. $P3$, $P5$, $P8$ and $P13$ were retrieved and reused/adapted as the new members of training samples. They were retained as learned cases into the first case library that consisted of the initial training samples.

As stated in Section 3.4.3, the objective of the first case library was to determine the number of active fixtures flowing in the system at specific machining centres during planned production periods. The number of these fixtures remained the same as the number of cases in the first case library after each arriving part order was processed. Because this case library was designed to consist the initial training samples/prior cases together with their assigned fixtures plus new cases that required manufacture of new fixtures. In this specific example, the system proposed that six fixtures were sufficient to process the sixteen batches of product orders presented in Table 4-7. The two fixtures, Fix101 and Fix201, were retrieved from the three prior cases; Fix305 was newly manufactured to replace the damaged fixture, Fix302. The remaining fixtures such as Fix502, Fix508 and Fix703 were also newly manufactured during the machining process when the fixture supply system was unable to adapt the retrieved fixtures due to unacceptable similarity measures between the corresponding new and retrieved cases.

While $P1$, $P6$, $P10$, $P11$, $P15$ and $P16$ were arriving at the system, reusing the retrieved fixtures were recommended as solutions by the developed DSS because the similarity measures between the current and prior cases, $sim(p, q)$ values were in the range of (0.910, 1.0]. When $P2$, $P4$, $P7$, $P9$, $P12$ and $P14$ entered into the system, adaptations of the retrieved fixtures were proposed since the similarity measures, $sim(p, q)$ values were within (0.743, 0.910]. These twelve cases were placed in the second case library that contained product orders as cases, which utilised the retrieved fixtures from the first case library. This case library was used to recommend what tasks should be performed for the current proposed decision of an adaptation, when similar experiences were stored in the second case library. In manufacturing processes, when the users of the system encounter any new cases that are very similar to one of the cases such as $P2$, $P4$, $P7$, $P9$, $P12$ and $P14$, the same adaptation procedures to these cases can be applied in the future. For example, $P2$ and $P12$ were recommended to adapt the same retrieved fixture from the first training sample $TS1$. The similarity measure between these two cases $sim(P12, P2)$ = 0.941. Because of a strong case similarity between these two cases, the users could follow similar procedures to adapt $TS1$ for $P12$ depending upon what activities were done to adapt $TS1$ for $P2$.

The copies of new cases were added into their corresponding case libraries and indexed using the library method "add(object)" in the "java.util.ArrayList" class. This method appended the new case at the end of the list (case library in this case). For example, in this example, $TS1$ was the first element and $P13$ was the last (6^{th}) element in the first case library. Similarly, $P1$ was the first object and $P16$ was the last (12^{th}) object in the second case library. In order to unveil the number of cases available every time in the case libraries, another Java library method "size()", which returns the number of elements on a list, was utilised in the same class.

In manufacturing situations, the cost effectiveness of an adaptation decision must be taken into account. In this example, $P6$ and $P7$, and $P11$ and $P12$ were consecutively arriving at the system in order to use the same retrieved fixtures (Fix305 and Fix101) without adaptations and with adaptations respectively. When these parts were processed on a single milling machine, a significant machine downtime was anticipated to adapt Fix305 for $P7$ and Fix101 for $P12$. The machine downtime and setup costs can be significant to reverse the previous recommended and implemented decisions based on similarity measures alone. In such conditions, as every adaptation decision was passed, comparing the cost of adaptations with the cost of manufacture of a new fixture was recommended. This idea becomes meaningful when these two categories of costs are appropriately estimated by other expert systems or human experts. Cost estimation was beyond the scope of this study as stated in Chapter 3.

In addition to the two case libraries, a database to present the availability of fixtures was designed using the Java "ArrayList" class. The database used the in-built Boolean library method "contains(object)" to returns "true" as the required fixture was found in the database. The method "remove(object)" was also included to remove the retrieved fixture from the database when the machining process begins using the retrieved fixture. In addition, the Java library method "add(object)" was employed to add newly manufactured and retrieved fixtures into this database as the machining process was finished. The method “size()” was exploited here in order to update the number of fixtures in the database.

Based on this database, when the decision to reuse or adapt was passed, the proposed DSS checked the availability of the retrieved fixture in its database. If it is available in a functional state then the fixture should be accessed from the database and assigned to the new part order arrival. Otherwise, it should be in the process and the current part order should wait the device from the process. These decisions should be employed when two or more machining centres were implemented to perform the same activities through sharing the same fixture. Since a single milling centre was treated in this numerical example, the availability checker was not implemented; however, the proposed DSS was capable to address this situation.

4.5 Scenario analysis

The results depicted in Table 4-9 were based on a single scenario proposed by the fuzzy CBR subsystem of the researched DSS. According to this scenario, six different types of fixtures were required to machine the sixteen batches of product orders. The solution was determined using the thresholds (*lm* and *mh*) when $l = m = h = 0.828$ in Equations (3.16) and (3.18). However, the users of the system could not be confident at that stage whether the proposed solution performed according to the expected performances of the machining centre. This was because several alternative solutions could be generated by changing the values of the parameters (*l, m and h)* in Figure 3-8. It was visible that the interval or thresholds of the proposed decisions to reuse and/or adapt the retrieved fixtures or manufacture new fixtures could be changed when the values of these parameters were altered. These values were strongly related to the weights of case attributes as explained in Section 4.4.2. For example, the values of these parameters could be varied by adding /removing some case attributes in the case representation matrix, revising weights allocated to case attributes, changing the shapes of the three fuzzy numbers presented in Figure 3-8 or changing the combinations of two or more of these factors.

4.5.1 Results from fuzzy CBR

In this numerical example, it was assumed that the selection of case attributes and rating their weights were carried out by experienced human experts. Reversing these two factors was relatively expensive in manufacturing situations. Changing the shapes of the fuzzy numbers in Figure 3-8 was much easier, simple to understand and more systematic to improve the initially proposed solution. One of the approaches to change the shapes of these fuzzy numbers was logically shifting the medium similarity measure, $sim_{med}(p,q)$, right and left. Using this strategy, three scenarios were compared and contrasted in this section.

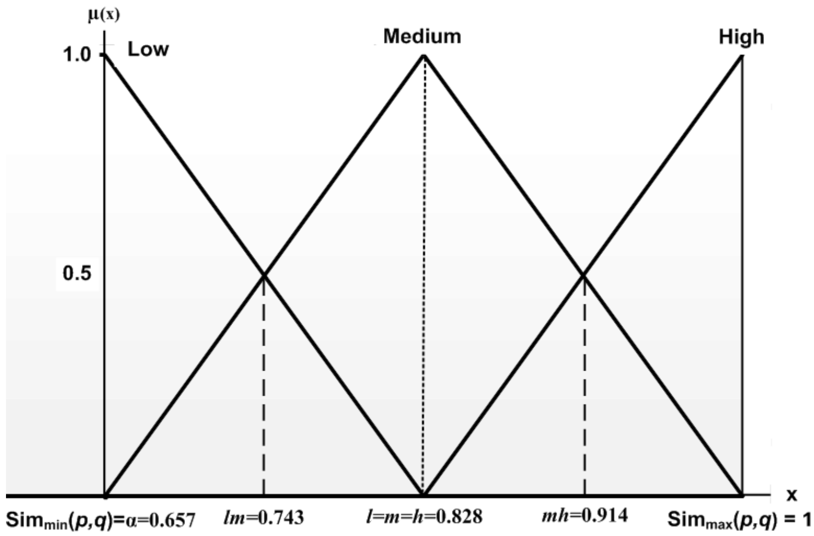


Figure 4-2: Values of parameters for the first scenario

In the first scenario, the value of $sim_{med}(p,q)$ = *m* = 0.828 (*l* = *m* = *h*) was determined as the mean values of $sim_{min}(p,q)$ and $sim_{max}(p,q)$ as presented in Figure 4-2. However, $sim_{med}(p,q)$ should not be necessarily calculated in this way. In this example, it was systematically applied to estimate the solution of the first scenario alone. This estimated solution was varied by moving $sim_{med}(p,q)$ right and left of this estimated value. The results from the first scenario were compiled in Table 4-9 and the rules of decisions were presented in Section 4.4.5.

The second scenario was proposed by shifting the value of $sim_{med}(p,q)$ right into the value of *m* = 0.950 and making *l* = *m* = *h* as depicted in Figure 4-3. The thresholds were calculated and found as *lm* = 0.803 and *mh* = 0.975 using Equations (3.16) and (3.18) respectively. The decision rules were modified and presented below.

- If $0.975 < sim(p,q) \leq 1.0$ is true, then reusing the retrieved fixture is recommended.
- If $0.803 < sim(p,q) \leq 0.975$ is true, then an adaptation of the retrieved fixture is the recommended solution.
- If $0.657 \leq sim(p,q) \leq 0.803$ is true, then manufacture of a new fixture is preferred.
- If $sim(p,q) > 1.0$ or $sim(p,q) < 0.657$, then the input is invalid.

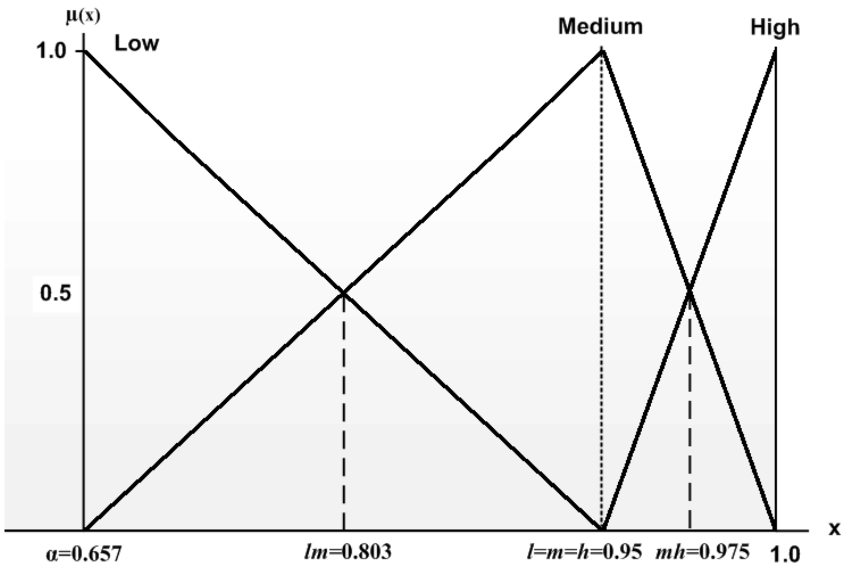


Figure 4-3: Parameter changes for the second scenario

Table 4-10: Summarised results from the second scenario

New product	Retrieved case	Similarity value	Proposed decision	Proposed fixture	No. of cases in 1st library	No. of cases in 2nd library
*P*1	*TS*2	0.952	Adapt*	Fix201	3	1
*P*2	*TS*1	0.888	Adapt	Fix101	3	2
*P*3	*TS*3	0.979	Remove	Fix302/Fix305	3	2
*P*4	*TS*2	0.798	Manufacture*	Fix405 (new)	4	2
*P*5	*TS*2	0.723	Manufacture	Fix502 (new)	5	2
*P*6	*P*3	0.979	Reuse	Fix305	5	3
*P*7	*P*3	0.784	Manufacture*	Fix407 (new)	6	3
*P*8	*TS*2	0.729	Manufacture	Fix508 (new)	7	3
*P*9	*P*5(*TS*4)	0.856	Adapt	Fix502	7	4
*P*10	*P*8(*TS*5)	0.940	Adapt*	Fix508	7	5
*P*11	*TS*1	0.995	Reuse	Fix101	7	6
*P*12	*TS*1	0.877	Adapt	Fix101	7	7
*P*13	*P*5(*TS*4)	0.737	Manufacture	Fix703 (new)	8	7
*P*14	*P*8(*TS*5)	0.758	Manufacture*	Fix804 (new)	9	7
*P*15	*P*5(*TS*4)	0.931	Adapt*	Fix502	9	8
*P*16	*P*13(*TS*6)	0.922	Adapt*	Fix703	9	9

(Note. * decision changes from reuse to adapt and adapt to manufacture as compared with the first scenario).

The results from the second scenario are presented in Table 4-10. According to this alternative solution, the DSS proposed manufacture of three extra fixtures as compared with the solution proposed in the first alternative. In addition, the proposed decisions of four product orders were modified from reusing to adapting the retrieved fixtures with reference to the first scenario; however, it was difficult in order to identify which alternative solution was able to perform better.

In the third scenario, the value of $sim_{med}(p,q)$ was shifted left i.e. m = 0.750 ($l = m = h$) as indicated in Figure 4-4. The thresholds were changed into lm = 0.704 and mh = 0.875 using Equations (3.16) and (3.18) respectively. The decision rules were altered and presented next.

- If $0.875 < sim(p,q) \leq 1.0$ is achieved, then reusing the retrieved fixture is recommended.
- If $0.703 < sim(p,q) \leq 0.875$ is attained, then an adaptation of the retrieved fixture is the recommended solution.
- If $0.657 \leq sim(p,q) \leq 0.703$ is attained, then manufacture of a new fixture is preferred.
- If $sim(p,q)$ > 1.0 or $sim(p,q)$ < 0.657, then the input is invalid.

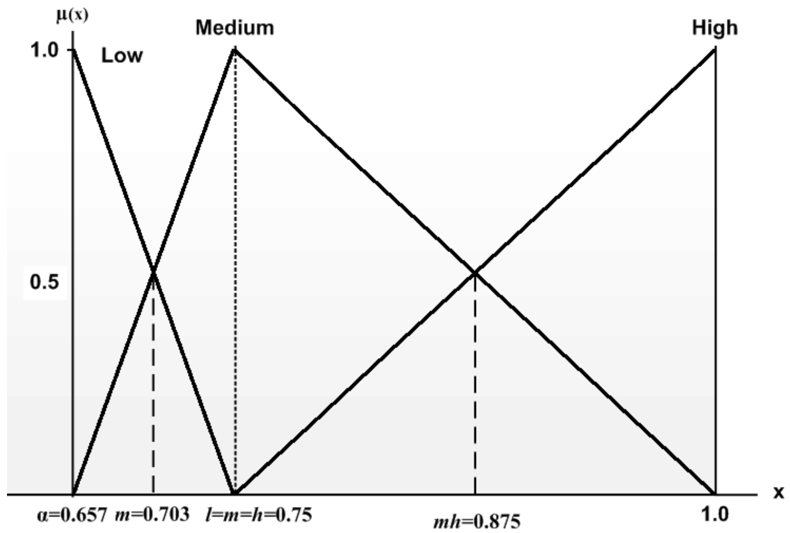


Figure 4-4: Parameter changes for the third scenario

The results from the third scenario are indicated in Table 4-11. According to the solution proposed from this alternative, no need of manufacture of new fixtures was recommended. The initial three fixtures were enough except replacing the third prior case, to process all the sixteen product orders.

The results from the three scenarios using the fuzzy CBR subsystem of the DSS are compiled in Table 4-12. The number of fixtures in the fixture database was the number cases in the first library (the number of active fixtures in the system) plus one. The database included the inactive fixture that was assigned to *TS*3.

Table 4-11: Summarised results from the third scenario

New product	Retrieved case	Similarity value	Proposed decision	Proposed fixture	No. of cases in 1st library	No. of cases in 2nd library
*P*1	*TS*2	0.952	Reuse	Fix201	3	1
*P*2	*TS*1	0.888	Reuse *	Fix101	3	2
*P*3	*TS*3	0.979	Remove	Fix302/Fix305	3	2
*P*4	*TS*2	0.798	Adapt	Fix201	3	3
*P*5	*TS*2*	0.723	Adapt*	Fix201	3	4
*P*6	*P*3	0.979	Reuse	Fix305	3	5
*P*7	*P*3	0.784	Adapt	Fix305	3	6
*P*8	*TS*2	0.729	Adapt*	Fix201	3	7
*P*9	*TS*1	0.742	Adapt	Fix101	3	8
*P*10	*TS*2	0.736	Adapt	Fix201	3	9
*P*11	*TS*1	0.995	Reuse	Fix101	3	10
*P*12	*TS*1	0.877	Reuse*	Fix101	3	11
*P*13	*TS*2	0.725	Adapt	Fix201	3	12
*P*14	*TS*2	0.729	Adapt	Fix201	3	13
*P*15	*TS*1	0.729	Adapt*	Fix201	3	14
*P*16	*TS*2	0.724	Adapt*	Fix201	3	15

(Note. * decision changes from adapt to reuse and manufacture to adapt with reference to the first scenario).

The roles of the two case libraries were explained in Section 3.4.3. The three scenarios generated three different results due to changes in parameter values in Figure 3-8. When the similarity measures between the current and retrieved cases are based on uniform distributions, the scenarios can generate the following chances for the decision alternatives from the proportion of the distance they cover.

a) The first scenario provides equal probabilities for reuse and manufacture decisions (25% for each) and 50% probability for an adaptation decision (Figure 4-2).
b) The second alternative favours the decision of manufacture. It gives 7%, 50% and 43% chances for reuse, adapt and manufacture decisions respectively (Figure 4-3).
c) The third alternative favours the decision of reuse. It provides 37%, 50%, and 13% probabilities for reuse, adapt and manufacture decisions respectively (Figure 4-4).

The results presented in Table 4-12 did not comply with these assumptions. This was because the similarity measures were not uniformly distributed. A few number of product orders were treated in the numerical example, whose similarity measures were skewed right. In addition, since the decision of the current order was based on the decisions of the preceding orders, it was difficult to find out uniformly distributed similarity measures between the current and retrieved cases.

However, when the scenarios were compared with each other, attractive results were found in this scenario analysis. The highest priority was given to the decision of manufacture of new fixtures in the second scenario and the lowest priority was provided to it in the third scenario. The second scenario was intended to reduce machine downtime due to fixture adaptations through manufacturing a number of fixtures during the planning time. The third scenario was proposed to utilise the available fixtures. As presented in Figure 4-5, the proposed stable numbers of fixtures in the milling centre were 6, 9 and 3 according to the first, second and third scenarios respectively.

Table 4-12: Summarised results from the three alternative solutions

Scenario	Number of proposed decision				No. of active fixtures (n)	No. of cases in 2nd library	No. of fixtures in database
	Reuse	Adapt	Manu.	Rem/manu.			
First	6	6	3	1	6	12	7
Second	2	7	6	1	9	9	10
Third	5	10	0	1	3	15	4

Note. Manu. = Manufacture, Rem/manu. = Remove and manufacture

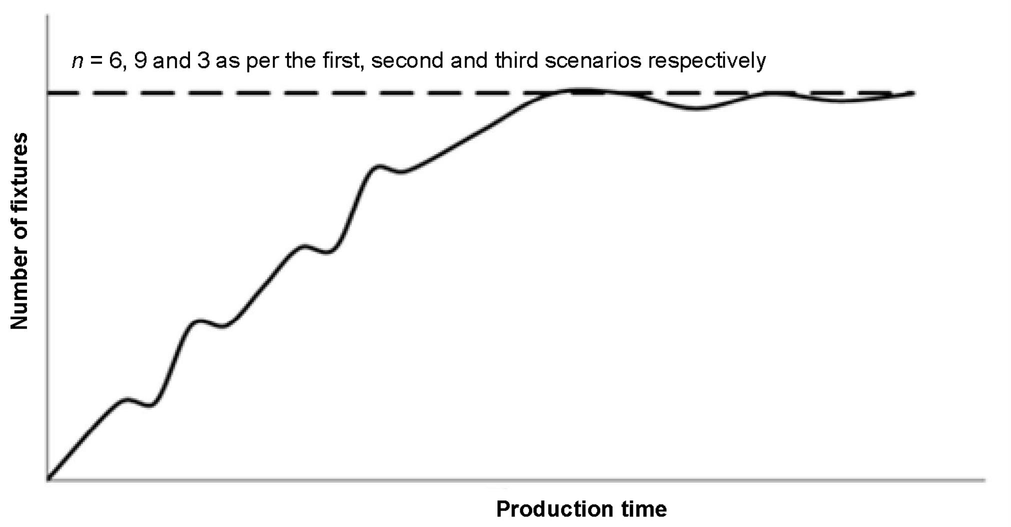


Figure 4-5: Proposed stable number of fixtures according to the three scenarios

At this stage, it was difficult to judge and identify which alternative solution could perform well using the results from the fuzzy CBR subsystem alone. Further analysis was inevitable in order to find sound and convincing results. According to this study, DES was recommended to validate the results from the fuzzy CBR component with reference to the explanations in Section 3.5.

4.5.2 Results from DES

The next step was simulating the necessary performances of the proposed solutions by the fuzzy CBR using DES models. The aim of this simulation study was to validate the performances of the three alternative solutions explained in Section 4.5.1. The proposed milling operations centre was modelled using FlexSim DES software package as stated in Section 3.5. A process flow diagram of the DES model is revealed in Figure 4-6. With reference to this figure, DES oriented terms such as "Source", "Queue" and "Sink" were included in the flow diagram. The term "Source" denoted a preceding machining centre or a storage of parts (workpieces) arriving at the ideal milling centre. The term "Queue" represented a waiting line or buffer of part orders waiting for the machining process. Finally, the "Sink" was used to designate a succeeding machining centre or a storage of processed orders.

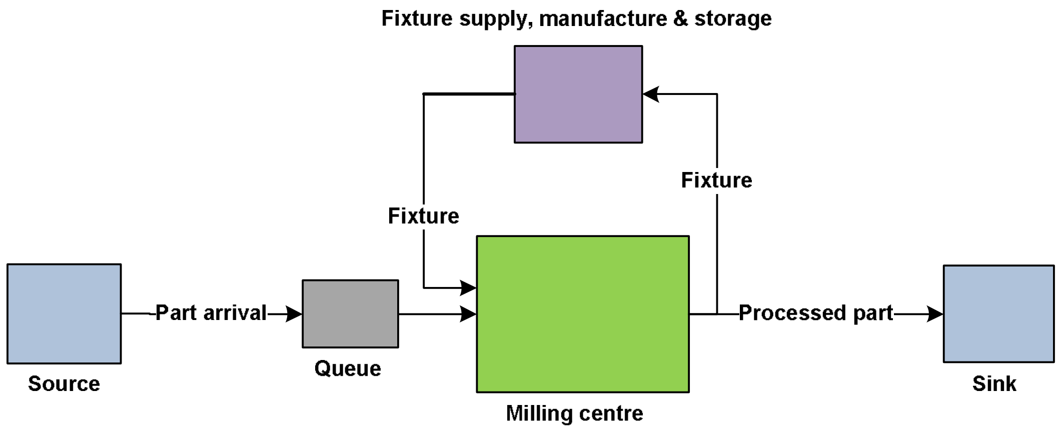
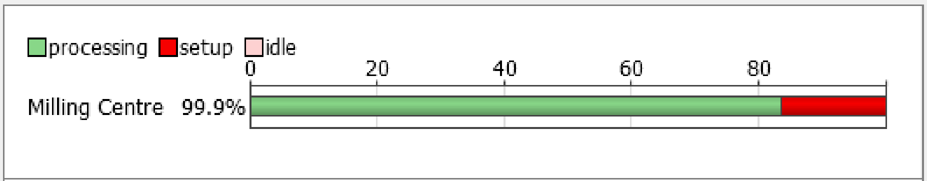


Figure 4-6: Flow diagram of a DES model for the numerical example

Among the three alternative arrival styles available in FlexSim such as "Inter-Arrival Time", "Arrival Schedule" and "Arrival Sequence", the "Arrival Sequence" was selected to define the arrival style of the product orders in this numerical example, assuming that these orders were scheduled and sequenced in advance. In the DES model, the effects of preceding and succeeding machining centres, and other resources were not taken into consideration in order to focus on the decision sets of fixture assignment problems. In other words, a set of proposed decisions were treated as the discrete-events of the system. This DES model was intended to determine the effects of decision-based part/fixture assignments at the fixture supply, manufacture and storage part on the performances of the proposed milling centre. These decisions affected the setup times and operational costs of fixtures, which were treated as random variables in this model. The setup state was different among the decision sets such as reuse, adapt and manufacture in the simulated operations centre. The same was true regarding the operational costs. For example, it was assumed that the required operational costs to manufacture a new fixture and those costs to reuse the retrieved fixture could not be the same in the milling centre.

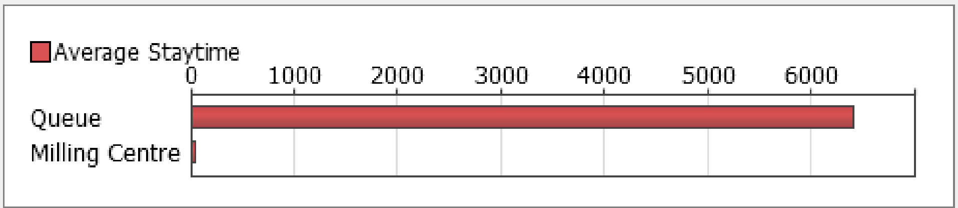
Knowledge uncertainties were addressed in this simulation model as much as possible in order to make the DES model more meaningful. Uncertainties can be articulated using either fuzzy set theory or probability distributions. In this DES model, uncertainties and imprecisions were expressed in terms of statistical distributions since the current version of FlexSim supports statistical distributions alone. For example, the machining process time and fixture setup time for each product order type were estimated using normal and exponential distributions respectively. Several customised rules were developed in the simulation software package to estimate these input variables (see Appendix D). An aggregated fixture setup time for every proposed decision per batch size was first estimated and the setup time per unit was calculated by dividing the cumulative by the quantity of product orders (batch size). The same was done to find the process time per unit in order to make easier the simulation process.

In order to compare the performances of the three proposed solution alternatives, machine utilisation, average stay time in the queue and machining centre, and operational costs of fixtures were regarded as the KPIs. Figures 4-7 to 4-9 present the utilisation of the milling centre, average stay time and operational costs of fixtures of the first scenario respectively. Similarly, Figures 4-10 to 4-12 and Figures 4-13 to 4-15 indicate the performances of the second and third scenarios respectively using the same KPIs as the first scenario. According to these DES results, the second alternative was identified as the best alternative and the third scenario was the worst alternative.



	Total	processing	setup	idle
Milling Centre	99.9%	83.5%	16.4%	0.1%

Figure 4-7: State analysis of a milling centre for the first scenario

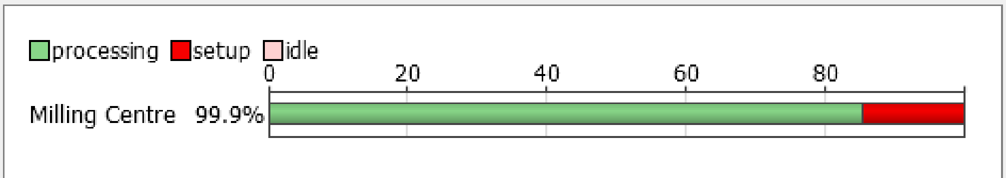


	Average Staytime
Queue	6381.1
Milling Centre	22.1

Figure 4-8: Average stay time of orders for the first scenario

	Cost
Totals	**$36,636.96**
Fixed	$0.00
Time	$0.00
State Fixed	$0.00
State Time	$34,511.29
Flowitems Fixed	$2,081.40
Flowitems Time	$44.27

Figure 4-9: Cost analysis for the first scenario



	Total	processing	setup	idle
Milling Centre	99.9%	85.5%	14.5%	0.1%

Figure 4-10: State analysis of a milling centre for the second scenario

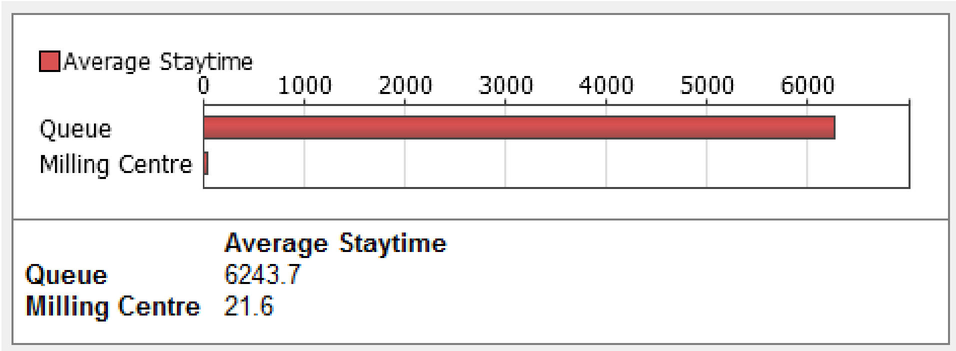


Figure 4-11: Average stay time of orders for the second scenario

	Cost
Totals	**$35,205.33**
Fixed	$0.00
Time	$0.00
State Fixed	$0.00
State Time	$32,717.28
Flowitems Fixed	$2,397.80
Flowitems Time	$90.25

Figure 4-12: Cost analysis for the second scenario

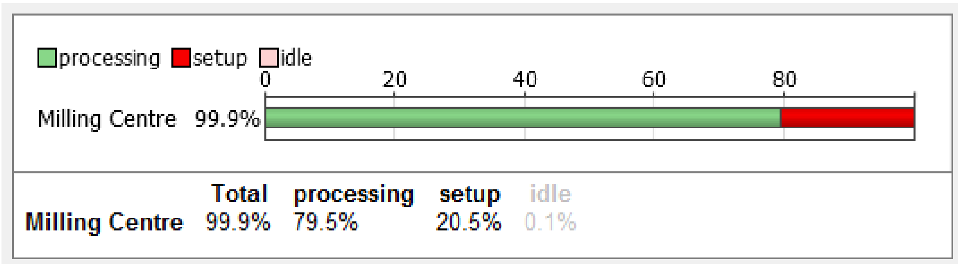


Figure 4-13: State analysis of a milling centre for the third scenario

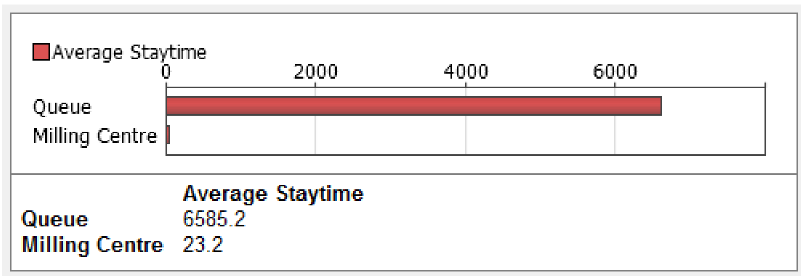


Figure 4-14: Average stay time of orders for third scenario

	Cost
Totals	**$40,495.25**
Fixed	$0.00
Time	$0.00
State Fixed	$0.00
State Time	$38,443.45
Flowitems Fixed	$2,051.80
Flowitems Time	$0.00

Figure 4-15: Cost analysis for third scenario

When the machine utilisation was treated as a key performance indicator (Figures 4-7, 4-10 and 4-13), the major factors to influence the performances were processing time and fixture setup time. The setup state incorporated only the fixture setup time to focus on the effect of fixture setup due to the three decision alternatives (reuse, adapt and manufacture). The author roughly estimated that the minimum setup time was required when the decision of reuse was passed and the maximum setup time was elapsed as the decision of an adaptation was implemented (reuse < manufacture < adapt). The first assumption was that when an adaptation decision was passed, more time was consumed for readjusting and/or reconfiguring the retrieved fixtures. The second assumption was that reusing the retrieved fixtures without any modifications was easier than fitting newly manufactured fixtures. These assumptions were applied for the simulated environment; however, in manufacturing environments, the setup time per every decision can be more precisely estimated by human experts or knowledge-based expert systems. In this regard, the best scenario was the one, which minimised the setup state and maximised the processing state. Since an idle state was the same for all the three scenarios, it was not considered in this scenario analysis.

The second performance indicator was the average stay time in the queue and milling centre as depicted in Figures 4-8, 4-11 and 4-14. With respect to this indicator, an alternative with the shortest stay time was preferred in order to reduce the Manufacturing Lead-Time (MLT) in the simulated milling process. A process with a higher setup state was considered to have a longer MLT that affected the delivery time and productivity of the milling process under consideration.

The operational costs of fixtures were treated as the third performance indicator. The results from this indicator are presented in Figures 4-9, 4-12 and 4-15. In these figures, six cost components were automatically generated by the simulation software package. The author utilised only those components, which were relevant to express the operational costs of fixtures. The first two cost indicators such as “Fixed” and “Time” were used to express fixed and variable plant costs. Since they were common to all solution alternatives and decision sets, they were excluded from the cost estimation process. The third indicator was the "State Fixed", which was used to measure the fixed costs of state changes (for example from processing state to setup state and vice versa). Because the numbers of state changes were the same in all scenarios, this cost component was not considered in the operational costs of fixtures. The "State Time" cost incorporated the total processing and setup state costs of the milling centre. The processing and setup states varied among the three alternative solutions in the planned production periods since the numbers of these decision sets were different among the scenarios as presented in Table 4-12. The setup state and processing state were highly dependent upon the types of the proposed decision sets. Because of this, they were treated as the significant cost components in this model. It was assumed that the cost of setup state per unit time greater than the cost of processing state per time in order to penalise alternatives with a lengthy setup time (please see Table E-1 in Appendix E).

The remaining two cost components were "Flowitems Fixed" and "Flowitems Time". The "Flowitems Fixed" cost was used to assign fixed costs to any flowing items (product orders, fixtures, tools, etc.). In this cost component, only fixture-oriented costs were included with the assumption that costs related to product orders and other resources were constant to all the three scenarios. A set of decisions passed during decision-based part/fixture assignments affected the fixed costs of fixtures in the system. For example, in the case of reuse, only fixture retrieval cost was included; in an adaptation decision, overhead and labour costs to readjust and/or reconfigure the retrieved fixture were estimated; and when a decision of manufacture was passed, the costs of fixture design, material and overhead were incorporated. Based on these cost components, the minimum estimated cost was incurred when the decision of reusing the retrieved fixture was passed and the maximum estimated cost was assigned when the decision of manufacture of a new fixture was passed. The "Flowitems Time" cost included the variable costs of flowing items. In this simulation process, the holding and storage costs of newly manufactured fixtures were incorporated. The holding and storages costs of the initially available fixtures were not taken into account because they were constant to all the three scenarios. These estimated costs are presented in Tables E-2 to E-4 in Appendix E with respect to the three solution alternatives.

Referring to Figures 4-9, 4-12 and 4-15, the highest "State Time" cost was found in the third scenario and the lowest was in the second scenario. However, the highest "Flowitems" cost was found in the second scenario and the lowest was in the third scenario. In total, the lowest operational costs of fixtures were recorded in the second scenario according to this numerical example.

4.6 Summary

In this chapter, a numerical analysis was performed to test the DSS developed in this study. The applicability of the solutions proposed by the fuzzy CBR component of the DSS was validated using a DES model. A milling operation centre was considered to demonstrate the numerical example. Twelve case attributes were hierarchically presented and selected to represent product orders as cases. The weights of these case attributes were determined using the fuzzy AHP. The fuzzy linguistic terms were converted into their equivalent fuzzy numbers using the ideas presented in Figures 3-4 and 3-5, and Table 3-3. The fuzzy numbers were transformed into their corresponding crisp scores using Equation (3.1). The thresholds for decision proposals and the lower bound of the case similarity measures were automatically determined when the DSS was fed the normalised weights of case attributes from the outputs of the fuzzy AHP.

The case representation was performed using with the help of an OO approach to address the flexibility required in the case representation process of this study. The twelve product attributes identified in Section 4.2 were expressed in terms of descriptive terms, linguistic terms, continuous numerical values and nominal values. Equations from (3.3) to (3.5) were applied to measure the individual distances with respect to the individual attributes in these four categories of case attributes. Through combining the normalised weights of case attributes and the distance measures from the individual attributes, the weighted Euclidean distance between new and prior cases was measured with the help of Equation (3.2). Using the context of an inverse relationship between the distance and similarity measures, Equation (3.12) was used to measure the case similarities between the current and prior cases. In addition, several in-built methods and classes from the Java library were exploited in order to retrieve the most similar cases from their case libraries and access the required fixtures from their database.

Three alternative solutions were proposed by the fuzzy CBR part of the proposed DSS by changing the values of the parameters in Figure 3-8 and implementing Equations (3.16) and (3.18). These alternatives proposed three stable numbers of fixtures to flow in the simulated machining centre in the planned production period due to the changes made in a set of decisions. The performances of these three scenarios were simulated using a DES model through treating a set of decisions as the discrete-events of the system. Machining process times, fixture setup times and fixture-oriented costs were regarded as random variables. The scenarios were compared in terms of KPIs termed machine utilisation, average stays in the queue and machining centre, and operational costs of fixtures incurred due to the changes decision sets in decision-based part/fixture assignments. The performances of the alternative solutions with the help of these KPIs were presented to identify the best solution alternative.

From this analysis, promising results were found in order to capitalise the concepts of combining CBR and DES methodologies for solving such kinds of complex problems, which have not been addressed yet in previous studies. It was noted that the findings presented in this chapter were in line with the theoretical framework synthesised in Section 2.8.

CHAPTER 5

5. DISCUSSION

5.1 Introduction

This chapter discusses the findings presented in Chapter 4 in line with the theories and methods presented in Chapters 2 and 3 respectively. The first two sections briefly explain whether the problem statement and objectives, which were stated in Chapter 1, are met with reference to the findings in the previous chapter. The problem statement and objectives are restated and discussed to answer the research questions, which were outlined as specific research objectives in Section 1.2.

The implications of the methods implemented in this study are discussed in the next section. The relationships between the current theories stated in Chapter 2 and the findings in Chapter 4 in the views of the methodological approaches in this study are explained. The implications from the combination between fuzzy CBR and DES approaches are described. The performances of the three scenarios are discussed in terms of the KPIs presented in Section 4.5. The relationship between the numbers of fixtures flowing in the simulated manufacturing process and operational costs of fixtures are discussed.

The research contributions and limitations are explained in the last two sections. The research contributions are discussed in two subsections. The first section discusses the contributions of the entire study with reference to the research contributions stated in Section 1.4 and the contributions of the findings in Chapter 4. The second section briefly explains the contributions of the individual publications listed in Section 1.5. The limitations of this study when the DSS is implemented in industrial systems are described in the last section.

5.2 Identified research gap

It was stated that fixtures are one of the main problematic components in manufacturing systems. They can directly affect the performances of manufacturing systems. Although fixtures are one of the influential factors, the problems of a part/fixture assignment and fixture flow control were not sufficiently addressed in past studies. This was because conventional manufacturing systems focus on the issues of part planning alone. Furthermore, research findings in fixture planning were mostly focused on the problems of fixture design and manufacture using CAFD facilities rather than utilising the available fixtures.

Several techniques were proposed to make fixture designs more reconfigurable and modular to accommodate various workpiece types. Adaptations of previous fixture designs were well articulated in the past; however, studies in part/fixture assignment and control techniques, which could improve the utilisation of the existing fixtures, were very limited. This was identified as the current research gap in fixture planning and management studies. The existing sources of literature revealed that this research area would need more explorations. A few studies were conducted to systematically assign fixtures to their corresponding part order and control the stable flows of fixtures as stated in Section 2.3.

It was implied that systematic fixture assignment and control techniques should have been required in the current manufacturing processes using the right DSS in order to alleviate such kinds of problems. This study researched and developed a DSS in order to carry out a decision-based part/fixture assignment and control using an illustrative numerical example. The methodological soundness of this DSS was validated in a simulated manufacturing environment. It was shown that the proposed DSS was capable to assist its users to retrieve the most similar prior cases to the current assignment problems. Furthermore, the researched DSS was capable to assist the users to evaluate the current states of the retrieved fixtures and propose a set of decisions such as reuse and/or adapt the retrieved fixtures or manufacture new fixtures. This was done depending upon the state of the retrieved fixture and the case similarity measures between the current and retrieved cases (refer to Tables 4-9 to 4-11). It was noted that the proposed DSS was regarded as a promising and novel approach with the potential to fill the current research gap in DSS studies. It was applied to utilise the available fixtures by stabilising the flows of fixtures in the planned production periods. It was implied that the research problem presented in Figures 1-1 was adequately articulated in this study.

5.3 Aim and objectives achieved

The aim of this study was to research and develop a DSS that operates on simple decision sets in order to ensure *n*-bounded growth in the fixture flow. In this regard, this study researched and developed a DSS in order to determine the stable number of fixtures flowing in manufacturing systems in specified production periods. In addition to proposing the stable number of fixtures as the proposed solutions using the fuzzy CBR subsystem of the DSS, the performances of the proposed solutions were analysed in a simulated machining process. It was found that the developed DSS was able to perform “what-if” analysis, and identify and implement the best alternative among several alternative solutions. The promising and novel findings in this aspect were presented in Section 4.5.

In order to attain the aim of this research project, the following specific objectives were presented in Section 1.2.

a) Research the current state of the arts in DSS and identify areas of original contribution potential of the entire study.
b) Research and develop a DSS framework that integrates CBR, RBR, fuzzy set theory and MADM approaches.
c) Construct and represent fuzzy cases using an OO method.
d) Research and develop the right case retrieval and retaining approach.
e) Implement an artificial manufacturing environment in software to support the research and development of the DSS.
f) Validate and test the DSS model with respect to various decision parameters.

In order to address the first objective, this study reviewed wide ranges of literature sources. It was found that the problems of part/fixture assignment and control approaches were the most vacant spaces in the current research. This study identified this research area to contribute a new approach to the current body of knowledge in DSS. Another important area of contribution potential was researching the combination of CBR and DES tools as a new methodological approach. It was reviewed that this combination strategy has not been explored yet to articulate complex problems like stated in this study. This strategy was investigated in this study using a numerical example.

Regarding the second objective, the principal method utilised in the AI subsystem of the DSS was a CBR methodology. Other intelligent and expert systems were integrated to improve the effectiveness of the researched CBR system. An integration of CBR and RBR systems was applied during the proposal of a set of decisions (decision-based part/fixture assignments) based on the state of the retrieved fixtures and case similarity measures (see Section 4.4.5). In the case representation process, fuzzy case attributes were incorporated in order to address the uncertainty and vagueness associated with human thoughts and reasoning. This fuzziness was utilised to improve the flexibility of the case representation and decision-making process as shown in Section 4.4.1. In addition, fuzzy rules were applied to propose a set of decision alternatives presented in Section 3.4.2. A fuzzy MADM approach, the fuzzy AHP was utilised for evaluating the weight of case attributes (refer to Sections 3.3 and 4.3). It was stated the AHP was widely implemented in MADM analysis to elicit and represent knowledge and experiences from human experts and the users of the system. In addition, a fuzzy SAW method was implemented in order to weight case attributes in the third published paper in Section 1.5.

The third objective was addressed in Sections 3.2 and 4.4. In order to attain the flexibility, reusability and data compactness required in this study, an OO approach was employed for the construction and representation of cases from new and prior part orders. With the help of this approach, three types of case constructors were defined, four different categories of case attributes (descriptive, fuzzy, numerical and categorical information) were represented (refer to Section 4.4.1). Furthermore, several in-built functions and classes in the Java library were utilised, and various user-defined functions and rules were developed in order to address the research problem using this case representation approach.

In order to attain the fourth objective, distance from target approach of MADM was applied to measure case similarities. Two case libraries were created to retain cases as learned experiences in line with their objectives stated in Section 3.4.3. When distance from target approach, specifically the weighted Euclidean distance was applied, new part orders were treated as the target and prior cases were treated as alternative solutions for retrieval and adaptations. In addition, the weighted Euclidean distance measure was regarded as one of the NN pattern matching functions in the discipline of AI technologies. In order to measure the distance between the current and prior cases, the individual distances with respect to the individual attributes were calculated first using Equations (3.3) - (3.5) with reference to Section 4.4.3. These distance measures were combined with the normalised weights of case attributes from the outputs of the AHP to calculate the cumulative weighted Euclidean distance using Equation (3.2). By considering the inverse relationship between distance and similarity measures in pattern recognitions theories, Equation (3.12) was applied to generate a list of case similarity measures between the current product order and prior cases. The Java in-built method “max(list)” was applied to select the most similar prior case (please see Section 4.4.4). Two case libraries were created using the “ArrayList” class from the Java library to retain the two categories of cases explained in Sections 3.4.3 and 4.4.5. Several in-built methods from the Java library were utilised to index cases and access the retrieved fixtures as presented in Section 4.4.5.

In order to articulate the fifth objective, an artificial manufacturing environment for a milling centre was created using FlexSim, which is one of the recognised DES software packages in the world. The performances of the three scenarios in Section 4.5, which were proposed by the fuzzy CBR subsystem of the DSS, were evaluated by the DES model presented in Figure

4.6. The recommended decision sets from the CBR subsystem of the DSS were treated as the discrete-events of the system. The setup time per unit order and operational costs of fixtures per the proposed decision were treated as the random variables of the DES model.

Regarding the last objective, the three scenarios were validated in terms of the KPIs named machine utilisation, average stays in the queue and machining centre, and operational costs of flowing fixtures, based on specific decisions at fixture supply, manufacture and storage system. The last two objectives were addressed and discussed in-depth in Section 4.5.2.

5.4 Methodological implications

CBR and DES were applied as the principal methodological approaches in this study. Fuzzy set theory, RBR and MADM approaches were integrated with the researched CBR system in order to construct the AI subsystem of the proposed DSS. The results of fuzzy CBR representation and fuzzy MADM approaches were integrated for searching the case similarities. A fuzzy CBR system was used to represent the uncertain and imprecise values of case attributes. A fuzzy MADM named the fuzzy AHP was used to elicit and represent experts' domain knowledge and experiences to prioritise the weights of case attributes. Fuzziness was required in this DSS to emulate human thoughts and judgements in uncertain manufacturing environments. A fuzzy ranking method was synthesised to defuzzify verbal terms into their estimated crisp values when the weights of case attributes were ranked. A DES model was utilised as one of the major elements of the DSS to evaluate and predict the near future performances of the proposed solution alternatives from the CBR subsystem.

The next subsections discuss the findings of this study with reference to the existing theories of the methodological approaches, which were reviewed in Chapter 2 and implemented in this study. The findings regarding the combination of fuzzy CBR and DES methodologies are explained in-depth.

5.4.1 Implications from fuzzy CBR results

In this research, it was found that the fuzzy CBR subsystem of the DSS was utilised to propose a set of decision alternatives to support human experts by retrieving the most similar prior part/fixture assignment decisions to the current assignment problems. These decisions were proposed depending upon the status of the retrieved fixtures and the similarity measures between the current and retrieved cases. Following these substantial process outputs, the CBR system presented in this research served as an advisory system to human experts by recommending alternative solutions to solve complex problems, Beemer and Gregg [15]. Its capabilities to support human experts in unstructured situations were presented. This was done by reusing and/or adapting previously encountered successful experiences to the current situations instead of replacing human experts in crucial decision-making strategies. This implied that the researched CBR system was pertinent to the objectives of DSS as stated in Arnott and Pervan [12]. The central notion that DSS are designed to support human experts in decision-making in complex situations but not to replace them, was validated in this study as stated in Alter [6], Power [98] and Er [48]. This was the reason why the fuzzy CBR subsystem was incorporated as one of the key elements of the DSS developed to address the research problem domain in this study.

As it was illustrated in the numerical example, the developed DSS started its tasks with three prior cases in the first case library. This number was gradually increased into six and nine cases to process the sixteen batches of product orders according to the first and second scenarios respectively. In addition, the newly retained cases in the first case library were retrieved when other succeeding similar product orders entered into the system (see Tables 4-9 and 4-10). With reference to these findings, it was found that the CBR system developed in this research was designed to update continuously the number of cases in its case library. This feature of the researched DSS improved its effectiveness through time to accommodate dynamically changing manufacturing situations. In addition, this proposed DSS was efficiently trained using a few cases or data; however, other systems like ANNs were unable to accommodate this problem in dynamic situations as reviewed in Oh and Kim [91].

The proposed CBR acquired incremental and progressive learning from accumulated experiences to solve new problems instead of starting from scratch every time, which was in line with the CBR concepts presented in Aamodt and Plaza [2] and Kolodner [72]. Human reasoners usually prefer to reuse and/or adapt their past similar situations to the current problems. Remembering previously solved problems are boring to human users; however, computers are best to perform this activity as described in Kolodner [72]. In this context, it was found that the researched CBR system was consistent with the natural reasoning process of people. This theory was validated using an illustrative numerical example in Section 4.4.5.

In addition to retaining successful new experiences into the case libraries, the CBR subsystem of the researched DSS removed and replaced nonconforming cases with the current cases. This was happened when the third part order, P3, arrived at the system. As P3 entered into the decision-based part/fixture assignment system, the retrieved fixture was in a failed state. This case was removed together with its assigned fixture and replaced with a new case including its newly manufactured fixture as presented in Section 4.4.5. This indicated that the proposed DSS was intended to manage not only successful experiences but also failed cases at the planning stages of manufacturing. This was in line with the CBR theory that CBR systems can learn from unsuccessful experiences in order to avoid the recurrences of past mistakes in the future, Kolodner [72].

In order to solve the research problem in this study, explicit domain models were not developed for knowledge elicitation, unlike rule-based experts systems. Only important case attributes were identified for case representation and case similarity searching operations. Two case libraries were created according to their purposes defined in Section 3.4.3 and implemented in Section 4.4.5. For case retrieval and retaining strategies, relatively simple analytical models, used-defined functions and in-built Java library methods were applied in the Java platform. It was implied that the decision and solution proposal approach (the CBR methodology) utilised in the researched DSS was relatively easier to identify case attributes, process the required knowledge and maintain the system without the need of complex and explicit domain knowledge-based models, Watson and Marir [136].

Referring to Section 4.4.1, the case representation in this research was simple, flexible, comprehensive and easy to understand by its users. An OO case presentation approach was implemented in order to describe product orders as cases. Prior orders were represented in terms of their product attributes as problem descriptions and their assigned fixtures as solution descriptions (Tables 4-7 and 4-8). New part orders were designated using their twelve product attributes alone as problem descriptions, which were used to characterise the similarity between part orders to carry out decision-based part/fixture assignments as presented in Table 4-7. Four different forms of case attributes were incorporated to meet the required flexibility of the case representation in the numerical example; however, the proposed DSS could address any other forms of knowledge depending upon the needs of the users of the system. In this regard, it was found that the case representation in this work was flexible enough to represent the reasoners’ previous experiences in order to attain the objectives of case reasoners as stated in Bergmann *et al.* [18].

The uncertainty and vagueness associated with human reasoning and thoughts were articulated using fuzzy set theory in the fuzzy CBR subsystem. The two case attributes, namely the machinability of workpieces and surface smoothness of processed products were described in terms of fuzzy terms rather than crisp values. They were expressed in the form of their estimated fuzzy numbers in Tables 4-7 and 4-8. Furthermore, the weights of case attributes were described in terms of linguistic terms as presented in Section 4.3. In the numerical example, it was realised that fuzzy case attributes could accommodate more flexibilities than numerical-valued attributes. For example, the numerical attributes named length, width and tolerance could not accommodate changes when new part orders unpredictably entered into the system with attribute values above the upper bounds or below the lower bounds of these attribute values. However, in the case of the fuzzy attributes such as machinability and surface smoothness, there were no any upper and lower bound restrictions. The fuzzy attributes of any part orders were evaluated using properly designed conversion scales to transform linguistic terms into their equivalent fuzzy numbers. It was understood that the CBR subsystem presented in this research assimilated fuzzy set theory into the classical CBR approach, which was useful to enhance the decision-making process and accommodate vague and imprecise knowledge stored in the form of past cases in the case base as reviewed in de Mantaras [39] and de Mántaras and Plaza [40]. In other words, the proposed DSS articulated fuzzy set theory in order to increase the flexibility and applicability of the researched CBR subsystem of the DSS as stated in Chang *et al*. [29] and Li and Ho [77].

General domain dependent knowledge in the form of guiding rules was included into the fuzzy CBR part of the DSS. For example, in Section 4.4.5, several rules were presented to propose a set of decisions depending upon the thresholds, minimum and maximum similarity values. In total, more than ninety rules were incorporated in the fuzzy CBR subsystem for improving the case reasoning process. In addition, several general domain knowledge-based rules could be included into the researched DSS to evaluate the current states of the retrieved devices and adapt the retrieved cases/fixtures. It was reviewed that CBR problems should assimilate the required general domain knowledge-based rules to increase the effectiveness of CBR systems. The integration of RBR and CBR approaches was largely utilised in the DSS to capitalise their strengths and minimise their weaknesses in line with the studies in Aamodt [1], de Mántaras and Plaza [40], de Mantaras [39], Prentzas and Hatzilygeroudis [103], Golding and Rosenbloom [55], Dutta and Bonissone [43] and Chi and Kiang [34].

The weights of case attributes were evaluated using the fuzzy AHP, which is one of the popular MADM approaches for eliciting knowledge and experiences to prioritise decision-making actions or criteria, Saaty [110]. It was reviewed that the AHP could systematically acquire and represent experts' domain knowledge for rating case attributes, Park and Han [95]. With reference to Section 2.5.1, a variety of recent the AHP and fuzzy AHP applications were reviewed. A fuzzy ranking method was synthesised through merging the popular ranking approaches proposed by Chen and Hwang [32] and Chen and Chen [33] to defuzzify the linguistic terms that were utilised to weight case attributes in the AHP (see Section 3.3.2). This implied that the attribute weighting approach in this research was consistent with the current literature; however, there were no any restrictions to use other multi-attribute weighting methods based on the behaviour of problems in consideration. For example, in the third publication (Section 1.5), a fuzzy SAW method was employed to weight case attributes.

In order to retrieve the most similar historical case to the current part order, distance from target approach of MADM was implemented. Firstly, the distance measures with respect to the individual attributes were calculated in line with the studies in Slonima and Schneider [116] and Faez *et al*. [49]. Finally, the weighted Euclidean distance between the current and prior cases was determined by combining the individual distance measures and the normalised weights of case attributes from the results of the AHP, Park and Han [95]. The weighted Euclidean distance was regarded as one of the popular NN pattern matching functions for measuring case similarities as per the discussions in Park and Han [95] and Pal and Shiu [94]. In this perspective, it was implied that the case retrieval approach in this study was based on well-established and recognised similarity searching knowledge and theories.

5.4.2 Implications from DES results

The performances of the three scenarios, which were proposed by the fuzzy CBR part of the DSS, were analysed using the DES model presented in Section 4.5.2. Different numbers of fixtures were proposed as solution alternatives through varying the values of the parameters in Figure 3-8. The findings from the three scenarios presented in Section 4.5 are summarised in Table 5-1 for the sake of discussion in this section.

Table 5-1: Summary of findings from the three scenarios

Scenario	Number of manufactured fixtures	Number of active fixtures	State [%]		Cost [$]			
			Process	Set-up	State Time	Flowsitem Fixed	Flowsitem Time	Total
First	3	6	83.5	16.4	34511.29	2081.40	44.27	36636.96
Second	6	9	85.5	14.5	32717.28	2397.80	90.25	35205.33
Third	0	3	79.5	20.5	38443.45	2051.80	0.00	40495.25

As presented in Table 5-1, the processing state and “Flowitems” costs increased when the number of active fixtures in the system increased. The setup state and “State Time” cost diminished as the number of active fixtures flowing in the system increased. The relationship between the operational costs of fixtures and the number of fixtures flowing in the simulated milling operations centre is depicted in Figure 5-1. In this operations centre, when a large number of fixtures were manufactured and available in a system, the fixture setup time elapsed for readjusting and reconfiguring (adapting) the available fixtures was reduced. For example, in the second scenario, the highest number of decisions to manufacture new fixtures was proposed instead of adapting the available ones. The decisions of three part orders were reshuffled from adaptations of the retrieved fixtures to manufacture new fixtures as compared with the first scenario (see Table 4-12). Due to this reason, the lowest fixture setup state and "State Time" cost, and the highest "Flowitems" costs were found in the second scenario.

A lower setup state means a lower "State Time" cost because the setup state per unit time costed higher than the processing state per unit time according to the author's assumption in Section 4.5. The highest fixture setup state and "State Time" costs were incurred in the third scenario because of the highest setup time to readjust and reconfigure the available fixtures alone instead of manufacturing new fixtures. Referring to Table 4-12, the highest number of adaptation decisions was proposed in this scenario.

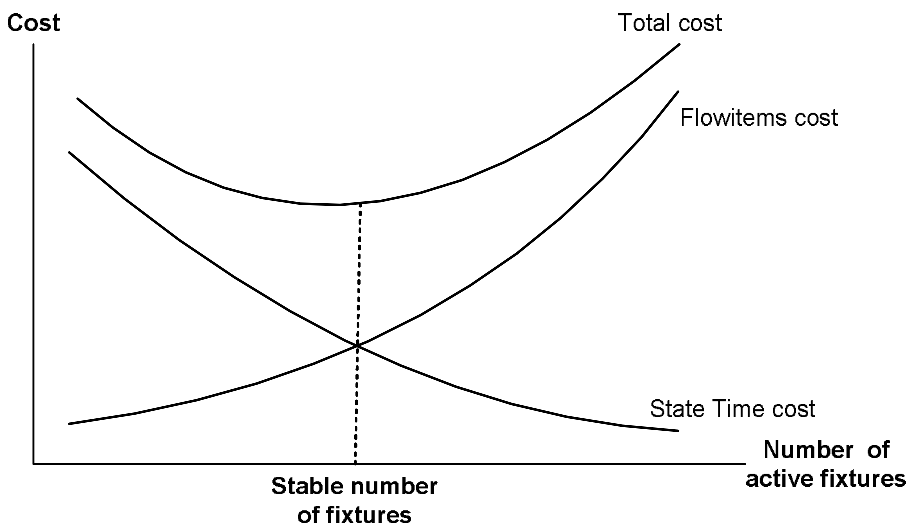


Figure 5-1: Costs of flowing fixtures

When a large number of fixtures were required in the system, the costs to manufacture new fixtures, and hold and store these newly manufactured fixtures were increased as shown in the second scenario. The reverse was true as a few number of fixtures were required as presented in the third scenario. These costs were categorised into "Flowitems" costs in

Section 4.5. Regarding these costs, the maximum was recorded in the second scenario because the maximum number of manufacture decisions was recommended in this alternative solution and the minimum was incurred in the third alternative since nothing was proposed to manufacture new fixtures.

With reference to Figure 5.1, a scenario with the right stable number of fixtures must be the one that minimises the total costs of the two cost components. It was assumed that the three curves of costs in Figure 5-1 were intractable to solve using mathematical equations in manufacturing environments. In this situation, the DES model was employed as the best methodological approach to model and solve such complex problems, which were beyond the scope of analytical models. Considering that the components of these cost categories were appropriately estimated by human experts or well-designed expert systems, the DES model proposed in this study was able to evaluate and predict the near future performance situations of several alternative solutions. From these results, it was shown that an alternative that could nearly determine the right stable number of fixtures could be identified and implemented as the best scenario among the available solution alternatives. It was implied that the DES model was capable to predict the near future situations of the proposed solutions from the fuzzy CBR using appropriate KPIs. This indicated that the findings from the researched DES model were consistent with the current DES theories stated in Chapter 2; however, this study addressed one of the dimensions of manufacturing systems, which has not been articulated in the current DSS research.

5.4.3 Implications from combination of fuzzy CBR and DES

In Chapter 4, three solution alternatives were proposed by the fuzzy CBR subsystem of the researched DSS through varying the values of parameters in Figure 3-8. It was so difficult that the users could not identify the best alternative solution using the results from the fuzzy CBR subsystem alone. In order to resolve this difficulty; the performances of the three scenarios were simulated and predicted using the DES model presented in Section 4.5.2. The proposed DSS determined the stable number of fixtures that could well perform in the proposed machining centre using the combination of the fuzzy CBR and DES methodologies. This combination was useful to reduce the uncertainties and risks of a single solution when knowledge and experience gaps were anticipated in case construction and representation, case attributes rating and case retrieval strategies.

As presented in the previous chapter, the fuzzy CBR was in charge to propose the stable number of fixtures required within the planned production period from a set of part/fixture assignment decisions using its accumulated experiences. The DES model was responsible to predict the near future performances of the proposed stable numbers of fixtures using the right KPIs (refer to Section 4.5.2). In this regard, it was found that the proposed DSS was able to perform decision-based part/fixture assignments and fixture flow control in order to reduce operational costs, increase productivity and enhance the on-time delivery of product orders, with reference to the results presented in Sections 4.4 and 4.5.

Using similar procedures it was implied that the proposed DSS could be extended to other machining operations such as turning, grinding, drilling, etc. For example, in order to implement this DSS at turning operations, system developers must identify and structure hierarchically specific part order attributes, which are suitable to create cases for decision-based part/fixture assignment strategies at a turning operation centre; and the weights of these attributes must be hierarchically evaluated as usual. For instance, in the case of turning operations, the shape of workpieces may not be significant because turning operations usually use workpieces with cylindrical shapes. It was found that the performance of the proposed DSS could be influenced by the selection of specific operation centres and the capabilities of system developers to represent experts' knowledge and judgements (Sections 3.2.1 and 4.2)

Acquiring the knowledge and experiences of experts or users at specific operations was highly appreciated in the DSS development process instead of imposing the developers' intentions on the users of the system (Sections 4.2 and 4.3). To meet this flexibility, the proposed DSS was designed that it must be highly interactive with its users for future learning and adaptations. Evolutionary learning, adaptation and flexibility were found the key features of DSS as reviewed in Chapter 2. In the case construction stage, when case attributes were suitably identified and their weights were appropriately evaluated using knowledgeable experts/users at specific operations, the DSS could support its users in the right way; otherwise, the reverse could be true. It was stated that the DES model in the DSS could be similarly utilised at other machining operation centres to evaluate and predict the performances of the proposed solutions by the fuzzy CBR (see Sections 3.5 and 4.5.2).

From the managerial perspective, it was realised that operational managers were able to plan fixtures in parallel to their part order plans and enumerate the available fixtures using the combination of CBR and DES approaches. It was found that the DSS was capable to avoid the unnecessary holding and downtime costs by stabilising the flows of fixtures during the planned production period as presented in Section 4.5.2. This was a new approach to improve the utilisation of the available limited resources in manufacturing processes through integrating CBR and DES methodologies. It was implied that operational managers and fixture planners were able to generate several solution alternatives in the fuzzy CBR part of the system. This could be done by adding into and or/deleting from the CBR system, some of the attributes of part orders; revising the weights of part order attributes; changing the attribute weighting methods (like fuzzy AHP, fuzzy SAW, etc.); varying the threshold values of decision alternatives; and changing the combination of two or more of these factors. These four factors had the potential to make differences in a set of decision alternatives that were treated as the discrete-events in the DES model to validate and predict the performances of various proposed scenarios as presented in Sections 3.5 and 4.5.

5.5 Research contribution

The contributions of this study are discussed in two ways in this section. Firstly, the contributions of the entire study are discussed. Secondly, the contributions of the individual publications, which were listed in Section 1.7, are briefly described.

5.5.1 Contribution of overall study

This study identified that a decision-based part/fixture assignment and control as one of the complex problems that have not been adequately studied in the past. It was reviewed that past studies regarding the determination of the stable number of fixtures within manufacturing processes were very limited. This study attempted this problem using a systematic fixture assignment and control strategy. This research problem was treated as the area of an original contribution potential in this study.

In this study, an intelligent DSS was presented to support decision-makers to carry out an on-demand fixture retrieval and propose decisions to reuse/adapt the retrieved fixtures or manufacture new fixtures depending upon the current state of the retrieved fixtures and the similarities between the current and retrieved cases. Using this strategy, the AI or fuzzy CBR part of the researched DSS was applied to perform systematic part/fixture assignments in parallel to any standard part planning approach. After completing these assignments, the stable numbers of fixtures required within the planned production time were determined as the final alternative solutions of the fuzzy CBR subsystem of the DSS (see Section 4.4.5). The performances of these alternative solutions were validated and predicted with the help of the DES model depicted in Figure 4-6. With reference to the current literature in DSS, the combination of fuzzy CBR and DES methodologies has not been utilised to solve such kinds of complex problems in the past. It was implied that the methodological approach presented in this study was regarded as a significant addition to the current DSS research. In addition, the DSS was developed to predict the near future situations of the proposed solutions using its DES part instead of using historical data to validate its accuracy. It was presented that the DSS could learn from the past using its fuzzy CBR subsystem and predict the near future situations using its DES component as explained in Sections 4.4.5 and 4.5.2. This powerful and novel methodological approach was not exploited in the past and this was the substantial contribution of this study to the current research in DSS.

The performances of the proposed solution alternatives were simulated in terms of machine utilisation, average stay time in a process and operational costs of fixtures. These KPIs indicated that the DSS could improve the utilisation of the available fixtures and other related limited resources, the manufacturing lead-time or on-time delivery and the operational costs of the processes under investigation. The relationship between the number of fixtures in the simulated milling centre and the operational costs of fixture was presented in Figure 5-1. This methodological approach was implemented as a novel and promising approach to find the stable number of fixtures that was able to minimise the total operational costs of fixtures and improve the productivity of manufacturing processes.

5.5.2 Contributions of individual publications

As presented in Section 1.5, the researcher co-authored six publications in international journals and conference proceedings. The contributions of these publications to this study are briefly described in this section.

In the first paper, an intelligent DSS was developed to carry out an on-demand fixture retrieval and propose a set decision alternatives such as reuse/adapt the retrieved fixture or manufacture new fixtures based on the case similarity measures. The research problem was addressed by integrating AI technologies such as CBR, RBR and fuzzy set theory. Fuzzy cases were represented using an OO approach to characterise order arrivals using their crucial attributes. The fuzzy version of the AHP was utilised to rate the importance of case attributes. The inverse of the Euclidean distance measure was applied for the sake of case retrieval. In order to rank fuzzy numbers, the right and left scoring approach was utilised using maximising and minimising sets. A demand-driven fixture retrieval and manufacture approach to perform a decision-based part/fixture assignment and fixture flow control was done using the proposed DSS. It provided special considerations for the decision of adaptations of the available fixtures in a system. A numerical example was illustrated using some operations of an ideal milling centre to unveil the soundness of the research findings.

In the second publication, the current literature in DSS studies was carefully reviewed to address the state of the arts in DSS, and the shortcomings of purely simulation-based and purely AI-based DSS. The dimensions of manufacturing systems, which have not been articulated in the current DSS literature, were identified as the areas of original contribution potential. A theoretical decision support framework, which integrates AI (largely CBR), DES and database management technologies in order to determine the steady state flow of items (e.g. fixtures, jigs, tools, etc.) in manufacturing, was proposed. A conceptual example was illustrated to reveal integrated performances of CBR and DES; taking into account the problems of flowing items such as fixtures, jigs and tools.

The third paper focused on the determination the stable number of fixtures based on its proposed part/fixture assignment and control approach. A DSS, which combines the CBR, RBR and fuzzy set theory elements of an AI approach, was presented in order to address its problem. Cases were represented with the help of an OO approach in order to characterise them by their feature vectors. Fuzzy SAW for weighting case attributes and the inverse of the Euclidean distance measure were combined for case retrieval activities. A numerical example was also illustrated to show the applicability of the proposed DSS.

The fourth publication articulated the problems of fixture planning and management in comparison with the attention paid to the design issues in fixture planning using a review of literature. A decision-based part/fixture assignment and control framework was proposed as the first step for future DSS development research. The theoretical framework integrated AI technologies, DES models and database management techniques. The AI subsystem revealed how CBR and RBR techniques could work in synergy. A decision-making algorithm was presented in order to show the required conditions for the proposal of decision alternatives.

In the fifth paper, a DSS was proposed in order to determine the stable flow of fixtures in manufacturing operations. A novel methodological approach was introduced through combing CBR, AHP, fuzzy set theory and DES approaches. Fuzzy cases were represented using an OO method to characterise cases with the attributes of product orders in *n*-dimensional Euclidean vector space. The fuzzy CBR and fuzzy AHP methods were combined in the case retrieval process using the inverse of the weighted Euclidean distance. A DES model was used to evaluate and predict the performance of a solution proposed by the CBR subsystem to minimise the uncertainties and risks of the proposed solution due to the lack of experiences in case and knowledge representations. A numerical example was illustrated to show the soundness of the proposed methodological approach.

The last paper was intended to extend the methodological approach of this study to articulate the cost estimation problems of new products. The paper presented how much the methods in this work are flexible and robust enough to address wide ranges of problem domains. A DSS was proposed to retrieve historical cases/products, which have the most similar cost estimates to the current product order. The implemented methodology combined CBR, AHP, fuzzy set theory and RBR approaches. Product orders as prior and new cases were represented using an OO representation approach. A numerical example was illustrated using lathe machine operations in order to show the applicability of the proposed DSS.

5.6 Research limitations

The research was implemented in a computerised laboratory environment with the help of an OO programming and DES modelling techniques; considering a single machining operation (milling) centre as a case study. Practical industrial environments and other operation centres were not addressed in this study due to several constraints such as time, finance and logistic problems. When the researched DSS is implemented in manufacturing environments, some challenges are anticipated to pass decisions with respect to the qualitative aspects of the DSS. These qualitative dimensions were treated in this study using various assumption. These expected challenges when the system is executed in industrial situations are discussed as the limitations of this study in this section.

In the case construction process, the selection of a few critical case attributes was done assuming that they were identified by human experts. However, it cannot be as simple as the situation presented in the numerical example in industrial systems. Knowledge elicitation may be challenging in order to identify the potential and key product attributes, which are useful to do a decision-based part/fixture assignment. Similarly, rating the importance of case attributes was carried out in subjective and judgemental manners using the fuzzy version of the AHP. In manufacturing environments, it can be difficult to elicit and represent the required knowledge and experiences from human experts to rate the case attributes.

Another important qualitative dimension at this stage was the conversion of linguistic terms into their equivalent fuzzy numbers and the estimation of their crisp values from these fuzzy numbers when the weights of case attributes were calculated. In the research, the conversion scales and fuzzy ranking method were proposed using the existing two theories (Section

3.3.2). In industrial situations, finding the correct conversion scales is challenging. The right conversion scales and fuzzy ranking approaches are required using knowledgeable and experienced human experts, which is difficult to address in industrial systems.

The determination of the threshold values for reuse/adaptations of the retrieved fixtures, building of new fixtures and accepting/rejecting the retrieved fixtures were decided based on the normalised weights of case attributes from the fuzzy AHP. It was assumed that the required experiences were acquired from experts as presented in Section 4.4.2. Unless the case attributes selection and evaluation of attributes are carried out in the right way, the threshold values can be vulnerable to faults in practical manufacturing situations.

When the decision of an adaptation was passed, which features of the retrieved fixture should be adapted was subjective and judgemental. The same was true during the evaluation of the current states of the retrieved fixtures, which features of the fixture should be evaluated was another qualitative and vague dimension. The problems need to be addressed by developing additional domain knowledge-based rules using the acquired knowledge from experts. Developing these guiding rules can be cumbersome in industrial situations.

Several input parameters such process time, setup time, process state cost and setup cost per unit time, decision-based operational costs of fixtures, etc. were estimated using different assumptions in the DES modelling process when a set of decision alternatives were treated as the discrete-events of the system as shown in Section 4.5.2. In manufacturing environments, estimating these input parameters can be very challenging unless knowledge is properly elicited from experienced human experts or well-designed expert systems.

In general, if these limitations are appropriately articulated in manufacturing environment, the proposed DSS can smoothly perform its intended tasks.

5.7 Summary

This chapter started with restating the problem statement of this research work. It thoroughly explained whether the findings of this study were consistent with the research problem, objectives and the current theories with respect to the methodological approaches. The aim of this study was discussed and compared with the results presented in Chapter 4. The specific objectives, which were outlined in Section 1.2, were elaborated with reference to the responses provided in the dedicated chapters of this thesis.

The methodological implications were explained by referring to the current theories in Chapter 2 and the findings in the previous chapter. It was implied that the methodological approaches were in line with the current literature in DSS studies. The newly synthesised methodological approach implemented in this study was discussed as depicted in Table 5-1 and Figure 5-1. This approach was used as a novel approach in order to determine the stable numbers of fixtures required in any manufacturing processes in planned production periods using fuzzy CBR and DES methodologies.

The contributions and limitations of the study were discussed in the last two sections. Firstly, the contributions of the entire study were deliberated with reference to the research contributions stated in Section 1.4. Secondly, the contributions of the individual publications listed in Section 1.5, were briefly explained. Lastly, the limitations of this study in real industrial situations were remarked.

CHAPTER 6

6. CONCLUSION AND FUTURE RESEARCH

This chapter consists of two sections. The first section recapitulates the findings and discussions of the entire study in the form of conclusions and the second section recommends future research works.

6.1 Conclusion

It was reviewed that DSS have passed several development stages and faced various challenges depending upon the innovation of driving technologies specifically in AI and DES technologies. DSS were initially created to support individual managers in making effective decisions. Presently, it has been furnished with sophisticated technologies in order to articulate complex and unstructured problems to meet business requirements in dynamic and uncertain situations. Embedding these advanced technologies in DSS was regarded as one of the opportunities to enhance the flexibility, effectiveness and efficiency of DSS in manufacturing processes.

It was found in the current DSS literature that research approaches integrating AI and DES technologies were widely studied in manufacturing environments. These approaches were utilised in several manufacturing operations such as shop floor control, parts scheduling and sequencing, FMS design, etc. Although they were widely accepted in manufacturing situations, some limitations were found in the current DSS research. Research gaps were identified that the current DSS research was unable to address other important dimensions of manufacturing systems such as integrated jigs, fixtures and tools planning and management problems. These dimensions of the current manufacturing systems were identified as one of the areas of original contribution potential to the current body of knowledge in DSS research. Specifically, the problems of a decision-based part/fixture assignment and fixture flow control were treated as the current research gap in DSS and problem statement to this study.

The problems of fixture assignment and flow control strategies were not adequately researched in the past. Most of the past DSS studies focused on the adaptations of prior fixture design cases instead of developing strategies for utilising physically available fixtures. According to the latest literature in DSS, systematic part/fixture assignment and fixture flow control approaches should have been developed to determine the stable number of fixtures required in specific production periods in manufacturing environments. This was required to reduce operational costs, and improve resources utilisations and on-time deliveries of product orders. This study researched and developed a DSS that was able to assist users to retrieve and enumerate the available fixtures, evaluate the current states of the retrieved fixtures, reuse and/or adapt the retrieved fixtures and manufacture new fixtures depending upon the demand of the system in consideration. This study found that the proposed DSS was a novel and promising approach to support human experts in complex decision-making situations. It was shown that it could increase the utilisation of the existing limited resources, reduce operational costs of fixtures, and improve MLT and delivery times of manufacturing processes using demand-driven decisions.

From the methodological perspective, CBR and DES systems were utilised as the principal research methods to construct the proposed DSS in this study. The AI subsystem of the DSS immensely used a fuzzy CBR system through integrating it with essential guiding domain knowledge-based rules (RBR systems). The reason why the CBR system was implemented as the driving element in the DSS was that the objectives of CBR approaches were mostly found consistent with the objectives of DSS. Both were intended to support human experts in unstructured situations instead of replacing them in important decision-making strategies. RBR, fuzzy set theory and MADM (the AHP) approaches were combined with the researched CBR system to retrieve the most similar case to the current product order from the required case library and propose the best decision alternative.

The fuzzy version of the CBR methodology was used to represent imprecise and uncertain knowledge in the case representation and retrieval, and decision proposal processes. The fuzzy AHP was also utilised for eliciting the knowledge and judgements of experts to prioritise the weights of case attributes. It was shown that the researched DSS was capable to analyse case attributes and their weights that were expressed in terms of qualitative linguistic terms. It was inferred this feature of the DSS was useful to utilise the judgements of human experts when adequate documented data were not available in manufacturing systems. In this way, the knowledge and experiences of experts could be well accommodated in the DSS development process rather than sticking on the knowledge of system developers alone.

The fuzzy CBR subsystem of the DSS was constructed using an OO case representation approach. The findings depicted that this representation approach in this study was flexible enough to incorporate fuzzy cases whose attribute were expressed using the combination of numerical values, nominal values, descriptive/symbolic values and fuzzy linguistic terms. Such kinds of unified representation techniques were able to emulate human thoughts and reasoning to process uncertain and imprecise knowledge in industrial situations. This implied that the proposed DSS was capable to process unstructured and incomplete knowledge in the form of fuzzy cases to articulate human reasoning and decision-making capabilities as much as possible in manufacturing situations.

The inverse of the weighted Euclidean distance, which was thought as one of the popular NN pattern matching functions, was applied for the case retrieval process using the outputs of the fuzzy case representation matrix and the normalised weights of case attributes from the fuzzy AHP. Several rules were developed to propose the best decision alternative based on the states of the retrieved fixtures and case similarity measures between the current and retrieved cases. Two case libraries were designed to determine the number of active fixtures in the simulated operation centre and propose what activities should be performed when the decision of an adaptation was passed. From this, it was concluded that the CBR subsystem of the researched DSS was designed to support human experts in complex situations, which were beyond the scope human experts, in line with the general objectives of DSS in the current DSS literature.

A DES model was applied to evaluate the performances of the solutions proposed by the fuzzy CBR subsystem of the developed DSS. The DES model was utilised to articulate the uncertainties and risks of the proposed solutions due to the lack of knowledge and experiences in CBR construction stages and weighting case attributes using the AHP. Three scenarios were simulated using three solutions proposed by the fuzzy CBR subsystem and the best solution was identified using the simulated performances in terms of three proposed KPIs. From this approach, it was implied that a number of performance scenarios could be generated as several proposed solutions (numbers of flowing fixtures). This could be done by varying the number of case attributes, the relative weights allocated to case attributes and the thresholds of decision sets such as reuse, adaptation and building a new fixture in the CBR subsystem and the combination of these factors.

This study was regarded as a significant contribution to the current knowledge in DSS research by proposing a strategy that combined fuzzy CBR and DES methodologies to solve such ill-defined problems in fixtures planning and management studies. The researched DSS was implemented to predict the near future situations of the proposed solutions instead of using historical data to validate its accuracy. This was a unique feature of the proposed DSS through combining fuzzy CBR and DES techniques in the problem domains of fixture planning strategies.

In order to validate the applicability of the proposed DSS, a numerical example was illustrated using a limited number of new cases and a few training samples. However, it was implied that the system was able to address any numbers of part order arrivals, prior cases and case attributes. This was because in industrial situations, product mix variations could be very high and a large number of fixtures could be required to accommodate these variations. In that case, fixtures must be retrieved and reused/adapted or manufactured, and supplied to the system when they are only required to improve the productivity of processes, utilisation of resources and delivery times of orders. In the numerical example, a milling operation was considered; however, the DSS could be applied to any manufacturing operations that need decision-based fixture assignment and control strategies using specific knowledge and experiences required at specific operation centres.

From the managerial perspective, it was shown that operational managers could plan fixtures in parallel to their part order plans and enumerate the available fixtures. It was implied that the researched DSS was capable to avoid the unnecessary holding, downtime, and fixture design and manufacture costs by stabilising the flows of fixtures during their planned production periods. This could reduce the unnecessary operational costs of fixtures and the wastages of limited resources. In this study, it was found that operational managers were able to improve the productivity and delivery times of their processes using the merits of integrating CBR and DES methodologies.

6.2 Future research

With reference to the limitations of this study, there are recommendations for future research. Implementing the proposed DSS in manufacturing environments can be the immediate future task to the researcher. In addition, the DSS should be implemented in several manufacturing operations to test and validate its flexibility. This future work can strongly improve the applicability and acceptance of the DSS and its limitations can be well articulated after these future works.

In order to improve the efficiency of the proposed system, the current version of the DSS should be upgraded into software level. This would help users to interact easily with the system with the help of a specific user interface system. To develop the required software package, collaborations with other disciplines, which are more dedicated in software engineering, or software-developing companies should be a substantial future work.

In the review of the literature, the dimensions of manufacturing systems, which have not been articulated in the current DSS studies, were identified. In this study, the problems of a part/fixture assignment and fixture flow control were solely researched. The problems of decision-based manufacturing tools and jigs assignment and flow control should be studied in the future to expand the applications of the researched DSS. The proposed DSS can be modified to address decision-based tools and jigs assignment and control problems.

The DSS in this study was designed to utilise the unstructured knowledge and experiences of its users or human experts at the case construction stages and weighting case attributes. In weighting case attributes, other AI technologies such as ANNs and GA can be integrated with the researched CBR system depending on the nature and demand of manufacturing processes in consideration. The major factors to integrate the fuzzy CBR subsystem of the proposed DSS and these AI technologies are the availability of historical data and level of the required automation. The integration of these AI technologies can be considered in the problem domain of this study in the future.

Appendix

A. Weighting the importance of secondary attributes

A.1 Physical features

Table A-1: Fuzzy reciprocal matrix of physical features

	Shape	Length	Width
Shape	1, 1, 1	1, 2, 3	2, 3, 4
Length	1/3, 1/2, 1	1, 1, 1	1, 1, 1
Width	1/3, 1/2, 1	1, 1, 1	1, 1, 1

Table A-2: Standard fuzzy numbers in (0, 1] for physical features

	Shape	Length	Width
Shape	0.10, 0.10, 0.10	0.10, 0.20, 0.30	0.20, 0.30, 0.40
Length	0.03, 0.05, 0.10	0.10, 0.10, 0.10	0.10, 0.10, 0.10
Width	0.03, 0.05, 0.10	0.10, 0.10, 0.10	0.10, 0.10, 0.10

Table A-3: Defuzzified numbers in (0, 1] for physical features

	Shape	Length	Width
Shape	0.1000	0.1849	0.2774
Length	0.0567	0.1000	0.1000
Width	0.0567	0.1000	0.1000
Column sum	0.2134	0.3849	0.4774

Table A-4: Normalised matrix of physical features

	Shape	Length	Width	W_i
Shape	0,4686	0.4804	0.5810	0.510
Length	0,2657	0.2598	0.2095	0.245
Width	0,2657	0.2598	0.2095	0.245
Sum				1.000

A.2 Operation types

Table A-5: Fuzzy reciprocal matrix of operation types

	End-mill	Plain-mill	Face-mill	Thread-cut	Gear-cut
End-mill	1, 1, 1	1, 1, 2	1, 1, 2	1, 1, 1	1, 1, 1
Plain-mill	1/2, 1, 1	1, 1, 1	1, 1, 1	1/2, 1, 1	1/2, 1, 1
Face-mill	1/2, 1, 1	1, 1, 1	1, 1, 1	1/2, 1, 1	1/2, 1, 1
Thread-cut	1, 1, 1	1, 1, 2	1, 1, 2	1, 1, 1	1, 1, 1
Gear-cut	1, 1, 1	1, 1, 2	1, 1, 2	1, 1, 1	1, 1, 1

Table A-6: Standard fuzzy numbers in (0, 1] for operation types

	End-mill	Plain-mill	Face-mill	Thread-cut	Gear-cut
End-mill	0.1, 0.1, 0.1	0.1, 0.1, 0.2	0.1, 0.1, 0.2	0.1, 0.1, 0.1	0.1, 0.1, 0.1
Plain-mill	0.05, 0.1, 0.1	0.1, 0.1, 0.1	0.1, 0.1, 0.1	0.05, 0.1, 0.1	0.05, 0.1, 0.1
Face-mill	0.05, 0.1, 0.1	0.1, 0.1, 0.1	0.1, 0.1, 0.1	0.05, 0.1, 0.1	0.05, 0.1, 0.1
Thread-cut	0.1, 0.1, 0.1	0.1, 0.1, 0.2	0.1, 0.1, 0.2	0.1, 0.1, 0.1	0.1, 0.1, 0.1
Gear-cut	0.1, 0.1, 0.1	0.1, 0.1, 0.2	0.1, 0.1, 0.2	0.1, 0.1, 0.1	0.1, 0.1, 0.1

Table A-7: Defuzzified numbers in (0, 1] for operation types

	End-mill	Plain-mill	Face-mill	Thread-cut	Gear-cut
End-mill	0.1000	0.1191	0.11905	0.10000	0.10000
Plain-mill	0.0854	0.1000	0.10000	0.08537	0.08537
Face-mill	0.0854	0.1000	0.10000	0.08537	0.08537
Thread-cut	0.1000	0.1191	0.11905	0.10000	0.10000
Gear-cut	0.1000	0.1191	0.11905	0.10000	0.10000
Column sum	0.4707	0.5571	0.55714	0.47073	0.47073

Table A-8: Normalised matrix of operation types

	End-mill	Plain-mill	Face-mill	Thread-cut	Gear-cut	W_i
End-mill	0.2124	0.2137	0.2137	0.2124	0.2124	0.2129
Plain-mill	0.1813	0.1795	0.1795	0.1813	0.1813	0.1806
Face-mill	0.1813	0.1795	0.1795	0.1813	0.1813	0.1806
Thread-cut	0.2124	0.2137	0.2137	0.2124	0.2124	0.2129
Gear-cut	0.2124	0.2137	0.2137	0.2124	0.2124	0.2129

A.3 Process requirements

Table A-9: Fuzzy reciprocal matrix of process requirements

	Material type	Machinability	Tolerance	Surface
Material type	1, 1, 1	1, 1, 2	1, 2, 3	1, 2, 3
Machinability	1/2, 1, 1	1, 1, 1	1, 2, 3	1, 2, 3
Tolerance	1/3, 1/2, 1	1/3, 1/2, 1	1, 1, 1	1, 1, 2
Surface	1/3, 1/2, 1	1/3, 1/2, 1	1/2, 1, 1	1, 1, 1

Table A-10: Standard fuzzy numbers in (0, 1] for process requirements

	Material type	Machinability	Tolerance	Surface
Material type	0.1, 0.1, 0,1	0.1, 0.1, 0.2	0.1, 0.2, 0.3	0.1, 0.2, 0.3
Machinability	0.05, 0.1, 0.1	0.1, 0.1, 0.1	0.1, 0.2, 0.3	0.1, 0.2, 0.3
Tolerance	0.033, 0.05, 0.1	0.033, 0.05, 0.1	0.1, 0.1, 0.1	0.1, 0.1, 0.2
Surface	0.033, 0.05, 0.1	0.033, 0.05, 0.1	0.05, 0.1, 0.1	0.1, 0.1, 0.1

Table A-11: Defuzzified numbers in (0, 1] for process requirements

	Material type	Machinability	Tolerance	Surface
Material type	0.1000	0.1190	0.1849	0.1849
Machinability	0.0854	0.1000	0.1849	0.1849
Tolerance	0.0567	0.0567	0.1000	0.1190
Surface	0.0567	0.0567	0.0854	0.1000
Column sum	0.2988	0.3324	0.5552	0.5888

Table A-12: Normalised matrix of process requirements

	Material type	Machinability	Tolerance	Surface	W_i
Material type	0.3347	0.3581	0.3331	0.3140	0.3350
Machinability	0.2857	0.3008	0.3331	0.3140	0.3084
Tolerance	0.1898	0.1706	0.1801	0.2022	0.1856
Surface	0.1898	0.1706	0.1538	0.1698	0.1710

B. Constructors and data fields of fuzzy CBR subsystem

```
import java.util.*;

public class PartOrder implements Cloneable
{
//Data fields
	private String partName;
// Attributes
	private String shape;
	private String material;
```

```
    private double[] machinability = new double[4];
    private double length;
    private double width;
    private double[] surface = new double[4]; //surface finish or roughness or texture
    private double tolerance;
    private int endMill;
    private int plainMill;
    private int faceMill;
    private int threadCut;
    private int gearCut;

    private int processTime;
    private static int numberOfParts = 0;
    private java.util.Date dateCreated;
    public String fixtureCode;

    private double[] stateOfRetrievedFixture  = new double[4];

//Lower and upper limits for numerical attributes (lL=lower limit and uL=upper limit)
    private double lLLength = 400.0;
    private double lLWidth = 150.0;
    private double lLTolerance = 2.0;
    private double uLLength = 1200.0;
    private double uLWidth = 500.0;
    private double uLTolerance = 10.0;
    //Attribute weights
    private double wtShape;
    private double wtMaterial;
    private double wtMachinability;
    private double wtLength;
    private double wtWidth;
    private double wtSurface;
    private double wtTolerance;
    private double wtEndMill;
    private double wtPlainMill;
    private double wtFaceMill;
    private double wtThreadCut;
    private double wtGearCut;

    private double[] attributeWeight = new double[12];

    //Cost of fixture
```

```
//Cost of reuse decision, which incurs only setup cost (setup downtime and setup overhead)
        private double costOfReuse;

//cost of adaptation decision which includes machine downtime, process overhead, material and setup
        private double[] costOfAdaptation = new double[4];

//Cost of manufacture decision, which includes design, process overhead, material, setup and storage
        private double[] costOfManufacture = new double[5];

//Constructors

public PartOrder()      //no-arg constructor
        {
                this("","","", new double[]{0,0,0,0}, new double[]{0,0,0,0}, 400, 150,   2, 0,0,0,0,0,0,"");
                numberOfParts++;
                dateCreated = new java.util.Date();
        }

public PartOrder(String partName, String shape, String material, double[] newMachinability, double[]
                newSurface, double newLength, double newWidth, double newTolerance, int newEndMill,
                int newPlainMill, int newFaceMill, int newThreadCut, int newGearCut, int
                newProcessTime, String fixtureCode)   // main constructor
        {
                this.partName = partName;
                this.shape = shape;
                this.material = material;
                setMachinability(newMachinability);
                setSurface(newSurface);
                setLength(newLength);
                setWidth(newWidth);
                setTolerance(newTolerance);
                setEndMill(newEndMill);
                setPlainMill(newPlainMill);
                setFaceMill(newFaceMill);
                setThreadCut(newThreadCut);
                setGearCut(newGearCut);
                setProcessTime(newProcessTime);
                this.fixtureCode = fixtureCode;
                numberOfParts++;
        }
```

```
public PartOrder(String partName, String shape, String material, double[] machinability, double[] surface, double
            length, double width, double tolerance, int endMill, int plainMill, int faceMill,int threadCut, int
            gearCut, int processTime)      //a constructor without fixture
this(partName,shape,material,machinability,surface,length,width,tolerance,
        endMill,plainMill,faceMill,threadCut,gearCut,processTime,"");
                    numberOfParts++;
        }
//Several accessible, mutable, instance, static and Java in-built library methods are included here.
}
```

C. Sample inputs and outputs of fuzzy CBR

These inputs and outputs were managed using NetBeans IDE for Java as user interface.

The Solution Created on Thu May 18 11:00:44 EAT 2017
Please enter the number of parts planned during your production period: 16
Please enter the normalised weight of each attribute from FAHP:
0.242 0.083 0.077 0.046 0.117 0.116 0.042 0.059 0.050 0.050 0.059 0.059
General information:
The minimum similarity value (lower bound) is 0.6567071221245626
The medium similarity value is 0.8283535610622813
The threshold to reuse the retrieved fixture is 0.9141767805311407
The threshold to adapt the retrieved fixture is 0.7425303415934219

For new part-order 1:
Please enter the name, shape and material type of the new order:
P1 Rectangular carbonSteel
Please enter the fuzzy machinability and surface smoothness values of the new order.
0.7 0.8 0.9 1.0; 0.8 0.9 0.9 1.0
Please enter the other attribute values of the new order.
585 290 8 1 1 1 0 0 12

The absolute difference between the new order and TS1 is:
{TS1-P1:
1,1,[0.39999999999999997,0.4,0.5,0.5],[0.10000000000000009,0.09999999999999998,0.09999999999999998,0.09999999999999998],0.7,0.17714285714285713,0.0,0,1,0,1,0,8}
The similarity value of the new part order and TS1 is 0.7159113275330277

The absolute difference between the new order and TS2 is:
{TS2-P1:
0,0,[0.10000000000000009,0.09999999999999998,0.0,0.0],[0.20000000000000007,0.20000000000000007,0.09999999999999998,0.09999999999999998],0.20625,0.17142857142857143,0.75,0,0,0,0,0,0}
The similarity value of the new part order and TS2 is 0.9520787557447173

The absolute difference between the new order and TS3 is:

{TS3-P1:
1,1,[0.6,0.6000000000000001,0.6000000000000001,0.6],[0.5,0.5,0.5,0.5],0.45625,0.45714285714285713,0.125,0,1,1,1,0,3}
The similarity value of the new part order and TS2 is 0.712738186075285

The best similarity value to this new case is 0.9520787557447173, which is retrieved case, TS2.
The most similar retrieved fixture is Fix201.
Please evaluate and enter the state of the retrieved fixture: 1 1 1 1
The retrieved fixture is in functional state and case similarity is acceptable for reuse.
Decision: Reuse this fixture.
The retrieved fixture's code to be reused is Fix201
The number of cases in the first case library is 3
The number of cases in the second case library is 1
The size of similarity list is 3
The fixture is available in the store. Retrieve and use it.
The number of fixtures in the database is 3

For new part-order 2:
Please enter the name, shape and material type of the new order.
P2 Cylindrical alloySteel
Please enter the fuzzy machinability and surface smoothness values of the new order.
0.5 0.6 0.6 0.7 0.6 0.7 0.8 0.9
Please enter the other attribute values of the new order.
410 155 9 1 0 1 1 0 15

The absolute difference between the new order and TS1 is:
{TS1-P2:
0,0,[0.2,0.19999999999999996,0.19999999999999996,0.19999999999999996],[0.09999999999999998,0.10000000000000009,0.0,0.0],0.91875,0.20857142857142857,0.125,0,0,0,0,0,5}
The similarity value of the new part order and TS1 is 0.8876079144491962

The absolute difference between the new order and TS2 is:
{TS2-P2:
1,1,[0.30000000000000004,0.30000000000000004,0.30000000000000004,0.30000000000000004],[0.0,0.0,0.0,0.0],0.0125,0.5571428571428572,0.875,0,1,0,1,0,-3}
The similarity value of the new part order and TS2 is 0.7212750394185193

The absolute difference between the new order and TS3 is:
{TS3-P2:
1,1,[0.4,0.39999999999999997,0.3,0.29999999999999993],[0.3,0.29999999999999993,0.4,0.4],0.675,0.8428571428571429,0.0,0,0,1,0,0,0}
The similarity value of the new part order and TS3 is 0.707759196761136

The best similarity value to this new case is 0.8876079144491962, which is retrieved case, TS1.

The most similar retrieved fixture is Fix101.

Please evaluate and enter the state of the retrieved fixture: 0.9 1 1 1

The retrieved fixture is in functional state and case similarity is acceptable for adaptation.

Decision: Adapt and use.

The retrieved fixture's code to be adapted is Fix101

The number of cases in the first case library is 3

The number of cases in the second case library is 2

The size of similarity list is 3

The fixture is available in the store. Retrieve and use it.

The number of fixtures in the database is 3

For new part-order 3:

Please enter the name, shape and material type of the new order.

P3 Hexagonal castIron

Please enter the fuzzy machinability and surface smoothness values of the new order.

0.2 0.3 0.4 0.5 0.4 0.5 0.5 0.6

Please enter the other attribute values of the new order.

1000 500 8 1 0 0 1 0 6

The absolute difference between the new order and TS1 is:

{TS1-P3:

1,1,[0.09999999999999998,0.10000000000000003,0.0,0.0],[0.29999999999999993,0.30000000000000004,0.30000000000000004,0.30000000000000004],0.18125,0.7771428571428571,0.0,0,0,1,0,0,14}

The similarity value of the new part order and TS1 is 0.7229163714866896

The absolute difference between the new order and TS2 is:

{TS2-P3:

1,1,[0.6000000000000001,0.6000000000000001,0.5,0.5],[0.19999999999999996,0.19999999999999996,0.30000000000000004,0.30000000000000004],0.725,0.42857142857142855,0.75,0,1,1,1,0,6}

The similarity value of the new part order and TS2 is 0.7038533332353308

The absolute difference between the new order and TS3 is:

{TS3-P3:

0,0,[0.1,0.09999999999999998,0.10000000000000003,0.09999999999999998],[0.10000000000000003,0.09999999999999998,0.09999999999999998,0.09999999999999998],0.0625,0.14285714285714285,0.125,0,0,0,0,0,9}

The similarity value of the new part order and TS3 is 0.9791170547779304

The best similarity value to this new case is 0.9791170547779304, which is retrieved case, TS3.

The most similar retrieved fixture is Fix302

Please evaluate and enter the state of the retrieved fixture: 0.7 0.8 0.8 0.9

The retrieved fixture is not in functional state.

Please enter a newly manufactured fixture code: Fix305

Decision: Manufacture a new fixture.

It has been removed from the case base for permanent edit/discard.

The manufactured fixture's code to be assigned is Fix305

The number of cases in the first case library is 3

The number of cases in the second case library is 2

The size of similarity list is 3

It is on the process. Wait and assign it.

The number of fixtures in the database is 4

D. DES custom code

D.1 Milling centre process time per unit product

```
/**Custom Code*/
treenode current = ownerobject(c);
treenode item = parnode(1);

if (getitemtype (item) == 1) return normal(15, 3.0, 0);
if (getitemtype (item) == 2) return normal(12, 2.5, 0);
if (getitemtype (item) == 3) return normal(18, 3.5, 0);
if (getitemtype (item) == 4) return normal(20, 4.0, 0);
if (getitemtype (item) == 5) return normal(27, 5.0, 0);
if (getitemtype (item) == 6) return normal(16, 3.0, 0);
if (getitemtype (item) == 7) return normal(12, 2.0, 0);
if (getitemtype (item) == 8) return normal(30, 5.0, 0);
if (getitemtype (item) == 9) return normal(33, 5.5, 0);
if (getitemtype (item) == 10) return normal(28, 4.5, 0);
if (getitemtype (item) == 11) return normal(15, 3.0, 0);
if (getitemtype (item) == 12) return normal(13, 2.5, 0);
if (getitemtype (item) == 13) return normal(11, 2.0, 0);
if (getitemtype (item) == 14) return normal(19, 4.0, 0);
if (getitemtype (item) == 15) return normal(31, 5.0, 0);

else return normal(12, 2.0 ,0);
```

D.2 Milling Centre setup time per unit product

a) First scenario

```
/**Custom Code*/
treenode current = ownerobject(c);
treenode item = parnode(1);
int port = parval(2);


if (getitemtype (item) == 1) return exponential(0, 3.0, 0);
if (getitemtype (item) == 2) return exponential(0, 3.5, 0);
if (getitemtype (item) == 3) return exponential(0, 2.3, 0);
if (getitemtype (item) == 4) return exponential(0, 4.0, 0);
if (getitemtype (item) == 5) return exponential(0, 3.5, 0);
if (getitemtype (item) == 6) return exponential(0, 2.0, 0);
if (getitemtype (item) == 7) return exponential(0, 5.5, 0);
if (getitemtype (item) == 8) return exponential(0, 5.5, 0);
```

```
if (getitemtype (item) == 9) return exponential(0, 3.0, 0);
if (getitemtype (item) == 10) return exponential(0, 3.3, 0);
if (getitemtype (item) == 11) return exponential(0, 2.0, 0);
if (getitemtype (item) == 12) return exponential(0, 5.0, 0);
if (getitemtype (item) == 13) return exponential(0, 3.1, 0);
if (getitemtype (item) == 14) return exponential(0, 5.0, 0);
if (getitemtype (item) == 15) return exponential(0, 6.0, 0);

else return exponential(0, 3.5, 0);
```

b) Second scenario

```
/**Custom Code*/
treenode current = ownerobject(c);
treenode item = parnode(1);
int port = parval(2);

if (getitemtype (item) == 1) return exponential(0, 3.2, 0);
if (getitemtype (item) == 2) return exponential(0, 3.5, 0);
if (getitemtype (item) == 3) return exponential(0, 2.3, 0);
if (getitemtype (item) == 4) return exponential(0, 2.7, 0);
if (getitemtype (item) == 5) return exponential(0, 3.5, 0);
if (getitemtype (item) == 6) return exponential(0, 2.0, 0);
if (getitemtype (item) == 7) return exponential(0, 2.2, 0);
if (getitemtype (item) == 8) return exponential(0, 5.5, 0);
if (getitemtype (item) == 9) return exponential(0, 3.0, 0);
if (getitemtype (item) == 10) return exponential(0, 3.5, 0);
if (getitemtype (item) == 11) return exponential(0, 2.0, 0);
if (getitemtype (item) == 12) return exponential(0, 5.0, 0);
if (getitemtype (item) == 13) return exponential(0, 3.1, 0);
if (getitemtype (item) == 14) return exponential(0, 2.3, 0);
if (getitemtype (item) == 15) return exponential(0, 8.0, 0);

else return exponential(0, 4.0, 0);
```

c) Third scenario

```
/**Custom Code*/
treenode current = ownerobject(c);
treenode item = parnode(1);
int port = parval(2);

if (getitemtype (item) == 1) return exponential(0, 3.0, 0);
if (getitemtype (item) == 2) return exponential(0, 3.5, 0);
```

```
if (getitemtype (item) == 3) return exponential(0, 2.3, 0);
if (getitemtype (item) == 4) return exponential(0, 4.0, 0);
if (getitemtype (item) == 5) return exponential(0, 8.0, 0);
if (getitemtype (item) == 6) return exponential(0, 2.0, 0);
if (getitemtype (item) == 7) return exponential(0, 5.5, 0);
if (getitemtype (item) == 8) return exponential(0, 8.5, 0);
if (getitemtype (item) == 9) return exponential(0, 4.0, 0);
if (getitemtype (item) == 10) return exponential(0, 7.0, 0);
if (getitemtype (item) == 11) return exponential(0, 2.0, 0);
if (getitemtype (item) == 12) return exponential(0, 5.0, 0);
if (getitemtype (item) == 13) return exponential(0, 6.5, 0);
if (getitemtype (item) == 14) return exponential(0, 5.6, 0);
if (getitemtype (item) == 15) return exponential(0, 15.0, 0);

else return exponential(0, 7.5, 0);
```

E. DES cost estimation

Table E-1: Estimated State Time costs

State	Processing	Setup	Idle	Blocked	Waiting
$/time	2.00	6.00	15.00	10.00	8.00

Flowitems Fixed cost = operational costs of fixtures per every decision set.

Table E-2: Estimated Flowitems costs for the first scenario

Part	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
$/batch	50	110	230	145	232	48	147	235	147	48	47	150	240	135	52	68
$/entry	1.5	3.0	5.2	3.8	8.0	1.0	3.2	13.0	3.0	1.6	1.0	3.0	7.5	3.0	4.0	2.5
$/time					0.015			0.024					0.036			

Table E-3: Estimated Flowitems costs for the second scenario

Part	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
$/batch	50	110	230	228	232	48	230	235	147	48	47	210	240	225	52	68
$/entry	1.5	3.0	5.2	6.0	8.0	1.0	5.0	13.0	3.0	1.6	1.0	4.2	7.5	5.0	4.0	2.5
$/time				0.017	0.015		0.024	0.024					0.036	0.015		

Table E-4: Estimated Flowitems costs for the third scenario

Part	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
$/batch	50	93	230	145	160	48	147	162	147	150	47	125	150	150	117	135
$/entry	1.5	2.5	5.2	3.8	5.5	1.0	3.2	9.0	3.0	5.0	1.0	2.5	4.7	3.3	9.0	5.0

Bibliography

[1] AAMODT, A. Explanation-Driven Case-Based Reasoning. In *Topics in Case-Based Reasoning.* Springer-Verlag, Berlin and Heidelberg, 1994, 274-288.

[2] AAMODT, A., AND PLAZA, E. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications 7*, 1 (1994), 39-59.

[3] ALDURGHAM, M. M., AND BARGHASH, M. A. A generalised framework for simulation-based decision support for manufacturing. *Production Planning & Control: The Management of Operations 19*, 5 (2008), 518-534.

[4] ALI, S. A., AND SEIFODDINI, H. Simulation intelligence and modelling for manufacturing uncertainties. In *Proceedings of the 2006 Winter Simulation Conference,* IEEE, Piscataway, NJ, 2006, 1920-1928.

[5] ALTER, S. Why is man-computer interaction important for decision support systems? *Interfaces 7*, 2 (1977), 109-115.

[6] ALTER, S. L. *Decision Support Systems: Current Practice and Continuing Challenges*. Addison-Wesley Publishing, Reading, Massachusetts, 1980.

[7] ALTER, S. L. Decision support systems for manufacturing. *Data Processing* 25, 10 (1983), 37-38.

[8] AN, S.-H., KIM, G.-H., AND KANG, K.-I. A case-based reasoning cost estimating model using experience by analytic hierarchy process. *Building and Environment 42*, 7 (2007), 2573-2579.

[9] ANDERSSON, M., AND OLSSON, G. A simulation based decision support approach for operational capacity planning in a customer order driven assembly line. In *Proceedings of the 1998 Winter Simulation Conference.* IEEE, Piscataway, New Jersey, 1998, 935-941.

[10] ANGEHRN, A. A., AND LÜTHI, H. J. Intelligent decision support systems: A visual interactive approach. *Interfaces 20*, 6 (1990), 17-28.

[11] ANTHONY, R. N. *Planning and Control Systems: A Framework for Analysis.* Cambridge, MA, Harvard University Graduate School of Business Administration, 1965.

[12] ARNOTT, D., AND PERVAN, G. A critical analysis of decision support systems research. *Journal of Information Technology 20*, 2 (2005), 67-87.

[13] BAAS, S. M., AND KWAKERNAAK, H. Rating and ranking of multiple-aspect alternatives using fuzzy sets. *Automatica 13*, 1 (1977), 47-58.

[14] BARRON, F. H., AND BARRETT, B. E. Decision quality using ranked attribute weights. *Management Science 42*, 11 (1996), 1515-1523.

[15] BEEMER, B. A., AND GREGG, D. G. Advisory systems to support decision making. In *Handbook on Decision Support Systems 1: Basic Themes.* Springer-Verlag, Berlin and Heidelberg, 2008, 511-528.

[16] BELLMAN, R. E., AND ZADEH, L. A. Decision-making in a fuzzy environment. *Management Science 17*, 4 (1970), 141-164.

[17] BENZ, R., AND MERTENS, P. Combining knowledge-based systems and simulation to solve rescheduling problems. *Decision Support Systems 17*, 2 (1996), 141-157.

[18] BERGMANN, R., KOLODNER, J., AND PLAZA, E. Representation in case-based reasoning. *The Knowledge Engineering Review 20*, 3 (2006), 209-213.

[19] BERGMANN, R., AND STAHL, A. Similarity Measures for Object-Oriented Case Representations. In *Advances in Case-Based Reasoning.* Springer-Verlag, Berlin and Heidelberg, 1998, 25-36.

[20] BI, Z. M., AND ZHANG, W. J. Flexible fixture design and automation: Review, issues and future directions. *International Journal of Production Research 39*, 13 (2001), 2867-2894.

[21] BONCZEK, R. H., HOLSAPPLE, C. W. AND WHINSTON, A. B. *Foundation of Decision Support Systems,* Academic Press, 1981.

[22] BOYLE, I., RONG, Y., AND BROWN, D. C. CAFixD: a case-based reasoning fixture design method: framework and indexing mechanisms. *Journal of Computing and Information Science in Engineering 6*, 1 (2006), 40-48.

[23] BOYLE, I., RONG, Y., AND BROWN, D. C. 2011. A review and analysis of current computer-aided fixture design approaches. *Robotics and Computer-Integrated Manufacturing 27*, 1 (2011), 1-12.

[24] BRUNELLI, M., AND MEZEI, J. How different are ranking methods for fuzzy numbers? A numerical study. *International Journal of Approximate Reasoning 54,* 5 (2013), 627-639.

[25] BUCKLEY, J. J. Fuzzy hierarchical analysis. *Fuzzy Sets and Systems 17*, 3 (1985), 233-247.

[26] CARLSSON, C., AND FULLÉR, R. Fuzzy multiple criteria decision making: Recent developments. *Fuzzy Sets and Systems 78*, 2 (1996), 139-153.

[27] CHAN, F. T. S., AND CHAN, H. K. A comprehensive survey and future trend of simulation study on FMS scheduling. *Journal of Intelligent Manufacturing 15*, 1 (2004), 87-102.

[28] CHAN, F. T. S., JIANG, B., AND TANG, N. K. H. 2000. The development of intelligent decision support tools to aid the design of flexible manufacturing systems *International Journal of Production Economics 65*, 1 (2000), 73-84.

[29] CHANG, P.-C., LIU, C.-H., AND LAI, R. K. A fuzzy case-based reasoning model for sales forecasting in print circuit board industries. *Expert Systems with Applications 34*, 3 (2008), 2049-2058.

[30] CHANGCHIEN, S. W., AND LIN, M.-C. Design and implementation of a case-based reasoning system for marketing plans. *Expert Systems with Applications 28*, 1 (2005), 43-53.

[31] CHEN, G., CHEN, J., ZHAO, Z., AND RUAN, X. Y. An object-oriented hierarchical case representation of automotive panels in a computer-aided process planning system. *The International Journal of Advanced Manufacturing Technology 26*, 11/12 (2004), 1323-1330.

[32] CHEN, S.-J., AND HWANG, C.-L. *Fuzzy Multiple Attribute Decision Making: Methods and Applications.* Springer-Verlag, Berlin Heidelberg, 1992.

[33] CHEN, S.-M., AND CHEN, J.-H. Fuzzy risk analysis based on ranking generalized fuzzy numbers with different heights and different spreads. *Expert Systems with Applications 36*, 3 (2009), 6833-6842.

[34] CHI, R. T. H., AND KIANG, M. Y. An integrated approach of rule-based and case-based reasoning for decision support. In *Proceedings of the 19th annual conference on Computer Science*, ACM, New York, NY, 1991, 255-267.

[35] CHIOUA, H.-K., TZENG, G.-H., AND CHENG, D.-C. Evaluating sustainable fishing development strategies using fuzzy MCDM approach. *Omega: The international Journal of management science 33*, 3 (2005), 223-234.

[36] CHUANG, T., AND YADAV, S. B. The development of an adaptive decision support system. *Decision Support Systems 24*, 2 (1998), 73-87.

[37] CHURCHMAN, C. W., AND ACKOFF, R. L. An approximate measure of value. *Journal of the operations Research Society of America 2*, 2 (1954), 172-187.

[38] CONTEH, N. W., AND FORGIONNE, G. A just-in-time approach to intelligent decision support systems. *Journal of Decision Systems 14*, 1/2 (2005), 39-54.

[39] DE MANTARAS, R. L. Case-Based Reasoning. In *Machine Learning and Its Applications.* Springer-Verlag, Berlin and Heidelberg, 2001, 127-145.

[40] DE MÁNTARAS, R. L., AND PLAZA, E. Case-based reasoning: An overview. *AI Communications 10*, 1 (1997), 21-29.

[41] DEMIREL, T., DEMIREL, N. C., AND KAHRAMAN, C. Fuzzy analytic hierarchy process and its application. In *Fuzzy Multi-criteria Decision Making: Theory and Applications with Recent Developments.* Springer, New York, NY, 2008, 53-83.

[42] DUBOIS, D., PRADE, H., ESTEVA, F., GARCIA, P., GODO, L., AND DE MÀNTARAS, R. L. Fuzzy set modelling in case-based reasoning. *International Journal of Intelligent Systems 13*, 4 (1998), 345-373.

[43] DUTTA, S., AND BONISSONE, P. P. Integrating case- and rule based reasoning. *International Journal of Approximate Reasoning 8*, 3 (1993), 163-203.

[44] EDWARDS, W. How to use multi-attribute utility measurement for social decision making. *IEEE Transactions on Systems, Man and Cybernetics SMC-7* (1977), 326-340.

[45] EDWARDS, W., AND BARRON, F. H. SMARTS and SMARTER: Improved simple methods for multi-attribute utility measurement. *Organizational Behavior and Human Decision Processes 60*, 3 (1994), 306-325.

[46] EOM, H. B., AND LEE, M. S. A survey of decision support system applications (1971-April 1988). *Interfaces 20*, 3 (1990), 65-79.

[47] EOM, S. B., LEE, S. M., KIM, E. B., AND SOMARAJAN, C. Survey of decision support system applications (1988-1994). *The Journal of the Operational Research Society 49*, 2 (1998), 109-120.

[48] ER, M. C. 1988. Decision support systems: A summary, problems, and future trends. *Decision Support Systems 4*, 3 (1988), 355-3363.

[49] FAEZ, F., GHODSYPOUR, S. H., AND O'BRIEN, C. Vendor selection and order allocation using an integrated fuzzy case-based reasoning and mathematical programming model. *International Journal of Production Economics 121*, 2 (2009), 395-408.

[50] FEKI, J., BEN MESSAOUD, I., AND ZURFLUH, G. Building an XML document warehouse. *Journal of Decision Systems 22,* 2 (2013), 122-148.

[51] FENG, S., LI, L. X., AND CEN, L. An object-oriented design tool to aid the design of manufacturing systems. *Knowledge-based systems 14,* 5/6 (2001), 225-232.

[52] FORGIONNE, G. A. An Architecture for the Integration of Decision Making Support Systems Functionalities. In *Decision-Making Systems: Achievements, Trends and Challenges for the New Decade*. Idea Group Publishing, Hershey, PA, 2003.

[53] FORMAN, E. H., AND GASS, S. I. The analytic hierarchy process: An exposition. *Operations Research 49,* 4 (2001), 469-486.

[54] GERRITY, T. P. Design of man-machine decision systems: An application to portfolio management. *Sloan Management Review 12*, 2 (1971), 59-75.

[55] GOLDING, A. R., AND ROSENBLOOM, P. S. Improving accuracy by combining rule-based and case-based reasoning. *Artificial Intelligence 97,* 1/2 (1996), 215-254.

[56] GORRY, G. A., AND MORTON, M. S. A framework for management information systems. *Sloan Management Review 13,* 1 (1971), 21-36.

[57] GOUL, M., HENDERSON, J., AND TONGE, F. The emergence of artificial intelligence as a reference discipline for decision support systems. *Decision Sciences 11*, 2 (1992), 1273-1276.

[58] HACKATHORN, R. D., AND KEEN, P. G. W. Organizational strategies for personal computing in decision support systems. *MIS Quarterly 5,* 3 (1981), 21-27.

[59] HEJAZI, S. R., DOOSTPARAST, A., AND HOSSEINI, S. M. An improved fuzzy risk analysis based on a new similarity measures of generalized fuzzy numbers. *Expert Systems with Applications 38*, 8 (2011), 9179-9185.

[60] HO, W. Integrated analytic hierarchy process and its applications – A literature review. *European Journal of Operational Research 186*, 1 (2008), 211-228.

[61] HOLSAPPLE, C. W., AND WHINSTON, A. B. *Decision Support Systems: A knowledge-based Approach*. West Publishing Company, 1996.

[62] IASSINOVSKI, S., ARTIBA , A., BACHELET, V., AND RIANE, F. Integration of simulation and optimization for solving complex decision making problems. *International Journal of Production Economics 85*, 1 (2003), 3-10.

[63] ISHIZAKA, A., AND LABIB, A. Review of the main developments in the analytic hierarchy process. *Expert Systems with Applications 38*, 11 (2011), 14336-14345.

[64] JAHANGIRIAN, M., ELDABI, T., NASEER, A., STERGIOULAS, L. K., AND YOUNG, T. Simulation in manufacturing and business: A review. *European Journal of Operational Research 203*, 1 (2010), 1-13.

[65] JANG, J.-S. R., SUN, C.-T., AND MIZUTANI, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, Upper Saddle River, NJ, 1997.

[66] KAHRAMAN, C. Multi-criteria decision making methods and fuzzy sets. In *Fuzzy Multi-criteria Decision Making: Theory and Applications with Recent Developments*. Springer, New York, NY, 2008, 1-18.

[67] KAHRAMAN, C., BIRGÜN, S., AND YENEN, V.Z. Fuzzy Multi-Attribute Scoring Methods with Applications. In *Fuzzy multi-criteria decision making: Theory and applications with recent developments*. Springer, New York, NY, 2008, 187-208.

[68] KEEN, P. G. W. Adaptive design for decision support systems. *Database, Sloan School of Management Review 12*, 1/2 (1980), 15-25.

[69] KEEN, P. G. W., AND MORTON, M. S. *Decision Support Systems: An Organizational Perspective*, Addison-Wesley Publishing, Reading, MA, 1978.

[70] KHANUM, A., MUFTI, M., JAVED, M. Y., AND SHAFIQ, M. Z. 2009. Fuzzy case-based reasoning for facial expression recognition. *Fuzzy Sets and Systems 160*, 2 (2009), 231-250.

[71] KOLODNER, J. Improving human decision making through case-based decision aiding. *AI Magazine 12*, 2 (1991), 52-68.

[72] KOLODNER, J. L. An Introduction to Case-Based Reasoning. *Artificial Intelligence Review 6*, 1 (1992), 3-34.

[73] KOLODNER, J. L. *Case-Based Reasoning, Morgan Kaufmann Publishers* San Francisco, CA, 1993.

[74] KUMAR, K. S., AND PAULRAJ, G. Analysis and optimization of fixture under dynamic machining condition with chip removal effect. *Journal of Intelligent Manufacturing 25*, 1 (2014), 85-98.

[75] KUO, T. C. Combination of case-based reasoning and analytical hierarchy process for providing intelligent decision support for product recycling strategies. *Expert Systems with Applications 37*, 8 (2010), 5558-5563.

[76] LEE, A. H. I., CHEN, W.-C., AND CHANG, C.-J. A fuzzy AHP and BSC approach for evaluating performance of IT department in the manufacturing industry in Taiwan. *Expert Systems with Applications 34*, 1 (2008), 96-107.

[77] LI, S.-T., AND HO, H.-F. Predicting financial activity with evolutionary fuzzy case-based reasoning. *Expert Systems with Applications 36*, 1 (2009), 411-422.

[78] LI, W., LI, P., AND RONG, Y. Case-based agile fixture design. *Journal of Materials Processing Technology 128*, 1-3 (2002), 7-18.

[79] LIANG, Y. D. 2015. *Introduction to Java Programming: Comprehensive Version.* Pearson Education, Upper Saddle River, NJ, 2015.

[80] LIAO , T. W., ZHANG, Z., AND MOUNT, C. R. Similarity measures for retrieval in case-based reasoning systems. *Applied Artificial Intelligence: An International Journal 12*, 4 (1998), 267-288.

[81] LIRAVIASL, K. K, ELMARAGHY, H., HANAFY, M., AND SAMY, S.N. A framework for modelling reconfigurable manufacturing systems using hybridized discrete-event and agent-based simulation. *IFAC-PapersOnLine 48*, 3 (2015), 1490-1495

[82] LITTLE, J. D. C. BRANDAID: A marketing-mix model (parts 1 & 2). *Operations Research 23*, 4 (1975), 628-673.

[83] LIU, S., DUFFY, A. H. B., WHITFIELD, R. I., AND BOYLE, I. M. Integration of decision support systems to improve decision support performance. *Knowledge and Information Systems 22,* 3 (2009), 261-286.

[84] MAHDAVI, I., SHIRAZI, B., AND SOLIMANPUR, M. Development of a simulation-based decision support system for controlling stochastic flexible job shop manufacturing systems. *Simulation Modelling Practice and Theory 18,* 6 (2010), 768-786.

[85] MARDANI, A., JUSOH, A., AND ZAVADSKAS, E. K. Fuzzy multiple criteria decision-making techniques and applications – Two decades review from 1994 to 2014. *Expert Systems with Applications 42,* 8 (2015), 4126-4148.

[86] MARLING, C., SQALLI, M., RISSLAND, E., MUÑOZ-AVILA, H., AND AHA, D. 2002. Case-cased reasoning integrations. *AI Magazine 23*, 1 (2002), 369-86.

[87] METAXIOTIS, K., PSARRAS, J., AND SAMOUILIDIS, E. Integrating fuzzy logic into decision support systems: current research and future prospects. *Information Management & Computer Security 11,* 2 (2003), 53-59.

[88] NEMATI, H. R., STEIGER, D. M., IYER, L. S., AND HERSCHEL, R. T. Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence and data warehousing. *Decision Support Systems 33*, 2 (2002), 143-161.

[89] NICHOLS, J., AND GOUL, M. Synergizing the artificial intelligence and decision support research streams: Over a decade of progress with new challenges on the horizon. In *Intelligent Decision-making Support Systems: Foundations, Applications and Challenges.* Springer-Verlag, London, 2006, 45-55.

[90] O'KEEFE, R. M., AND ROACH, J. W. Artificial intelligence approaches to simulation. *The Journal of the Operational Research Society 38,* 8 (1987), 713-722.

[91] OH, K. J., AND KIM, T. Y. Financial market monitoring by case-based reasoning. *Expert Systems with Applications 32,* 3 (2007), 789-800.

[92] OSTOJIC, G., STANKOVSKI, S., VUKELIC, D., LAZAREVIC, M., HODOLIC, J., TADIC, B., AND ODRI, S. Implementation of Automatic Identification Technology in a Process of Fixture Assembly/Disassembly. *Strojniški vestnik – Journal of Mechanical Engineering 57*, 11 (2011), 819-825.

[93] ÖZBAYRAK, M., AND BELL, R. A knowledge-based decision support system for the management of parts and tools in FMS. *Decision Support Systems 35*, 4 (2003), 487-515.

[94] PAL, S. K., AND SHIU, S. C. K. *Foundations of soft case-based reasoning*. John Wiley & Sons, Hoboken, New Jersey, 2004.

[95] PARK, C.-S., AND HAN, I. A case-based reasoning with the feature weights derived by analytic hierarchy process for bankruptcy prediction. *Expert Systems with Applications 23*, 3 (2002), 255-264.

[96] PEHRSSON, L., AMOS, H. C., NG, A. H. L., AND STOCKTON, D. Industrial cost modeling and multi-objective optimisation for decision support in production systems development. *Computers & Industrial Engineering 66*, 4 (2013), 1036-1048

[97] PENG, G., CHEN, G., WU, C., XIN, H., AND JIANG, Y. Applying RBR and CBR to develop a VR based integrated system for machining fixture design. *Expert Systems with Applications 38*, 1 (2011), 26-38.

[98] POWER, D. J. *Decision Support Systems: Concepts and Resources for Managers*. Greenwood Publishing Group, Westport, CT, 2002.

[99] POWER, D. J. Specifying an expanded framework for classifying and describing decision support systems. *Communications of the Association for Information Systems 13*, 1 (2004), 158-166.

[100] POWER, D. J. Decision support systems: A historical overview. In *Handbook on Decision Support Systems 1: Basic Themes*. Springer-Verlag, Berlin and Heidelberg, 2008, 121-140.

[101] POWER, D. J., AND SHARDA, R. Model-driven decision support systems: Concepts and research directions. *Decision Support Systems 43*, 3 (2007), 1044-1061.

[102] PÖYHÖNEN, M., AND HÄMÄLÄINEN, R. P. On the convergence of multiattribute weighting methods. *European Journal of Operational Research 129*, 3 (2001), 569-585.

[103] PRENTZAS, J., AND HATZILYGEROUDIS, I. Categorizing approaches combining rule-based and case-based reasoning. *Expert Systems 24*, 2 (2007), 97-122.

[104] RAHIMIFARD, S., AND NEWMAN, S. T. Simultaneous scheduling of workpieces, fixtures and cutting tools within flexible machining cells. *International Journal of Production Research 35*, 9 (1997), 2379-2396.

[105] RAHIMIFARD, S., AND NEWMAN, S. T. A reactive multi-flow approach to the planning and control of flexible machining facilities. *International Journal of Computer Integrated Manufacturing 13*, 4 (2000), 311-323.

[106] RAPHAEL, B., AND KUMAR, B. Object oriented representation of design cases. *Computers & Structures 63*, 4 (1997), 663-668.

[107] RIBEIRO, R. A. Fuzzy multiple attribute decision making: A review and new preference elicitation techniques. *Fuzzy Sets and Systems 78*, 2 (1996), 155-181.

[108] ROGERS, P., AND GORDEN, R. J. Simulation for real-time decision making in manufacturing systems. In *Proceedings of the 1993 Winter simulation conference*. IEEE, Piscataway, New Jersey, 1993, 866-874.

[109] SAATY, T. L. *The Analytic Hierarchy Process*. McGraw-Hill, New York, NY, 1980.

[110] SAATY, T. L. How to make a decision: The analytic hierarchy process. *Interfaces 24*, 6 (1994), 19-43.

[111] SAATY, T. L. Decision-making with the AHP: Why is the principal eigenvector necessary. *European Journal of Operational Research 145*, 1 (2003), 85-91.

[112] SCHANK, R. *Dynamic memory: A theory of reminding and learning in computers and people*. Cambridge University Press, New York, NY, 1982.

[113] SHAFER, S. M., AND SMUNT, T. L. Empirical simulation studies in operations management: Context, trends, and research opportunities. *Journal of Operations Management 22*, 4 (2004), 345-354.

[114] SHIM, J. P., WARKENTIN, M., COURTNEY, J. F., POWER, D. J., SHARDA, R., AND CARLSSON, C. Past, present, and future of decision support technology. *Decision Support Systems 33*, 2 (2002), 111-126.

[115] SIMON, H. A. *The new science of management decision*. Harper Brothers, New York, 1960.

[116] SLONIMA, T. Y., AND SCHNEIDER, M. Design issues in fuzzy case-based reasoning. *Fuzzy Sets and Systems 117*, 3 (2001), 251-267.

[117] SMITH, J. S. Survey on the use of simulation for manufacturing system design and operation. *Journal of Manufacturing Systems 22*, 2 (2003), 157-171.

[118] SPRAGUE, R. H. A framework for the development of decision support systems. *MIS Quarterly 4*, 4 (1980), 1-26.

[119] SPRAGUE, R. H. J., AND CARLSON, E. D. *Building Effective Decision Support Systems*. Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[120] STECKE, K. E. Design, planning, scheduling, and control problems of flexible manufacturing systems. *Annals of Operations Research 3*, 1 (1985), 1-12.

[121] SUN, S. H., AND CHEN, J. L. A fixture design system using case-based reasoning. *Engng Applic. Artif. Intell. 9*, 5 (1996), 533-540.

[122] SURI, R., AND WHITNEY, C. K. Decision support requirements in flexible manufacturing. *Journal of Manufacturing Systems 3*, 1 (1984), 61-69.

[123] TAN, K. H., LIM, C. P., PLATTS, K., AND KOAY, H. S. An intelligent decision support system for manufacturing technology investments. *International Journal of Production Economics 104*, 1 (2006), 179-190.

[124] THIERAUF, R. J. *Decision Support Systems for Effective Planning and Control: A Case Study Approach*, Prentice-Hall, Upper Saddle River, NJ, 1982.

[125] TSENG, F. S. C., AND CHOU, A. Y. H. The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence. *Decision Support Systems 42*, 2 (2006), 727-744.

[126] TURBAN, E., ARONSON, L. E., AND LIANG, T. P. *Decision Support Systems and Intelligent Systems*. Prentice-Hall, Upper Saddle River, NJ, 2005.

[127] TURBAN, E., AND WATKINS, P. R. 1986. Integrating expert systems and decision support systems. *MIS Quarterly 10*, 2 (1986), 121-136.

[128] VAIDYA, O. S., AND KUMAR, S. Analytic hierarchy process: An overview of applications. *European Journal of Operational Research 169*, 1 (2006), 1-29.

[129] VAN LAARHOVEN, P. J. M., AND PEDRYCZ, W. A fuzzy extension of Saaty's priority theory. *Fuzzy Sets and Systems 11*, 3 (1983), 229-241.

[130] VON WINTERFELDT, D., AND EDWARDS, W. *Decision Analysis and Behavioral Research*. Cambridge University Press, Cambridge, 1986.

[131] VÖLKNER, P., AND WERNERS, B. A decision support system for business process planning. *European Journal of Operational Research 125*, 3 (2000), 633-647.

[132] VÖLKNER, P., AND WERNERS, B. A simulation-based decision support system for business process planning. *Fuzzy Sets and Systems 125*, 3 (2002), 275-287.

[133] WANG, H., AND RONG, Y. Case based reasoning method for computer aided welding fixture design. *Computer-Aided Design 40*, 12 (2008), 1121-1132.

[134] WANG, H., RONG, Y., LI, H., AND SHAUN, P. Computer aided fixture design: Recent research and trends. *Computer-Aided Design 42*, 12 (2010), 1085-1094.

[135] WATSON, I. Case-based reasoning is a methodology not a technology. *Knowledge-Based Systems 12,* 5-6 (1999), 303-308.

[136] WATSON, I., AND MARIR, F. Case based reasoning: A review. *The Knowledge Engineering Review 9,* 4 (1994), 327-354.

[137] WEBER, M., AND BORCHERDING, K. Behavioral influences on weight judgments in multiattribute decision making. *European Journal of Operational Research 67*, 1 (1993), 1-12.

[138] WIND, Y., AND SAATY, T. L. Marketing applications of the analytic hierarchy Process. *Management Science 26*, 7 (1980), 641-658.

[139] WU, M.-C., LO, Y.-F. & HSU, S.-H. 2008. A fuzzy CBR technique for generating product ideas. *Expert Systems with Applications,* 34, 530-540.

[140] XU, L., LI, Z., LI, S., AND TANG, F. A decision support system for product design in concurrent engineering. *Decision Support Systems 42,* 4 (2007), 2029-2042.

[141] ZAHEDI, F. The analytic hierarchy process: A survey of the method and its applications. *Interfaces 16*, 4 (1986), 96-108.

[142] ZEDAH, L. A. Fuzzy sets. *Information and control 8*, 3 (1965), 338-353.

[143] ZHOU, Y., LI, Y., AND WANG, W. A feature-based fixture design methodology for the manufacturing of aircraft structural parts. *Robotics and Computer-Integrated Manufacturing 27*, 6 (2011), 986-993.

[144] ZIMMERMANN, H. J. *Fuzzy Set Theory—and Its Applications.* Springer, New York, NY, 2001.

[145] ZÜLCH, G., AND BECKER, M. A simulation-supported approach for man-machine configuration in manufacturing. *International Journal of Production Economics 125*, 1 (2010), 41-51.