

2000

## **A Neuro-Expert Approach for Decision -Making in Welding Environment.**

Vivek Goel

*Louisiana State University and Agricultural & Mechanical College*

Follow this and additional works at: [https://digitalcommons.lsu.edu/gradschool\\_disstheses](https://digitalcommons.lsu.edu/gradschool_disstheses)

---

### **Recommended Citation**

Goel, Vivek, "A Neuro-Expert Approach for Decision -Making in Welding Environment." (2000). *LSU Historical Dissertations and Theses*. 7198.

[https://digitalcommons.lsu.edu/gradschool\\_disstheses/7198](https://digitalcommons.lsu.edu/gradschool_disstheses/7198)

This Dissertation is brought to you for free and open access by the Graduate School at LSU Digital Commons. It has been accepted for inclusion in LSU Historical Dissertations and Theses by an authorized administrator of LSU Digital Commons. For more information, please contact [gradetd@lsu.edu](mailto:gradetd@lsu.edu).

## **INFORMATION TO USERS**

**This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.**

**The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.**

**In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.**

**Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.**

**Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.**

**Bell & Howell Information and Learning  
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA  
800-521-0600**

**UMI<sup>®</sup>**



**A NEURO-EXPERT APPROACH FOR DECISION-MAKING  
IN WELDING ENVIRONMENT**

**A Dissertation**

**Submitted to the Graduate Faculty of the  
Louisiana State University and  
Agricultural and Mechanical College  
in partial fulfillment of the  
requirements for the degree of  
Doctor of Philosophy**

**in**

**The Interdepartmental Program in Engineering Science**

**by**

**Vivek Goel**

**B.E., University of Roorkee, India, 1989**

**M.E., University of Roorkee, India, 1991**

**M.S., Louisiana State University, 1995**

**May, 2000**



**UMI Number: 9979262**



---

**UMI Microform 9979262**

**Copyright 2000 by Bell & Howell Information and Learning Company.**

**All rights reserved. This microform edition is protected against  
unauthorized copying under Title 17, United States Code.**

---

**Bell & Howell Information and Learning Company  
300 North Zeeb Road  
P.O. Box 1346  
Ann Arbor, MI 48106-1346**

## **ACKNOWLEDGMENTS**

I would like to express my deep sense of gratitude to the Chairman, Dr. Warren T. Liao for his valuable guidance and generous help at all stages of this study. His meticulous cares in going through the manuscript and valuable suggestions are greatly acknowledged.

I also wish to thank Dr. A. Raman, Professor, Mechanical Engineering department, Dr. Bush Jones, Associate Professor, Computer Science, and Dr. Jianhua Chen, Associate Professor, Computer Science for being on my committee and for their guidance and help at all stages of my dissertation. I wish to acknowledge the help provided by Dr. S. R. Gupta of University of Roorkee in going through and correcting the expert system rule-base.

Finally, I owe it to my family and especially my parents for their encouragement which gave me the courage and confidence to materialize my dreams.

# TABLE OF CONTENTS

<b>ACKNOWLEDGMENTS.....</b>	<b>ii</b>
<b>LIST OF TABLES.....</b>	<b>viii</b>
<b>LIST OF FIGURES.....</b>	<b>ix</b>
<b>ABSTRACT .....</b>	<b>xi</b>
<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1 Welding.....	1
1.2 Decision-Making in Welding .....	2
1.2.1 Planning in Welding .....	2
1.2.1.1 Selection of the Process.....	3
1.2.1.2 Selection of Process Parameters.....	4
1.2.1.3 Weld Procedure Specification (WPS).....	8
1.2.1.4 Economic Evaluation .....	8
1.2.1.5 Defect Analysis.....	8
1.2.1.6 Trouble Shooting .....	9
1.2.2 Execution and Control in Welding.....	9
1.2.3 Quality Evaluation in Welding.....	10
1.2.3.1 Nondestructive Testing.....	10
1.2.3.2 Destructive Testing.....	11
<b>2. LITERATURE AND SOFTWARE REVIEW.....</b>	<b>12</b>
2.1 Expert Systems.....	13
2.1.1 Expert Systems for Planning in Welding .....	14
2.1.2 Expert Systems for Execution and Control in Welding.....	14
2.1.3 Expert Systems in Quality Evaluation.....	15
2.1.4 Limitations of Expert Systems .....	15
2.2 Neural Networks .....	15
2.2.1 Neural Networks in Planning for Welding.....	16
2.2.2 Neural Networks in Execution and Control of Welding.....	17
2.2.3 Neural Networks for Quality Evaluation in Welding .....	17
2.2.4 Neural Network Training.....	18
2.2.5 Pruning Approaches in Neural Networks.....	18
2.3 Neuro-Expert Systems .....	19
2.3.1 Divide and Conquer Neuro-Expert Systems .....	21
2.3.2 Embedded Neuro-Expert Systems.....	21
2.3.3 Explanation by Confabulation .....	21
2.3.4 Artificial Expert .....	22
2.3.4.1 Coding Rules in Neural Network Structures and Learning Algorithms .....	23

2.3.4.2 Knowledge Extraction in Artificial Expert Systems.....	24
2.3.4.2.1 Rule Format Criteria for Rule Extraction Classification.....	24
2.3.4.2.2 Quality Criteria for Rule Extraction Classification .....	25
2.3.4.2.3 Translucency Criteria for Rule Extraction Classification .....	25
2.3.4.2.4 Algorithmic Complexity Criteria for Rule Extraction Classification ...	26
2.3.4.2.5 Portability Criteria for Rule Extraction Classification .....	26
2.3.5 Coupled Neuro-Expert System.....	26
2.3.6 Neuro-Expert Systems in Welding.....	27
2.3.7 Verification and Validation of Neuro-Expert Systems .....	28
2.4 Fuzzy Systems.....	29
2.5 Fuzzy Neural Networks .....	30
2.6 Other Approaches for Control in Welding.....	31
2.7 Other Software in Welding.....	32
<b>3. OBJECTIVES OF THE RESEARCH .....</b>	<b>33</b>
<b>4. NEURO EXPERT SYSTEM DESIGN AND SPECIFICATIONS .....</b>	<b>36</b>
4.1 Process Selector Design.....	40
4.1.1 Process Selector Expert System .....	41
4.1.1.1 Data Entry .....	42
4.1.1.2 Data Conversion .....	43
4.1.1.3 Weight Assignment and Rule Mapping to Neural Network .....	43
4.1.1.4 Ranking of Welding Process .....	45
4.1.2 Process Selector Neural Network.....	47
4.2 Parameter Selector .....	52
4.2.1 Knowledge Representation for the Parameter Selector .....	53
4.3 Economic Evaluator.....	53
4.3.1 Economic Evaluator Cost Computation.....	54
4.4 Defect Analyzer Design - A Coupled System.....	55
4.4.1 Knowledge Representation and Categorization for Defect Analyzer.....	55
4.4.2 Defect Analyzer Neural Network.....	58
4.4.2.1 Initial Weight Assignment in Defect Analyzer Neural Network.....	61
4.4.2.2 Network Training .....	62
4.4.2.3 Weight Modification and Knowledge Extraction.....	62
4.4.3 Rule-Based Parameter Adjustment Strategy in Defect Analyzer .....	62
4.4.3.1 Design of the Parameter Adjustment Module of Defect Analyzer.....	65
4.4.3.2 Design of the WPS Tips Generator Module of Defect Analyzer .....	67
4.4.4 Interactive Defect Analyzer .....	68
4.5 Trouble Shooter .....	71
4.6 WPS Generator .....	72
<b>5. WELDING PROCESS SELECTOR - AN ARTIFICIAL EXPERT .....</b>	<b>73</b>
5.1 Process Selector Neural Network Training.....	73
5.1.1 Analysis of initial results .....	73

5.1.2 Pruning of the Neural Network based on Ineffectual Criteria .....	74
5.1.3 Effect of Pruning on the Neural Network Performance.....	80
5.2 Knowledge Extraction and Analysis .....	82
5.3 Knowledge Correction .....	93
5.4 Neuro-Expert System Results.....	94
5.4.1 Effect of the cut-off value in the modified Fu's Algorithm .....	94
5.4.2 Advantages of using Validation Step in Process Selector .....	98
5.4.3 Advantages of using the Neuro-Expert Approach .....	98
5.5 Validation of Results .....	103
5.6 Comparison of Process Selector to other commercially available Software.....	104
5.7 Summary of Process Selector .....	104
<b>6. PARAMETER SELECTOR AND ECONOMIC EVALUATOR.....</b>	<b>105</b>
<b>7. DEFECT ANALYZER – A COUPLED NEURO-EXPERT SYSTEM</b>	
<b>APPROACH .....</b>	<b>109</b>
7.1 A Neural Network for Defect Prediction .....	111
7.1.1. Network Initialization .....	111
7.1.2 Network training .....	111
7.1.3 Weight Modification in trained Neural Network .....	113
7.2 Expert System for Parameter Adjustment and WPS Tips Generation .....	115
7.2.1 Parameter Adjustment Module.....	117
7.2.2 WPS Tips Generator Module .....	119
7.3 Interactive Defect Analyzer .....	119
7.4 Analysis of Results .....	120
<b>8. TROUBLE SHOOTER - A DIAGNOSTIC EXPERT SYSTEM .....</b>	<b>121</b>
<b>9. WELDING PROCEDURE SPECIFICATION GENERATOR - TYING IT</b>	
<b>ALL TOGETHER .....</b>	<b>123</b>
<b>10. CONCLUSIONS .....</b>	<b>127</b>
10.1 Contributions of this Research .....	127
10.2 Advantages of using the Neuro Expert System.....	128
10.3 Salient Features of the Neuro-Expert System .....	128
10.4 Limitations of the Neuro-Expert System.....	129
10.5 Capabilities of the Neuro-Expert System.....	129
10.5.1 Automation of Decision Making in Welding.....	129
10.5.2 Optimal Selection of Process.....	130
10.5.3 Adjustment of Process Parameters and Welding Procedure to minimize Defects.....	130
10.5.4 Economic Evaluation.....	130
10.5.5 Defect Analysis .....	130
10.5.6 Trouble Shooting .....	130

10.7 Scope of the Neuro-Expert System .....	131
10.8 Validation of the Neuro-Expert System .....	131
10.9 Future Scope of Research .....	131
<b>REFERENCES.....</b>	<b>134</b>
<b>INTERNET WEB-SITE REFERENCES.....</b>	<b>144</b>
<b>GLOSSARY.....</b>	<b>145</b>
<b>APPENDIX .....</b>	<b>146</b>
<b>APPENDIX A - EXPERT SYSTEM RULES.....</b>	<b>147</b>
A.1 Conversion of Quantitative Job Variables to Qualitative Values in Process Selector.....	147
A.2 Validation Rules for Expert System for Process Selection.....	148
A.3 SMAW ELECTRODE SELECTION RULES .....	149
A.4 RULES FOR EXPERT SYSTEM FOR TROUBLESHOOTING.....	150
A.4.1 Welding Set Related Troubles.....	150
A.4.2 Arc Related Troubles .....	151
A.4.3 Wire Feeder Related Troubles .....	151
A.4.4 Flux Feeder Related Troubles.....	152
A.4.5 Miscellaneous Troubles .....	153
A.5 RULES FOR EXPERT SYSTEM FOR DEFECT ANALYSIS .....	154
A.5.1 Defect Analysis Rules for Shielded Metal Arc Welding (SMAW).....	154
A.5.2 Defect Analysis Rules for Gas Metal Arc Welding (GMAW).....	155
A.5.3 Gas Tungsten Arc Welding (GTAW) .....	156
A.5.4 Submerged Arc Welding (GTAW) .....	158
A.6 Sample Rules For WPS Tips Generation Mode of Defect Analyzer.....	159
A.6.1 Shielded Metal Arc Welding (SMAW).....	159
A.7 Sample Rules for the WPS Generator .....	160
<b>APPENDIX B - NEURO-EXPERT SYSTEM PSEUDO CODE.....</b>	<b>162</b>
B.1 Program Listing (Pseudo Code).....	162
B.1.1 Process Selector.....	162
B.1.2 Parameter Selection:.....	165
B.1.3 Defect Analyzer:.....	166
<b>APPENDIX C - TRAINING AND TESTING DATA .....</b>	<b>168</b>
C.1 Training & Testing Data for Process Selector Neural Network in Spreadsheet.....	168
C.2 Weight Change comparison for pruned and unpruned networks for Process Selector .....	170
C.3 Training Data for Defect Analyzer Data.....	174

<b>APPENDIX D - LISTING OF KNOWLEDGE BASES IN PROLOG .....</b>	<b>175</b>
D.1 CRITERIA.DBA file for storing criteria of Process Selector .....	175
D.2 ELECTROD.DBA file for storing electrode selection rules of Parameter Selector .....	175
D.3 FIRMDAT.DBA file for storing cost data for Economic Evaluator .....	176
D.4 STD_DATA.DBA file for storing Standard Data for Parameter Selector .....	177
<b>APPENDIX E - LISTING OF PROLOG PROGRAMS.....</b>	<b>188</b>
E.1 PROCESS.PRO - Process Selector.....	188
E.2 PAR_SEL.PRO - Program for Selecting Process Parameters.....	203
E.3 ECONEVAL.PRO - Prolog Program for Economic Evaluation of SMAW and GMAW .....	208
E.4 ADJUSTEN.PRO - Prolog Program for Knowledge Correction Task in Expert Network.....	213
E.5 WPS_GEN.PRO - Prolog Program for generating the Welding Procedure Specification.....	223
E.6 DEF_ANAL.PRO - Prolog Program for Defect Analyzer .....	231
E.7 TROUB_SH.PRO - Prolog Program for Trouble Shooting.....	259
E.8 INTERFAC.PRO - Prolog Program for User Interface.....	273
E.9 STD_DATA.PRO - Prolog Program for Standard Data Entry .....	280
E.10 Job_Data.pro - Prolog Program for Job Data Entry .....	288
<b>APPENDIX F - COMPARISON OF RULE EXTRACTION METHODS .....</b>	<b>292</b>
F.1 PROCESS SELECTOR - Comparison of various methods.....	292
<b>VITA .....</b>	<b>295</b>

## **LIST OF TABLES**

1.1 A Typical Welding Procedure Specification .....	7
4.1 Current Specifications of the Neuro-Expert System .....	37
4.2 Initial Design Parameters for the Process Selector Neural Network .....	47
4.3 Derivation of Initial Weight Matrix from Expert System Rules for First Layer .....	50
4.4 Initial Weight Matrix for the Process Parameter Neural Network for Second Layer.....	51
4.5 Electrode Classification for Defect Analyzer .....	57
4.6 Initial Weight Matrix for SMAW Defect Analysis from Expert Knowledge .....	63
4.7 Initial Weight Matrix for GMAW Defect Analysis from Expert Knowledge.....	64
4.8 Parameter Adjustment Priority List for Each Defect in SMAW .....	69
4.9 Parameter Adjustment Priority List for Each Defect in GMAW.....	70
5.1 Process Selector Results for 14 Job Samples. ....	75
5.2 Initialization Matrix for Pruned Neural Network.....	76
5.3 Comparison of Pruning vs. Unpruned Neural Network.....	79
5.4 Weight Changes in the Unpruned Neural Network after Training .....	90
5.5 Weight Changes in Pruned Neural Network .....	92
5.6 New Criteria Weights using Fu's Method and Modified Fu's Method.....	96
5.7 Effects of Expert System Knowledge Updates Using Neural Network Weights.....	99
5.8 Confidence Factors for Different Weight Modification Methods for the 14 Samples.....	101
5.9 Process Selector Results Using N-Fold Validation.....	103
7.1 Weight Modifications After Training the Neural Network.....	116
9.1 Overall Welding Procedure Specification for SMAW of Sample 8.....	125



## **LIST OF FIGURES**

4.1 Overall System Architecture and Information Flow .....	36
4.2: Neuro-Expert System Detailed Architecture & Information Flow .....	38
4.3 Neuro-Expert approaches for various modules of the System.....	39
4.4 Flowchart for the Process Selector .....	42
4.5 Expert System Rules Initial Mapping to Neural Network .....	44
4.6 Expert System Rules Mapping to Neural Network after training.....	44
4.7 Example of Overall Expert System Rule for SMAW .....	45
4.8 Example of Overall Positive Expert System Rule for SMAW .....	46
4.9 Structure of the Neural Network for Process Selection .....	51
4.10 Flowchart for the Parameter Selector, Economic Evaluator, and Defect Analyzer .....	52
4.11 Detailed Data Flow in the Defect Analyzer.....	56
4.12 Neural Network Configuration for SMAW Defect Analyzer .....	59
4.13 Neural Network Configuration for GMAW Defect Analyzer .....	60
4.14 Rule-Based Parameter Adjustment Strategy in Defect Analyzer.....	66
5.1 Pruning of Input Nodes in Process Selector NN based on Weight Modification.....	78
5.2 Pruned Process Selector NN .....	79
6.1 Information Flow in Parameter Selector and Economic Evaluator .....	105
6.2 Parameter Selector Results for Job Sample 3 of Table 5.1 .....	106
6.3 Parameter Selector Results for Job Sample 8 of Table 5.1 .....	107
6.4 Economic Evaluator for Job Sample 3 .....	107

<b>6.5 Economic Evaluator Results for Job Sample 8 .....</b>	<b>108</b>
<b>7.1 Parameter Adjustment Session for Sample 8 of Table 5.1.....</b>	<b>118</b>
<b>7.2 Automated Tips Generation in Defect Analyzer for Sample 8 (Table 5.1) .....</b>	<b>119</b>
<b>7.3 Interactive Defect Analyzer Sample Run .....</b>	<b>120</b>
<b>8.1 Interactive Trouble Shooting Sample Screens.....</b>	<b>121</b>
<b>8.2 Automated WPS Trouble Shooting Tips Sample Screen .....</b>	<b>122</b>
<b>9.1 Welding Procedure Specification Run for SMAW of Sample 8 .....</b>	<b>123</b>
<b>9.2 Welding Procedure Specification Sample Run for GMAW of Sample 8 .....</b>	<b>124</b>
<b>9.3 Modified Welding Procedure Specification for SMAW of Sample 8 .....</b>	<b>124</b>

## **ABSTRACT**

Decision making in welding is very important for achieving a good quality welded joint for the least possible cost. Of particular interest is decision making involving the selection of process, parameters, weld procedure specification, defect analysis and trouble shooting.

This research has provided a means of capturing the planning knowledge in a Neuro-Expert System in a form that is capable of learning new information, correcting old information and automating the decision-making process in a welding environment. A strategy is formulated for the representation of knowledge in the form of a neural links and the translation of rules into neural link weights. After training those weights were converted back into rules to find out the inconsistent rules and capture new rules using a new approach.

The various job variables affecting the process of welding are identified in detail and a Neuro-Expert system for the selection of process, parameters and weld procedure specification is developed. The neural networks are integrated with an expert system for decision making in welding environment. Apart from providing the initial parameters of welding, the expert system is used to validate the output of the neural network and served as a user-friendly interface for the neural network.

Defect Analysis is performed in welding domain by mapping the welding parameters and defect patterns in a neural network. A neural network based approach for representing the knowledge in expert system is utilized for this purpose as the modification and updating of the knowledge was easier.

# **1. INTRODUCTION**

## **1.1 Welding**

Welding is one of the most popular fabrication processes used very frequently. It is used in almost all industries, e.g. automobile, aircraft, ships, electronic equipment, machinery, household appliances etc. Welding joins different metals/alloys with the help of a number of processes in which heat is supplied either electrically or by means of a gas torch. In general, welding has been classified into gas welding, arc welding, resistance welding, solid state welding, thermo-chemical welding and radiant energy welding processes [Giachino and Weeks (1976)]. This study is restricted to arc welding.

Arc welding is a group of processes wherein coalescence is produced by heating with an electric arc or arcs, with or without the use of filler metal depending upon the base plate thickness. The main arc welding processes are:

- |                                      |                        |
|--------------------------------------|------------------------|
| 1. Carbon-arc welding                | 6. Electroslag welding |
| 2. Shielded-metal arc welding (SMAW) | 7. Electrogas welding  |
| 3. Submerged arc welding (SAW)       | 8. Plasma arc welding  |
| 4. Gas tungsten arc welding (GTAW)   | 9. Arc spot welding    |
| 5. Gas metal arc welding (GMAW)      | 10. Stud (arc) welding |

The SMAW, SAW, GTAW and GMAW processes are used most frequently. The scope of this dissertation was limited to these processes as standard data is available only for these processes.

Arc welding is widely used for joining metals because it is more economical, faster, less noisy, produces stronger joints and has many other similar advantages compared to

riveting and casting. The use of welding in industry is extensive and its growth has been faster than the general industrial growth.

## **1.2 Decision making in welding**

Decision-making in welding can be categorized into three phases: planning, execution and control, and quality evaluation. Planning in welding involves the selection of the process, the process parameters, and welding procedure specification. Execution and control involves the control of parameters during welding to minimize the defects. Quality evaluation is the inspection of welding job after the welding has been performed and the determination whether the job is acceptable or not. This study has focused on the planning phase of decision making.

### **1.2.1 Planning in welding**

There is no precise method for selecting process, parameters, and welding procedure specification for a particular job to be welded. Also, defect analysis and trouble-shooting involve some decision-making on the part of the welding engineer.

Accurate planning in welding is extremely important because of the following reasons:

1. Failure of welds resulting from improper decision-making lead to loss of life and property.
2. Welding process and parameters have a profound effect on the quality of welding and their proper selection is very important to the welding industry.
3. There is a lack of a comprehensive resource for the selection of process, parameters and techniques based on the job variables and the type of industry and so sometimes improper procedures and techniques are followed for welding resulting in poor welds.

4. Welding skills and knowledge are acquired with practice and novice welders may not always produce proper welds. There is a need for a system that will acquire, store and modify weld knowledge and help the welders produce consistent weld quality each time.

Sometimes, the inter-relationship between different planning decisions to be made in welding may have an effect on the overall quality. The process selected may have a bearing on the welding parameters which in turn decide the weld procedure specification. Similarly, defect analysis helps in optimization of welding parameters resulting in better quality welds. It is difficult to keep track of all these inter-relationships between the various decisions and an automated system avoids these drawbacks.

#### **1.2.1.1 Selection of the process**

The selection of the process is based on the experience, thumb rules and heuristics. Also the selection of a process is affected by economics of the process, criticality of application and many such other factors [Goel and Singh (1992), Goel<sup>1</sup> et al. (1993), Goel (1991), Goel<sup>2</sup> et al.(1993)]. Some of the job variables affecting the selection of a welding process [The Procedure Handbook of Arc Welding (1973)] are as follows:

- |                             |                               |                                      |
|-----------------------------|-------------------------------|--------------------------------------|
| 1. Production volume.       | 6. Ability to follow the seam | 11. Cleanliness requirements         |
| 2. Seam length              | 7. Set up time required.      | 12. Operator skill                   |
| 3. Equipment/tooling costs. | 8. Fixture costs              | 13. House keeping cost.              |
| 4. Ranges of weld sizes.    | 9. Material costs.            | 14. Flexibility in joint preparation |
| 5. Application flexibility  | 10. Fit up flexibility.       |                                      |

### 1.2.1.2 Selection of Process Parameters

Some of the job variables that affect the selection of process parameters are as follows:

1. Parent Material
2. Type of joint
3. Position of Welding
4. Shape of the welding job.
5. Tensile Strength
6. Bead Shape and Appearance
7. Thickness of Plates

The process parameters [Nadkarni (1988)] in welding depend on the type of the welding process. For e.g. different types of electrodes are available for SMAW, GMAW and SAW, while electrode in GTAW is non-consumable. A Filler Rod is utilized in GTAW for depositing Weld Metal. Other arc welding process parameters include:

1. **Current (amperage):** Is the actual electron movement through the conductor. The heat of the arc increases with the current. The polarity of the current affects the location of the heat. With Direct Current Reverse Polarity (DCRP) (also called Direct Current electrode positive (DCEP)), about two thirds of the heat produced is at the electrode tip. This happens because the electrons, which are traveling from negative to positive, are bombarding and decelerating on the electrode tip. This energy turns into heat energy.

On Direct Current straight polarity (DCSP) (also called Direct Current electrode positive(DCEP)), the electrode is negative and the plate is positive. This causes the electrons to flow from the negative electrode to the positive plate where they bombard and decelerate causing most of the heat to be produced at this part of the weld and in the weld.

In AC supply unit the current alternates back and forth between straight and reverse polarity. Heat balance with AC is one-half at the plate and one half at the electrode. Changing polarity may produce a pronounced effect on the penetration, the way the electrode operates, and even the way the arc sounds. With DCRP a good, deep penetration with a biting arc is observed while with DCSP less penetration, a softer arc and a different sounding arc is found. With DCSP, more heat is located at the plate and much less at the electrode.

The various considerations in determining current are listed as follows:

- a) Heat of arc is directly proportional to Amperage
- b) DC reverse polarity results in  $\frac{2}{3}$  of the heat to be concentrated on the electrode tip. Thus, DCRP leads to higher rate of weld deposit and more penetration.
- c) DC Straight Polarity causes two-thirds of the heat to be concentrated on the plate leading to a lower rate of weld deposit and less penetration
- d) AC supply results in half of the heat to be produced on the plate leading to medium rate of weld deposit and medium penetration.

2. **Voltage (potential):** It is the electrical force that causes or pushes the flow of electrons. During arc welding, holding a short arc length results in the lowering of arc voltage. When the arc length is increased, the voltage rises until the arc length exceeds the arc voltage capacity of the power source, and arc is extinguished.

- a) **Electrode size:** Large electrode may stick to plate or arc may be difficult to start. The final bead will also be too convex and overlapped if an overlarge

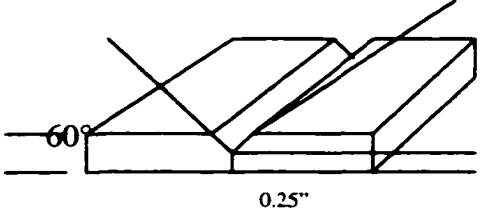


electrode is used. Electrode that are too small may spatter and may even catch fire. It will also be difficult to hold a steady arc length because the electrode will be consumed too fast if it is too small. So selection of appropriate electrode size is very important.

3. **Electrode angle:** The electrode angle is of two types:
  - a) **Travel angle:** If the travel angle is too great from perpendicular to the plate, a bead with poor penetration and too convex in shape will result.
  - b) **Angle of attack:** The angle of attack should always be about one-half the joint angle. For example, for a tee joint, the electrode angle (Angle of attack) should be 45 degrees. An incorrect angle will result in uneven leg length, poor fusion, and possibly, undercut on the top leg.
4. **Travel speed:** . If the travel speed is too low, the bead will pile up and overlap will result. If it is too fast, the bead will be sparse and have poor fusion. Thus travel speed is a very important parameter for proper weld deposit.
5. **Arc length:** The arc length will be changing continuously as the electrode is consumed. If it gets too long, an erratic arc spatter will result. It is good to hold a close arc, especially with low hydrogen electrodes. But if it gets much too close, the rod will not arc properly. Along with improper shielding, sticking and erratic arc may result.
6. **Gas flow rate:** This parameter is very important for processes using inert gases/CO<sub>2</sub> for shielding purposes. It determines the various properties of the welds and an optimal gas flow rate is desired for good weldment.

The above parameters are very important and should be checked first if beads are not proper during welding. Chances are that an adjustment of one of them will solve the problem. Selection of parameters is a difficult task and only very experienced welders can select proper parameters for a particular welding job.

**Table 1.1 A Typical Welding Procedure Specification**

<b>Process: Gas Metal Arc Welding</b> <b>Type of joint: Butt</b> <b>Plate thickness: ½</b> <b>Position: Flat</b> <b>Welded from: Both Sides</b> <b>Polarity: DCRP</b>	
Pass Face	1
Pass Back	1
Electrode Size (in.)	1/16
Current (Amp)	240-280
Volts	28-30
Arc Speed (ipm)	14-18
Argon Gas (cfh)	40-60
Gas cup opening (in.)	½
Preheating and Temperature Control	Preheating to 100 F before Welding
Post-heating	No stress relieving required by this specification
Welding Technique	Weld metal shall be deposited so that there shall be no undercutting of the side walls of the welding groove or the adjoining base metal.
Cleaning	None
Defects	Any cracks or holes that appear on the surface of weld bead shall be removed by chipping, grinding or gouging before depositing the next successive bead

### **1.2.1.3 Weld Procedure Specification (WPS)**

A WPS is the outline of the detailed methods and practices including all joint welding procedures involved in the production of a weldment. A typical weld procedure specification lists the materials, parameters and practices employed in the welding of a particular joint. A typical WPS for GMAW of Aluminum is shown in Table 1.1.

The decision regarding the procedure to be used for welding is specified using WPS and the welder/the welding engineer has to refer to AWS Codes to know about the Weld Procedure Specification (WPS) for a particular job.

### **1.2.1.4 Economic Evaluation**

A cost analysis may be required in advance of actual welding - in which case the results would constitute a cost estimate - or the cost analysis may be made after welding is in progress for various evaluation or management purposes. Cost estimation is beneficial for comparing various competing welding processes also. Similarly, the estimation of the weight of metal deposited is advantageous for purposes like safety of the welded structure. The estimation of the time required to complete the job is advantageous for project management.

### **1.2.1.5 Defect Analysis**

Weld defects result in unsatisfactory welds which cannot function properly in their destined service conditions. It is very important to analyze the various weld defects, their causes and remedies for them. It is easier to diagnose the defect if the data on weld defects is codified in a software which can also suggest causes and possible remedies for the defects.

#### **1.2.1.6 Trouble Shooting**

Sometimes the welders experience problems in using the welding equipment and they have to refer to manuals or consult mechanics for trouble which they themselves could have diagnosed. If all the possible difficulties encountered in welding are codified along with the remedies then the job of welder will be greatly facilitated by the easier trouble-shooting.

#### **1.2.2 Execution and Control in Welding**

The execution and control phase in welding involves the control of welding process and parameters in real time. Most of the times the welder adjusts the parameters manually by observing the arc and weld pool characteristics. An automated sensor and control system based on vision, ultrasonic, arc and laser may be utilized for better welding of geometrically complex fabrications. An adaptive robot control is also used in industry for controlling the parameters by using the sensor information and a knowledge based system to generate the robotic welding program based on that information. The various tasks performed during execution and control in welding are as follows:

1. **Location of part:** The purpose is to locate the start of weld path. Ultrasonic sensors are used in welding robots for the precise location of start of weld path and for collision avoidance.
2. **Joint tracking:** The welding seam is tracked in this task so that the weld electrode or torch movement is along the joint track. Vision Cameras are used in robotic welding for this purpose.

3. **Arc Control:** The characteristics of the arc and the weld pool are monitored and parameters adjusted if any abnormalities are noticed.
4. **Quality Verification:** The quality of the weld is tracked in this task so that the parameters can be adjusted if any defects or trouble are encountered.

### **1.2.3 Quality Evaluation in Welding**

Once a welding job is completed, a number of nondestructive and destructive testing methods are utilized for testing the quality of weldment as follows:

#### **1.2.3.1 Nondestructive Testing**

- |                              |                                     |                      |
|------------------------------|-------------------------------------|----------------------|
| a) Ultrasonic Inspection     | d) Fluorescent Penetrant Inspection | g) Leak Testing      |
| b) X-Ray Radiography         | e) Magnetic Particle Inspection     | h) Acoustic Testing  |
| c) $\gamma$ -Ray Radiography | f) Eddy Current Testing             | i) Visual Inspection |

Ultrasonic inspection can be used to detect and locate internal defects such as cracks, porosity, inclusions, lack of fusion and incomplete penetration. X-ray and  $\gamma$ -ray radiography can be utilized for detecting sub-surface and surface flaws such as slag, flux or oxide inclusions, cracks, porosity, incomplete penetration, and lack of fusion. Magnetic particle inspection is used on magnetic ferrous weldments for detecting invisible surface or slightly subsurface defects like quenching cracks, thermal cracks, seams, laps, grinding cracks, overlaps, non-metallic inclusions, fatigue cracks, hot tears etc. Fluorescent-penetrant testing can be used for detecting small surface cracks, shrinkage and porosity on both ferrous and non-ferrous welded jobs. Eddy current testing can be used to detect flaws close to the surface such as cracks, weld porosity, poor fusion or any linear discontinuity. Acoustic testing can detect defects such as cracks, lack of fusion, slag inclusions etc. Leak

testing can provide information on the soundness of the weld in the working environment. Visual inspection can reveal a lot of defects present on the surface of the weld like cracks, porosity, incomplete fusion and overlaps.

#### **1.2.3.2 Destructive Testing**

- |                 |                    |                     |
|-----------------|--------------------|---------------------|
| a) Tensile Test | c) Impact Test     | e) Hardness Testing |
| b) Bend Test    | d) Nick Break Test | f) Etch Testing     |

The destructive tests provide information about quantitative mechanical and metallurgical properties of the welds. Tensile test is used to determine Tensile Strength, Yield Strength, Elasticity, and Percent Elongation. The bend test are used for determination of ductility, weld penetration, fusion, strength and crystalline structure of the welds. Impact Test provide the Impact Strength of the Weld i.e. How much impact it will absorb before failure. Nick Break Test is used for finding gas pockets, slag inclusions, porosity, degree of fusion and ductility. Various hardness testing methods like Rockwell, Brinell and Vickers provide an idea of resistance to wear of the weld metal. Macro-Etch Test reveals Cracks, Blow holes, Slag Inclusions, Shrinkage Porosity and Penetration of the Weld. Micro-Etch Test provide details on cracks, inclusions, grain boundaries, solidification structures of weld metal, heat affected zone and the base metal and distribution of micro-constituents in the weld metal.

## **2. LITERATURE AND SOFTWARE REVIEW**

There is a lack of a survey of all the various methods used for decision support in welding and this research has assembled a lot of pertinent information about this subject. It has also pointed out a new direction in welding decision-making to provide a comprehensive solution for the problem.

Till recently, decision making in welding is done using manuals [Arc Welding Standard Data Handbook (1975), The Procedure Handbook of Arc Welding (1973), Smith (1984)], experience of the welders, thumb rules learned over time, and heuristics known to experts. Also, most of the critical industrial applications have to adhere to a Code like ASME Boiler and Pressure Vessel Code Section 9 (1998), AWS Standard Welding Procedure Specification (Ref. <http://www.aws.org>) etc. It is difficult to remember the procedure to be used for welding different materials using different processes.

The advent of expert systems for welding made the task of planning in welding easier but the expert systems also suffer from disadvantages like difficulty in representing knowledge [Goel et al. (1992), Goel<sup>1</sup> et al. (1993), Goel (1991) and Goel<sup>2</sup> et al. (1993)]. Also updating the knowledge of the expert system is a difficult task as it involves either the modification of the rules contained in the expert system or manual updating using a user interface which may be difficult and time consuming for novice users of the system.

This led to the utilization of neural networks which can store the knowledge using the same concept as the human brain. But neural networks, alone suffer from disadvantages

such as slow convergence, tendency to memorize sample data and difficulty in generalization [Blum (1992), Illingsworth (1990)] and incomprehensible knowledge.

This led to the development of Neuro-Expert system, an integrated system utilizing the versatility, flexibility and user-friendliness of expert system and the capability of the neural network to store dynamic knowledge. Even the Neuro-Expert approach had disadvantages such as lack of interface between Neural Network and Expert System and hence the inability of the Expert System to learn new rules or to modify existing knowledge. These approaches for decision making have been discussed in the following sections.

## **2.1 Expert Systems**

An expert system is an intelligent computer program that uses knowledge and inference procedures to solve problems difficult enough to require significant human expertise for their solution.

A lot of Researchers have worked on applications of Expert Systems in Industry. Sun et al. (1999) developed an expert system that employed both heuristic and generic knowledge to diagnose the abnormal behaviors that are observed in the solar heating system. Tabatabai (1998) developed a framework for an expert analysis and forecasting system for construction projects.

Matsatsinis et al. (1997) developed an approach for knowledge acquisition and representation for expert systems in the field of financial analysis. Jiang et al. (1998) developed a process planning expert system based on a flexible-digit length coding scheme.



### **2.1.1 Expert Systems for Planning in Welding**

Goel et al. (1992), Goel<sup>1</sup> et al. (1993), Goel (1991) and Goel<sup>2</sup> et al. (1993) have worked on the implementation of an expert system approach for various tasks in welding, ranging from training in SMAW environment to planning. Their approach was quite successful except that the representation of knowledge for a larger domain proved to be difficult due to limitation on the computational capability of the system. The updating and modification of the knowledge base also proved to be difficult. They also computed the cost of the weldment based on the data stored in the database, but since the updating of the knowledge base was required quite frequently, the approach proved to be quite cumbersome.

Taylor (1989) also used the knowledge based approach for the automated generation of procedures (parameters and edge design) in the submerged arc welding domain, and his paper provides useful insight into the problems encountered during the development of expert systems. Napolitano et al. (1992) developed a knowledge-based system for aluminum welding.

### **2.1.2 Expert Systems for Execution and Control in Welding**

Reeves et al. (1988) used expert systems for weld process control and used sensor fusion output in conjunction with a rule base to reconcile competing goals such as cost, quality and productivity in order to make effective decisions during arc-on time. Their expert system is excellent for application in a shipyard. They concluded that as competent welders become more and more scarce, an expert system can be used to extract their

knowledge in a form suitable for control purposes. The same is true for training of inexperienced welders.

Sicard and Levine (1987) developed an expert robot welding system, which can be used for automation of welding system and used for automation of welding processes like gas tungsten arc welding (GTAW). They concluded that a sophisticated expert system should be able to guide the user in the preparation and planning phase of the welding process. The system must also be able to plan the welding task in accordance with user specifications. Lucas (1987) and Kerth (1985) have also discussed the advantages of using expert systems for welding.

### **2.1.3 Expert Systems in Quality Evaluation**

Paton et al. (1990) have listed the principles of constructing expert systems for evaluating quality of welded components.

### **2.1.4 Limitations of Expert Systems**

Using an expert system approach alone has many limitations. The domain for the expert system is limited because the performance of the system becomes very slow if the domain is large. The updating of the knowledge base is also cumbersome and it is very difficult to design a system with transparent updating of knowledge. Also, there are no means for validating the knowledge contained in the expert system.

## **2.2 Neural Networks**

Neural networks can take input data and determine the data that is relevant, the irrelevant data is taken care of by assigning a low connection strength to all of the output neurons that

it results in no effect. Neural networks have been used for sensor processing, control, data analysis and a lot of other tasks.

Luchetta et al. (1998) utilized a Neural Network trained using Back-propagation for diagnosis and control of Highway Surfaces. Their system proved to be reliable and improved the manageability of safety and intervention teams.

Boger (1992) has reviewed the application of neural networks to water and wastewater treatment plant operation. He concluded that the application of neural network for learning from past experience has great potential in many diverse areas such as steady-state and dynamic modeling, fault diagnosis, process monitoring, expert rule extraction, and eventual process control.

Lewis (1996) utilized neural network for the control of Robot Manipulators as a powerful and robust alternative to adaptive control.

### **2.2.1 Neural Networks in Planning for Welding**

Fukada and Yoshikawa (1990) used a neural network approach for the determination of welding sequence. Their approach was very useful as welding sequence determines the shrinkage, distortion, residual stresses and/or restraint stress and hence certain heuristic rules have to be followed. They represented the weld lines and the torch in form of a neural network, which was later trained using actual welding. They concluded that using a neural network approach is better than a knowledge based approach as the presence of distortion and other defects is dependent on many factors and it is difficult to modify the weights using a knowledge based approach. In the case of a neural network, a solution fitted

to the altered goal may easily be sought by merely changing the value of weights on restrictions. One limitation in their model is that the solution cannot be determined uniquely, and the user has to decide whether the solution is appropriate or not.

### **2.2.2 Neural Networks in Execution and Control of Welding**

Gabor et al. (1992) have discussed the applications of the neural network approach for the modeling and control of welding processes. They have modeled the static and dynamic behavior of the process but they do not discuss how to modify and optimize the process parameters based on the neural network output.

### **2.2.3 Neural Networks for Quality Evaluation in Welding**

Liao and Tang (1997) utilized MLP neural networks for automated extraction of welds from digitized radiographic images. Onda et al. (1992) used an artificial neural network for welding defect identification.

Zheng et al. (1993) used neural networks for parameter prediction and quality inspection in TIG welding. Their approach involved the input of job variables directly to the neural network. The neural network then predicted the process parameters, which in turn served as input to another neural network that predicted the weld quality. This approach results in slow convergence for training and is cumbersome. They have also not discussed the means by which they are modifying the input process parameters.

Gaillard et al. (1990) proposed a neural hyper-column architecture for the preprocessing of radiographic weld images.

#### **2.2.4 Neural Network Training**

Fukuoka et al. (1998) have proposed a modified back-propagation method to avoid false local minima. They keep the sigmoid derivative relatively large while the error signals are large. They substantiated the validity of the method by numerical experiments.

Porto et al. (1995) have investigated three potential neural network training algorithms in processing active sonar returns and demonstrated that the stochastic training methods of simulated annealing and evolutionary programming can outperform back-propagation.

Fu (1996) has proposed an incremental knowledge acquisition in supervised learning networks by using a bounded weight modification to update its connection weights and perform structural learning. He also developed strategies for preventing overtraining and incrementally growing and pruning the network.

#### **2.2.5 Pruning Approaches in Neural Networks**

Sometimes the Neural Networks are too complex to be trained efficiently and lack generalization capability and the number of input nodes has to be reduced. Ponnappalli et al. (1999) proposed a selection and pruning technique based on the concept of relative sensitivity for feedforward artificial neural networks. Their method does not support Neural Networks with multiple input and output systems. Other techniques like optimal brain surgeon (OBS) [Hassibi et al. (1993)] and optimal brain damage (OBD) [Cun et al. (1990)] have been used to fully prune a fully connected Feedforward Artificial Neural Network, but they require considerable additional computation. Lu et al. (1996) have applied a rule

extraction algorithm to Data Mining which uses a clustering method to merge two sets or rules into one and thereby reducing the complexity of the network. Setiono and Leow (1999) used a pruning based method for mapping decision trees to neural networks which reduced the size of the Neural Network and the derived decision tree considerably.

Lovell and Bradley (1996) have proposed a Multiscale Classifier (MSC) for rule based inductive learning. They compared their approach to other decision tree classifiers and neural networks, and found that MSC offered faster learning and comparable generalization performance. Looney (1996) described some training approaches for pruning adjacent and hidden nodes and described the benefits of pruning.

### **2.3 Neuro-Expert Systems**

Many investigators have used an expert system integrated with a neural network for various tasks. They have also been called expert networks, hybrid systems, and Neuro-Expert Systems. Several strategies have been suggested for the integration of neural networks and expert systems into hybrid systems. Medsker and Bailey (1992) suggested five methods for integration, which include:

- a)     **Standalone:** This model consists of independent software components that do not interact in any way.
- b)     **Transformational:** This model begins as one type of system and end up as another.
- c)     **Loose coupling:** Loosely coupled models consist of separate components that communicate through data files.

- d) **Tight coupling:** Tightly coupled models pass information through memory resident data structures rather than external data files.
- e) **Full integration:** Fully integrated models share data structures and knowledge representation. The communication between components is through the dual nature of the structure.

Caudill (1991) has provided an overview of Expert Networks and described the advantages of using expert networks. The author has described four approaches that may be used for combining neural networks and rule-based systems:

- a) **Divide and Conquer:** This strategy involves partitioning a problem into parts, solving each part by whichever is more appropriate, an expert system or neural network.
- b) **Embedded:** The embedded approach makes the neural network a part of the rule-based system.
- c) **Explanation by Confabulation:** This approach uses a neural network to solve a problem and an expert system to confabulate a justification of the network's answer.
- d) **Artificial Expert:** In this method, a neural network is trained to solve a problem and then it is analyzed to extract a set of rules from the trained network.

Some examples of these four approaches have been discussed below. A coupled approach has also been used by other researchers [Xu et al. (1998)] for implementing neuro-expert systems. In this approach a set of modules are developed and they transfer

information to each other in order to interact and come up with a unified solution to the problem. As discussed earlier, coupled neuro-expert System can be of two types - loosely coupled and tightly coupled.

### **2.3.1 Divide and Conquer Neuro-Expert Systems**

Oskada and Yang (1991) applied neural network to some problems associated with the cold forging expert system. These problems include the determination of forming methods from the final product shape, the prediction of the most probable number of forming steps by considering the shape complexity and material property, and the prediction of the die fracture and surface defect in the formed product. Satisfactory results are obtained by combining the neural networks with the expert systems in most cases.

### **2.3.2 Embedded Neuro-Expert Systems**

Looney (1993) demonstrated that the high level decision making function of expert systems, which depend upon many levels of logic, could be implemented in a neural network without the engineering of a detailed structure. He also discussed the various advantages of using this approach, namely the ability of the neural network to interpolate and extrapolate a discrete set of associated input and output vectors, so that the output vector space is continuous.

### **2.3.3 Explanation by Confabulation**

Rosen and Silverman (1992) used neural networks for extracting regularities from case histories and predicting the Stress Corrosion Cracking (SCC) risk in various materials in different environments in an automated way. The neural network was then incorporated



within an expert system to provide simple consistency check, and to interpret the numerical output of the neural network to make the final prediction. The rule-based expert system also provides for incorporation of the rules needed to interpret the numerical output so as to present the final predictions in a readily usable form.

#### **2.3.4 Artificial Expert**

Artificial expert systems have been used by some researchers, which involves the development of an intelligent hybrid system [Fu (1993), Fu (1991), Rosen et al. (1992), Hillman (1990)]. In this approach, three major issues must be addressed. First, useful domain attributes and concepts are identified and linked in a way consistent with initial domain knowledge, and then the links are weighed properly so as to maintain the semantics. Secondly, a learning algorithm is used to adapt new data into existing knowledge. Lastly, the learned network is interpreted to extract the knowledge.

Fu (1996) has discussed that acquiring new knowledge without interfering with old knowledge is a key issue in designing an incremental-learning system. His paper describes an incremental-learning network for pattern recognition that uses a rule based connectionist technique to represent general domain and case specific knowledge, uses bounded weight modification to update its connection weights, and also performs structural learning. Specific strategies are developed for preventing over-training and for incrementally growing and pruning network. The soundness of this approach is demonstrated by empirical studies in two independent domains.

#### **2.3.4.1 Coding Rules in Neural Network Structures and Learning Algorithms**

Loukatzikos and Galletly (1994) described a PC-based expert system rule-to-neural network translator, which translates knowledge expressed as rules to a neural network representation.

Lacher et al. (1992) developed back-propagation learning for acyclic, event-driven networks in general and derived a specific algorithm for learning in EMYCIN-derived expert networks.

Fu (1998) based the activation function of neural network model on Certainty Factor and demonstrated the reduction in complexity of rule learning by capitalizing on the activation function of the Neural networks.

Petridis et al. (1998) used a hybrid neural-genetic multimodel parameter estimation algorithm and evaluated the algorithm by numerical simulations for a planar robotic manipulator and a waste water treatment plant. Jagielska et al. (1999) presented some highlights in the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition.

Wetst and Vanthienen (1997) extended a tabular knowledge-based framework with feature selection. They pre-processed the raw data to reduce it using feature selection and then generated the rules from it and modeled the knowledge by decision tables. After verification and validation, the modeled knowledge was incorporated in an expert system to facilitate the consultation of the knowledge base.

#### **2.3.4.2 Knowledge Extraction in Artificial Expert Systems**

Tickle et al. (1998) have surveyed the directions and challenges in extracting the knowledge embedded within trained Artificial Neural Networks (ANN). Their paper has identified some of the key research questions in extracting the knowledge embedded within ANN's including the need for a consistent theoretical basis for what has been, a disparate collection of empirical results. They also came up with five primary classification criteria for rule extraction from ANN's - expressive power or rule format, quality, translucency, algorithmic complexity and portability.

##### **2.3.4.2.1 Rule Format Criteria for Rule Extraction Classification**

This classification has three groupings of rules:

- 1) Conventional symbolic rules
- 2) Rules based on fuzzy sets or logic
- 3) Rules expressed in first order logic, i.e. rules with quantifiers and variables

Huang and Endsley (1997) provided an understanding of the behavior of feedforward neural networks with respect to expressive power or rule format. They argue that problem solving knowledge of a neural network is represented at a subsymbolic level and hence is difficult for human user to comprehend. They also propose a method for fuzzy rule extraction from neural networks with continuous-valued input, as they felt that most methods for rule extraction from feedforward neural networks dealt with networks with

binary inputs only. Their approach has a limitation that it can only be applied to feedforward neural networks with a sigmoidal nodal function.

#### **2.3.4.2.2 Quality Criteria for Rule Extraction Classification**

Andrews et al. (1995) suggested four measures for this criterion - accuracy, fidelity, consistency and comprehensibility.

Taha and Ghosh (1999) have provided a good analysis of the quality criteria for rule extraction. They presented three techniques for extracting rules from Neural Networks in Artificial Expert Systems. They also proposed a rule-evaluation technique, which orders extracted rules based on three performance measures. They applied the three techniques to several applications and evaluated the extracted rules qualitatively and quantitatively. They concluded that their techniques are less complex and that the rules extracted by them are efficient, comprehensible and powerful.

#### **2.3.4.2.3 Translucency Criteria for Rule Extraction Classification**

Researchers [Andrews (1995), Fu (1998)] have classified symbolic knowledge extraction into two approaches using a translucency approach - the open box (or decompositional) and the black box (or pedagogical) approach, the former being more analytical compared to the latter which is more empirical in nature.

The decompositional techniques looks at the rules from a low level i.e. each node of the neural network is analyzed for rule extraction while the pedagogical method looks at the global level without going into the details of each node. Since some of the rule extraction techniques may not fall into either of these categories - an eclectic category was formed to

classify the essentially hybrid techniques which analyze the ANN at a node level but extracts the rules at a global level.

#### **2.3.4.2.4 Algorithmic Complexity Criteria for Rule Extraction Classification**

Fu (1998) based the activation function of neural network model on Certainty Factor and demonstrated the reduction in complexity of rule learning by capitalizing on the activation function of the Neural networks.

Setiono and Liu (1996) proposed an algorithm to symbolically represent the neural network so that the prediction can be explicit and understandable. Magrez and Rosseau (1992) have provided methods for symbolic interpretation for back-propagation networks. They used NN training examples as input to an algorithm, which produces a set of implication rules. These rules accurately model the NN behavior. They tried to resolve two issues, namely what do the output values produced by the activation of the NN mean, and the possibility of symbolically describing the content of a NN.

#### **2.3.4.2.5 Portability Criteria for Rule Extraction Classification**

This classification scheme portrays the applicability of the rule extraction technique across ANN architectures so that the technique may be categorized as a specific purpose technique for cases with limited scope or general purpose for the techniques applicable in more than one ANN architecture.

#### **2.3.5 Coupled Neuro-Expert System**

Xu et al. (1998) used a loosely coupled neuro-expert system for microscopic wear particle analysis using simple message interaction. The neural network was used to classify

particle features and the expert system did the overall wear assessment. They concluded that their system was adaptable, extendable, modular and fast.

Lee (1999) used a tightly coupled-ART neural network for modularized categorization of patterns and concluded that coupled modules were able to provide higher accuracy than the uncoupled ART neural network.

Chiang (1998) used a coupled hybrid system for handwritten work recognition and concluded that his system performs better than the base line system.

### **2.3.6 Neuro-Expert Systems in Welding**

Neuro-Expert systems have not been used extensively in planning for welding. Liao et al. (1993) utilized a Neuro-Expert system approach for the automated selection of parameters in Metal Inert Gas (MIG) welding and modeling of an automated MIG welding process.

An expert system was used for deciding the initial process parameters based on job variables. A neural network was then utilized for mapping an input/output pattern for different combinations of job variables. This neural network also was used for validating the process parameters arrived upon by the expert system. Thus the integrated Neuro-Expert system proved to be very useful for automated control of MIG welding.

They demonstrated that a Neuro-Expert system can overcome the problem of incorrect selection of parameters, by integrating the knowledge of the experts and the learning capabilities of a neural network, and utilizing this expertise for selecting the correct process parameters of MIG welding.

### **2.3.7 Verification and Validation of Neuro-Expert Systems**

Many investigators have discussed the issue of validation of knowledge based systems. Murrell and Plant (1997) have surveyed the various tools for the validation and verification of knowledge-based systems.

O'Keefe et al. (1987) have discussed means to avoid the limitations of traditional validation techniques. The limitations may be selection of test choices resulting in misleading performance, testing of system performance against the same expert whose knowledge was used for building the system.

Moore (1998) has analyzed the objective and subjective measures of error-proneness for rule-based programs. He proposed a number of measures for quality assessment of rule-based structures.

Tsai<sup>1</sup> et al. (1999) presented perspectives on issues and problems that impact the verification and validation of knowledge based systems along with an overview of different techniques and tools that have been developed for performing verification and validation.

Tsai<sup>2</sup> et al. (1999) have surveyed knowledge-based software architectures for acquisition, specification and verification. They provided a basis for comparing the various knowledge-based systems.

Howes and Crook (1999) have investigated three related measures of input parameter influence for supporting explanation facilities for neural networks. They also presented an algorithm for generating rules from real-valued networks based on the influence measures.

## **2.4 Fuzzy Systems**

Fuzzy systems have been applied by many researchers to same problems as Neural Networks [Welstead, 1994] as they can model nonlinear processes as well, along with the advantage of being able to provide an insight into their operation by providing a commonsense description of their actions. Fuzzy systems take one or more variable as input, defined in terms of fuzzy sets, with a collection of fuzzy rules to produce an output value.

Funabashi et al. (1995) have examined the practicality of using a next generation of fuzzy expert systems that combine techniques such as logic programming, fuzzy theory, and neural networks to improve performance and productivity. They examined the practical use of two such hybrid architectures: combination and fusion. Wong and Lin (1997) proposed a method based on genetic algorithms to automatically extract fuzzy rules to identify a system where only its input-output data are available. Their method can determine a fuzzy system with fewer fuzzy rules as well as the antecedent and consequent parameters of the fuzzy rules at the same time. A nonlinear system was utilized to illustrate the efficiency of the proposed method in the rule extraction for fuzzy modeling.

Nishina and Hagiwara (1997) proposed a fuzzy inference neural network, which automatically partitions an input-output pattern-space and extracts if-then rules from numerical data. They used Kohonen's learning algorithm in a two layer network and demonstrated the effectiveness of their method in fuzzy control of an unmanned vehicle and prediction of the trend of stock prices by using computer simulation.



Delgado et al. (1998) presented a methodology to model fuzzy systems using fuzzy clustering in a rapid prototyping approach. They proposed several approaches to the problem of fuzzy rules extraction by using fuzzy clustering. Fuzzy Systems are not used in this research as most of the rules were found to be quite crisp. In case of fuzzy rules, heuristic reasoning and thumb rules (from welding experts and literature) were utilized to convert them into crisp rules. Liao and Li (1996) used a fuzzy K-NN algorithm for extracting welds and identifying tool failures. Their results show that the fuzzy K-NN classifier yielded high successful rates for recognition for both applications.

Kim and Na (1993) used a self-organizing fuzzy control approach to arc sensor for weld joint tracking in gas metal arc welding of butt joints. Parsons (1996) also surveyed methods for representing and reasoning with imperfect information. Jota et al. (1998) also demonstrated the effectiveness of hybrid intelligent systems in equipment fault diagnosis in electric power systems.

## **2.5 Fuzzy Neural Networks**

Recently, research has been done in Fuzzy Neural Networks and the integration of fuzzy sets and neural networks. Farag et al. (1998) proposed a combination of the fuzzy logic theory, neural networks and Genetic Algorithms (GA's) to achieve linguistic modeling of complex irregular systems which can handle quantitative and qualitative knowledge. Their learning algorithm is composed of three phases. In the first Phase, the initial membership function of the fuzzy model is determined. A new algorithm is developed in

the second phase for extracting the linguistic-fuzzy rules. A multi-resolutional dynamic genetic algorithm is used to tune the membership functions in the third phase.

Meneganti et al. (1998) utilized a fuzzy neural network for classification and detection of anomalies and overcame some of the undesired properties of Simpson's fuzzy min-max neural network specifically, in it there are neither thresholds that bound the dimensions of the hyperboxes nor sensitivity parameters.

Kuo and Cohen (1998) utilized a radial basis function and a fuzzy neural network for on-line tool wear estimation. They concluded that the proposed system could significantly increase the accuracy of the Product Profile.

## **2.6 Other Approaches for Control in Welding**

Hendersen et al. (1993) used a pseudo-gradient adaptive algorithm to self-tune a proportional-integral (PI) puddle width controller for consumable-electrode gas metal arc welding. But for general applications, just controlling the puddle width is not enough and for achieving good overall weld quality controlling other factors like bead shape, spatter etc. are also important.

Sugitani and Mao (1995) used an arc sensor for automatic simultaneous control of bead height and shape in one sided welding with a backing plate.

Kim and Na (1993) used a self-organizing fuzzy control approach for weld joint tracking in Gas Metal Arc Welding of Butt Joints. Their system has a very limited domain and cannot be applied in broader domains and for other welding processes.

## **2.7 Other Software in Welding**

A number of commercial software programs are available for performing various decision-making tasks in welding planning, as follows:

1. Arc Works WeldSelector (Lincoln Electric Company): Aids in selecting Welding Electrode by combining fuzzy logic, rule based induction systems, mathematical models, and decision trees to suggest matching welding electrodes.
2. InfoWeld PC [Wijtek Ltd., available through American Welding Society (AWS)]: Calculates the cost per meter of weld, job costing, weld time and weight of filler metal.
3. Weldspec 4: Co-developed by C-spec and TWI (available through AWS), this software helps in the creation and management of multi-code WPSs and PQRs.

None of these programs provide a comprehensive solution for decision making in Welding i.e. An integrated approach for selection of welding process, parameters, WPS, economic evaluation, trouble shooting and defect analysis is lacking. Also, since the various modules do not interact, the optimization of parameters or cost is not possible.

### **3. OBJECTIVES OF THE RESEARCH**

The ultimate objective of this research is to build a neuro-expert system to support decision-making in welding planning. The selection of process, parameters and consumables and the development of weld procedure specification (WPS) are still being carried out manually through manuals and catalogs [Reeves et al. (1988)]. Despite the existing commercial software programs and past works on standalone experts system and neural network applications, as reviewed in Chapter 2, a better tool to support decision-making in welding planning is needed. This research focus on the following planning tasks in the welding domain:

1. Process Selection
2. Selection of weld parameters
3. Economic Evaluation
4. Defects Analysis
5. Trouble Shooting
6. Weld procedure specification

The system is demonstrated, through a built prototype, to have the following advantages:

- a) **Learning and knowledge discovery capability:** The Neuro-Expert System learns from user-interaction and modifies the rules based upon this interaction. It also learn from data that later become available.

- b) **Faster Decision-making:** Selection of Process, parameters and calculation of cost, time for welding and weight of metal deposited is done in a very short time, compared to the manual method.
- c) **Factual and Accurate Decision-Making:** As the Neuro-Expert system does not overlook any relevant information, the decisions are more accurate. As the system is directed to utilize relevant information, it doesn't forget to apply an internal knowledge source and it does not miss problem data supplied by the user. Even a skilled expert in welding may overlook some important aspect or information.
- d) **Decision-making with Expertise:** The Neuro-Expert system can be used by a novice in welding domain and thus dependence on welding experts for deciding job parameters, consumables etc. is reduced. In addition, the Neuro-Expert System stores cost and time information and optimizes the decision by trying to minimize the time and cost within the constraints of job requirements.

Research issues related to building the system that are addressed in this dissertation include:

1. **Design an overall architecture of the neuro-expert system:** This involves the determination of system components and their relationships. This architecture should enable the optimality in decision making for welding planning in a

manner so that it selects a process, which minimizes the cost for the welding job.

2. **Formulate a Knowledge Extraction strategy for decision-making for various modules:** Devise a suitable strategy for representing, refining and updating the knowledge contained in the Neuro-Expert system by extracting new rules and modifying existing rules.
3. **Demonstrate the effectiveness of knowledge extraction by developing a prototype:** Design and develop a neuro-expert system prototype for the six tasks described earlier.
4. **Validate the neuro-expert system approach for decision-making:** Test the system by verifying the results with handbooks and evaluate the performance of the system.

## 4. NEURO EXPERT SYSTEM DESIGN AND SPECIFICATIONS

This chapter deals with the design and specification of the overall neuro-expert system. This task was achieved by investigating the various research issues and approaches pursued in the field of neuro-expert systems and then devising an approach that avoids their pitfalls. The various modules in this system, used different strategies for knowledge representation, updating and refinement. Overall, a coupled neuro-expert system strategy was used for the whole system. The overall high level system architecture showing the information flow between the six modules is given in Figure 4.1 below. The overall specifications of the system are in table 4.1.

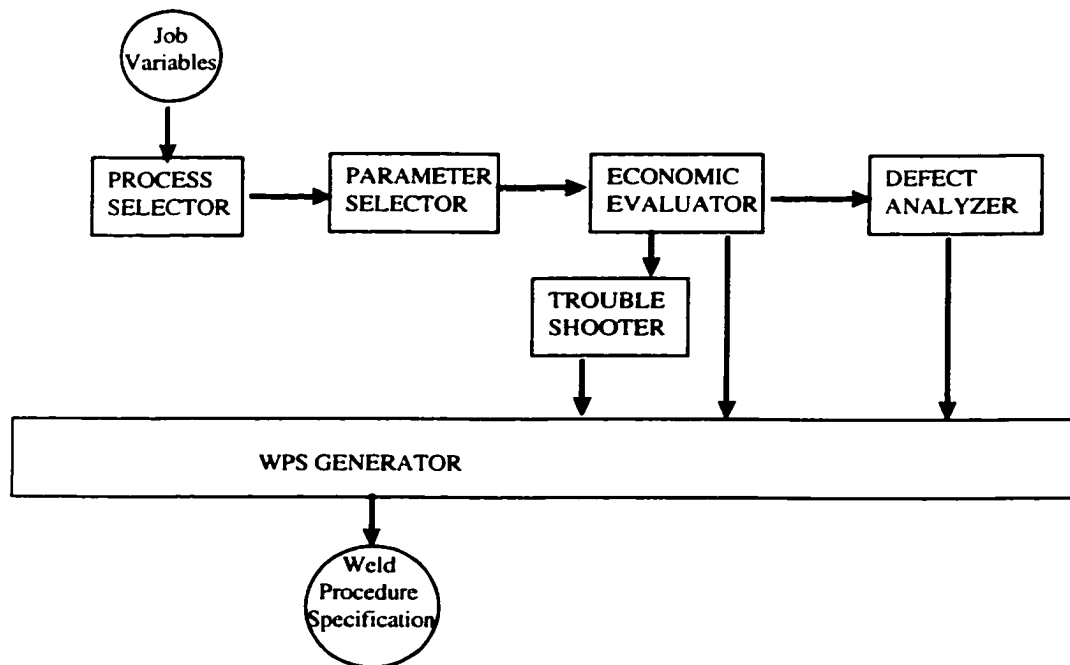


Figure 4.1 Overall System Architecture and Information Flow

**Table 4.1 Current Specifications of the Neuro-Expert System**

<b>MODULE</b>	<b>TASKS PERFORMED</b>	<b>PROCESSES</b>
<b>Process Selector</b>	Select Process/set of applicable processes (Neuro-Expert System) for Carbon Steel, Aluminum and Stainless Steel.	SMAW, GMAW, GTAW, SAW
<b>Interactive Defect Analyzer</b>	Trouble shoot problems encountered in welding interactively	SMAW, GMAW, GTAW, SAW
<b>Interactive Trouble Shooter</b>	Analyze defects and suggests causes and remedies interactively.	SMAW, GMAW, GTAW, SAW
<b>Parameter Selector</b>	Compute Parameters (Database Query System). The program is capable of supporting more processes when standard data becomes available.	SMAW, GMAW
<b>Economic Evaluator</b>	Calculate Cost, Time, Weight deposited (Procedural Code). Select Process with the least cost	SMAW, GMAW
<b>WPS Generator</b>	Generate complete WPS (Expert System)	SMAW GMAW
<b>Defect Analyzer</b>	Predict Defects (Neural Network) Adjust Process Parameters/WPS (Expert Network)	SMAW
<b>Trouble Shooter</b>	Suggest Remedial Procedures proactively to WPS Generator based on process and parameters selected.	SMAW GMAW

The following are some key design features of our approach:

1. Process selector selects a process or a set of processes based on the job variables.
2. Parameter selector selects the parameters for this process or set of processes.



38

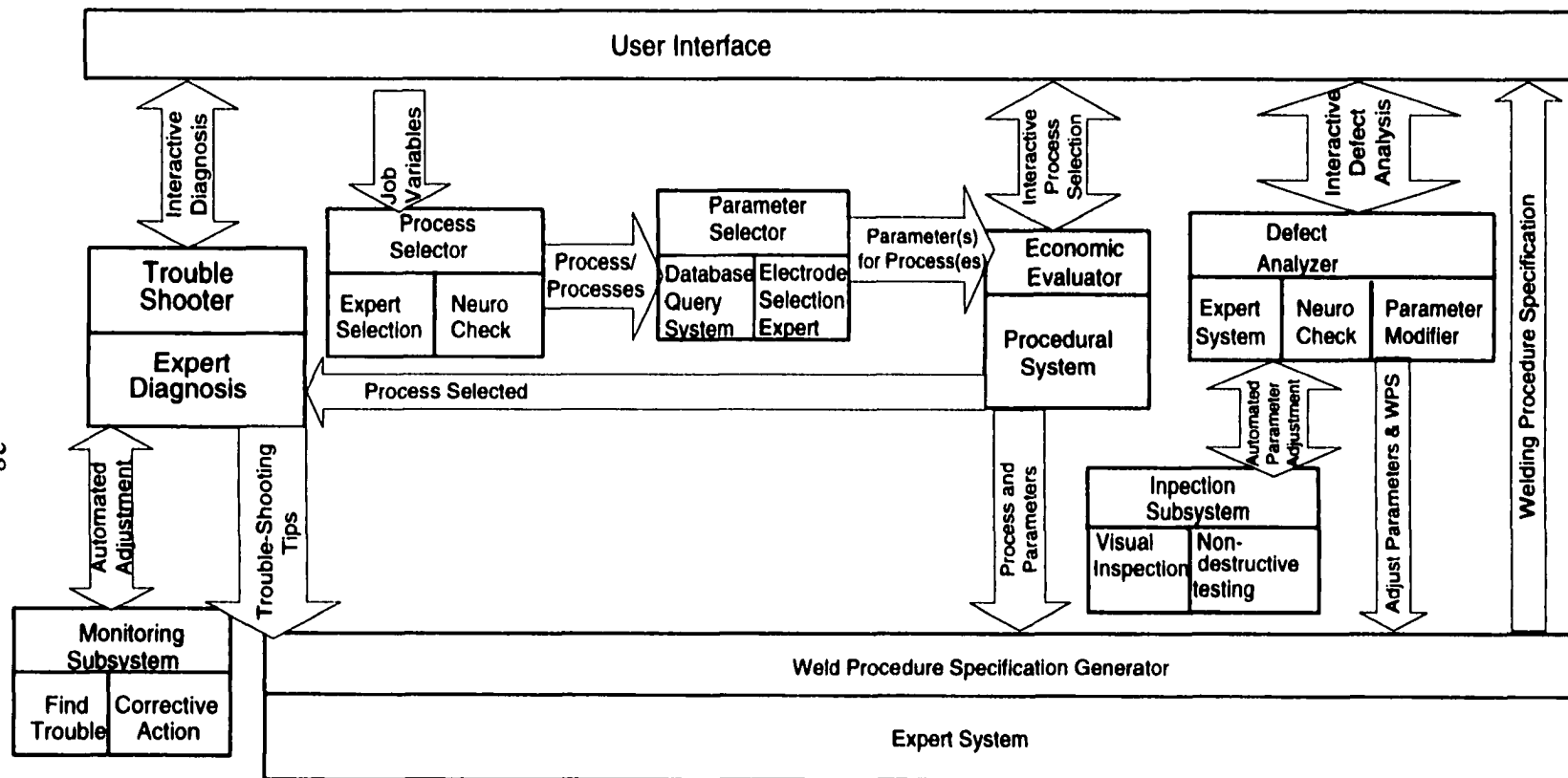


Figure 4.2 Neuro-Expert System Detailed Architecture & Information Flow

3. Economic evaluator helps in selecting a process that is most economical. It then stores the process selected in a working knowledge base.
4. Defect analyzer predicts any defects for the set of parameters selected. It adjusts the parameters to minimize the defects. If defects are predicted even after adjusting the parameters, instructions (WPS Tips) are inserted in the working knowledge base for the WPS generation.
5. Based on the process selected, an applicable set of trouble-shooting tips is inserted into the WPS working knowledge base by the trouble-shooter.
6. The WPS generator uses the process, parameter, trouble shooting and defect prevention tips information and applies its internal rules to determine a final WPS for the job.

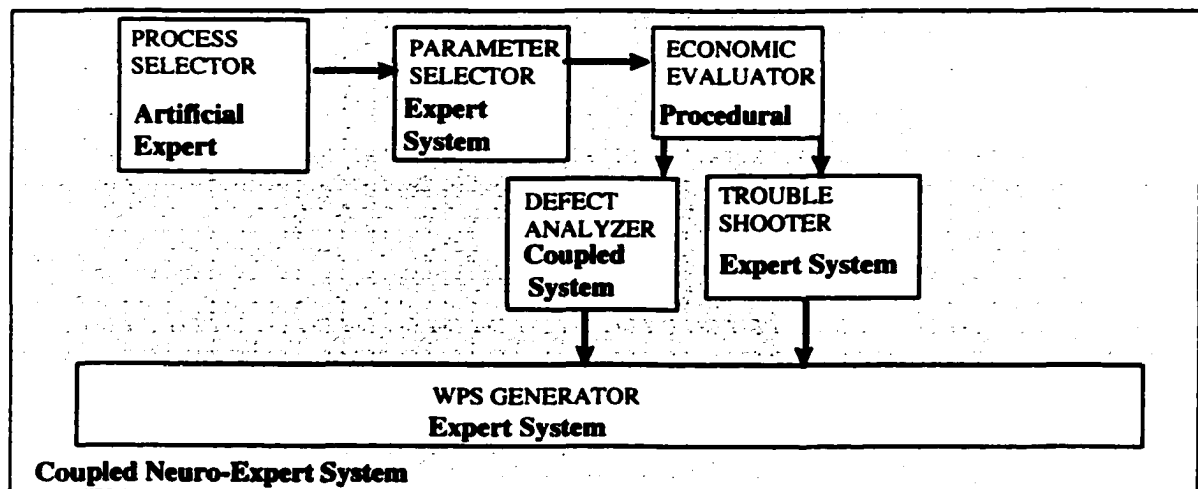


Figure 4.3 Neuro-expert approaches for various modules of the System

This proactive approach ensures that the welding procedure specification generated from the system is optimal and results in high-quality defect-free welds at most economical

cost. Fig. 4.2 shows the neuro-expert system in greater details. Figure 4.3 shows the various approaches followed for the various planning tasks in the neuro-expert system.

#### **4.1 Process Selector Design**

An *artificial expert* approach was used for the process selector. The following steps are used in this approach:

1. A neural network is designed and its weights derived from expert system rules.
2. The neural network is trained with examples.
3. Neural network weight changes are analyzed and the expert system rules updated.
4. Both the updated expert system and the neural network attempt to find out the applicable process for the job variables and a confidence factor is computed for both and the selection with the higher confidence factor is recommended.

Thus the process selector has 2 components - expert system and neural network. The flowchart in figure 4.2 shows the method used by the process selector for selection of the welding process for a set of job variables.

This dual strategy of using an expert system and neural network may seem to involve some redundancy. But this approach is justified because of the following reasons:

1. Research has shown that using multiple classifiers yields better results than using a single classifier alone.

For example, hand-writing recognition using a single approach may not always classify all the characters [Kittler et al. (1998)].

2. Sometimes, the expert system and the neural network have different opinions - based on factors like training set, number of rules etc. Any discrepancies in the knowledge base can thus be discovered and fixed. In such cases, the conflict is resolved by using a confidence factor that determines the confidence with which the expert system or neural network is selecting the process. If the expert system selects a different process with a higher confidence than the neural network is trained. If the neural network confidence is higher for a different process then the expert system rules are modified.
3. Just using a neural network approach may result in selection of an inappropriate process because the neural network may select a process that is not applicable for that job at all. For example, neural network may recommend submerged arc welding for a job of overhead welding, which is not feasible. In such cases, the expert system *throws the invalid process out* of the selection set and compares the neural network's next selection against the expert system selection.

The Process Selector generates a list of applicable and valid process (or processes) in a suitability ranking order, based on the input job variables, and the selected process (or list of processes) is communicated to the parameter selector using a working knowledge base stored in a prolog data file. This communication is important, because the selection of parameters depends on the process to be used.

#### **4.1.1 Process Selector Expert System**

The expert system takes the input job variables and recommends an arc welding process (or processes) to be used for the welding using the following steps:

#### 4.1.1.1 Data Entry

The job variables are entered in this task. A prolog program displays a screen with the job variable fields and the user can select the various qualitative job variables from pop-up menu and enter the quantitative variables in the fields. The variables are then stored in data file to be retrieved later by other modules.

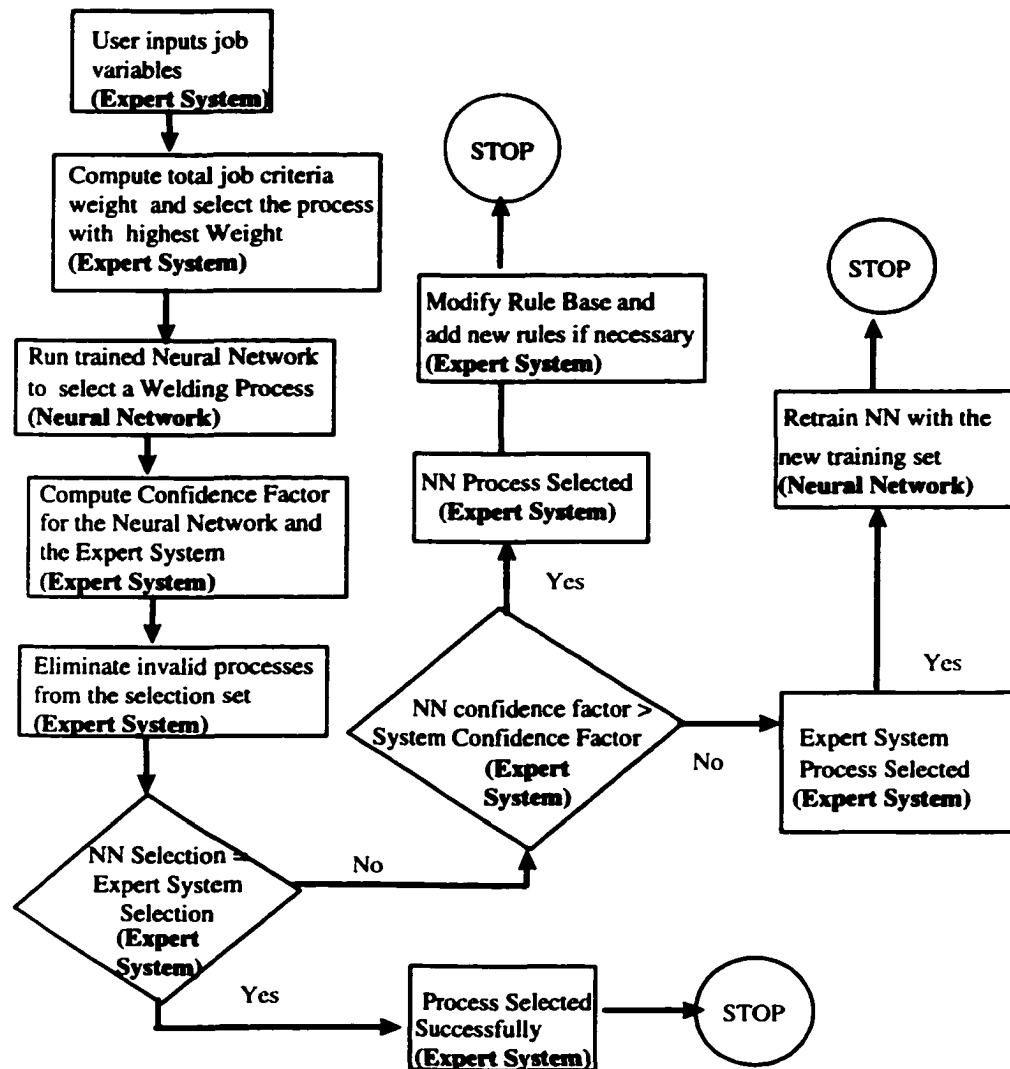


Figure 4.4 Flowchart for the Process Selector

#### 4.1.1.2 Data Conversion

Quantitative job variables are categorized or converted into qualitative variables for the expert system (see Appendix A - Table A.1) because the expert system cannot handle continuously variable data. This improved the convergence of the neural network

For example, plate thickness < 5 mm is categorized under low, plate thickness between 5 and 10 mm is medium and plate thickness > 10 mm is categorized as high. Since all the rules relate to these categories, it is easier to convert the rules into neural network weights and vice versa.

#### 4.1.1.3 Weight Assignment and Rule Mapping to Neural Network

Based on the set of job variables, a criteria weight is assigned to each welding process for each job variable. The rules used for determination of criteria weight for each job variable were determined by applying heuristic rules and are given in table 4.3. These weights are also used for initializing the neural network and are just used once in the life cycle of the neural network, as after that the modified weights are stored in the expert system rule base and the neural network link weights. The format of storage of the rule in the expert system rule base (prolog internal database) is as follows:

*Criteria(CriteriaValue,SMAW\_Weight, GMAW\_Weight, GTAW\_Weight, SAW\_Weight)*

For example, the rule **parent\_material\_criteria** (“ALUMINUM”, -1,1,1,1) implies that Aluminum cannot be welded by using SMAW and hence a weight of -1 is applied to SMAW and +1 to the other Processes as they are suitable for Aluminum as shown in figure 4.5.

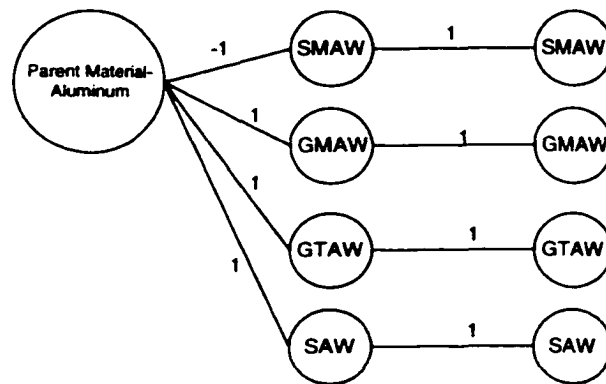


Figure 4.5 Expert System Rules Initial Mapping to Neural Network

After training the weights may get modified (figure 4.6) and that represent a possible modification in the rules (or the weights of that job variable or criteria) for that process. This decision is made based on a set of criteria given in section 4.1.2.

There is a conjunctive node for each process and all the weights joined at this node represent the overall rule for the selection of that process. If the inputs of 1 are coming from nodes that are linked to conjunction nodes by negative weights (figure 4.7) then the process represented by that conjunctive node is not selected.

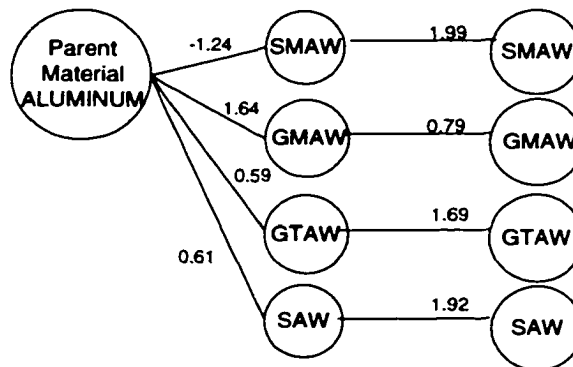


Figure 4.6 Expert System Rules Mapping to Neural Network after training

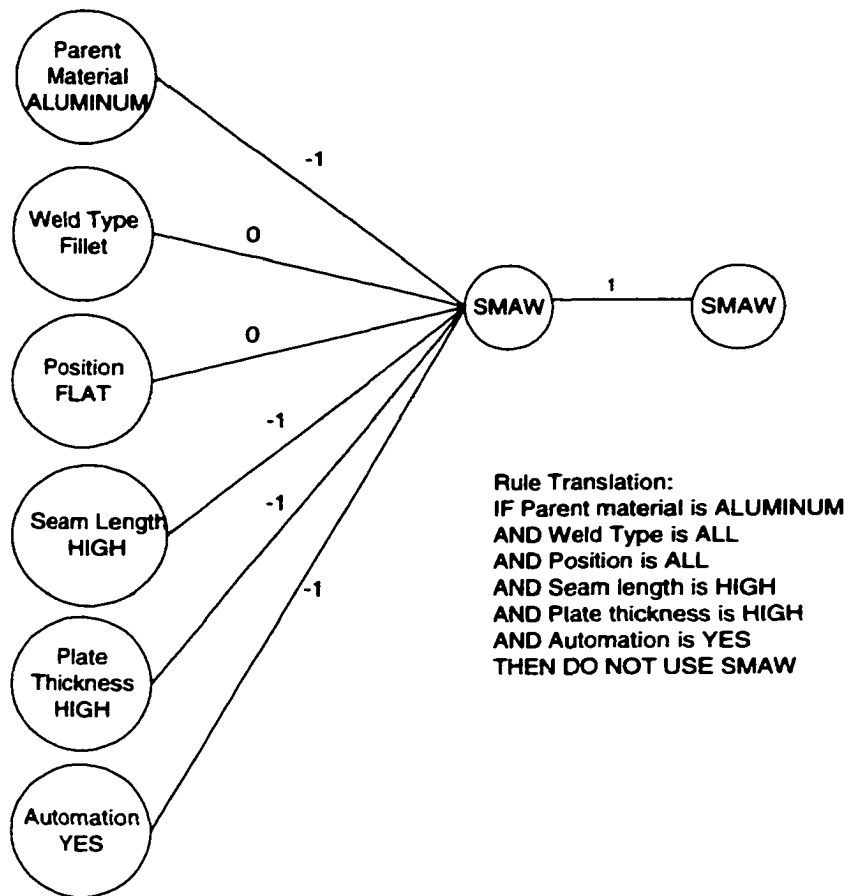


Figure 4.7 Example of Overall Expert System Rule for SMAW

If the most of the inputs come from nodes linked with positive weights (Figure 4.8) then that process might be selected.

**4.1.1.4 Ranking of Welding Process:** All the weights assigned in step 4.1.1.3 are summed for each process and stored in a knowledge base in descending order. An expert system confidence factor is computed for all the applicable welding process, say n number of them, using the formula given below:

$$CF_i \text{ (Confidence Factor for applicable process } i) = (TC_i - TC_{i-1}) / TC_i$$



where  $i = 1$  to  $n$ ,

$TC_1$  = The applicable process with the highest total criteria weight

$TC_2$  = The applicable process with the 2<sup>nd</sup> highest total criteria weight ,

$TC_n$  = The applicable process with the  $n^{\text{th}}$  lowest total criteria weight

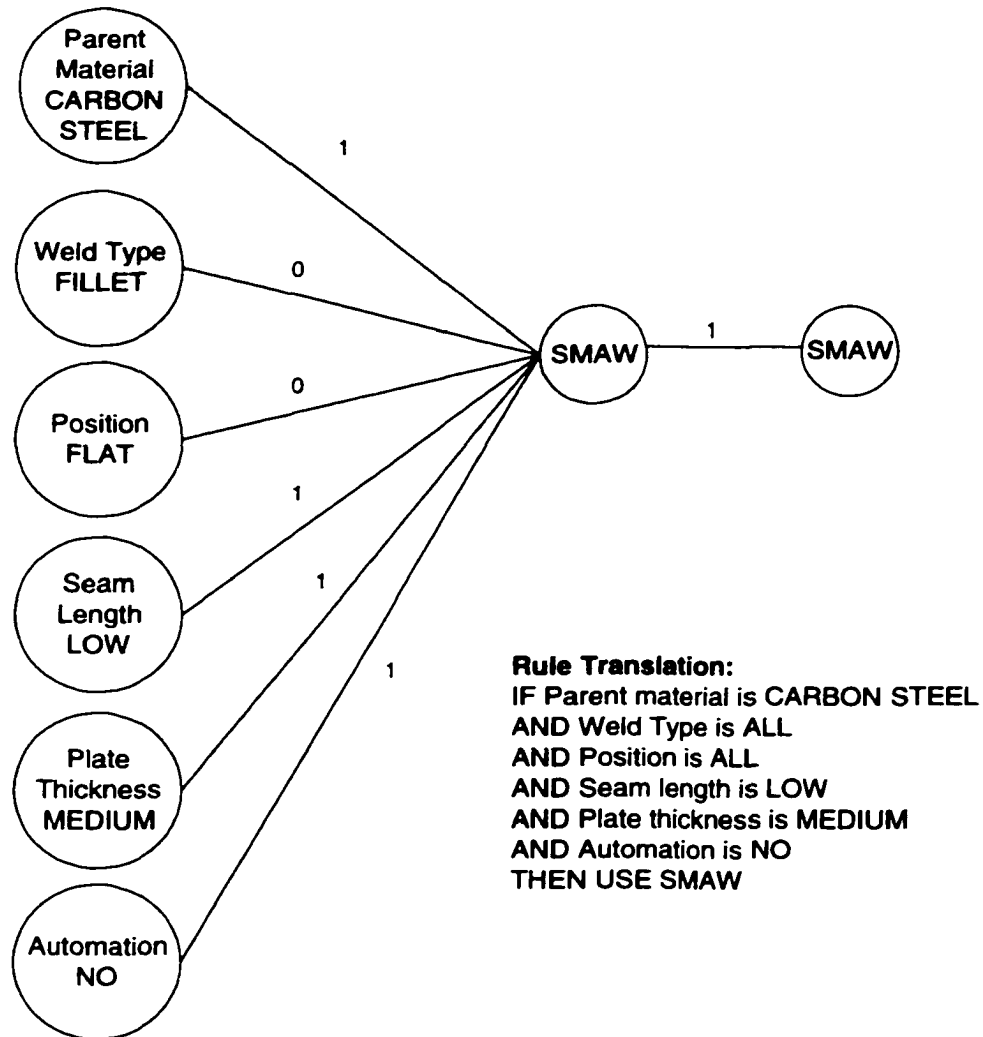


Figure 4.8 Example of Overall Positive Expert System Rule for SMAW

A lower CF implies that the system is not sure about that process as compared to the next applicable process in the hierarchy. Higher Confidence Factor is better as it means that the confidence for that process is higher as the next process has a much lower weight.

#### **4.1.2 Process Selector Neural Network**

The job variables serve as the input layer of the neural network and the output layer contains the four welding processes. The output node with the maximum output is the most suitable process to be used (Figure 4.9). The initial parameters used for the neural network are given in table 4.2. Each input node represents a possible attribute value.

The number of hidden nodes was kept the same as the output node for ease of translation of rules into link weights and vice versa. This is done to allow for a one-to-one mapping (figures 4.5 to 4.8) of rules to the weights. The initial weights were determined by using the expert system rules given in table 4.3.

The neural network was trained using back-propagation. Zero weights were assigned to links between job variables and the processes where all the processes were equally applicable for the job variable. This led to faster convergence of the neural network.

**Table 4.2. Initial Design Parameters for the Process Selector Neural Network**

Neural Network Parameter	Value
Input Nodes	32
Hidden Nodes	4
Output Nodes	4
Rate	0.2
Momentum	0.2
Tolerance	0.05

The training time was reduced from several days to a few hours on a Windows/NT platform. The mapping of job variable criteria weights from expert system rules to corresponding links in the neural network is given in table 4.3 and table 4.4. The criteria weight values were determined using heuristic rules and expert knowledge obtained from welding experts. Since there was a one-to-one mapping between expert system rules (criteria weights) and the neural network links, the difference in the initial weight and the final weight obtained was used to determine whether an adjustment to the criteria weight in the rule-base of the expert system was needed after the neural network is trained with new examples. This adjustment is applied to the links between input and hidden layer only as the rules are represented by these links only. There are no rules to be extracted from the links between hidden and output layer as the hidden nodes serve as the conjunction nodes for the various processes.

A modified form of Fu's (1993) algorithm was used for expert rule extraction and modification after comparing several different strategies. More details on this comparison are provided in chapter 5. Both the original Fu's weight extraction algorithm and the modified one are summarized below.

#### **Fu's Weight Extraction Algorithm**

$W_{ij}$  = Initial Weight of link between  $i^{\text{th}}$  neuron of the 1<sup>st</sup> layer and  $j^{\text{th}}$  neuron of the 2<sup>nd</sup> layer,

$M_{ij}$  = Modified weight of link between  $i^{\text{th}}$  neuron of the 1<sup>st</sup> layer and  $j^{\text{th}}$  neuron of the 2<sup>nd</sup> layer,

$N_{ij}$  = New weight of link between  $i^{\text{th}}$  neuron of the 1st layer and the  $j^{\text{th}}$  neuron of the 2nd layer,

$x$  = number of neurons in the 1<sup>st</sup> layer,  $y$  = number neurons in the 2<sup>nd</sup> layer .

For  $i = 1$  to  $x$  and  $j = 1$  to  $y$ , modify the criteria values in the following steps:

1. IF  $|M_{ij} - W_{ij}| > |W_{ij}|$ , then  $N_{ij} = M_{ij}$ .
2. IF  $|M_{ij} - W_{ij}| < |W_{ij}|$ , then  $N_{ij} = W_{ij}$ .

### **Modified Algorithm for Weight Extraction**

In the modified algorithm, the criteria values are modified as follows:

1. IF  $|M_{ij} - W_{ij}| > |W_{ij}|$  and  $W_{ij} \diamond 0$ , then  $N_{ij} = M_{ij}$ .
2. IF  $|M_{ij} - W_{ij}| < |W_{ij}|$  and  $|W_{ij}| \diamond 0$ , then  $N_{ij} = W_{ij}$ .
3. IF  $W_{ij} = 0$  and  $|M_{ij}| > 0.20$ , then  $N_{ij} = M_{ij}$ .
4. IF  $W_{ij} = 0$  and  $|M_{ij}| \leq 0.20$ , then  $N_{ij} = W_{ij}$ .

As you will notice that steps 3 and 4 have been added for the initial weights that were zero.

The cut-off percentage of weight change was decided to be 0.20 and it was derived empirically after several runs of the process selector with cut-off percentages of 10%, 20% and 30% (see chapter 5 for more details), which established that the accuracy and performance was best with 20% cut-off criteria. Also, there is a validation check step that takes place for all the processes in the selection set and the rules for that validation check have been provided in Appendix A.2.

Table 4.3 Derivation of Initial Weight Matrix from Expert System Rules for first layer

Expert System Rule/Criteria	Input Node to SMAW	Input Node to GMAW	Input Node to GTAW	Input Node to SAW	Explanation
parent_material_criteria("ALUMINUM",-1,1,1,1)	-1	1	1	1	SMAW is not suitable for Aluminum
parent_material_criteria("MILD STEEL",0,0,0,0)	0	0	0	0	All processes are suitable for Mild Steel
parent_material_criteria("CARBON STEEL",0,0,0,0)	0	0	0	0	All processes are suitable for Carbon Steel
parent_material_criteria("STAINLESS STEEL",0,0,0,0)	0	0	0	0	All processes are suitable for Stainless Steel.
parent_material_criteria("CAST IRON",0,0,0,0)	0	0	0	0	All processes are applicable for Cast Iron
weld_type_criteria("BUTT",0,0,0,0)	0	0	0	0	All processes are suitable for Butt Welding
weld_type_criteria("FILLET",0,0,0,0)	0	0	0	0	All processes are suitable for Fillet Welding
position_criteria("FLAT",0,0,0,0)	0	0	0	0	All processes are suitable for Flat Position
position_criteria("HORIZONTAL",0,0,0,0)	0	0	0	0	All processes are suitable for Horizontal Position
position_criteria("VERTICAL",1,1,1,-1)	1	1	1	-1	SAW cannot be utilized in Vertical Position
position_criteria("DOWNWARD",1,1,1,-1)	1	1	1	-1	SAW is not applicable in Downward Position
position_criteria("OVERHEAD",1,1,1,-1)	1	1	1	-1	SAW is not applicable in Overhead Position
seam_length_criteria("HIGH",-1,1,1,1)	-1	1	1	1	SMAW is not suitable for high seam length
seam_length_criteria("LOW",0,0,0,0)	0	0	0	0	All processes are suitable for low seam length
plate_thickness_criteria("HIGH",-1,1,-1,1)	-1	1	-1	1	SMAW and GTAW are less suitable for high plate thickness
plate_thickness_criteria("MEDIUM",0,0,0,0)	0	0	0	0	All processes are suitable for medium plate thickness
plate_thickness_criteria("LOW",1,1,1,-1)	1	1	1	-1	SAW is not suitable for low plate thickness
strength_criteria("LOW",0,0,0,0)	0	0	0	0	All processes are applicable for low strength requirements
strength_criteria("HIGH",0,0,0,0)	0	0	0	0	All processes are applicable for HIGH strength requirements
appearance_criteria("FLAT",0,0,0,0)	0	0	0	0	All processes are applicable for Flat Bead Shape
appearance_criteria("CONVEX",0,0,0,0)	0	0	0	0	All processes are applicable for Convex Bead Shape
appearance_criteria("CONCAVE",0,0,0,0)	0	0	0	0	All processes are applicable for Concave Bead Shape
deposition_criteria("LOW",0,0,0,0)	0	0	0	0	Deposition can be low for all processes
deposition_criteria("MEDIUM",0,0,0,0)	0	0	0	0	Deposition can be medium in all processes
deposition_criteria("HIGH",-1,1,-1,1)	-1	1	-1	1	GMAW and SAW are more suitable for high deposition
weld_quality_criteria("LOW",0,0,0,0)	0	0	0	0	All processes are applicable for LOW quality
weld_quality_criteria("MEDIUM",0,0,0,0)	0	0	0	0	All processes are applicable for MEDIUM
weld_quality_criteria("HIGH",-1,1,-1,1)	-1	1	-1	1	All processes are applicable for HIGH quality requirements
automation_criteria("NO",1,1,1,-1)	1	1	1	-1	SAW is not applicable if No Automation is
automation_criteria("YES",-1,1,1,1)	-1	1	1	1	If Automation required then SMAW is not applicable.
leg_length_criteria("LOW",0,0,0,0)	0	0	0	0	All processes are applicable for low leg length
leg_length_criteria("HIGH",0,1,0,1)	0	1	0	1	GMAW and SAW are more suitable for high leg length

Table 4.4 Initial Weight Matrix for the Process Parameter Neural Network for second layer

Hidden Node	SMAW Output Node	GMAW Output Node	GTAW Output Node	SAW Output Node	Comment
Hidden Node 1 (SMAW)	1	0	0	0	Since the first hidden and output node are for SMAW, the weight assigned is 1 for SMAW and zero for rest of the processes
Hidden Node 2 (GMAW)	0	1	0	0	Since the second hidden and output node are for GMAW, the weight assigned is 1 for GMAW and zero for rest of the processes
Hidden Node 3 (GTAW)	0	0	1	0	Since the third hidden and output node are for GTAW, the weight assigned is 1 for GTAW and zero for rest of the processes
Hidden Node 4 (SAW)	0	0	0	1	Since the fourth hidden and output node are for GMAW, the weight assigned is 1 for GTAW and zero for rest of the processes

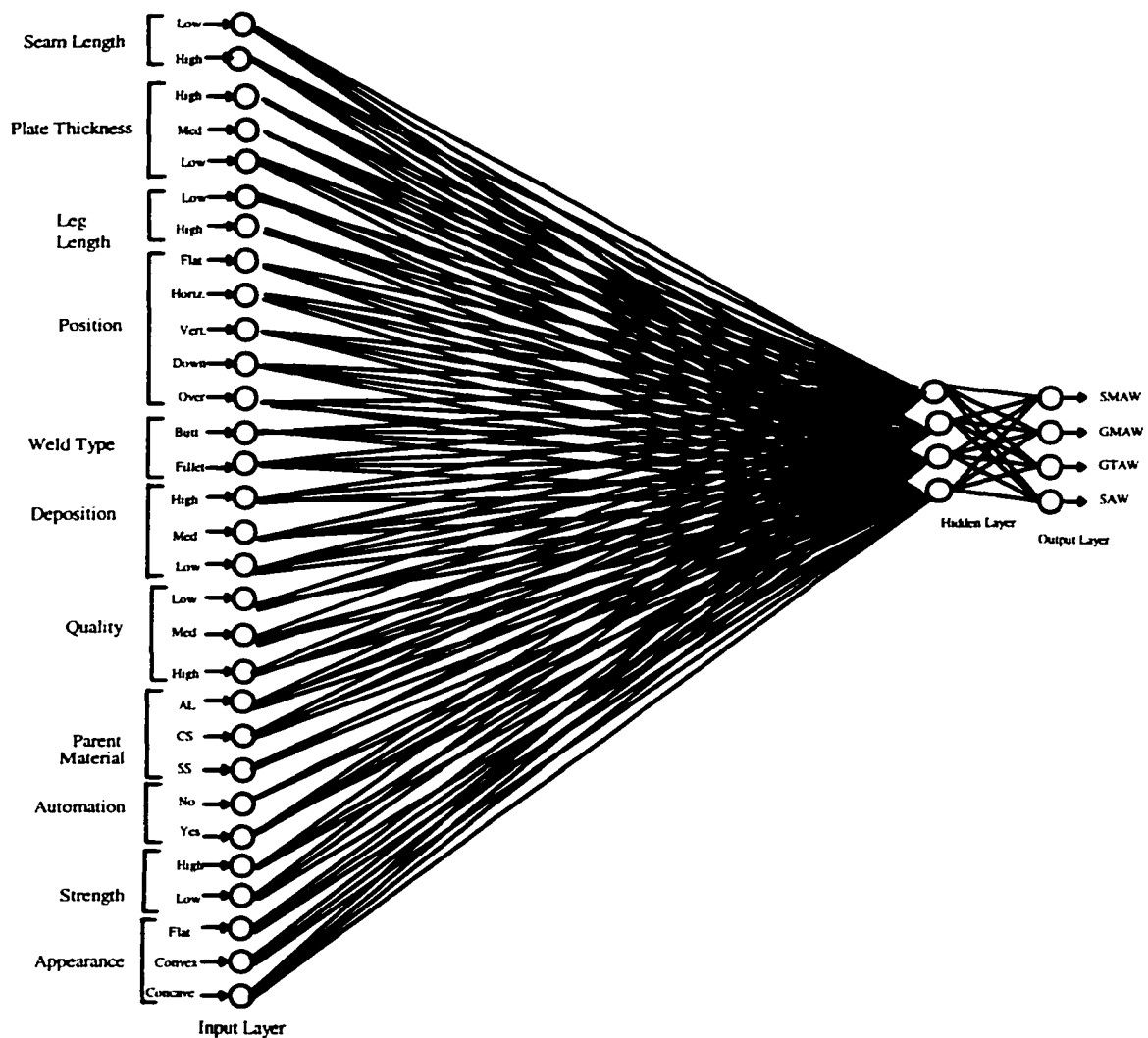


Figure 4.9 Structure of the Neural Network for Process Selection

## 4.2 Parameter Selector

An *expert system* combined with a database query system was used for this module. For each applicable process, the expert system selects the electrode based on the job variables specified. A representative set of sample rules used by the expert system for electrode selection are given in appendix A (table A.3). The database query system then selects the process parameters to be used for the set of applicable welding processes by querying a database. This database contains records (see section 4.2.1) with various combinations of job variables (including electrode specification) and the corresponding process parameters for those combinations of job variables. This parameter database is created from standard data available from arc welding standard data handbook (1975). The applicable process(es) and their corresponding process parameters provide input to the economic evaluator to determine their cost and time.

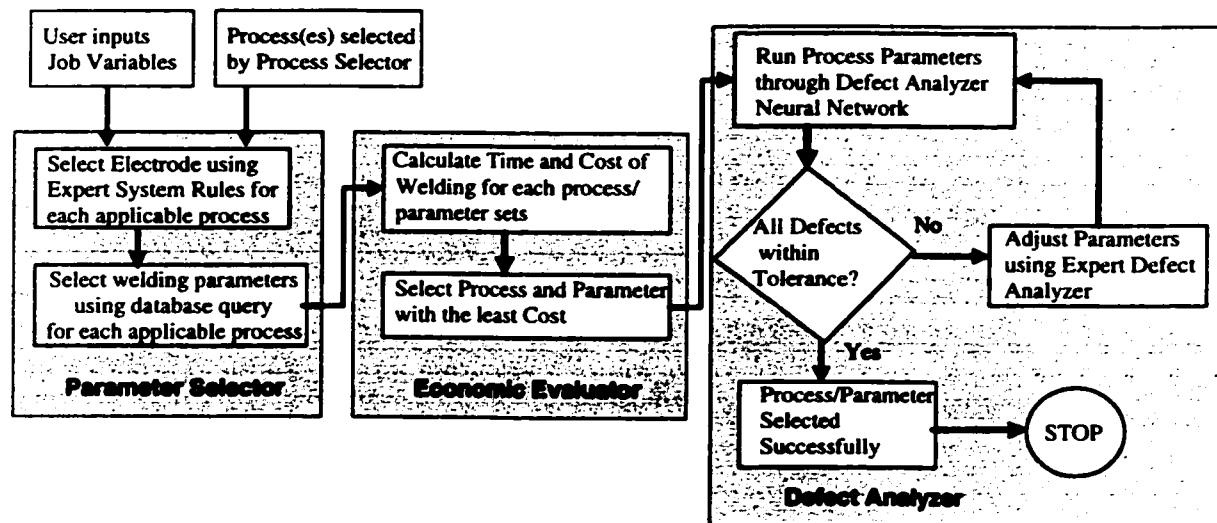


Figure 4.10 Flowchart for the Parameter Selection and Adjustment

Fig. 4.10 shows the process flow between three modules of the system: parameter selector, economic evaluator, and defect analyzer.

#### **4.2.1 Knowledge Representation for the Parameter Selector**

The database for the parameter selector is stored using the internal database feature of PDC Prolog. The database contains standard data in various fields and the system uses a search & match strategy to determine the process parameters. A sample record of the internal database is:

*std\_data("SMAW","Fillet",5,"Flat",2,4.0,120,0.153,4.2,1.61,6.7)*

The format of the record is:

*std\_data(PROCESS,WELD TYPE,LEG LENGTH,POSITION,RUNS,ELECTRODE  
SIZE,CURRENT,WEIGHT PER METER,NUMBER OF ELECTRODES,  
OVERHEAD TIME,ARC TIME)*

The same database is utilized for the economic evaluator module also.

#### **4.3 Economic Evaluator**

A procedural system approach is utilized for calculating the cost and time for welding. The time and electrode consumption data is stored in standard data files. The cost data for electrodes and power is stored in the company data files along with overhead and labor cost data. All this information along with the information on selected process and process parameters is utilized to calculate the time and cost for welding.

The economic evaluator is a module that computes the cost and time spent for welding using a process and a set of welding parameters for that process. The number of



electrodes consumed, arc time, overhead time etc. are derived from standard data stored in the knowledge base of the system. The economic evaluator computes the cost for welding for the different set of welding process and parameters recommended by process selector and parameter selector and the set of welding process and parameter that results in the least cost is recommended.

The final process parameters selected by the economic evaluator serves as an input to the neural network of the defect analyzer which determines whether these process parameters result in satisfactory weld or not.

#### **4.3.1 Economic Evaluator Cost Computation**

CE = Cost in dollars per electrode unit (unit = number for SMAW and meter for GMAW),

NE = Number of electrodes consumed per meter of weld seam

AT = Arc time (minutes per meter of weld), OT = Overhead time (minutes per meter of weld),

CM = Material cost per meter of weld (\$), CLO = Cost of labor/Overhead per min. (\$)

CPM = Cost of operating the welding set per minute, PC = Power cost in kWh (\$),

COST = Total Cost (\$) of Welding per meter of Weld

Then  $CM = CE \times NE$ ,

$$CL = CLO \times (AT + OT)$$

$$CPM = PC \times (\text{Welding current} \times \text{Arc voltage} \times \text{Arc time in minutes})/60000$$

$$\text{COST (\$/meter weld)} = CM + CL + CPM$$

#### **4.4 Defect Analyzer Design - A Coupled System**

The purpose of the defect analyzer is to predict if any defects will be there for a certain set of process parameters for welding a job. If any defects are predicted, then the adjustable welding parameters are adjusted using a rule-based parameter adjustment module and the WPS is modified to minimize the defects.

The defect analyzer has several modules that share information using data files (see figure 4.110). The process parameters selected by economic evaluator module serve as input to the system. The parameter adjustment module then converts them into an input file for the defect predictor neural network which predicts the defects and stores them in an output file. The parameter adjustment module interprets the output file and creates a defects list and uses its internal rule base to adjust the process parameters.

The WPS tips generation module modifies the welding procedure specification using its rule base to minimize the predicted defects. The knowledge extraction module can analyze the weight changes and modify the rule base to improve the performance of the parameter adjustment module but it has not been implemented in this research.

##### **4.4.1 Knowledge Representation and Categorization for Defect Analyzer**

Some of the user-defined parameters like position, leg length, plate thickness, strength requirements cannot be changed as they are user-defined job variables and hence they are not mapped to the input nodes. Also, during the process and parameter selection phases, these parameters have already been considered and so the process and parameters selected support the user input parameters.

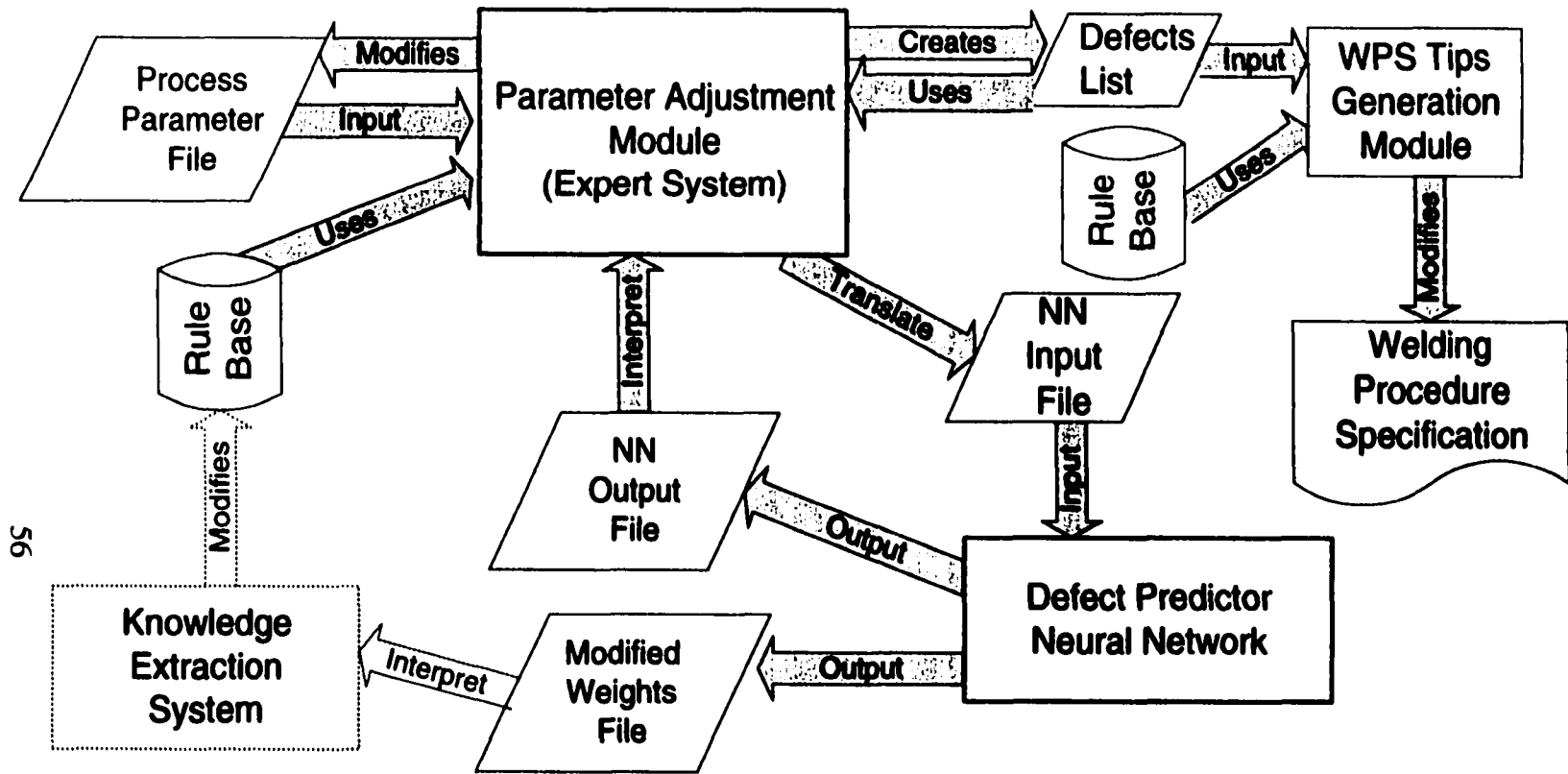


Figure 4.11 Detailed Data Flow in the Defect Analyzer

Some of the variables are continuous (like current, speed) and others are discrete (like electrode specification). Hence, they are categorized into broad categories to avoid the neural network from becoming too large. Also, defects have dependencies on these categories of parameters. For example, defects depend on the type of coating used in the electrode and so the electrodes have been classified by their coating in the SMAW defect analyzer. Most of the defect remedies for adjusting electrode involve going from one category to another. For example, in case of porosity, low hydrogen electrodes are recommended to avoid hydrogen pockets. For SMAW, electrodes have been categorized into four categories: low hydrogen, cellulosic, rutile and basic. Table 4.5 categorizes the various electrodes into these four categories.

**Table 4.5 Electrode Classification for Defect Analyzer**

<b>Electrode Coating Categorization</b>	<b>AWS Specification of Electrode</b>
<b>Low Hydrogen</b>	<b>E6016, E7016</b>
<b>Cellulosic</b>	<b>E6010, E6011, E7010, E7011</b>
<b>Rutile</b>	<b>E6012, E6013, E7013, E8013</b>
<b>Basic</b>	<b>E6030, E7030</b>

For GMAW, since the electrode has the same composition as the job material, only electrode size has been considered as an electrode input parameter. The parameter adjustment subsystem using the neural network has been designed to adjust weld parameters for SMAW and GMAW currently, even though the defect analyzer expert system can handle all four processes - SMAW, GMAW, GTAW and SAW.

If the weld quality predicted by the neural network is not good then the parameter modifier of the defect analyzer modifies the process parameters till all the defects are predicted to be within acceptable tolerance. The inspection system can then be used after welding to determine the weld quality. If the actual weld quality is observed to be improper after actually carrying out the welding, then the neural network is retrained, and the database of the parameter selector is updated. The defect analyzer has a relationship with the WPS Generator as adjustments made by defect analyzer to the weld procedure specification eliminate or reduce the number of defects.

#### **4.4.2 Defect Analyzer Neural Network**

The design of the neural network is done by identifying the domain knowledge about various defects and the parameters that affect them. This knowledge is used to map input nodes, hidden nodes and output nodes to process parameters, intermediate hypothesis and defects so as to develop a neural network. To obtain the training data, welding can be performed using a set of weld parameters suggested by the parameter selector module for various jobs. A visual and non-destructive examination inspection of the weldment can be done for defects like undercuts, lack of fusion, weld spatter, cracks etc. and then the decision is taken whether the weld is acceptable or not. Thus a set of sample data can become available for input / output patterns for defect analysis. The neural network is then trained on this set of sample data. After training the neural network completely, the neural network is used to predict weld defects for a set of process parameters.

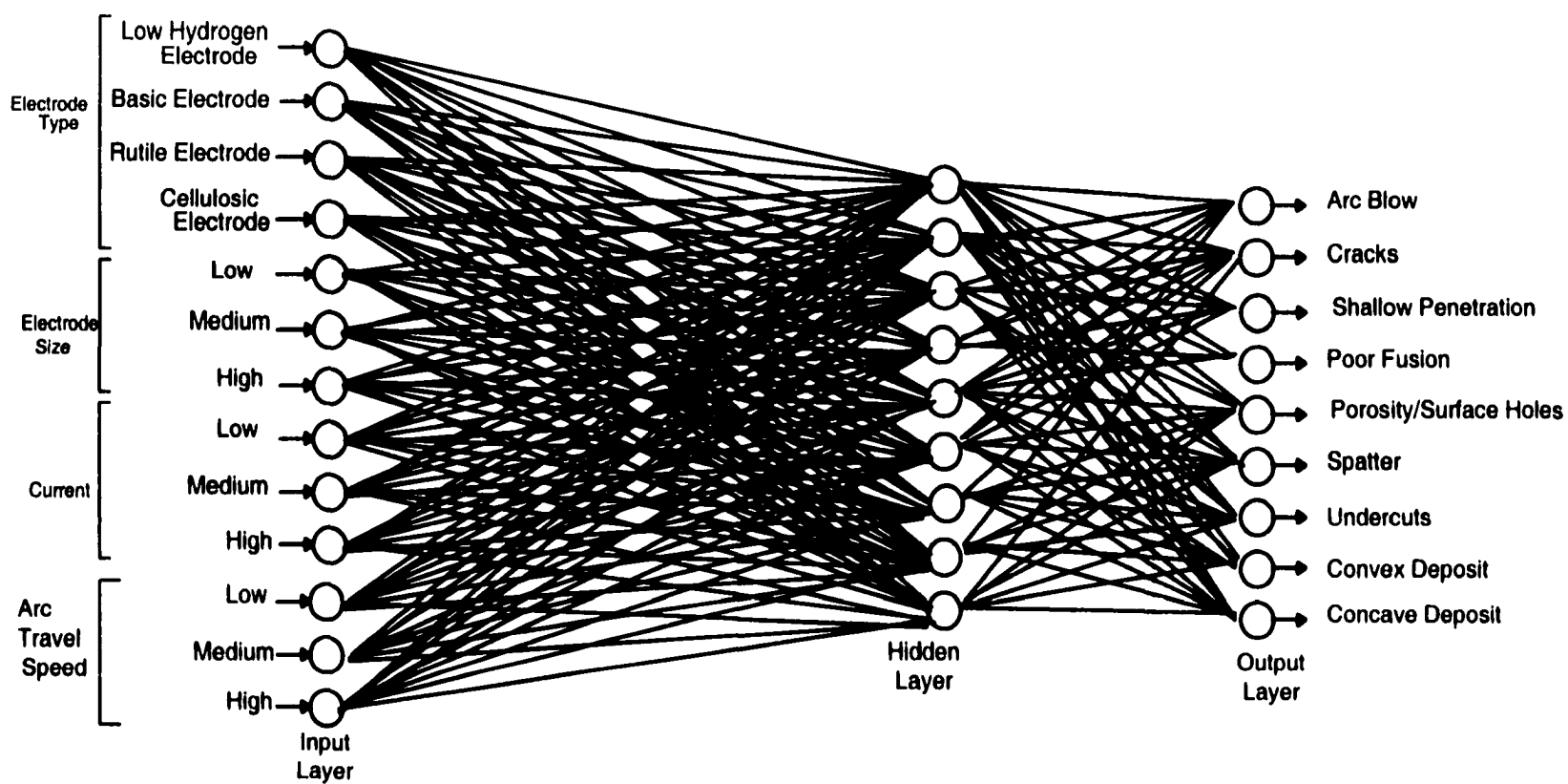


Figure 4.12 Neural Network Configuration for SMAW Defect Analyzer

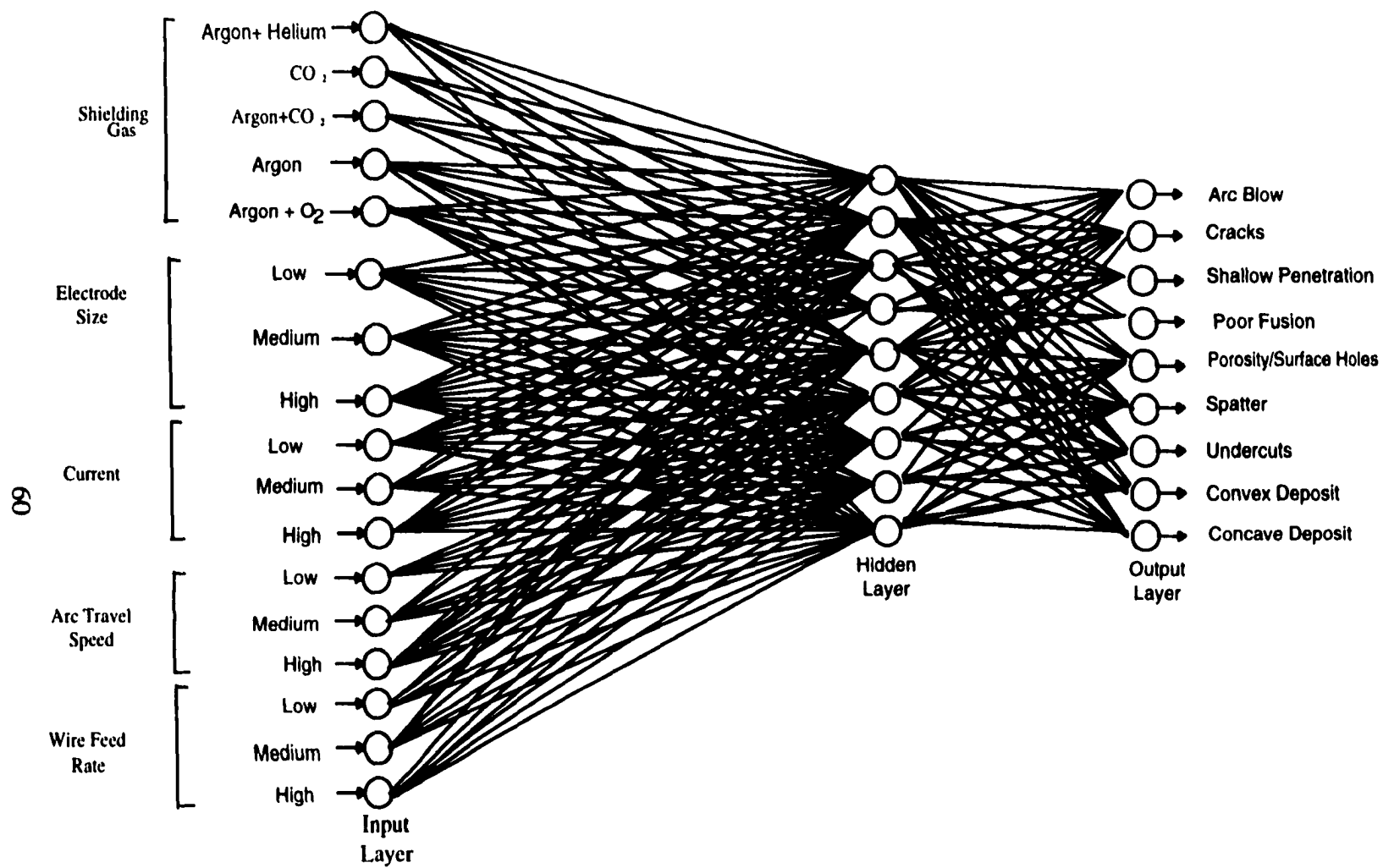


Figure 4.13 Neural Network Configuration for GMAW Defect Analyzer

The topology of the neural networks for SMAW and GMAW is provided in figures 4.11 and 4.12. The process parameters are mapped to the input node while the defect types are mapped to the output nodes. The number of hidden nodes is kept the same as the output node for ease of interpretation of the weight changes as each hidden node is treated as a conjunctive node for the defects.

The process parameters have been categorized so that the parameter adjustment is easier and the size of the neural network does not become very large, as described in section 4.4. Heuristic knowledge is used to categorize these parameters and the expert system handles the translation of the parameters into these categories and the interpretation of the output of the neural network. But since the neural network can be used in stand-alone mode also, the overall defect analyzer system is a loosely coupled system as the expert system reads from output files and writes into the input files of the neural network. A separate neural network is utilized for GMAW defect analysis as the input process parameters and the output defects are different.

#### **4.4.2.1 Initial Weight Assignment in Defect Analyzer Neural Network**

It is assumed that the objective is to minimize the output node activations and bring them as close to zero as possible. Positive weights are assigned to links that increase the defects, negative weights are assigned to links that reduce the defect and zero weights are assigned if there is no known correlation between the defect and the parameter. This weight assignment is done for SMAW in table 4.6 and GMAW in table 4.7 using expert knowledge obtained from handbooks [The Procedure Handbook of Arc Welding (1973)].



#### **4.4.2.2 Network Training**

Similar to the process selector, the neural network was trained with the back-propagation algorithm using data that has been provided in appendix C - section C.3. Since we did not have a welding facility available this data was obtained using handbook data and making adjustments to the parameters so that certain defects will be introduced. For e.g. the current was increased and then spatter and slag inclusions were assumed to be there in the weld.

#### **4.4.2.3 Weight Modification and Knowledge Extraction**

A weight modification and knowledge extraction strategy similar to process selector module can be utilized. As we will discuss in chapter 7, the implementation of the prolog program for weight extraction and modification of rules is outside the scope of this research. The newly learned rules and weight modification analysis is provided in chapter

#### **4.4.3 Rule-Based Parameter Adjustment Strategy in Defect Analyzer**

A rule-based strategy was utilized for adjusting the parameters in defect analyzer because of the following reasons:

1. All defects do not have dependencies on all parameters. If neural network based optimization is utilized for this purpose then the parameters that do not have dependencies may get adjusted leading to even more defects.
2. Certain input parameters have to be discrete. For example, shielding gas or electrode type have to be either 0 or 1 and only one node is allowed to have 1 as the input. It is feasible to apply such constraints in rule based systems only.

Table 4.6 Initial Weight Matrix for SMAW Defect Analysis from Expert Knowledge

Parameter	Arc Blow	Shallow Penetration	Poor Fusion	Porosity	Spatter	Undercut	Cracks	Slag Inclusion	Expert Knowledge
Low H2 electrode	0	-0.5	-0.5	-1	0.5	0	-1	0.5	Low Hydrogen electrodes causes some spatter and slag inclusions and so weight assigned is 0.5. They prevent shallow penetration, poor fusion and cracking and so weight assigned is -0.5,-0.5 and -1. No
Basic Electrode	0	1	1	1	1	0	1	1	Basic Electrodes cause shallow penetration, poor fusion and slag inclusion and so the weight assigned is 1. Some porosity and cracks is there and so weight assigned is 0.5. No effect on arc blow and undercuts
Rutile Electrode	0	-1	-1	1	-1	0	1	-1	Rutile electrode cause some porosity and cracks and so weight assigned is 0.5. It provides good penetration, good fusion and no spatter or slag inclusions and so weight assigned is -1. No effect on arc
Cellulosic Electrode	0	-1	-1	1	1	0	1	1	Cellulosic Electrodes cause porosity, cracks, and slag inclusions and so -1 weight is assigned. Some spatter is there and so 0.5 weight is assigned. Good penetration and fusion is there and hence -1 weight
Electrode Size Low	-1	-1	-1	-1	0	-1	-1	-1	Low Electrode size reduces all defects except spatter and so -1 weight is assigned. No effect on Spatter and so weight assigned is zero.
Electrode Size Med.	0	0	0	0	0	0	0	0	No effect on other defects and so weight assigned is zero.
Electrode Size High	1	1	1	1	0	1	1	1	High Electrode size increases chances of all defects except spatter and so +1 weight is assigned to these nodes. No effect on spatter and so zero weight is assigned.
Current Low	-1	1	1	-1	-1	-1	0	-1	Low Current may cause shallow penetration and poor fusion and so weight assigned is +1. Low Current prevents arc blow, spatter, undercuts, porosity and slag inclusions and so weight assigned is -1. No
Current Medium	0	0	0	0	0	0	0	0	No effect on any defects and so weight assigned is zero.
Current High	1	-1	-1	1	1	1	1	1	High Current causes arc blow, spatter and undercuts and so weight assigned is +1. It causes some porosity and so +0.5 weight is assigned. High current prevents shallow penetration, poor fusion and slag

Table 4.7 Initial Weight Matrix for GMAW Defect Analysis from Expert Knowledge

Parameter	Arc Blow	Shallow Penetration	Poor Fusion	Porosity	Spatter	Undercut	Cracks	Convex Bead	Concave Bead	Expert Knowledge
Ar+ He Shielding Gas	0	-0.5	-1	-1	-1	0	0	0	0	Ar+He provides good penetration and fusion and low porosity and Spatter and hence -1 weights are assigned to those links. No effect on other defects and so weight assigned is zero.
Ar + CO2 Shield. Gas	0	-0.5	-1	-1	0.5	0	0	0	0	Ar+CO2 provides good penetration and fusion and low porosity hence -1 weights are assigned to these links. It does cause some spatter an so 0.5 weight is assigned to Spatter. No effect on other defects and so weight
Ar Shielding Gas	0	1	1	-1	-1	0	0	0	0	Ar provides less penetration and fusion hence +1 weight is assigned to these links. It does not cause spatter or porosity and so -1 weight is assigned to them. No effect on other defects and so weight assigned is zero.
Ar + O2 Shield. gas	0	0.5	0.5	0	-1	0	0	0	0	Ar+O2 provides less penetration and fusion hence 0.5 weight is assigned to these links. It does not cause spatter and so -1 weight is assigned to it. No effect on other defects and so weight assigned is zero.
Electrode Size Low	-1	-1	-1	-1	0	-1	0	0	0	Low Electrode size causes less arc blow, porosity or undercuts and provides good penetration and so -1 weight is assigned to these nodes. No effect on other defects and so weight assigned is zero.
Electrode Size Med.	0	0	0	0	0	0	0	0	0	No effect on other defects and so weight assigned is zero.
Electrode Size High	1	1	1	1	0	1	0	0	0	High Electrode size causes arc blow, porosity, undercuts, poor fusion and shallow penetration and so +1 weight is assigned to these nodes.No effect on other defects
Current Low	-1	1	1	-1	-1	-1	0	-1	1	Low Current may cause shallow penetration, poor fusion and concave bead and so weight assigned in +1. Low Current prevents arc blow, spatter, undercuts and convex beads and so weight assigned is -1. No effect on
Current Medium	0	0	0	0	0	0	0	0	0	No effect on any defects and so weight assigned is zero.
Current High	1	-1	-1	0.5	1	1	0	1	-1	High Current causes arc blow, spatter, undercuts and convex bead and so weight assigned is +1. It also causes some porosity and so +0.5 weight is assigned. High current prevents shallow penetration, poor fusion and
Travel Speed Low	0	-1	-1	-1	0	-1	-1	1	-1	Low arc travel causes convex bead and so weight assigned is 1. No effect on spatter or arc blow and so weight assigned is 0. -1 is assigned for the rest of the defects as they are minimized by low arc travel speed.
Travel Speed Med.	0	0	0	0	0	0	0	0	0	No effect on any defects and so weight assigned is zero.
Travel Speed High	0	1	1	0.5	0	1	1	-1	1	Causes shallow penetration, poor fusion, undercuts, cracks and concave bead and so weight assigned is 1, Causes some porosity and so weight assigned is 0.5. No effect on spatter hence 0 weight for it and prevents concave bead and so weight assigned is -1.
Wire Feed Rate Low	0	-1	-1	-1	0	-1	-1	-1	1	Low arc travel causes concave bead and so weight assigned is 1. No effect on spatter or arc blow and so weight assigned is 0. -1 is assigned for the rest of the defects as they are minimized by low wire feed rate.
Wire Feed Rate Med.	0	0	0	0	0	0	0	0	0	No effect on any defects and so weight assigned is zero.
Wire Feed Rate High	0	1	1	0.5	0	1	1	1	-1	Causes shallow penetration, poor fusion, undercuts, cracks and convex bead and so weight assigned is 1, Causes some porosity and so weight assigned is 0.5. No effect on spatter hence 0 weight for it and prevents concave bead and so weight assigned is -1.

3. The parameters have to be adjusted within a certain range depending on other parameters and hence neural network approach will not be feasible.

For example, the range of current will be 80-120 amperes for a 1.5 mm electrode while 150-250 amperes for a 5 mm electrode. Thus the input parameters have dependencies on each other and these dependencies can be captured in rule based framework only.

#### **4.4.3.1 Design of the Parameter Adjustment Module of Defect Analyzer**

The adjustments are made to the parameter as per a priority list (based on heuristics and thumb-rules) for each defect. This priority list (given in the next section) ensures that the parameters are adjusted in the most effective way. For example, electrode/shielding gas category change has the lowest priority as the parameter selector has already taken the job requirements into account during parameter selection. Current has the highest priority, as it is easy to adjust it.

The flow-chart depicting the logic used by the parameter adjustment module is shown in Figure 4.14.

The pseudo code for the parameter adjustment sub-module of the defect analyzer has been provided in Appendix B - Section B.3. The detailed algorithm used for this module is given in chapter 7.

The constraints under which the parameter adjustment module of the defect analyzer has to work are as follows.

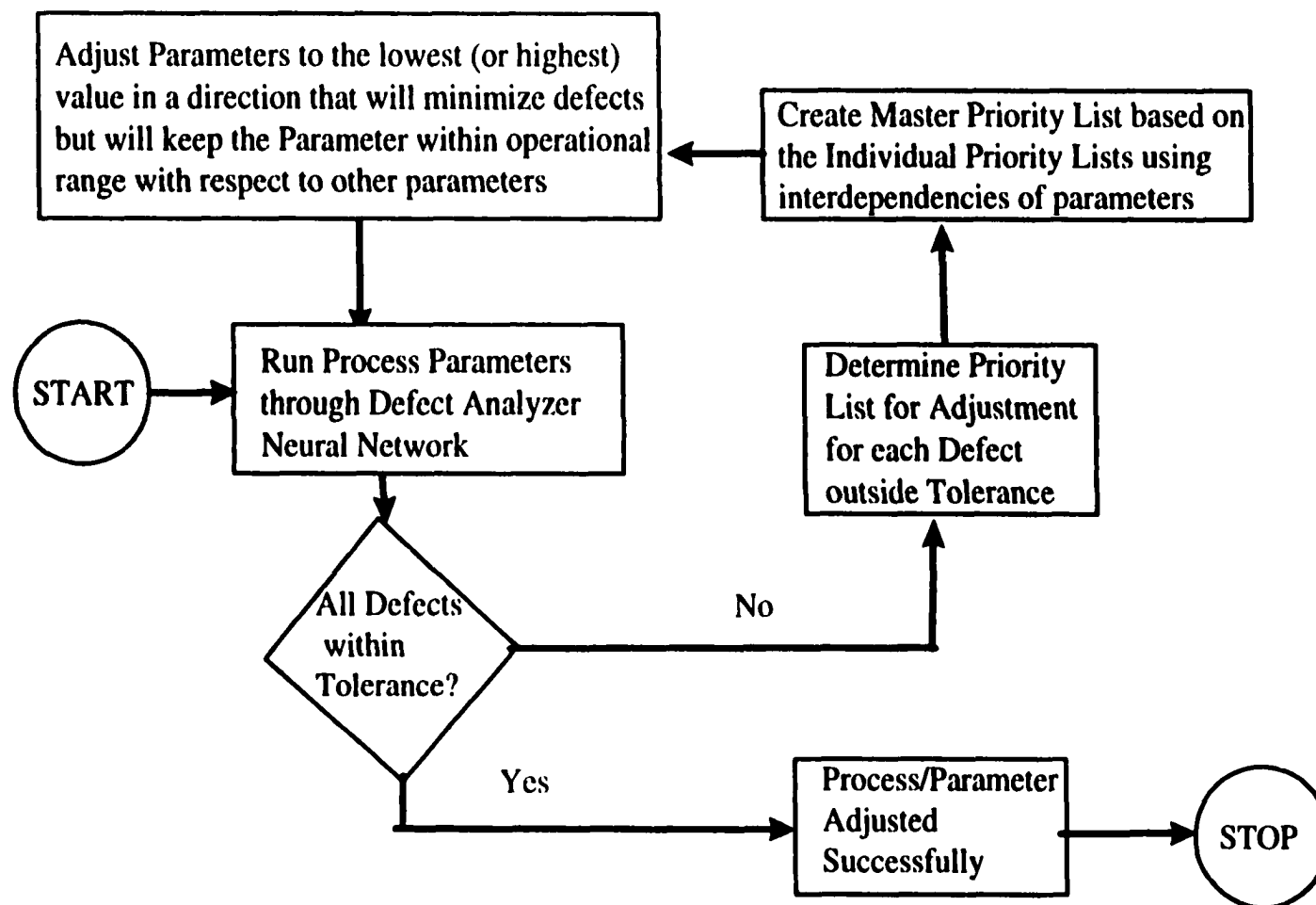


Figure 4.14 Rule-Based Parameter Adjustment Strategy in Defect Analyzer

#### Constraints for a fixed set of volatile<sup>1</sup> parameters

V1 <= VOLTAGE <= V2 (Minimum and maximum voltage capacity)

I1 <= Current <= I2 (Minimum and maximum current capacity)

S1 <= Arc travel speed <= S2 (Minimum and maximum speed capacity)

l <= Number of runs, Position = P, Leg length = L

The constraints may vary if a volatile parameter is modified.

Further details on parameter adjustment are provided in the chapter 7.

#### **4.4.3.2 Design of the WPS Tips Generator Module of Defect Analyzer**

After the parameter adjustment module has adjusted the parameters to the best possible combination, the WPS tips generator module (for sample runs of the system see chapter 7) of the defect analyzer is used for creating some WPS instructions for the welder that will minimize the predicted defect(s). These WPS remedies are shown in tables 4.7 and 4.8 for SMAW and GMAW respectively.

The WPS tips generation module is an expert system that uses its knowledge base (3<sup>rd</sup> column in tables 4.7 and 4.8) to create a list of procedures to be used for avoiding the predicted defects. It takes the list of the defects and the process as input and produces a list of procedures as the output. Some sample rules have been provided in section A.6 of appendix A. A sample rule is as follows:

---

<sup>1</sup> Parameters that affect a change in other parameters. e.g. Electrode Size, Electrode Type, Shielding Gas

*IF defect\_predicted = ("Porosity" AND/OR "Cracking")*  
*AND Process = "SMAW"*  
*THEN wps\_tip1 = "Bake Electrodes before Use"*  
*AND wps\_tip2 = "Clean the Plates thoroughly with a wire brush"*  
*AND wps\_tip3 = "Provide Pre-heating and Interpass Heating"*  
*AND wps\_tip4 = "Use Weaving Techniques"*  
*AND wps\_tip5 = "Back-step to fill craters"*

#### 4.3.4 Interactive Defect Analyzer

Besides parameter adjustment and WPS tips generation, the defect analyzer can be used interactively by the welder to diagnose defects in real time. The interactive defect analyzer module contains IF-THEN-ELSE rules for identification of the probable causes of the predicted defects and suggests remedial measures. A sample rule for the interactive defect analyzer module is given below:

<i>IF</i>	<i>Defect detected</i>	<i>Undercuts</i>
<i>AND</i>	<i>Welding Process</i>	<i>Shielded Metal Arc Welding</i>
<i>THEN</i>	<i>Causes may be</i>	<i>Too large electrode diameter</i>
		<i>Higher currents</i>
		<i>Longer arcs</i>
		<i>Faster arc travel speeds</i>
	<i>Remedial Measures</i>	<i>Reduce Current</i>
		<i>Reduce arc travel speed</i>
		<i>Use smaller electrode size</i>

The difference between WPS tips generator module of Defect Analyzer and interactive defect analyzer is that the former uses the process parameter information to predict a defect and then generate the tips into the WPS while the latter is used in real-time after observing the defect in actual welding. So in essence, the former is proactive and while the later is interactive and is useful in actual welding defect diagnosis.

**Table 4.8 Parameter Adjustment Priority List for each Defect in SMAW**

<b>Defect</b>	<b>Parameter Adjustment Remedies</b>	<b>WPS Generator Remedies.</b>
Arc Blow	<ol style="list-style-type: none"> <li>① Reduce Current</li> <li>② Reduce Electrode Size</li> </ol>	<ol style="list-style-type: none"> <li>1. Use AC Welding Machine.</li> <li>2. Adjust Electrode Angle.</li> <li>3. Rearrange or split ground clamp.</li> <li>4. Use brass or copper backup bar.</li> </ol>
Shallow Penetration	<ol style="list-style-type: none"> <li>① Reduce welding Speed</li> <li>② Increase Current</li> <li>③ Decrease Electrode Size</li> </ol>	<ol style="list-style-type: none"> <li>1. Adjust root opening.</li> <li>2. Adjust root-face dimensions.</li> <li>3. Adjust joint included angle.</li> </ol>
Poor Fusion	<ol style="list-style-type: none"> <li>① Reduce welding Speed</li> <li>② Increase Current</li> <li>③ Decrease Electrode Size</li> </ol>	<ol style="list-style-type: none"> <li>1. Adjust root opening.</li> <li>2. Adjust root-face dimensions .</li> <li>3. Adjust joint included angle.</li> </ol>
Porosity	<ol style="list-style-type: none"> <li>① Reduce welding Speed</li> <li>② Decrease Current</li> <li>③ Decrease Electrode Size</li> </ol>	<ol style="list-style-type: none"> <li>1. Bake electrode before use.</li> <li>2. Clean the plates with wire brush.</li> <li>3. Pre-heat and post heat .</li> <li>4. Use weaving technique.</li> <li>5. Back-step to fill craters</li> </ol>
Spatter	<ol style="list-style-type: none"> <li>① Reduce Current</li> <li>② Reduce Electrode Size</li> <li>③ Change Electrode</li> </ol>	<ol style="list-style-type: none"> <li>1. Use AC Welding Machine.</li> <li>2. Adjust Electrode Angle.</li> <li>3. Rearrange or split ground clamp.</li> <li>4. Use brass or copper backup bar.</li> </ol>
Undercut	<ol style="list-style-type: none"> <li>① Reduce current</li> <li>② Reduce welding speed</li> <li>③ Change electrode size</li> </ol>	<ol style="list-style-type: none"> <li>1. Shorten arc length.</li> <li>2. Change angle of electrode so that arc force helps fill undercut.</li> </ol>
Cracks	<ol style="list-style-type: none"> <li>① Reduce welding speed</li> <li>② Reduce electrode size</li> <li>③ Change electrode to low H<sub>2</sub></li> </ol>	<ol style="list-style-type: none"> <li>1. Bake electrode before use.</li> <li>2. Clean the plates with wire brush.</li> <li>3. Pre-heat and post heat .</li> <li>4. Use weaving technique.</li> <li>5. Back-step to fill craters</li> </ol>
Slag Inclusion	<ol style="list-style-type: none"> <li>① Increase welding current</li> <li>② Reduce welding speed</li> <li>③ Change electrode</li> </ol>	<ol style="list-style-type: none"> <li>1. Proper deslagging between passes</li> <li>2. Use preheat and higher weld heat input</li> <li>3. Avoid joint design contour difficult to penetrate with arc.</li> </ol>



**Table 4.9 Parameter Adjustment Priority List for each Defect in GMAW**

Defect	Parameter Selector Remedies	WPS Generator Remedies
Arc Blow	① Reduce Current ② Reduce Electrode Size	1. Adjust Electrode Angle 2. Rearrange or split ground clamp 3. Use brass or copper backup bar
Shallow Penetration	① Reduce welding Speed ② Increase Current ③ Decrease Electrode Size	1. Adjust root opening 2. Adjust root-face dimensions 3. Adjust joint included angle 4. Reduce Arc Length.
Poor Fusion	① Reduce welding Speed ② Increase Current ③ Decrease Electrode Size	1. Adjust root opening 2. Adjust root-face dimensions 3. Adjust joint included angle
Porosity	① Reduce welding Speed ② Decrease Current ③ Decrease Electrode Size	1. Bake electrode before use. 2. Clean the plates with wire brush. 3. Pre-heat and post heat . 4. Use weaving technique. 5. Back-step to fill craters
Spatter	① Reduce Current ② Reduce Electrode Size ③ Change Electrode	1. Adjust Electrode Angle 2. Rearrange or split ground clamp 3. Use brass or copper backup bar
Undercut	① Reduce Current ② Reduce Welding Speed ③ Change Electrode Size	1. Shorten Arc Length 2. Change angle of electrode so that arc force helps fill undercut
Cracks	① Reduce Welding Speed ② Change Shielding Gas	1. Change Gun Angle to improve deposition. 2. Change joint design to reduce rigidity. 3. Pre-heat and Post-Heat 4. Use Weaving Technique 5. Bake electrode before use. 6. Clean the plates with wire brush. 7. Use weaving technique. 8. Back-step to fill craters
Convex Bead	① Decrease Welding Current ② Increase Welding Speed	
Concave Bead	① Increase Welding Current ② Decrease Welding Speed	

## 4.5 Trouble Shooter

An expert system strategy is used for trouble diagnosis and for recommending some remedies for the trouble. This module works interactively with the user to perform trouble diagnosis. The user can also ask it to insert some general trouble-shooting tips in to the WPS for the welding process selected earlier so that the troubles can be remedied during welding. This will be the second set of tips for preventing any troubles during welding and are mainly used for reference purposes by the welder if any welding equipment related troubles are encountered during welding. In contrast, the defect analyzer tips should be followed before starting the welding job.

In the interactive mode, any troubles encountered during welding are entered into the system. Then the system recommends remedies after the user selects a cause from the list of probable causes. The rules used for this module are given in Appendix A - Section A.5. A sample rule for the trouble shooter's interactive expert system sub-module is given below:

<i>IF</i>	<i>Trouble is</i>	<i>Loud and spattery arc</i>
<i>AND</i>	<i>Welding Process</i>	<i>Shielded Metal Arc Welding</i>
<i>THEN</i>	<i>Causes may be</i>	<i>High Current</i>
		<i>Long Arc Length</i>
		<i>Incorrect Electrode</i>
	<i>Remedies are</i>	<i>Reduce Current</i>
		<i>Reduce Arc Length</i>
		<i>Use Correct Electrode</i>

## 4.6 WPS Generator

An expert system approach for generating the weld procedure specification is used. The expert system collects the information about the process, parameters and welding procedure from the other modules (see figure 4.2) and generates a partial welding procedure. The WPS generator generates the rest of the welding procedure specification (containing information like heat treatment, welding technique etc.) by utilizing the rules stored in its knowledge base about the welding procedure specification and its dependencies on the welding parameters and job variables.

The internal rules used for WPS generator are given in appendix A - section A.7. A sample rule is given as follows:

*IF Parent Material is Mild Steel  
AND Plate thickness 0.5"  
AND Position FLAT  
AND Process is GMAW  
AND Shielding Gas is Argon  
THEN Preheat to 100 °F  
AND Postheat NONE  
AND Welding Technique 'Use Arc Force to fill Undercutting of side Wall  
AND Cleaning NONE  
AND Joint Preparation 'Included Angle 60°, Root Face 0.25 inches'*

## **5. WELDING PROCESS SELECTOR - AN ARTIFICIAL EXPERT**

The source code for the various modules was written using PDC Prolog and Borland C++ in a MS-DOS environment. The source code has been provided in section E.1 of appendix E. The expert network was tested with a set of data and the output was analyzed to verify the accuracy of the results.

### **5.1 Process Selector Neural Network Training**

The (unpruned) neural network with the initial weights as per matrix given in table 4.3 and 4.4 was trained using 108 samples see section D.1 in appendix D) obtained from *Arc Welding Standard Data Handbook (1975)*. The system was asked to select the process for a set of 14 job samples different from training samples (from *The Procedure Handbook of Arc Welding (1973)*) with care to ensure that all the categories for the job variables were covered. The set of input job variables and the output of neural network (column L) and the expert system (column N) for the 14 samples is provided in table 5.1 for analysis purposes and compared to the actual process (column Q) recommended by welding experts. The weight modifications observed after the neural network was trained have been shown in table 5. 4 and table 5.5 for the unpruned and pruned network, respectively. This process was repeated using the N-fold validation method described in section 5.5.

#### **5.1.1 Analysis of initial results**

In table 5.1 (column N) the results of the expert system indicate that most of the time the expert system recommends more than one process with the same confidence. This is due to

the initial criteria assignment being based on the applicability of the four welding processes for those job variables. Later, we will demonstrate how the knowledge of the Expert System is modified to produce a list of applicable processes with varying degrees of confidence.

### **5.1.2 Pruning of the Neural Network based on Ineffectual Criteria**

It was observed that the time for training the neural network was too long and it was taking 3-4 days on a Windows/NT server to train a **32 x 4 x 4** neural network and the convergence was extremely slow. A heuristic analysis revealed that some of the input parameter did not have any effect on the output of the neural network as all the processes were capable of supporting those input parameters. For example, low deposition can be provided by all processes by controlling the parameters. Similarly, other job variables were controlled by some other aspect of welding (like parameter selection or welding procedure specification). For example, appearance depends on the welding speed and not on the process selected and all processes can produce concave, convex or flat beads by controlling the process parameters. These job variables were resulting in noisy training examples, which caused difficulty in training the neural network. Such input job variables, listed below, were then categorized as ineffectual and were pruned from the input parameters based on heuristic analysis (See Figure 5.1 for the pruned network):

1. **Low deposition requirements:** All processes can provide **low** deposition.
2. **Low seam lengths:** All processes can be used for **low** seam lengths.
3. **Flat position:** All processes can be used in **flat** position.
4. **Low quality:** All processes can be used for jobs requiring “LOW” quality.

Table 5.1 Process Selector Results for 14 job samples.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Job ID	Leg Length/ Plate thickness (mm)	Seam Length (meters)	Horizontal, Vertical, Downward, Overhead)	Weld Type (Fillet, Butt)	Strength	Deposition	Quality	Automation	Appearance (Flat, Convex, Concave)	Parent Material (Stainless Steel, Carbon Steel, Aluminum Alloy)	Unpruned Neural Network Recommendation	Pruned Neural Network Recommendation (Confidence Factor)	Expert System Recommendation (Confidence Factor)	Final Selection by Process Selector Module (Unpruned Neuro Expert System)	Final Selection by Process Selector Module (Pruned Neuro Expert System)
1	5	15	Flat	Butt	High	Low	High	Yes	Flat	Carbon Steel	GTAW (0.6)	GTAW (0.87)	SAW(0), GTAW(0), GMAW (0)	GTAW	GTAW
2	10	15	Hor.	Fillet	High	High	High	Yes	Flat	Carbon Steel	SAW (0.99)	SAW (1)	SAW (0), GMAW (0)	SAW	SAW
3	6	5	Over.	Butt	Low	Low	High	No	Flat	Stainless St.	SMAW (0.94)	SMAW (0.89)	SMAW(0), GTAW(0), GMAW(0)	SMAW	SMAW
4	4	8	Vert.	Butt	Low	Low	Low	No	Flat	Stainless St.	SMAW (0.94)	SMAW (0.62)	SMAW(0), GTAW(0), GMAW(0)	SMAW	SMAW
5	8	8	Flat	Fillet	High	High	Low	No	Flat	Stainless St.	SMAW (0.71)	SAW (0.99)	SMAW(0),GMAW(0)	SMAW	SAW
6	3	20	Flat	Butt	High	High	High	Yes	Flat	Aluminum Alloy	SAW (0.96)	SAW (0.98)	GTAW(0),GMAW(0)	GMAW	GMAW
7	13	20	Flat	Butt	Low	High	High	Yes	Flat	Aluminum Alloy	SAW (0.97)	SAW(1), GTAW(0.53)	GMAW(0),SAW(0)	SAW	SAW
8	13	8	Hor.	Butt	High	High	High	No	Flat	Aluminum Alloy	GMAW (0.56)	GMAW (0.99)	GMAW(0)	GMAW	GMAW
9	6	15	Vert.	Butt	Low	Low	Med.	No	Flat	Aluminum Alloy	SMAW (0.92)	GTAW(0.63)	GMAW(0),GTAW(0)	SMAW	GTAW
10	25	3	Vert.	Butt	High	High	Med.	No	Flat	Aluminum Alloy	GTAW (0.98)	GTAW (0.45)	GMAW(0)	GTAW	GMAW
11	19	12	Over.	Fillet	High	High	Med.	No	Flat	Aluminum Alloy	GMAW (0.84)	GTAW (0.12)	GMAW(1.0)	GMAW	GMAW
12	3	12	Hor.	Fillet	High	Low	High	No	Flat	Stainless St.	SMAW (0.94)	GMAW (0.76)	GMAW(0),GTAW(0)	SMAW	GMAW
13	5	12	Flat	Butt	High	Low	High	Yes	Flat	Aluminum Alloy	SAW (0.63)	GTAW (0.69)	GTAW(0),SAW(0)	SAW	GTAW
14	13	12	Hor.	Butt	High	High	High	Yes	Flat	Aluminum Alloy	SAW (0.96)	SAW (0.99)	GMAW(0),SAW(0)	GMAW	GMAW

5. **Low strength:** All processes can be used for jobs requiring “LOW” strength.
6. **Appearance (3 nodes):** Since appearance is controlled by the process parameters and all the four arc welding processes can provide all the 3 bead shapes, it was decided to remove this parameter.

Table 5.2 : Initialization Matrix for Pruned Neural Network

<b>Input Node</b>	<b>Hidden Node</b>			
	<b>SMAW</b>	<b>GMAW</b>	<b>GTAW</b>	<b>SAW</b>
Aluminum Alloys	-1	1	1	1
Carbon Steel	0	0	0	0
Stainless Steel	0	0	0	0
Butt	0	0	0	0
Fillet	0	0	0	0
Horizontal	0	0	0	0
Vertical	1	1	1	-1
Downward	1	1	1	-1
Overhead	1	1	1	-1
Seam Length High	-1	1	1	1
Plate Thickness High	-1	1	-1	1
Plate Thickness Medium	0	0	0	0
Plate Thickness Low	1	1	1	-1
Strength High	0	0	0	0
Deposition - High	-1	1	-1	1
Deposition - Medium	0	0	0	0
Weld Quality Medium	0	0	0	0
Weld Quality High	-1	1	-1	1
Automation Yes	-1	1	1	1
Automation No	1	1	1	-1
Leg Length Low	0	0	0	0
Leg Length High	0	1	0	1
SMAW	1	0	0	0
GMAW	0	1	0	0
GTAW	0	0	1	0
SAW	0	0	0	1

7. **Parent Material (2 nodes):** Since no training samples were available for cast iron and mild steel, these nodes were pruned.

Thus the number of input nodes was reduced to 22 from 32. This led to faster convergence of the neural network. The rules of the expert system were modified to assign 0 weights to these job variables. The initialization matrix was also modified as per table 5.2.

After training the unpruned neural network, it was observed that some of the link weights between the input and the hidden nodes did not get modified. Later on in section 5.2, we will demonstrate how such input nodes can be identified and pruned by observing the weight modification patterns. Upon analysis, it was also found that one of the reasons is that there were no training examples available in the handbooks referred in this research for those nodes as none of the processes could support those job variables very well. They are as follows:

1. **Parent material - cast iron and mild steel:** It is very difficult to weld cast iron and only certain types of cast irons are weldable. It is difficult to find data on welding of cast iron in handbooks. Cast iron welding requires special electrodes and procedures. Mild steel sample data was also not input because mild steel is a type of carbon steel with less than 0.25% and it was already covered under carbon steel.
2. **Welding appearance - concave/convex:** Most of the welding jobs do not require concave or convex bead shape and so no training examples were available with concave appearance requirements. The appearance criteria nodes were removed (see Section 5.1.1).



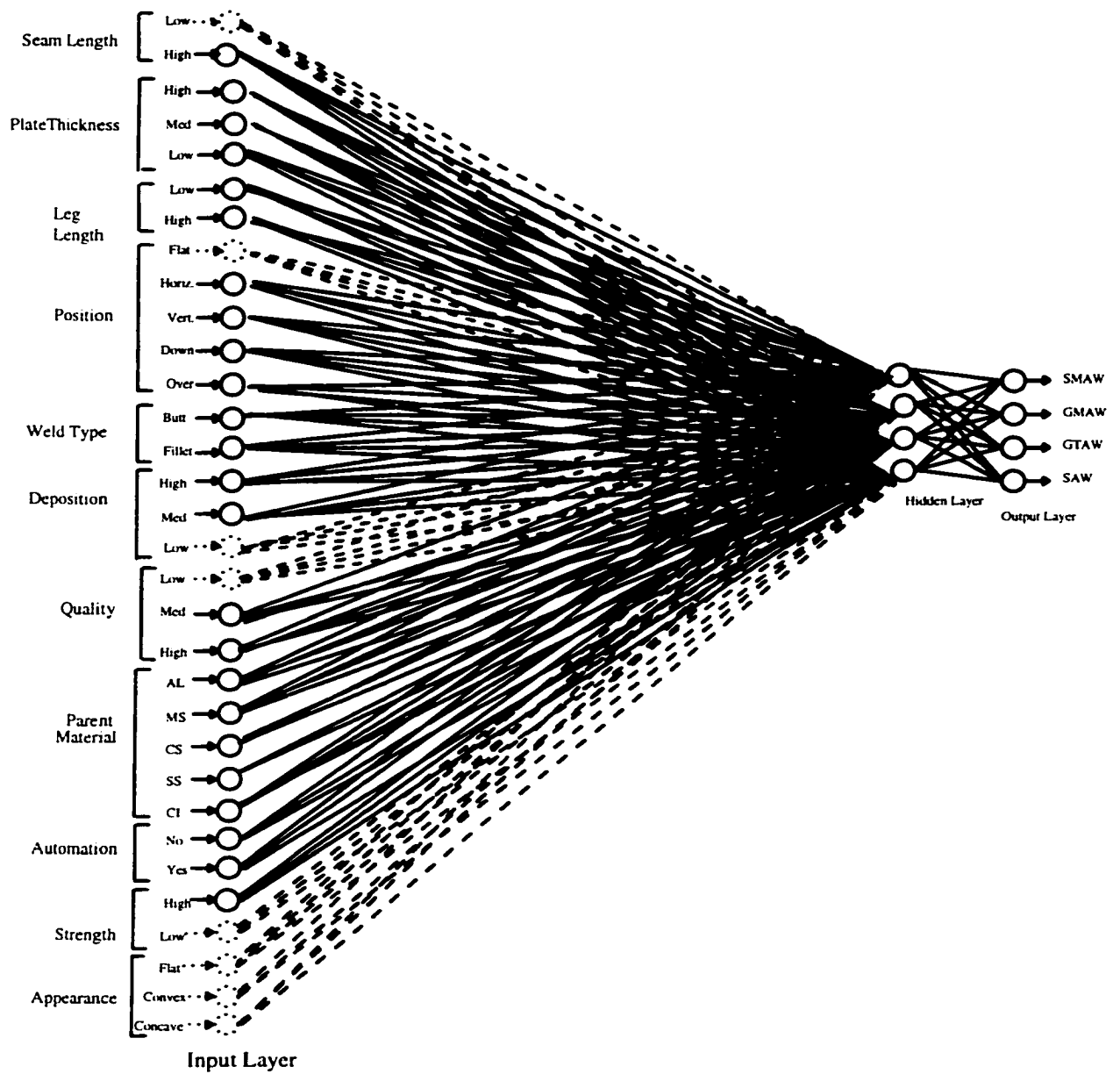


Figure 5.1 Pruning of Input Nodes in Process Selector NN based on Weight Modification

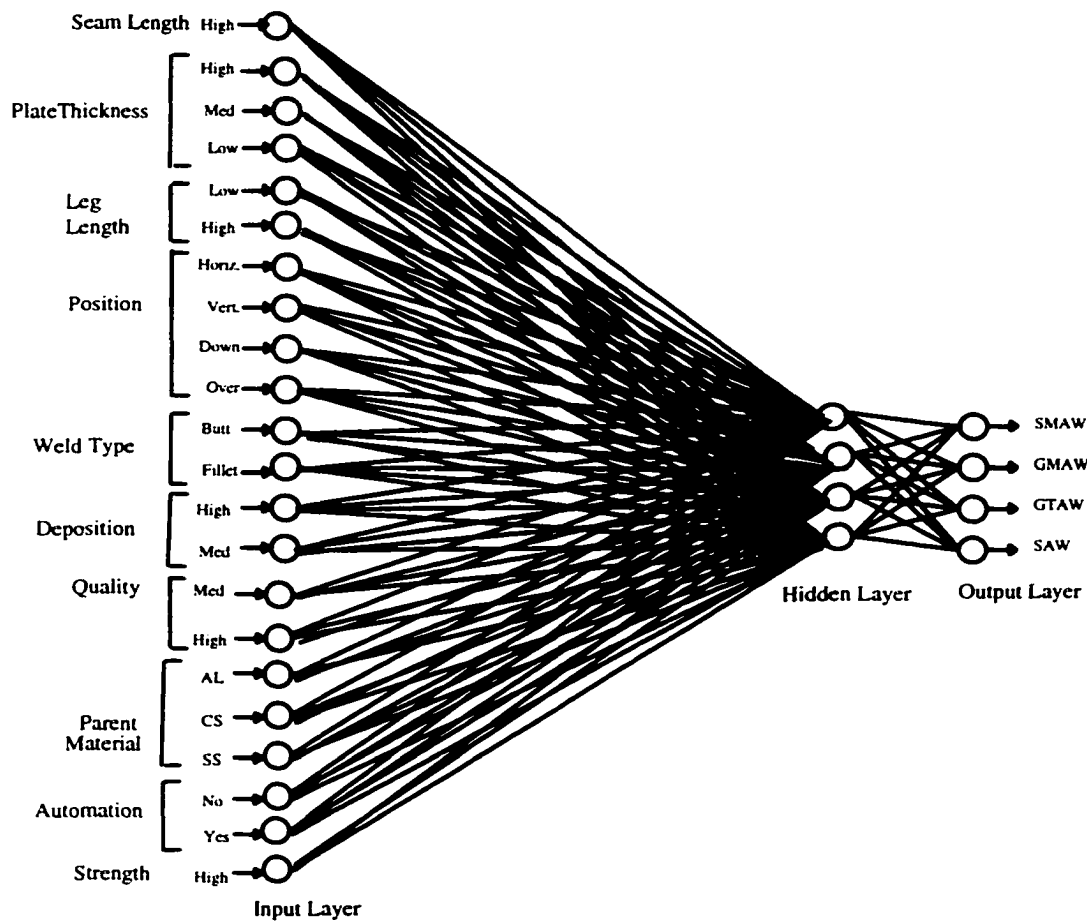


Figure 5.2 Pruned Process Selector NN Table

### 5.3 Comparison of Pruning vs. Unpruned Neural Network

	Unpruned Neural Network	Pruned Network
Total Samples	14	14
Number of Invalid Selections	3	2
Number of Matches with Overall Output	8	8
Unable to determine Solution	0	2
Number of Valid Non-Matches	3	2
Error Rate (Invalid Selection)	21.4% (3/14)	14.2% (2/14)
Success Rate (Matching Selection)	57.14% (8/14)	66.7% (8/12)
Neuro Expert Success Rate (Match)	71.42 (10/14)	85.71% (12/14)

### 5.1.3 Effect of Pruning on the Neural Network Performance

The section discusses the effect of pruning on the performance of the neural network in terms of training time and accuracy, as follows:

1. **Time for Training:** On an average, the number of cycles required after pruning compared to that of before pruning was less.
1. **Accuracy of Results:** The accuracy of results (see table 5.3) was better in most cases for the pruned neural network. The cases in which there was a different final selection are discussed below (see columns L and M in table 5.1 for details):
  - a) Job ID 1-4: Pruned and unpruned network recommend the same process.
  - b) Job ID 5: The pruned network recommends SAW and the unpruned network recommends SMAW, which is closer to final process selected but SAW is not invalid for these job variables and this represents some degree of generalization that the neural net has gone through as a result of pruning.
  - c) Job ID 6-7: Pruned and unpruned network recommend the same process.
  - d) Job ID 8: The unpruned network recommends GMAW with a confidence of 0.56 compared to the pruned network's output of GMAW with a confidence of 0.99. Thus the degree of confidence is higher for the pruned network while the unpruned network output is just on the verge of getting rejected (less than 0.5 get rejected).
  - e) Job ID 9: The unpruned network recommends SMAW with a confidence of 0.92 compared to the pruned network's recommendation of GTAW with a

confidence of 0.63. Welding aluminum with SMAW is very difficult and so the recommendation of the pruned neural network is valid compared to the unpruned network.

- f) Job ID 10: Both recommend the same process but since the confidence factor for pruned network is less than 0.5, it is rejected and the expert system selects GMAW. The unpruned NN Confidence is 0.98 in GTAW and the system is forced to select the incorrect process of GTAW leading to error.
- g) Job ID 11: The unpruned network recommends GMAW (0.84) while the pruned network is unable to determine a solution which ultimately results in the selection of GMAW by the Expert System.
- h) Job ID 12: The unpruned network recommends SMAW (0.94) while the pruned network recommends GMAW (0.76) which is the correct solution as recommended by the expert system. Using the unpruned neuro-expert will lead to incorrect solution due to the high degree of confidence in the wrong process.
- i) Job ID 13: The pruned network accurately predicts GTAW (0.69) while the unpruned network recommends SAW, which is not advisable because of low plate thickness.
- j) Job ID 14: Both recommend SAW, which is rejected by the expert system as SAW is not suitable for very thin plates in butt welds. So this is a case where

invalid selections are made by both the pruned and the unpruned neural network.

It is evident that pruned network recommended an invalid process for two of the sample jobs while the unpruned network recommended invalid processes five times. Hence performance of the pruned neural network is better compared to the unpruned network.

Table 5.3 also indicates that using a pruned neural network in a hybrid environment is beneficial and results in a more accurate decision-making environment.

## **5.2 Knowledge Extraction and Analysis**

As discussed earlier, the expert system and the neural network share a common knowledge base representation structure (see table 4.3). Each job variable has certain weight associated for certain value and when all the weights are added up for the process, we get a total weight for that process which represents the suitability of that process. A higher weight represents more suitability. Similarly in the neural network, if the weights for links are positive then those links are imparting a favorable bias for that process provided the input activation is 1. If the weights are negative then a negative bias is introduced for that process.

Tables 5.4 and 5.5 give the weight changes in the unpruned and pruned neural networks after training them with 108 training samples. The following facts can be inferred from the pattern of weight changes in the unpruned neural network given in table 5.4:

1. Most of the initial bias values have remained unchanged, That is, weights that were initialized with negative values remained negative while the positive ones remained positive. This implies that the bias in the rule-based initialization of NN weights was fairly accurate. The exceptions are the newly learned rules as follows:
  - a) GTAW and Plate Thickness - High: Positive bias implies that our initial rule that GTAW is not suitable for plates of high thickness is not true.
  - b) GTAW and Weld Quality - High: The positive weight means that GTAW does provide high quality and our initial rule that it is not suitable for high quality requirements is not true.
  - c) GMAW and Automation - No: The negative weight means that GMAW should not be used where Automation is not desired.
2. For some input nodes there were no weight changes as there was no training data available for those categories and hence they were pruned as discussed in section 5.1.
3. The new rules that are derived along with reinforcement or weakening of initial rules as a result of the NN training are as follows:
  - a) **Parent material nodes:**
    - i) Aluminum Alloy: GMAW, GTAW and SAW are more suitable for welding aluminum alloys compared to SMAW. Our rule “SMAW is not suitable for aluminum alloy” got further credence.

- ii) **Mild steel and cast iron:** Pruned due to lack of weight change.
  - iii) **Carbon steel:** SAW is more suitable for Carbon steel compared to other processes.
- b) **Weld type nodes:** It seems from the weight changes that GTAW is utilized more for butt welding while SAW is more appropriate for fillet welding. We did not find any such rule in welding handbooks or welding literature and we think that this may be a random pattern due to less training examples. As the system is used further, stronger patterns might emerge. The number of examples used for training was just 75 and the number of job variable combinations possible is much higher (approximately 64,800).
- c) **Welding position input nodes:**
- i) **Flat Position:** There were no substantial<sup>2</sup> weight changes for FLAT position for SMAW. The positive weight changes for other 3 processes can be attributed to number of training examples with flat position for them being more than other positions (14 out of 42 GMAW samples and 8 out of total 17 GTAW samples, 14 out of 20 for SAW).

---

<sup>2</sup> IF Weight Change is less than 20% for initial weight=0 (See section 6.4.2)

or IF Absolute Weight Change > Absolute Initial Weight for Initial Weight  $\leq 0$  (Fu's (1993) Method)

All the welding experts will agree that flat position is the easiest position for welding and all processes are applicable for it. Hence this node can be pruned using the rule “**All processes can be used in FLAT position**”.

- ii) **Horizontal Position:** It seems that SAW is more appropriate than other processes in this position.
  - iii) **Vertical, downward and overhead position:** No substantial<sup>2</sup> weight changes indicate that our original rules and weight assignments are accurate.
- d) **Seam Length:**
- i) **High seam length:** No substantial<sup>2</sup> weight changes indicate that the initial weight assignment was accurate. Also our rule “**Do not use SMAW for high seam lengths**” is strengthened.
  - ii) **LOW seam length:** No substantial weight changes in any of the links indicate that this node can be pruned.
- e) **Plate thickness:** For low and high thickness no substantial weight changes were noticed except the one discussed in 1-(a) above. However, for medium thickness all the processes have substantial positive bias indicating that they are all suitable for plates of medium thickness.



- f) **Strength:**
  - i) **High Strength:** Positive bias in GMAW and SAW indicates that they are more suitable for high strength job requirements compared to SMAW which have slight negative bias.
  - ii) **Low Strength:** No substantial weight changes and hence this node is a candidate for pruning.
- g) **Appearance Nodes:** These were pruned using the rule "**Bead Shape is controlled by Parameters rather than by Process**".
- h) **Deposition:**
  - i) **High:** No substantial weight changes indicate that the initial weight assignments were accurate.
  - ii) **Medium:** The positive bias on SAW indicates that it is suitable for medium deposition requirements compared to other processes. The other weights did not get modified substantially indicating that it is neutral for medium deposition requirements.
  - iii) **Low:** The negative bias on SMAW and GTAW indicates that it is less suitable for low deposition requirements compared to GMAW and SAW which sounds reasonable. As discussed earlier, all processes can provide low deposition and hence this node was pruned.

- iv) **Weld Quality:** We pruned the **low** quality node using the rule that low quality requirements can be met by all welding processes. However, the weight changes have a different story as they indicate that SMAW is used more for low quality jobs compared to GMAW, GTAW and SAW which have negative weights. No substantial weight changes for high quality requirements except for GTAW, which has been discussed in 1-(b) above. The weight changes in medium quality shows that SAW is better for medium quality job requirements than other processes.
- i) **Automation:**
  - i) **Yes:** The substantial modification for GTAW has been discussed in 1-(c). The other substantial modification for SAW is just reinforcing our rule “SAW is suitable for Automation jobs”.
  - ii) **No:** The substantial weight modification for GMAW has been discussed above in GMAW. The substantial increase in negative bias of SAW indicates that SAW is really unsuitable for jobs where automation is not possible while the substantial positive modification in GTAW demonstrates that it is quite suitable for manual jobs. Experts will agree that automation in GTAW is difficult due to problems in setting up the filler wire feed and tungsten electrodes.

j) **Leg Length:**

- i) **High:** No substantial weight modification was observed in GMAW and SAW, while the negative bias in SMAW, indicate that it is not suitable for high leg length, which is a newly learned rule. Also, the positive bias in GTAW indicate that it is suitable for high leg length.
- ii) **Low:** No substantial weight modification was observed in any process indicating that this node may be a candidate for pruning.

Similarly, the weight changes in the pruned network were analyzed and the results were very similar to the unpruned neural network. The weight changes in the pruned network (table 5.5) have been compared against the unpruned neural network weight changes (table 5.4). Some of the prominent changes have been highlighted and numbered for the discussion below:

- a) The pruned network weight is positive which is correct when compared to the negative weight for the unpruned network. GTAW can be used for horizontal position.
- b) The pruned network has negative weight for SAW implying that it is less suitable for horizontal position. This has more credibility than the unpruned network bias of positive weight because it is difficult to use SAW in horizontal position and it is applicable only for fillet welds only.

- c) The pruned network has negative bias which indicates that GMAW is not suitable for plates of medium thickness compared to unpruned which indicates otherwise.
- d) The pruned network has negative bias which indicates that GTAW is not suitable for plates of medium thickness compared to unpruned which indicates otherwise.
- e) The pruned network has negative bias which indicates that SAW is not suitable for plates of medium thickness compared to unpruned which indicates otherwise.
- f) The pruned network has negative bias indicating that GMAW is not appropriate for low leg lengths compared to insignificant weight change for unpruned network. The pruned network classification is more appropriate as GMAW is more applicable for jobs with high leg lengths.
- g) The unpruned network has negative bias indicating that GTAW is not suitable for low leg length while pruned network indicates otherwise. Welding experts will agree with the latter as GTAW is usually used for jobs of less leg length.
- h) The unpruned network has positive bias indicating that GTAW is more suitable for high leg length while pruned network indicates otherwise. Welding experts will agree with the latter as GTAW is usually used for jobs of less leg length.

Table 5.4 Weight Changes in the Unpruned Neural Network after Training

		Weight Change after training with 108 samples											
		Initial	SMAW	Unpruned	Difference	Initial	GMAW	Unpruned	Difference	Initial	GTAW	Unpruned	Difference
Parent Material	Aluminum Alloy	-1	-1.56	-0.56	1	0.91	-0.09	1	2.02	1.02	1	0.35	-0.65
	Mild Steel	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00
	Carbon Steel	0	-0.12	-0.12	0	-1.19	-1.19	0	-0.18	-0.18	0	2.95	2.95
	Stainless Steel	0	0.42	0.42	0	0.56	0.56	0	-0.01	-0.01	0	-1.86	-1.86
	Cast Iron	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00
Weld Type	Butt	0	-0.31	-0.31	0	-0.29	-0.29	0	1.31	1.31	0	-1.80	-1.80
	Fillet	0	0.05	0.05	0	-0.42	-0.42	0	-0.48	-0.48	0	1.55	1.55
Position	Flat	0	0.03	0.03	0	0.57	0.57	0	1.09	1.09	0	0.80	0.80
	Horizontal	0	-0.26	-0.26	0	-0.12	-0.12	0	-0.32	-0.32	0	0.43	0.43
	Vertical	1	0.88	-0.12	1	1.04	0.04	1	1.04	0.04	-1	-0.88	0.12
	Downward	1	1.00	0.00	1	1.00	0.00	1	1.00	0.00	-1	-1.00	0.00
	Overhead	1	1.09	0.09	1	0.93	-0.07	1	1.02	0.02	-1	-1.01	-0.01
Seam Length	High	-1	-1.25	-0.25	1	0.28	-0.72	1	1.83	0.83	1	0.75	-0.25
	Low	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00
Plate Thickness	High	-1	-1.03	-0.03	1	0.98	-0.02	-1	1.34	2.34	1	0.69	-0.31
	Medium	0	1.12	1.12	0	0.61	0.61	0	1.16	1.16	0	1.03	1.03
	Low	1	0.66	-0.34	1	0.70	-0.30	1	1.34	0.34	-1	-0.98	0.02

Table 5.4 Weight Changes in the Unpruned Neural Network after Training (Continued)

		Initial	SMAW Unpruned	Difference	Initial	GMAW Unpruned	Difference	Initial	GTAW Unpruned	Difference	Initial	SAW Unpruned	Difference
Strength	High	0	-0.25	-0.25	0	0.72	0.72	0	0.83	0.83	0	0.25	0.25
	Low	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00
Appearance	Flat	0	0.13	0.13	0	0.15	0.15	0	-0.87	-0.87	0	0.90	0.90
	Convex	0	-0.22	-0.22	0	-0.37	-0.37	0	0.78	0.78	0	-0.54	-0.54
	Concave	0	-0.16	-0.16	0	-0.50	-0.50	0	0.92	0.92	0	-0.61	-0.61
Deposition	High	-1	-0.83	0.17	1	0.35	-0.65	-1	-0.57	0.43	1	0.07	-0.93
	Medium	0	-0.05	-0.05	0	0.05	0.05	0	-0.18	-0.18	0	0.34	0.34
	Low	0	-0.37	-0.37	0	0.58	0.58	0	-0.56	-0.56	0	0.48	0.48
Weld Quality	Low	0	0.94	0.94	0	-0.27	-0.27	0	-0.28	-0.28	0	-0.07	-0.07
	Medium	0	-0.13	-0.13	0	-0.50	-0.50	0	0.01	0.01	0	1.20	1.20
	High	-1	-1.06	-0.06	1	0.05	-0.95	-1	1.10	2.10	1	1.38	0.38
Automation	Yes	-1	-1.01	-0.01	1	1.29	0.29	-1	0.45	-0.55	1	1.73	0.73
	No	1	0.76	-0.24	1	-0.002	-1.002	1	2.38	1.38	-1	-1.98	-0.98
Leg Length	Low	0	-0.01	-0.01	0	0.09	0.09	0	-0.07	-0.07	0	-0.08	-0.08
	High	0	-0.24	-0.24	1	0.20	-0.80	0	0.90	0.90	1	0.83	-0.17
Hidden Layer	SMAW	1	2.36	1.36	0	-1.61	-1.61	0	-1.65	-1.65	0	-1.40	-1.40
	GMAW	0	-0.29	-0.29	1	2.00	1.00	0	-1.88	-1.88	0	-1.58	-1.58
	GTAW	0	-0.63	-0.63	0	-0.72	-0.72	1	0.90	-0.10	0	-0.56	-0.56
	SAW	0	-0.37	-0.37	0	0.31	0.31	0	-1.98	-1.98	1	1.62	0.62

Table 5.5 Weight Changes in Pruned Neural Network

Job Variable	Value	Weight Changes after training with 108 samples														
		Initial	SMAW Unpruned	SMAW Pruned	Initial	GMAW Unpruned	GMAW Pruned	Initial	GTAW Unpruned	GTAW Pruned	Initial	SAW Unpruned	SAW Pruned			
Parent Material	Aluminum Alloys	-1	-1.56	-2.27	1	0.91	3.48	-1	2.02	0.90	1	0.35	-0.08			
	Carbon Steel	0	-0.12	0.13	0	-1.19	-2.34	0	-0.18	1.41	0	2.95	1.92			
	Stainless Steel	0	0.42	0.88	0	0.56	-0.51	0	-0.01	0.63	0	-1.86	-1.62			
Weld Type	Butt	0	-0.31	-0.62	0	-0.29	2.14	0	1.31	-2.34	0	-1.80	-1.01			
	Fillet	0	0.05	0.35	0	-0.42	-2.51	0	-0.48	2.48	0	1.55	0.39			
Position	Horizontal	0	-0.26	-0.38	0	-0.12	0.14	a	0	-0.32	0.62	b	0	0.43	-0.84	
	Vertical	1	0.88	0.50	1	1.04	0.97	1	1.04	0.40	-1	-0.88	-0.13			
	Downward	1	1.00	1.00	1	1.00	1.00	1	1.00	1.00	-1	-1.00	-1.00			
	Overhead	1	1.09	1.01	1	0.93	1.37	1	1.02	0.66	-1	-1.01	-1.25			
Seam Length	High	-1	-1.25	-1.26	1	0.28	0.63	1	1.83	1.14	1	0.75	0.37			
Plate Thickness	High	-1	-1.03	-1.71	1	0.98	1.36	-1	1.34	1.02	1	0.69	0.70			
	Medium	0	1.12	0.94	c	0	0.61	-0.38	d	0	1.16	-0.13	e	0	1.03	-0.53
	Low	1	0.66	0.51	1	0.70	0.66	1	1.34	1.25	-1	-0.98	-0.80			
Strength	High	0	-0.25	-0.26	0	0.72	0.37	0	0.83	0.14	0	0.25	0.63			
Deposition	High	-1	-0.83	-0.50	1	0.35	-0.51	-1	-0.57	-0.05	-1	0.07	0.30			
	Medium	0	-0.05	0.10	0	0.05	-0.11	0	-0.18	-0.02	0	0.34	0.15			
Quality	Medium	0	-0.13	-0.37	0	-0.50	-0.19	0	0.01	-0.51	0	1.20	1.14			
	High	-1	-1.06	-1.77	1	0.05	1.34	-1	1.10	1.26	1	1.38	2.15			
Automation	Yes	-1	-1.01	-1.55	1	1.29	1.71	-1	0.45	1.03	1	1.73	0.11			
	No	1	0.76	1.29	1	0.00	-0.08	1	2.38	1.11	-1	-1.98	-0.73			
Leg Length	Low	0	-0.01	0.02	f	0	0.09	-0.81	g	0	-0.07	1.43	0	-0.08	-0.42	
	High	0	-0.24	-0.28	1	0.20	1.44	h	0	0.90	-1.29	1	0.83	0.80		
Hidden Layer	SMAW	1	2.36	2.77	0	-1.61	-2.88	0	-1.65	-1.51	0	-1.40	-0.67			
	GMAW	0	-0.29	-0.75	1	2.00	1.56	0	-1.88	-1.23	0	-1.58	-0.58			
	GTAW	0	-0.63	-0.53	0	-0.72	-1.00	1	0.90	1.54	0	-0.56	-0.50			
	SAW	0	-0.37	-0.62	0	0.31	-2.92	0	-1.98	-1.55	1	1.62	2.71			

It is evident from the above discussion that the performance of the pruned neural network is better than unpruned neural network in classification of the process selection solution space.

### **5.3 Knowledge Correction**

The weights of the various links were extracted from the trained neural network using a prolog program and were used to correct the criteria factors in the process selection knowledge base of the expert system. Since from the previous section it is obvious that some of the job variable have no bearing on the process selection, the criteria for those job variables (pruned) was set to zero as discussed in section 5.2. Fu's (1993) algorithm was utilized for modifying the weights and the prolog program (adjusten.pro) for carrying out the knowledge modification has been given in section E.4 of appendix E. The pseudo code has also been provided in section B.1 of appendix B. This program runs in following steps:

1. If the actual process selected is same as the one recommended by process selector artificial-expert then no knowledge is modified.
2. If the actual process selected is in expert systems selection set but is not recommended by neural network then the neural network is retrained.
3. If the actual process selected is recommended by the neural network also but is missing from the expert system selection set, then the NN weights are extracted and the expert system rules are adjusted using modified Fu's algorithm (see section 4.1.2). Using Fu's algorithm in our system, introduces too many new rules because any weights that are zero initially, are set to the new weight of the link obtained after training. We propose



that any weight change up to 20% should be ignored as the performance of the process selector neuro-expert system is degraded as associations get created based on noisy data or imperfect training sets.

Using the modified method, the effect of the noise and imperfections of the training set effect on the neural network is eliminated. The performance of Fu's algorithm is compared against the modified algorithm in Table 5.6.

4. If the actual process selected is not in neural network or expert system selection set then the neural network is trained for this new example, weights are extracted and then the expert system rules are adjusted using the modified Fu's algorithm.

Table 5.7 gives the effect of applying the above weight change strategies (Fu's method and modified Fu's method) and the effect on the results of the expert system.

## **5.4 Neuro-Expert System Results**

As mentioned earlier the initial criteria assignment in the expert system result in the selection of more than one process with the same confidence level. This problem is rectified after knowledge modification using modified Fu's algorithm (20% method). Fu's original method and the variations of Fu's method (10%, 20% and 30% cut-off points for weights that were initially zero) are compared in section 5.4.1.

### **5.4.1 Effect of the cut-off value in the modified Fu's Algorithm**

As discussed earlier, Fu's method was modified to take care of the weight changes weights that were initialized as zero. From tables 5.7 and 5.8, the following observations can be made when comparing our modified approach with 10% cut-off to Fu's method:

1. There was not much change in the output of the two modified expert systems. The final selections are essentially the same for all 14 jobs.
2. The confidence level for the top priority process selected was always higher for our method compared to the original Fu's algorithm.
3. The number of rules modified was less compared to Fu's method and the two modified expert systems performed comparably same. So it is obvious that some of the rules were getting modified due to insignificant rule changes.

Therefore, it was decided to investigate the effect of using 20% and 30% as the cut-off point for modifying the weights that were zero initially and the results are provided in tables 5.7 and 5.8. These results can be analyzed as follows:

- a) At 20% cut-off point the modified expert system performs very well and the bias towards GMAW is removed to some extent. It does recommend processes other than GMAW with a high confidence for jobs 3,4,5, and 12.
- b) At 30% cut-off, the modified expert system loses its capability to provide unique recommendations (see samples 4 and 9). It loses its ability to provide unbiased results as GMAW is recommended for all samples except one.

Based on the above results, it was decided to use 20% cut-off point for deciding whether to modify the weights which are zero. This results in an modified expert system that provides the closest results to the actual artificial-expert output as the accuracy improved to 64% from 56% using fu's method or the 10% weight modification approach. More details are provided in section 5.4.3 (see table 5.9).

Table 5.6 New Criteria Weights using Fu's Method and Modified Fu's Method

	Initial	SMAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	GMAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	GTAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	SAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)
Mild Steel	0	0.00	0	0.00	0.00	0	0.00	0	0.00	0.00	0	0.00	0	0.00	0.00	0	0.00	0	0.00	0.00
Stainless Steel	0	0.42	0.42	0.42	0.42	0	0.56	0.56	0.56	0.56	0	-0.01	-0.01	-0.01	0.00	0	-1.86	-1.86	-1.86	-1.86
Butt	0	-0.31	-0.31	-0.31	-0.31	0	-0.29	-0.29	-0.29	-0.29	0	1.31	1.31	1.31	1.31	0	-1.80	-1.80	-1.80	-1.80
Flat	0	0.03	0.03	0.03	0.03	0	0.57	0.57	0.57	0.57	0	1.09	1.09	1.09	1.09	0	0.80	0.80	0.80	0.80
Vertical	1	0.88	-0.12	1.00	1.00	1	1.04	0.04	1.00	1.00	1	1.04	0.04	1.00	1.00	-1	-0.88	0.12	-1.00	-1.00
Overhead	1	1.09	0.09	1.00	1.00	1	0.93	-0.07	1.00	1.00	1	1.02	0.02	1.00	1.00	-1	-1.01	-0.01	-1.00	-1.00
Seam Length Low	0	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00
Plate Thickness Med.	0	1.12	1.12	1.12	1.12	0	0.61	0.61	0.61	0.61	0	1.16	1.16	1.16	1.16	0	1.03	1.03	1.03	1.03

(table continued)

	Initial	SMAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	GMAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	GTAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)	Initial	SAW Weight after Training NN	Difference	New Weight (Fu's Algorithm)	New Weight (Modified Fu's Algorithm - 20% Method)
Strength High	0	-0.25	0.13	0.13	-0.25	0	0	0.72	0.72	0.72	0	0	0.83	0.83	0.83	0	0	0.25	0.25	0.25
Appearance Flat	0	0.13	0.13	0.13	0.00	0	0	0.15	0.15	0.15	0	0	-0.87	-0.87	-0.87	0	0	0.90	0.90	0.90
Appearance Concave	0	-0.16	-0.16	-0.16	0.00	0	0	-0.50	-0.50	-0.50	0	0	0.92	0.92	0.92	0	0	-0.61	-0.61	-0.61
Deposition - Medium	0	-0.05	-0.05	-0.05	0.00	0	0	0.05	0.05	0.05	0	0	-0.18	-0.18	-0.18	0	0	0.34	0.34	0.34
Weld Quality - Low	0	0.94	0.94	0.94	0.94	0	0	-0.27	-0.27	-0.27	0	0	-0.28	-0.28	-0.28	0	0	-0.07	-0.07	0.00
Weld Quality High	-1	-1.06	-0.06	-1.00	-1.00	1	1	0.05	-0.95	1.00	1.00	-1	1.10	2.10	-1.00	-1.00	1	1.38	0.38	1.00
Automation No	1	0.76	-0.24	1.00	1.00	1	1	-0.002	-1.00	1.00	1.00	1	2.38	1.38	2.38	2.38	-1	-1.98	-0.98	-1.00
Leg Length High	0	-0.24	-0.24	-0.24	-0.24	0	0	0.20	-0.80	1.00	1.00	0	0.90	0.90	0.90	0.90	1	0.83	-0.17	1.00
GMAW	0	-0.29	-0.29	-0.70	-0.70	1	1	2.00	1.00	-0.70	1.00	0	-1.88	-1.88	-1.88	-1.88	0	-1.58	-1.58	-1.58
SAW	0	-0.37	-0.37	-0.36	-0.36	0	0	0.31	0.31	-0.36	0.31	0	-1.98	-1.98	-0.36	-1.98	1	1.62	0.62	-0.36

#### **5.4.2 Advantages of using Validation Step in Process Selector**

There is a validation check that is performed in the process selector neuro-expert system for each applicable process before the list is finalized, i.e. the selection of the modified expert system (using 20% approach) is subject to validation check within the neuro-expert system. So, even if the confidence factor of neural network is higher than that of expert system, the expert system throws the invalid process recommended by neural network and selects the expert system selection. The following job ID demonstrate the usefulness of the validation step in increasing the accuracy of the neuro-expert system:

1. Job ID 6: It would seem that the column H should be SAW instead of GMAW but the validation step fails for SAW as the plate thickness is 3 mm which is too little for SAW.
2. Job ID 14: It would seem that SAW should have been recommended. Since SAW cannot be used for butt welds in horizontal position, the validation step rejected neural network recommendation and accepted the expert system recommendation.

#### **5.4.3 Advantages of using the Neuro-Expert Approach**

1. **Neuro-expert systems are not biased:** The results in table 5.7 (column c) show that the modified expert system (using original Fu's approach) recommends GMAW with the highest confidence for all the 14 job samples. On analysis, this prediction is justifiable as GMAW is the most flexible method and can be used in any situation and does not suffer from some of the disadvantages of other processes which are described as follows:

Table 5.7 Effects of Expert System Knowledge Updates using Neural Network Weights

A	B	C	D	E	F	G	H	I	J	I
Job ID	Original Expert System (Confidence Factor)	Modified Expert System Results (Using Fu's approach)	Modified Expert System Results (Using modified 10% approach)	Modified Expert System Results (Using modified 20% approach)	Modified Expert System Results (Using modified 30% approach)	Pruned Neural Network Selection	Final Selection by Process Selector Module (Pruned Neuro Expert System)	Actual Process Selected in handbook (SMAW, GMAW, GTAW, SAW)	Accuracy of Expert System, Neural Network and Neuro Expert Performance (0 - Incorrect, 1 - Correct)	Invalid Selection by Expert System, Neural Network and Neuro Expert performance (0 - Invalid, 1 - Valid)
1	SAW(0), GTAW(0), GMAW(0)	GMAW(0.37)	GMAW(0.37)	GMAW(0.21)	GMAW(0.39)	GTAW (0.87)	GTAW	GTAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
2	SAW(0), GMAW(0)	GMAW(0.33)	GMAW(0.34)	GMAW(0.05)	GMAW(0.31)	SAW (1)	SAW	SAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
3	SMAW(0), GTAW(0), GMAW(0)	GMAW(0.39)	GMAW(0.43)	SMAW(0.17)	GMAW(0.20)	SMAW (0.89)	SMAW	SMAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
4	SMAW(0), GTAW(0), GMAW(0)	GMAW(0.18)	GMAW(0.11)	SMAW(0.01)	GMAW(0), SMAW (0), GTAW(0)	SMAW (0.62)	SMAW	SMAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
5	SMAW(0), GMAW(0)	GMAW(0.44)	GMAW(0.52)	SMAW(0.16)	GMAW(0.33)	SAW (0.99)	SAW	SMAW	Expert System: 0 Neural Network: 0 Neuro-Expert: 0	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
6	GTAW(0), GMAW(0)	GMAW(0.40)	GMAW(0.40)	GMAW(0.39)	GMAW(0.36)	SAW (0.98)	GMAW (NN Selection invalid)	GMAW	Expert System: 1 Neural Network: 0 Neuro-Expert: 1	Expert System: 1 Neural Network: 0 Neuro-Expert: 1
7	GMAW(0), SAW(0)	GMAW(0.53)	GMAW(0.53)	GMAW(0.27)	GMAW(0.44)	SAW (1), GTAW (0.53)	SAW	SAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1

(table continued)

A	B	C	D	E	F	G	H	I	J	I
Job ID	Original Expert System Recommendation (Confidence Factor)	Modified Expert System Results (Using Fu's approach)	Modified Expert System Results (Using modified 10% approach)	Modified Expert System Results (Using modified 20% approach)	Modified Expert System Results (Using modified 30% approach)	Pruned Neural Network Selection	Process Selector Module (Pruned Neuro Expert Actual Process)	Selected in handbook (SMAW, GMAW, GTAW,	Accuracy of Expert System, Neural Network and Neuro Expert performance (0 - Incorrect, 1 - Correct)	Invalid Selection by Expert System, Neural Network and Neuro Expert performance (0 - Invalid, 1 - Valid)
8	GMAW(0)	GMAW(0.69)	GMAW(0.68)	GMAW(0.83)	GMAW(0.68)	GMAW (0.99)	GMAW	GMAW	Expert System: 1 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
9	GMAW(0), GTAW(0.0)	GMAW(0.08)	GMAW(0.08)	GMAW(0.20)	GMAW(0), GTAW(0)	GTAW (0.63)	GTAW	GTAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
10	GMAW(0)	GMAW(0.59)	GMAW(0.59)	GMAW(0.65)	GMAW(0.57)	GTAW (0.45)	GMAW	GMAW	Expert System: 1 Neural Network: 0 Neuro-Expert: 1	Expert System: 1 Neural Network: 0 Neuro-Expert: 1
11	GMAW(1.0)	GMAW(0.51)	GMAW(0.51)	GMAW(0.52)	GMAW(0.49)	GTAW (0.12)	GMAW	GMAW	Expert System: 1 Neural Network: 0 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
12	GMAW(0), GTAW(0)	GMAW(0.07)	GMAW(0.05)	GTAW(0.06)	GTAW(0.03)	GMAW (0.76)	GMAW	GTAW	Expert System: 0 Neural Network: 0 Neuro-Expert: 0	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
13	GTAW(0), SAW(0)	GMAW(0.33)	GMAW(0.32)	GMAW(0.24)	GMAW(0.26)	GTAW (0.69)	GTAW	GTAW	Expert System: 0 Neural Network: 1 Neuro-Expert: 1	Expert System: 1 Neural Network: 1 Neuro-Expert: 1
14	GMAW(0), SAW(0)	GMAW(0.58)	GMAW(0.44)	GMAW(0.27)	GMAW(0.34)	SAW (0.99)	GMAW (NN Selection Invalid)	GMAW	Expert System: 1 Neural Network: 0 Neuro-Expert: 1	Expert System: 1 Neural Network: 0 Neuro-Expert: 1

- a) SAW can be used in flat position (for all weld types) and horizontal position (for fillet welds) only.
- b) SAW cannot be used where automation is not desired.
- c) GTAW is not highly recommended for plates with high thickness.
- d) SMAW cannot be used where automation is desirable.

Table 5.8 Total criteria weights for different weight extraction methods for 14 samples

	Fu's Method				10% Rule				20% Rule				30% Rule			
	SMAW	GMAW	GTAW	SAW	SMAW	GMAW	GTAW	SAW	SMAW	GMAW	GTAW	SAW	SMAW	GMAW	GTAW	SAW
1	0	<b>2.66</b>	1.71	1.13	0	<b>2.73</b>	1.71	1.13	0	<b>3.44</b>	2.53	2.72	0	<b>2.59</b>	1.59	1.24
2	0	<b>4.88</b>	1.6	3.29	0	<b>4.96</b>	1.6	3.29	0	<b>3.2</b>	0.59	3.03	0	<b>4.96</b>	1.6	3.41
3	1.77	<b>2.91</b>	1.64	0	1.68	<b>2.98</b>	1.71	0	<b>3.72</b>	2.98	3.07	0	2	<b>2.51</b>	1.59	0
4	2.78	<b>3.4</b>	3.04	0	2.68	<b>3.48</b>	3.11	0	<b>3.57</b>	3.55	2.73	0	3	<b>3</b>	3	0
5	0.94	<b>1.68</b>	0.53	-1.65	0.85	<b>1.76</b>	0.59	-1.65	<b>3.2</b>	2.68	1.26	-0.78	1	<b>1.5</b>	0.59	-1.3
6	0	<b>5.42</b>	3.23	0	0	<b>5.49</b>	3.31	0	0	<b>5.27</b>	3.19	0	0	<b>5</b>	3.19	0
7	0	<b>6.93</b>	2.21	3.29	0	<b>7</b>	2.28	3.29	0	<b>4.75</b>	1.59	3.46	0	<b>6.5</b>	2.16	3.64
8	-1.19	<b>5.37</b>	1.65	0.61	-1.31	<b>5.44</b>	1.72	0.61	-0.67	<b>3.55</b>	0.6	0.46	-1	<b>4.96</b>	1.6	0.96
9	0.22	<b>4.4</b>	4.04	0	-0.31	<b>4.48</b>	4.11	0	1.14	<b>5.69</b>	4.53	0	0	<b>4</b>	4	0
10	-0.23	<b>6.42</b>	2.61	0	-0.32	<b>6.5</b>	2.69	0	0.32	<b>4.55</b>	1.59	0	0	<b>6.01</b>	2.57	0
11	-1.06	<b>7.19</b>	3.5	0	-1.14	<b>7.2</b>	3.5	0	-0.73	<b>5.45</b>	2.59	0	-1	<b>7.01</b>	3.57	0
12	0.98	<b>3.11</b>	2.9	-1.33	0.85	<b>3.19</b>	3.03	-1.33	1.52	<b>2.93</b>	<b>3.13</b>	-1.2	1	<b>2.94</b>	<b>3.03</b>	-0.98
13	0	<b>3.91</b>	2.64	1.89	0	<b>3.99</b>	2.71	1.89	0	<b>4.68</b>	3.54	3.49	0	<b>3.51</b>	2.59	2.24

This results in GMAW scoring the maximum points in all situations. Using the artificial-expert system approach rectifies this situation by comparing the neural network and expert system outputs, and selecting the process with higher confidence level as evident from table 5.8.

2. **Artificial-expert systems provide higher accuracy:** The accuracy of artificial-expert in predicting the outcome was 88% compared to modified expert system's accuracy of 56% (64% if using 20% weight modification rule) and neural network



accuracy of 68% (78% if using pruned neural network). See table 5.9 in section 5.5 for these results.

3. **Artificial-expert systems do not recommend invalid processes (see table 5.3):**

The 85.7% accuracy of neuro-expert system does not mean that its recommendation was invalid for 14.3% of the cases. It is just that their recommendation was different compared to the human expert (or the welding handbook in our case). If only the (pruned) neural network is used then 2 out of 14 processes (see section 5.4.2) are invalid (i.e. not at all applicable) resulting in 8% invalid process selections. Our system does have a checking mechanism to prevent these invalid processes from being recommended. Thus hybrid systems perform better than just the neural network as invalid processes are not recommended.

The number of jobs in which the neural network output differs from the actual process is six out of fourteen and so the accuracy of neural network alone is quite low (57%). The modified expert system (column E of table 5.7) alone did not give any invalid (i.e. not applicable) recommendations even though the results are biased towards GMAW.

Thus it is evident that the neuro-expert systems provide more generalization, higher accuracy and higher validity of result compared to just using the expert system or the neural network alone. We get the best of the two systems by using a neuro-expert approach.

## 5.5 Validation of Results

To ensure that the above result is not obtained by chance, a N-fold validation technique was followed to ensure the accuracy of results. The details of this validation results have been provided in Appendix F. The summary is given in table 5.9.

One hundred and eight training samples were divided into four sets (say A,B,C,D) of 27 each. The first three sets - A, B and C were used for training the neural network and the fourth set (d) was used for testing the neural network (both pruned and unpruned), the original expert system, the modified expert system (four cases - Fu's method, 10% rule, 20% rule and 30% rule). This process was repeated with the various combinations, namely ABD-C, BCD-A and ACD-B (the first 3 letters represent the training set and the last letter represents the testing set).

The accuracy of each of the seven methods was investigated using N-fold validation and is provided in table 5.9 below.

Table 5.9 Process selector results using N-fold validation

	Unpruned	Pruned	Fu's Method	10% Rule	20% Rule	30% Rule	Unpruned Hybrid (20% Rule)	Pruned Hybrid (20% Rule)
<b>First Set - Accuracy</b>	80.0%	88.0%	52.0%	60.0%	84.0%	48.0%	80.0%	92.0%
<b>Second Set - Accuracy</b>	56.0%	60.0%	72.0%	72.0%	76.0%	72.0%	72.0%	76.0%
<b>Third Set - Accuracy</b>	81.5%	92.0%	40.0%	40.0%	45.0%	40.0%	92.6%	92.6%
<b>Fourth Set - Accuracy</b>	52.0%	74.0%	63.0%	63.0%	69.0%	63.0%	74.1%	92.6%
<b>Overall Accuracy</b>	68%	78%	56%	58%	64%	55%	80%	88%

It is evident from the table 5.9 that the artificial-expert system with the 20% rule for rule extraction gives the highest accuracy in predicting the welding process. It also gives a low level of deviation and is close to that of the actual process selection. When comparing

the knowledge extraction strategies, the 20% rule gives the highest accuracy in predicting the correct welding process. The pruned neural network also performs better than the unpruned neural network in terms of accuracy and generalization as stated earlier in section 5.4.3.

## **5.6 Comparison of Process Selector to other commercially available Software**

We could only find one software called *ArcWorks Weld Selector* that comes closer to our work but it is used for welding electrode selection and the process has to be specified by the user. So, it was not possible to compare our approach to other welding software as we did not find anything comparable.

## **5.7 Summary of Process Selector**

Our results have demonstrated the following:

1. A hybrid approach is better than using a standalone neural network or expert system.
2. Pruning of neural networks results in greater accuracy, generalization capability and a faster convergence for the neural network. It prevents the selection of invalid processes.
3. Assignment of zero weights to links having no effect on the output nodes results in faster convergence of the network.
4. A modification to Fu's algorithm provides similar results with higher confidence and requires less modification of the expert system rules. If the cut-off point for weight modification is changed to 20% then the accuracy of the neural network increases to 65% from 35% and the generalization capability of the expert network is enhanced. Also, the ability to provide unique solutions is retained.

## 6. PARAMETER SELECTOR AND ECONOMIC EVALUATOR

After the process is selected, the electrodes are selected using a rule-based approach. Then the parameters for all applicable processes/electrodes are retrieved from a database storing valid combinations of job variables and process parameters. The cost and time for the

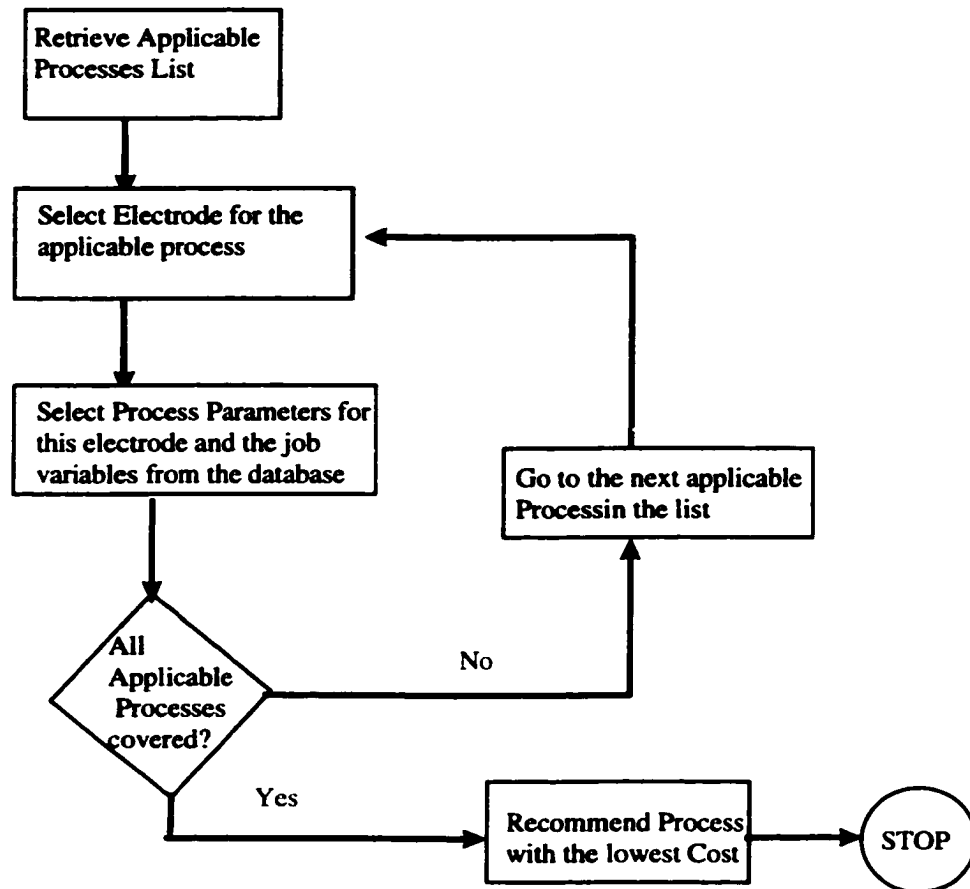


Figure 6.1 Information Flow in Parameter Selector and Economic Evaluator

applicable processes are calculated and the process with the least cost is recommended if it is acceptable to the expert system and the neural network both. As mentioned in section 4.2, the defect analyzer might later on adjust the selected parameters to minimize the defects in

welding. The prolog program for parameter selector and economic evaluator have been provided in sections E.2 and E.3 of Appendix E. Figure 6.1 denotes the flow of information in the program.

Currently these modules support SMAW and CO<sub>2</sub> shielded GMAW only, as the standard data for other processes is not available to us. Also since the **labor cost, overhead cost** and **electrode costs** will vary from one company to another, it was not possible for us to come up with an accurate cost estimation for analysis purposes. When implemented in an industrial environment, our system will be able to select a process with the least cost. We did perform some tests on the system using hypothetical cost data and were able to provide cost estimation for SMAW and GMAW for some test cases in section 5.1. Figures 6.2 and 6.3 show the results of parameter selection for job 3 and 8, respectively. This is purely hypothetical data assuming certain costs for the electrodes, labor and electricity and will vary from one company to another.

Process Parameters			
JOB VARIABLES			
=====			
Parent Material: STAINLESS STEEL	Weld Type: BUTT	Position: OVERHEAD	
Seam Length: 5 meters	Plate Thickness: 6 mm		
Strength: 60000 psi	Appearance: FLAT	Leg Length: 6	
Deposition Rate: LOW	Weld Quality: HIGH	Automation: NO	
SMAW PARAMETERS			
=====			
Run Number:1	Electrode Type: E308-17	Electrode Size: 4 mm	Current: 170 Amp
Run Number:2	Electrode Type: E308-17	Electrode Size: 4 mm	Current: 170 Amp
GMAW PARAMETERS			
=====			
Run Number : 1	S_Gas: CO2	Elec. Dia.: 0.8mm	Current: 155 Amp
Voltage: 25 U	Wire Feed: 640 a/hr	Speed:12.8 a/hr	
Run Number : 2	S_Gas: CO2	Elec. Dia.: 0.8mm	Current: 155 Amp
Voltage: 21 U	Wire Feed: 640 a/hr	Speed:12.8 a/hr	

Figure 6.2 Parameter Selector Results for Job Sample 3 of Table 5.1

Process Parameters			
<b>JOB VARIABLES</b>			
-----			
Parent Material: ALUMINUM ALLOY Weld Type: BUTT Position: HORIZONTAL			
Seam Length: 8m	Plate Thick.: 13 mm	Strength: 98000 psi	Appear.: FLAT
Leg Length: 13	Dep. Rate: HIGH	Quality: HIGH	Automation: NO
<b>SMAW PARAMETERS</b>			
-----			
Run Number:1	Electrode Type: E4043	Electrode Size: 3.15 mm	Current: 80 Amp
Run Number:2	Electrode Type: E4043	Electrode Size: 4 mm	Current: 145 Amp
Run Number:3	Electrode Type: E4043	Electrode Size: 5 mm	Current: 200 Amp
Run Number:4	Electrode Type: E4043	Electrode Size: 6.3 mm	Current: 275 Amp
<b>GMAW PARAMETERS</b>			
-----			
Run Number : 1	S_Gas: CO2	Elec. Dia.: 0.8mm	Current: 195 Amp
Voltage: 25 V	Wire Feed: 900 m/hr	Speed:7.1 m/hr	
Run Number : 2	S_Gas: CO2	Elec. Dia.: 0.8mm	Current: 195 Amp
Voltage: 25 V	Wire Feed: 900 m/hr	Speed:7.1 m/hr	
Run Number : 3	S_Gas: CO2	Elec. Dia.: 0.8mm	Current: 195 Amp
Voltage: 25 V	Wire Feed: 900 m/hr	Speed:7.1 m/hr	

Figure 6.3 Parameter Selector Results for Job Sample 8 of Table 5.1

The sample runs in figures 6.4 and 6.5 demonstrates the applicability of our system in comparing costs of using different processes. The results of economic evaluator are shown in figures 6.4 and 6.5 for jobs 3 and 8, respectively.

Economic Evaluation		
<b>Job Variables</b>		
-----		
Position: OVERHEAD Joint Type: BUTT Plate thickness: 6 mm		
Leg Length: 6 mm Seam Length: 5 meters		
<b>Economic Evaluation</b>		
-----		
	<b>SMAW</b>	<b>GMAW</b>
	----	----
Cost:	60.35	35.625
Time:	54.7	23.45
Wt. Deposited (Kg):	1.01	0.45
Electrodes Consumed:	27.5nos	239 meters
GMAW is recommended because of Lower Cost		
Do you wish to override? (y/n):_		

Figure 6.4 Economic Evaluator for Job Sample 3

The system recommends the process that has the least cost thereby optimizing the cost of welding. The user does have final control of the process as he can override the expert system selection.

Economic Evaluation		
Job Variables		
=====		
Position:	HORIZONTAL	Joint Type: BUTT
Plate thickness:	13 mm	
Leg Length:	13 mm	Seam Length: 8 meters
Economic Evaluation		
=====		
	SMAW	GMAW
	----	----
Cost:	182.88	162.4
Time:	92.8	83.68
Wt. Deposited (Kg):	1.84	1.512
Electrodes Consumed:	46.4meters	1285.6 meters
SMAW is recommended because of Lower Cost		
Do you wish to override? (y/n):		

Figure 6.5 Economic Evaluator Results for Job Sample 8

## **7. DEFECT ANALYZER – A COUPLED NEURO-EXPERT SYSTEM APPROACH**

The **Defect Analyzer** analyzes the welding process and parameters to predict any defects that might be encountered. One might argue that the process parameters have already been decided by the expert system and why should they result in defects. The fact is that it is impossible to achieve perfect weld and defects will always be there even if they are within the acceptable tolerance. For example, sometimes it may be acceptable to use cellulosic electrodes even though they result in high porosity, as the porosity may be within the permissible limits. Similarly, other parameters may result in other defects.

The objective of this module is to facilitate and improve the quality of the welding process by warning the welder of the possible defects and advising him of changes in the parameters that might minimize those defects.

Also on the basis of parameters selected, the **Defect Analyzer** is able to predict defects and insert instructions in the WPS in advance of welding so that the defects can be minimized.

The defect analyzer has four components. More detailed descriptions are given in the following sections. Each component is briefly described below:

1. A neural network for defect prediction: This approach is taken so that defect prediction knowledge can be refined when new data become available, but currently the system does not modify the expert system rules for parameter adjustment.



currently the system does not modify the expert system rules for parameter adjustment.

Technically, it may be possible to analyze the weight changes and figure out which input parameter results in a particular defect and modify the parameter adjustment expert system rules but due to the large scope of this research this task is left for the future researchers. We did identify the parameters that cause certain defects in section 7.1.3 using Fu's approach with the 20% cut-off criteria for rule extraction and thus presented the methodology for modifying the parameter adjustment. The implementation of the prolog program to take this newly identified knowledge and refine the parameter adjustment expert rules can be done in future.

2. **An Expert System for Parameter Adjustment:** As mentioned in the previous discussion, the rules for parameter adjustment are static and based on heuristic knowledge. It may be possible for future researchers to analyze the weight changes in the defect predictor neural network and modify the rules stored in the parameter adjustment expert system.
3. **WPS Tips Generation Module:** This module takes the process and the defect-list as the input and modifies the WPS using its internal rule base so that the defects can be minimized.

4. **An Interactive Expert System for Defect Analysis:** This module allows the welder to analyze the defects encountered after welding inspection, to diagnose the causes and to recommend the remedies for those defects online.

## **7.1 A Neural Network for Defect Prediction**

### **7.1.1. Network Initialization**

The initial heuristic knowledge that was used for assigning the initial weights to the neural network topology has been provided in tables 4.6 and 4.7 for SMAW and GMAW respectively. The neural network topology has been given in figures 4.11 and 4.12 for SMAW and GMAW.

### **7.1.2 Network training**

The training data used for the neural network is given in section C.3 of appendix C. The method for obtaining the training data has been described in section 4.4.3.2. The neural network was initially a 13 x 8 x 8 node design, which was found to be impossible to train with 50 examples. It was decided to reduce the size of the network by reducing the number of output nodes to 5 by removing the following nodes :

1. **Incomplete fusion:** This defect is closely related to incomplete penetration and they are usually present at the same time.
2. **Spatter:** Arc blow is often accompanied by spatter and so the same parameter adjustments will fix both the problems.
3. **Cracking:** Cracking is often accompanied by porosity and same parameter adjustments will fix both problems.

After eliminating these three output nodes, the neural network was still having problems in convergence with even a reduced set of examples (20). An incremental approach was then taken in which less number of examples (20) were used initially and then examples were added after training the neural network as follows:

1. All the samples of any electrode type (say, cellulosic) are selected and the neural network is trained for these samples. This helps in reducing the number of different patterns as the pattern for electrode type is same for all training examples.
2. Select the next electrode type (say rutile).
3. Two examples of second electrode type (i.e. rutile) are added to the training set and the neural network is trained again.
4. Step 3 is repeated till the neural network had been trained for all the examples for the second electrode type.
5. Go to step 2 and select the next electrode type. Repeat steps 2 through 5 till all the electrode types have been considered and the neural network has been trained for all the samples.

The faster convergence of the neural network with less number of examples can be attributed to the fact that essentially one input node is non-existent if all the examples are of the same electrode type hence the solution space is much smaller and training is faster. Later on, when more examples are added then the back-propagation algorithm has to retrain for these two types of electrodes and since other links have already been trained, the training is much faster. For example, we have 10 input nodes and 4 of these are for electrode type. If

you pick all the samples for Rutile Electrode only, then the input activation for the 4 nodes of electrode type will always be 0 0 1 0 and hence the number of input nodes is effectively 7 as the zero input do not have any effect on the weights. i.e. the neural network has less number of input-output patterns to learn.

### **7.1.3 Weight Modification in trained Neural Network**

The weight modification was substantial in the neural network but as this is a stand alone sub-module, there is no need of applying any rule extraction strategy. Table 7.1 provides the weight changes observed after training the neural network using the incremental approach described above. It is evident from the weight modifications that new knowledge about defects is learned by the neural network as a result of training. Modified Fu's algorithm with 20% cut-off criteria was used for analyzing the significant weight changes, purely for the purposes of demonstrating the new knowledge learned by the neural network. Since we do not have an expert system for predicting defects, these weight changes remain in neural network and extraction into expert system was not done.

As mentioned earlier, it may be possible to write a program that will convert this new knowledge into parameter adjustment rules. Currently, the system has static parameter adjustment rules.

The newly learned knowledge that could be extracted if an artificial expert approach is followed include:

1. Arc Blow is caused by medium sized electrodes and medium current also and is avoided by small size electrodes and smaller currents.

**IF Defect Predicted = "Arc Blow"**  
**AND Electrode Size = "Medium"**  
**AND Current = "Medium"**  
**THEN (Electrode Size = "Low"**  
**AND CURRENT = "Low")**

2. Incomplete penetration is avoided by using rutile electrode, lower electrode sizes and higher currents as evident by the high negative and positive activation.

**IF Defect Predicted = "Incomplete Penetration"**  
**AND Electrode Type <> "Rutile"**  
**AND Electrode Size <> "Low"**  
**AND CURRENT <> "High"**  
**THEN (Electrode Type = "Rutile"**  
**AND Electrode Size = "Low"**  
**AND CURRENT = "High")**

3. Porosity is mainly caused by cellulosic electrodes and high currents as evident by the high positive activation.

**IF Defects Predicted = "Porosity"**  
**AND Electrode Type = "Cellulosic"**  
**AND Current <> "Low"**  
**THEN (Electrode Type <> "Cellulosic"**  
**AND Current = "Low")**

4. Undercuts are caused by high currents and are less in welds made using cellulosic electrodes.

**IF Defects Predicted = "Undercuts"**

AND Electrode Type  $\diamond$  "Cellulosic"  
AND Current  $\diamond$  "Low"  
THEN (Electrode Type = "Cellulosic"  
AND Current = "Low")

5. Slag inclusions are mainly due to high currents as evident by the large positive activation for that node and are avoided by using low electrode size as evident by the large negative activation.

IF Defects Predicted = "Slag Inclusions"  
AND Electrode Size  $\diamond$  "Low"  
AND Current  $\diamond$  "Low"  
THEN (Electrode Size = "Low"  
AND Current = "Low")

As mentioned earlier, currently these newly learned rules are not stored into the parameter adjustment module and the implementation of the prolog program for this purpose is outside the scope of this research.

## **7.2 Expert System for Parameter Adjustment and WPS Tips Generation**

According to fig. 4.13, the Expert Systems for parameter adjustment and WPS tips generation follow the following steps:

1. Run the trained *neural network* program with the selected process parameters and take the output of the neural network, interpret it and store the predicted defects in a working knowledge base.

2. Create a priority list of the adjustable parameters that cause these defects and a direction of adjustment for each parameter along with an applicable range of adjustment. It marks the parameters as ***Volatile*** or ***Non-Volatile*** based on their effect on other parameters. If adjusting the parameter results in a change in the applicable range of other parameters, then it is considered ***Volatile***. For instance, in SMAW, increasing electrode size increases the applicable current range and the travel speed. Also, changing electrode type changes the applicable current range and the travel speed.
3. Adjusts the volatile parameters to the maximum in the direction that will minimize the defects. If none of the parameters are volatile, go to step 4.
4. Adjust non-volatile parameters in a direction that will minimize the defect. For e.g. If increasing electrode size then use the average settings for current, wire feed and arc travel speed setting for that electrode size. Input the modified process parameters to the NN and get a prediction of expected defects and a list of parameters that cause them. If none of the parameters are adjustable then go to step 5 otherwise go to step 2.
5. Create weld procedure instructions for WPS generator using the defect analyzer rules for defect avoidance.

This approach has been implemented using prolog program given in section E.6 of appendix E. The calls to the neural network are made by the prolog program. The neural network can function as a stand-alone program also and so this is a coupled approach as discussed earlier in section 4.4. The flowchart for this methodology is provided in figure

4.14. Two components of the expert system - Parameter Adjustment and WPS Tips Generator are described below.

**Table 7.1 Weight Modifications after training the Neural Network**

	Arc Blow	After Training	Difference	Shallow Penetration	Shallow Penetration after training	Difference	Porosity	Porosity after training	Difference	Undercut	After Training	Difference	Slag Inclusion	Inclusion after training	Difference
Low H2 electrode	0	0.15	-0.15	-1	-1.83	1.33	-1	-1.20	0.70	0	2.63	-2.63	0.5	-1.47	1.97
Basic Electrode	0	0.00	0.00	1	-1.97	2.97	0.5	-0.98	1.48	0	-1.81	1.81	1	0.92	0.08
Rutile Electrode	0	0.04	-0.04	-1	-6.89	5.89	0.5	2.80	-2.30	0	2.44	-2.44	-1	-4.08	3.08
Cellulosic Electrode	0	0.01	-0.01	-1	-2.66	1.66	1	3.04	-2.04	0	-3.01	3.01	1	3.97	-2.97
Electrode Size Low	-1	-0.57	-0.43	-1	-9.82	8.82	-1	0.53	-1.53	-1	-0.70	-0.30	-1	-6.90	5.90
Electrode Size Med.	0	1.25	-1.25	0	0.13	-0.13	0	-0.59	0.59	0	-0.18	0.18	0	1.34	-1.34
Electrode Size High	1	-0.48	1.48	1	11.61	-10.61	1	-0.46	1.46	1	0.12	0.88	1	3.42	-2.42
Current Low	-1	-2.73	1.73	1	4.20	-3.20	-1	-2.41	1.41	-1	-2.09	1.09	-1	0.11	-1.11
Current Medium	0	1.33	-1.33	0	0.52	-0.52	0	-1.80	1.80	0	-1.44	1.44	0	0.27	-0.27
Current High	1	1.60	-0.60	-1	-2.80	1.80	0.5	3.20	-2.70	1	2.78	-1.78	1	-2.53	3.53
Hidden Node 1	1	3.35	-2.35	0	-2.45	2.45	0	0.24	-0.24	0	0.26	-0.26	0	0.42	-0.42
Hidden Node 2	0	-1.74	1.74	1	-0.33	1.33	0	-1.54	1.54	0	-2.46	2.46	0	-2.33	2.33
Hidden Node 3	0	1.48	-1.48	0	-0.17	0.17	1	3.24	-2.24	0	2.35	-2.35	0	-3.29	3.29
Hidden Node 4	0	-2.22	2.22	0	-2.33	2.33	0	1.53	-1.53	1	0.64	0.36	0	-1.11	1.11
Hidden Node 5	0	0.41	-0.41	0	-0.26	0.26	0	-1.10	1.10	0	-0.48	0.48	1	2.53	-1.53

### 7.2.1 Parameter Adjustment Module

The parameter adjustment module allows the adjustment of parameters selected by the parameter selector so that the defects can be minimized. A sample run of the defect analyzer parameter adjustment module is given in Figure 7.1.

The welder enters the parameter adjustment option in the defect analyzer menu and the expert system creates an input file for the neural network from the stored process parameters. The previously trained neural network predicts the defects based on these parameters (porosity and cracking in the example above). The expert system makes a list of the adjustable parameters (electrode type, electrode size and current) and asks the user that



he does want to change the parameter values to the recommended ones. As mentioned earlier in table 4.7, porosity and cracks are reduced by using low hydrogen electrodes, low current and smaller electrodes. The expert system uses its internal knowledge base to determine the new values for the parameters selected. It modifies the volatile parameters

```
SELECT AN ACTION
INTERACTIVE
PARAMETER ADJUSTMENT
AUTOMATED UPS TIPS
QUIT
```

```
Defect Analysis
Defect Predicted: Arc Blow
Defect Predicted: Porosity
Defect Predicted: Undercuts
Parameter adjustment recommended: Use Low Hydrogen Electrode
Electrode <Old>: E4043
Electrode <New>: E6016
Do you want to adjust it? <y/n>
Parameter adjustment recommended: Decrease Electrode Size
Expert Advice: 3.25 is the lowest size. Please select NO
Electrode Size <OLD>: 3.25
Electrode Size <NEW>:
Do you want to adjust it? <y/n>
Parameter adjustment recommended: DECREASE CURRENT
Old Current Value: 145
New Current Value: 100
Do you want to adjust it? <y/n>
-
```

Figure 7.1 Parameter Adjustment Session for Sample 8 of Table 5.1

first as they affect the non-volatile parameters. For example, in the sample run in figure 7.2, electrode type is changed first as electrode size and current have dependencies on it and so they are selected later.

## 7.2.2 WPS Tips Generator Module

After all the possible adjustments have been made to the parameters, if some defects are still predicted by the neural network, then the automated WPS tips options can be used to generate some tips for the welder that can help in minimizing the defects.

In the sample session (figure 7.2) for job 8 of table 5.1, despite all the adjustments made to the parameters, undercuts, porosity and arc blow are still predicted. The Automated WPS Tips Module inserts the instructions for avoiding these problems.

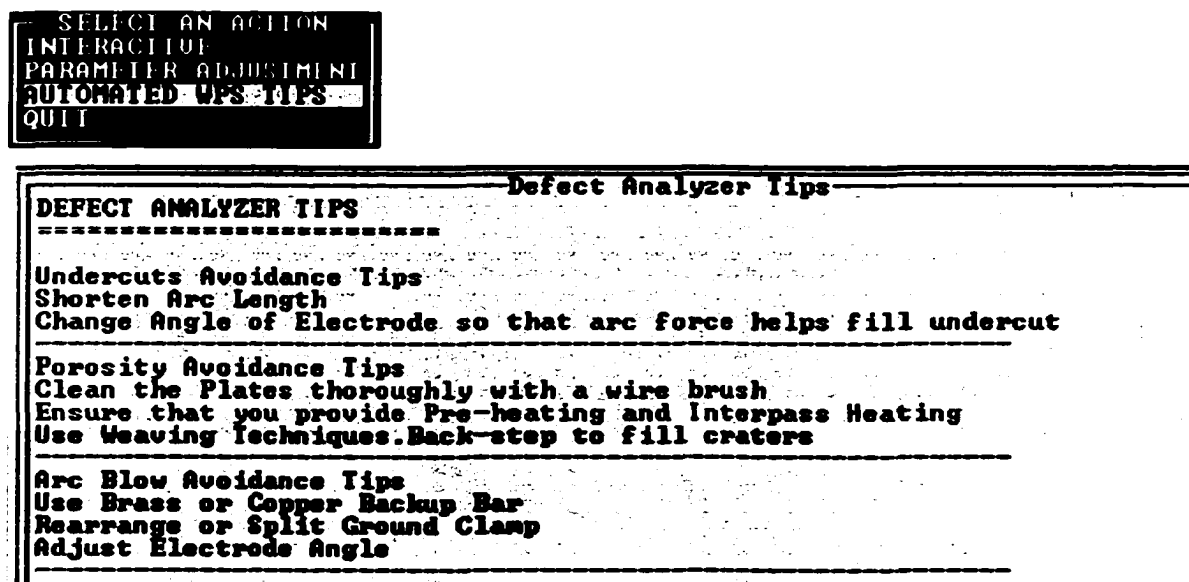


Figure 7.2 Automated Tips Generation in Defect Analyzer for Sample 8 (Table 5.1)

## 7.3 Interactive Defect Analyzer

Interactive defect analyzer allows the welder to perform defect analysis by responding to the systems questions. A sample run of the defect analyzer module is given in figure 7.3.

## 7.4 Analysis of Results

It was not possible for us to test the accuracy of the neural network and the expert system for defect analysis as defect samples were not available in handbooks. After this system is implemented in an industrial environment, it will be possible to train the neural network with actual data and get some accuracy measurements. The purpose of this research for defect analysis was to provide a framework for the design of a defect analysis system and implement it. The actual use of the system is possible in an industrial environment only when the defect data become available.

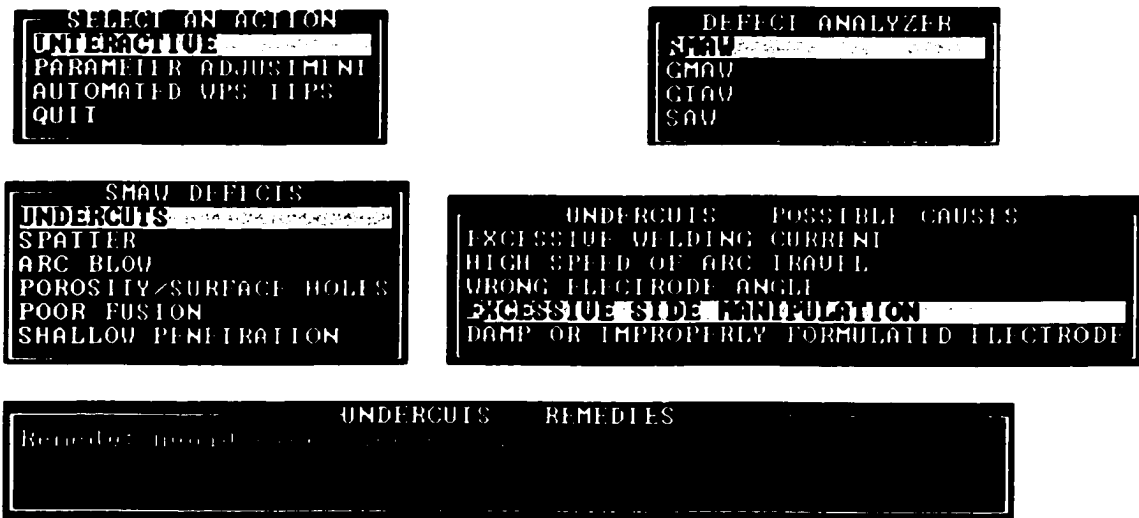


Figure 7.3 Interactive Defect Analyzer Sample Run

## 8. TROUBLE SHOOTER - A DIAGNOSTIC EXPERT SYSTEM

Since trouble shooter is mainly a diagnostic tool, it is only able to make general recommendation to the WPS generator for avoiding trouble based on the process selected. When used in interactive mode, it can diagnose the actual trouble based on user input of the symptoms of the problem. These tips prove to be useful as they help the welder as a reference in case of difficulties. For example, if the welder is electrode stubbing to the work-piece in GMAW, the WPS section on trouble-shooting contains a useful tip to increase the voltage and/or decrease the wire feed.

```
SELECT AN ACTION
INTERACTIVE
AUTOMATED WPS TIPS
QUIT
```

```
TRUBLE SHOOTER
WELDING-SET RELATED
ARC RELATED PROBLEMS
WIRE FEEDER RELATED
FLUX FEEDER RELATED
MISCELLANEOUS
```

```
WELDING SET
DOES NOT START
STARTS BUT BLOWS FUEL
WELDS BUT STOPS SOON
WOULD NOT SHUT OFF
POLARITY SWITCH DOES NOT TURN
ARCING AT GROUND CLAMP
ELECTRODE HOLDER BECOMES HOT
QUIT
```

```
WELD SET CHANGES
Line Switch not turned ON
Supply line fuse blown
Power circuit dead
Overload relay tripped
Loose or broken power, electrode or ground lead
Wrong Voltage
Polarity switch not centered
Open circuit to starter switch
```

```
TRUBLE SHOOTING
Remedy: Turn line switch on
```

Figure 8.1 Interactive Trouble Shooting Sample Screens

```

--SELECT AN ACTION
INTERACTIVE
AUTOMATED WPS TIPS
QUIT

```

```

=====
TROUBLE SHOOTING TIPS
=====
SMAW TROUBLE SHOOTING TIPS
=====
WELD REMAINS IN BALLS: Check Electrode Type
CRACKING: Bake electrodes
SPATTER: Reduce Voltage & Check Electrode Type
SLUGGISH ARC: Increase Current
LOUD & SPATTERY ARC: Reduce Current
UNSTABLE ARC: Shorten Arc

```

Figure 8.2 Automated WPS Trouble Shooting Tips Sample Screen

Figure 8.1 shows an interactive trouble shooting sample session using the expert system. The system searches the knowledge base for probable causes and remedies for the trouble encountered by the welder.

Figure 8.2 shows a sample of WPS tips generated by the trouble shooter module based on the process selected by the economic evaluator module. The welder can use these tips if he encounters any of the typical trouble listed and avoid the trouble.

## 9. WELDING PROCEDURE SPECIFICATION GENERATOR - TYING IT ALL TOGETHER

The WPS generator takes input from three modules - economic evaluator, defect analyzer and trouble shooter, applies internal rules (see appendix A - section A.7) for welding procedures and finally generates a WPS. The prolog program has been listed in section E.5 of appendix E.

Part Name:	1000	Part No.:	1000	Part ID:	1000
Electrode:	1000	Electrode No.:	1000	Electrode ID:	1000
Preheat Temp.:	250	Preheat Temp. No.:	250	Preheat Temp. ID:	250
Interpass Temp.:	250	Interpass Temp. No.:	250	Interpass Temp. ID:	250
Welding Process:	1000	Welding Process No.:	1000	Welding Process ID:	1000
Welding Speed:	1000	Welding Speed No.:	1000	Welding Speed ID:	1000
Joint:	1000	Joint No.:	1000	Joint ID:	1000
Root:	1000	Root No.:	1000	Root ID:	1000
Root Gap:	1000	Root Gap No.:	1000	Root Gap ID:	1000
Root Gap:	1000	Root Gap No.:	1000	Root Gap ID:	1000
Cleaning:	1000	Cleaning No.:	1000	Cleaning ID:	1000
Run Number:	1	Run Number No.:	1	Run Number ID:	1
Run Number:	2	Run Number No.:	2	Run Number ID:	2
Run Number:	3	Run Number No.:	3	Run Number ID:	3
Run Number:	4	Run Number No.:	4	Run Number ID:	4

Figure 9.1 Welding Procedure Specification Run for SMAW of Sample 8

Figure 9.1 and 9.2 show the sample run of the WPS Generator for SMAW and GMAW respectively, without performing defect analysis and trouble shooting for sample 8 of table 5.1. The sample 8 was used as an example to show the overall working of the neuro-expert system.

GMAW Welding Procedure Specification			
Type of Joint:	BUTT	Plate thickness:	13
Electrode Specification:	ALUMINUM ALLOY	Shielding Gas:	CO2
Gas Cup Opening:	5/8 inches		
Preheating:	20 degree C	Interpass Heating:	20 degree C
Welding Technique:	Side-to-side weaving		
Welded from:	Both Sides		
Joint Design:	Double-Sided X Shape		
Root Face:	2		
Root Angle:	60 degree		
Root Gap:	2		
Cleaning:	No Cleaning Required between passes		
Run Number:	1	Electrode Size:	0.8
Voltage (U):	25	Wire Feed (m/hour):	900
		Current (Amp):	195
		Speed (m/hour):	7.1
Run Number:	2	Electrode Size:	0.8
Voltage (U):	25	Wire Feed (m/hour):	900
		Current (Amp):	195
		Speed (m/hour):	7.1
Run Number:	3	Electrode Size:	0.8
Voltage (U):	25	Wire Feed (m/hour):	900
		Current (Amp):	195
		Speed (m/hour):	7.1

Figure 9.2 Welding Procedure Specification Sample Run for GMAW of Sample 8

If the Defect Analyzer module is utilized for predicting the defect and then making changes to the Welding Parameters then the WPS will get modified and has been shown in Figure 9.3 for SMAW.

Welding Procedure Specification			
Process:	Shielded Metal Arc Welding		
Type of Joint:	BUTT	Plate thickness:	13
Position:	HORIZONTAL		
Electrode Specification:	E4043		
Preheating:	20 degree C		
Interpass Heating:	20 degree C		
Welding Technique:	Side-to-side weaving		
Welded from:	Both Sides		
Joint Design:	Double-Sided X Shape		
Root Face:	2		
Root Angle:	60 degree		
Root Gap:	2		
Cleaning:	No Cleaning Required between passes		
Run Number:	1	Electrode Size:	3.25
		Current (Amp):	100
Run Number:	2	Electrode Size:	4
		Current (Amp):	145
Run Number:	3	Electrode Size:	5
		Current (Amp):	200
Run Number:	4	Electrode Size:	6.3
		Current (Amp):	275

Figure 9.3 Modified Welding Procedure Specification for SMAW of Sample 8

**Table 9.1 Overall Welding Procedure Specification for SMAW of Sample 8**

<p>           Process: Shielded Metal Arc Welding            Type of Joint: BUTT            Plate thickness:            Position: HORIZONTAL            Welded from: Both Sides            Preheating and Temperature Control: 20 degree C            Interpass Heating: 20 degree C            Welding Technique: Side-to-side weaving            Joint Design: Double-Sided X Shape            Root Face: 2            Root Angle: 60 degree            Root Gap: 2            Cleaning: Clean with wire brush and chipping hammer in between passes         </p> <hr/> <p>           Run Number: 1 Electrode Size: 3.25 Current (Amp): 100         </p> <hr/> <p>           Run Number: 2 Electrode Size: 4 Current (Amp): 145         </p> <hr/> <p>           Run Number: 3 Electrode Size: 5 Current (Amp): 200         </p> <hr/> <p>           Run Number: 4 Electrode Size: 6.3 Current (Amp): 275         </p> <hr/>
<p> <b>DEFECT ANALYZER TIPS</b>            =====         </p> <p>           Undercuts Avoidance Tips            Shorten Arc Length            Change Angle of Electrode so that arc force helps fill undercut         </p> <hr/> <p>           Porosity Avoidance Tips            Clean the Plates thoroughly with a wire brush            Ensure that you provide Pre-heating and Interpass Heating            Use Weaving Techniques. Back-step to fill craters         </p> <hr/> <p>           Arc Blow Avoidance Tips            Use Brass or Copper Backup Bar            Rearrange or Split Ground Clamp            Adjust Electrode Angle         </p> <hr/>
<p> <b>TROUBLE SHOOTING TIPS</b>            =====         </p> <p> <b>WELD REMAINS IN BALLS:</b> Check Electrode Type  <b>CRACKING:</b> Bake electrodes  <b>SPATTER:</b> Reduce Voltage &amp; Check Electrode Type  <b>SLUGGISH ARC:</b> Increase Current  <b>LOUD &amp; SPATTERY ARC:</b> Reduce Current  <b>UNSTABLE ARC:</b> Shorten Arc         </p>

Table 9.1 lists the complete WPS for sample 8 of table 5.1 after defect analysis, WPS modification and trouble shooting has been performed. The advantages of using the integrated expert system approach for WPS generation are:



1. **Defects are minimized:** Since the defect analyzer creates (if needed) some of the procedures to minimize the defects, the WPS generated produces less defects in welding.
2. **Trouble-shooting tips are there:** The trouble-shooter inserts some tips for the welder in case he encounters some troubles in welding. For example, in low hydrogen electrode welding, electrodes should be baked before use, otherwise cracking may be observed. Similarly, if stubbing is observed in GMAW, the voltage has to be increased and the wire feed has to be reduced. The trouble-shooter is able to identify these needs and inserts the appropriate comments in the WPS for the welder.
3. **WPS Generator does not miss any details:** The WPS generator uses its internal rules to ensure that all the WPS parameters are covered. Details like preheating, interpass heating, joint preparation etc. are very important to ensure proper welding.

Most of the software available in the market does not allow the generation of WPS but only stores it. *Arc Works Weldselector* by Lincoln Electric allows you to specify the welding parameters, joint design and welding procedure and stores it in a database for future reference. Our approach allows for the selection of process, parameters and WPS based on job variables, and generates a WPS. Our approach also allows the defect analyzer and the trouble-shooter to have a say in the final WPS so that defects and trouble can be avoided proactively.

## **10. CONCLUSIONS**

This research developed, implemented and tested a neuro-expert system to facilitate decision making in welding planning. Several tests were performed to validate the system developed as part of this research and it is determined that automated decision-making can be easily performed with the neuro-expert system developed. Since this work was done for Ph.D. in engineering science, the focus was more on the welding application aspects of neural networks and expert systems rather than development of new computational and mathematical algorithms or models for general applications. Nevertheless, we did describe some methods for improving the performance of the neural network and improving the accuracy of the neuro-expert system.

### **10.1 Contributions of this Research**

1. Successful demonstration of the feasibility of using an Neuro-Expert System Approach for decision-making in a synergistic and holistic fashion.
2. Analysis of pruned neural network approach in the process selector to demonstrate the advantages of using it, including the development of a new pruning approach by looking at data patterns and rule based analysis.
3. The artificial expert approach to process selection is demonstrated to be superior to using the expert system or the neural network approach alone.
4. Demonstration of the usefulness of knowledge extraction from neural networks using a modified Fu's algorithm.

5. The effect of the cut-off point for weight modification in the modified Fu's algorithm was also studied and reported.
6. Demonstration of the coupled approach for the defect analyzer.

### **10.2 Advantages of using the Neuro Expert System**

1. This study provides a scientific procedure to overcome the problems encountered in decision making in a welding environment. The problems may be the non-optimal selection of process and parameters due to tiredness, monotonicity or carelessness of the process planner.
2. The cost and time estimation becomes less tedious as search for relevant standard data from tables is eliminated. The storage of knowledge is dynamic and the updating is transparent to the User.
3. The time required for preparing WPS is shortened and WPS generation for a large variety of jobs is possible in a very short time. The customer can be told about the estimated time and costs for the completion of the job.

### **10.3 Salient Features of the Neuro-Expert System**

1. The system is capable of learning new rules, modifying existing rules and discovering new relationships between input job variables and output parameters.
2. The Neuro-Expert system when implemented in an industrial environment will take over the decision making task for welding and will assist the welding engineer and other users in selecting the process, parameters, WPS for particular job variables.

3. The *defect analyzer* and the *trouble shooter* are able to diagnose the causes for the defects or troubles, and suggest corrective action.
4. An integrated approach for decision making is provided which takes all the aspects of welding into consideration when making various decisions. The various modules have strong interaction and provide well-balanced decisions.

#### **10.4 Limitations of the Neuro-Expert System**

1. All the standard data is not available for GMAW, GTAW and SAW and so most of the modules support SMAW and GMAW (with CO<sub>2</sub> Shielding Gas) only.
2. The number of rules stored in the system is limited. Since the purpose of the research was to develop a methodology for decision-making and implement it on a limited scope, all the rules and standard data has not been entered into the system for parameter selector, WPS generation and defect analysis.
3. The system is written in PDC prolog, which has since then been replaced by Visual Prolog. The system uses MS-DOS as the operating system, which is less popular than Windows. The migration path to Visual Prolog from PDC is not straightforward.

#### **10.5 Capabilities of the Neuro-Expert System**

##### **10.5.1 Automation of Decision Making in Welding**

The system facilitates the process of decision making in welding by using a neuro-expert approach for decision making, controlling defects and trouble shooting in the welding domain.

### **10.5.2 Optimal Selection of Process**

The neuro-expert system selects a process that minimizes the cost for producing the welding job. The link between the Economic Evaluator module, the Parameter Selector and the Process Selector module enables the Neuro-Expert system to select optimum process.

### **10.5.3 Adjustment of Process Parameters and Welding Procedure to minimize Defects**

The system generates a complete weld procedure specification (WPS) for the welding job.

### **10.5.4 Economic Evaluation**

The Neuro-Expert System calculates the cost and time required for the welding job along with the weight of the weld metal deposited.

### **10.5.5 Defect Analysis**

The dual strategy used for defect analysis enables the user to model the weld process so that the resulting weld is predicted to be relatively defect-free. The input process parameters decided upon by the Parameter Selector module of the Neuro-Expert system (after economic evaluation) are fed to Defect Analyzer module, which predicts the defects that may be encountered for this set of process parameters and then the expert system adjusts the process parameters to achieve a defect free weld. The system also allows for interactive defect analysis.

### **10.5.6 Trouble Shooting**

A strategy similar to the one used in the interactive component of the Defect Analyzer was used for development of interactive trouble shooting module. Since it is very difficult to predict any trouble in advance of welding as they have no relationship to the welding

parameters. The WPS Tips Generator sub-module of the Trouble Shooter inserts some generic trouble shooting tips based on the selected process into the WPS which can be used as a handy reference during welding.

### **10.7 Scope of the Neuro-Expert System**

Although the Neuro-Expert system is currently developed with a limited scope, but it has means of adding additional knowledge so that it can work in a broader domain.

### **10.8 Validation of the Neuro-Expert System**

Once the *Neuro-Expert System* for welding has been developed, the various modules of the system were validated and the advantages of using this approach over other alternatives were investigated.

The performance of the *Neuro-Expert System* is compared to an human expert/expert data. Any shortcomings in the knowledge base would have become very evident. The decisions were compiled from welding handbooks for certain job variables and the results were compared with that of the Neuro-Expert system. The Neuro-Expert system agreed with the compiled decisions for 85-90% cases and hence the Neuro-Expert system performance is satisfactory. Also, the expert system did not give invalid decisions for any set of job variables.

### **10.9 Future Scope of Research**

No work is beyond improvements and the neuro-expert system developed in this research can also be improved in several ways. Some of the enhancements proposed as a conclusion are as follows:

1. If the system is directly interfaced with inspection and monitoring subsystems, then the system can be used to apply the corrective action in real time and eliminate the need of a welding operator.
2. If all the standard data becomes available then the system will be able to perform parameter selection and economic evaluation for processes like GTAW and SAW. Another approach might be to follow a rule based approach for parameter selection. The rules may be extracted by analyzing huge volumes of procedure data (available from industries/handbooks) and applying the concepts of data mining, decision-tree or neural network based extraction.
3. Some fuzzy logic concepts can be applied for determination of membership functions for various job variables and parameter in this research. We chose the categories for some parameters and variables on heuristic basis, which may differ, from one expert to another. For example, we considered a plate thickness of less than 5 mm to be THIN while others may argue that 3 mm should be the threshold.
4. As mentioned in chapter 7, the interface between the neural network for defect prediction and the expert system for parameter adjustment and the rules of the expert system are static. The algorithm, knowledge representation scheme and the prolog program can be developed for storing the expert system rules dynamically and extracting new rules and modification of existing rules using weight extraction can be extremely useful for the welding community.

5. Since, access to welding equipment was not there, we hope that someone will actually use the system in actual welding to test the performance of the system in real-world. If performing actual welding using the process and parameters selected by the system for different job variables, results in a good weld 85-90% of the time then the system will be considered to be validated.
6. The investigation of using an artificial expert system for the defect analyzer instead of the coupled system implemented in this research.



## REFERENCES

- Ajjanagdde V., Shastri L. Rules and Variables in Neural Nets, Neural Computation 3 (1991), pp. 121-134.
- Al-Tabtabai, H., A framework for developing an expert analysis and forecasting system for construction projects. Expert Systems with Applications 14(3) (1998), pp. 259-273
- Andrews, R., Diederich, J., Tickle, A.B., A survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-based Systems 8(6) (1995), pp.373-389,.
- Arc Welding Standard Data Handbook, The Welding Institute, U.K. 1975.
- ASME Boiler and Pressure Vessel Code Section 9 Welding and Brazing Qualifications, 1998
- AWS Standard Welding Procedure Specification for Gas Metal Arc Welding of Galvanized Steel, 1998.
- Barborak, D.M., Dickinson, Madigan, R.B., PC-Based Expert Systems and their Applications to Welding, Welding Journal, Jan 1991, pp. 29-39.
- Bench-Capon, T., Coenen, F., Nwana, H., Paton, R., Shave, M., Two aspects of the validation and Verification of Knowledge-Based systems. IEEE Expert, June 1993, pp. 76-81.
- Benitez, J.M., Castro, J.L., Requena, I., Are Artificial Neural Networks Black Boxes. IEEE transactions on Neural Networks 8 (5) (1997).
- Blum, A., Neural Networks in C++, John Wiley & Sons, Inc., 1992.
- Boger, Z., Application of Neural Networks to Water and Wastewater Treatment Plant Operation, ISA Transactions 31(1) (1992).
- Broner, Israel, Comstock, Carlton R., Combining expert systems and neural networks for learning site-specific conditions, Computers and Electronics in Agriculture 19 (1997), pp. 37-53.
- Caudill, M., Expert Networks. Byte, October 1991, pp. 108-116.

Chandel, R.S., The Effect of Process Variables on the Flux Consumption in Submerged Arc Welding. Materials and Manufacturing Processes 13(2) (1998), pp. 181-188,.

Chandrashekhara, A.S., Ananthapadmanabha, T., Kulkarni, A.D., A Neuro-Expert for planning and load forecasting of distribution systems. Electrical Power and Energy Systems 21 (1999), pp. 309-314.

Cun, Y. Le, Denker J. S., Solla, S. A., Optimal Brain Damage, Neur. Inform. Proc. Systems 2 (1990), pp. 598-605.

Czejdo, B., Eick, C.F., Taylor, M., Integrating sets, Rules, and Data in an object-oriented environment, IEEE Expert, February 1993.

Darwish, S. M., Al-Tamimi, A., and Al-Habdan, S., A knowledge base for metal welding process selection, Int. J. Mach. Tools Manufact., 37(7) (1997), 1007-1023.

Delgado, M., Gomez-Skarmeta A. F., Martin F., A methodology to model fuzzy systems using fuzzy clustering in a rapid prototyping approach, Fuzzy Sets and Systems 97 (1998), pp. 287-301.

DeLOACH, J.J., Null, C., Fiore, S., Konkol, P., The Right Welding Wire Could Help the U.S. Navy Save Millions. Welding Journal, June 1999, pp. 55-58.

Farag, W.A., Quintana, V.H., Lambert-Torres G., Genetic-based neuro-fuzzy approach for modeling and control of dynamical systems, IEEE transactions on Neural Networks 9(5) (1998).

Feuring, Thomas, Lippe Wolfram-M., The fuzzy neural network approximation lemma, Fuzzy Sets and Systems 102 (1999) pp. 227-236

Franco, C.V., Barbossa, R.P., Martinelli, A.E., Buschnelli, A.J.A., Study of the influence of welding parameters on the stress corrosion resistance of AISI 304 steel. Materials and Corrosion 49 (1998), pp. 496-504.

Fu, L., Neural-Network model for learning domain rules based on activation function characteristics. IEEE Transactions on Neural Networks 9(5) (1998), pp. 787-795.

Fu, L.M., Incremental Knowledge Acquisition In Supervised Learning Networks. IEEE transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 26, No.6, November 1996.

Fu, L.M., Knowledge-based Connectionism for Revising Domain Theories. IEEE transactions on Systems, Man, and Cybernetics, 23, No.1, January/February 1993.

Fu, L.M., Knowledge Base Refinement by Backpropagation. Data and Knowledge Engineering 7 (1991), North Holland, pp. 35-46.

Fukada, S., Yoshikawa, K., Determination of welding sequence: a neural net approach, 1990.

Fukuoka, Y., Matsuki, H., Minamitani, H., Ishida, A., A modified back-propagation method to avoid false local minima. Neural Networks 11 (1998) pp. 1059-1072.

Funabashi, M., Maeda, A., Morooka, Y., Mori, K., Fuzzy and Neural Hybrid Expert Systems: Synergetic AI, IEEE Expert 10 (4) (1995), pp. 32-40.

Gabor, K. , Andersen K., Cook, G.E., Barnett, R.J., Neural Network Methods for the modeling and control of welding processes, Journal of intelligent manufacturing 3 (1992), pp. 229-235.

Gaillard, A., Wunsch, D. C., Esscobedo, R. A., Neural hypercolumn architecture for the preprocessing of radiographic weld images, SPIE Vol. 1294 Applications of Artificial Neural Networks, 1990, pp. 378-388.

Giachino, J.W., and Weeks, W. Welding Skills and Practice, American Technical Society, 1976, pp. 76-107.

Goel, V., Singh, C.K., A knowledge based system for shielded metal arc welding. Proceedings of the Int. Conference on systems and signals, Delhi Dec 11-12, 1992.

Goel, V.<sup>1</sup>, Liao, W. T., Lee, K. S., Expert Shielded Metal Arc Welding. Computers in Industry 21 (1993), pp. 121-129.

Goel, V., A Knowledge-based approach for the selection of parameters and economic evaluation for shielded metal arc welding. M.E. Thesis, University of Roorkee, Roorkee, India (1991).

Goel, V.<sup>2</sup>, Liao, W. T., Lee, K. S., Shielded Metal Arc Welding Training - An Expert system approach. Proceedings of the Second International Conference on Flexible Automation and Information Management, FAIM 1992, Blacksburg, Virginia.

Hassibi, B., Stork, D., Wolff, G., Optimal Brain Surgeon", IEEE International Conference on Neural Networks 1 (1993), pp. 293-299.

Henderson, D.E., Kokotovic, P.V., Schiano, J.L., Rhode, D.S., Adaptive control of an arc welding process, IEEE Control Systems 13(1) (1993).

Hillman, D.A., Integrating Neural Nets and Expert Systems, AI Expert, June 1990.

Hongjun, L., Rudy S., and Huan, L., Effective Data Mining Using Neural Networks, IEEE Transactions on Knowledge and Data Engineering 8(6) (1996).

Hoskins, D.A., Hwang, J.N., Vagners, J., Iterative Inversion of Neural Networks and its Application to adaptive control. IEEE trans. on neural networks 3(2) (1992).

Howes, P., Crook, N., Using input parameter influences to support the decisions of feedforward neural networks. Neurocomputing 24 (1999), pp. 191-206

Huang, S.H., Endsley, M.R. Providing Understanding of the Behavior of Feedforward Neural Networks. IEEE transactions on Systems, Man, and Cybernetics 27(3) (1997), pp. 465-474.

Illingsworth, N. A practical guide to Neural Nets, Addison-Wesley Publishing Company Inc., 1990.

Jagielska, I., Matthews, C., Whitfort, T., An investigation into the application of neural networks, fuzzy logic, genetic algorithms, and rough sets to automated knowledge acquisition for classification problems. Neurocomputing 24 (1999), pp. 37-54

Jafar, M., Bahill, A. T., Interactive verification of knowledge based systems, IEEE Expert, February 1993.

Jawahar, N., Aravindan, P., Ponnambalam, S.G., Karthikeyan, A.A., A genetic algorithm-based scheduler for setup-constrained FMC. Computers in Industry 35 (1998), pp. 291-310

- Jiang, Bing, Felix, Henry Lau Chan, T. S., A process planning expert system based on a flexible digit length coding scheme, Expert Systems with Applications 14(3) (1998), pp. 291-301
- Jimmerson, D.G., Troubleshooting Robot Problems with Expert Systems. Welding Journal, Nov 1987, pp. 57-65.
- Jota, P.R.S., Islam, S.M., Wu, T., Ledwich, G., A class of hybrid intelligent system for fault diagnosis in electric power systems, Neurocomputing 23 (1998), pp. 207-224.
- Kerth, R. J., Knowledge based expert welding. Robots 9 Conf. Proc., June 2-6, 1985, Robotics International of the Society of Mechanical Engineers, Dearborn, MI.
- Kim, J.W., Na, S.J., A self-organizing fuzzy control approach to Arc Sensor for Weld Joint Tracking in Gas Metal Arc welding of Butt joints, Welding Journal 72(1993), pp. 60-66.
- Kittler, J., Hatef, M., Duin, R. P. W., and Matas, J., On Combining Classifiers, IEEE Trans. on Pattern Analysis and Machine Intelligence 20 (3) (1998), pp. 226-239.
- Kotecki, D., Welding Stainless Steel, Advanced Materials & Processes 5 (1999), pp. 41-44..
- Krishnan, R., Sivkumar, G., Bhattacharya, P., A search technique for rule extraction from trained neural networks, Pattern Recognition Letters 20 (1999), pp. 273-280.
- Kuo, R.J., Cohen, P.H., Multi-sensor integration for on-line tool wear estimation through radial basis function networks and fuzzy neural network. Neural Networks 12 (1999) 355-370.
- Lee, N. I., A Coupled-ART neural network capable of modularized categorization of patterns, Pattern Recognition Letters 20 (1999), pp. 131-140.
- Lee, Sunro, O'Keefe, Robert M., The effect of Knowledge Representation Schemes on Maintainability of Knowledge-Based Systems, IEEE Transactions on Knowledge and Data Engineering 8(1) (1996).
- Lewis, F.L., Neural Network Control of Robot Manipulators, IEEE Expert 11(3) (1996), pp. 64-75.
- Liao T. W., Li, D., Two manufacturing applications of the Fuzzy K-NN algorithm. Fuzzy sets and Systems 92 (1997), pp. 289-303.

Liao T. W., Goel, V., Lee, K. S., A Neuro-Expert System for Modeling and Planning of Automated MIG Welding. International Conference on Modeling and Control of Welding Process, Dec. 8-10, 1993, Orlando, FL.

Liao, T. W., Tang, K., Automated extraction of welds from digitized radiographic images based on MLP neural networks, AAI 11 (1997), pp. 197-218.

Looney, C.G., Neural networks as Expert Systems, Expert systems with Applications 6 (1993), pp. 129-136.

Lovell, B.C., Bradley, A.P., The Multiscale Classifier. IEEE transactions on Pattern Analysis and Machine Intelligence 18(2) (1996).

Looney, C.G., Advances in Feedforward Neural networks: Demystifying Knowledge Acquiring Black Boxes. IEEE transactions on Knowledge and Data Engineering, 8, No. 2, April 1996 pp. 211-228.

Lu, H., Setiono, R., Liu, H., Effective Data Mining using Neural Networks. IEEE transactions on Knowledge and Data Engineering 8(6) (1996).

Lucas, W., Expert systems for welding. Weld Des. Fabr. 60 (7) (1987), pp. 47-50.

Lucas, W., Expert systems for welding, Welding Journal, April 1987, pp. 19-26.

Lucas, W., and Brightmore, A.D., Expert systems for welding engineers, Metal Construction, May 1987, 254-260.

Luchetta, A., Manetti, S.F., Forecast: A Neural System for Diagnosis and Control of Highway Surfaces. IEEE INTELLIGENT SYSTEMS 1998, pp. 1094-7167

Luke, S. O. Selecting FCAW Electrodes to ensure Proper Weld Deposit Hardness, Welding Journal, May 1999, pp. 39-42.

Loukatzikos, C. and Galletly, J. E., Constructing neural networks from expert system rules. J. of Microcomputer Applications 17 (1994), pp. 35-53.

Magrez, P., Rousseau, A., A symbolic interpretation for back-propagation networks. International Journal of Intelligent systems 7 (1992), pp. 339-360.

Matsatsinis, N. F., Doumpos M., Zopounidis, C., Knowledge Acquisition and Representation for Expert Systems in the Field of Financial Analysis, Expert Systems with Applications 12(2) (1997), pp. 247-262.

McConnel, I. A., McPherson, N. A., The Application of Statistical Process Design to a FCAW Process. Welding Research Supplement, October 1997.

Medsker, L. R., Bailey, D. L., Models and Guidelines for Integrating Expert Systems and Neural Networks, Hybrid Architectures for Intelligent Systems, ed. By A. Kandel and G. Langholz, Boca Raton, FL, CRC Press, 1992.

Meneganti, M., Saviello, F.S., Tagliaferri, R., Fuzzy neural networks for classification and detection of anomalies, IEEE transactions on Neural Networks 9(5) (1998), pp. 848-861.

Mengshoel, O. J., Knowledge Validation: Principles and Practice, IEEE Expert, June 1993.

Mookerjee, V. S., Mannino, M. V., Sequential Decision Models for Expert System Optimization, IEEE Transactions on Knowledge and Data Engineering 9(5) (1997), pp. 675-685.

Moon, H. S. and Na, S. J., A neuro-fuzzy approach to select welding conditions for welding quality improvement in horizontal fillet welding, J. Manufacturing Systems 15(6), (1996), pp. 392-403.

Moore, Trevor T., Objective vs. Subjective Measures of Error-Proneness for Rule-based Programs, IEEE Transactions on Knowledge and Data Engineering 10(6) (1998), pp. 1008-1012.

Miller, Duane K., What every Engineer Should Know about Welding Procedure, Welding Journal, Aug 1999

Murrell, Stephen, Plant, Robert T. A survey of tools for the validation and verification of knowledge-based systems: 1985-1995. Decision Support Systems 21(4) (1997), pp. 307-323

Nadkarni, S.V. Modern Arc Welding Technology, Oxford and IBH Publishing Company Co. Pvt. Ltd. ,1988, pp. 82-86.

Napolitano, R. J., Kulluk, H., Nagurka, M. L., Martukanitz, R. P., and Dickerson, P. B., Development of knowledge-based systems for aluminum welding, Welding J., December 1992, pp. 43-47.

Narazaki, H., Watanabe, T., Yamamoto, M., Reorganizing Knowledge in Neural Networks: An Exploratory Mechanism For Neural Networks in Data Classification Problems, IEEE transactions on systems, man, and Cybernetics - Part B: Cybernetics 26(1) (1996).

Nishina, T., Hagiwara, M., Fuzzy inference neural network. Neurocomputing 14(3) (1997), pp. 223-239.

Nuclear Engr. Int., Framatome develops an expert system to characterize defects, Dec. 1987.

O'Keefe, R.M., Balci, O., Smith, E.P. Validating Expert Systems Performance, IEEE Expert, Winter 1987, pp. 81-88.

Onda, H., Nishinaga, Y., and Ono, K., Welding defect identification by artificial neural networks, JAPAN/USA Symposium on Flexible Automation 1 (1992), 313-316.

Oskada, K., Yang, G. Application of neural networks to an expert system for cold forging. Int. J. Mach. Tools Manufac. 31 (4) (1991), pp. 577-587.

Osyk, B. A., Vijayraman, B. S., Integrating Expert Systems and Neural Nets: Exploring the Boundaries of AI. Information Systems Management, Spring 1995.

Parsons, S., Current Approaches to Handling Imperfect Information in Data and Knowledge Bases. IEEE Transactions on Knowledge and Data Engineering, 8 (3) (1996).

Paton, B. E., Troitskii, V. A., and Lukash, A. G., Principles of constructing expert systems for evaluating quality of welded components, Technical Diagnostics and NDT, 2(4) (1990), pp. 3-9.

Petridis, V., Paterakis, E., Kehagias, A., Hybrid neural-genetic multimodel parameter estimation algorithm, IEEE transactions on neural networks 9(5) (1998).

Pierre S. and Martin D.L., MARS: An expert robot welding system, Machine Intelligence and Knowledge Engineering for Robotics Applications, NATO ASI Series, F33, Springer-Verlag, Berlin, 1987.

Polat, F., Guvenir, H. A., UVT: A unification-based tool for knowledge base verification, IEEE Expert, June 1993.



Ponnappalli, P. V. S., Ho, K. C., Thomson, M., A Formal Selection and Pruning Algorithm for Feedforward Artificial Neural Network Optimization, IEEE transactions on Neural Networks, 10 (4) (1999).

Porto, V .W., Fogel, D. B., Fogel, L. J., Alternative Neural Network Training Methods, IEEE Expert: Intelligent Systems and their Application 10 (3) (1995), pp. 16-22.

Prechelt, Lutz. Connection Pruning with static and adaptive pruning schedules, Neurocomputing 16(1) (1997), pp. 49-61.

Reeves, R.E., Manley, T.D., Potter A., and Ford D.R., Expert system technology - An avenue to an intelligent weld process control system, Welding Journal, June 1988, pp. 33-41.

Rosen, E.M., Silverman, D.C., Corrosion Prediction from Polarization Scans, Using an Artificial Neural Network Integrated with an Expert System, Corrosion 48(9) (1992), pp. 734-744.

Setiono, R., Leow, W.K., On mapping decision trees and neural networks, Knowledge based systems 12 (3) (1999), pp. 95-99.

Setiono, Rudy, Liu, Huan., Symbolic Representation of Neural Networks, IEEE Computer, March 1996.

Setiono, R., Leow, W.K., On mapping decision trees and neural networks, Knowledge-Based Systems 12 (1999), pp. 95-99.

Shin, C.K., Park, S.C., Memory and neural network based expert system. Expert Systems with Applications 16 (1999), pp. 145-155.

Shirali, A.A., Mills, K.C., The Effect of Welding Parameters on Penetration in GTA Welds, Welding Research Supplement, July 1993, pp. 347-353.

Smartt, H. B., Johnson, J. A., and Einerson, C. J., The role of intelligent systems in weld process control, Materials Evaluation, October 1993, pp. 1166-1173.

Smets, H. M. G., Boegarts, W. F. L., SAC Analysis of austenitic stainless steels in Chloride-bearing water by neural network techniques, Corrosion, 48(9) (1992), pp. 618-623,.

Smith, D., Welding Skills and Technology, McGraw-Hill, Inc. 1984.

Subramaniam, S., White, D.R., Jones, J.E., Lyons, D.W. Experimental Approach to Selection of Pulsing Parameters in Pulsed GMAW. Welding Research Supplement, May 1999, pp. 166-172.

Sugitani, Y. and Mao, W., Automatic simultaneous control of bead height and back bead shape using an arc sensor in one-sided welding with a backing plate, Welding Int. 9(5) (1995), pp. 9-17.

Sun, C. K., Chan, C. W., Tontiwachwuthikul, P., Intelligent diagnostic system for a solar heating system, Expert Systems with Applications 16 (2) (1999), pp. 157-171

Szabo, R.R., Pandya, A.S., Szabo, B., A Knowledge based system using a Neural Network, SPIE 1468 Applications of Artificial Intelligence IX (1991).

Taylor, A. An investigation of knowledge based models for automated procedure generation in arc welding domain. Int. J. Prod. Res. 27(11) (1989), pp. 1855-1862.

The Procedure Handbook of Arc Welding. The Lincoln Electric Company, 1973.

Thompson, D.R., Ray, A., Kumara, S. A hierarchically structured knowledge-based system for welding automation and control. Journal of Engineering for Industry, February 1988, pp. 110-120.

Tickle, A. B., Andrews, R., G., Mostefa, Diederich, J., The truth will Come to Light: Directions and Challenges in Extracting the Knowledge Embedded Within Trained Artificial Neural Network, IEEE Transactions on Neural Networks 9 (6) (1998), pp. 1057-1067.

Tsai<sup>1</sup>, W., Vishnuvajjala, R., Zhang, Du., Verification and Validation of Knowledge-based Systems, IEEE Transactions on Knowledge and Data Engineering 11 (1) (1999).

Tsai<sup>2</sup>, Jeffrey J.P., Liu, A., Juan, E., Sahay, A., Knowledge-Based Software Architectures: Acquisition, Specification, and Verification, IEEE Transactions on Knowledge and Data Engineering 11 (1) (1999).

Vercesi, J., Surian, E. The effect of Welding Parameters on High-Strength SMAW All-Weld-Metal – Part 2: AWS E10018-M and E12018-M, Welding Research Supplement, May 1998.

Wan A., W.A.T. Logic programming on a neural network, International Journal of Intelligent systems, 7, pp. 513-519, 1992.

Wetst, G., Vanthienen J., Piramuthu S., A Tabular Knowledge-Based Framework with Feature Selection. Expert Systems With Application, 13(2) (1997), pp. 101-119,

Wong, C., Lin, N., Rule extraction for fuzzy modeling. Fuzzy Sets and Systems 88(1) (1997) 23-30

Xu, K., Luxmoore, A.R., Jones, L.M., Deravi, F., Integration of neural networks and expert systems for microscopic wear particle analysis. Knowledge-Based Systems 11 (1998) 213–227

Zheng, X. M., Lucas, J., Fang, M. T. C., Neural Networks for parameter prediction and quality inspection in TIG welding. Trans Inst MC Vol 15 No.2, 1993.

Zhao, Xiao-Jia M., Dillon, Tharam S., Theoretical and Practical Considerations of uncertainty and complexity in Automated Knowledge Acquisition, IEEE Transactions on Knowledge and Data Engineering, 7 (5) (1995).

## **INTERNET WEB-SITE REFERENCES**

<http://www.aws.org/>

<http://www.millerwelds.com/education>

<http://www.liquidarc.com.au/>

<http://www.pro-fusiononline.com/welding/>

## **GLOSSARY**

<b>Activation Function</b>	A function used by a neural network neuron to compute the output or activation of the unit. Common activation functions include sigmoid, hyperbolic tangent, signum, and Gaussian functions. Also referred to as the transfer function.
<b>Decay</b>	Indicates the degree to which a weight can decay over a time period.
<b>Momentum</b>	The momentum factor determines the proportion of the last weight change that is added into the new weight change. It affects how quickly learning accelerates or decelerates depending on the differences between the desired pattern and the estimated pattern. Using momentum can often decrease training times by moving weights faster in the direction of prior changes.
<b>Initial Range</b>	Initialization Weight Range
<b>Learning Rate</b>	During the learning process, each time a pattern is presented to the network, the weights leading to an output node are modified slightly in the direction required to produce a smaller error the next time the same pattern is presented. The amount of weight modification is the learning rate times the error. Usually used to refer to how much weights in the backpropagation algorithm are changed on each update in neural network training.
<b>Tolerance</b>	The acceptable range of Error for the Output Activation of a neural network.
<b>Backpropagation</b>	A multilayer feedforward neural net architecture which uses the supervised mode of learning. This is the most widely used type of neural net.
<b>Hidden Layer</b>	A layer of processing elements between a NN's input layer and its output layer.
<b>Input Layer</b>	A layer of processing elements that receives the input to a neural net.
<b>Neural Network (NN)</b>	A system modeled after the neurons (nerve cells) in a biological nervous system. A neural network is designed as an interconnected system of processing elements, each with a limited number of inputs and outputs. Rather than being programmed, these systems learn to recognize patterns.
<b>Output Layer</b>	The layer of processing elements which produce a neural net's output

## **APPENDIX**

## APPENDIX A - EXPERT SYSTEM RULES

### A.1 Conversion of Quantitative Job Variables to Qualitative Values in Process Selector

Rule in Prolog Program	Rule in English
Seam_length_crit (Seam_Length,Factor) :- Seam_Length > 10, Factor = "HIGH".	If Seam Length is greater than 10 meters then assign a factor of HIGH to Seam Length.
Seam_length_crit (Seam_Length,Factor) :- Seam_Length <= 10, Factor = "LOW".	If Seam Length is lesser than or equal to 10 meters then assign a factor of HIGH to Seam Length.
Plate_thickness_crit (Plate_thickness,Factor) :- Plate_thickness > 10, Factor = "HIGH".	If Plate Thickness is greater than or equal to 5 mm then assign a factor of HIGH to Plate Thickness.
Plate_thickness_crit (Plate_thickness,Factor) :- Plate_thickness > 5, Plate_thickness < 10, Factor = "MEDIUM".	If Plate Thickness is greater than 5 mm and less than 10 mm then assign a factor of MEDIUM to Plate Thickness.
Plate_thickness_crit (Plate_thickness,Factor) :- Plate_thickness <= 5, Factor = "LOW".	If Plate Thickness is less than or equal to 5 mm then assign a factor of LOW to Plate Thickness.
Strength_crit (Strength,Factor) :- Strength <= 70000, Factor = "LOW".	If desired Strength is less than 70000 psi then assign a factor of LOW.
Strength_crit (Strength,Factor) :- Strength > 70000, Factor = "HIGH".	If desired Strength is more than 70000 psi then assign a factor of HIGH.
Leg_length_crit (Leg_length,Factor):- Leg_length >= 10, Factor = "HIGH".	If Leg Length is greater than or equal to 10 mm then assign a factor of HIGH to Leg Length.
Leg_length_crit (Leg_length,Factor):- Leg_length < 10, Factor = "LOW".	If Leg Length is less than 10 mm then assign a factor of HIGH to Leg Length.

## A.2 Validation Rules for Expert System for Process Selection

Rule in Prolog Program	Rule in English
<pre>validate_position(Position,"neural",Y):-     Y = "SAW",     Position = "FLAT",     write("\n\nSAW can be used as the position is flat"),nl,!,</pre>	SAW can be used if the position is FLAT.
<pre>validate_position(Position,"neural",Y):-     Y = "SAW",     Position = "HORIZONTAL",     Weld Type = "FILLET"     write("\n\nSAW can be used as the position is Horizontal in a          FILLET Weld"),nl, !,     retract(process(X,Y)).</pre>	SAW can be used if the position is HORIZONTAL in a Fillet Weld.
<pre>validate_position(Position,"neural",Y):-     Y = "SAW",     Position &lt;&gt; "FLAT",     write("\n\nSAW cannot be used as the position is not flat"),nl,     retract(process(X,Y)).</pre>	SAW cannot be used if the position is not FLAT.
<pre>Validate_plate_thickness(Plate_thickness,X,Y):-     Y = "SAW",     Plate_thickness &lt;= 5,     write("\n\nSAW cannot be used as the plate is too thin"),nl,     retract(process(X,Y)). validate_plate_thickness(_,_,_).</pre>	SAW cannot be used if the plate thickness is less than 5 mm.
<pre>validate_automation(Automation,"neural",Y):-     Y = "SMAW",     Automation = "YES",     write("\n\nSMAW cannot be used as Automation is not possible"),nl,     retract(process(X,Y)).</pre>	If Automation is required then SMAW cannot be used.
<pre>validate_automation(Automation, "neural", Y):-     Y = "SAW",     Automation = "NO",     write("\n\n SAW cannot be used as Automation is not         required"),nl,     retract(process(X,Y)).</pre>	If Automation is not required then SAW cannot be used.

### A.3 SMAW ELECTRODE SELECTION RULES

Rule in Prolog Program	Rule in English
electrode("E6010","MILD STEEL","ALL","ALL","LOW","FLAT")	Use E6010 for Mild Steel for ALL positions, Flat Bead, Low Strength and all joint types.
Electrode("E6011","MILD STEEL","ALL","ALL","MEDIUM","FLAT")	Use E6011 for Mild Steel for ALL positions, Flat Bead, Low Strength and all joint types.
Electrode("E6011","STAINLESS STEEL","ALL","ALL","HIGH","FLAT")	Use E6011 for Stainless Steel for ALL positions, Flat Bead, High Strength and all joint types.
Electrode("E6012","MILD STEEL","ALL","ALL","LOW","CONVEX")	Use E6012 for Mild Steel for ALL positions, Convex Bead, Low Strength and all joint types.
Electrode("E6013","MILD STEEL","ALL","ALL","LOW","FLAT")	Use E6013 for Mild Steel for ALL positions, Flat Bead, Low Strength and all joint types.
Electrode("E6015","MILD STEEL","ALL","ALL","LOW","FLAT/CONVEX")	Use E6015 for Mild Steel for ALL positions, Flat/Convex Bead, Low Strength and all joint types.
Electrode("E6020","MILD STEEL","FH","ALL","LOW","FLAT/CONCAVE")	Use E6020 for Mild Steel for Flat/Horizontal positions, Flat/Concave Bead, Low Strength and all joint types.
Electrode("E6027","MILD STEEL","FH","ALL","LOW","FLAT")	Use E6027 for Mild Steel for Flat/Horizontal positions, Flat Bead, Low Strength and all joint types.
Electrode("E7010","MILD STEEL","ALL","ALL","MEDIUM","FLAT/CONCAVE")	Use E7010 for Mild Steel for ALL positions, Flat/Concave Bead, Medium Strength and all joint types.
Electrode("E7014","MILD STEEL","ALL","ALL","MEDIUM","FLAT")	Use E7014 for Mild Steel for ALL positions, Flat Bead, Medium Strength and all joint types.
Electrode("E7013","MILD STEEL","ALL","ALL","MEDIUM","FLAT")	Use E7013 for Mild Steel for ALL positions, Flat Bead, Medium Strength and all joint types.
Electrode("E7016","MILD STEEL","ALL","ALL","MEDIUM","FLAT")	Use E7016 for Mild Steel for ALL positions, Flat Bead, Medium Strength and all joint types.
Electrode("E7018","MILD STEEL","ALL","ALL","MEDIUM","FLAT")	Use E7018 for Mild Steel for ALL positions, Flat Bead, Medium Strength and all joint types.
Electrode("E8016","MILD STEEL","ALL","ALL","HIGH","FLAT")	Use E8016 for Mild Steel for ALL positions, Flat Bead, High Strength and all joint types.
Electrode("E8018G","MILD STEEL","ALL","ALL","HIGH","FLAT")	Use E8018G for Mild Steel for ALL positions, Flat Bead, High Strength and all joint types.



#### A.4 RULES FOR EXPERT SYSTEM FOR TROUBLESHOOTING

##### A.4.1 Welding Set Related Troubles

Welding does not start		
Sub-Category	Possible Causes	Remedies
Does not start at all	Line Switch not turned on	Place Line switch on 'ON' position
	Supply Line Fuse Blown	Replace the Fuse
	Power Circuit Dead	Check Input Welding
	Overload Relay Tripped	Check for Overloading and switch on the Relay
	Loose or broken power, electrode or ground lead	Replace Lead or tighten and repair connection
	Wrong Voltage	Check input against Instrument
	Polarity Switch not centered	Center Switch handle on +,- or AC
Starts but blows fuse	Open circuit to Starter Switch	Repair Starter Switch
	Short Circuit in motor or other connection due to high current	Check setting and output with Ammeter
	Fuse too Small	Replace fuse
Welds but stops soon	Improper Ventilation	Make sure all case openings are free for proper air circulation
	Overloading due to welding in excess of rating	Operate welder at rated load
	Fan may be inoperative	Check leads and connection of fan
	Motor Generator Set Problems	Check name plate for correct motor speed
	Wrong direction of travel	Check thermostat's coils and other components
Would not shut off	Brushes worn/missing	
	Line switch may have failed mechanically	Check it and replace it if found defective
Polarity Switch does not turn	Contacts may have become rough and pitted due to improper turning under load	Replace polarity switch
Arcing at ground clamp	Connection may be loose or spring may be weak	Tighten connection or replace clamp
Electrode Holder becomes hot	Loose connection to holder / the duty cycle may be inadequate	Tighten connection or replace holder and use proper duty cycle

#### **A.4.2 Arc Related Troubles**

<b>Trouble Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
<b>Variable and Sluggish Arc</b>	<b>Current too low</b>	<b>Check recommended currents for rod type and size being used</b>
	<b>Low Line Voltage</b>	<b>Check with power company</b>
	<b>Welding Leads too small</b>	<b>Use recommended welding leads</b>
	<b>Improper connections</b>	<b>Check all connections. Clean, repair or replace as required</b>
	<b>Improper Weld Set Brushes or Springs</b>	<b>Clean and repair</b>
	<b>Rough or Dirty Commutator</b>	<b>Turn down or clean commutator</b>
<b>Load and Spattery</b>	<b>Current Setting is too high</b>	<b>Check setting and output with ammeter</b>
	<b>Polarity may be wrong</b>	<b>Try reversing polarity or an electrode of reverse polarity</b>
<b>Unstable and Goes Out</b>	<b>Arc may be too long</b>	<b>Shorten Arc for proper penetration</b>

#### **A.4.3 Wire Feeder Related Troubles**

<b>Trouble Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
<b>Wire Feed Unit stops feeding</b>	<b>Electrical Fuse Blown</b>	<b>Have qualified electrician check electric power circuits</b>
	<b>Control relay of MIG gun defective</b>	<b>Replace defective relay switches</b>
	<b>Loss of wire feed roller tension</b>	<b>Inspect and replace all defective feed drive rolls</b>
<b>Wire feeds but no arcing</b>	<b>Electrical fuse blown in primary, power source or wire feed unit</b>	<b>Have qualified electrician check electrical power circuits</b>
<b>Wire Electrode stubs</b>	<b>Excess slope</b>	<b>Reduce Slope</b>
	<b>Low Voltage</b>	<b>Increase Voltage</b>
	<b>High Wire Feed</b>	<b>decrease wire feed</b>

#### A.4.4 Flux Feeder Related Troubles

Trouble Category	Possible Causes	Remedies
Flux Feeding stops	Flux tube in the gun will be blocked	Blow out hose with air if necessary
	Magnetic particles may cause binding	If there is no flux in the tank, check the tank outlet for large pieces of slag
	Flux Feed Unit Stops	Set Pressure Regulator at 26-30 lbs/sq.in. For gun cable extension assembly, set pressure for 55-60 lbs/sq.in.
Flux Stoppage not in gun	Piece of Slag in tank	Remove the gun from the cable. Check the tube in the gun and the cable handle
	Flux tank outlet may be clogged	Pass flux over a magnetic separator when filling the flux tank.
	Pressure in the tank may be set improperly	Decrease torch travel speed
Excessive Air Blow/ Uneven flux flow	Tank may be empty	Refill the tank
	Flux may be falling away from the weld faster than it is being fed	Alter procedure or make a flux dam
	Pressure in the tank may be too high	Set Pressure Regulator at 26-30 lbs/sq.in. For gun cable extension assembly, set pressure for 55-60 lbs/sq.in.
Flux in tanks gets wet	There may be water in the Airline	Blow out Airline before connecting them to the tank
	The Copper bleeder may be clogged	Make sure that a slight amount of air is escaping from the crimped end of the copper tube below the flux tank

#### A.4.5 Miscellaneous Troubles

Trouble Category	Possible Causes	Remedies
Spatter	Excess gas flow	Reduce gas flow
	Insufficient Slope setting	Adjust Slope
	Excessive Voltage	Reduce Voltage
	Wrong shielding gas	Use recommended gas coverage
	Wrong Electrode	Use recommended wire electrode
Bad Arc Starts	Wrong Voltage	Adjust Voltage
	Wrong Slope	Adjust Slope
	Wrong Inductance	Adjust Inductance
	Glass Slag	Chip and brush welds
Fluid Puddle	Wrong Inductance	Adjust Inductance
	Wrong Electrode	Use recommended wire electrode
	Wrong shielding gas	Use recommended gas coverage
Rigid Puddle	Wrong Inductance	Adjust Inductance
	Wrong Electrode	Use recommended wire electrode
	Wrong shielding gas	Use recommended gas coverage
Convex Deposit	Wrong Voltage or Wire Feed	Adjust Controls
	Wrong Inductance	Adjust Inductance
	Wrong torch manipulation	Use recommended torch motion
	Wrong travel speed	Increase travel speed
Concave Deposit	Wrong Voltage or Wire Feed	Adjust Controls
	Wrong torch manipulation	Use recommended torch motion
	Wrong travel speed	Decrease travel speed
Weld remains in balls	Electrode being used may be wrong	Use proper electrode

## A.5 RULES FOR EXPERT SYSTEM FOR DEFECT ANALYSIS

### A.5.1 Defect Analysis Rules for Shielded Metal Arc Welding (SMAW)

Defect Category	Possible Causes	Remedies
Undercuts	Excessive Weld Current	Reduce Welding Current
	High Speed of Arc Travel	Reduce travel speed
	Wrong Electrode Angle	Change Electrode Angle so that Arc Force holds the metal in the corners
	Excessive Side Manipulation	Avoid excessive weaving
	Damp or improperly formulated electrode	Use dry Electrodes
Spatter	High Current	Check current against manufacturer's recommendations
	Wrong Polarity	Try reversing polarity/Use Electrode of reverse polarity
	Too long Arc Length	Try using a shorter arc length
	Wet Electrode	Bake electrodes before use
	Wrong Electrode	Use recommended Electrode
Arc Blow	Change to AC Welding	
	Use lower currents and smaller electrodes	
	Reduce Arc Length	
	Weld in direction of Blow	
	Change the electrical path in work by: a) Shifting work connection to other end b) Welding towards heavy tacks, finished welds or back-stepping on long welds c) Using run-out tabs, adding steel blocks to change work current path or tacking small plates	
Porosity/Surface Holes	Work piece dirty	Remove Scale/Dirt/Rust/Paint/Moisture from the joint
	Steel has low carbon	Use low hydrogen electrodes. Minimize admixture of base weld by using low currents and faster travel speeds for less penetration
	Steel has high Manganese	Use low hydrogen electrodes. Minimize admixture of base weld by using low currents and faster travel speeds for less penetration

<b>Defect Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
	Steel has high Sulfur or Phosphorus	Use low hydrogen electrodes. Minimize admixture of base weld by using low currents and faster travel speeds for less penetration
	Too long an Arc Length	Try using a shorter Arc Length
Poor Fusion	Insufficient gap between two plates	Provide better fit up or use a weave technique to fill up the gap
	Dirty Edges	Remove Scale/Dirt/Rust/Paint/Moisture from the joint
	Current too low	Try a higher current and use stringer bead technique
Shallow Penetration	Incorrect Electrode Size	Use smaller electrode to reach down into narrow grooves
	Low Welding Current	Use higher current
	Inaccurate joint preparation	Allow some gap at the bottom of the joint

#### **A.5.2 Defect Analysis Rules for Gas Metal Arc Welding (GMAW)**

<b>Defect Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
Undercuts	Wrong Torch Manipulation	Use recommended torch manipulation
	Wrong Shielding Gas	Use recommended Shielding Gas
Spatter	Excessive/insufficient gas flow	Adjust gas flow
	Insufficient Slope Setting	Adjust Slope
	Excessive Voltage	Adjust Voltage
	Wrong Shielding gas	Use recommended shielding gas
	Wrong Electrode	Use recommended Electrode
Arc Blow	Use lower currents and smaller electrodes	
	Reduce Arc Length	
	Weld in direction of Blow	
	Change the electrical path in work by: a) Shifting work connection to other end b) Welding towards heavy tacks, finished welds or back-stepping on long welds c) Using run-out tabs, adding steel blocks to change work current path or tacking small plates	
Porosity/	Work piece dirty	Clean base metal with Stainless Steel Brush or

<b>Defect Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
Surface Holes		chemicals
	Faulty Gas System	Check and replace the defective components of the system
	Faulty Torch Assembly	Replace the contact tip centering device
	Improper torch manipulation	Use recommended torch manipulation
	Spatter Clogged Nozzle	Clean the nozzle
Poor Fusion	Insufficient gap between two plates	Provide better fit up or use a weave technique to fill up the gap
	Dirty Edges	Remove Scale/Dirt/Rust/Paint/Moisture from the joint
	Current too low	Try a higher current and use stringer bead technique
Shallow Penetration	Incorrect Electrode Size	Use smaller electrode to reach down into narrow grooves
	Low Welding Current	Use higher current
	Inaccurate joint preparation	Allow some gap at the bottom of the joint
Convex Deposit	Wrong Voltage/Wire Feed	Use correct Voltage/Wire Feed
	Wrong Inductance	Use wrong Inductance
	Wrong Torch Manipulation	Use recommended torch manipulation
	Wrong Travel Speed	Increase travel speed
Concave Deposit	Wrong Voltage/Inductance	Use correct Voltage/Inductance
	Wrong Torch Manipulation	Use recommended torch manipulation
	Wrong Travel Speed	Decrease travel speed

### **A.5.3 Gas Tungsten Arc Welding (GTAW)**

<b>Defect Category</b>	<b>Possible Causes</b>	<b>Remedies</b>
Undercuts	Wrong Torch Manipulation	Use recommended torch manipulation
	Wrong Shielding Gas	Use recommended Shielding Gas
Spatter	Excessive/ insufficient gas flow	Adjust gas flow
	Insufficient Slope Setting	Adjust Slope

	Excessive Voltage	Adjust Voltage
	Wrong Shielding gas	Use recommended shielding gas
	Wrong Electrode	Use recommended Electrode
Arc Blow		Use lower currents and smaller electrodes
		Reduce Arc Length
		Weld in direction of Blow
		Change the electrical path in work by: a) Shifting work connection to other end b) Welding towards heavy tacks, finished welds or back-stepping on long welds c) Using run-out tabs, adding steel blocks to change work current path or tacking small plates
Porosity/ Surface Holes	Work piece dirty	Clean base metal with Stainless Steel Brush or chemicals
	Trapped Gas/Travel Speed too fast	Correct travel speed
	Wrong Gas Flow	Increase gas flow
Poor Fusion	Insufficient gap between two plates	Provide better fit up or use a weave technique to fill up the gap
	Dirty Edges	Remove Scale/Dirt/Rust/Paint/Moisture from the joint
	Current too low	Try a higher current and use stringer bead technique
Shallow Penetration	Incorrect Electrode Size	Use smaller electrode to reach down into narrow grooves
	Low Welding Current	Use higher current
	Inaccurate joint preparation/ Fitup	Allow some gap at the bottom of the joint
Tungsten Inclusion	Touching work with electrode	Use high frequency on continuous for AC or start only for DC polarity
	Tungsten Melting and mixing with base metal	Use less current/increase diameter of the Tungsten Electrode
	Shattering of Tungsten with high input	Make sure electrode ends are properly prepared or increase the diameter of the electrode to carry the high currents



#### A.5.4 Submerged Arc Welding (GTAW)

Defect Category	Possible Causes	Remedies
Undercuts	Wrong Torch Manipulation	Use recommended torch manipulation
	Wrong Flux	Use recommended Flux
Porosity	Work piece dirty	Clean base metal with Stainless Steel Brush or chemicals
	Electrode Contamination	Use clean and dry electrode
	Insufficient flux coverage	Provide sufficient Flux coverage
	Contaminants in the flux	Use clean and dry flux
	Entrapped flux at the bottom of joint	Use proper backing
	Segregation of constituents	use proper joint preparation and weld preparation
	Excessive travel speed	Reduce travel speed
Poor Fusion	Slag residue from tack welds made with covered electrodes	Use electrode which do not leave slag residue
	Insufficient gap between two plates	Provide better fit up or use a weave technique to fill up the gap
	Dirty Edges	Remove Scale/Dirt/Rust/Paint/Moisture from the joint
	Current too low	Try a higher current and use stringer bead technique
Shallow Penetration	Incorrect Electrode Size	Use smaller electrode to reach down into narrow grooves
	Low Welding Current	Use higher current
	Inaccurate joint preparation/ Fitup	Allow some gap at the bottom of the joint

## **A.6 Sample Rules For WPS Tips Generation Modele of Defect Analyzer**

For a complete set of rules please refer to the prolog program in appendix E.

### **A.6.1 Shielded Metal Arc Welding (SMAW)**

```
IF defect_predicted = Slag Inclusions
AND Process = 'SMAW'
THEN wps_tip1 = "Use Preheat and higher Weld heat input"
AND wps_tip2 = "Proper deslagging in between passes"
AND wps_tip3 = "Avoid joint design contour difficult to penetrate with
Arc"
```

```
IF defect_predicted = Undercuts
AND Process = ('SMAW' or 'GMAW')
THEN wps_tip1= "Shorten Arc Length"
AND wps_tip2 = "Change Angle of Electrode so that arc force fills
undercut"
```

```
IF defect_predicted = (Porosity OR Cracking)
AND Process = 'SMAW'
THEN wps_tip1 = "Bake Electrodes before Use"
AND wps_tip2 ="Clean the Plates thoroughly with a wire brush"
AND wps_tip3 ="Provide Pre-heating and Interpass Heating"
AND wps_tip4 ="Use Weaving Techniques"
AND wps_tip5 ="Back-step to fill craters"
```

## A.7 Sample Rules for the WPS Generator

Prolog Rule	Meaning in English
<pre>preheating_req ("GMAW",Plate_Thickness,                 "CARBON STEEL","Ar He",                 Preheat,Interpassheat):-     Plate_Thickness &lt;= 20,     Preheat = "20 degree C",     Interpassheat = "20 degree C".</pre>	Preheat before and between passes upto 20° C for GMAW of Carbon Steel of thickness less than 20 mm when using Ar He shielding gas.
<pre>preheating_req ("GMAW",Plate_Thickness,                 "CARBON STEEL","Ar He",                 Preheat,Interpassheat):-     Plate_Thickness &gt; 20,     Plate_Thickness &lt;= 40,     Preheat = "70 degree C",     Interpassheat = "70 degree C".</pre>	Preheat before and between passes upto 70° C for GMAW of Carbon Steel plates of thickness less than 40 mm and more than 20 mm when using Ar He shielding gas.
<pre>preheating_req ("GMAW",Plate_Thickness,                 "CARBON STEEL","Ar He",                 Preheat,Interpassheat):-     Plate_Thickness &gt; 40,     Plate_Thickness &lt;= 65,     Preheat = "70 degree C",     Interpassheat = "110 degree C".</pre>	Preheat before first pass upto 70° C for GMAW of Carbon Steel plates of thickness more than 40 mm and less than 65 mm when using Ar He shielding gas. Perform Interpass heating to 110° C .
<pre>preheating_req ("GMAW",Plate_Thickness,                 "CARBON STEEL","Ar He",                 Preheat,Interpassheat):-     Plate_Thickness &gt; 65,     Preheat = "110 degree C",     Interpassheat = "150 degree C".</pre>	Preheat before first pass upto 70° C for GMAW of Carbon Steel plates of thickness more than 40 mm and less than 65 mm when using Ar He shielding gas. Perform Interpass heating to 110° C .
<pre>welded_from(Plate,"CARBON STEEL",Welded_from):-     Plate &lt;= 12,     Welded_from="One Side".</pre>	Carbon Steel Plates less than 12 mm can be welded from one side.
<pre>welded_from(Plate,"CARBON STEEL",Welded_from):-     Plate &gt; 12,     Welded_from="Both Sides".</pre>	Carbon Steel Plates more than 12 mm should be welded from both sides.
<pre>get_joint_design(Plate):-     Plate &lt;= 6,     JointDesign="No Edge Preparation".</pre>	No edge preparation is required for plates less than 6 mm
<pre>get_joint_design(Plate):-     Plate &lt; 12,     Plate &gt;= 6,     JointDesign="Single-Sided V Shape",     RootFace=round(Plate/6),     RootGap=round(Plate/6),     RootAngle=60.</pre>	Use V-shape joint preparation for plates less than 12 mm. Use Root Face and Root Gap of 1/6 <sup>th</sup> of the Plate thickness and a Root Angle of 60°.

Prolog Rule	Meaning in English
<pre> get_joint_design(Plate):-     Plate &gt; 12,     JointDesign="Double-Sided V Shape",     RootFace=round(Plate/6),     RootGap=round(Plate/6),     RootAngle=60. </pre>	Use Double-sided X-shape joint preparation for plates less than 12 mm. Use Root Face and Root Gap of 1/6 <sup>th</sup> of the Plate thickness and a Root Angle of 60°.
<pre> gas_cup_opening("GMAW","ALUMINUM ALLOY","Ar He",     "FLAT",Opening,Plate) :-     Plate &lt;= 13,     Opening = "1/2 inches". </pre>	Use Gas Cup Opening of 1/2" for GMAW of Aluminum Alloy Plates less than 13 mm
<pre> gas_cup_opening("GMAW","ALUMINUM ALLOY","Ar He",     "FLAT",Opening,Plate) :-     Plate &gt; 13,     Opening = "5/8 inches". </pre>	Use Gas Cup Opening of 5/8" for GMAW of Aluminum Alloy Plates less than 13 mm
<pre> welding_technique("GMAW",Plate,"ALUMINUM ALLOY",     "Ar He",Weld_Technique):-     Plate &lt; 5,     Weld_Technique="Use Copper Plate Backing". </pre>	Use Copper Plate Backing for GMAW of Aluminum Alloys of plate thickness less than 5 mm.
<pre> welding_technique ("GMAW",Plate,"CARBON STEEL",     "Ar He",Weld_Technique):-     Plate &gt; 5,     Weld_Technique="Side-to-side weaving". </pre>	Use side-to-side weaving for Carbon Steel Plates more than 5 mm thick when using Ar He shielding Gas in GMAW.

## **APPENDIX B - NEURO-EXPERT SYSTEM PSEUDO CODE**

### **B.1 Program Listing (Pseudo Code)**

#### **B.1.1 Process Selector**

1. Read following Job Variables from the data file:

Parent Material, Weld type, Position, Seam length, Plate thickness, Strength,  
Appearance, Deposition, Weld Quality, Automation, Leg length

2. Compute (and store) Criteria (Weight) for the above job variables from the Knowledge Base stored in the Expert System for the following:

- a) Automation criteria (Yes/No)
- b) Parent material criteria (Mild Steel, Carbon Steel etc.)
- c) Weld type criteria (Butt, Fillet, Lap etc.)
- d) Position criteria (Flat, Vertical, Horizontal, Downward, Overhead)
- e) Appearance criteria (Flat, Convex, Concave)
- f) Deposition criteria (High, Medium, Low)
- g) Weld quality criteria (Low, Medium, High)

Compute the qualitative value (Low, Medium, High etc.) using expert rules for the following quantitative job variables :

- h) Plate thickness criteria (mm)
- i) Strength criteria (psi)
- j) Seam length criteria (meters)

**k) Leg length criteria (mm)**

Compute (and store) Criteria using the qualitative values for the above job variables from the Knowledge Base stored in the Expert System.

3. Based on above criteria values determine weights for the four processes: SMAW, GMAW, GTAW and SAW. The process with the maximum weight is selected. Store the selected process in memory (and file).
4. Convert above qualitative job variables into numerical values. The neural network has previously been trained using job data available in handbooks. Use previously trained Neural Network (Back propagation method) to compute the output for the input job variables. Compute the highest output value and store the corresponding process as the selection by neural network.
5. Validate the choice made by neural network using expert rules for position, plate thickness, and automation as follows:
  - a) SAW can be used in Flat Position only.
  - b) SMAW cannot be used if Automation is required.
  - c) SAW cannot be used on plates less than 2 mm thick.
6. If the choices made by expert system and neural network are identical then that process is selected and no knowledge correction or neural network re-training is necessary. If the expert system and neural network differ then deadlock is resolved by using expert rules as follows:

A factor of confidence is calculated for the Expert System and the Neural Network and the higher factor determines the process as follows:

$E1 = \text{Total Criteria Weight}$ ,  $E2 = \text{Highest Weight attainable}$

Then, ***Expert System Confidence Factor =  $E1/E2$***

***Neural Network Confidence Factor = Activation for the Neural Network***

The process with the higher Confidence factor is selected.

7. If the expert system selection is used then the neural network is re-trained with the new output process by adding the job variable and the output to the training set and running the training cycle again.
8. If the neural network selection is used then the expert network knowledge is modified using the following procedure:

For each criteria, the difference between the criteria (expert knowledge) and the neural Weight for the corresponding connection is computed. If the absolute difference is greater than the criteria then the criteria weight is set to the neural weight otherwise the weight is kept the same.

### **B.1.2 Parameter Selection:**

1. Read the following Job Variables from the data file:

***Parent Material, Weld type, Position, Seam length, Plate thickness, Strength, Appearance, Deposition, Weld Quality, Automation, Leg length***

Also determine the process selected in the Process Selection Module.

2. Compute (and store) the following process parameters for the above job variables from the Knowledge Base (built from standard data) stored in the Expert System:

***electrode, weld type, leg length, position, runs, Arc Travel Speed, electrode size, current***

Convert any qualitative job variables into numerical values for the neural defect analyzer.

The neural network has previously been trained using job/parameter data available in handbooks. Use previously trained welding process specific Neural Network (Back propagation method) to compute the output defect activation for the input parameters (job variables/ parameters: ***Parent Material, Weld type, Position, Plate thickness (leg length for fillet), electrode, electrode size, Voltage, Current, Speed, Shielding Gas, Number of Runs etc.***). If all the output values are within 10% (adjustable) i.e. 0.1 then the parameters are applicable otherwise the computed parameters (runs, ***Arc Travel Speed***, electrode size, current) are adjusted using the logic given in pseudo-code in section A.1.3. The rule values are modified using the approach used for Process Selection Module.



### **B.1.3 Defect Analyzer:**

1. Determine the process selected in the Process Selection Module.
2. Query the following process parameters from the Working Knowledge Base:

***electrode, weld type, leg length, position, runs, Arc Travel Speed, electrode size, current, Shielding Gas (GMAW), Wire Feed Rate (GMAW)***

Convert any quantitative job variables into qualitative values for the neural defect analyzer. The neural network has previously been trained using parameter data available in handbooks.

3. Run the trained Neural Defect Analyzer with the adjustable parameters as input and determine the predicted defects.
4. Create a priority list of the adjustable parameters that cause these defects and a direction of adjustment for each parameter along with an applicable range of adjustment. Also mark the parameters as ***volatile*** or ***non-volatile***. i.e. If adjusting them results in change in applicable range of other parameters, then they are considered Volatile. For e.g. In SMAW, increasing Electrode Size increases the applicable current range and the Travel Speed. Also, changing Electrode Type changes the applicable current range and the Travel Speed.
5. Adjust the volatile parameters to the maximum in the direction that will minimize the defects. If none of the parameters are volatile, go to step 5.
6. Adjust non-volatile parameters in a direction that will minimize the defect. For e.g. If increasing Electrode Size then use the average settings for current, wire feed and arc

travel speed setting for that electrode size. Input the modified process parameters to the NN and get a prediction of expected defects and a list of parameters that cause them. If none of the parameters are adjustable then go to step 5 otherwise go to step 2.

7. Create weld procedure instructions for WPS Generator using the DEFECT ANALYZER Rules for Defect Avoidance.

## APPENDIX C - TRAINING AND TESTING DATA

### C.1 Training & Testing Data for Process Selector Neural Network in Spreadsheet

Job ID	Process	Parent Material	Weld Type	Position	Seam Length	Plate Thickness	Strength	Appearance	Deposition	Weld Quality	Automation	Leg Length
1	SAW	CARBON STEEL	BUTT	FLAT	HIGH	5	HIGH	FLAT	HIGH	LOW	YES	0
2	SAW	CARBON STEEL	BUTT	FLAT	HIGH	6	HIGH	FLAT	MEDIUM	LOW	YES	0
3	SAW	CARBON STEEL	BUTT	FLAT	HIGH	10	HIGH	FLAT	HIGH	MEDIUM	YES	0
4	SAW	CARBON STEEL	BUTT	FLAT	HIGH	13	HIGH	FLAT	LOW	MEDIUM	YES	0
5	SAW	CARBON STEEL	BUTT	FLAT	HIGH	19	HIGH	FLAT	HIGH	LOW	YES	0
6	SAW	CARBON STEEL	BUTT	FLAT	HIGH	25	HIGH	FLAT	MEDIUM	LOW	YES	0
7	SAW	CARBON STEEL	BUTT	FLAT	HIGH	32	HIGH	FLAT	HIGH	MEDIUM	YES	0
8	SAW	CARBON STEEL	BUTT	FLAT	HIGH	38	HIGH	FLAT	LOW	MEDIUM	YES	0
9	SAW	CARBON STEEL	FILLET	FLAT	HIGH	4	HIGH	FLAT	HIGH	LOW	YES	3
10	SAW	CARBON STEEL	FILLET	FLAT	HIGH	5	HIGH	FLAT	MEDIUM	LOW	YES	4
11	SAW	CARBON STEEL	FILLET	FLAT	HIGH	6	HIGH	FLAT	HIGH	MEDIUM	YES	5
12	SAW	CARBON STEEL	FILLET	FLAT	HIGH	8	HIGH	FLAT	LOW	MEDIUM	YES	6
13	SAW	CARBON STEEL	FILLET	FLAT	HIGH	10	HIGH	FLAT	HIGH	HIGH	YES	8
14	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	4	HIGH	FLAT	HIGH	LOW	YES	3
15	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	5	HIGH	FLAT	MEDIUM	LOW	YES	4
16	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	6	HIGH	FLAT	HIGH	MEDIUM	YES	5
17	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	8	HIGH	FLAT	LOW	MEDIUM	YES	6
18	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	10	HIGH	FLAT	HIGH	HIGH	YES	8
19	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	1	HIGH	FLAT	LOW	LOW	NO	0
20	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	2	HIGH	FLAT	LOW	LOW	NO	0
21	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	4	HIGH	FLAT	MEDIUM	LOW	NO	0
22	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	5	HIGH	FLAT	MEDIUM	LOW	NO	0
23	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	6	HIGH	FLAT	HIGH	LOW	NO	0
24	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	10	HIGH	FLAT	HIGH	HIGH	NO	0
25	SMAW	STAINLESS STEEL	BUTT	FLAT	LOW	13	HIGH	FLAT	HIGH	HIGH	NO	0
26	SMAW	STAINLESS STEEL	BUTT	OVERHEAD	LOW	2	HIGH	FLAT	HIGH	LOW	NO	0
27	SMAW	STAINLESS STEEL	BUTT	OVERHEAD	LOW	4	HIGH	FLAT	HIGH	LOW	NO	0
28	SMAW	STAINLESS STEEL	BUTT	OVERHEAD	LOW	5	HIGH	FLAT	HIGH	LOW	NO	0
29	SMAW	STAINLESS STEEL	BUTT	OVERHEAD	LOW	6	HIGH	FLAT	HIGH	LOW	NO	0
30	SMAW	STAINLESS STEEL	BUTT	VERTICAL	LOW	2	HIGH	FLAT	HIGH	LOW	NO	0
31	SMAW	STAINLESS STEEL	BUTT	VERTICAL	LOW	4	HIGH	FLAT	HIGH	LOW	NO	0
32	SMAW	STAINLESS STEEL	BUTT	VERTICAL	LOW	5	HIGH	FLAT	HIGH	LOW	NO	0
33	SMAW	STAINLESS STEEL	BUTT	VERTICAL	LOW	6	HIGH	FLAT	HIGH	LOW	NO	0
34	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	2	HIGH	FLAT	HIGH	LOW	NO	2
35	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	3	HIGH	FLAT	HIGH	LOW	NO	3
36	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	5	HIGH	FLAT	HIGH	LOW	NO	5
37	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	6	HIGH	FLAT	HIGH	LOW	NO	6
38	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	8	HIGH	FLAT	HIGH	LOW	NO	8
39	SMAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	2	HIGH	FLAT	HIGH	LOW	NO	2
40	SMAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	3	HIGH	FLAT	HIGH	LOW	NO	3
41	SMAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	5	HIGH	FLAT	HIGH	LOW	NO	5
42	SMAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	6	HIGH	FLAT	HIGH	LOW	NO	6
43	SMAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	8	HIGH	FLAT	HIGH	LOW	NO	8
44	GMAW	STAINLESS STEEL	BUTT	FLAT	HIGH	3	HIGH	FLAT	HIGH	HIGH	YES	0
45	GMAW	STAINLESS STEEL	BUTT	FLAT	LOW	6	HIGH	CONVEX	LOW	LOW	YES	0
46	GMAW	STAINLESS STEEL	BUTT	FLAT	HIGH	10	HIGH	CONCAVE	MEDIUM	MEDIUM	YES	0
47	GMAW	STAINLESS STEEL	BUTT	FLAT	LOW	13	HIGH	FLAT	LOW	LOW	YES	0
48	GMAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	3	HIGH	FLAT	HIGH	HIGH	YES	0

Job ID	Process	Parent Material	Weld Type	Position	Seam Length	Plate Thickness	Strength	Appearance	Deposition	Weld Quality	Automation	Leg Length
49	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	5	HIGH	FLAT	HIGH	HIGH	YES	0
50	GMAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	6	HIGH	FLAT	HIGH	HIGH	YES	0
51	GMAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	10	HIGH	FLAT	HIGH	HIGH	YES	0
52	GMAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	13	HIGH	FLAT	HIGH	HIGH	YES	0
53	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	16	HIGH	FLAT	HIGH	HIGH	YES	0
54	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	19	HIGH	FLAT	HIGH	HIGH	YES	0
55	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	25	HIGH	FLAT	HIGH	HIGH	YES	0
56	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	32	HIGH	FLAT	HIGH	HIGH	YES	0
57	GMAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	38	HIGH	FLAT	HIGH	HIGH	YES	0
58	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	3	HIGH	FLAT	HIGH	HIGH	YES	0
59	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	HIGH	5	HIGH	FLAT	HIGH	HIGH	NO	0
60	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	6	HIGH	FLAT	HIGH	HIGH	YES	0
61	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	HIGH	10	HIGH	FLAT	HIGH	HIGH	NO	0
62	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	HIGH	13	HIGH	FLAT	HIGH	HIGH	YES	0
63	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	19	HIGH	FLAT	HIGH	HIGH	NO	0
64	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	25	HIGH	FLAT	HIGH	HIGH	YES	0
65	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	LOW	3	HIGH	FLAT	HIGH	HIGH	YES	0
66	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	HIGH	5	HIGH	FLAT	HIGH	HIGH	NO	0
67	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	LOW	6	HIGH	FLAT	HIGH	HIGH	YES	0
68	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	HIGH	10	HIGH	FLAT	HIGH	HIGH	NO	0
69	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	HIGH	13	HIGH	FLAT	HIGH	HIGH	YES	0
70	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	LOW	19	HIGH	FLAT	HIGH	HIGH	NO	0
71	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	LOW	25	HIGH	FLAT	HIGH	HIGH	YES	0
72	GMAW	ALUMINUM ALLOY	FILLET	HORIZONTAL	LOW	10	HIGH	FLAT	HIGH	HIGH	NO	6
73	GMAW	ALUMINUM ALLOY	FILLET	HORIZONTAL	HIGH	16	HIGH	FLAT	HIGH	HIGH	NO	10
74	GMAW	ALUMINUM ALLOY	FILLET	HORIZONTAL	HIGH	19	HIGH	FLAT	HIGH	HIGH	NO	13
75	GMAW	ALUMINUM ALLOY	FILLET	VERTICAL	LOW	10	HIGH	FLAT	HIGH	HIGH	NO	6
76	GMAW	ALUMINUM ALLOY	FILLET	VERTICAL	HIGH	16	HIGH	FLAT	HIGH	HIGH	NO	10
77	GMAW	ALUMINUM ALLOY	FILLET	VERTICAL	HIGH	19	HIGH	FLAT	HIGH	HIGH	NO	13
78	GMAW	ALUMINUM ALLOY	FILLET	OVERHEAD	LOW	10	HIGH	FLAT	HIGH	HIGH	NO	6
79	GMAW	ALUMINUM ALLOY	FILLET	OVERHEAD	HIGH	16	HIGH	FLAT	HIGH	HIGH	NO	10
80	GMAW	ALUMINUM ALLOY	FILLET	OVERHEAD	HIGH	19	HIGH	FLAT	HIGH	HIGH	NO	13
81	GTAW	STAINLESS STEEL	BUTT	FLAT	LOW	1.6	HIGH	FLAT	HIGH	HIGH	NO	0
82	GTAW	STAINLESS STEEL	BUTT	HORIZONTAL	LOW	2.4	HIGH	CONVEX	HIGH	HIGH	NO	0
83	GTAW	STAINLESS STEEL	BUTT	FLAT	LOW	3.2	HIGH	CONVEX	HIGH	HIGH	NO	0
84	GTAW	STAINLESS STEEL	BUTT	HORIZONTAL	LOW	4.8	HIGH	CONCAVE	HIGH	HIGH	NO	0
85	GTAW	STAINLESS STEEL	FILLET	FLAT	LOW	1.6	HIGH	FLAT	HIGH	HIGH	NO	0
86	GTAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	2.4	HIGH	CONVEX	HIGH	HIGH	NO	0
87	GTAW	STAINLESS STEEL	FILLET	FLAT	LOW	3.2	HIGH	CONVEX	HIGH	HIGH	NO	0
88	GTAW	STAINLESS STEEL	FILLET	HORIZONTAL	LOW	4.8	HIGH	CONCAVE	HIGH	HIGH	NO	0
89	GTAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	1.6	HIGH	FLAT	HIGH	HIGH	NO	0
90	GTAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	3.2	HIGH	CONVEX	HIGH	HIGH	NO	0
91	GTAW	ALUMINUM ALLOY	BUTT	FLAT	LOW	6.4	HIGH	CONVEX	HIGH	HIGH	NO	0
92	GTAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	9.5	HIGH	CONCAVE	HIGH	HIGH	NO	0
93	GTAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	13	HIGH	CONCAVE	HIGH	HIGH	NO	0
94	GTAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	25	HIGH	CONCAVE	HIGH	HIGH	NO	0
95	GTAW	CARBON STEEL	BUTT	FLAT	HIGH	5	HIGH	FLAT	LOW	HIGH	YES	0
96	SAW	CARBON STEEL	FILLET	HORIZONTAL	HIGH	10	HIGH	FLAT	HIGH	HIGH	YES	10
97	SMAW	STAINLESS STEEL	BUTT	OVERHEAD	LOW	6	HIGH	FLAT	LOW	HIGH	NO	0
98	SMAW	STAINLESS STEEL	BUTT	VERTICAL	LOW	4	HIGH	FLAT	LOW	LOW	NO	0
99	SMAW	STAINLESS STEEL	FILLET	FLAT	LOW	8	HIGH	FLAT	HIGH	LOW	NO	8
100	GMAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	3	HIGH	FLAT	HIGH	HIGH	YES	0
101	SAW	ALUMINUM ALLOY	BUTT	FLAT	HIGH	13	LOW	FLAT	HIGH	HIGH	YES	0
102	GMAW	ALUMINUM ALLOY	BUTT	HORIZONTAL	LOW	13	HIGH	FLAT	HIGH	HIGH	NO	0
103	GTAW	ALUMINUM ALLOY	BUTT	VERTICAL	HIGH	6	LOW	FLAT	LOW	MEDIUM	NO	0
104	GMAW	ALUMINUM ALLOY	BUTT	VERTICAL	LOW	25	HIGH	FLAT	HIGH	MEDIUM	NO	0
105	GMAW	ALUMINUM ALLOY	FILLET	OVERHEAD	HIGH	19	HIGH	FLAT	HIGH	MEDIUM	NO	19
106	GMAW	STAINLESS STEEL	FILLET	HORIZONTAL	HIGH	3	HIGH	FLAT	LOW	HIGH	NO	3
107	SMAW	CARBON STEEL	BUTT	FLAT	HIGH	5	HIGH	FLAT	LOW	HIGH	YES	0
108	SMAW	CARBON STEEL	BUTT	HORIZONTAL	HIGH	13	HIGH	FLAT	HIGH	HIGH	YES	0

## C.2 Weight Change comparison for pruned and unpruned networks for Process Selector

		First Set											
		Initial	SMAW Unpruned	SMAW Pruned	Initial	GMAW Unpruned	GMAW Pruned	Initial	GTAW Unpruned	GTAW Pruned	Initial	SAW Unpruned	SAW Pruned
Parent Material	Aluminum Alloys	-1	-1.72	-1.12	1	1.36	0.80	1	1.51	1.73	1	0.04	0.34
	Carbon Steel	0	-0.40	-0.51	0	-1.36	-0.85	0	-0.24	0.10	0	2.20	0.98
	Stainless Steel	0	0.56	0.49	0	0.20	0.23	0	-0.14	-0.33	0	-0.98	-0.71
Weld Type	Butt	0	-0.71	-0.38	0	-0.04	-0.32	0	0.42	0.96	0	-0.93	-0.89
	Fillet	0	0.15	0.24	0	-0.76	-0.49	0	-0.29	-0.46	0	1.19	0.50
Position	Horizontal	0	-0.34	0.03	0	0.08	0.38	0	-0.25	-0.02	0	0.10	0.11
	Vertical	1	0.97	1.08	1	1.11	1.03	1	0.92	0.93	-1	-0.98	-1.07
	Downward	1	1	1.00	1	1	1.00	1	1	1.00	-1	-1	-1.00
	Overhead	1	0.66	0.47	1	0.60	1.08	1	1.77	1.65	-1	-1.00	-1.01
Seam Length	High	-1	-1.56	-1.14	1	0.20	0.19	1	1.13	1.50	1	1.26	0.61
Plate Thickness	High	-1	-1.53	-1.16	1	0.75	0.58	-1	0.70	1.04	1	1.59	1.34
	Medium	0	1.11	0.58	0	0.75	0.27	0	0.95	-0.44	0	0.75	-0.54
	Low	1	0.86	0.44	1	0.70	0.34	1	1.48	1.89	-1	-1.08	-1.19
Strength	High	0	-0.56	-0.14	0	0.80	0.81	0	0.13	0.50	0	0.26	0.39
Deposition	High	-1	-0.91	-0.34	1	0.02	0.10	-1	-0.94	-1.20	1	0.88	0.62
	Medium	0	-0.09	-0.04	0	0.20	0.02	0	-0.38	-0.36	0	0.45	0.37
Quality	Medium	0	-0.19	-0.07	0	-0.11	-0.17	0	-0.77	-0.12	0	1.07	0.27
	High	-1	-1.82	-1.70	1	0.46	0.13	-1	2.25	2.76	1	1.05	1.08
Automation	Yes	-1	-1.48	-1.29	1	1.68	1.46	1	-0.84	0.23	1	2.65	1.38
	No	1	0.91	1.15	1	-0.48	-0.27	1	2.97	2.26	-1	-2.39	-1.77
Leg Length	Low	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00	0	0.00	0.00
	High	0	-0.56	-0.14	1	0.20	0.19	0	0.13	0.50	1	1.26	0.61
Hidden Layer	SMAW	1	2.81	2.12	0	-2.00	-1.32	0	-2.41	-1.86	0	-1.41	-0.63
	GMAW	0	-0.36	-0.25	1	2.83	2.32	0	-2.35	-1.71	0	-1.80	-1.76
	GTAW	0	-0.53	-0.63	0	-0.46	-0.57	1	1.57	0.95	0	-1.61	-0.54
	SAW	0	-0.57	-0.52	0	-0.26	-0.23	0	-0.96	-1.75	1	1.48	1.88

Job Variable	Value	Second Set											
		Initial	SMAW Unpruned	SMAW Pruned	Initial	GMAW Unpruned	GMAW Pruned	Initial	GTAW Unpruned	GTAW Pruned	Initial	SAW Unpruned	SAW Pruned
Parent Material	Aluminum Alloys	-1	-1.58	-1.94	1	1.42	1.97	1	0.75	0.34	1	-0.49	0.27
	Carbon Steel	0	-0.72	-0.05	0	0.17	-0.48	0	-0.10	0.06	0	2.25	1.22
	Stainless Steel	0	0.68	0.55	0	-0.26	-0.26	0	-0.12	0.04	0	-1.29	-1.04
Weld Type	Butt	0	-1.03	-1.13	0	-0.11	0.59	0	0.11	0.00	0	-1.87	-1.04
	Fillet	0	0.41	0.69	0	0.43	-0.35	0	-0.59	-0.56	0	1.35	0.49
Position	Horizontal	0	-0.34	-0.29	0	0.21	0.15	0	-0.34	-0.35	0	-0.43	-0.45
	Vertical	1	1.15	1.42	1	1.30	0.93	1	0.19	0.42	-1	-1.17	-0.86
	Downward	1	1	1.00	1	1	1.00	1	1	1.00	-1	-1	-1.00
	Overhead	1	1.35	0.89	1	1.10	1.04	1	0.83	1.11	-1	-1.14	-0.98
Seam Length	High	-1	-1.62	-1.44	1	1.32	1.23	1	0.53	0.44	1	0.47	0.45
Plate Thickness	High	-1	-1.03	-1.26	1	1.80	1.53	-1	0.36	0.59	1	1.40	0.71
	Medium	0	0.68	0.44	0	-0.30	-0.24	0	0.11	0.10	0	-0.78	-0.60
	Low	1	-0.27	0.38	1	0.82	0.94	1	1.06	0.74	-1	-1.14	-0.66
Strength	High	0	-0.62	-0.44	0	0.32	0.23	0	-0.47	-0.56	0	-0.53	-0.55
Deposition	High	-1	-0.38	-0.33	1	0.92	0.80	-1	-1.52	-1.72	1	-0.30	-0.12
	Medium	0	-0.26	-0.04	0	0.08	-0.06	0	-0.36	-0.37	0	0.87	0.85
Quality	Medium	0	-0.15	0.05	0	0.58	0.02	0	-0.72	-0.30	0	0.79	0.45
	High	-1	-1.90	-2.22	1	0.75	1.46	-1	0.27	0.18	1	1.47	-1.43
Automation	Yes	-1	-1.68	-1.75	1	1.36	1.20	1	0.50	0.96	1	1.87	1.09
	No	1	1.06	1.31	1	0.96	1.03	1	1.03	0.47	-1	-2.39	-1.64
Leg Length	Low	0	0.06	0.01	0	0.33	0.03	0	0.05	-0.06	0	-0.02	0.04
	High	0	-0.68	-0.46	1	0.99	1.20	0	-0.52	-0.51	1	0.49	0.41
Hidden Layer	SMAW	1	2.33	1.99	0	-1.88	-1.88	0	-0.15	0.23	0	-0.37	-1.08
	GMAW	0	-1.12	-0.78	1	0.85	0.93	0	-0.64	-0.49	0	-0.22	-0.58
	GTAW	0	-2.00	-1.80	0	-1.86	-1.55	1	2.44	2.06	0	-1.51	-1.94
	SAW	0	-1.08	-0.24	0	-2.06	-1.68	0	0.60	-0.60	1	1.37	1.91

Job Variable	Value	Third Set											
		Initial	SMAW Unpruned	SMAW Pruned	Initial	GMAW Unpruned	GMAW Pruned	Initial	GTAW Unpruned	GTAW Pruned	Initial	SAW Unpruned	SAW Pruned
Parent Material	Aluminum Alloys	-1	-1.80	-1.49	1	1.07	3.33	1	0.62	-0.28	1	0.77	0.35
	Carbon Steel	0	0.10	-2.68	0	0.27	-2.29	0	-0.88	0.04	0	1.01	7.76
	Stainless Steel	0	0.63	2.23	0	-0.79	-0.70	0	1.00	1.78	0	-0.75	-9.91
Weld Type	Butt	0	-0.33	-1.00	0	-0.62	-0.18	0	-0.48	0.49	0	0.31	-1.83
	Fillet	0	0.27	0.06	0	0.16	-0.49	0	0.23	0.06	0	-0.27	-0.97
Position	Horizontal	0	0.18	-0.03	0	-0.49	1.30	0	0.14	0.59	0	-0.43	-2.64
	Vertical	1	0.34	0.62	1	1.45	2.25	1	0.73	0.10	-1	-0.93	-1.05
	Downward	1	1	1.00	1	1	1.00	1	1	1.00	-1	-1	-1.00
	Overhead	1	1.38	1.07	1	0.60	0.97	1	0.81	0.93	-1	-0.91	-1.00
Seam Length	High	-1	-1.07	-1.93	1	0.55	0.33	1	0.75	1.55	1	1.03	-1.80
Plate Thickness	High	-1	-0.96	-0.99	1	1.01	1.25	-1	0.87	-0.63	1	1.12	4.77
	Medium	0	0.33	0.69	0	0.75	0.88	0	0.30	-0.30	0	-0.52	-2.35
	Low	1	0.56	-0.63	1	1.07	-0.80	1	0.57	3.48	-1	-0.57	-5.22
Strength	High	0	-0.07	-0.93	0	0.32	-0.67	0	-0.25	0.55	0	0.03	-2.80
Deposition	High	-1	-0.61	-1.02	1	0.92	-0.94	-1	-0.64	-2.40	1	0.45	3.43
	Medium	0	-0.01	0.12	0	0.18	-0.51	0	-0.62	-1.37	0	0.72	2.56
Quality	Medium	0	-0.09	-0.50	0	0.24	0.17	0	-0.31	-1.85	0	0.17	4.62
	High	-1	-1.26	-6.53	1	0.09	2.52	-1	0.73	14.02	1	0.68	-15.05
Automation	Yes	-1	-1.26	-4.04	1	1.02	1.55	1	0.48	0.35	1	1.64	4.53
	No	1	1.19	3.10	1	0.53	-0.22	1	1.27	2.20	-1	-1.61	-7.33
Leg Length	Low	0	-0.02	0.00	0	-0.02	0.02	0	0.01	0.00	0	0.00	-0.02
	High	0	-0.05	-0.93	1	0.56	0.31	0	-0.27	0.55	1	1.03	-1.78
Hidden Layer	SMAW	1	1.99	4.14	0	-1.94	-1.90	0	-0.34	-4.28	0	-0.45	-1.69
	GMAW	0	-1.60	-0.95	1	-0.22	3.16	0	0.48	-1.52	0	-0.04	-2.65
	GTAW	0	-0.44	-0.49	0	1.82	0.21	1	0.79	1.21	0	-2.17	-1.27
	SAW	0	-0.08	-0.70	0	1.27	0.41	0	-2.51	-4.75	1	1.10	3.68

Job Variable	Value	Fourth Set											
		Initial	SMAW Unpruned	SMAW Pruned	Initial	GMAW Unpruned	GMAW Pruned	Initial	GTAW Unpruned	GTAW Pruned	Initial	SAW Unpruned	SAW Pruned
Parent Material	Aluminum Alloys	-1	-1.35	-1.50	1	0.52	2.78	1	2.44	0.68	1	0.17	0.05
	Carbon Steel	0	0.00	-0.26	0	-1.31	-1.35	0	-0.71	-2.13	0	2.40	3.57
	Stainless Steel	0	0.04	0.58	0	1.14	-0.91	0	-0.38	2.74	0	-1.65	-2.23
Weld Type	Butt	0	-0.16	-0.21	0	-0.02	-0.20	0	0.03	0.77	0	-0.55	-0.47
	Fillet	0	-0.16	0.03	0	-0.63	-0.28	0	0.32	-0.48	0	0.47	0.86
Position	Horizontal	0	-0.39	-0.49	0	-0.35	-0.40	0	0.47	1.44	0	0.28	0.02
	Vertical	1	0.85	0.63	1	1.40	3.24	1	0.73	-1.35	-1	-1.08	-1.32
	Downward	1	1	1.00	1	1	1.00	1	1	1.00	-1	-1	-1.00
	Overhead	1	1.05	1.25	1	1.21	1.03	1	0.87	0.79	-1	-1.20	-1.09
Seam	High	-1	-1.32	-1.18	1	0.35	0.52	1	1.36	1.29	1	0.92	1.39
Plate Thickness	High	-1	-1.14	-1.19	1	0.95	1.58	-1	0.51	-0.77	1	1.57	2.15
	Medium	0	0.38	0.68	0	0.16	0.81	0	-0.29	-1.86	0	-0.63	-0.45
	Low	1	0.44	0.33	1	0.24	-0.86	1	2.13	4.93	-1	-1.02	-1.31
Strength	High	0	-0.32	-0.18	0	-0.65	-0.48	0	0.36	0.29	0	-0.08	0.39
Deposition	High	-1	-0.91	-0.69	1	0.00	-1.26	-1	0.00	1.27	1	0.20	1.34
	Medium	0	0.06	0.06	0	-0.10	-0.04	0	-0.15	-0.12	0	0.28	0.24
Quality	Medium	0	-0.20	-0.46	0	-0.27	1.43	0	-0.40	-1.39	0	0.97	0.14
	High	-1	-1.20	-1.30	1	0.06	-0.87	-1	1.60	4.22	1	1.23	-1.06
Automation	Yes	-1	-1.40	-1.39	1	1.11	1.10	1	0.00	-1.22	1	1.80	3.10
	No	1	1.08	1.21	1	0.24	0.43	1	2.36	3.51	-1	-1.88	-2.70
Leg Length	Low	0	0.01	0.03	0	0.20	0.32	0	0.08	-0.40	0	-0.33	0.05
	High	0	-0.33	-0.21	1	0.15	0.21	0	0.27	0.69	1	1.25	1.34
Hidden Layer	SMAW	1	2.55	2.47	0	-1.99	-2.28	0	-2.01	-2.23	0	-0.88	-1.34
	GMAW	0	-0.35	-0.40	1	2.49	2.08	0	-2.03	-0.90	0	-2.00	-1.83
	GTAW	0	-0.65	-0.47	0	-0.61	1.32	1	0.92	1.81	0	-0.70	-1.99
	SAW	0	-0.52	-0.58	0	0.04	-0.42	0	-1.91	-1.71	1	1.85	2.46



### C.3 Training Data for Defect Analyzer Data

Low H2 Electrode	Basic Electrode	Rutile Electrode	Cellulosic Electrode	Electrode Size			Current			Arc Blow	Incomplete Penetration	Porosity	Undercut	Slag Inclusions
				low	med.	high	low	med	high					
1	0	0	0	0	0	1	1	0	0	0	1	0	0	1
1	0	0	0	0	0	1	0	1	0	0	0	0	0	1
1	0	0	0	0	0	1	0	0	1	1	0	1	1	0
1	0	0	0	0	1	0	1	0	0	0	1	0	0	1
1	0	0	0	0	1	0	0	1	0	1	0	0	0	1
1	0	0	0	0	1	0	0	0	1	1	0	1	1	0
1	0	0	0	1	0	0	1	0	0	0	1	0	0	1
1	0	0	0	1	0	0	0	1	0	1	1	1	1	1
1	0	0	0	1	0	0	0	0	1	1	0	1	1	0
0	1	0	0	0	0	1	1	0	0	0	1	0	0	1
0	1	0	0	0	0	1	0	1	0	1	1	0	0	1
0	1	0	0	0	0	1	0	0	1	1	0	1	1	0
0	1	0	0	0	1	0	1	0	0	0	1	0	0	1
0	1	0	0	0	1	0	0	0	1	1	0	0	0	1
0	1	0	0	0	1	0	0	0	1	1	0	1	1	0
0	1	0	0	1	0	0	1	0	0	0	1	1	0	1
0	1	0	0	1	0	0	0	1	0	1	1	0	1	1
0	1	0	0	1	0	0	0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	1	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	1	1	0	1	1	0
0	0	1	0	0	1	0	1	0	0	0	1	0	0	0
0	0	1	0	0	1	0	0	0	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	1	1	0	1	1	0
0	0	1	0	1	0	0	1	0	0	0	1	0	0	0
0	0	1	0	1	0	0	0	1	0	1	1	1	1	0
0	0	1	0	1	0	0	0	0	1	1	0	1	1	0
0	0	0	1	0	0	1	1	0	0	0	1	0	0	1
0	0	0	1	0	0	1	0	1	0	1	1	0	0	1
0	0	0	1	0	0	1	0	0	1	1	0	1	1	1
0	0	0	1	0	1	0	1	0	0	0	1	0	0	1
0	0	0	1	0	1	0	0	1	0	1	0	0	0	1
0	0	0	1	0	1	0	0	0	1	1	0	1	1	1
0	0	0	1	1	0	0	1	0	0	0	1	1	0	1
0	0	0	1	1	0	0	0	1	0	1	1	0	1	1
0	0	0	1	1	0	0	0	0	1	1	0	0	1	1

## APPENDIX D - Listing of Knowledge bases in Prolog

### D.1 CRITERIA.DBA file for storing criteria of Process Selector

```
automation_criteria("NO",1,1,1,-1)
automation_criteria("YES",-1,1,1,1)
plate_thickness_criteria("LOW",1,1,1,-1)
plate_thickness_criteria("HIGH",-1,1,-1,1)
plate_thickness_criteria("MEDIUM",1.141355,0.930733,0.939914,1.024172)
strength_criteria("LOW",0,0,0,0)
strength_criteria("HIGH",0,0.249589,0,-0.236336)
seam_length_criteria("LOW",0,0,0,0)
seam_length_criteria("HIGH",-10,1,1,1)
leg_length_criteria("HIGH",0,0,0,0)
leg_length_criteria("LOW",0,0.509483,-0.40324,-0.301041)
parent_material_criteria("MILD STEEL",0,0,0,0)
parent_material_criteria("CARBON STEEL",0,0,0,0)
parent_material_criteria("CAST IRON",-1,1,1,1)
parent_material_criteria("STAINLESS STEEL",0,0,0,0)
parent_material_criteria("ALUMINUM ALLOY",-1,1,1,1)
weld_type_criteria("LAP",0,0,0,0)
weld_type_criteria("FILLET",0,0,0,0)
weld_type_criteria("BUTT",0,0.230419,0,-0.449493)
position_criteria("HORIZONTAL",0,0,0,0)
position_criteria("VERTICAL",1,1,1,-1)
position_criteria("DOWNWARD",1,1,1,-1)
position_criteria("OVERHEAD",1,1,1,-1)
position_criteria("FLAT",0.215006,0,0,0)
appearance_criteria("CONCAVE",0,0,0,0)
appearance_criteria("CONVEX",0,0,0,0)
appearance_criteria("FLAT",0,0,0,0)
deposition_criteria("MEDIUM",0,0,0,0)
deposition_criteria("HIGH",0,0,0,0)
deposition_criteria("LOW",-1.004342,1.50948,-1.40324,0.698959)
weld_quality_criteria("MEDIUM",0,0,0,0)
weld_quality_criteria("LOW",0,0,0,0)
weld_quality_criteria("HIGH",0,0.509483,-0.40324,-0.301041)
```

### D.2 ELECTROD.DBA file for storing electrode selection rules of Parameter Selector

```
electrode("E6010","CARBON STEEL","ALL","ALL","LOW","FLAT")
electrode("E6011","CARBON STEEL","ALL","ALL","MEDIUM","FLAT")
electrode("E6012","CARBON STEEL","ALL","ALL","LOW","CONVEX")
electrode("E6013","CARBON STEEL","ALL","ALL","HIGH","FLAT")
electrode("E6015","CARBON STEEL","ALL","ALL","LOW","FLAT/CONVEX")
electrode("E6020","CARBON STEEL","FH","ALL","LOW","FLAT/CONCAVE")
electrode("E6027","CARBON STEEL","FH","ALL","LOW","FLAT")
electrode("E7010","CARBON STEEL","ALL","ALL","MEDIUM","FLAT/CONCAVE")
electrode("E7014","CARBON STEEL","ALL","ALL","MEDIUM","FLAT")
electrode("E7013","CARBON STEEL","ALL","ALL","MEDIUM","FLAT")
electrode("E7016","CARBON STEEL","ALL","ALL","MEDIUM","FLAT")
```

```

electrode("E7018","CARBON STEEL","ALL","ALL","MEDIUM","FLAT")
electrode("E8016","CARBON STEEL","ALL","ALL","HIGH","FLAT")
electrode("E8018G","CARBON STEEL","ALL","ALL","HIGH","FLAT")
electrode("E308-17","STAINLESS STEEL","ALL","ALL","HIGH","FLAT")
electrode("E309-17","STAINLESS STEEL","ALL","ALL","HIGH","FLAT")
electrode("E316-17","STAINLESS STEEL","ALL","ALL","HIGH","FLAT")
electrode("E310-16","STAINLESS STEEL","ALL","ALL","HIGH","FLAT/CONVEX")
electrode("E308-17","STAINLESS STEEL","ALL","ALL","LOW","FLAT")
electrode("E309-17","STAINLESS STEEL","ALL","ALL","LOW","FLAT")
electrode("E316-17","STAINLESS STEEL","ALL","ALL","LOW","FLAT")
electrode("E310-16","STAINLESS STEEL","ALL","ALL","LOW","FLAT/CONVEX")
electrode("E4043","ALUMINUM ALLOY","ALL","FLAT","LOW","FLAT/CONVEX")
electrode("E4043","ALUMINUM ALLOY","ALL","ALL","HIGH","FLAT/CONVEX")

```

### **D.3 FIRMDAT.DBA file for storing cost data for Economic Evaluator**

```

firm_machine_data("GMAW",10)
firm_machine_data("SMAW",6)
firm_labor_data(15,15)
elect_dat("E6010",1.3)
elect_dat("E6012",1.1)
elect_dat("E6013",1.2)
elect_dat("E7018",1.2)
elect_dat("E7024",1.2)
elect_dat("E4043",1.2)
elect_dat("E308-17",1.2)
GMAW_elect_dat("STAINLESS STEEL",0.8,0.1)
GMAW_elect_dat("CARBON STEEL",0.8,0.1)
GMAW_elect_dat("ALUMINUM ALLOY",0.8,0.1)
GMAW_elect_dat("STAINLESS STEEL",1,0.1)
GMAW_elect_dat("CARBON STEEL",1,0.1)
GMAW_elect_dat("ALUMINUM ALLOY",1,0.25)
GMAW_elect_dat("STAINLESS STEEL",2,1)
GMAW_elect_dat("CARBON STEEL",2,1)
GMAW_elect_dat("ALUMINUM ALLOY",2,1)
GMAW_elect_dat("STAINLESS STEEL",3,1)
GMAW_elect_dat("CARBON STEEL",3,1)
GMAW_elect_dat("ALUMINUM ALLOY",3,1)
GMAW_elect_dat("STAINLESS STEEL",4,1)
GMAW_elect_dat("CARBON STEEL",4,1)
GMAW_elect_dat("ALUMINUM ALLOY",4,1)
GMAW_elect_dat("STAINLESS STEEL",5,1)
GMAW_elect_dat("CARBON STEEL",5,1)
GMAW_elect_dat("ALUMINUM ALLOY",5,1)

```

#### D.4 STD\_DATA.DBA file for storing Standard Data for Parameter Selector

```
SMAW_data("FILLET",5,"FLAT",1,4,120,4.2,0.153,6.7,1.61,"E6010")
SMAW_data("FILLET",5,"FLAT",1,5,160,2.7,0.153,4.81,0.85,"E6010")
SMAW_data("FILLET",5,"HORIZONTAL",1,4,110,4.2,0.153,7.43,1.61,"E6010")
SMAW_data("FILLET",5,"HORIZONTAL",1,5,160,2.7,0.153,4.81,0.85,"E6010")
SMAW_data("FILLET",5,"VERTICAL",1,3,25,85,7.7,0.185,12.26,3.33,"E6010")
SMAW_data("FILLET",5,"VERTICAL",1,4,110,5.1,0.185,8.93,2.04,"E6010")
SMAW_data("FILLET",5,"DOWNWARD",1,4,160,3.4,0.123,3.86,1.2,"E6010")
SMAW_data("FILLET",5,"DOWNWARD",1,5,190,2.2,0.123,3.19,0.59,"E6010")
SMAW_data("FILLET",6,"FLAT",1,5,160,2.6,0.208,6.54,1.34,"E6010")
SMAW_data("FILLET",6,"FLAT",1,6,3,230,2.3,0.208,4.38,0.66,"E6010")
SMAW_data("FILLET",6,"HORIZONTAL",1,5,160,2.6,0.208,6.54,1.34,"E6010")
SMAW_data("FILLET",6,"HORIZONTAL",1,6,3,220,2.3,0.208,4.6,0.66,"E6010")
SMAW_data("FILLET",6,"VERTICAL",1,5,85,9.8,0.237,15.75,4.42,"E6010")
SMAW_data("FILLET",6,"VERTICAL",1,6,3,110,6.5,0.237,11.48,2.77,"E6010")
SMAW_data("FILLET",6,"DOWNWARD",1,3,25,190,3.0,0.170,4.39,1,"E6010")
SMAW_data("FILLET",6,"DOWNWARD",1,4,255,1.9,0.170,3.19,0.44,"E6010")
SMAW_data("FILLET",8,"FLAT",1,5,160,6.0,0.342,10.73,2.52,"E6010")
SMAW_data("FILLET",8,"FLAT",1,6,3,230,3.8,0.342,7.19,1.4,"E6010")
SMAW_data("FILLET",8,"HORIZONTAL",1,4,110,9.4,0.342,16.56,4.22,"E6010")
SMAW_data("FILLET",8,"HORIZONTAL",2,4,110,9.4,0.342,16.56,4.22,"E6010")
SMAW_data("FILLET",8,"HORIZONTAL",1,5,160,6.0,0.342,10.73,2.52,"E6010")
SMAW_data("FILLET",8,"HORIZONTAL",2,5,160,6.0,0.342,10.73,2.52,"E6010")
SMAW_data("FILLET",8,"VERTICAL",1,3,25,85,15.1,0.364,24.18,7.05,"E6010")
SMAW_data("FILLET",8,"VERTICAL",1,4,110,10.0,0.364,17.61,4.52,"E6010")
SMAW_data("FILLET",10,"FLAT",1,5,160,9.0,0.508,15.92,3.98,"E6010")
SMAW_data("FILLET",10,"FLAT",1,6,3,230,5.7,0.508,10.67,2.32,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",1,4,110,14.0,0.508,24.57,6.5,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",2,4,110,14.0,0.508,24.57,6.5,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",3,4,110,14.0,0.508,24.57,6.5,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",1,5,160,9.0,0.508,15.92,3.98,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",2,5,160,9.0,0.508,15.92,3.98,"E6010")
SMAW_data("FILLET",10,"HORIZONTAL",3,5,160,9.0,0.508,15.92,3.98,"E6010")
SMAW_data("FILLET",10,"VERTICAL",1,3,25,85,21.6,0.520,34.56,10.29,"E6010")
SMAW_data("FILLET",10,"VERTICAL",2,3,25,85,21.6,0.520,34.56,10.29,"E6010")
SMAW_data("FILLET",10,"VERTICAL",1,4,110,14.3,0.520,25.18,6.67,"E6010")
SMAW_data("FILLET",10,"VERTICAL",2,4,110,14.3,0.520,25.18,6.67,"E6010")
SMAW_data("FILLET",12,"FLAT",1,6,3,230,7.8,0.705,14.81,3.41,"E6010")
SMAW_data("FILLET",12,"FLAT",2,6,3,230,7.8,0.705,14.81,3.41,"E6010")
SMAW_data("FILLET",12,"HORIZONTAL",1,6,3,220,7.8,0.705,15.54,3.41,"E6010")
SMAW_data("FILLET",12,"HORIZONTAL",2,6,3,220,7.8,0.705,15.54,3.41,"E6010")
SMAW_data("FILLET",12,"HORIZONTAL",3,6,3,220,7.8,0.705,15.54,3.41,"E6010")
SMAW_data("FILLET",12,"VERTICAL",1,4,110,19.5,0.706,34.19,9.23,"E6010")
SMAW_data("FILLET",12,"VERTICAL",2,4,110,19.5,0.706,34.19,9.23,"E6010")
SMAW_data("FILLET",12,"VERTICAL",3,4,110,19.5,0.706,34.19,9.23,"E6010")
SMAW_data("FILLET",12,"VERTICAL",4,4,110,19.5,0.706,34.19,9.23,"E6010")
SMAW_data("FILLET",5,"FLAT",1,4,180,3.2,0.116,4.66,1.07,"E6012")
SMAW_data("FILLET",5,"FLAT",1,5,220,2.0,0.116,3.75,0.51,"E6012")
SMAW_data("FILLET",5,"HORIZONTAL",1,4,180,3.2,0.116,4.66,1.07,"E6012")
```

SMAW\_data("FILLET",5,"HORIZONTAL",1,5,220,2.0,0.116,3.75,0.51,"E6012")  
 SMAW\_data("FILLET",5,"VERTICAL",1,3,25,115,6.2,0.148,9.86,2.61,"E6012")  
 SMAW\_data("FILLET",5,"VERTICAL",1,4,145,4.1,0.148,7.59,1.52,"E6012")  
 SMAW\_data("FILLET",5,"OVERHEAD",1,4,170,4.1,0.148,6.36,1.52,"E6012")  
 SMAW\_data("FILLET",5,"OVERHEAD",1,5,200,2.6,0.148,5.33,0.79,"E6012")  
 SMAW\_data("FILLET",6,"FLAT",1,5,220,2.9,0.164,5.33,0.93,"E6012")  
 SMAW\_data("FILLET",6,"FLAT",1,6,3,300,1.8,0.164,3.84,0.4,"E6012")  
 SMAW\_data("FILLET",6,"HORIZONTAL",1,4,180,4.5,0.164,6.63,1.74,"E6012")  
 SMAW\_data("FILLET",6,"HORIZONTAL",1,5,220,2.9,0.164,5.33,0.93,"E6012")  
 SMAW\_data("FILLET",6,"VERTICAL",1,3,25,115,8.5,0.202,13.46,3.74,"E6012")  
 SMAW\_data("FILLET",6,"VERTICAL",1,4,145,5.5,0.202,10.35,2.26,"E6012")  
 SMAW\_data("FILLET",6,"OVERHEAD",1,4,170,5.5,0.202,8.68,2.26,"E6012")  
 SMAW\_data("FILLET",6,"OVERHEAD",2,4,170,5.5,0.202,8.68,2.26,"E6012")  
 SMAW\_data("FILLET",6,"OVERHEAD",1,5,200,3.5,0.202,7.28,1.26,"E6012")  
 SMAW\_data("FILLET",6,"OVERHEAD",2,5,200,3.5,0.202,7.28,1.26,"E6013")  
 SMAW\_data("FILLET",8,"FLAT",1,5,220,7.5,0.287,9.31,2,"E6013")  
 SMAW\_data("FILLET",8,"FLAT",1,6,3,300,3.2,0.287,6.7,1.08,"E6013")  
 SMAW\_data("FILLET",8,"HORIZONTAL",1,4,180,7.8,0.287,11.57,3.41,"E6013")  
 SMAW\_data("FILLET",8,"HORIZONTAL",2,4,180,7.8,0.287,11.57,3.41,"E6013")  
 SMAW\_data("FILLET",8,"HORIZONTAL",3,4,180,7.8,0.287,11.57,3.41,"E6013")  
 SMAW\_data("FILLET",8,"HORIZONTAL",1,5,220,5.0,0.287,9.31,2,"E6013")  
 SMAW\_data("FILLET",8,"HORIZONTAL",2,5,220,5.0,0.287,9.31,2,"E6013")  
 SMAW\_data("FILLET",8,"VERTICAL",1,3,25,115,14.0,0.334,22.24,6.51,"E6013")  
 SMAW\_data("FILLET",8,"VERTICAL",2,3,25,115,14.0,0.334,22.24,6.51,"E6013")  
 SMAW\_data("FILLET",8,"VERTICAL",1,4,145,9.1,0.334,17.11,4.05,"E6013")  
 SMAW\_data("FILLET",8,"VERTICAL",2,4,145,9.1,0.334,17.11,4.05,"E6013")  
 SMAW\_data("FILLET",8,"OVERHEAD",1,4,170,9.1,0.334,14.35,4.06,"E6013")  
 SMAW\_data("FILLET",8,"OVERHEAD",2,4,170,9.1,0.334,14.35,4.06,"E6013")  
 SMAW\_data("FILLET",8,"OVERHEAD",3,4,170,9.1,0.334,14.35,4.06,"E6013")  
 SMAW\_data("FILLET",8,"OVERHEAD",1,5,200,5.8,0.334,12.02,2.41,"E6013")  
 SMAW\_data("FILLET",8,"OVERHEAD",2,5,200,5.8,0.334,12.02,2.41,"E6013")  
 SMAW\_data("FILLET",10,"FLAT",1,5,220,7.7,0.443,14.36,3.35,"E6013")  
 SMAW\_data("FILLET",10,"FLAT",1,6,3,300,4.9,0.443,10.33,1.94,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",1,4,180,12.1,0.443,17.84,5.53,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",2,4,180,12.1,0.443,17.84,5.53,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",3,4,180,12.1,0.443,17.84,5.53,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",4,4,180,12.1,0.443,17.84,5.53,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",1,5,220,7.7,0.443,14.36,3.35,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",2,5,220,7.7,0.443,14.36,3.35,"E6013")  
 SMAW\_data("FILLET",10,"HORIZONTAL",3,5,220,7.7,0.443,14.36,3.35,"E6013")  
 SMAW\_data("FILLET",10,"VERTICAL",1,3,25,115,20.9,0.498,33.14,9.94,"E6013")  
 SMAW\_data("FILLET",10,"VERTICAL",2,3,25,115,20.9,0.498,33.14,9.94,"E6013")  
 SMAW\_data("FILLET",10,"VERTICAL",3,3,25,115,20.9,0.498,33.14,9.94,"E6013")  
 SMAW\_data("FILLET",10,"VERTICAL",1,4,145,13.6,0.498,25.49,6.29,"E6013")  
 SMAW\_data("FILLET",10,"VERTICAL",2,4,145,13.6,0.498,25.49,6.29,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",1,4,170,13.6,0.498,21.38,6.29,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",2,4,170,13.6,0.498,21.38,6.29,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",3,4,170,13.6,0.498,21.38,6.29,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",4,4,170,13.6,0.498,21.38,6.29,"E6013")

SMAW\_data("FILLET",10,"OVERHEAD",1,5,200,8.7,0.498,17.91,3.84,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",2,5,200,8.7,0.498,17.91,3.84,"E6013")  
 SMAW\_data("FILLET",10,"OVERHEAD",3,5,200,8.7,0.498,17.91,3.84,"E6013")  
 SMAW\_data("FILLET",12,"FLAT",1,6.3,300,7.0,0.631,14.74,2.97,"E6013")  
 SMAW\_data("FILLET",12,"FLAT",2,6.3,300,7.0,0.631,14.74,2.97,"E6013")  
 SMAW\_data("FILLET",12,"HORIZONTAL",1,5.3,275,7.0,0.631,16.15,2.97,"E6013")  
 SMAW\_data("FILLET",12,"HORIZONTAL",2,5.3,275,7.0,0.631,16.15,2.97,"E6013")  
 SMAW\_data("FILLET",12,"HORIZONTAL",3,5.3,275,7.0,0.631,16.15,2.97,"E6013")  
 SMAW\_data("FILLET",12,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",12,"VERTICAL",2,5,170,8.4,0.482,20.7,1.48,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",1,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",2,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",3,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",4,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",5,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",12,"OVERHEAD",6,5,200,12.1,0.694,24.94,5.54,"E6013")  
 SMAW\_data("FILLET",15,"FLAT",1,6.3,300,10.7,0.976,22.78,4.87,"E6013")  
 SMAW\_data("FILLET",15,"FLAT",2,6.3,300,10.7,0.976,22.78,4.87,"E6013")  
 SMAW\_data("FILLET",15,"HORIZONTAL",1,6.3,275,10.7,0.976,24.97,4.87,"E6013")  
 SMAW\_data("FILLET",15,"HORIZONTAL",2,6.3,275,10.7,0.976,24.97,4.87,"E6013")  
 SMAW\_data("FILLET",15,"HORIZONTAL",3,6.3,275,10.7,0.976,24.97,4.87,"E6013")  
 SMAW\_data("FILLET",15,"HORIZONTAL",4,6.3,275,10.7,0.976,24.97,4.87,"E6013")  
 SMAW\_data("FILLET",15,"HORIZONTAL",5,6.3,275,10.7,0.976,24.97,4.87,"E6013")  
 SMAW\_data("FILLET",15,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",15,"VERTICAL",2,5,170,14.5,0.835,35.82,2.71,"E6013")  
 SMAW\_data("FILLET",15,"VERTICAL",3,5,170,14.5,0.835,35.82,2.71,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",1,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",2,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",3,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",4,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",5,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",6,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",7,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",8,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",15,"OVERHEAD",9,5,200,18.2,1.046,37.61,8.61,"E6013")  
 SMAW\_data("FILLET",18,"FLAT",1,6.3,300,15.3,1.394,32.54,7.17,"E6013")  
 SMAW\_data("FILLET",18,"FLAT",2,6.3,300,15.3,1.394,32.54,7.17,"E6013")  
 SMAW\_data("FILLET",18,"FLAT",3,6.3,300,15.3,1.394,32.54,7.17,"E6013")  
 SMAW\_data("FILLET",18,"FLAT",4,6.3,300,15.3,1.394,32.54,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",1,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",2,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",3,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",4,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",5,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",6,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",7,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"HORIZONTAL",8,6.3,275,15.3,1.394,35.67,7.17,"E6013")  
 SMAW\_data("FILLET",18,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",18,"VERTICAL",2,5,170,21.9,1.258,53.99,4.18,"E6013")

SMAW\_data("FILLET",18,"VERTICAL",3,5,170,21.9,1.258,53.99,4.18,"E6013")  
 SMAW\_data("FILLET",18,"VERTICAL",4,5,170,21.9,1.258,53.99,4.18,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",1,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",2,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",3,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",4,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",5,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",6,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",7,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",8,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",9,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",10,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",11,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",18,"OVERHEAD",12,5,200,25.6,1.304,52.83,12.29,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",1,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",2,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",3,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",4,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",5,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"FLAT",6,6.3,300,18.9,1.713,39.99,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",1,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",2,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",3,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",4,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",5,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",6,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",7,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",8,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",9,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"HORIZONTAL",10,6.3,275,18.9,1.713,43.84,8.93,"E6013")  
 SMAW\_data("FILLET",20,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",20,"VERTICAL",2,5,170,27.5,1.580,67.79,5.3,"E6013")  
 SMAW\_data("FILLET",20,"VERTICAL",3,5,170,27.5,1.580,67.79,5.3,"E6013")  
 SMAW\_data("FILLET",20,"VERTICAL",4,5,170,27.5,1.580,67.79,5.3,"E6013")  
 SMAW\_data("FILLET",20,"VERTICAL",5,5,170,27.5,1.580,67.79,5.3,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",1,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",2,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",3,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",4,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",5,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",6,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",7,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",8,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",9,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",10,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",11,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",12,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",13,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",20,"OVERHEAD",14,5,200,31.2,1.791,64.39,15.09,"E6013")

SMAW\_data("FILLET",20,"OVERHEAD",15,5,200,31.2,1.791,64.39,15.09,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",1,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",2,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",3,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",4,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",5,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",6,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",7,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"FLAT",8,6.3,300,22.7,2.065,48.21,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",1,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",2,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",3,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",4,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",5,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",6,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",7,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",8,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",9,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",10,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",11,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"HORIZONTAL",12,6.3,275,22.7,2.065,52.84,10.86,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",2,5,170,33.7,1.933,82.94,6.53,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",3,5,170,33.7,1.933,82.94,6.53,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",4,5,170,33.7,1.933,82.94,6.53,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",5,5,170,33.7,1.933,82.94,6.53,"E6013")  
 SMAW\_data("FILLET",22,"VERTICAL",6,5,170,33.7,1.933,82.94,6.53,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",1,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",2,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",3,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",4,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",5,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",6,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",7,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",8,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",9,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",10,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",11,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",12,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",13,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",14,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",15,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",16,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",17,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",18,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",19,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",22,"OVERHEAD",20,5,200,37.3,2.144,77.09,18.17,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",1,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",2,6.3,300,29.2,2.654,61.95,14.1,"E6013")



SMAW\_data("FILLET",25,"FLAT",3,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",4,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",5,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",6,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",7,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",8,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",9,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"FLAT",10,6.3,300,29.2,2.654,61.95,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",1,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",2,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",3,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",4,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",5,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",6,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",7,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",8,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",9,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",10,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",11,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",12,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",13,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",14,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"HORIZONTAL",15,6.3,275,29.2,2.654,67.9,14.1,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",1,4,145,5.8,0.211,10.81,0.95,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",2,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",3,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",4,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",5,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",6,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",7,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"VERTICAL",8,5,170,43.9,2.521,108.19,8.58,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",1,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",2,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",3,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",4,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",5,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",6,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",7,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",8,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",9,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",10,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",11,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",12,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",13,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",14,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",15,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",16,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",17,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",18,5,200,47.6,2.732,98.24,23.29,"E6013")

SMAW\_data("FILLET",25,"OVERHEAD",19,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",20,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",21,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",22,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",23,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",24,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",25,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",26,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",25,"OVERHEAD",27,5,200,47.6,2.732,98.24,23.29,"E6013")  
 SMAW\_data("FILLET",5,"FLAT",1.4,255,2.2,0.114,2.8,0.58,"E7014")  
 SMAW\_data("FILLET",5,"FLAT",1.5,300,1.4,0.114,2.2,0.19,"E7014")  
 SMAW\_data("FILLET",5,"HORIZONTAL",1.4,245,2.2,0.114,2.98,0.58,"E7014")  
 SMAW\_data("FILLET",5,"HORIZONTAL",1.5,270,1.4,0.114,2.57,0.19,"E7014")  
 SMAW\_data("FILLET",5,"VERTICAL",1.4,150,3.3,0.173,11.65,1.14,"E7014")  
 SMAW\_data("FILLET",5,"OVERHEAD",1.4,150,3.1,0.163,10.98,1.05,"E7014")  
 SMAW\_data("FILLET",5,"OVERHEAD",1.5,200,2.0,0.163,6.01,0.48,"E7014")  
 SMAW\_data("FILLET",6,"FLAT",1.4,255,3.1,0.163,4.1,0.04,"E7014")  
 SMAW\_data("FILLET",6,"FLAT",1.5,300,2.0,0.163,3.14,0.48,"E7014")  
 SMAW\_data("FILLET",6,"HORIZONTAL",1.4,245,3.1,0.163,4.26,1.04,"E7014")  
 SMAW\_data("FILLET",6,"HORIZONTAL",1.5,270,2.0,0.163,3.67,0.48,"E7014")  
 SMAW\_data("FILLET",6,"VERTICAL",1.4,150,4.2,0.222,14.89,1.6,"E7014")  
 SMAW\_data("FILLET",6,"VERTICAL",2.4,150,4.2,0.222,14.89,1.6,"E7014")  
 SMAW\_data("FILLET",6,"OVERHEAD",1.4,150,4.3,0.227,15.26,1.65,"E7014")  
 SMAW\_data("FILLET",6,"OVERHEAD",1.5,200,2.7,0.227,8.35,0.87,"E7014")  
 SMAW\_data("FILLET",8,"FLAT",1.5,300,3.4,0.285,5.51,1.22,"E7014")  
 SMAW\_data("FILLET",8,"FLAT",1.6,370,2.2,0.285,4.14,0.59,"E7014")  
 SMAW\_data("FILLET",8,"HORIZONTAL",1.5,270,3.4,0.285,6.43,1.22,"E7014")  
 SMAW\_data("FILLET",8,"HORIZONTAL",1.6,3,305,2.2,0.285,5.38,0.59,"E7014")  
 SMAW\_data("FILLET",8,"VERTICAL",1.4,150,6.5,0.340,22.85,2.73,"E7014")  
 SMAW\_data("FILLET",8,"VERTICAL",2.4,150,6.5,0.340,22.85,2.73,"E7014")  
 SMAW\_data("FILLET",8,"OVERHEAD",1.4,150,7.3,0.383,27.74,3.13,"E7014")  
 SMAW\_data("FILLET",8,"OVERHEAD",2.4,150,7.3,0.383,27.74,3.13,"E7014")  
 SMAW\_data("FILLET",8,"OVERHEAD",3.4,150,7.3,0.383,27.74,3.13,"E7014")  
 SMAW\_data("FILLET",8,"OVERHEAD",1.5,200,4.6,0.383,14.09,1.81,"E7014")  
 SMAW\_data("FILLET",8,"OVERHEAD",2.5,200,4.6,0.383,14.09,1.81,"E7014")  
 SMAW\_data("BUTT",3,"FLAT",1.3,15,80,3.375,0.080,5.02,1.69,"E6010")  
 SMAW\_data("BUTT",4,"FLAT",1.3,15,80,5.125,0.120,7.6,2.56,"E6010")  
 SMAW\_data("BUTT",5,"FLAT",1.3,15,80,7.3,0.160,10.82,3.65,"E6010")  
 SMAW\_data("BUTT",6,"FLAT",1.3,15,80,5.85,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",6,"FLAT",2.4,145,2.175,0.080,3.03,1.09,"E6010")  
 SMAW\_data("BUTT",8,"FLAT",1.3,15,80,5.85,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",8,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6010")  
 SMAW\_data("BUTT",8,"FLAT",3.5,200,2.125,0.120,3.43,1.07,"E6010")  
 SMAW\_data("BUTT",10,"FLAT",1.3,15,80,5.85,0.130,8.68,2.93,"E6010")  
 SMAW\_data("BUTT",10,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6010")  
 SMAW\_data("BUTT",10,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6010")  
 SMAW\_data("BUTT",10,"FLAT",4.6,3,275,1.9,0.170,3.6,0.96,"E6010")  
 SMAW\_data("BUTT",12.5,"FLAT",1.3,15,80,5.8,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",12.5,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6010")

SMAW\_data("BUTT",12.5,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6010")  
 SMAW\_data("BUTT",12.5,"FLAT",4.6,3,275,4.9,0.140,9.19,2.45,"E6010")  
 SMAW\_data("BUTT",15,"FLAT",1.3,15,80,5.8,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",15,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6010")  
 SMAW\_data("BUTT",15,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6010")  
 SMAW\_data("BUTT",15,"FLAT",4.6,3,275,4.25,0.380,7.975,2.125,"E6010")  
 SMAW\_data("BUTT",15,"FLAT",5.6,3,275,4.25,0.380,7.975,2.125,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",1.4,80,5.8,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",2.5,145,2.6,0.090,3.63,1.3,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",3.6,3,200,2.1,0.120,3.45,1.07,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",4.6,3,275,4.4,0.395,8.2525,2.1975,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",5.6,3,275,4.4,0.395,8.2525,2.1975,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",6.6,3,275,4.4,0.395,8.2525,2.1975,"E6010")  
 SMAW\_data("BUTT",20,"FLAT",7.6,3,275,4.4,0.395,8.2525,2.1975,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",1.4,80,5.8,0.130,8.68,2.92,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",2.5,145,2.6,0.090,3.63,1.2,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",3.6,3,200,2.1,0.120,3.45,1.07,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",4.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",5.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",6.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",7.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",8.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",9.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",10.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",25,"FLAT",11.6,3,275,3.65,0.328,6.8475,1.82375,"E6010")  
 SMAW\_data("BUTT",3,"FLAT",1.3,15,80,3.375,0.080,5.02,1.69,"E6011")  
 SMAW\_data("BUTT",4,"FLAT",1.3,15,80,5.125,0.120,7.6,2.56,"E6011")  
 SMAW\_data("BUTT",5,"FLAT",1.3,15,80,7.3,0.160,10.82,3.65,"E6011")  
 SMAW\_data("BUTT",6,"FLAT",1.3,15,80,5.85,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",6,"FLAT",2.4,145,2.175,0.080,3.03,1.09,"E6011")  
 SMAW\_data("BUTT",8,"FLAT",1.3,15,80,5.85,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",8,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",8,"FLAT",3.5,200,2.125,0.120,3.43,1.07,"E6011")  
 SMAW\_data("BUTT",10,"FLAT",1.3,15,80,5.85,0.130,8.68,2.93,"E6011")  
 SMAW\_data("BUTT",10,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",10,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",10,"FLAT",4.6,3,275,1.9,0.170,3.6,0.96,"E6011")  
 SMAW\_data("BUTT",12.5,"FLAT",1.3,15,80,5.8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",12.5,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",12.5,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",12.5,"FLAT",4.6,3,275,4.9,0.140,9.19,2.45,"E6011")  
 SMAW\_data("BUTT",15,"FLAT",1.3,15,80,5.8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",15,"FLAT",2.4,145,2.6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",15,"FLAT",3.5,200,2.1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",15,"FLAT",4.6,3,275,4.25,0.380,7.975,2.125,"E6011")  
 SMAW\_data("BUTT",15,"FLAT",5.6,3,275,4.25,0.380,7.975,2.125,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",1.4,80,5.8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",2.5,145,2.6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",3.6,3,200,2.1,0.120,3.45,1.07,"E6011")

SMAW\_data("BUTT",20,"FLAT",4,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",5,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",6,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"FLAT",7,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",1,4,80,5,8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",2,5,145,2,6,0.090,3.63,1.2,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",3,6,3,200,2,1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",4,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",5,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",6,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",7,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",8,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",9,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",10,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"FLAT",11,6,3,275,3,65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",3,"HORIZONTAL",1,3,15,80,3,375,0.080,5.02,1.69,"E6011")  
 SMAW\_data("BUTT",4,"HORIZONTAL",1,3,15,80,5,125,0.120,7.6,2.56,"E6011")  
 SMAW\_data("BUTT",5,"HORIZONTAL",1,3,15,80,7,3,0.160,10.82,3.65,"E6011")  
 SMAW\_data("BUTT",6,"HORIZONTAL",1,3,15,80,5,85,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",6,"HORIZONTAL",2,4,145,2,175,0.080,3.03,1.09,"E6011")  
 SMAW\_data("BUTT",8,"HORIZONTAL",1,3,15,80,5,85,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",8,"HORIZONTAL",2,4,145,2,6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",8,"HORIZONTAL",3,5,200,2,125,0.120,3.43,1.07,"E6011")  
 SMAW\_data("BUTT",10,"HORIZONTAL",1,3,15,80,5,85,0.130,8.68,2.93,"E6011")  
 SMAW\_data("BUTT",10,"HORIZONTAL",2,4,145,2,6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",10,"HORIZONTAL",3,5,200,2,1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",10,"HORIZONTAL",4,6,3,275,1,9,0.170,3,6,0.96,"E6011")  
 SMAW\_data("BUTT",13,"HORIZONTAL",1,3,15,80,5,8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",13,"HORIZONTAL",2,4,145,2,6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",13,"HORIZONTAL",3,5,200,2,1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",13,"HORIZONTAL",4,6,3,275,4,9,0.140,9.19,2.45,"E6011")  
 SMAW\_data("BUTT",13,"HORIZONTAL",1,3,25,80,5,8,0.130,8.68,2.92,"E4043")  
 SMAW\_data("BUTT",13,"HORIZONTAL",2,4,145,2,6,0.090,3.63,1.3,"E4043")  
 SMAW\_data("BUTT",13,"HORIZONTAL",3,5,200,2,1,0.120,3.45,1.07,"E4043")  
 SMAW\_data("BUTT",13,"HORIZONTAL",4,6,3,275,4,9,0.140,9.19,2.45,"E4043")  
 SMAW\_data("BUTT",15,"HORIZONTAL",1,3,15,80,5,8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",15,"HORIZONTAL",2,4,145,2,6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",15,"HORIZONTAL",3,5,200,2,1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",15,"HORIZONTAL",4,6,3,275,4,25,0.380,7.975,2.125,"E6011")  
 SMAW\_data("BUTT",15,"HORIZONTAL",5,6,3,275,4,25,0.380,7.975,2.125,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",1,4,80,5,8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",2,5,145,2,6,0.090,3.63,1.3,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",3,6,3,200,2,1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",4,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",5,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",6,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",20,"HORIZONTAL",7,6,3,275,4,4,0.395,8.2525,2.1975,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",1,4,80,5,8,0.130,8.68,2.92,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",2,5,145,2,6,0.090,3.63,1.2,"E6011")

SMAW\_data("BUTT",25,"HORIZONTAL",3,6.3,200,2.1,0.120,3.45,1.07,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",4,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",5,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",6,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",7,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",8,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",9,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",10,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",25,"HORIZONTAL",11,6.3,275,3.65,0.328,6.8475,1.82375,"E6011")  
 SMAW\_data("BUTT",5,"OVERHEAD",1,5,200,2.6,0.148,5.33,0.79,"E6012")  
 SMAW\_data("BUTT",6,"FLAT",1,5,220,2.9,0.164,5.33,0.93,"E6012")  
 SMAW\_data("BUTT",6,"FLAT",1,6.3,300,1.8,0.164,3.84,0.4,"E6012")  
 SMAW\_data("BUTT",6,"HORIZONTAL",1,4,180,4.5,0.164,6.63,1.74,"E6012")  
 SMAW\_data("BUTT",6,"HORIZONTAL",1,5,220,2.9,0.164,5.33,0.93,"E6012")  
 SMAW\_data("BUTT",6,"VERTICAL",1,3.25,115,8.5,0.202,13.46,3.74,"E6012")  
 SMAW\_data("BUTT",6,"VERTICAL",1,4,145,5.5,0.202,10.35,2.26,"E6012")  
 SMAW\_data("BUTT",6,"OVERHEAD",1,4,170,5.5,0.202,8.68,2.26,"E308-17")  
 SMAW\_data("BUTT",6,"OVERHEAD",2,4,170,5.5,0.202,8.68,2.26,"E308-17")  
 SMAW\_data("BUTT",6,"OVERHEAD",1,5,200,3.5,0.202,7.28,1.26,"E308-17")  
 SMAW\_data("BUTT",6,"OVERHEAD",2,5,200,3.5,0.202,7.28,1.26,"E308-17")  
 GMAW\_data("FILLET",3,"FLAT",1,0.8,0.058,21,600,150,15.5,37.1,1.62,"CO2")  
 GMAW\_data("FILLET",3,"FLAT",1,0.8,0.058,28,1120,230,15.5,67.9,0.88,"CO2")  
 GMAW\_data("FILLET",3,"HORIZONTAL",1,0.8,0.048,21,575,145,13.42,5,1.41,"CO2")  
 GMAW\_data("FILLET",3,"HORIZONTAL",1,0.8,0.048,26,950,205,13.69,0.87,"CO2")  
 GMAW\_data("FILLET",3,"DOWNWARD",1,0.8,0.049,19,425,115,1301,31.5,1.9,"CO2")  
 GMAW\_data("FILLET",3,"DOWNWARD",1,0.8,0.049,21,725,170,1301,52.8,1.14,"CO2")  
 GMAW\_data("FILLET",3,"OVERHEAD",1,0.8,0.058,21,640,155,15.5,39.5,1.52,"CO2")  
 GMAW\_data("FILLET",3,"OVERHEAD",1,0.8,0.058,25,900,195,15.5,55,1.09,"CO2")  
 GMAW\_data("FILLET",4,"FLAT",1,0.8,0.091,21,600,150,24.2,23.7,2.53,"CO2")  
 GMAW\_data("FILLET",4,"FLAT",1,0.8,0.091,28,1120,230,24.2,43.3,1.38,"CO2")  
 GMAW\_data("FILLET",4,"HORIZONTAL",1,0.8,0.084,21,575,145,22.6,24.4,2.46,"CO2")  
 GMAW\_data("FILLET",4,"HORIZONTAL",1,0.8,0.084,26,950,205,22.6,39.6,1.51,"CO2")  
 GMAW\_data("FILLET",4,"DOWNWARD",1,0.8,0.08,19,425,115,21.5,19.2,3.13,"CO2")  
 GMAW\_data("FILLET",4,"DOWNWARD",1,0.8,0.08,20,725,170,21.5,32.1,1.87,"CO2")  
 GMAW\_data("FILLET",4,"OVERHEAD",1,0.8,0.091,21,640,155,24.2,25.2,2.38,"CO2")  
 GMAW\_data("FILLET",4,"OVERHEAD",1,0.8,0.091,25,900,195,24.2,35.1,1.71,"CO2")  
 GMAW\_data("FILLET",5,"FLAT",1,0.8,0.131,21,600,150,35,16.4,3.66,"CO2")  
 GMAW\_data("FILLET",5,"FLAT",1,0.8,0.131,28,1120,230,35,30.2,"CO2")  
 GMAW\_data("FILLET",5,"HORIZONTAL",1,0.8,0.13,21,575,145,34.7,15.9,3.78,"CO2")  
 GMAW\_data("FILLET",5,"HORIZONTAL",1,0.8,0.13,26,950,205,34.7,25.8,2.33,"CO2")  
 GMAW\_data("FILLET",5,"DOWNWARD",1,0.8,0.119,19,425,115,32,12.9,4.66,"CO2")  
 GMAW\_data("FILLET",5,"DOWNWARD",1,0.8,0.119,21,725,170,32,21.6,2.78,"CO2")  
 GMAW\_data("FILLET",5,"OVERHEAD",1,0.8,0.131,21,640,155,35,17.5,3.44,"CO2")  
 GMAW\_data("FILLET",5,"OVERHEAD",1,0.8,0.131,25,900,195,35,24.3,2.47,"CO2")  
 GMAW\_data("FILLET",6,"FLAT",1,0.8,0.090,21,600,150,47.8,12.5,"CO2")  
 GMAW\_data("FILLET",6,"FLAT",2,0.8,0.090,21,600,150,47.8,12.5,"CO2")  
 GMAW\_data("FILLET",6,"FLAT",1,0.8,0.179,28,1120,230,47.8,21.9,2.73,"CO2")  
 GMAW\_data("FILLET",6,"HORIZONTAL",1,0.8,0.093,21,575,145,49.4,11.1,5.38,"CO2")  
 GMAW\_data("FILLET",6,"HORIZONTAL",2,0.8,0.093,21,575,145,49.4,11.1,5.38,"CO2")

GMAW\_data("FILLET",6,"HORIZONTAL",1,0.8,0.185,28,950,205,49.4,18.1,3.31,"CO2")  
 GMAW\_data("FILLET",6,"DOWNWARD",1,0.8,0.167,20,425,115,44.5,9.2,6.5,"CO2")  
 GMAW\_data("FILLET",6,"DOWNWARD",1,0.8,0.167,19,725,170,44.5,15.5,3.88,"CO2")  
 GMAW\_data("FILLET",6,"OVERHEAD",1,0.8,0.090,25,640,155,47.8,12.8,4.69,"CO2")  
 GMAW\_data("FILLET",6,"OVERHEAD",2,0.8,0.090,21,640,155,47.8,12.8,4.69,"CO2")  
 GMAW\_data("FILLET",6,"OVERHEAD",1,0.8,0.090,25,900,195,47.8,17.8,3.38,"CO2")  
 GMAW\_data("FILLET",6,"OVERHEAD",2,0.8,0.090,25,900,195,47.8,17.8,3.38,"CO2")  
 GMAW\_data("BUTT",6,"OVERHEAD",1,0.8,0.090,25,640,155,47.8,12.8,4.69,"CO2")  
 GMAW\_data("BUTT",6,"OVERHEAD",2,0.8,0.090,21,640,155,47.8,12.8,4.69,"CO2")  
 GMAW\_data("BUTT",6,"OVERHEAD",1,0.8,0.090,25,900,195,47.8,17.8,3.38,"CO2")  
 GMAW\_data("BUTT",6,"OVERHEAD",2,0.8,0.090,25,900,195,47.8,17.8,3.38,"CO2")  
 GMAW\_data("FILLET",8,"FLAT",1,0.8,0.149,21,600,150,79.6,7.2,8.32,"CO2")  
 GMAW\_data("FILLET",8,"FLAT",2,0.8,0.149,21,600,150,79.6,7.2,8.32,"CO2")  
 GMAW\_data("FILLET",8,"FLAT",1,0.8,0.298,28,1120,230,79.6,13.2,4.55,"CO2")  
 GMAW\_data("FILLET",8,"HORIZONTAL",1,0.8,0.162,21,575,145,86.3,6.4,9.4,"CO2")  
 GMAW\_data("FILLET",8,"HORIZONTAL",2,0.8,0.162,21,575,145,86.3,6.4,9.4,"CO2")  
 GMAW\_data("FILLET",8,"HORIZONTAL",1,0.8,0.162,26,950,205,86.3,10.4,5.79,"CO2")  
 GMAW\_data("FILLET",8,"HORIZONTAL",2,0.8,0.162,26,950,205,86.3,10.4,5.79,"CO2")  
 GMAW\_data("FILLET",8,"DOWNWARD",1,0.8,0.142,19,425,115,76.5,4.1,1.08,"CO2")  
 GMAW\_data("FILLET",8,"DOWNWARD",2,0.8,0.142,19,425,115,76.5,4.1,1.08,"CO2")  
 GMAW\_data("FILLET",8,"DOWNWARD",1,0.8,0.142,21,725,170,76.9,1.6,6.1,"CO2")  
 GMAW\_data("FILLET",8,"DOWNWARD",2,0.8,0.142,21,725,170,76.9,1.6,6.1,"CO2")  
 GMAW\_data("FILLET",8,"OVERHEAD",1,0.8,0.149,21,640,155,79.6,7.7,7.82,"CO2")  
 GMAW\_data("FILLET",8,"OVERHEAD",2,0.8,0.149,21,640,155,79.6,7.7,7.82,"CO2")  
 GMAW\_data("FILLET",8,"OVERHEAD",1,0.8,0.149,25,900,195,79.6,10.7,5.62,"CO2")  
 GMAW\_data("FILLET",8,"OVERHEAD",2,0.8,0.149,25,900,195,79.6,10.7,5.62,"CO2")  
 GMAW\_data("FILLET",10,"FLAT",1,0.8,0.224,21,600,150,119.7,4.8,12.51,"CO2")  
 GMAW\_data("FILLET",10,"FLAT",2,0.8,0.224,21,600,150,119.7,4.8,12.51,"CO2")  
 GMAW\_data("FILLET",10,"FLAT",1,0.8,0.447,28,1120,230,119.7,8.8,6.85,"CO2")  
 GMAW\_data("FILLET",10,"HORIZONTAL",1,0.8,0.166,21,575,145,133.4,1.14,4.9,"CO2")  
 GMAW\_data("FILLET",10,"HORIZONTAL",2,0.8,0.166,21,575,145,133.4,1.14,4.9,"CO2")  
 GMAW\_data("FILLET",10,"HORIZONTAL",3,0.8,0.166,21,575,145,133.4,1.14,4.9,"CO2")  
 GMAW\_data("FILLET",10,"HORIZONTAL",1,0.8,0.249,26,950,205,133.6,7.8,8.92,"CO2")  
 GMAW\_data("FILLET",10,"HORIZONTAL",2,0.8,0.249,26,950,205,133.6,7.8,8.92,"CO2")  
 GMAW\_data("FILLET",10,"VERTICAL",1,0.8,0.065,19,250,80,17.4,1.9,32.39,"CO2")  
 GMAW\_data("FILLET",10,"VERTICAL",2,0.8,0.382,19,200,70,102.3,1.9,30.9,"CO2")  
 GMAW\_data("FILLET",10,"VERTICAL",1,0.8,0.144,21,790,180,380.5,7.3,8.2,"CO2")  
 GMAW\_data("FILLET",10,"VERTICAL",2,0.8,0.303,22,575,145,81.2,6.8,8.84,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",1,0.8,0.149,21,640,155,119.7,5.1,11.75,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",2,0.8,0.149,21,640,155,119.7,5.1,11.75,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",3,0.8,0.149,21,640,155,119.7,5.1,11.75,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",1,0.8,0.149,25,900,195,119.7,7.1,8.46,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",2,0.8,0.149,25,900,195,119.7,7.1,8.46,"CO2")  
 GMAW\_data("FILLET",10,"OVERHEAD",3,0.8,0.149,25,900,195,119.7,7.1,8.46,"CO2")  
 GMAW\_data("BUTT",13,"HORIZONTAL",1,0.8,0.189,25,900,195,150.7,7.1,10.46,"CO2")  
 GMAW\_data("BUTT",13,"HORIZONTAL",2,0.8,0.189,25,900,195,150.7,7.1,10.46,"CO2")  
 GMAW\_data("BUTT",13,"HORIZONTAL",3,0.8,0.189,25,900,195,150.7,7.1,10.46,"CO2")

## **APPENDIX E - Listing of Prolog Programs**

### **E.1 PROCESS.PRO - Process Selector**

project "neurex"

include "global.pro"

include "tdoms.pro"

include "tpreds.pro"

include "menu.pro"

database - criteria

Automation\_criteria (string,real,real,real,real)

Plate\_thickness\_criteria (string,real,real,real,real)

Strength\_criteria (string,real,real,real,real)

Seam\_length\_criteria(string,real,real,real,real)

Leg\_length\_criteria(string,real,real,real,real)

Parent\_material\_criteria (string,real,real,real,real)

Weld\_type\_criteria (string,real,real,real,real)

Position\_criteria (string,real,real ,real ,real )

Appearance\_criteria (string,real,real,real,real)

Deposition\_criteria (string,real,real,real,real)

Weld\_quality\_criteria (string,real,real,real,real)

database - process

process(string,string,real)

predicates

get\_max\_four(real,real,real,real,real)

get\_max\_criteria(real)

get\_max\_PM(real)

get\_max\_WT(real)

get\_max\_POS(real)

get\_max\_SL(real)

get\_max\_PT(real)

get\_max\_str(real)

get\_max\_app(real)

get\_max\_dep(real)

get\_max\_auto(real)

get\_max\_qual(real)

get\_max\_LL(real)

get\_process

compare\_selections

consult\_usr\_data

consult\_neural\_net

validate\_SMAW(real,real)

validate\_saw(real,real)

get\_PM(string,string)

get\_WT(string,string)

get\_POSI(string,string)

get\_APP(string,string)

```

get_DEP(string,string)
get_WQ(string,string)
get_AUT(string,string)
get_strength(string,string)
get_LL(string,string)
get_PT(string,string)
get_SL(string,string)
consult_criteria
consult_weights
Seam_length_crit (real,string)
Plate_thickness_crit (real,string)
Strength_crit (real,string)
Leg_length_crit (real,string)
final_process(string,real,real,real,real,real,real)
final_neural_process(string,real,real,real,real)
get_process_param (string, string)
retrain_neural_net (string)

```

clauses

**/\* Clauses for the PROCESS SELECTION MODULE \*/**

**Process\_Select :-**

```

makewindow(91,15,24,"Process Selection",2,0,23,80),
get_process,
write("Press any key to continue ....."),
readchar(_),
removewindow(91,1),
makewindow(92,7,14,"Neural Network Consultation",0,0,25,80),
write("Please wait ...."),nl,
write("Let me consult the neural network"),
consult_neural_net,
save("process.dba",process).

```

**get\_max\_four (A,B,C,D,MAX) :-**

```

    A >= B,
    A >= C,
    A >= D,
    MAX = A.

```

**get\_max\_four (A,B,C,D,MAX) :-**

```

    B >= A,
    B >= C,
    B >= D,
    MAX = B.

```

**get\_max\_four (A,B,C,D,MAX) :-**

```

    C >= B,
    C >= A,

```



C >= D,  
MAX = C.

get\_max\_four (A,B,C,D,MAX) :-

D >= B,  
D >= C,  
D >= A,  
MAX = D.

get\_max\_PM(PM) :-

Parent\_material\_criteria ("STAINLESS STEEL",SMAW1,GMAW1,GTAW1,SAW1),  
Parent\_material\_criteria ("CARBON STEEL",SMAW2,GMAW2,GTAW2,SAW2),  
Parent\_material\_criteria ("ALUMINUM ALLOY",SMAW3,GMAW3,GTAW3,SAW3),  
get\_max\_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),  
get\_max\_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),  
get\_max\_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),  
MAX4 = 0,  
get\_max\_four (MAX1,MAX2,MAX3,MAX4,MAX),  
PM = MAX.

get\_max\_WT(WT):-

Weld\_type\_criteria ("BUTT",SMAW1,GMAW1,GTAW1,SAW1),  
Weld\_type\_criteria ("FILLET",SMAW2,GMAW2,GTAW2,SAW2),  
get\_max\_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),  
get\_max\_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),  
MAX3 = 0,  
MAX4 = 0,  
get\_max\_four (MAX1,MAX2,MAX3,MAX4,MAX),  
WT = MAX.

get\_max\_POS(POS):-

Position\_criteria ("FLAT",SMAW1,GMAW1,GTAW1,SAW1),  
Position\_criteria ("HORIZONTAL",SMAW2,GMAW2,GTAW2,SAW2),  
Position\_criteria ("VERTICAL",SMAW3,GMAW3,GTAW3,SAW3),  
Position\_criteria ("DOWNWARD",SMAW4,GMAW4,GTAW4,SAW4),  
Position\_criteria ("OVERHEAD",SMAW5,GMAW5,GTAW5,SAW5),  
get\_max\_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),  
get\_max\_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),  
get\_max\_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),  
get\_max\_four (SMAW4,GMAW4,GTAW4,SAW4,MAX4),  
get\_max\_four (SMAW5,GMAW5,GTAW5,SAW5,MAX5),  
get\_max\_four (MAX1,MAX2,MAX3,MAX4,MAX),  
MAX6=0,  
MAX7=0,  
get\_max\_four (MAX,MAX5,MAX6,MAX7,MAXX),  
POS = MAXX.

get\_max\_SL(SL):-

```
Seam_length_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
Seam_length_criteria ("LOW",SMAW2,GMAW2,GTAW2,SAW2),
get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
MAX3 = 0,
MAX4 = 0,
get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
SL = MAX.
```

get\_max\_PT(PT):-

```
Plate_thickness_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
Plate_thickness_criteria ("MEDIUM",SMAW2,GMAW2,GTAW2,SAW2),
Plate_thickness_criteria ("LOW",SMAW3,GMAW3,GTAW3,SAW3),
get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
get_max_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),
MAX4 = 0,
get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
PT = MAX.
```

get\_max\_str(STR):-

```
Strength_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
Strength_criteria ("LOW",SMAW2,GMAW2,GTAW2,SAW2),
get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
MAX3 = 0,
MAX4 = 0,
get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
STR = MAX.
```

get\_max\_app(APP):-

```
Appearance_criteria("FLAT",SMAW1,GMAW1,GTAW1,SAW1),
Appearance_criteria ("CONVEX",SMAW2,GMAW2,GTAW2,SAW2),
Appearance_criteria ("CONCAVE",SMAW3,GMAW3,GTAW3,SAW3),
get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
get_max_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),
MAX4 = 0,
get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
APP = MAX.
```

get\_max\_dep(DEP) :-

```
Deposition_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
Deposition_criteria ("MEDIUM",SMAW2,GMAW2,GTAW2,SAW2),
Deposition_criteria ("LOW",SMAW3,GMAW3,GTAW3,SAW3),
```

```

get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
get_max_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),
MAX4 = 0,
get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
DEP = MAX.

```

```

get_max_qual(QUAL) :-
    Weld_quality_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
    Weld_quality_criteria ("MEDIUM",SMAW2,GMAW2,GTAW2,SAW2),
    Weld_quality_criteria ("LOW",SMAW3,GMAW3,GTAW3,SAW3),
    get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
    get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
    get_max_four (SMAW3,GMAW3,GTAW3,SAW3,MAX3),
    MAX4 = 0,
    get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
    QUAL = MAX.

```

```

get_max_auto (AUTO) :-
    Automation_criteria("YES",SMAW1,GMAW1,GTAW1,SAW1),
    Automation_criteria ("NO",SMAW2,GMAW2,GTAW2,SAW2),
    get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
    get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
    MAX3 = 0,
    MAX4 = 0,
    get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
    AUTO = MAX.

```

```

get_max_LL(LL):-
    Leg_length_criteria ("HIGH",SMAW1,GMAW1,GTAW1,SAW1),
    leg_length_criteria ("LOW",SMAW2,GMAW2,GTAW2,SAW2),
    get_max_four (SMAW1,GMAW1,GTAW1,SAW1,MAX1),
    get_max_four (SMAW2,GMAW2,GTAW2,SAW2,MAX2),
    MAX3 = 0,
    MAX4 = 0,
    get_max_four (MAX1,MAX2,MAX3,MAX4,MAX),
    LL = MAX.

```

```

get_max_criteria(MAXCRIT) :-
    consult_criteria,
    get_max_PM(PM),
    get_max_WT(WT),
    get_max_POS(POS),
    get_max_SL(SL),
    get_max_PT(PT),
    get_max_str(STR),
    get_max_app(APP),

```

```

get_max_dep(DEP),
get_max_qual(QUAL),
get_max_LL(LL),
get_max_auto (AUTO),
MAXCRIT=PM+WT+POS+SL+PT+STR+APP+DEP+QUAL+LL+AUTO,
write("Maximum Criteria Weight: ",MAXCRIT),nl,!.

get_process :-
!,
consult_usr_data,
consult_criteria,
job_data(Parent_Material,Weld_type,Position,Seam_length,
        Plate_thickness,Strength,Appearance,Deposition,
        Weld_Quality.Automation,Leg_length),
get_max_criteria (MAXCRIT),
Parent_material_criteria (Parent_Material,SMAW1,GMAW1,GTAW1,SAW1),
Weld_type_criteria (Weld_type,SMAW2,GMAW2,GTAW2,SAW2),
Position_criteria (Position,SMAW3,GMAW3,GTAW3,SAW3),
Seam_length_crit (Seam_Length,Factor4),
Seam_length_criteria (Factor4,SMAW4,GMAW4,GTAW4,SAW4),
Plate_thickness_crit (Plate_thickness,Factor5),
Plate_thickness_criteria (Factor5,SMAW5,GMAW5,GTAW5,SAW5),
Strength_crit (Strength,Factor6),
Strength_criteria (Factor6,SMAW6,GMAW6,GTAW6,SAW6),
Appearance_criteria (Appearance,SMAW7,GMAW7,GTAW7,SAW7),
Deposition_criteria (Deposition,SMAW8,GMAW8,GTAW8,SAW8),
        Weld_quality_criteria (Weld_Quality,SMAW9,GMAW9,GTAW9,SAW9),
        Automation_criteria (Automation,SMAW10,GMAW10,GTAW10,SAW10),
        Leg_length_crit (Leg_length,Factor11),
        Leg_length_criteria (Factor11,SMAW11,GMAW11,GTAW11,SAW11),

TOTAL_SMAW=SMAW1+SMAW2+SMAW3+SMAW4+SMAW5+SMAW6+SMAW7+SMAW8
+SMAW9+SMAW10+SMAW11,

TOTAL_GMAW=GMAW1+GMAW2+GMAW3+GMAW4+GMAW5+GMAW6+GMAW7+GMA
W8+GMAW9+GMAW10+GMAW11,

TOTAL_GTAW=GTAW1+GTAW2+GTAW3+GTAW4+GTAW5+GTAW6+GTAW7+GTAW8+
GTAW9+GTAW10+GTAW11,

TOTAL_SAW=SAW1+SAW2+SAW3+SAW4+SAW5+SAW6+SAW7+SAW8+SAW9+SAW10+
SAW11,
        validate_SMAW(TOTAL_SMAW,TOTAL_SMAW1),
        validate_saw(TOTAL_SAW,TOTAL_SAW1),
        write(TOTAL_SMAW1," ",TOTAL_GMAW," ",TOTAL_GTAW,"
",TOTAL_SAW1),nl,
        write("Press any key to continue .. "),
        readchar(_),
        write("\n\nPROCESSES APPLICABLE\n====="),nl,

```

```
final_process(X,TOTAL_SMAW1,TOTAL_GMAW,TOTAL_GTAW,TOTAL_SAW1,Confidence,
MAXCRIT),
```

```
    write("\nExpert System recommends ",X," with a confidence factor of ",Confidence),nl.
```

```
final_process("SMAW",A,B,C,D,E,MAX) :-
```

```
    A > B,
```

```
    A > C,
```

```
    A > D,
```

```
    E=A/MAX,
```

```
    assert(process("expert","SMAW",E)),
```

```
    write("Expert System recommends SMAW with a Confidence Factor of ",E).
```

```
final_process("GMAW",A,B,C,D,E,MAX) :-
```

```
    B > A,
```

```
    B > C,
```

```
    B > D,
```

```
    E=B/MAX,
```

```
    assert(process("expert","GMAW",E)),
```

```
    write("GMAW\n").
```

```
final_process("GTAW",A,B,C,D,E,MAX) :-
```

```
    C > B,
```

```
    C > A,
```

```
    C > D,
```

```
    E=C/MAX,
```

```
    assert(process("expert","GTAW",E)),
```

```
    write("GTAW\n").
```

```
final_process("SAW",A,B,C,D,E,MAX) :-
```

```
    D > B,
```

```
    D > C,
```

```
    D > A,
```

```
    E=D/MAX,
```

```
    assert(process("expert","SAW",E)),
```

```
    write("\n Expert System recommends SAW\n").
```

```
final_process("ALL",A,B,C,D,E,MAX) :-
```

```
    D = B,
```

```
    D = C,
```

```
    D = A,
```

```
    E=D/MAX,
```

```
    assert(process("expert","SMAW",E)),
```

```
    assert(process("expert","GMAW",E)),
```

```
    assert(process("expert","GTAW",E)),
```

```
    assert(process("expert","SAW",E)),
```

```
    write("\n Expert System recommends SMAW, GMAW, GTAW or SAW\n").
```

```
final_process("THREE",A,B,C,D,E,MAX) :-
```

```

D = B,
D = C,
D > A,
E=D/MAX,
assert(process("expert","GTAW",E)),
assert(process("expert","GMAW",E)),
assert(process("expert","SAW",E)),
write("\n Expert System recommends GTAW, GMAW or SAW\n").

```

```

final_process("THREE",A,B,C,D,E,MAX) :-
    D = B,
    D > C,
    D = A,
    E=D/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","GMAW",E)),
    assert(process("expert","SAW",E)),
    write("\n Expert System recommends SMAW, GMAW or SAW\n").

```

```

final_process("THREE",A,B,C,D,E,MAX) :-
    A = B,
    B = C,
    D < A,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","GMAW",E)),
    assert(process("expert","GTAW",E)),
    write("\n Expert System recommends SMAW, GMAW or GTAW\n").

```

```

final_process("THREE",A,B,C,D,E,MAX) :-
    A = C,
    D = C,
    D > B,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","SAW",E)),
    assert(process("expert","GTAW",E)),
    write("\n Expert System recommends SMAW, GTAW or SAW\n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    A = B,
    A > C,
    A > D,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","GMAW",E)),
    write("\n Expert System recommends SMAW or GMAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-

```

```

A = C,
A > B,
A > D,
E=A/MAX,
assert(process("expert","SMAW",E)),
assert(process("expert","GTAW",E)),
write("\n Expert System recommends SMAW or GTAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    A = D,
    A > B,
    A > C,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","SAW",E)),
    write("\n Expert System recommends SMAW or SAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    C = B,
    C > A,
    C > D,
    E=C/MAX,
    assert(process("expert","GTAW",E)),
    assert(process("expert","GMAW",E)),
    write("\n Expert System recommends GMAW or GTAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    B = D,
    B > A,
    B > C,
    E=B/MAX,
    assert(process("expert","GMAW",E)),
    assert(process("expert","SAW",E)),
    write("\n Expert System recommends GMAW or SAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    C = D,
    C > A,
    C > B,
    E=C/MAX,
    assert(process("expert","GTAW",E)),
    assert(process("expert","SAW",E)),
    write("\n Expert System recommends GTAW or SAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    A = B,
    A > C,
    A > D,
    E=A/MAX,

```

```

assert(process("expert","SMAW",E)),
assert(process("expert","GMAW",E)),
write("\n Expert System recommends SMAW or GMAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    A = B,
    A > C,
    A > D,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","GMAW",E)),
    write("\n Expert System recommends SMAW or GMAW \n").

```

```

final_process("TWO",A,B,C,D,E,MAX) :-
    A = B,
    A > C,
    A > D,
    E=A/MAX,
    assert(process("expert","SMAW",E)),
    assert(process("expert","GMAW",E)),
    write("\n Expert System recommends SMAW or GMAW \n").

```

```

consult_neural_net :-
    consult_usr_data,
    job_data(Parent_Material,Weld_type,Position,Seam_length,
        Plate_thickness,Strength,Appearance,Deposition,
        Weld_Quality,Automation,Leg_length),
    get_PM(Parent_Material,PM_net),
    get_WT(Weld_type,WT_net),
    get_POSI(Position,POS_net),
    get_APP(Appearance,APP_net),
    get_DEP(Deposition,DEP_net),
    get_WQ(Weld_Quality,WQ_net),
    get_AUT(Automation,AUT_net),
    Leg_length_crit(Leg_length,LL_Factor),
    Seam_length_crit(Seam_length,SL_Factor),
    Plate_thickness_crit(Plate_thickness,PT_Factor),
    Strength_crit(Strength,ST_Factor),
    get_strength(ST_Factor,ST),
    get_LL(LL_Factor,LL),
    get_PT(PT_Factor,PT),
    get_SL(SL_Factor,SL),
    openwrite (outfile,"procprun.in"),
    writedevise (outfile),
    write (PM_net," ",WT_net," ",POS_net," ",SL,
        " ",PT," ",ST," ",APP_net,
        " ",DEP_net," ",WQ_net," ",AUT_net," ",
        LL,""),nl,
    closefile (outfile).

```



```

system("testbp -R procprun"),
existfile("procprun.out"),!,
openread(infile,"procprun.out"),
readdevice(infile),
readln(X),
closefile(infile),
readdevice(Keyboard),
readchar(_),
fronttoken(X,SMAW,Rest1),
fronttoken(Rest1,GMAW,Rest2),
fronttoken(Rest2,GTAW,Rest3),
fronttoken(Rest3,SAW,Rest4),
str_real(SMAW,SMAW1),
str_real(GMAW,GMAW1),
str_real(GTAW,GTAW1),
str_real(SAW,SAW1),
readchar(_),
validate_SMAW(SMAW1,SMAW2),
%validate_GMAW(GMAW1),
%validate_gtaw(GTAW1),
validate_saw(SAW1,SAW2),
final_neural_process(X1,SMAW2,GMAW1,GTAW1,SAW2),
write("\n\nSMAW:",SMAW2),nl,
write("GMAW:",GMAW),nl,
write("GTAW:",GTAW),nl,
write("SAW: ",SAW2),nl,
compare_selections.

validate_SMAW (Points,New_Points) :-
    consult_usr_data,
    job_data(Parent_Material,Weld_type,Position,Seam_length,
    Plate_thickness,Strength,Appearance,Deposition,
    Weld_Quality,Automation,Leg_length),
    Automation= "NO",!,
    New_Points=Points.

validate_SMAW (Points,New_Points) :-
    job_data(Parent_Material,Weld_type,Position,Seam_length,
    Plate_thickness,Strength,Appearance,Deposition,
    Weld_Quality,Automation,Leg_Length),
    Automation="YES",
    New_Points=0.

validate_saw (Points,New_Points) :-
    consult_usr_data,
    job_data(Parent_Material,Weld_type,Position,Seam_length,
    Plate_thickness,Strength,Appearance,Deposition,
    Weld_Quality,Automation,Leg_length),
    Position = "FLAT",

```

```

    Plate_thickness >= 5,
    New_Points= Points.

validate_saw (Points,New_Points) :-
    job_data(Parent_Material,Weld_type,Position,Seam_length,
    Plate_thickness,Strength,Appearance,Deposition,
    Weld_Quality,Automation,Leg_length),
    Position <> "FLAT",
    Position <> "HORIZONTAL",
    New_Points=0,!.

validate_saw (Points,New_Points) :-
    job_data(Parent_Material,Weld_type,Position,Seam_length,
    Plate_thickness,Strength,Appearance,Deposition,
    Weld_Quality,Automation,Leg_length),
    Plate_thickness < 5,
    Weld_Type <> "FILLET",
    New_Points=0.

validate_saw (Points,New_Points) :-
    New_Points=Points.

final_neural_process("SMAW",A,B,C,D) :-
    A > 0.5,
    A >= B,
    A >= C,
    A >= D,
    assert(process("neural","SMAW",A)),
    write("\nNeural Network recommends SMAW\n").

final_neural_process("GMAW",A,B,C,D) :-
    B > 0.5,
    B >= A,
    B >= C,
    B >= D,
    assert(process("neural","GMAW",B)),
    write("\nNeural Network recommends GMAW\n").

final_neural_process("GTAW",A,B,C,D) :-
    C > 0.5,
    C >= B,
    C >= A,
    C >= D,
    assert(process("neural","GTAW",C)),
    write("\nNeural Network recommends GTAW\n").

final_neural_process("SAW",A,B,C,D) :-
    D >= B,

```

```

        D > 0.5,
        D >= C,
        D >= A,
        assert(process("neural","SAW",D)),
        write("\nNeural Network recommends SAW\n").

final_neural_process("NONE",A,B,C,D):-
    write("\nNeural Network recommends:\nDO NOT USE SMAW,GMAW, GTAW or
SAW"),nl,
    assert (process("neural","NONE",0)),
    readchar(_).

compare_selections :-
    process("expert",Y,E),
    process("neural",X1,F),
    X1 = Y,
    write("Both Expert System and Neural Network recommend ",X1),nl,
    write("Overall Recommendation: ",Y),nl,
    readchar(Reply),
    write("Press any key to continue ..."),
    readchar (_).

compare_selections :-
    process("expert",Y,E),
    process("neural",X1,F),
    X1 <> Y,
    E > F,
    write("Expert System recommends ",Y," with a Confidence Factor of ",E),nl,
    write("Neural Network recommends ",X1," with a Confidence Factor of ",F),nl,
    write("Overall Recommendation: ",Y),nl.

compare_selections.

retrain_neural_net (X):-
    consult_usr_data,
    job_data(Parent_Material,Weld_type,Position,Seam_length,
        Plate_thickness,Strength,Appearance,Deposition,
        Weld_Quality,Automation,Leg_length),
    get_PM(Parent_Material,PM_net),
    get_WT(Weld_type,WT_net),
    get_POSI(Position,POS_net),
    get_APP(Appearance,APP_net),
    get_DEP(Deposition,DEP_net),
    get_WQ(Weld_Quality,WQ_net),
    get_AUT(Automation,AUT_net),
    Leg_length_crit(Leg_length,LL_Factor),
    Seam_length_crit(Seam_length,SL_Factor),
    Plate_thickness_crit(Plate_thickness,PT_Factor),
    Strength_crit(Strength,ST_Factor),

```

```

get_strength(ST_Factor,ST),
get_LL(LL_Factor,LL),
get_PT(PT_Factor,PT),
get_SL(SL_Factor,SL),
get_process_param(X,X1),
openappend (outfile,"procprun.fct"),
writedevise (outfile),
write (PM_net," ",WT_net," ",POS_net," ",SL,
      " ",PT," ",ST," ",DEP_net," ",WQ_net," ",AUT_net," ",
      LL," ",X1," "),nl,
closefile (outfile),
write ("Training ....."),
system("testbp -L procprun").

get_process_param ("SMAW",X1) :-
    X1 = "1 0 0 0".

get_process_param ("GMAW",X1) :-
    X1 = "0 1 0 0".

get_process_param ("GTAW",X1) :-
    X1 = "0 0 1 0".

get_process_param ("SAW",X1) :-
    X1 = "0 0 0 1".

consult_weights:-
    !,existfile("weights.dba"),
    consult("weights.dba").
consult_weights.

get_PM("CAST IRON","0 0 1 ").
get_PM("STAINLESS STEEL","0 0 1 ").
get_PM("MILD STEEL","0 0 0 ").
get_PM("CARBON STEEL","0 1 0 ").
get_PM("ALUMINUM ALLOY","1 0 0 ").

get_WT("FILLET","0 1 ").
get_WT("BUTT","1 0 ").

get_POSI("FLAT","0 0 0 0 ").
get_POSI("HORIZONTAL","1 0 0 0 ").
get_POSI("VERTICAL","0 1 0 0 ").
get_POSI("DOWNWARD","0 0 1 0 ").
get_POSI("OVERHEAD","0 0 0 1 ").

get_APP("FLAT","1 0 0 ").
get_APP("CONCAVE","0 0 1 ").

```

get\_APP("CONVEX"," 0 1 0 ").

get\_DEP("HIGH"," 1 0 ").  
get\_DEP("MEDIUM"," 0 1 ").  
get\_DEP("LOW"," 0 0 ").

get\_strength("HIGH"," 1 ").  
get\_strength("LOW"," 0 ").  
get\_PT("HIGH"," 1 0 0 ").  
get\_PT("MEDIUM"," 0 1 0 ").  
get\_PT("LOW"," 0 0 1 ").  
get\_SL("HIGH"," 1 ").  
get\_SL("LOW"," 0 ").  
get\_LL("HIGH"," 1 0 ").  
get\_LL("LOW"," 0 1 ").

get\_WQ("HIGH"," 1 0 ").  
get\_WQ("MEDIUM"," 0 1 ").  
get\_WQ("LOW"," 0 0 ").

get\_AUT("YES"," 1 0 ").  
get\_AUT("NO"," 0 1 ").

Seam\_length\_crit (Seam\_Length,Factor) :-  
    Seam\_Length >= 10,  
    Factor = "HIGH".

Seam\_length\_crit (Seam\_Length,Factor) :-  
    Seam\_Length < 10,  
    Factor = "LOW".

Plate\_thickness\_crit (Plate\_thickness,Factor) :-  
    Plate\_thickness >= 10,  
    Factor = "HIGH".

Plate\_thickness\_crit (Plate\_thickness,Factor) :-  
    Plate\_thickness >= 5,  
    Plate\_thickness < 10,  
    Factor = "MEDIUM".

Plate\_thickness\_crit (Plate\_thickness,Factor) :-  
    Plate\_thickness < 5,  
    Factor = "LOW".

Strength\_crit (Strength,Factor) :-  
    Strength <= 70000,  
    Factor = "LOW".

Strength\_crit (Strength,Factor) :-

```

        Strength > 70000,
        Factor = "HIGH".

Leg_length_crit (Leg_length,Factor):-
    Leg_length >= 10,
    Factor = "HIGH".

Leg_length_crit (Leg_length,Factor):-
    Leg_length < 10,
    Factor = "LOW".

consult_usr_data :-
    existfile("JOB_DATA.DBA"),!,
    consult("JOB_DATA.DBA").
consult_usr_data.

consult_criteria :-
    existfile("criteria.db"),!,
    consult("criteria.db",criteria).
consult_criteria.

goal
    process_select.

E.2 PAR_SEL.PRO - Program for Selecting Process Parameters
project "neurex"

include "global.pro"

/* Code for Parameter Selection by Expert Select */

/* std_data files :
    SMAW_std.db
        format: SMAW_std(electrode,weld_type,leg_length,position,
            runs,elec_size,current,weight,num_elec,Overhead_time,
            Arc_time)

    GMAW_std.db
        format: GMAW_std(weld_type,leg_length,position,
            runs,elec_size,current,weight,num_elec,Overhead_time,
            Arc_time)

    gtaw_std.db
        format: gtaw_std(weld_type,leg_length,position,runs,elec_size,
            current,weight,num_elec,Overhead_time,Arc_time)

```

```

saw_std.dba
    format: saw_std(weld_type,leg_length,position,runs,elec_size,
        current,weight,num_elec,Overhead_time,Arc_time)
*/

/* Par_Selection for SMAW */
/*
database - std_data2
    SMAW_data2(string,real,string,real,real,real,real,real,real,real,string)
    GMAW_data2(string,real,string,real,real,real,real,real,real,real,string)
    gtaw_data2(string,real,string,real,real,real,real,real,real,real,real,string)
    saw_data2(string,real,string,real,real,real,real,real,real,real,real,string)

database - job_data1
    job_data(string,string,string,real,real,real,string,string,string,string,real)
*/
database - temp
    electrode_selected(string)

database - electrode
    electrode(string,string,string,string,string,string)

database - result
    SMAW_par(string,real,string,real,real,real,real,real,real,real,string)
    GMAW_par(string,real,string,real,real,real,real,real,real,real,real,string)

predicates
    Select_SMAW_electrode
    Select_SMAW_parameters(string,string,real,string,integer)
    Select_GMAW_parameters(string,real,string,integer)
    consult_job_data
    consult_std_data
    consult_par_data
    consult_SMAW_electrode_data
    consult_GMAW_electrode_data
    Get_Strength_value(string,real)
    weld_type(string,string)
    position_check(string,string)
    appearance_check(string,string)
    parameter_selection

clauses
    parameter_selection:-
        consult_job_data,
        makewindow(1,112,112,"Process Parameters",0,0,25,80),nl,
        write("JOB VARIABLES"),nl,

```

```

write("====="),nl,
job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,
        Appearance,Dep_Rate,Weld_Qual,Automation,Leg_length),
write("Parent Material: ",Parent_Mat),
write("\tWeld Type: ",Weld_Type),
write("\tPosition: ",Position),nl,
write("Seam Length: ",Seam,"m"),
write("\tPlate Thick.: ",Plate,"mm"),
write("\tStrength: ",Stren,"psi"),
write("\tAppear.: ",Appearance),nl,
write("Leg Length: ",Leg_length),
write("\tDep. Rate: ",Dep_Rate),
write("\tQuality: ",Weld_Qual),
write("\tAutomation: ",Automation),
consult_job_data,
consult_SMAW_electrode_data,
consult_std_data,
par_sel_SMAW,
par_sel_GMAW,
write("Press any key to continue..."),
readchar(_).

```

Select\_SMAW\_electrode:-

```

job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,Appearance,_,_,_,_),
Get_Strength_value(Strength,Stren),
electrode(X,Parent_Mat,Weld_Type1,Position1,Strength,Appearance1),
position_check(Position1,Position),
weld_type(Weld_Type1,Weld_Type),
appearance_check(Appearance1,Appearance),
assert(electrode_selected(X)).

```

```

appearance_check("FLAT/CONCAVE","CONCAVE").
appearance_check("FLAT/CONVEX","CONVEX").
appearance_check("FLAT/CONCAVE","FLAT").
appearance_check("FLAT/CONVEX","FLAT").
appearance_check("CONCAVE","CONCAVE").
appearance_check("CONVEX","CONVEX").
appearance_check("FLAT","FLAT").
position_check("ALL","FLAT").
position_check("ALL","HORIZONTAL").
position_check("FH","FLAT").
position_check("FH","HORIZONTAL").
position_check("ALL","VERTICAL").
position_check("ALL","DOWNWARD").
position_check("ALL","OVERHEAD").
position_check("FLAT","FLAT").
position_check("HORIZONTAL","HORIZONTAL").
position_check("VERTICAL","VERTICAL").
position_check("DOWNWARD","DOWNWARD").

```



```
position_check("OVERHEAD","OVERHEAD").
```

```
weld_type("ALL","BUTT").  
weld_type("ALL","FILLET").  
weld_type("BUTT","BUTT").  
weld_type("FILLET","FILLET").
```

```
Get_Strength_value (Strength,Stren) :-  
    Stren <= 60000,  
    Strength = "LOW".
```

```
Get_Strength_value (Strength,Stren):-  
    Stren > 60000,  
    Stren <=70000,  
    Strength = "MEDIUM".
```

```
Get_Strength_value (Strength,Stren):-  
    Stren >= 70000,  
    Strength = "HIGH".
```

```
consult_par_data :-  
    existfile("par_sel.txt"),!,  
    consult("par_sel.txt",result).  
consult_par_data.
```

```
consult_SMAW_electrode_data :-  
    existfile("electrod.db"),!,  
    consult("electrod.db",electrode).  
consult_SMAW_electrode_data.
```

```
consult_GMAW_electrode_data :-  
    existfile("electrod.db"),!,  
    consult("electrod.db",electrode).  
consult_GMAW_electrode_data.
```

```
consult_std_data:-  
    existfile("std_data.txt"),!,  
    consult("std_data.txt").  
consult_std_data.
```

```
Par_sel_SMAW :-  
    write("\n\nSMAW PARAMETERS"),nl,  
    write("====="),nl,  
    Select_SMAW_electrode,  
    job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,  
        Appearance,_,_,Leg_length),  
    electrode_selected(Electrode),
```

```

Counter = 1,
Select_SMAW_parameters(Electrode,Weld_type,Leg_length,
                        Position,Counter),
save("par_sel.dba".result).
Par_sel_SMAW.

Par_sel_GMAW :-
write("\n\nGMAW PARAMETERS "),nl,
write("====="),nl,
job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,
        Appearance,_,_,_,Leg_length),
Counter = 1,
Select_GMAW_parameters(Weld_type,Leg_length,Position,Counter),
save("par_sel.dba".result).
Par_sel_GMAW.

Select_SMAW_parameters (Electrode,Weld_type,Leg_length,Position,Counter):-
SMAW_data(Weld_type,Leg_length,Position,Counter,Elec_size,
        Current,Num_elec,Weight,Arc_time,
        Overhead_time,Electrode),
write("Run Number:",Counter),
write(" Electrode Type: ",Electrode),
write(" Electrode Size: ",Elec_size," mm"),
write(" Current: ",Current," Amp").nl,
readchar(_),
assert(SMAW_par(Weld_type,Leg_length,Position,Counter,
        Elec_size,Current,Num_elec,Weight,
        Arc_time,Overhead_time,Electrode)),
Counter1 = Counter + 1,
Select_SMAW_parameters (Electrode,Weld_type,Leg_length,Position,Counter1).
Select_SMAW_parameters (_,_,_,_,_).

Select_GMAW_parameters (Weld_type,Leg_length,Position,Counter) :-
%consult_std_data,
GMAW_data(Weld_type,Leg_length,Position,Counter,Elec_size,
        Wt,Voltage,Wire_Feed,Current,Num_elec,Speed,
        Arc_time,S_Gas),
write("Run Number : ",Counter),
write("\tS_Gas: ",S_GAS),
write("\tElec. Dia.: ",Elec_size,"mm"),
write("\tCurrent: ",Current," Amp").nl,
write("\tVoltage: ",Voltage," V"),
write("\tWire Feed: ",Wire_Feed," m/hr"),
write("\tSpeed:",Speed," m/hr").nl,
readchar(_),
assert(GMAW_par(Weld_type,Leg_length,Position,Counter,
        Elec_size,Wt,Voltage,Wire_Feed,Current,Num_elec,
        Speed,Arc_time,S_Gas)),

```

```

        Counter1=Counter+1,
        Select_GMAW_parameters (Weld_Type,Leg_length,Position,Counter1).
select_GMAW_parameters (_,_,_,_).

consult_job_data:-
    existfile("job_data.db"),!,
    consult("job_data.db").
consult_job_data.

```

### **E.3 ECONEVAL.PRO - Prolog Program for Economic Evaluation of SMAW and GMAW**

project "neurex"

```

include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu2.pro"

database - finproc
    process(String,string,Real)

database - wps
    wps_process(string,string)

database - econeval
    econeval(string,real,real,real,real)
    compare(string,string,real,real)

database - par_sel
    SMAW_par(string,real,string,real,real,real,real,real,real,string)
    GMAW_par(string,real,string,real,real,real,real,real,real,real,real,string)

database - firm_data
    firm_machine_data(string,real)
    firm_labor_data(real,real)
    elect_dat(string,real)
    GMAW_elect_dat(string,real,real)

predicates
    consult_proc_data
    consult_par_data
    consult_job_data
    consult_firm_data
    consult_elec_data
    econ_eval_gen1 (string)
    compare_cost
    compare_cost1
    compare_cost2

```

```

compare_cost3
final_result(real,real,real,real,real,real)
process_reply(char)

clauses
econ_eval_gen :-
    econ_eval_gen1 ("SMAW"),
    econ_eval_gen1 ("GMAW"),
    consult_proc_data,
    econeval("SMAW",C1,T1,N1,W1),
    econeval("GMAW",C2,T2,N2,W2),
    write ("Economic Evaluation"),nl,
    write ("====="),nl,
    write("\t\tSMAW\t\tGMAW"),nl,
    write("\t\t=====\t\t====="),nl,
    write("Cost:\t\t",C1," \t\t",C2),nl,
    write("Time:\t\t",T1," \t\t",T2),nl,
    write("Wt. Deposited (Kg):\t",W1," \t\t",W2),nl,
    write("Electrodes Consumed:\t",N1," nos \t",N2," meters"),nl,
    final_result(C1,C2,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank),
    write("Do you wish to override? (y/n):"),
    readchar(Reply),
    process_reply(Reply),
    save("wps.db",wps),
    readchar(_),
    closefile(outfile).

process_reply('n').
process_reply('y'):-
    write("\nPlease enter your choice (SMAW/GMAW)?"),
    readln(Choice),
    retractall(wps_process(_,_)),
    assert(wps_process("user selection",Choice)).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    SMAW_Cost > GMAW_Cost,
    write("\nGMAW is recommended because of Lower Cost\n"),
    assert(wps_process("expert network","GMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    GMAW_Cost > SMAW_Cost,
    write("\nSMAW is recommended because of Lower Cost\n"),
    assert(wps_process("expert network","SMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    SMAW_Cost > GMAW_Cost,
    GMAW_Rank >= SMAW_NRank,
    write("\nGMAW is recommended because of Lower Cost and higher confidence\n"),

```

```

assert(wps_process("expert","GMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    GMAW_Cost > SMAW_Cost,
    write("\nSMAW is recommended because of Lower Cost and higher confidence\n"),
    assert(wps_process("expert","SMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    SMAW_Cost > GMAW_Cost,
    write("\nSMAW is recommended as Neural Network has higher confidence in it"),
    assert(wps_process("expert network","SMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    GMAW_Cost > SMAW_Cost,
    %GMAW_Rank >= SMAW_Rank,
    write("\nGMAW is recommended because Neural Network has higher confidence in it\n"),
    assert(wps_process("expert network","GMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    SMAW_Cost > GMAW_Cost,
    %GMAW_Rank < SMAW_NRank,
    %SMAW_Rank < GMAW_Rank,
    write("\nSMAW is recommended as Neural Network has higher confidence in it"),
    assert(wps_process("neural","SMAW")).

final_result (SMAW_Cost,GMAW_Cost,SMAW_NRank,GMAW_NRank,SMAW_Rank,GMAW_Rank):-
    GMAW_Cost > SMAW_Cost,
    write("\nGMAW is recommended because Neural Network has higher confidence in it\n"),
    assert(wps_process("neural","GMAW")).

compare_cost:-
    process("neural","GMAW",Ranking1),
    econeval("GMAW",C,T,_),
    assert(compare("neural","GMAW",C,Ranking1)),
    compare_cost.

compare_cost1 :-
    process("neural","SMAW",Ranking1),
    econeval("SMAW",C,T,_),
    assert(compare("neural","SMAW",C,Ranking1)),
    compare_cost1.

compare_cost2 :-
    process("expert","GMAW",Ranking1),
    econeval("GMAW",C,T,_),
    assert(compare("expert","GMAW",C,Ranking1)),
    compare_cost2.

compare_cost3 :-

```

```

process("expert","SMAW",Ranking1),
econeval(Process,C,T,_.),
assert(compare("expert",Process,C,Ranking1)).
compare_cost3.

econ_eval_gen1 ("SMAW") :-
    consult_par_data,
    %consult_proc_data,
    consult_job_data,
    consult_firm_data,
    job_data(PARENT_MAT,WELD_TYPE,POSITION,Seam,Plate,Stren,
        APPEARANCE,DEP_RATE,WELD_QUAL,AUTOMATION,LEG_L),
    SMAW_par(Weld_type,Leg_length,Position,l,
        Elec_size,Current,Num_elec,Weight,
        Arc_time,Overhead_time,Electrode),
    firm_labor_data(LC,OH),
    elect_dat (Electrode,CE),
    makewindow(1,112,112,"Economic Evaluation",0,0,25,80),nl,
    T = Seam * (Overhead_time+Arc_time),
    N = Num_elec * Seam,
    C = ((T * LC + OH * T )/60)+ N * CE ,
    W2 = Seam * Weight,
    write ("Job Variables"),nl,
    write ("====="),nl,
    write("Position: ",POSITION),
    write(" Joint Type: ",Weld_type),
    write(" Plate thickness: ",Plate," mm"),nl,
    write("Leg Length: ",Leg_L," mm"),
    write(" Seam Length: ",Seam," meters"),nl,nl,
    assert(econeval("SMAW",C,T,N,W2)).

econ_eval_gen1 ("GMAW") :-
    consult_par_data,
    consult_proc_data,
    consult_job_data,
    job_data(PARENT_MAT,WELD_TYPE,POSITION,Seam,Plate,Stren,
        APPEARANCE,DEP_RATE,WELD_QUAL,AUTOMATION,LEG_L),
    GMAW_par(Weld_type,Leg_length,Position,l,
        Elec_size,Wt,Voltage,Wire_Feed,Current,Num_elec,
        Speed,Arc_time,S_Gas),nl,
    firm_labor_data(LC,OH),
    GMAW_elect_dat (Parent_Mat,Elec_size,CE),
    T = Seam * (Arc_time),
    N = Num_elec * Seam,
    C = ((T * LC + OH * T )/60)+ N * CE ,
    W2 = Seam * Wt,
    assert(econeval("GMAW",C,T,N,W2)),
    readchar(_).

```

```
econ_eval_gen1 (_).
```

```
consult_firm_data :-  
    existfile("FIRM_DAT.DBA"),!,  
    consult("FIRM_DAT.DBA",firm_data).  
consult_firm_data.
```

```
consult_proc_data :-  
    existfile("process.dba"),!,  
    consult("process.dba",finproc).  
consult_proc_data.
```

```
consult_par_data :-  
    existfile("par_sel.dba"),!,  
    consult("par_sel.dba",par_sel).  
consult_par_data.
```

```
consult_job_data :-  
    existfile("job_DATA.DBA"),!,  
    consult("job_DATA.DBA").  
consult_job_data.
```

```
consult_elec_data :-  
    existfile("ELEC_DAT.DBA"),!,  
    consult("ELEC_DAT.DBA").  
consult_elec_data.
```

#### E.4 ADJUSTEN.PRO - Prolog Program for Knowledge Correction Task in Expert Network

```
project "neurex"

include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu.pro"

database - criteria
    Automation_criteria (string,real,real,real,real)
    Plate_thickness_criteria (string,real,real,real,real)
    Strength_criteria (string,real,real,real,real)
    Seam_length_criteria(string,real,real,real,real)
    Leg_length_criteria(string,real,real,real,real)
    Parent_material_criteria (string,real,real,real,real)
    Weld_type_criteria (string,real,real,real,real)
    Position_criteria (string,real,real,real,real)
    Appearance_criteria (string,real,real,real,real)
    Deposition_criteria (string,real,real,real,real)
    Weld_quality_criteria (string,real,real,real,real)

database - process
    process(string,string,real)

database - wps
    wps_process(string,string)

predicates
    consult_usr_data
    consult_process_selected
    get_PM(string,string)
    get_WT(string,string)
    get_POSI(string,string)
    get_APP(string,string)
    get_DEP(string,string)
    get_WQ(string,string)
    get_AUT(string,string)
    get_strength(string,string)
    get_LL(string,string)
    get_PT(string,string)
    get_SL(string,string)
    consult_criteria
    consult_weights
    Seam_length_crit (real,string)
    Plate_thickness_crit (real,string)
    Strength_crit (real,string)
    Leg_length_crit (real,string)
    adjust_knowledge
    change_PM_criteria(string)
    change_WT_criteria(string)
    change_POS_criteria(string)
    change_AUT_criteria(string)
    change_WQ_criteria(string)
    change_APP_criteria(string)
    change_DEP_criteria(string)
    change_Strength_criteria(string)
    change_Seam_Length_criteria(string)
```



```

change_Plate_thickness_criteria(string)
change_Leg_Length_criteria(string)
Get_Adjustment(real,real,real,real)
get_process_param (string, string)
retrain_neural_net (string)
consult_wps_db
check_need_for_adjustment
adjust_expert_system_criteria
store_knowledge(string,string,string)
check_minus(string,string,string,string)

clauses

/* Clauses for the KNOWLEDGE ADJUSTMENT MODULE */

adjust_knowledge:-
    consult_wps_db,
    wps_process(Method,Process),
    check_need_for_adjustment.

check_need_for_adjustment:-
    wps_process("expert network",Process),
    write("No need for adjustment as both Neural Network"),nl,
    write("and Expert System recommended ",Process),!.

check_need_for_adjustment:-
    wps_process("expert",Process),
    store_net_weights,
    retrain_neural_net (Process).

check_need_for_adjustment:-
    wps_process("neural",Process),
    adjust_expert_system_criteria.

store_net_weights:-
    makewindow(91,15,24,"Results",0,0,25,80),
    existfile("process.mat"),!,
    openread(infile,"process.mat"),
    readdevice(infile),
    readln(X),
    readln(X1),
    readln(X2),
    readln(X3),
    readln(X4),
    readln(X5),
    readln(X6),
    readln(X7),
    readln(X8),
    readln(X9),
    readln(X10),
    readln(X11),
    readln(X12),
    readln(X13),
    readln(X14),
    readln(X15),
    readln(X16),
    readln(X17),

```

```

readln(X18),
readln(X19),
readln(X20),
readln(X21),
readln(X22),
readln(X23),
readln(X24),
readln(X25),
readln(X26),
readln(X27),
readln(X28),
readln(X29),
readln(X30),
readln(X31),
readln(X32),
readln(X33),
readln(X34),
readln(X35),
closefile(infile),
store_knowledge("PM","ALUMINUM",X1),
store_knowledge("PM","MILD STEEL",X2),
store_knowledge("PM","CARBON STEEL",X3),
store_knowledge("PM","STAINLESS STEEL",X4),
store_knowledge("PM","CAST IRON",X5),
store_knowledge("Weld_Type","BUTT",X6),
store_knowledge("Weld_Type","FILLET",X7),
store_knowledge("Position","FLAT",X8),
store_knowledge("Position","HORIZONTAL",X9),
store_knowledge("Position","VERTICAL",X10),
store_knowledge("Position","DOWNWARD",X11),
store_knowledge("Position","OVERHEAD",X12),
store_knowledge("Seam_Length","HIGH",X13),
store_knowledge("Seam_Length","LOW",X14),
store_knowledge("Plate_Thickness","HIGH",X15),
store_knowledge("Plate_Thickness","MEDIUM",X16),
store_knowledge("Plate_Thickness","LOW",X17),
store_knowledge("Strength","HIGH",X18),
store_knowledge("Strength","LOW",X19),
store_knowledge("Appearance","FLAT",X20),
store_knowledge("Appearance","CONVEX",X21),
store_knowledge("Appearance","CONCAVE",X22),
store_knowledge("Deposition","LOW",X23),
store_knowledge("Deposition","MEDIUM",X24),
store_knowledge("Deposition","HIGH",X25),
store_knowledge("Weld_Quality","LOW",X26),
store_knowledge("Weld_Quality","MEDIUM",X27),
store_knowledge("Weld_Quality","HIGH",X28),
store_knowledge("Automation","YES",X29),
store_knowledge("Automation","NO",X30),
store_knowledge("Leg_Length","LOW",X31),
store_knowledge("Leg_Length","HIGH",X32),
save("weights.dba").

store_knowledge(Attrib,PM,X):-
    fronttoken(X,SMAW_PM1,Rest1),
    check_minus(SMAW_PM1,SMAW_FINAL,Rest1,Rest1_1),
    write("\n\nSMAW Weight:",SMAW_FINAL),nl,
    fronttoken(Rest1_1,GMAW_PM1,Rest2),

```

```

        check_minus (GMAW_PM1,GMAW_FINAL,Rest2,Rest2_1),
        write("\n\nGMAW Weight:",GMAW_FINAL),
        fronttoken(Rest2_1,GTAW_PM1,Rest3),
        check_minus (GTAW_PM1,GTAW_FINAL,Rest3,Rest3_1),
        write("\n\nGTAW Weight:",GTAW_FINAL),nl,
        fronttoken(Rest3_1,SAW_PM1,Rest4),
        check_minus (SAW_PM1,SAW_FINAL,Rest4,Rest4_1),
        write("\n\nSAW Weight:",SAW_FINAL),
        str_real (SMAW_FINAL,SMAW_PM),
        str_real (GMAW_FINAL,GMAW_PM),
        str_real (GTAW_FINAL,GTAW_PM),
        str_real (SAW_FINAL,SAW_PM),

assert (process_knowledge (Attrib,PM,SMAW_PM,GMAW_PM,GTAW_PM,SAW_PM) )

        readchar(_).

check_minus ("-",FINAL,Rest1,Rest2):-
        fronttoken(Rest1,X1,Y1),
        concat("-",X1,Z1),
        FINAL=Z1,
        Rest2=Y1.

check_minus (X,FINAL,Rest1,Rest2):-
        Rest2=Rest1,
        FINAL=X.

retrain_neural_net (X):-
        consult_usr_data,
        job_data(Parent_Material,Weld_type,Position,Seam_length,
        Plate_thickness,Strength,Appearance,Deposition,
        Weld_Quality,Automation,Leg_length),

get_PM(Parent_Material,PM_net),
get_WT(Weld_type,WT_net),
get_POSI(Position,POS_net),
get_APP(Appearance,APP_net),
get_DEP(Deposition,DEP_net),
get_WQ(Weld_Quality,WQ_net),
get_AUT(Automation,AUT_net),
Leg_length_crit(Leg_length,LL_Factor),
Seam_length_crit(Seam_length,SL_Factor),
Plate_thickness_crit(Plate_thickness,PT_Factor),
Strength_crit(Strength,ST_Factor),
get_strength(ST_Factor,ST),
get_LL(LL_Factor,LL),
get_PT(PT_Factor,PT),
get_SL(SL_Factor,SL),
get_process_param(X,X1),
openappend (outfile,"proc_sel.fct"),
writedevise (outfile),
write (PM_net," ",WT_net," ",POS_net," ",SL,
        " ",PT," ",ST," ",APP_net,
        " ",DEP_net," ",WQ_net," ",AUT_net," ",
        LL," ",X1,""),nl,
closefile (outfile),
write ("Sleeping ....."),
system("testbp -L proc_sel").

```

```

get_process_param ("SMAW",X1) :-
    X1 = "1 0 0 0 ".

get_process_param ("GMAW",X1) :-
    X1 = "0 1 0 0 ".

get_process_param ("GTAW",X1) :-
    X1 = "0 0 1 0 ".

get_process_param ("SAW",X1) :-
    X1 = "0 0 0 1 ".

consult_weights:-
    !,existfile("weights.dba"),
    consult("weights.dba").
consult_weights.

adjust_expert_system_criteria :-
    job_data(Parent_Material,Weld_type,Position,Seam_length,
        Plate_thickness,Strength,Appearance,Deposition,
        Weld_Quality,Automation,Leg_length),
    %expert_points(MAX1,X1,X2,X3,X4),
    consult_weights,
    change_PM_criteria(Parent_Material),
    change_WT_criteria(Weld_type),
    change_POS_criteria(Position),
    change_AUT_criteria(Automation),
    change_WQ_criteria(Weld_Quality),
    change_APP_criteria(Appearance),
    change_DEP_criteria(Deposition),
    Strength_crit (Strength,Factor6),
    Leg_length_crit (Leg_length,Factor11),
    Plate_thickness_crit (Plate_thickness,Factor5),
    Seam_length_crit (Seam_Length,Factor10),
    change_Strength_criteria(Factor6),
    change_Seam_Length_criteria(Factor10),
    change_Plate_thickness_criteria(Factor5),
    change_Leg_Length_criteria(Factor11),
    !,
    save("criteria.dba",criteria).

change_PM_criteria(Parent_Material):-
    retract(parent_material_criteria(Parent_Material,SMAW_PM,
        GMAW_PM,GTAW_PM,SAW_PM)),

process_knowledge("PM",Parent_Material,SMAW_Weight,GMAW_Weight,
    GTAW_Weight,SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_PM),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_PM),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_PM),
    SAW_DIFF=abs(SAW_Weight-SAW_PM),
    Get_Adjustment(SMAW_DIFF,SMAW_Weight,SMAW_PM,SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF,GMAW_Weight,GMAW_PM,GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_PM,GTAW_Adjust),
    Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_PM,SAW_Adjust),

```

```

        assert(parent_material_criteria(Parent_Material, SMAW_Adjust,
                                        GMAW_Adjust, GTAW_Adjust, SAW_Adjust)).

Get_Adjustment(DIFF, NN_Weight, CRIT_WEIGHT, ADJUSTED_WEIGHT) :-
    DIFF >= abs(CRIT_WEIGHT),
    ADJUSTED_WEIGHT = NN_Weight.

Get_Adjustment(DIFF, NN_Weight, CRIT_WEIGHT, ADJUSTED_WEIGHT) :-
    DIFF < abs(CRIT_WEIGHT),
    ADJUSTED_WEIGHT = CRIT_WEIGHT.

change_WT_criteria(Weld_type) :-

retract(weld_type_criteria(Weld_type, SMAW_WT, GMAW_WT, GTAW_WT, SAW_WT
)),
    process_knowledge("Weld_Type", Weld_type, SMAW_Weight,
                     GMAW_Weight, GTAW_Weight, SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_WT),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_WT),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_WT),
    SAW_DIFF=abs(SAW_Weight-SAW_WT),
    Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_WT, SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_WT, GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF, GTAW_Weight, GTAW_WT, GTAW_Adjust),
    Get_Adjustment(SAW_DIFF, SAW_Weight, SAW_WT, SAW_Adjust),
    assert(weld_type_criteria(Weld_type, SMAW_Adjust, GMAW_Adjust,
                             GTAW_Adjust, SAW_Adjust)).

change_POS_criteria(Position) :-

retract(position_criteria(Position, SMAW_POS, GMAW_POS, GTAW_POS, SAW_P
OS)),
    process_knowledge("Position", Position, SMAW_Weight, GMAW_Weight,
                     GTAW_Weight, SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_POS),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_POS),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_POS),
    SAW_DIFF=abs(SAW_Weight-SAW_POS),
    Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_POS, SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_POS, GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF, GTAW_Weight, GTAW_POS, GTAW_Adjust),
    Get_Adjustment(SAW_DIFF, SAW_Weight, SAW_POS, SAW_Adjust),
    assert(position_criteria(Position, SMAW_Adjust, GMAW_Adjust,
                             GTAW_Adjust, SAW_Adjust)).

change_AUT_criteria(Automation) :-
    retract(automation_criteria(Automation, SMAW_Aut, GMAW_Aut,
                                GTAW_Aut, SAW_Aut)),

process_knowledge("Automation", Automation, SMAW_Weight, GMAW_Weight,
                 GTAW_Weight, SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_Aut),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_Aut),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_Aut),
    SAW_DIFF=abs(SAW_Weight-SAW_Aut),
    Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_Aut, SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_Aut, GMAW_Adjust),

```

```

        Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_Aut,GTAW_Adjust),
        Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_Aut,SAW_Adjust),

assert(automation_criteria(Automation,SMAW_Adjust,GMAW_Adjust,
        GTAW_Adjust,SAW_Adjust)).

change_WQ_criteria(Weld_Quality):-
    retract(weld_quality_criteria(Weld_Quality,SMAW_WQ,GMAW_WQ,
        GTAW_WQ,SAW_WQ)),
    process_knowledge("Weld_Quality",Weld_Quality,SMAW_Weight,
GMAW_Weight,GTAW_Weight,SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_WQ),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_WQ),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_WQ),
    SAW_DIFF=abs(SAW_Weight-SAW_WQ),
    Get_Adjustment(SMAW_DIFF,SMAW_Weight,SMAW_WQ,SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF,GMAW_Weight,GMAW_WQ,GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_WQ,GTAW_Adjust),
    Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_WQ,SAW_Adjust),

assert(weld_quality_criteria(Weld_Quality,SMAW_Adjust,GMAW_Adjust,
        GTAW_Adjust,SAW_Adjust)).

change_APP_criteria(Appearance):-
    retract(appearance_criteria(Appearance,SMAW_APP,GMAW_APP,
        GTAW_APP,SAW_APP)),

process_knowledge("Appearance",Weld_type,SMAW_Weight,GMAW_Weight,
        GTAW_Weight,SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_APP),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_APP),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_APP),
    SAW_DIFF=abs(SAW_Weight-SAW_APP),
    Get_Adjustment(SMAW_DIFF,SMAW_Weight,SMAW_APP,SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF,GMAW_Weight,GMAW_APP,GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_APP,GTAW_Adjust),
    Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_APP,SAW_Adjust),

assert(appearance_criteria(Appearance,SMAW_Adjust,GMAW_Adjust,
        GTAW_Adjust,SAW_Adjust)).

change_DEP_criteria(Deposition):-
    retract(deposition_criteria(Deposition,SMAW_DEP,GMAW_DEP,
        GTAW_DEP,SAW_DEP)),

process_knowledge("Deposition",Deposition,SMAW_Weight,GMAW_Weight,
        GTAW_Weight,SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_DEP),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_DEP),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_DEP),
    SAW_DIFF=abs(SAW_Weight-SAW_DEP),
    Get_Adjustment(SMAW_DIFF,SMAW_Weight,SMAW_DEP,SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF,GMAW_Weight,GMAW_DEP,GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_DEP,GTAW_Adjust),
    Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_DEP,SAW_Adjust),

```

```

assert(deposition_criteria(Deposition, SMAW_Adjust, GMAW_Adjust,
                           GTAW_Adjust, SAW_Adjust)).

change_Strength_criteria(Strength) :-
retract(strength_criteria(Strength, SMAW_DEP, GMAW_DEP, GTAW_DEP, SAW_D
EP)),

process_knowledge("Strength", Strength, SMAW_Weight, GMAW_Weight,
                  GTAW_Weight, SAW_Weight),
SMAW_DIFF=abs(SMAW_Weight-SMAW_DEP),
GMAW_DIFF=abs(GMAW_Weight-GMAW_DEP),
GTAW_DIFF=abs(GTAW_Weight-GTAW_DEP),
SAW_DIFF=abs(SAW_Weight-SAW_DEP),
Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_DEP, SMAW_Adjust),
Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_DEP, GMAW_Adjust),
Get_Adjustment(GTAW_DIFF, GTAW_Weight, GTAW_DEP, GTAW_Adjust),
Get_Adjustment(SAW_DIFF, SAW_Weight, SAW_DEP, SAW_Adjust),
assert(strength_criteria(Strength, SMAW_Adjust, GMAW_Adjust,
                           GTAW_Adjust, SAW_Adjust)).

change_Seam_Length_criteria(Seam_Length) :-
retract(seam_length_criteria(Seam_Length, SMAW_DEP, GMAW_DEP,
                              GTAW_DEP, SAW_DEP)),

process_knowledge("Seam_Length", Seam_Length, SMAW_Weight, GMAW_Weight
                  GTAW_Weight, SAW_Weight),
SMAW_DIFF=abs(SMAW_Weight-SMAW_DEP),
GMAW_DIFF=abs(GMAW_Weight-GMAW_DEP),
GTAW_DIFF=abs(GTAW_Weight-GTAW_DEP),
SAW_DIFF=abs(SAW_Weight-SAW_DEP),
Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_DEP, SMAW_Adjust),
Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_DEP, GMAW_Adjust),
Get_Adjustment(GTAW_DIFF, GTAW_Weight, GTAW_DEP, GTAW_Adjust),
Get_Adjustment(SAW_DIFF, SAW_Weight, SAW_DEP, SAW_Adjust),

assert(seam_length_criteria(Seam_Length, SMAW_Adjust, GMAW_Adjust,
                              GTAW_Adjust, SAW_Adjust)).

change_Plate_thickness_criteria(Plate_thickness) :-
retract(plate_thickness_criteria(Plate_thickness, SMAW_DEP,
                                  GMAW_DEP, GTAW_DEP, SAW_DEP)),

process_knowledge("Plate_Thickness", Plate_thickness, SMAW_Weight,
                  GMAW_Weight, GTAW_Weight, SAW_Weight),
SMAW_DIFF=abs(SMAW_Weight-SMAW_DEP),
GMAW_DIFF=abs(GMAW_Weight-GMAW_DEP),
GTAW_DIFF=abs(GTAW_Weight-GTAW_DEP),
SAW_DIFF=abs(SAW_Weight-SAW_DEP),
Get_Adjustment(SMAW_DIFF, SMAW_Weight, SMAW_DEP, SMAW_Adjust),
Get_Adjustment(GMAW_DIFF, GMAW_Weight, GMAW_DEP, GMAW_Adjust),
Get_Adjustment(GTAW_DIFF, GTAW_Weight, GTAW_DEP, GTAW_Adjust),
Get_Adjustment(SAW_DIFF, SAW_Weight, SAW_DEP, SAW_Adjust),

assert(plate_thickness_criteria(Plate_thickness, SMAW_Adjust, GMAW_Ad
just,                               GTAW_Adjust, SAW_Adjust)).

```

```

change_Leg_Length_criteria(Leg_length):-
    retract(leg_length_criteria(Leg_length,SMAW_DEP,GMAW_DEP,
                                GTAW_DEP,SAW_DEP)),
    process_knowledge("Leg_Length",Leg_length,SMAW_Weight,
                     GMAW_Weight,GTAW_Weight,SAW_Weight),
    SMAW_DIFF=abs(SMAW_Weight-SMAW_DEP),
    GMAW_DIFF=abs(GMAW_Weight-GMAW_DEP),
    GTAW_DIFF=abs(GTAW_Weight-GTAW_DEP),
    SAW_DIFF=abs(SAW_Weight-SAW_DEP),
    Get_Adjustment(SMAW_DIFF,SMAW_Weight,SMAW_DEP,SMAW_Adjust),
    Get_Adjustment(GMAW_DIFF,GMAW_Weight,GMAW_DEP,GMAW_Adjust),
    Get_Adjustment(GTAW_DIFF,GTAW_Weight,GTAW_DEP,GTAW_Adjust),
    Get_Adjustment(SAW_DIFF,SAW_Weight,SAW_DEP,SAW_Adjust),

assert(leg_length_criteria(Leg_length,SMAW_Adjust,GMAW_Adjust,
                           GTAW_Adjust,SAW_Adjust)),

    save("criteria.dba").

get_PM("CAST IRON"," 0 0 0 0 1 ").
get_PM("STAINLESS STEEL"," 0 0 0 1 0 ").
get_PM("MILD STEEL","0 1 0 0 0 ").
get_PM("CARBON STEEL","0 0 1 0 0 ").
get_PM("ALUMINUM ALLOY"," 1 0 0 0 0 ").

get_WT("FILLET"," 0 1 ").
get_WT("BUTT"," 1 0 ").

get_POSI("FLAT"," 1 0 0 0 0 ").
get_POSI("HORIZONTAL"," 0 1 0 0 0 ").
get_POSI("VERTICAL"," 0 0 1 0 0 ").
get_POSI("DOWNWARD"," 0 0 0 1 0 ").
get_POSI("OVERHEAD"," 0 0 0 0 1 ").

get_APP("FLAT","1 0 0 ").
get_APP("CONCAVE"," 0 0 1 ").
get_APP("CONVEX"," 0 1 0 ").

get_DEP("HIGH"," 1 0 0 ").
get_DEP("MEDIUM"," 0 1 0 ").
get_DEP("LOW"," 0 0 1 ").

get_strength("HIGH"," 1 0 ").
get_strength("LOW"," 0 1 ").
get_PT("HIGH"," 1 0 0 ").
get_PT("MEDIUM"," 0 1 0 ").
get_PT("LOW"," 0 0 1 ").
get_SL("HIGH"," 1 0 ").
get_SL("LOW"," 0 1 ").
get_LL("HIGH"," 1 0 ").
get_LL("LOW"," 0 1 ").

get_WQ("HIGH"," 0 0 1 ").
get_WQ("MEDIUM"," 0 1 0 ").
get_WQ("LOW"," 0 0 1 ").

get_AUT("YES"," 1 0 ").
get_AUT("NO"," 0 1 ").

```



```

Seam_length_crit (Seam_Length,Factor) :-
    Seam_Length >= 10,
    Factor = "HIGH".

Seam_length_crit (Seam_Length,Factor) :-
    Seam_Length < 10,
    Factor = "LOW".

Plate_thickness_crit (Plate_thickness,Factor) :-
    Plate_thickness >= 10,
    Factor = "HIGH".

Plate_thickness_crit (Plate_thickness,Factor) :-
    Plate_thickness >= 5,
    Plate_thickness < 10,
    Factor = "MEDIUM".

Plate_thickness_crit (Plate_thickness,Factor) :-
    Plate_thickness < 5,
    Factor = "LOW".

Strength_crit (Strength,Factor) :-
    Strength <= 70000,
    Factor = "LOW".

Strength_crit (Strength,Factor) :-
    Strength > 70000,
    Factor = "HIGH".

Leg_length_crit (Leg_length,Factor):-
    Leg_length >= 10,
    Factor = "HIGH".

Leg_length_crit (Leg_length,Factor):-
    Leg_length < 10,
    Factor = "LOW".

consult_usr_data :-
    existfile("JOB_DATA.DBA"),!,
    consult("JOB_DATA.DBA").
consult_usr_data.

consult_process_selected :-
    existfile("process.dba"),!,
    consult("process.dba",process).
consult_process_selected.

consult_wps_db :-
    existfile("wps.dba"),!,
    consult("wps.dba",wps).
consult_wps_db.

consult_criteria :-
    existfile("criteria.dba"),!,
    consult("criteria.dba",criteria).
consult_criteria.

```

```
goal
    adjust_knowledge.
```

### **E.5 WPS\_GEN.PRO - Prolog Program for generating the Welding Procedure Specification**

```
project "neurex"
include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu2.pro"
```

```
database - counter
    counter (integer)
```

```
database - input_par
    SMAW_par(string,real,string,real,real,real,real,real,real,string)
    GMAW_par(string,real,string,real,real,real,real,real,real,real,real,string)
```

```
database - wps
    wps_process(string,string)
```

```
predicates
    wps_gen2 (string, integer)
    wps_gen1 (string,integer)
    consult_proc_data
    consult_par_data
    consult_job_data
    gas_cup_opening(string,string,string,string,string,real)
    preheating_req (string,real,string,string,string,string)
    SMAW_welding_technique (string,real,string,string)
    welding_technique (string,real,string,string,string)
    get_joint_design(real)
    welded_from(real,string,string)
```

```
clauses
    consult_job_data:-
        existfile("job_data.dba"),!,
        consult("job_data.dba").
    consult_job_data.
```

```
consult_par_data :-
    existfile("par_sel.dba"),!,
    consult("par_sel.dba",input_par).
consult_par_data.
```

```
consult_proc_data :-
```

```

    existfile("wps.dba"),!,
    consult("wps.dba",wps).
    consult_proc_data.

wps_gen :-
    consult_par_data,
    consult_proc_data,
    consult_job_data,
    wps_process(Y,X),
    Counter = 1,
    wps_gen1 (X,Counter).

wps_gen2 ("SMAW",Counter) :-
    SMAW_par(Weld_type,Leg_length,Position,Counter,
              Elec_size,Current,Num_elec,Weight,
              Arc_time,Overhead_time,Electrode),
    write ("\tRun Number: ",Counter),
    write ("\tElectrode Size: ",Elec_size),
    write ("\tCurrent (Amp): ",Current),nl,
    Counter2 = Counter + 1,
    write("\t-----"),nl,
    wps_gen2 ("SMAW", Counter2).

wps_gen2 ("GMAW",Counter) :-
    GMAW_par(Weld_type,Leg_length,Position,Counter,
              Elec_size,Wt,Voltage,Wire_Feed,Current,Num_elec,
              Speed,Arc_time,S_Gas),
    write ("\tRun Number: ",Counter),
    write ("\tElectrode Size: ",Elec_size),
    write ("\tCurrent (Amp): ",Current),nl,
    write ("\tVoltage (V): ",Voltage),
    write ("\tWire Feed (m/hour): ",Wire_Feed),
    write ("\tSpeed (m/hour): ",Speed),nl,
    write("\t-----"),nl,
    Counter2 = Counter + 1,
    wps_gen2 ("GMAW", Counter2).

wps_gen2 (_,_).

wps_gen1 ("SMAW",Counter) :-
    job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,
              Appearance,_,_,Leg_length),
    SMAW_par(Weld_type,Leg_length,Position,Counter,
              Elec_size,Current,Num_elec,Weight,
              Arc_time,Overhead_time,Electrode),
    makewindow(1,112,112,"Welding Procedure Specification",0,0,25,80),
    write ("\tProcess: Shielded Metal Arc Welding"),nl,
    write ("\tType of Joint: ",Weld_type),

```

```

write ("\tPlate thickness: ",Plate),nl,
write ("\tPosition: ", Position),nl,
write ("\tElectrode Specification:",Electrode),nl,
preheating_req ("SMAW",Plate,Parent_Material,_,
               Preheat,Interpassheat),
write ("\tPreheating: ",Preheat),nl,
write ("\tInterpass Heating: ",Interpassheat),nl,
SMAW_welding_technique ("SMAW",Plate,Parent_Material,Weld_Technique),
write ("\tWelding Technique: ",Weld_Technique),nl,
write ("\tWelded from: "),
welded_from(Plate,Parent_Mat,Welded_from),
write (Welded_from),nl,
get_joint_design(Plate),
write ("\tCleaning: No Cleaning Required between passes"),nl,
write("\t-----"),nl,
wps_gen2("SMAW",Counter),
readchar(_),
openwrite (outfile,"wps_gen.out"),
writedevise (outfile),
write ("\tProcess: Shielded Metal Arc Welding"),nl,
write ("\tType of Joint: ",Weld_type),nl,
write ("\tPlate thickness: "),nl,
write ("\tPosition: ", Position),nl,
write ("\tWelded from: "),
welded_from(Plate,Parent_Mat,Welded_from),
write (Welded_from),nl,
preheating_req ("SMAW",Plate,Parent_Material,
               S_Gas,Preheat,Interpassheat),
write ("\tPreheating and Temperature Control: ",Preheat),nl,
write ("\tInterpass Heating: ",Interpassheat),nl,
SMAW_welding_technique ("SMAW",Plate,Parent_Material,Weld_Technique),
write ("\tWelding Technique: ",Weld_Technique),nl,
get_joint_design(Plate),
write ("\tCleaning: Clean with wire brush and chipping hammer in between passes"),nl,
wps_gen2 ("SMAW",Counter),
closefile (outfile),
writedevise(screen).

wps_gen1 ("GMAW",Counter) :-
    job_data(Parent_Mat,Weld_Type,Position,Seam,Plate,Stren,
             Appearance,_,_,_,Leg_length),
    GMAW_par(Weld_type,Leg_length,Position,Counter,
             Elec_size,Wt,Voltage,Wire_Feed,Current,Num_elec,
             Speed,Arc_time,S_Gas),
    openwrite (outfile,"wps_gen.out"),
    writedevise (outfile),
    write ("\tProcess: Gas Metal Arc Welding"),nl,
    write ("\tType of Joint: ",Weld_type),
    write ("\tPlate thickness: ",Plate),

```

```

write ("\tPosition: ", Position),nl,
write ("\tWelded from: "),
welded_from(Plate,Parent_Mat,Welded_from),
write (Welded_from),
write ("\tElectrode Specification:",Parent_Mat),
write ("\tShielding Gas: ",S_Gas),nl,
gas_cup_opening("GMAW",Parent_Mat,S_Gas,Position,Opening,Plate),
write ("\tGas Cup Opening: ",Opening),nl,
preheating_req ("GMAW",Plate,Parent_Material,
S_Gas,Preheat,Interpassheat),
write ("\tPreheating: ",Preheat),
write ("\tInterpass Heating: ",Interpassheat),
welding_technique ("GMAW",Plate,Parent_Material,S_Gas,Weld_Technique),
write ("\tWelding Technique: ",Weld_Technique),
get_joint_design(Plate),nl,
write ("\tCleaning: No Cleaning Required between passes"),nl,
write ("\tElectrode Size: ",Elec_size),nl,
Counter1 = Counter+1,
wps_gen2("GMAW",Counter),
closefile (outfile),
writedevic (screen),
makewindow(1,112,112,"GMAW Welding Procedure Specification",0.0,25,80),
write ("\tProcess: Gas Metal Arc Welding"),nl,
write ("\tType of Joint: ",Weld_type),
write ("\tPlate thickness: ",Plate),
write ("\tPosition: ", Position),nl,
write ("\tElectrode Specification:",Parent_Mat),
write ("\tShielding Gas: ",S_Gas),nl,
write ("\tGas Cup Opening: ",Opening),nl,
write ("\tPreheating: ",Preheat),
write ("\tInterpass Heating: ",Interpassheat),nl,
write ("\tWelding Technique: ",Weld_Technique),nl,
write ("\t-----"),nl,
write ("\tWelded from: "),
write (Welded_from),nl,
get_joint_design(Plate),
write ("\tCleaning: No Cleaning Required between passes"),nl,
write ("\t-----"),nl,
Counter = 1,
wps_gen2("GMAW",Counter),
readchar(_).

wps_gen1 ("GMAW",_).
welded_from(Plate,"CARBON STEEL",Welded_from) :-
    Plate <= 12,
    Welded_from="One Side".

welded_from(Plate,"CARBON STEEL",Welded_from) :-
    Plate > 12,

```

```

Welded_from="Both Sides".

welded_from(Plate,"ALUMINUM ALLOY",Welded_from) :-
    Plate <= 12,
    Welded_from="One Side".

welded_from(Plate,"ALUMINUM ALLOY",Welded_from) :-
    Plate > 12,
    Welded_from="Both Sides".

get_joint_design(Plate):-
    Plate < 6,
    JointDesign="No Edge Preparation is Necessary".

get_joint_design(Plate):-
    Plate < 12,
    Plate > 6,
    JointDesign="Single-Sided V Shape",
    RootFace=round(Plate/6),
    RootGap=round(Plate/6),
    write("\tJoint Design: ",JointDesign),
    Write("\tRoot Face: ",RootFace),nl,
    write("\tRoot Angle: 60 degree"),
    write("\tRoot Gap: ",RootGap),nl.

get_joint_design(Plate):-
    Plate >= 12,
    JointDesign="Double-Sided X Shape",
    RootFace=round(Plate/6),
    RootGap=round(Plate/6),
    write("\tJoint Design: ",JointDesign),nl,
    Write("\tRoot Face: ",RootFace),nl,
    write("\tRoot Angle: 60 degree"),nl,
    write("\tRoot Gap: ",RootGap),nl.

SMAW_welding_technique ("SMAW",Plate,"CARBON STEEL",Weld_Technique):-
    Plate >= 5,
    Weld_Technique="Side-to-side weaving",!..

SMAW_welding_technique ("SMAW",Plate,"CARBON STEEL",Weld_Technique):-
    Plate < 5,
    Weld_Technique="Use Copper Plate Backing",!..

welding_technique ("GMAW",Plate,"ALUMINUM ALLOY","CO2",Weld_Technique):-
    Plate > 5,
    Weld_Technique="Side-to-side weaving",!..

welding_technique ("GMAW",Plate,"CARBON STEEL","CO2",Weld_Technique):-
    Plate > 5,

```

```

Weld_Technique="Side-to-side weaving",!.

welding_technique ("GMAW",Plate,"ALUMINUM ALLOY","CO2",Weld_Technique):-
    Plate <= 5,
    Weld_Technique="Use Copper Plate Backing",!.

welding_technique ("GMAW",Plate,"CARBON STEEL","CO2",Weld_Technique):-
    Plate < 5,
    Weld_Technique="Use Copper Plate Backing",!.

preheating_req ("GMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness <= 20,
    Preheat = "20 degree C",
    Interpassheat = "20 degree C",!.

preheating_req ("GMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 20,
    Plate_Thickness <= 40,
    Preheat = "70 degree C",
    Interpassheat = "70 degree C",!.

preheating_req ("GMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 40,
    Plate_Thickness <= 65,
    Preheat = "70 degree C",
    Interpassheat = "110 degree C",!.

preheating_req ("GMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 65,
    Preheat = "110 degree C",
    Interpassheat = "150 degree C",!.

preheating_req ("SMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness <= 20,
    Preheat = "20 degree C",
    Interpassheat = "20 degree C",!.

preheating_req ("SMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 20,
    Plate_Thickness <= 40,
    Preheat = "70 degree C",
    Interpassheat = "70 degree C",!.

preheating_req ("SMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 40,
    Plate_Thickness <= 65,
    Preheat = "70 degree C",
    Interpassheat = "110 degree C",!.

```

```

preheating_req ("SMAW",Plate_Thickness,"CARBON STEEL","CO2",Preheat,Interpassheat):-
    Plate_Thickness > 65,
    Preheat = "110 degree C",
    Interpassheat = "150 degree C",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","FLAT",Opening,Plate) :-
    Plate <= 13,
    Opening = "1/2 inches",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","FLAT",Opening,Plate) :-
    Plate > 13,
    Opening = "5/8 inches",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","HORIZONTAL",Opening,Plate) :-
    Plate <= 10,
    Opening = "1/2 inches",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","HORIZONTAL",Opening,Plate) :-
    Plate > 10,
    Opening = "5/8 inches",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","OVERHEAD",Opening,Plate) :-
    Plate <= 10,
    Opening = "1/2 inches",!

gas_cup_opening("GMAW","ALUMINUM ALLOY","CO2","OVERHEAD",Opening,Plate) :-
    Plate > 10,
    Opening = "5/8 inches",!

gas_cup_opening("GMAW","CARBON STEEL","CO2","FLAT",Opening,Plate) :-
    Plate <= 13,
    Opening = "1/2 inches",!

gas_cup_opening("GMAW","CARBON STEEL","CO2","FLAT",Opening,Plate) :-
    Plate > 13,
    Opening = "5/8 inches",!

gas_cup_opening("GMAW","CARBON STEEL","CO2","HORIZONTAL",Opening,Plate) :-
    Plate <= 10,
    Opening = "1/2 inches",!

gas_cup_opening("GMAW","CARBON STEEL","CO2","HORIZONTAL",Opening,Plate) :-
    Plate > 10,
    Opening = "5/8 inches",!

gas_cup_opening("GMAW","CARBON STEEL","CO2","OVERHEAD",Opening,Plate) :-
    Plate <= 10,
    Opening = "1/2 inches",!

```



```

gas_cup_opening("GMAW","CARBON STEEL","CO2","OVERHEAD".Opening,Plate) :-
    Plate > 10,
    Opening = "5/8 inches",!.

/* wps_gen1 ("GTAW",Counter,Counter1) :-
    consult_par_data,
    GMAW_par(Weld_type,Leg_length,Position,Runs,Elec_size,Voltage,Current,Weight,Num_elec,Ove
rhead_time,Arc_time,G_Flow,S_Gas),
    openwrite (outfile,"wps_gen.out"),
    writedevise (outfile),
    write ("Process: Gas Tungsten Arc Welding"),nl,
    write ("Type of Joint: ",Weld_type),nl,
    write ("Plate thickness/Leg Length: ", Leg_length),nl,
    write ("Position: ", Position),nl,
    write ("Welded from: "),nl,
    write ("Electrode Size: ",Elec_size),nl,
    write ("Current (Amp): ",Current),nl,
    write ("Voltage (V): ",Voltage),nl,
    write ("Arc Speed (m/s): "),nl,
    write ("Shielding Gas: "),nl,
    write ("Gas Flow Rate: ",G_Flow),nl,
    write ("Gas Cup Opening: "),nl,
    write ("Preheating and Temperature Control: "),nl,
    write ("Post Heating: "),nl,
    write ("Welding Technique: "),nl,
    write ("Cleaning: "),nl,
    write ("Defects: "),nl,
    closefile (outfile).

wps_gen1 ("SAW",Counter,Counter1) :-
    consult_par_data,
    saw_data(Weld_type,Leg_length,Position,Runs,Elec_size,Voltage,Current,Speed,Num_elec,Arc_T,O
verhead_time,Weight,T_Flux),
    openwrite (outfile,"wps_gen.out"),
    writedevise (outfile),
    write ("Process: Submerged Arc Welding"),nl,
    write ("Flux: ",T_Flux),nl,
    write ("Type of Joint: ",Weld_type),nl,
    write ("Plate thickness: "),nl,
    write ("Position: ", Position),nl,
    welded_from(Plate,Parent_Mat,Welded_From),
    write ("Welded from: ",Welded_From),nl,
    write ("Electrode Size: ",Elec_size),nl,
    write ("Current (Amp): ",Current),nl,
    write ("Voltage (V): ",Voltage),nl,
    write ("Arc Speed (m/s): "),nl,
    write ("Preheating and Temperature Control: "),nl,
    write ("Post Heating: "),nl,
    write ("Welding Technique: "),nl,

```

```

        write ("Cleaning: "),nl,
        write ("Defects: "),nl,
        closefile (outfile).
*/

```

## E.6 DEF\_ANAL.PRO - Prolog Program for Defect Analyzer

project "neurex"

```

include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu2.pro"

domains
    %D,F,S,T = integer

database - defect_par
    defect_parameter (integer, string, string)

database - wps
    wps_process(string,string)

database - defect
    defects(integer,string)
    parameter_adjustment (string)

database - parameter
    SMAW_par(string,real,string,real,real,real,real,real,real,real,string)
    GMAW_par(string,real,string,real,real,real,real,real,real,real,real,real,string)

    gtaw_par(string,real,string,real,real,real,real,real,real,real,real,real,string)
    saw_par(string,real,string,real,real,real,real,real,real,real,real,real,string)

predicates
    process_choice (integer)
    SMAW_defect(integer)
    SMAW_undercuts (integer)
    SMAW_spatter (integer)
    SMAW_shallow_penetration (integer)
    SMAW_poor_fusion (integer)
    SMAW_porosity(integer)
    GMAW_defect (integer)
    GMAW_undercuts (integer)
    GMAW_spatter(integer)
    GMAW_shallow_penetration(integer)
    GMAW_poor_fusion(integer)
    GMAW_porosity(integer)
    GMAW_concave_deposit (integer)

```

```

GMAW_convex_deposit (integer)
gtaw_defect(integer)
gtaw_undercuts(integer)
gtaw_spatter(integer)
gtaw_shallow_penetration(integer)
gtaw_poor_fusion(integer)
gtaw_porosity(integer)
gtaw_tungsten_inclusion (integer)
saw_defect (integer)
saw_undercuts(integer)
saw_shallow_penetration(integer)
saw_poor_fusion(integer)
saw_porosity(integer)
Adjust_Current (integer)
Adjust_Electrode_Type (integer)
Adjust_Electrode_Size (integer)
get_electrode_category (string,string)
get_shielding_gas (string,string)
SMAW_Neural_Net_Prediction (integer)
GMAW_Neural_Net_Prediction
Store_SMAW_Defects (real,real,real,real,real)
Store_GMAW_Defects (real,real,real,real,real,real,real,real)
consult_parameter_data
consult_wps_process
consult_defects
get_electrode_size_category(real,string)
get_electrode_size(real,real)
get_speed_category(real,string)
get_current_category (real,string)
final_list
get_electrode_category_name (string,string)
get_current(string,string,real,real).
get_electrode (string,string,string,string)
Defect_Analyzer_Main_Menu
Defect_Analyzer_choice (integer)
consult_wps_proc
write_wps_proc(string)
%Parameter_adjustment(string)
defect_avoidance_wps (string)
defect_avoidance_wps(string)
hydrogen_defect_avoidance_wps(string)
spatter_defect_Avoidance (string)
joint_defect_avoidance(string)
Parameter_adjustments (string)
my_assert (string)
save_parameter_data

clauses
Defect_Analyzer_Main_Menu :-

```

```

        %consult_parameter_data,
        menu(5,5,7,7,["INTERACTIVE","PARAMETER ADJUSTMENT","AUTOMATED WPS
TIPS","QUIT"],"SELECT AN ACTION",0,CHOICE),
        Defect_Analyzer_choice (CHOICE).

Defect_Analyzer_choice (1):-
    Defect_Analyzer.

Defect_Analyzer_choice (2):-
    consult_wps_proc,
    wps_process(_,Process),
    Parameter_adjustments (Process).

Defect_Analyzer_choice (3):-
    consult_wps_proc,
    wps_process(_,Process),
    write_wps_proc(Process).

Defect_Analyzer_choice (4):-
    exit.

consult_wps_proc:-
    existfile("wps.db"),!,
    consult("wps.db",wps).
consult_wps_proc.

write_wps_proc ("SMAW"):-
    existfile("wps_gen.out"),
    openappend(outfile,"wps_gen.out"),
    writedevise(outfile),
    write("\n\nDEFECT ANALYZER TIPS \n"),
    write("===== ") ,nl,!,
    consult_defects,
    defect_avoidance_wps("SMAW"),
    hydrogen_defect_avoidance_wps("SMAW"),
    spatter_defect_Avoidance ("SMAW"),
    joint_defect_avoidance("SMAW"),
    closefile(outfile),
    writedevise(screen),
    makewindow(1,112,112,"Defect Analyzer Tips",0,0,20,80),
    write("\n\nDEFECT ANALYZER TIPS \n"),
    write("===== ") ,nl,!,
    consult_defects,
    defect_avoidance_wps("SMAW"),
    hydrogen_defect_avoidance_wps("SMAW"),
    spatter_defect_Avoidance ("SMAW"),
    joint_defect_avoidance("SMAW"),
    closefile(outfile),
    writedevise(screen),

```

```

% write("All WPS instructions written successfully").
readchar(_).

write_wps_proc ("SMAW"):-
    openwrite(outfile,"wps_gen.out"),
    writedevise(outfile),
    write("\n\nDEFECT ANALYZER TIPS \n"),
    write("===== ").nl,!,
    consult_defects,
    defect_avoidance_wps("SMAW"),
    hydrogen_defect_avoidance_wps("SMAW"),
    spatter_defect_Avoidance ("SMAW"),
    joint_defect_avoidance ("SMAW"),
    closefile(outfile),
    writedevise(screen),
    write("All WPS instructions written successfully"),
    readchar(_).

defect_avoidance_wps("SMAW"):-
    defects(_,"Slag Inclusions"),
    write("Slag Inclusions Avoidance Tips").nl,
    write("Use Preheat and higher Weld heat input."),
    write("Proper deslagging in between passes").nl,
    write("Avoid joint design contour difficult to penetrate with Arc").nl,
    write("-----").nl.

defect_avoidance_wps("SMAW"):-
    defects(_,"Undercuts").nl,
    write("Undercuts Avoidance Tips").nl,
    write("Shorten Arc Length").nl,
    write("Change Angle of Electrode so that arc force helps fill undercut").nl,!,
    write("-----").nl.

defect_avoidance_wps("SMAW").

Hydrogen_defect_avoidance_wps("SMAW"):-
    defects(_,"Porosity"),
    defects(_,"Cracks"),
    write("Porosity and Cracking may be there due to the parameters chosen").nl,
    write("To minimize them, please do the following").nl,
    write("Bake Electrodes before Use").nl,
    write("Clean the Plates thoroughly with a wire brush").nl,
    write("Ensure that you provide Pre-heating and Interpass Heating").nl,!,
    write("Use Weaving Techniques").nl,
    write("Back-step to fill craters").nl,
    write("-----").nl.

Hydrogen_defect_avoidance_wps("SMAW"):-
    defects(_,"Porosity"),

```

```

write("Porosity Avoidance Tips"),nl,
write("Clean the Plates thoroughly with a wire brush"),nl,
write("Ensure that you provide Pre-heating and Interpass Heating"),nl,
write("Use Weaving Techniques."),
write("Back-step to fill craters"),nl,!,
write("-----"),nl.

```

```

Hydrogen_defect_avoidance_wps("SMAW"):-
defects(_, "Cracks"),
write("Cracks may be there due to the parameters chosen"),nl,
write("To minimize them, please do the following"),nl, write("Bake Electrodes before Use"),nl,
write("Clean the Plates thoroughly with a wire brush"),nl,
write("Ensure that you provide Pre-heating and Interpass Heating"),nl,
write("Use Weaving Techniques"),nl,
write("Back-step to fill craters"),nl,
write("-----"),nl.

```

```

Hydrogen_defect_avoidance_wps("SMAW").

```

```

spatter_defect_avoidance("SMAW"):-
defects(_, "Arc Blow"),
defects(_, "Spatter"),
write("Arcblow and Spatter may be there due to the parameters chosen"),nl,
write("To minimize them, please do the following"),nl, write("Use AC Welding Machine, if
possible"),nl,
write("Use Brass or Copper Backup Bar"),nl,
write("Rearrange or Split Ground Clamp"),nl,
write("Adjust Electrode Angle"),nl,!,
write("-----"),nl.

```

```

spatter_defect_avoidance("SMAW"):-
defects(_, "Arc Blow"),
write("Arc Blow Avoidance Tips"),nl,
write("Use Brass or Copper Backup Bar"),nl,
write("Rearrange or Split Ground Clamp"),nl,
write("Adjust Electrode Angle"),nl,!,
write("-----"),nl.

```

```

spatter_defect_avoidance("SMAW"):-
defects(_, "Spatter"),
write("Spatter may be there due to the parameters chosen"),nl,
write("To minimize them, please do the following"),nl, write("Use AC Welding Machine, if
possible"),nl,
write("Use Brass or Copper Backup Bar"),nl,
write("Rearrange or Split Ground Clamp"),nl,
write("Adjust Electrode Angle"),nl,
write("-----"),nl.
spatter_defect_avoidance("SMAW").

```

```

joint_defect_avoidance("SMAW"):-
    defects(_,"Shallow Penetration"),
    defects(_,"Poor Fusion"),
    write("Shallow Penetration and Poor Fusion may be there due to the parameters chosen"),nl,
    write("To minimize them, please do the following"),nl,
    write("Use larger root opening and root face"),nl,
    write("Use larger Root Angle"),!,
    write("-----"),nl.

joint_defect_avoidance("SMAW"):-
    defects(_,"Shallow Penetration"),
    write("Shallow Penetration may be there due to the parameters chosen"),nl,
    write("To minimize them, please do the following"),nl,
    write("Use larger root opening and root face"),nl,
    write("Use larger Root Angle"),!,
    write("-----"),nl.

joint_defect_avoidance("SMAW"):-
    defects(_,"Poor Fusion"),
    write("Slag Inclusions may be there due to the parameters chosen"),nl,
    write("To minimize them, please do the following"),nl,
    write("Use larger root opening and root face"),nl,
    write("Use larger Root Angle"),!,
    write("-----"),nl.
joint_defect_avoidance("SMAW").

/*
write_wps_proc ("GMAW"):-
    existfile("wps_gen.out"),
    openappend(outfile,"wps_gen.out"),
    writedevise(outfile),
    write("\n\nDEFECT ANALYZER TIPS \n"),
    write("===== "),nl,!,
    closefile(outfile).

write_wps_proc ("GMAW"):-
    openwrite(outfile,"wps_gen.out"),
    writedevise(outfile),
    write("\n\nDEFECT ANALYZER TIPS \n"),
    write("===== "),nl,!,
    closefile(outfile).

*/
save_parameter_data :-
    save("par_sel.dbf",parameter).

Parameter_Adjustments ("SMAW") :-
    % consult_parameter_data,

```

```

SMAW_Neural_Net_Prediction (1).
Adjust_Electrode_Type (1).
Adjust_electrode_size (1).
Adjust_Current (1).
% save("par_sel.dba",parameter),!.

Parameter_Adjustments ("SMAW"):-
    save("par_Sel.dba",parameter).

SMAW_Neural_Net_Prediction (Counter) :-
    consult_parameter_data,
    makewindow(1,112,112,"Defect Analysis",0,0,25,80),nl,
    SMAW_par(Weld_type,Leg_length,Position,Counter,
        Elec_size,Current,Num_elec,Weight,
        Arc_time,Overhead_time,Electrode),
    get_electrode_category(Electrode,Category),
    openwrite(outfile,"defSMAW3.in"),
    writedevise(outfile),
    get_electrode_size_category(Elec_size,Elec_size_cat),
    get_current_category (Current,Current_Cat),
    write (Category," ",Elec_size_cat," ",Current_Cat,"."),
    closefile(outfile),
    system("testbp -R defSMAW3"),
    existfile("defSMAW3.out"),!,
    openread(infile,"defSMAW3.out"),
    readdevice(infile),
    readln(X),
    closefile(infile),
    fronttoken(X,ArcBlow,Rest1),
    fronttoken(Rest1,ShallowPenetration,Rest2),
    %fronttoken(Rest2,PoorFusion,Rest3),
    fronttoken(Rest2,Porosity,Rest4),
    %fronttoken(Rest4,Spatter,Rest5),
    fronttoken(Rest4,Undercut,Rest6),
    %fronttoken(Rest6,Cracks,Rest7),
    fronttoken(Rest6,SlagInclusions,Rest8),
    str_real(ArcBlow,ArcBlow1),
    str_real(ShallowPenetration,ShallowPenetration1),
    %str_real(PoorFusion,PoorFusion1),
    str_real(Porosity,Porosity1),
    str_real(Undercut,Undercut1),
    %str_real(Cracks,Cracks1),
    str_real(SlagInclusions,SlagInclusions1),
    %str_real(Spatter,Spatter1),
    Store_SMAW_Defects (ArcBlow1,ShallowPenetration1,Porosity1,Undercut1,SlagInclusions1),
    final_list,
    save("defects.dba",defect),
    Counter1=Counter+1,
SMAW_Neural_net_prediction (Counter1).

```



SMAW\_Neural\_net\_prediction (\_).

Store\_SMAW\_Defects (A,B,C,D,E) :-

```
A > 0.5,  
  assert (defects (5,"Arc Blow")),  
  write ("Defect Predicted: Arc Blow\n"),  
  my_assert ("DECREASE CURRENT"),  
  my_assert ("DECREASE ELECTRODE SIZE"),  
  fail.
```

Store\_SMAW\_Defects (A,B,C,D,E) :-

```
B > 0.5,  
  assert (defects(4,"Shallow Penetration")),  
  write ("Defect Predicted: Shallow Penetration\n"),  
  my_assert ("INCREASE CURRENT"),  
  my_assert ("DECREASE ELECTRODE SIZE"),  
  my_assert ("DECREASE WELDING SPEED"),  
  fail.
```

Store\_SMAW\_Defects (A,B,C,D,E) :-

```
C > 0.5,  
  assert(defects(3,"Porosity")),  
  write ("Defect Predicted: Porosity\n"),  
  my_assert ("USE LOW HYDROGEN ELECTRODE"),  
  my_assert ("DECREASE CURRENT"),  
  my_assert ("DECREASE ELECTRODE SIZE"),  
  my_assert ("DECREASE WELDING SPEED"),  
  fail.
```

Store\_SMAW\_Defects (A,B,C,D,E) :-

```
D > 0.5,  
  assert(defects(2,"Undercuts")),  
  write ("Defect Predicted: Undercuts\n"),  
  my_assert ("DECREASE CURRENT"),  
  my_assert ("DECREASE ELECTRODE SIZE"),  
  my_assert ("DECREASE WELDING SPEED"),  
  fail.
```

Store\_SMAW\_Defects (A,B,C,D,E) :-

```
E > 0.5,  
  assert(defects(1,"Slag Inclusions")),  
  write ("Defect Predicted: Slag Inclusions\n"),  
  my_assert ("USE RUTILE ELECTRODE"),  
  my_assert ("INCREASE CURRENT"),  
  my_assert ("DECREASE WELDING SPEED"),  
  fail.
```

```
Store_SMAW_Defects (_,_,_,_,_).
```

```
my_assert (X) :-  
    parameter_adjustment (X),!;  
    assert(parameter_adjustment(X)).
```

```
GMAW_Neural_Net_Prediction:-  
    consult_parameter_data.  
    GMAW_par(Weld_type,Leg_length,Position,Counter,  
        Elec_size,Wt,Voltage,Wire_Feed,Current,Num_elec,  
        Speed,Arc_time,S_Gas),  
    get_shielding_gas(Shielding_Gas,Category),  
    %Speed=(1/Arc_time)*100, %Speed in centimeters/min  
    openwrite(outfile,"def_GMAW.in"),  
    writedevise(outfile),  
    write (Category," ",Elec_size," ",Current," ",Speed,""),  
    closefile(outfile),  
    system("testbp -R def_GMAW"),  
    existfile("def_SMAW.out"),!,  
    openread(infile,"def_GMAW.out"),  
    readdevice(infile),  
    readln(X),  
    closefile(infile),  
    fronttoken(X,ArcBlow,Rest1),  
    fronttoken(Rest1,ShallowPenetration,Rest2),  
    fronttoken(Rest2,PoorFusion,Rest3),  
    fronttoken(Rest3,Porosity,Rest4),  
    fronttoken(Rest4,Undercut,Rest5),  
    fronttoken(Rest5,Cracks,Rest6),  
    fronttoken(Rest6,ConvexBead,Rest7),  
    fronttoken(Rest7,ConcaveBead,Rest8),  
    str_real(ArcBlow,ArcBlow1),  
    str_real(ShallowPenetration,ShallowPenetration1),  
    str_real(PoorFusion,PoorFusion1),  
    str_real(Porosity,Porosity1),  
    str_real(Undercut,Undercut1),  
    str_real(Cracks,Cracks1),  
    str_real(ConcaveBead,ConcaveBead1),  
    str_real(ConvexBead,ConvexBead1),  
    Store_GMAW_Defects (ArcBlow1,ShallowPenetration1,PoorFusion1,  
        Porosity1,Undercut1,Cracks1,ConvexBead1,ConcaveBead1),  
    final_list,  
    save("DEFECTS.DBA",defect),  
    Adjust_Electrode_Type(1),  
    Adjust_Electrode_Size(1),  
    Adjust_Current (1).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    A > 0.5,
    assert(defects(6,"Arc Blow")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    B > 0.5,
    assert(defects(5,"Shallow Penetration")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    C > 0.5,
    assert(defects(4,"Poor Fusion")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    D > 0.5,
    assert(defects(3,"Porosity")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    E > 0.5,
    assert(defects(2,"Undercuts")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    F > 0.5,
    assert(defects(1,"Cracks")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    G > 0.5,
    assert(defects(7,"Convex Beads")).
```

```
Store_GMAW_Defects (A,B,C,D,E,F,G,H) :-
    H > 0.5,
    assert(defects(8,"Concave Beads")).
```

```
final_list :-
    parameter_adjustment("DECREASE CURRENT"),
    parameter_adjustment("INCREASE CURRENT"),
    retract(parameter_adjustment("DECREASE CURRENT")),
    retract(parameter_adjustment("INCREASE CURRENT")),
    fail.
```

```
final_list :-
    parameter_adjustment("DECREASE ARC TRAVEL SPEED"),
    parameter_adjustment("INCREASE ARC TRAVEL SPEED"),
    retract(parameter_adjustment("DECREASE ARC TRAVEL SPEED")),
    retract(parameter_adjustment("INCREASE ARC TRAVEL SPEED")),
    fail.
```

```
final_list :-
```

```

parameter_adjustment("DECREASE ELECTRODE SIZE"),
parameter_adjustment("INCREASE ELECTRODE SIZE"),
retract(parameter_adjustment("DECREASE ELECTRODE SIZE")),
retract(parameter_adjustment("INCREASE ELECTRODE SIZE")),
fail.

final_list :-
parameter_adjustment("DO NOT USE RUTILE ELECTRODE"),
parameter_adjustment("USE RUTILE ELECTRODE"),
retract(parameter_adjustment("DO NOT USE RUTILE ELECTRODE")),
retract(parameter_adjustment("USE RUTILE ELECTRODE")),
fail.

final_list :-
parameter_adjustment("DO NOT USE CELLULOSE ELECTRODE"),
parameter_adjustment("USE CELLULOSE ELECTRODE"),
retract(parameter_adjustment("DO NOT USE CELLULOSE ELECTRODE")),
retract(parameter_adjustment("USE CELLULOSE ELECTRODE")), fail.

final_list :-
parameter_adjustment("DO NOT USE BASIC ELECTRODE"),
parameter_adjustment("USE BASIC ELECTRODE"),
retract(parameter_adjustment("DO NOT USE BASIC ELECTRODE")),
retract(parameter_adjustment("USE BASIC ELECTRODE")), fail.

final_list :-
parameter_adjustment("USE LOW HYDROGEN ELECTRODE"),
parameter_adjustment("DO NOT USE LOW HYDROGEN ELECTRODE"),
retract(parameter_adjustment("USE LOW HYDROGEN ELECTRODE")),
retract(parameter_adjustment("DO NOT USE LOW HYDROGEN ELECTRODE")), fail.

final_list :-
parameter_adjustment("USE LOW HYDROGEN ELECTRODE"),
parameter_adjustment("USE RUTILE ELECTRODE"),
retract(parameter_adjustment("USE RUTILE ELECTRODE")),
fail.

final_list.

Adjust_Electrode_Type (Counter) :-
%consult_parameter_data,
parameter_adjustment("USE RUTILE ELECTRODE"),
write ("Parameter adjustment recommended: Use Rutile Electrode"),nl,
SMAW_par(Weld_type,Leg_length,Position,Counter,
Elec_size,Current,Num_elec,Weight,
Arc_time,Overhead_time,Electrode),
get_electrode_category_name (Electrode, Electrode_Category),
get_electrode("Rutile",Electrode_Category,Electrode,Electrode_New),
write("Electrode (Old): ",Electrode),nl,
write("Electrode (New): ",Electrode_New),nl,

```

```

write("Do you want to adjust it? (y/n)",nl,
readchar(Reply),
Reply = 'y',
retract(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
assert(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode_New)),!,
save_parameter_data.
% Counter I=Counter+1,
% Adjust_Electrode_Type(Counter I).
% fail.

Adjust_Electrode_Type (Counter) :-
parameter_adjustment("USE LOW HYDROGEN ELECTRODE"),
write ("Parameter adjustment recommended: Use Low Hydrogen Electrode"),nl,
SMAW_par(Weld_type,Leg_length,Position,Counter,
          Elec_size,Current,Num_elec,Weight,
          Arc_time,Overhead_time,Electrode),
get_electrode_category_name (Electrode, Electrode_Category),
get_electrode("Low Hydrogen",Electrode_Category,Electrode,Electrode_New),
write("Electrode (Old): ",Electrode),nl,
write("Electrode (New): ",Electrode_New),nl,
write("Do you want to adjust it? (y/n)",nl,
readchar(Reply),
Reply = 'y',
retract(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
assert(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode_New)),!,
save_parameter_data.
Adjust_Electrode_Type (_).

get_electrode_size(_,3.25):-!, write ("Expert Advice: 3.25 is the lowest size. Please select NO\n").
get_electrode_size(3.25,4):-!.
get_electrode_size(4,5):-!.
get_electrode_size(5,6.3):-!.

Adjust_Electrode_Size (Counter) :-
parameter_adjustment("INCREASE ELECTRODE SIZE"),
write ("Parameter adjustment recommended: Increase Electrode Size"),nl,
SMAW_par(Weld_type,Leg_length,Position,Counter,
          Elec_size,Current,Num_elec,Weight,
          Arc_time,Overhead_time,Electrode),
get_electrode_size (Elec_size, Elec_size_new),
write("Electrode Size (OLD): ",Elec_size),nl,

```

```

write("Electrode Size (NEW): ",Elec_size_new),nl,
write("Do you want to adjust it? (y/n)",nl,
readchar(Reply),
Reply = 'y',
retract(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
assert(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size_new,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
!.save_parameter_data.

```

```

Adjust_Electrode_Size (Counter) :-
parameter_adjustment("DECREASE ELECTRODE SIZE"),
write ("Parameter adjustment recommended: Decrease Electrode Size"),nl,nl,
SMAW_par(Weld_type,Leg_length,Position,Counter,
          Elec_size,Current,Num_elec,Weight,
          Arc_time,Overhead_time,Electrode),
get_electrode_size (Elec_size_new, Elec_size),
write("Electrode Size (OLD): ",Elec_size),nl,
write("Electrode Size (NEW): ",Elec_size_new),nl,
write("Do you want to adjust it? (y/n)",nl,
readchar(Reply),
Reply = 'y',
retract(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
assert(SMAW_par(Weld_type,Leg_length,Position,Counter,
                  Elec_size_new,Current,Num_elec,Weight,
                  Arc_time,Overhead_time,Electrode)),
!.save_parameter_data.

```

```
Adjust_Electrode_Size(_).
```

```

Adjust_Current (Counter) :-
parameter_adjustment("DECREASE CURRENT"),
write ("Parameter adjustment recommended: DECREASE CURRENT"),nl,nl,
SMAW_par(Weld_type,Leg_length,Position,Counter,
          Elec_size,Current,Num_elec,Weight,
          Arc_time,Overhead_time,Electrode),
get_electrode_category_name (Electrode, Electrode_Category),
get_current("LOW",Electrode_Category,Elec_Size,Current I),
write("Old Current Value: ",Current),nl,
write("New Current Value: ",Current I),nl,
write("Do you want to adjust it? (y/n)",nl,
readchar(Reply),
Reply = 'y',
retract(SMAW_par(Weld_type,Leg_length,Position,Counter,

```



```

process_choice (2) :-
    menu(5,5,7,7,{"UNDERCUTS","SPATTER","ARC BLOW","POROSITY/SURFACE
HOLES","POOR FUSION",
        "SHALLOW PENETRATION","CONVEX DEPOSIT","CONCAVE
DEPOSIT"},"GMAW DEFECTS",0,CHOICE),
    GMAW_defect (CHOICE),
    removewindow(1, 1).

process_choice (3) :-
    menu(5,5,7,7,{"UNDERCUTS","SPATTER","ARC BLOW","POROSITY/SURFACE
HOLES","POOR FUSION",
        "SHALLOW PENETRATION","TUNGSTEN INCLUSIONS"},"GTAW
DEFECTS",0,CHOICE),
    gtaw_defect(CHOICE),
    removewindow(1, 1).

process_choice (4) :-
    menu(5,5,7,7,{"UNDERCUTS","POROSITY","POOR FUSION","SHALLOW
PENETRATION"},"SAW DEFECTS",0,CHOICE),
    saw_defect (CHOICE),
    removewindow(1, 1).

SMAW_undercuts (1) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Decrease Current\n"), readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_undercuts (2) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Reduce Travel Speed\n"), readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_undercuts (3) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Change Electrode Angle so that Arc Force holds the metal in the corners\n"),
    readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_undercuts (4) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Avoid excessive weaving\n"), readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_undercuts (5) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Use dry electrodes\n"), readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_spatter (1) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Check current against manufacturer's recommendations\n"),
    readchar(_),removewindow(1,1), Defect_analyzer.

```



```

SMAW_spatter (2) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Try Reversing Polarity/Use Electrode of reverse polarity\n"),
    readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_spatter (3) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Try using a shorter arc length\n"), readchar(_),removewindow(1,1), Defect_analyzer.

SMAW_spatter (4) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Use dry electrodes/bake electrodes before use\n"), readchar(_),removewindow(1,1),
    Defect_analyzer.

SMAW_spatter (5) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended electrode angle\n"), readchar(_),removewindow(1,1),
    removewindow(1,1), Defect_analyzer.

SMAW_defect(1) :-
    menu(5,5,7,7,{"EXCESSIVE WELDING CURRENT","HIGH SPEED OF ARC
    TRAVEL","WRONG ELECTRODE ANGLE","EXCESSIVE SIDE MANIPULATION","DAMP OR
    IMPROPERLY FORMULATED ELECTRODE"},"UNDERCUTS - POSSIBLE CAUSES",0,CHOICE),
    SMAW_undercuts (CHOICE).

SMAW_defect(2) :-
    menu(5,5,7,7,{"HIGH CURRENT","WRONG POLARITY","TOO LONG ARC LENGTH","WET
    ELECTRODE","WRONG ELECTRODE ANGLE"},"SPATTER - POSSIBLE CAUSES",0,CHOICE),
    SMAW_spatter (CHOICE).

SMAW_defect(3) :-
    makewindow (1,5,7,"ARC BLOW - REMEDIES",5,5,20,60),
    write("Remedy 1: Change to AC Welding\n"),
    write("Remedy 2: Use lower currents and smaller electrodes\n"),
    write("Remedy 3: Reduce Arc Length\n"),
    write("Remedy 4: Weld in direction of blow\n"),
    write("Remedy 5: Change the electrical path to work by:\n"),
    write("    a) Shifting work connection to other end\n"),
    write("    b) Welding towards heavy tacks, finished welds,\n"),
    write("        or back-stepping on long welds\n"),
    write("    c) Using run-out tabs, adding steel blocks to\n"),
    write("        change work current path or tacking small\n        plates\n"),
    write("Press any key to continue ....."),
    readchar(_),removewindow(1,1), removewindow(1,1), Defect_analyzer.

SMAW_defect (4):-
    menu(5,5,7,7,{"WORK PIECE DIRTY","STEEL HAS LOW CARBON","STEEL HAS HIGH
    MANGANESE","STEEL HAS HIGH SULFUR/PHOSPHORUS","TOO LONG AN ARC
    LENGTH"},"POROSITY - POSSIBLE CAUSES",0,CHOICE),

```

```

SMAW_porosity(CHOICE).

SMAW_defect(5):-
    menu(5,5,7,7,{"INSUFFICIENT GAP BETWEEN TWO PLATES","DIRTY EDGES","CURRENT
    TOO LOW"},"POOR FUSION - POSSIBLE CAUSES",0,CHOICE),
    SMAW_poor_fusion(CHOICE).

SMAW_defect(6):-
    menu(5,5,7,7,{"INCORRECT ELECTRODE SIZE","LOW WELDING
    CURRENT","INACCURATE JOINT PREPARATION/FITUP"},"SHALLOW PENETRATION -
    POSSIBLE CAUSES",0,CHOICE),
    SMAW_shallow_penetration(CHOICE).

SMAW_shallow_penetration (1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use smaller electrodes to reach down into the narrow grooves\n"),
    readchar(_),removewindow(1,1), removewindow(1,1), Defect_analyzer.

SMAW_shallow_penetration (2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use higher current\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

SMAW_shallow_penetration (3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Allow some gap at the bottom of the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

SMAW_poor_fusion (1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Provide better fit-up or use a weave technique to fill up the gap\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

SMAW_poor_fusion (2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Remove scale/dirt/rust/paint/moisture from the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

SMAW_poor_fusion (3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Try a higher current and stringer bead technique\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

SMAW_porosity (1) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Remove scale/dirt/rust/paint/moisture from the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

SMAW_porosity (2) :-

```

```

makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Use low-hydrogen electrodes. Minimize admixture of base metal\n"),
write("    by using low currents and faster travel speeds for less penetration \n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

SMAW_porosity (3) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Use low-hydrogen electrodes. Minimize admixture of base metal\n"),
write("    by using low currents and faster travel speeds for less penetration \n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

SMAW_porosity (4) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Use low-hydrogen electrodes. Minimize admixture of base metal\n"),
write("    by using low currents and faster travel speeds for less penetration \n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

SMAW_porosity (5) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Try using shorter Arc Lengths\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

**/\* DEFECT ANALYSIS FOR GMAW \*/**

```

GMAW_undercuts (1) :-
makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
write("Remedy: Use Recommended Torch
Manipulation\n"),readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

GMAW_undercuts (2) :-
makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
write("Remedy: Use recommended shielding gas\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

GMAW_spatter(1) :-
makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
write("Remedy: Adjust gas Flow\n"), readchar(_),removewindow(1,1),removewindow(1,1),
Defect_analyzer.

```

```

GMAW_spatter(2) :-
makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
write("Remedy: Adjust Slope\n"), readchar(_),removewindow(1,1),removewindow(1,1),
Defect_analyzer.

```

```

GMAW_spatter(3) :-
makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),

```

```

        write("Remedy: Adjust Voltage\n"), readchar(_).removewindow(1,1),removewindow(1,1),
Defect_analyzer.

```

```

GMAW_spatter(4) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended shielding gas\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

GMAW_spatter(5) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended wire electrode\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

GMAW_defect(1) :-
    menu(5,5,7,7,["WRONG TORCH MANIPULATION","WRONG SHIELDING
GAS"],"UNDERCUTS - POSSIBLE CAUSES",0,CHOICE),
    GMAW_undercuts (CHOICE).

```

```

GMAW_defect (2) :-
    menu(5,5,7,7,["EXCESSIVE/INSUFFICIENT GAS FLOW","INSUFFICIENT SLOPE
SETTING","EXCESSIVE VOLTAGE","WRONG SHIELDING GAS","WRONG
ELECTRODE"],"SPATTER - POSSIBLE CAUSES",0,CHOICE),
    GMAW_spatter (CHOICE).

```

```

GMAW_defect (3) :-
    makewindow (1,5,7,"ARC BLOW - REMEDIES",5,5,20,60),
    write("Remedy 1: Use lower currents and smaller filler wire\n"),
    write("Remedy 2: Reduce Arc Length\n"),
    write("Remedy 3: Weld in direction of blow\n"),
    write("Remedy 4: Change the electrical path to work by:\n"),
    write("    a) Shifting work connection to other end\n"),
    write("    b) Welding towards heavy tacks, finished welds,\n"),
    write("        or back-stepping on long welds\n"),
    write("    c) Using run-out tabs, adding steel blocks to\n"),
    write("        change work current path or tacking small\n        plates\n"),
    write("Press any key to continue ....."),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

GMAW_defect(4):-
    menu(5,5,7,7,["WORK PIECE DIRTY","FAULTY GAS SYSTEM","FAULTY TORCH
ASSEMBLY","IMPROPER TORCH MANIPULATION","SPATTER CLOGGED NOZZLE"],"POROSITY
- POSSIBLE CAUSES",0,CHOICE),
    GMAW_porosity(CHOICE).

```

```

GMAW_defect(5):-
    menu(5,5,7,7,["INSUFFICIENT GAP BETWEEN TWO PLATES","DIRTY EDGES","CURRENT
TOO LOW"],"POOR FUSION - POSSIBLE CAUSES",0,CHOICE),
    GMAW_poor_fusion(CHOICE).

```

```

GMAW_defect(6):-
    menu(5,5,7,7,["INCORRECT ELECTRODE SIZE","LOW WELDING
CURRENT","INACCURATE JOINT PREPARATION/FITUP"],"SHALLOW PENETRATION -
POSSIBLE CAUSES",0,CHOICE),
    GMAW_shallow_penetration(CHOICE).

GMAW_defect(7):-
    menu(5,5,7,7,["WRONG VOLTAGE/WIRE FEED","WRONG INDUCTANCE","WRONG
TORCH MANIPULATION","WRONG TRAVEL SPEED"],"CONVEX DEPOSIT - POSSIBLE
CAUSES",0,CHOICE),
    GMAW_convex_deposit(CHOICE).

GMAW_defect(8):-
    menu(5,5,7,7,["WRONG VOLTAGE/INDUCTANCE","WRONG TORCH
MANIPULATION","WRONG TRAVEL SPEED"],"CONCAVE DEPOSIT - POSSIBLE
CAUSES",0,CHOICE),
    GMAW_concave_deposit(CHOICE).

GMAW_shallow_penetration(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use smaller electrodes to reach down into the narrow grooves\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_shallow_penetration(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use higher current\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

GMAW_shallow_penetration(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Allow some gap at the bottom of the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_poor_fusion(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Provide better fit-up or use a weave technique to fill up the gap\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_poor_fusion(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Remove scale/dirt/rust/paint/moisture from the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_poor_fusion(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Try a higher current and stringer bead technique\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_porosity(1) :-

```

```

makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Clean base metal with Stainless Steel Brush or chemicals\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

GMAW_porosity(2) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Check and replace the defective components of the system\n").

GMAW_porosity(3) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Replace the contact tip centering device\n").

GMAW_porosity(4) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Use recommended torch manipulation\n").

GMAW_porosity(5) :-
makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
write("Remedy: Clean the nozzle\n"), readchar(_),removewindow(1,1),removewindow(1,1),
Defect_analyzer.

GMAW_convex_deposit (1) :-
makewindow (1,5,7,"CONVEX DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Use correct voltage / wire feed\n").

GMAW_convex_deposit (2) :-
makewindow (1,5,7,"CONVEX DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Use recommended Inductance\n").

GMAW_convex_deposit (3) :-
makewindow (1,5,7,"CONVEX DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Use recommended torch Manipulation\n").

GMAW_convex_deposit (4) :-
makewindow (1,5,7,"CONVEX DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Increase Travel Speed \n").

GMAW_concave_deposit (1) :-
makewindow (1,5,7,"CONCAVE DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Use correct voltage / inductance\n").

GMAW_concave_deposit (2) :-
makewindow (1,5,7,"CONCAVE DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Use recommended torch Manipulation\n").

GMAW_concave_deposit (3) :-
makewindow (1,5,7,"CONCAVE DEPOSIT - REMEDIES",10,10,5,60),
write("Remedy: Increase Travel Speed \n").

```

```

/* DEFECT ANALYSIS FOR GTAW */

gtaw_undercuts(1) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Use Recommended Torch Manipulation\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_undercuts(2) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended shielding gas\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_spatter(1) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Adjust gas Flow\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_spatter(2) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Adjust Slope\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_spatter(3) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Adjust Voltage\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_spatter(4) :-
    makewindow (1,5,7,"SPATTER - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended shielding gas mix\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_defect(1) :-
    menu(5,5,7,7,["WRONG TORCH MANIPULATION","WRONG SHIELDING
GAS"],"UNDERCUTS - POSSIBLE CAUSES",0,CHOICE),
    gtaw_undercuts (CHOICE).

gtaw_defect(2) :-
    menu(5,5,7,7,["EXCESSIVE/INSUFFICIENT GAS FLOW","INSUFFICIENT SLOPE
SETTING","EXCESSIVE VOLTAGE","WRONG SHIELDING GAS","WRONG
ELECTRODE"],"SPATTER - POSSIBLE CAUSES",0,CHOICE),
    gtaw_spatter (CHOICE).

gtaw_defect(3) :-
    makewindow (1,5,7,"ARC BLOW - REMEDIES",5,5,20,60),
    write("Remedy 1: Use lower currents and smaller filler wire\n"),
    write("Remedy 2: Reduce Arc Length\n"),
    write("Remedy 3: Weld in direction of blow\n"),

```

```

write("Remedy 4: Change the electrical path to work by:\n"),
write("    a) Shifting work connection to other end\n"),
write("    b) Welding towards heavy tacks, finished welds.\n"),
write("        or back-stepping on long welds\n"),
write("    c) Using run-out tabs, adding steel blocks to\n"),
write("        change work current path or tacking small      plates\n"),
write("Press any key to continue ....."),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_defect(4):-
    menu(5,5,7,7,["WORK PIECE DIRTY","TRAPPED GAS/TRAVEL SPEED TOO
FAST","WRONG GAS FLOW"],"POROSITY - POSSIBLE CAUSES",0,CHOICE),
    gtaw_porosity(CHOICE).

gtaw_defect(5):-
    menu(5,5,7,7,["INSUFFICIENT GAP BETWEEN TWO PLATES","DIRTY EDGES","CURRENT
TOO LOW"],"POOR FUSION - POSSIBLE CAUSES",0,CHOICE),
    gtaw_poor_fusion(CHOICE).

gtaw_defect(6):-
    menu(5,5,7,7,["INCORRECT ELECTRODE SIZE","LOW WELDING
CURRENT","INACCURATE JOINT PREPARATION/FITUP"],"SHALLOW PENETRATION -
POSSIBLE CAUSES",0,CHOICE),
    gtaw_shallow_penetration(CHOICE).

gtaw_defect(7):-
    menu(5,5,7,7,["TOUCHING WORK WITH ELECTRODE","TUNGSTEN MELTING AND
MIXING WITH BASE METAL","SHATTERING OF TUNGSTEN WITH HIGH INPUT"],"SHALLOW
PENETRATION - POSSIBLE CAUSES",0,CHOICE),
    gtaw_tungsten_inclusion(CHOICE).

gtaw_shallow_penetration(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use smaller electrode to reach down into the narrow grooves\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_shallow_penetration(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use higher current\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

gtaw_shallow_penetration(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Allow some gap at the bottom of the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_poor_fusion(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),

```



```

        write("Remedy: Provide better fit-up or use a weave technique to fill up the gap\n").
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_poor_fusion(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Remove scale/dirt/rust/paint/moisture from the joint\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_poor_fusion(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Try a higher current and stringer bead technique\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_porosity(1) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Clean base metal with SS brush / chemicals\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

gtaw_porosity(2) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Correct travel speed\n").

gtaw_porosity(3) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Increase gas flow\n").

gtaw_tungsten_inclusion (1) :-
    makewindow (1,5,7,"TUNGSTEN INCLUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use high frequency on continuous for AC or \n start only for DC polarity\n").

gtaw_tungsten_inclusion (2) :-
    makewindow (1,5,7,"TUNGSTEN INCLUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use less current / increase diameter of the Tungsten Electrode\n").

gtaw_tungsten_inclusion (3) :-
    makewindow (1,5,7,"TUNGSTEN INCLUSION - REMEDIES",10,10,5,60),
    write("Remedy: Make sure electrode ends are properly prepared\n or increase the diameter of the
electrode \n to carry the high currents\n").

/* DEFECT ANALYSIS FOR SAW */

saw_undercuts(1) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Use Recommended Torch Manipulation\n"),
readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

saw_undercuts(2) :-
    makewindow (1,5,7,"UNDERCUTS - REMEDIES",10,10,5,60),
    write("Remedy: Use recommended shielding gas\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_defect(1) :-
    menu(5,5,7,7,["WRONG TORCH MANIPULATION","WRONG FLUX"],"UNDERCUTS -
    POSSIBLE CAUSES",0,CHOICE),
    saw_undercuts(CHOICE).

saw_defect(2):-
    menu(5,5,7,7,["WORK PIECE DIRTY","ELECTRODE CONTAMINATION","INSUFFICIENT
    FLUX COVERAGE","CONTAMINANTS IN THE FLUX","ENTRAPPED FLUX AT BOTTOM
    OF JOINT","SEGREGATION OF CONSTITUENTS","EXCESSIVE TRAVEL SPEED",
    "SLAG RESIDUE FROM TACK WELDS MADE WITH COVERED
    ELECTRODES"],"POROSITY - POSSIBLE CAUSES",0,CHOICE),
    saw_porosity(CHOICE).

saw_defect(3):-
    menu(5,5,7,7,["INSUFFICIENT GAP BETWEEN TWO PLATES","DIRTY EDGES","CURRENT
    TOO LOW"],"POOR FUSION - POSSIBLE CAUSES",0,CHOICE),
    saw_poor_fusion(CHOICE).

saw_defect(4):-
    menu(5,5,7,7,["INCORRECT ELECTRODE SIZE","LOW WELDING CURRENT",
    "INACCURATE JOINT PREPARATION/FITUP"],"SHALLOW PENETRATION - POSSIBLE
    CAUSES",0,CHOICE),
    saw_shallow_penetration(CHOICE).

saw_shallow_penetration(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use smaller electrode to reach down into the narrow grooves\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_shallow_penetration(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Use higher current\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

saw_shallow_penetration(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Allow some gap at the bottom of the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_poor_fusion(1) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Provide better fit-up or use a weave technique to fill up the gap\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

saw_poor_fusion(2) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Remove scale/dirt/rust/paint/moisture from the joint\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_poor_fusion(3) :-
    makewindow (1,5,7,"POOR FUSION - REMEDIES",10,10,5,60),
    write("Remedy: Try a higher current and stringer bead technique\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_porosity(1) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Clean base metal with SS brush / chemicals\n"),
    readchar(_),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_porosity(2) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Use clean and dry electrode\n"),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

saw_porosity(3) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Provide sufficient Flux Coverage\n"),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

saw_porosity(4) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Use Clean and Dry Flux\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

saw_porosity(5) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Use Proper Backing\n"),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_porosity(6) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Use proper joint preparation and weld
    preparation\n"),removewindow(1,1),removewindow(1,1), Defect_analyzer.

saw_porosity(7) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Reduce travel speed\n"), readchar(_),removewindow(1,1),removewindow(1,1),
    Defect_analyzer.

saw_porosity(8) :-
    makewindow (1,5,7,"POROSITY - REMEDIES",10,10,5,60),
    write("Remedy: Use electrodes which do not leave slag
    residue\n"),removewindow(1,1),removewindow(1,1), Defect_analyzer.

```

```

get_electrode_category ("E4043","0 0 1 0").
get_electrode_category ("E6012","0 0 1 0").
get_electrode_category ("E6013","0 0 1 0").
get_electrode_category ("E7013","0 0 1 0").
get_electrode_category ("E8013","0 0 1 0").
get_electrode_category ("E6030","0 1 0 0").
get_electrode_category ("E7030","0 1 0 0").
get_electrode_category ("E6016","1 0 0 0").
get_electrode_category ("E7016","1 0 0 0").
get_electrode_category ("E6010","0 0 0 1").
get_electrode_category ("E6011","0 0 0 1").
get_electrode_category ("E7010","0 0 0 1").
get_electrode_category ("E7011","0 0 0 1").

```

```

get_shielding_gas ("Ar He","1 0 0 0 0").
get_shielding_gas ("CO2","0 1 0 0 0").
get_shielding_gas ("Ar CO2","0 0 1 0 0").
get_shielding_gas ("Ar","0 0 0 1 0").
get_shielding_gas ("Ar O2","0 0 0 0 1").

```

```

get_electrode_category_name ("E4043","Rutile").
get_electrode_category_name ("E6012","Rutile").
get_electrode_category_name ("E6013","Rutile").
get_electrode_category_name ("E7013","Rutile").
get_electrode_category_name ("E8013","Rutile").
get_electrode_category_name ("E6030","Basic").
get_electrode_category_name ("E7030","Basic").
get_electrode_category_name ("E6016","Low Hydrogen").
get_electrode_category_name ("E7016","Low Hydrogen").
get_electrode_category_name ("E6010","Cellulosic").
get_electrode_category_name ("E6011","Cellulosic").
get_electrode_category_name ("E7010","Cellulosic").
get_electrode_category_name ("E7011","Cellulosic").

```

```

get_current("LOW","Rutile",3.25,100).
get_current("HIGH","Rutile",3.25,150).
get_current("LOW","Rutile",4,150).
get_current("HIGH","Rutile",4,255).
get_current("LOW","Rutile",5,260).
get_current("HIGH","Rutile",5,300).
get_current("LOW","Rutile",6.30,300).

```

```

get_current("HIGH","Rutile",6.30,390).
get_current("LOW","Basic",3.25,100).
get_current("HIGH","Basic",3.25,150).
get_current("LOW","Basic",4,130).
get_current("HIGH","Basic",4,170).

```

```

get_current("LOW","Basic",5,160).
get_current("HIGH","Basic",5,250).
get_current("LOW","Basic",6.30,260).
get_current("HIGH","Basic",6.30,300).
get_current("LOW","Low Hydrogen",3.25,100).
get_current("HIGH","Low Hydrogen",3.25,150).
get_current("LOW","Low Hydrogen",4,130).
get_current("HIGH","Low Hydrogen",4,170).
get_current("LOW","Low Hydrogen",5,160).
get_current("HIGH","Low Hydrogen",5,250).
get_current("LOW","Low Hydrogen",6.30,260).
get_current("HIGH","Low Hydrogen",6.30,300).
get_current("LOW","Cellulosic",3.25,80).
get_current("HIGH","Cellulosic",3.25,115).
get_current("LOW","Cellulosic",4,110).
get_current("HIGH","Cellulosic",4,180).
get_current("LOW","Cellulosic",5,140).
get_current("HIGH","Cellulosic",5,220).
get_current("LOW","Cellulosic",6.30,275).
get_current("HIGH","Cellulosic",6.30,300).

get_electrode ("Rutile","Cellulosic","E6010","E6012").
get_electrode ("Rutile","Cellulosic","E7010","E7012").
get_electrode ("Rutile","Cellulosic","E6011","E6013").
get_electrode ("Rutile","Cellulosic","E7011","E7013").
get_electrode ("Low Hydrogen","Cellulosic","E6010","E6016").
get_electrode ("Low Hydrogen","Cellulosic","E7010","E7016").
get_electrode ("Low Hydrogen","Cellulosic","E6011","E6016").
get_electrode ("Low Hydrogen","Cellulosic","E7011","E7016").
get_electrode ("Low Hydrogen","Rutile","E4043","E6016").

consult_parameter_data:-
    existfile("par_sel.db"),!,
    consult("par_sel.db",parameter).
consult_parameter_data.

get_electrode_size_category(3.25," 0 0 1 ").
get_electrode_size_category(4," 0 0 1 ").
get_electrode_size_category(5," 0 1 0 ").
get_electrode_size_category(6.30," 1 0 0 ").

get_speed_category(Speed,Speed_Cat) :-
    Speed <= 10,
    Speed_Cat = " 0 0 1 ".

get_speed_category(Speed,Speed_Cat) :-
    Speed > 10,
    Speed < 20,
    Speed_Cat = " 0 1 0 ".

```

```

get_speed_category(Speed,Speed_Cat) :-
    Speed >= 20,
    Speed_Cat = " 1 0 0 ".

get_Current_category(Current,Current_Cat) :-
    Current <= 150,
    Current_Cat = " 0 0 1 ".

get_Current_category(Current,Current_Cat) :-
    Current > 150,
    Current < 300,
    Current_Cat = " 0 1 0 ".

get_Current_category(Current,Current_Cat) :-
    Current >= 300,
    Current_Cat = " 1 0 0 ".

consult_wps_process :-
    existfile("wps.dba"),!,
    consult("wps.dba").
consult_wps_process.

consult_defects :-
    existfile("defects.dba"),!,
    consult("defects.dba",defect).
consult_defects.

goal
    defect_analyzer_main_menu.

```

### **E.7 TROUB\_SH.PRO - Prolog Program for Trouble Shooting**

```

project "neurex"
include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu2.pro"

database - wps
    wps_process(string,string)

predicates
    trouble_shoot (integer)
    welding_set(integer)
    arc_trouble (integer)
    wire_feeder (integer)

```

```

flux_feeder (integer)
miscellaneous (integer)
welding_set_causes(integer)
current_causes(integer)
welding_set_stoppage_causes(integer)
bcause(integer)
wire_feed_stops (integer)
flux_feed_stops (integer)
flux_feed_stops2 (integer)
flux_feed_stops4 (integer)
flux_feed_stop (integer)
wire_stubs (integer)
excessive_spatter(integer)
bad_arc_starts(integer)
fluid_puddle(integer)
rigid_puddle(integer)
convex_deposit(integer)
concave_deposit(integer)
write_wps_proc (string)
consult_wps_proc
trouble_shooter_main_menu
trouble_shoot_choice (integer)

clauses

trouble_Shooter_Main_Menu :-
    menu(5,5,7,7,["INTERACTIVE","AUTOMATED WPS TIPS","QUIT"],"SELECT AN
ACTION",0,CHOICE),
    trouble_shoot_choice (CHOICE).

trouble_shoot_choice (1):-
    Trouble_shooter.

trouble_shoot_choice (2):-
    consult_wps_proc,
    wps_process(_,Process),
    write_wps_proc(Process).

trouble_shoot_choice (3):-
    exit.

consult_wps_proc:-
    existfile("wps.dba"),!,
    consult("wps.dba",wps).
consult_wps_proc.

write_wps_proc ("SMAW") :-
    existfile("wps_gen.out"),
    openappend(outfile,"wps_gen.out"),

```

```

writedevicе(outfile),
write("\n\nTROUBLE SHOOTING TIPS \n"),
write("===== ").nl,
write("WELD REMAINS IN BALLS: Check Electrode Type").nl,
write("CRACKING: Bake electrodes").nl,
write("SPATTER: Reduce Voltage & Check Electrode Type").nl,
write("SLUGGISH ARC: Increase Current").nl,
write("LOUD & SPATTERY ARC: Reduce Current").nl,
write("UNSTABLE ARC: Shorten Arc").nl,
closefile(outfile),
writedevicе(screen),
makewindow(1,9,5,"TROUBLE SHOOTING TIPS",0,0,25,60),
write("\n\nSMAW TROUBLE SHOOTING TIPS \n"),
write("===== ").nl,
write("WELD REMAINS IN BALLS: Check Electrode Type").nl,
write("CRACKING: Bake electrodes").nl,
write("SPATTER: Reduce Voltage & Check Electrode Type").nl,
write("SLUGGISH ARC: Increase Current").nl,
write("LOUD & SPATTERY ARC: Reduce Current").nl,
write("UNSTABLE ARC: Shorten Arc").nl,

write_wps_proc ("SMAW") :-
    openwrite(outfile,"wps_gen.out"),
    writedevicе(outfile),
    write("\n\nTROUBLE SHOOTING TIPS \n"),
    write("===== ").nl,
    write("WELD REMAINS IN BALLS: Check Electrode Type").nl,
    write("CRACKING: Bake electrodes").nl,
    write("SPATTER: Reduce Voltage & Check Electrode Type").nl,
    write("SLUGGISH ARC: Increase Current").nl,
    write("LOUD & SPATTERY ARC: Reduce Current").nl,
    write("UNSTABLE ARC: Shorten Arc"),
    closefile(outfile).

write_wps_proc ("GMAW") :-
    existfile("wps_gen.out"),!,
    openappend(outfile,"wps_gen.out"),
    writedevicе(outfile),
    write("\n\nTROUBLE SHOOTING TIPS \n"),
    write("===== ").nl,
    write("STUBBING: Increase the voltage and decrease the wire feed rate").nl,
    write("CONVEX BEADS: Increase the welding speed and decrease the wire feed rate").nl,
    write("CONCAVE BEADS: Decrease the welding speed and increase the wire feed rate").nl,
    closefile(outfile).

write_wps_proc ("GMAW") :-
    openwrite(outfile,"wps_gen.out"),
    writedevicе(outfile),
    write("\n\nTROUBLE SHOOTING TIPS \n"),

```



```

write("===== ").nl,
write("STUBBING: Increase the voltage and decrease the wire feed rate").nl,
write("CONVEX BEADS: Increase the welding speed and decrease the wire feed rate").nl,
write("CONCAVE BEADS: Decrease the welding speed and increase the wire feed rate").nl,
closefile(outfile).

Trouble_Shooter :-
    menu(5,5,7,7,["WELDING SET RELATED","ARC RELATED PROBLEMS","WIRE FEEDER
RELATED","FLUX FEEDER RELATED","MISCELLANEOUS"],"TROUBLE SHOOTER".0,CHOICE),
    trouble_shoot (CHOICE).

/* TROUBLE SHOOTING FOR SMAW */

trouble_shoot (1) :-
    menu(5,5,7,7,["DOES NOT START","STARTS BUT BLOWS FUSE","WELDS BUT STOPS
SOON","WOULD NOT SHUT OFF",
    "POLARITY SWITCH DOES NOT TURN","ARCING AT GROUND
CLAMP","ELECTRODE HOLDER BECOMES HOT","QUIT"],
    "WELDING SET".0,CHOICE),
    welding_set(CHOICE).

trouble_shoot (2) :-
    menu(5,5,7,7,["VARIABLE AND SLUGGISH","LOUD AND SPATTERY","UNSTABLE &
GOES OUT","QUIT"],
    "ARC RELATED TROUBLE".0,CHOICE),
    arc_trouble (CHOICE).

trouble_shoot (3) :-
    menu(5,5,7,7,["Wire Feed Unit stops feeding","Wire Feeds but no arcing","Wire electrode stubs"
],"WIRE FEEDER TROUBLE".0,CHOICE),
    wire_feeder (CHOICE).

trouble_shoot (4) :-
    menu(5,5,7,7,["FLUX FEEDING STOPS","FLUX STOPPAGE NOT IN GUN",
    "EXCESSIVE AIR BLOW/UNEVEN FLUX FLOW","FLUX IN TANKS GETS
WET"],"FLUX FEEDER TROUBLE".0,CHOICE),
    flux_feeder (CHOICE).

trouble_shoot (5) :-
    menu(5,5,7,7,["SPATTER","BAD ARC STARTS","FLUID PUDDLE", "RIGID
PUDDLE","CONVEX DEPOSIT","CONCAVE DEPOSIT",
    "WELD REMAINS IN BALLS"],"WELDING SET".0,CHOICE),
    miscellaneous(CHOICE).

welding_set(1) :-
    menu(10,5,7,7,["Line Switch not turned 'ON',"Supply Line Fuse blown","Power circuit dead",
    "Overload relay Tripped","Loose or broken power, electrode or ground lead",
    " Wrong Voltage","Polarity switch not centered"," Open circuit to starter switch"],

```

```

        "WELD SET CAUSES",0,CHOICE),
welding_set_causes(CHOICE).

welding_set(2) :-
    menu (10,5,7,7,["Short circuit in motor or other connection","Fuse too small"],"CHOOSE",0,Choice),
    current_causes(Choice).

welding_set(3) :-
    menu (10,5,7,7,["Proper ventilation hindered","Overloading due to welding in excess of rating",
        "Fan may be inoperative","Motor generator set problems","Wrong direction of rotation",
        "Brushes worn or missing","Wrong driving speed"],"CHOOSE",0,Choice),
    welding_set_stoppage_causes(Choice).

welding_set(4) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause : Line Switch may have failed mechanically"),
    write("\nRemedy : Check it and replace it if found defective"),
    readchar(_), removewindow().

welding_set(5):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nCause : Contacts may have become rough and pitted due to improper \n\nturning under load\n"),
    write("\nRemedy : Replace Polarity switch "),
    readchar(_),removewindow().

welding_set(6) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nCause : Connection may be loose or spring may be weak \n"),
    write("\nRemedy : Tighten connection or replace Clamp\n"),
    readchar(_),removewindow().

welding_set(7) :- /* electrode holder becomes hot */
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause : Loose Connection to Holder / The duty cycle may be inadequate \n"),
    write("Remedy : Tighten connection or replace Holder and use proper duty cycle\n"),
    readchar(_),removewindow().

welding_set(8).

arc_trouble (1) :-
    menu(5,5,7,7,["CURRENT TOO LOW","LOW LINE VOLTAGE","WELDING LEADS TOO SMALL",
        "IMPROPER CONNECTIONS","IMPROPER WELD SET BRUSHES OR SPRINGS",
        "ROUGH OR DIRTY COMMUTATOR"],"CHOOSE",0, CHOICE),
    bcause(CHOICE).

arc_trouble (2) :-
    menu(5,5,7,7,["Current setting is too high","Polarity may be wrong"],"CHOOSE",0,Choice),
    current_causes (Choice).

```

```

arc_trouble (3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause : Arc may be too Long"),
    write("Remedy: Shorten arc for proper penetration"),removewindow().

arc_trouble (4).

current_causes(1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause : Current Setting is too High"),
    write("\nRemedy: Check setting and output with ammeter "),
    readchar(_),removewindow().

current_causes(2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause: Polarity is wrong\n"),
    write("Remedy: Check polarity ; Try reversing or an electrode of opposite polarity"),
    readchar(_),removewindow().

welding_set_causes(1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Place Line switch in 'ON' position\n"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Replace the Fuse \n"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Check input Voltage \n"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(4) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check for overloading and switch on the Relay"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(5) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Replace lead or tighten and repair connection \n"),
    beep(),
    readchar(_),removewindow().

```

```

welding_set_causes(6) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check input voltage against instructions \n"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(7) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Center switch handle on +, -, or AC\n"),
    beep(),
    readchar(_),removewindow().

welding_set_causes(8):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Repair starter Button \n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Make sure all case openings are free for proper air circulation\n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Operate Welder at rated load and duty cycle\n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Check leads and connections of fan\n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(4) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check connection diagram "),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(5) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check brushes for pressure on commutator\n"),
    beep(),
    readchar(_),removewindow().

```

```

welding_set_stoppage_causes(6) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check name plate for correct motor speed\n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(7) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \n Clean thermostat's coils and other components\n"),
    beep(),
    readchar(_),removewindow().

welding_set_stoppage_causes(8) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \n I can't help you in troubleshooting.removewindow(). Please consult a Mechanic"),nl,
    beep(),
    readchar(_),removewindow().

bcause(1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Check recommended currents for rod type and size being used\n"),
    beep(),
    readchar(_),removewindow().

bcause(2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\n Remedy:Check with power company\n"),
    beep(),
    readchar(_),removewindow().

bcause(3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Use recommended Welding Leads \n"),
    beep(),
    readchar(_),
    removewindow().

bcause(4) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check all connections.removewindow(). Clean, Repair or Replace as required"),
    beep(),
    readchar(_),removewindow().

bcause(5) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write(" \nRemedy: Check and Repair \n"),
    beep(),
    readchar(_),removewindow().

```

```

because(6) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("\nRemedy: Turn down or clean Commutator\n"),
    beep(),
    readchar(_),removewindow().

wire_feeder (1) :-
    menu(5,5,7,7,["Electrical Fuse Blown","Control Relay of MIG gun defective",
        "Loss of wire feed roller tension"],"WIRE FEED UNIT STOPS - POSSIBLE
CAUSES",0,CHOICE),
    wire_feed_stops (CHOICE).

wire_feeder (2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write ("Cause: Electrical fuse blown in primary power, power source"),nl,
    write ("      or wire feed unit"),nl,
    write("Remedy: Have qualified electrician check electric power circuits"),nl,
    readchar(_),removewindow().

wire_feeder (3) :-
    menu(5,5,7,7,["EXCESS SLOPE","LOW VOLTAGE","HIGH WIRE FEED","QUIT"],
        "WIRE FEEDER STOPS - POSSIBLE CAUSES",0,CHOICE),
    wire_stubs(CHOICE).

wire_stubs (1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write ("Remedy: Excessive Slope"),nl,
    readchar(_), removewindow().

wire_stubs (2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write ("Remedy: Increase Voltage"),nl,
    readchar(_), removewindow().

wire_stubs (3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write ("Remedy: Decrease wire feed"),nl,
    readchar(_), removewindow().

miscellaneous(1) :-
    menu(5,5,7,7,["EXCESS GAS FLOW","INSUFFICIENT SLOPE SETTING",
        "EXCESSIVE VOLTAGE","WRONG SHIELDING GAS","WRONG ELECTRODE"],
        "EXCESSIVE SPATTER - POSSIBLE CAUSES",0,CHOICE),
    excessive_spatter (CHOICE).

miscellaneous (2) :-
    menu(5,5,7,7,["WRONG VOLTAGE","WRONG SLOPE","WRONG INDUCTANCE","GLASS
SLAG"],
        "BAD ARC STARTS - POSSIBLE CAUSES",0,CHOICE),

```

```

        bad_arc_starts(CHOICE).

miscellaneous (3) :-
    menu(5,5,7,7,["WRONG INDUCTANCE","WRONG ELECTRODE","WRONG SHIELDING
GAS"],
        "FLUID PUDDLE - POSSIBLE CAUSES",0,CHOICE),
    fluid_puddle(CHOICE).

miscellaneous (4) :-
    menu(5,5,7,7,["WRONG INDUCTANCE","WRONG ELECTRODE","WRONG SHIELDING
GAS"],
        "RIGID PUDDLE - POSSIBLE CAUSES",0,CHOICE),
    rigid_puddle(CHOICE).

miscellaneous (5) :-
    menu(5,5,7,7,["WRONG VOLTAGE OR WIRE FEED","WRONG INDUCTANCE","WRONG
TORCH MANIPULATION","WRONG TRAVEL SPEED"],
        "CONVEX DEPOSIT - POSSIBLE CAUSES",0,CHOICE),
    convex_deposit(CHOICE).

miscellaneous (6) :-
    menu(5,5,7,7,["WRONG VOLTAGE OR INDUCTANCE","WRONG TORCH
MANIPULATION","WRONG TRAVEL SPEED"],"CONCAVE DEPOSIT - POSSIBLE
CAUSES",0,CHOICE),
    concave_deposit(CHOICE).

miscellaneous (7) :- /* weld remains in balls */
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Cause : Electrode being used may be wrong\n"),
    write("Remedy : Use proper electrode\n"),
    readchar(_),removewindow().

wire_feed_stops (1):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Have qualified electrician check electric power circuits"),
    readchar(_),removewindow().

wire_feed_stops (2):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Replace defective relay switches"),
    readchar(_),removewindow().

wire_feed_stops (3):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Inspect and replace all defects feed drive roller"),
    readchar(_),removewindow().

excessive_spatter (1):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),

```

```
write("Remedy: Adjust Gas Flow"),  
readchar(_),removewindow().
```

```
excessive_spatter (2):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Slope"),  
readchar(_),removewindow().
```

```
excessive_spatter (3):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Voltage"),  
readchar(_),removewindow().
```

```
excessive_spatter (4):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended gas coverage"),nl,  
readchar(_),removewindow().
```

```
excessive_spatter (5):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended wire electrodes"),nl,  
readchar(_),removewindow().
```

```
bad_arc_starts (1):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Voltage"),  
readchar(_),removewindow().
```

```
bad_arc_starts (2):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Slope"),  
readchar(_),removewindow().
```

```
bad_arc_starts (3):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Inductance"),  
readchar(_),removewindow().
```

```
bad_arc_starts (4):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Chip and brush welds"),  
readchar(_),removewindow().
```

```
fluid_puddle (1):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Inductance"),  
readchar(_),removewindow().
```

```
fluid_puddle (2):-
```



```
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended wire electrode"),  
readchar(_),removewindow().
```

```
fluid_puddle (3):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended gas coverage"),  
readchar(_),removewindow().
```

```
rigid_puddle (1):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Inductance"),  
readchar(_),removewindow().
```

```
rigid_puddle (2):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended wire electrode"),  
readchar(_),removewindow().
```

```
rigid_puddle (3):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended gas coverage"),  
readchar(_),removewindow().
```

```
convex_deposit (1):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust Controls"),  
readchar(_),removewindow().
```

```
convex_deposit (2):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Adjust inductance"),  
readchar(_),removewindow().
```

```
convex_deposit (3):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Use recommended torch motion"),  
readchar(_),removewindow().
```

```
convex_deposit (4):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),  
write("Remedy: Increase travel speed"),  
readchar(_),removewindow().
```

```
concave_deposit (1):-  
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
```

```

write("Remedy: Adjust Controls").
readchar(_),removewindow().

concave_deposit (2):-
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
write("Remedy: Use recommended torch motion"),
readchar(_),removewindow().

concave_deposit (3):-
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
write("Remedy: Decrease torch travel speed"),
readchar(_),removewindow().

flux_feeder (1) :-
menu(5,5,7,7,["FLUX TUBE IN THE GUN MAY BE BLOCKED","MAGNETIC PARTICLES
MAY CAUSE BRIDGING"],"FLUX FEED UNIT STOPS - POSSIBLE CAUSES",0,CHOICE),
flux_feed_stops (CHOICE).

flux_feeder (2) :-
menu(5,5,7,7,["PIECE OF SLAG IN TANK","FLUX TANK OUTLET MAY BE CLOGGED",
"PRESSURE IN THE TANK MAY BE SET IMPROPERLY"],"FLUX FEEDER PROBLEMS -
POSSIBLE CAUSES",0,CHOICE),
flux_feed_stops2 (CHOICE).

flux_feeder (3) :-
menu(5,5,7,7,["TANK MAY BE EMPTY","FLUX MAY BE FALLING AWAY FROM THE
WELD FASTER THAN IT IS BEING FED",
"PRESSURE IN THE TANK MAY BE TOO HIGH"],"FLUX FEEDER PROBLEMS - POSSIBLE
CAUSES",0,CHOICE),
flux_feed_stop (CHOICE).

flux_feeder (4) :-
menu(5,5,7,7,["THERE MAY BE WATER IN THE AIR LINE","THE COPPER BLEEDER MAY
BE CLOGGED"],"FLUX FEED UNIT STOPS - POSSIBLE CAUSES",0,CHOICE),
flux_feed_stops4(CHOICE).

flux_feed_stops2 (1):-
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
write("Remedy: Work back along the hose squeezing it until flux can be\n"),
write("      felt. Shake the hose and feel for slag at this point. Blow\n"),
write("      out hose with air if necessary"),nl,
readchar(_),removewindow().

flux_feed_stops2 (2):-
makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
write("Remedy: If there is no Flux in the tank, check the tank outlet for\n"),
write("      large pieces of slag or paper"),nl,
readchar(_),removewindow().

```

```

flux_feed_stops2 (3):-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Set pressure regulator for 26-30 lbs/sq.in.\n      For gun cable extension assembly, set
pressure for 55-60 lbs/sq. in."),nl,
    readchar(_),removewindow().

flux_feed_stops (1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Remove the gun from the cable. Check the tube \n      in the gun and the cable handle"),
    readchar(_),removewindow().

flux_feed_stops (2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Pass flux over a magnetic separator when \n      filling the flux tank"),
    readchar(_),removewindow().

flux_feed_stops (3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Decrease torch travel speed"),
    readchar(_),removewindow().

flux_feed_stop (1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Refill the tank"),
    readchar(_),removewindow().

flux_feed_stop (2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Alter procedure or make a flux dam"),
    readchar(_),removewindow().

flux_feed_stop (3) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Set pressure regulator for 26-30 lbs./sq.in.\n      or 55-60 lbs/sq.in. for gun-cable extension
assembly"),
    readchar(_),removewindow().

flux_feed_stops4 (1) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Blow out air lines before connecting them to the tank"),
    readchar(_),removewindow().

flux_feed_stops4 (2) :-
    makewindow (1,5,7,"TROUBLE SHOOTING",10,10,5,60),
    write("Remedy: Make sure that a slight amount of air is escaping from \n      the crimped end of the
copper tube below the flux tank"),
    readchar(_),removewindow().

```

## E.8 INTERFAC.PRO - Prolog Program for User Interface

project "neurex"

```
include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "menu2.pro"
include "status.pro"
include "pulldown.pro"
include "editbp.pro"
```

Database - menu\_style  
frame\_is(attr)

domains  
choice = integer

Predicates  
msg(ROW,COL,STRING)  
other\_than\_style\_change(integer,integer)  
GO  
do\_par\_scl (integer)  
do\_std (integer)  
%do\_data\_manipulation (integer)  
%do\_std\_data\_editing (Choice)  
%do\_std\_data\_listing (Choice)  
%do\_std\_data\_deletion (Choice)  
%electrode\_data\_entry  
firm\_data\_entry (integer)  
electrode\_data (integer)  
train\_neural\_net (integer)  
do\_edit\_bp(integer)  
get\_process(integer,string)  
firm\_machine\_data\_entry  
consult\_firm\_machine\_data  
save\_firm\_machine\_data  
consult\_firm\_labor\_data  
save\_firm\_labor\_data

goal  
GO.

CLAUSES

/\* Clauses for the Main Menu \*/

GO :-  
textmode(ROWS,COLS),  
ROWS1=ROWS-1,

```

makewindow(1,23,0,"",0,0,ROWS1,COLS).
makestatus(112," Select with arrows or use first upper case letter").
field_str(15,13,50,"015 PRESS ESC for exiting any Pull-down menu").
scr_attr(15,13,220).
asserta(frame_is(0)).
repeat.
frame_is(Fattr).
pulldown(66,Fattr.[
    curtain("Data Input",[
        elem("Enter Standard Data",top_menu,top_menu),
        elem("Enter Firm Data",top_menu,top_menu),
        elem("Enter Job Data",top_menu,top_menu)
    ],32),
    curtain("Expert-Select",[
        elem("Process Selector",top_menu,top_menu),
        elem("Parameter Evaluator",top_menu,top_menu),
        elem("WPS Generator",top_menu,top_menu),
        elem("Economic Evaluator",top_menu,top_menu),
        elem("Defect Analyzer",top_menu,top_menu),
        elem("Trouble Shooter",top_menu,top_menu)
    ],32),
    curtain("Neural-Net",[
        elem("Retrain Neural Network",top_menu,top_menu),
        elem("Test Neural Network",top_menu,top_menu),
        elem("Change NN Parameters",top_menu,top_menu)
    ],7),
    curtain("Help",[
        elem("Using the System",sub_menu,sub_menu),
        elem("Data Entry",sub_menu,sub_menu),
        elem("Expert-Select",sub_menu,sub_menu),
        elem("Neuro-Check",top_menu,top_menu)
    ],48),
    elem("Quit",quit,quit)
],CH,SUBCH),
    other_than_style_change(CH, SUBCH),!.
    write("\n  CH = ",CH,"\n SUBCH = ",SUBCH,\n).

```

/\* Clauses for options of the Main Menu \*/

pdwaction(1,1):- std\_data.

pdwaction(1,2):- firm\_data.

pdwaction(1,3):- input\_data.

pdwaction(2,1):- Process\_Select.

pdwaction(2,2):-

```

Parameter_Select,
system("testbp -R par_sel"),
existfile("par_sel.out"),!,
openread(infile,"par_sel.out"),
readdevice(infile),
readln(X),
closefile(infile),
readdevice(Keyboard),
makewindow(4,5,7,"Result of Validation",10,0,15,40),
write(X),
write("\n\nAbove value > 0.6 => Valid Process Parameters"),nl,
write("Above value < 0.6 => Invalid Process Parameters"),nl,
write("Press any key to continue ...."),
readchar(_),
removewindow(4,1).

pdwaction(2,3):- wps_gen.

pdwaction(2,4):- econ_eval_gen.

pdwaction(2,5) :-
    system("testbp -R def_anal"),
    existfile("def_anal.out"),!,
    openread(infile,"def_anal.out"),
    readdevice(infile),
    readln(X),
    closefile(infile),
    readdevice(Keyboard),
    makewindow(4,5,7,"Result of Validation",10,0,15,40),
    write(X),
    write("\n\nAbove value > 0.6 => Valid Process Parameters"),nl,
    write("Above value < 0.6 => Invalid Process Parameters"),nl,
    write("Press any key to continue ...."),
    readchar(_),
    removewindow(4,1),
    Defect_analyzer.

pdwaction(2,6). % :- Trouble_Shooter.

pdwaction(3,1):-
    menu(15,15,10,10,["Train Process Select Neural Net",
        "Train Parameter Select Neural Net",
        "Train Defect Analysis Neural Net"],
        "Choose",0,Choice),
    train_neural_net (Choice).

pdwaction(3,2):-
    system("testbp -R proc_sel").

```

```

pdwaction(3,3):-
    clearwindow,
    menu(15,15,10,10,["Edit Process Select NN Parameters",
        "Edit Parameter Select NN Parameters",
        "Edit Defect Analysis NN Parameters"],
        "Choose",0,Choice),
    do_edit_bp (Choice).

pdwaction(4,2):-
    msg(4,45,"Follow the Sequence\n  Job Entry \n  Process Selection \n  Parameter Evaluation \n
    Neuro-Check").

pdwaction(4,3):-
    retract(frame_is(0)),!,
    asserta(frame_is(66)).

pdwaction(4,3):-
    retract(frame_is(66)),!,
    asserta(frame_is(0)).

pdwaction(5,0):- exit.

do_edit_bp (1) :-
    edit_bp("proc_sel.def").

do_edit_bp (2) :-
    clearwindow,
    menu(15,15,10,10,["Edit NN Parameters for SMAW Parameter Selection",
        "Edit NN Parameters for GMAW Parameter Selection",
        "Edit NN Parameters for GTAW Parameter Selection",
        "Edit NN Parameters for SAW Parameter Selection"],
        "Choose",0,Choice),
    get_process(Choice, Process),
    edit_bp(Process).

do_edit_bp (3) :-
    edit_bp("def_anal.def").

get_process(1,"SMAW_par.def").
get_process(2,"GMAW_par.def").
get_process(3,"gtaw_par.def").
get_process(4,"saw_par.def").

msg(R,C,S):-
    makestatus(112,"Press any key"),
    makewindow(1,32,4,"",R,C,5,30),
    window_str(S),

```

```

        readkey(_),
        removewindow,
        removestatus.

other_than_style_change(4,3) :- !, fail.
other_than_Style_change(_,_).


std_data :-
    menu(5,5,10,10,["EDIT EXISTING DATA","INSERT NEW DATA","DELETE EXISTING
DATA","LIST EXISTING DATA"],"Choose",0,Choice1),
    do_data_manipulation(Choice1).

do_data_manipulation(1):-
    menu(5,5,10,10,["SMAW","GMAW","GTAW","SAW"],"Choose",0,Choice),
    do_std_data_editing(Choice).

do_data_manipulation(2):-

do_data_manipulation(3):-
    menu(5,5,10,10,["SMAW","GMAW","GTAW","SAW"],"Choose",0,Choice),
    do_std_data_deletion (Choice).

do_data_manipulation(4):-
    menu(5,5,10,10,["SMAW","GMAW","GTAW","SAW"],"Choose",0,Choice),
    do_std_data_listing (Choice).

do_std (1) :-
    SMAW_data_entry.

do_std (2) :-
    GMAW_data_entry.

do_std (3) :-
    gtaw_data_entry.

do_std (4) :-
    saw_data_entry.

do_std_data_editing (1) :-
    SMAW_data_editing.

do_std_data_editing (2) :-
    GMAW_data_editing.

do_std_data_editing (3) :-
    gtaw_data_editing.

```



do\_std\_data\_editing (4) :-  
    saw\_data\_editing.

do\_std\_data\_deletion (1) :-  
    SMAW\_data\_deletion.

do\_std\_data\_deletion (2) :-  
    GMAW\_data\_deletion.

do\_std\_data\_deletion (3) :-  
    gtaw\_data\_deletion.

do\_std\_data\_deletion (4) :-  
    saw\_data\_deletion.

do\_std\_data\_listing (1) :-  
    SMAW\_data\_listing.

do\_std\_data\_listing (2) :-  
    GMAW\_data\_listing.

do\_std\_data\_listing (3) :-  
    gtaw\_data\_listing.

do\_std\_data\_listing (4) :-  
    saw\_data\_listing.

Parameter\_Select :-  
    menu(5,5,10,10,["SMAW","GMAW","GTAW","SAW"],"Choose",0,Choice),  
    do\_par\_sel (Choice).

do\_par\_sel (1) :-  
    par\_sel\_SMAW.

do\_par\_sel (2) :-  
    par\_sel\_GMAW.

do\_par\_sel (3) :-  
    par\_sel\_gtaw.

do\_par\_sel (4) :-  
    par\_sel\_saw.

electrode\_data (1) :-  
    electrode\_data\_SMAW.

electrode\_data (2) :-

```

electrode_data_GMAW.

electrode_data (3) :-
    electrode_data_gtaw.

electrode_data (4) :-
    electrode_data_saw.

firm_data_entry (1) :-
    menu(5,5,10,10,["Electrode Data for SMAW",
                    "Electrode data for GMAW",
                    "Electrode Data for SAW"],
        "Choose",0,Choice),
    electrode_data (Choice).

firm_data_entry (3):-
    firm_labor_data_input.

firm_data_entry (2) :-
    firm_machine_data_entry.

consult_firm_machine_data :-
    existfile("machine.DBA"),!,
    consult("machine.DBA").
consult_firm_machine_data.

save_firm_machine_data :-
    save("machine.dba").

firm_machine_data_entry :-
    makewindow(1,5,7,"Firm Machine Data",5,5,15,45),
    write("Please enter machine name/number:"),
    readln(MachineNo),
    write("\nPlease enter Process supported:"),
    readln(Process),
    consult_firm_machine_data,
    assert(firm_machine_data(MachineNo,Process)),
    save_firm_machine_data,
    removewindow.

firm_labor_data_input :-
    makewindow(1,5,7,"Firm labor Data",5,5,15,45),
    write("Please enter Labor Cost/hour:"),
    readreal(Labor_cost),
    write("\nPlease enter Overhead Cost/hour:"),
    readreal(Overhead_cost),
    consult_firm_labor_data,
    assert(firm_labor_data(Labor_cost,Overhead_cost)),
    save_firm_labor_data,

```

```

        removewindow.

consult_firm_labor_data :-
    existfile("labor.DBA"),!,
    consult("labor.DBA").
consult_firm_labor_data.

save_firm_labor_data :-
    save("labor.dba").

train_neural_net (1):-
    system("testbp -L proc_sel").

train_neural_net (2):-
    system("testbp -L par_sel").

train_neural_net (3):-
    system("testbp -L def_anal").

train_neural_net (4).

firm_data :-
    menu(5,5,10,10,["Cost Data for Supplies",
                    "Data for Welding Sets",
                    "Cost Data for Labor & Overheads"],
        "Choose".0,Choice).
    firm_data_entry (Choice).

```

### **E.9 STD\_DATA.PRO - Prolog Program for Standard Data Entry**

```

%race saw_data, SAW_DATA11
project "neurex"

```

```

include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "status.pro"

```

```

DOMAINS
FIELD_NAME = SYMBOL
TYPE       = int(); str(); real()

```

```

/*****
*                               *
*           Database Predicates           *
*                               *
*****/

```

```

DATABASE - name

```

```

field(FIELD_NAME, TYPE, ROW, COL, LEN)
txtfield(ROW, COL, LEN, STRING)
value(FIELD_NAME, STRING)
windowsize(ROW, COL)

/*****
*                               *
*               Include Tools   *
*                               *
*****/

include "menu.pro"
#include "ui\status.pro"
include "vscrhnd.pro"           % Or vscrhnd.pro
/*
database - std_data1
    SMAW_data1(string,real,string,real,real,real,real,real,real,real,real,real,string)
    GMAW_data1(string,real,string,real,real,real,real,real,real,real,real,real,real,string)
    gtaw_data1(string,real,string,real,real,real,real,real,real,real,real,real,real,string)
    saw_data1(string,real,string,real,real,real,real,real,real,real,real,real,real,string)
*/
database - temp11
    temp11(string)
    temp12(string)
    temp13(string)
    temp14(string)

predicates
    consult_std_data
    save_std_data
    get_process(integer,string)
    get_position(integer,string)
    get_weld_type(integer,string)
    SMAW_data11(real,real,string,string,real,real,string)
    GMAW_data11(real,real,string,real,string,real,string)
    gtaw_data11(real,real,string,real,string,real,string)
    saw_data11(real,real,string,real,string,real,string)
%    put_SMAW_data(string,integer,string,integer,real,integer,real,real,real,real,real,real,string)

CLAUSES
/*****
*                               *
*               Field Actions   *
*                               *
*****/

field_action("Weld_type") :-
    menu(5,5,10,10,["FILLET","BUTT"],"Choose",0,Choice),
    get_weld_type(Choice,Weld_type),
    write(Weld_type),
    retractall(temp12(_)),

```

```

        assert(temp12(Weld_type)).

field_action ("Position") :-
    menu(5,5,10,10,["FLAT","HORIZONTAL","VERTICAL","DOWNWARD","OVERHEAD"],"Choose",0,Choice),
    get_position (Choice,Position),
    write (Position),
    retractall(temp13(_)),
    assert(temp13(Position)).

field_action(_) :- fail.

/*****
*                               Field Values                               *
*****/

field_value(FIELD_NAME, VALUE) :- value(FIELD_NAME, VALUE), !.

/*****
*                               No Input Fields                               *
*****/

noinput(_) :- fail.

get_process(1,"SMAW").
get_process(2,"GMAW").
get_process(3,"GTAW").
get_process(4,"SAW").

get_position(1,"FLAT").
get_position(2,"HORIZONTAL").
get_position(3,"VERTICAL").
get_position(4,"DOWNWARD").
get_position(5,"OVERHEAD").

get_weld_type (1,"FILLET").
get_weld_type (2,"BUTT").

save_std_data :-
    save("std_data.db").

consult_std_data:-
    existfile("std_data.db"),!,
    consult("std_data.db").
consult_std_data.

SMAW_data_entry:-
    consult_std_data,

```

```

consult("SMAW1_dat.scr",name),
makestatus(112,"Fill Out the Fields & Press F10 to END"),
createwindow(on),
scrhnd(on, _),
clearwindow,
value("Process",PROCESS),
%
value("Weld_type",WELD_TYPE),
%
value("Electrode",ELECTRODE),
value("Leg_length",LEG_LENGTH),
%
value("Position",POSITION),
value("Runs",RUNS),
value("Elec_size",ELEC_SIZE),
clearwindow,
scrhnd(off, _),
retractall(_,name),
removewindow,
temp12(WELD_TYPE),
temp13(POSITION),
str_real(ELEC_SIZE,Elec_S),
str_real(LEG_LENGTH,Leg_L),
str_real(RUNS,Run),
consult("SMAW2_dat.scr",name),
SMAW_data11(Run,1,WELD_TYPE,ELECTRODE,Elec_S,Leg_L,Position),
save_std_data.

```

SMAW\_data11 (RUNS,Y,WELD\_TYPE,ELECTRODE,Elec\_S,Leg\_L,Position) :-

```

RUNS >= Y,
makestatus(112,"Fill Out the Fields & Press F10 to END"),
createwindow(on),
scrhnd(on, _),
clearwindow,
value("Voltage",VOLTAGE),
value("Current",CURRENT),
value("Speed",SPEED),
value("Num_elec",NUM_ELEC),
value("Weight",WEIGHT),
value("Over_time",OVER_TIME),
value("Arc_time",ARC_TIME),
clearwindow,
scrhnd(off, _),
removewindow,
shiftwindow(81),
temp12(WELD_TYPE),
temp13(POSITION),
str_real(VOLTAGE,VOLT),
str_real(CURRENT,Curr),
str_real(SPEED, W_Speed),
str_real(NUM_ELEC,Num_E),
str_real(ARC_TIME,Arc_T),

```

```

str_real(OVER_TIME,Over_T),
str_real(WEIGHT,Wt),
assert(SMAW_data(WELD_TYPE,Leg_L,POSITION,Y,Elec_S,VOLT,
    Curr,W_Speed,Num_E,Wt,Arc_T,Over_T,ELECTRODE)),
%retractall(_std_data1),
X = Y + 1,
save_std_data,
SMAW_data11 (RUNS,X,WELD_TYPE,ELECTRODE,Elec_S,Leg_L,Position).

```

GMAW\_data\_entry:-

```

consult_std_data,
consult("GMAW1_dat.scr",name),
makestatus(112,"Fill Out the Fields & Press F10 to END"),
createwindow(on),
scrhnd(on,_),
clearwindow,
% value("Weld_type",WELD_TYPE),
value("Leg_length",LEG_LENGTH),
value("Shield_gas",S_GAS),
value("Runs",RUNS),
value("Elec_size",ELEC_SIZE),
clearwindow,
scrhnd(off,_),
retractall(_name),
removewindow,
shiftwindow(81),
temp12(WELD_TYPE),
temp13(POSITION),
str_real(LEG_LENGTH,Leg_L),
str_real(RUNS,Run),
str_real(ELEC_SIZE,Elec_S),
consult("GMAW2_dat.scr",name),
GMAW_data11(Run,1,WELD_TYPE,Leg_L,POSITION,Elec_S,S_GAS).

```

GMAW\_data11(Run,Y,WELD\_TYPE,Leg\_L,POSITION,Elec\_S,S\_GAS):-

```

Run >= Y,
makestatus(112,"Fill Out the Fields & Press F10 to END"),
createwindow(on),
scrhnd(on,_),
clearwindow,
value("Voltage",VOLTAGE),
value("Current",CURRENT),
value("Speed",SPEED),
value("Num_elec",NUM_ELEC),
value("Weight",WEIGHT),
value("Over_time",OVER_TIME),
value("Arc_time",ARC_TIME),
value("G_Flow",G_FLOW),

```

```

    %value("Shield_gas",S_GAS),
    clearwindow,
    scrhnd(off, _),
    shiftwindow(81),
    temp12(WELD_TYPE),
    temp13(POSITION),
    str_real(VOLTAGE,VOLT),
    write(VOLT),
    str_real(CURRENT,Curr),
    write(Curr),
    readchar(_),
    str_real(SPEED,Sp),
    str_real(NUM_ELEC,Num_E),
    str_real(ARC_TIME,Arc_T),
    str_real(OVER_TIME,Over_T),
    str_real(WEIGHT,Wt),
    str_real(G_FLOW,Gflow),
    assert(GMAW_data(WELD_TYPE,Leg_L,POSITION,Y,Elec_S,VOLT,Curr,Sp,Num_E,Wt,Arc_T,
Over_T,Gflow,S_GAS)),
    X = Y +1,
    save_std_data,
    GMAW_data11 (Run,X,WELD_TYPE,Leg_L,POSITION,Elec_S,S_GAS).

```

gtaw\_data\_entry:-

```

%    consult_std_data,
    consult("gtaw1_dat.scr",name),
    makestatus(112,"Fill Out the Fields & Press F10 to END"),
    createwindow(on),
    scrhnd(on, _),
    clearwindow,
%    value("Weld_type",WELD_TYPE),
    value("Leg_length",LEG_LENGTH),
    value("Shield_gas",S_GAS),
    value("Runs",RUNS),
    value("Elec_size",ELEC_SIZE),
    clearwindow,
    scrhnd(off, _),
    retractall(_,name),
    removewindow,
    shiftwindow(81),
    temp12(WELD_TYPE),
    temp13(POSITION),
    str_real(LEG_LENGTH,Leg_L),
    str_real(RUNS,Run),
    str_real(ELEC_SIZE,Elec_S),
    consult("gtaw2_dat.scr",name),
    gtaw_data11(Run,1,WELD_TYPE,Leg_L,POSITION,Elec_S,S_GAS).

```



```

gtaw_data11(Run,Y,WELD_TYPE,Leg_L,POSITION,Elec_S,S_GAS):-
    Run >= Y,
    makestatus(112,"Fill Out the Fields & Press F10 to END"),
    createwindow(on),
    scrhnd(on, _),
    clearwindow,
    value("Voltage",VOLTAGE),
    value("Current",CURRENT),
    value("Speed",SPEED),
    value("Num_elec",NUM_ELEC),
    value("Weight",WEIGHT),
    value("Over_time",OVER_TIME),
    value("Arc_time",ARC_TIME),
    value("G_Flow",G_FLOW),
    %value("Shield_gas",S_GAS),
    clearwindow,
    scrhnd(off, _),
    shiftwindow(81),
    temp12(WELD_TYPE),
    temp13(POSITION),
    str_real(VOLTAGE,VOLT),
    write(VOLT),
    str_real(CURRENT,Curr),
    write(Curr),
    readchar(_),
    str_real(SPEED,Sp),
    str_real(NUM_ELEC,Num_E),
    str_real(ARC_TIME,Arc_T),
    str_real(OVER_TIME,Over_T),
    str_real(WEIGHT,Wt),
    str_real(G_FLOW,Gflow),
    assert(gtaw_data(WELD_TYPE,Leg_L,POSITION,Y,Elec_S,VOLT,Curr,Sp,Num_E,Wt,Arc_T,Over_T,Gflow,S_GAS)),
    X = Y + 1,
    save_std_data,
    gtaw_data11 (Run,X,WELD_TYPE,Leg_L,POSITION,Elec_S,S_GAS).

```

```

saw_data_entry:-
    consult_std_data,
    consult("saw1_da.scr",name),
    makestatus(112,"Fill Out the Fields & Press F10 to END"),
    createwindow(on),
    scrhnd(on, _),
    clearwindow,
    %
    value("Weld_type",WELD_TYPE),
    value("Leg_length",LEG_LENGTH),
    value("T_Flux",T_FLUX),

```

```

value("Runs",RUNS),
value("Elec_size",ELEC_SIZE),
clearwindow,
scrhnd(off, _),
retractall(_,name),
removewindow,
shiftwindow(81),
temp12(WELD_TYPE),
temp13(POSITION),
str_real(LEG_LENGTH,Leg_L),
str_real(RUNS,Run),
str_real(ELEC_SIZE,Elec_S),
consult("saw2_da.scr",name),
saw_data11(Run,1,WELD_TYPE,Leg_L,POSITION,Elec_S,T_FLUX).

```

saw\_data11(Run,Y,WELD\_TYPE,Leg\_L,POSITION,Elec\_S,T\_FLUX):-

```

Run >= Y,
makestatus(112,"Fill Out the Fields & Press F10 to END"),
createwindow(on),
scrhnd(on, _),
clearwindow,
value("Current",CURRENT),
value("Voltage",VOLTAGE),
value("Speed",SPEED),
value("Num_elec",NUM_ELEC),
value("Weight",WEIGHT),
value("Over_time",OVER_TIME),
value("Arc_time",ARC_TIME),
clearwindow,
scrhnd(off, _),
shiftwindow(81),
temp12(WELD_TYPE),
temp13(POSITION),
str_real(VOLTAGE,VOLT),
str_real(CURRENT,Curr),
str_real(SPEED,Sp),
str_real(NUM_ELEC,Num_E),
str_real(ARC_TIME,Arc_T),
str_real(OVER_TIME,Over_T),
str_real(WEIGHT,Wt),
assert(saw_data(WELD_TYPE,Leg_L,POSITION,Y,Elec_S,VOLT,Curr,Sp,Num_E,Arc_T,Over_T,
Wt,T_FLUX)),
X = Y + 1,
save_std_data,
saw_data11 (Run,X,WELD_TYPE,Leg_L,POSITION,Elec_S,T_FLUX).

```

### E.10 Job\_Data.pro - Prolog Program for Job Data Entry

```
%trace field_action
project "neurex"

include "global.pro"
include "tdoms.pro"
include "tpreds.pro"
include "status.pro"

DOMAINS
FIELD_NAME = SYMBOL
TYPE       = int(); str(); real()

/*****
*           Database Predicates           *
*****/

DATABASE - name
field(FIELD_NAME, TYPE, ROW, COL, LEN)
txtfield(ROW, COL, LEN, STRING)
value(FIELD_NAME, STRING)
windowsize(ROW, COL)

include "menu.pro"
include "scrhnd.pro"           % Or vscrhnd.pro

database - job_temp
    temp1(string)
    temp2(string)
    temp3(string)
    temp4(string)
    temp5(string)
    temp6(string)
    temp7(string)
    temp8(string)

predicates
    consult_input_data
    %input_data
    save_input_data
    get_mat (integer,string)
    get_position (integer,string)
    get_appearance (integer,string)
```

```

get_weld_type (integer,string)
get_weld_qual (integer,string)
get_dep (integer,string)
get_automation (integer,string)

```

## CLAUSES

```

field_action ("Parent_Mat") :-
    menu(5,5,10,10,["Mild Steel","Aluminum Alloy","Carbon Steel","Stainless Steel","Cast
Iron"],"Choose",0,Choice),
    get_mat (Choice.Mat),
    write (Mat),
    retractall(temp1(_)),
    assert(temp1(Mat)).

field_action ("Weld_Type") :-
    menu(5,5,10,10,["FILLET","BUTT"],"Choose",0,Choice),
    get_weld_type (Choice.Mat),
    write (Mat),
    retractall(temp2(_)),
    assert(temp2(Mat)).

field_action ("Position") :-
    menu(5,5,10,10,["FLAT","HORIZONTAL","VERTICAL","DOWNWARD","OVERHEAD"],"Cho
ose",0,Choice),
    get_position (Choice.Mat),
    write (Mat),
    retractall(temp3(_)),
    assert(temp3(Mat)).

field_action ("Appearance") :-
    menu(5,5,10,10,["FLAT","CONCAVE","CONVEX"],"Choose",0,Choice),
    get_appearance (Choice.Mat),
    write (Mat),
    retractall(temp4(_)),
    assert(temp4(Mat)).

field_action ("Dep_Rate") :-
    menu(5,5,10,10,["HIGH","MEDIUM","LOW"],"Choose",0,Choice),
    get_dep(Choice.Dep_Rate),
    write(Dep_Rate),
    retractall(temp5(_)),
    assert(temp5(Dep_Rate)).

field_action ("Weld_Qual") :-
    menu(5,5,10,10,["HIGH","MEDIUM","LOW"],"Choose",0,Choice),
    get_weld_qual (Choice.Weld_Q),
    write(Weld_Q),
    retractall(temp6(_)),

```

```

        assert(temp6(Weld_Q)).

field_action ("Automation") :-
    menu(5,5,10,10,["YES","NO"],"Choose",0,Choice).
    get_automation (Choice,Aut),
    write(Aut),
    retractall(temp7(_)),
    assert(temp7(Aut)).

field_action(_) :- fail.

/*****
*           Field Values           *
*****/

field_value(FIELD_NAME, VALUE) :- value(FIELD_NAME, VALUE), !.

/*****
*           No Input Fields        *
*****/
noinput(_) :- fail.

get_mat(1,"MILD STEEL").
get_mat(2,"ALUMINUM ALLOY").
get_mat(3,"CARBON STEEL").
get_mat(4,"STAINLESS STEEL").
get_mat(5,"CAST IRON").

get_position(1,"FLAT").
get_position(2,"HORIZONTAL").
get_position(3,"VERTICAL").
get_position(4,"DOWNWARD").
get_position(5,"OVERHEAD").

get_weld_type (1,"FILLET").
get_weld_type (2,"BUTT").

get_appearance (1,"FLAT").
get_appearance (2,"CONCAVE").
get_appearance (3,"CONVEX").

get_dep(1,"HIGH").
get_dep(2,"MEDIUM").
get_dep(3,"LOW").

get_weld_qual(1,"HIGH").
get_weld_qual(2,"MEDIUM").

```

```

get_weld_qual(3,"LOW").

get_automation(1,"YES").
get_automation(2,"NO").

consult_input_data:-
    existfile("job_data.dba"),!,
    consult("job_data.dba").
consult_input_data.

input_data:-
    consult("job_data.scr",name),
    makestatus(112,"Fill Out the Fields & Press F10 to END"),
    createwindow(on),
    scrhnd(on, _),
    value("Seam_Length",SEAM_LENGTH),
    value("Thickness",PLATE_THICKNESS),
    value("Strength",STRENGTH),
    value("Leg_Length",LEG_LENGTH),
    clearwindow,
    scrhnd(off, _),
    removewindow,
    temp1(PARENT_MAT),
    temp2(WELD_TYPE),
    temp3(POSITION),
    temp4(APPEARANCE),
    temp5(DEP_RATE),
    temp6(WELD_QUAL),
    temp7(AUTOMATION),
    str_real(SEAM_LENGTH,Scam),
    str_real(PLATE_THICKNESS,Plate),
    str_real(STRENGTH,Stren),
    str_real(LEG_LENGTH,LEG_L),
    retractall(_),
    assert(job_data(PARENT_MAT,WELD_TYPE,POSITION,Scam,Plate,Stren,APPEARANCE,DEP_
RATE,WELD_QUAL,AUTOMATION,LEG_L)),
    save_input_data.

input_data:-
    clearwindow,
    write("Press ESC to continue ....."), !,
    scrhnd(off, _),
    removewindow,
    shiftwindow(81).

save_input_data :-
    save("job_data.dba").

```

# APPENDIX F - COMPARISON OF RULE EXTRACTION METHODS

## F.1 PROCESS SELECTOR - Comparison of various methods

	Fu's Method					10% Rule					20% Rule					30% Rule					Process Selected by various methods							
	SMAW	GMAW	GMAW	SAW	Confidence Factor	SMAW	GMAW	GMAW	SAW	Confidence Factor	SMAW	GMAW	GMAW	SAW	Confidence Factor	SMAW	GMAW	GMAW	SAW	Confidence Factor	Actual	Unpruned Neural Network	Pruned	Fu's Method	10% Rule	20% Rule	30% Rule	Unpruned Hybrid (20%)
1	0.00	-0.08	-2.03	7.69	1.00	0.00	0.00	-2.00	7.70	1.00	0.00	0.00	-2.15	7.61	1.00	0.00	0.00	-1.90	7.00	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
2	0.00	0.13	-2.86	8.21	0.98	0.00	0.17	-2.80	8.20	0.98	0.00	0.00	-2.99	8.06	1.00	0.00	0.00	-2.70	7.50	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
3	0.00	4.54	1.07	3.31	0.27	0.00	4.42	1.08	3.53	0.20	0.00	0.15	-0.80	6.31	0.98	-4.93	7.92	2.10	0.70	0.91	SAW	SAW	SAW	GMAW	GMAW	SAW	GMAW	SAW
4	0.00	5.14	-0.85	3.01	0.42	0.00	5.01	-0.84	3.22	0.36	0.00	0.70	-0.31	6.57	0.89	0.00	5.14	-0.85	0.45	0.91	SAW	SAW	SAW	GMAW	GMAW	SAW	GMAW	SAW
5	0.00	2.96	1.79	2.52	0.15	0.00	2.84	1.80	2.90	0.02	0.00	0.98	-1.24	5.33	0.82	0.00	2.96	1.79	0.26	0.40	SAW	SAW	SAW	GMAW	SAW	SAW	GMAW	SAW
6	0.00	2.64	1.03	3.39	0.22	0.00	2.51	1.04	3.77	0.33	0.00	0.53	-0.75	5.18	0.90	0.00	2.64	1.03	0.23	0.61	SAW	SAW	SAW	GTAW	SAW	SAW	GMAW	SAW
7	0.00	4.54	1.07	3.31	0.27	0.00	4.42	1.08	3.52	0.20	0.00	1.15	-0.80	6.31	0.82	0.00	4.54	1.07	0.40	0.76	SAW	SAW	SAW	GMAW	GMAW	SAW	GMAW	SAW
8	0.00	5.14	-0.85	3.00	0.42	0.00	5.01	-0.84	3.22	0.36	0.00	0.70	-0.31	6.57	0.89	0.00	5.14	-0.85	0.45	0.91	SAW	SAW	SAW	GMAW	GMAW	SAW	GMAW	SAW
9	0.00	-0.54	-2.60	8.06	1.00	0.00	0.53	-2.68	8.13	0.93	0.00	-0.53	-2.80	7.99	1.00	0.00	-0.51	-2.28	7.50	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
10	0.00	-0.24	-1.39	4.56	1.00	0.00	-0.15	-1.39	4.64	1.00	0.00	-0.15	-1.23	4.46	1.00	0.00	-0.15	-1.52	4.18	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
11	0.00	0.47	-1.28	5.51	0.91	0.00	0.47	-1.28	5.59	0.92	0.00	0.48	-1.28	5.59	0.92	0.00	0.74	-1.56	5.59	0.87	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
12	0.00	0.02	-0.79	5.78	1.00	0.00	0.02	-0.79	5.85	1.00	0.00	0.02	-0.79	5.85	1.00	0.00	0.29	-1.07	5.85	0.95	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
13	0.00	2.92	1.73	3.70	0.21	0.00	2.86	1.65	3.67	0.22	0.00	2.69	1.65	3.81	0.29	0.00	2.42	1.67	4.08	0.41	SAW	GTAW	GMAW	SAW	SAW	SAW	SAW	GTAW
14	0.00	1.57	0.52	2.20	0.29	0.00	1.59	0.51	2.17	0.27	0.00	1.42	0.37	2.30	0.38	0.00	1.15	0.40	2.58	0.55	SAW	GTAW	SAW	SAW	SAW	SAW	SAW	GTAW
15	0.00	-0.39	-0.79	3.54	1.00	0.00	-0.38	-0.80	3.51	1.00	0.00	-0.72	-0.94	3.51	1.00	0.00	-0.99	-1.21	3.78	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
16	0.00	1.06	0.19	2.31	0.54	0.00	1.08	0.18	2.27	0.53	0.00	0.91	0.04	2.24	0.59	0.00	0.40	-0.23	2.51	0.84	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
17	0.00	0.19	-0.45	2.89	0.93	0.00	0.21	-0.45	2.86	0.93	0.00	0.05	-0.59	2.69	0.98	0.00	-0.21	-0.86	2.96	1.00	SAW	SAW	SAW	SAW	SAW	SAW	SAW	SAW
18	0.00	2.44	1.80	2.97	0.18	0.00	2.36	1.79	2.94	0.20	0.00	2.20	1.65	3.08	0.29	0.00	1.93	1.67	3.35	0.42	SAW	GTAW	SMAW	SAW	SAW	SAW	SAW	GTAW
19	0.03	2.56	-0.50	0.00	0.99	-0.12	2.00	-0.50	0.00	1.06	2.20	1.98	2.55	0.00	0.14	0.03	2.55	-0.50	0.23	0.99	SMAW	GTAW	SMAW	GMAW	GMAW	GTAW	GMAW	GTAW
20	0.03	2.55	-0.50	0.00	0.99	-0.12	2.00	-0.50	0.00	1.06	2.20	1.98	2.55	0.00	0.14	0.03	2.55	-0.50	0.23	0.99	SMAW	GTAW	SMAW	GMAW	GMAW	GTAW	GMAW	GTAW
21	0.42	1.64	0.66	0.00	0.60	0.00	1.08	0.66	0.00	0.39	3.21	1.98	2.54	0.00	0.21	0.41	1.64	0.66	0.15	0.60	SMAW	SMAW	SMAW	GMAW	GMAW	SMAW	GMAW	SMAW
22	1.37	0.64	-0.33	-6.32	0.53	0.95	-0.20	-0.22	-5.90	1.21	3.35	1.91	0.13	-3.49	0.43	1.37	0.64	-0.33	0.12	0.53	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
23	0.39	0.96	0.42	-7.19	0.56	0.23	0.11	0.53	-6.81	0.57	2.89	2.36	-0.36	-3.34	0.18	0.39	0.96	0.42	0.08	0.00	SMAW	SMAW	SMAW	SMAW	GTAW	SMAW	GMAW	SMAW
24	-4.62	3.73	0.79	-6.88	0.79	-4.78	3.17	1.06	-6.66	0.67	-1.53	3.38	2.88	-2.94	0.15	-4.62	3.72	0.79	0.33	0.79	SMAW	SMAW	SMAW	GMAW	GMAW	GMAW	GMAW	SMAW
25	-4.62	3.72	0.79	-6.88	0.79	-4.78	3.17	1.06	-6.66	0.67	-1.53	3.38	2.88	-2.94	0.15	-4.62	3.72	0.79	0.33	0.79	SMAW	SMAW	SMAW	GMAW	GMAW	GMAW	GMAW	SMAW
26	3.85	0.22	1.55	0.00	0.60	3.93	0.24	1.53	0.00	0.61	4.13	0.24	1.54	0.00	0.63	4.13	0.24	1.79	0.00	0.57	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
27	3.85	0.22	1.55	0.00	0.60	3.93	0.24	1.53	0.00	0.61	4.13	0.24	1.54	0.00	0.63	4.13	0.24	1.79	0.00	0.57	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
28	3.18	-1.31	0.84	0.00	0.74	3.27	-1.30	0.83	0.00	0.75	3.46	-1.29	0.83	0.00	0.76	3.46	-1.29	0.79	0.00	0.77	SMAW	NONE	GTAW	SMAW	SMAW	SMAW	SMAW	SMAW
29	3.18	-1.31	0.85	0.00	0.73	3.27	-1.30	0.83	0.00	0.75	3.46	-1.29	0.83	0.00	0.76	3.46	-1.29	0.79	0.00	0.77	SMAW	NONE	GTAW	SMAW	SMAW	SMAW	SMAW	SMAW
30	3.85	0.22	1.55	0.00	0.60	3.93	0.23	1.53	0.00	0.61	4.13	0.24	1.53	0.00	0.63	4.13	0.24	1.79	0.00	0.57	SMAW	NONE	GTAW	SMAW	SMAW	SMAW	SMAW	SMAW
31	3.85	0.22	1.55	0.00	0.60	3.93	0.24	1.53	0.00	0.61	4.13	0.23	1.53	0.00	0.63	4.13	0.23	1.79	0.00	0.57	SMAW	SMAW	GMAW	SMAW	SMAW	SMAW	SMAW	SMAW
32	3.18	-1.31	0.85	0.00	0.73	3.27	-1.29	0.83	0.00	0.75	3.46	-1.29	0.83	0.00	0.76	3.46	-1.29	0.79	0.00	0.77	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
33	3.18	-1.31	0.85	0.00	0.73	3.27	-1.29	0.84	0.00	0.74	3.46	-1.29	0.84	0.00	0.76	3.46	-1.29	0.79	0.00	0.77	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
34	3.40	0.00	1.33	-1.81	0.61	3.54	0.02	1.25	-1.84	0.65	3.73	0.15	1.25	-1.71	0.66	3.47	-0.14	1.27	-1.43	0.63	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
35	2.64	1.82	2.38	-2.69	0.10	2.59	1.82	2.37	-2.61	0.08	2.75	1.82	2.37	-2.78	0.14	2.52	1.82	2.09	-2.78	0.17	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW	SMAW
37	2.78	1.76	-0.04	-2.33	0.37	2.73	1.76	-0.04	-2.24	0.36	2.89	1.75	-0.04	-2.42	0.39	2.66	1.76	-0.32	-2.42	0.34	SMAW	SMAW	GTAW	SMAW	SMAW	SMAW	SMAW	SMAW

	Fu's Method				10% Rule				20% Rule				30% Rule				Process Selected by various methods								
	SMAW	GMAW	SMAW	Confidence Factor	SMAW	GMAW	SMAW	Confidence Factor	SMAW	GMAW	SMAW	Confidence Factor	SMAW	GMAW	SMAW	Confidence Factor	Actual	Unpruned Neural Network	Pruned	Fu's Method	10% Rule	20% Rule	30% Rule	Unpruned Hybrid (20%)	
38	2.78	1.76	-0.04	-2.33	0.37	2.73	1.76	-0.04	-2.24	0.36	2.89	1.76	-0.04	-2.42	0.39	2.66	1.75	-0.32	-2.42	0.34	SMAW	SMAW	SMAW	SMAW	SAW
39	2.34	2.39	2.57	-2.33	0.07	1.99	2.38	2.56	-2.33	0.07	2.15	2.38	2.56	-2.51	0.07	1.92	2.38	2.56	-2.78	0.07	SMAW	SAW	SAW	SAW	SAW
40	2.05	2.38	2.56	-2.34	0.07	1.99	2.38	2.56	-2.33	0.07	2.15	2.38	2.56	-2.51	0.07	1.92	2.38	2.56	-2.78	0.07	SMAW	SAW	SAW	SAW	SAW
41	2.19	2.32	0.15	-1.97	0.05	2.13	2.31	0.14	-1.97	0.08	2.29	2.32	0.15	-2.14	0.01	2.06	2.31	0.14	-2.42	0.11	SMAW	SAW	SAW	SAW	SAW
42	2.26	1.39	-1.36	-4.41	0.38	2.10	1.15	-1.36	-4.24	0.45	2.39	2.31	0.15	-2.14	0.03	2.26	1.39	-1.35	0.20	0.38	SMAW	SAW	SAW	SAW	SAW
43	2.26	1.39	-1.36	-4.41	0.38	2.10	1.15	-1.36	-4.24	0.45	2.39	2.31	0.14	-2.15	0.03	2.26	1.39	-1.35	0.20	0.38	SMAW	SAW	SAW	SAW	SAW
44	0.00	4.73	2.42	0.00	0.49	0.00	4.17	2.70	0.00	0.35	0.00	4.36	3.14	0.00	0.28	0.00	4.72	2.43	0.42	0.49	SMAW	SAW	SAW	SAW	SAW
45	0.00	1.55	-1.50	-4.10	1.00	0.00	0.71	-1.39	-3.72	1.00	0.00	1.91	-2.22	-1.08	1.00	0.00	1.55	-1.50	0.14	1.00	SMAW	SAW	SAW	SAW	SAW
46	0.00	4.23	0.31	0.64	0.85	0.00	3.67	0.30	0.85	0.77	0.00	3.15	0.02	2.11	0.33	0.00	4.22	0.30	0.37	0.91	SMAW	SAW	SAW	SAW	SAW
47	0.00	2.56	-1.14	-2.32	1.44	0.00	2.00	-1.13	-1.94	1.57	0.00	1.98	-1.42	0.55	0.72	0.00	2.55	-1.13	0.23	0.91	SMAW	SAW	SAW	SAW	SAW
48	0.00	2.87	3.02	0.00	0.05	0.00	2.81	2.94	0.00	0.04	0.00	2.80	2.94	0.00	0.05	0.00	2.80	3.19	0.00	0.12	SMAW	SAW	SAW	SAW	SAW
49	0.00	0.34	1.31	1.75	0.25	0.00	0.28	1.24	1.73	0.28	0.00	0.27	1.24	1.86	0.33	0.00	0.27	1.19	1.86	0.36	SMAW	SAW	SAW	SAW	SAW
50	0.00	1.34	2.31	2.75	0.16	0.00	1.27	2.24	2.72	0.18	0.00	1.28	2.24	2.86	0.22	0.00	1.28	2.19	2.86	0.23	SMAW	SAW	SAW	SAW	SAW
51	0.00	2.87	2.89	4.27	0.32	0.00	2.80	2.81	4.24	0.34	0.00	2.80	2.82	4.38	0.36	0.00	2.80	3.07	4.38	0.30	SMAW	SAW	SAW	SAW	SAW
52	0.00	2.87	2.89	4.27	0.32	0.00	2.80	2.81	4.25	0.34	0.00	2.80	2.82	4.38	0.36	0.00	2.80	3.07	4.38	0.30	SMAW	SAW	SAW	SAW	SAW
53	0.00	1.87	1.89	3.27	0.42	0.00	1.80	1.81	3.24	0.44	0.00	1.80	1.82	3.38	0.46	0.00	1.80	2.07	3.38	0.39	SMAW	SAW	SAW	SAW	SAW
55	0.00	3.24	3.37	1.63	0.04	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.06	1.70	0.06	SMAW	SAW	SAW	SAW	SAW
56	0.00	3.24	3.37	1.63	0.04	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.06	1.70	0.06	SMAW	SAW	SAW	SAW	SAW
57	0.00	3.24	3.37	1.63	0.04	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.34	1.70	0.03	0.00	3.24	3.06	1.70	0.06	SMAW	SAW	SAW	SAW	SAW
58	0.00	3.80	5.18	0.00	0.27	0.00	3.80	5.14	0.00	0.26	0.00	3.80	5.14	0.00	0.26	0.00	3.80	5.14	0.00	0.26	SMAW	SAW	SAW	SAW	SAW
59	-2.14	4.73	6.11	-0.65	0.23	-2.15	4.73	6.08	-0.65	0.22	-1.93	4.73	6.08	-0.65	0.22	-2.22	4.73	6.08	-0.92	0.22	SMAW	SAW	SAW	SAW	SAW
60	0.00	3.73	2.76	0.35	0.26	0.00	3.73	2.72	0.35	0.27	0.00	3.73	2.72	0.34	0.27	0.00	3.73	2.72	0.07	0.27	SMAW	SAW	SAW	SAW	SAW
61	-4.27	4.80	6.91	0.98	0.31	-4.29	4.80	6.88	0.98	0.30	-4.13	4.80	6.88	0.98	0.30	-4.36	4.80	6.88	0.70	0.30	SMAW	SAW	SAW	SAW	SAW
62	0.00	4.80	4.56	2.98	0.05	0.00	4.80	4.53	2.98	0.06	0.00	4.80	4.52	2.98	0.06	0.00	4.80	4.52	2.70	0.06	SMAW	SAW	SAW	SAW	SAW
63	-3.20	3.80	5.92	-0.02	0.36	-3.29	3.80	5.88	-0.02	0.35	-3.13	3.80	5.88	-0.01	0.35	-3.36	3.80	5.88	-0.30	0.35	SMAW	SAW	SAW	SAW	SAW
64	0.00	3.80	3.56	1.90	0.06	0.00	3.80	3.53	1.98	0.07	0.00	3.80	3.52	1.98	0.07	0.00	3.80	3.52	1.70	0.07	SMAW	SAW	SAW	SAW	SAW
65	0.00	5.15	5.70	0.00	0.10	0.00	5.15	5.67	0.00	0.09	0.00	5.15	5.67	0.00	0.09	0.00	5.15	5.67	0.00	0.09	SMAW	SAW	SAW	SAW	SAW
66	-0.75	6.08	6.64	0.00	0.08	-0.77	6.08	6.61	0.00	0.08	-0.60	6.08	6.61	0.00	0.08	-0.83	6.08	6.61	0.00	0.08	SMAW	SAW	SAW	SAW	SAW
67	0.00	2.65	3.12	0.00	0.15	0.00	2.70	3.12	0.00	0.13	0.00	2.70	3.00	0.00	0.10	0.00	2.49	3.00	0.00	0.17	SMAW	SAW	SAW	SAW	SAW
68	0.00	5.08	3.29	0.00	0.35	0.00	5.08	3.25	0.00	0.36	0.00	5.08	3.25	0.00	0.36	0.00	5.08	3.25	0.00	0.36	SMAW	SAW	SAW	SAW	SAW
69	-4.31	2.42	7.60	0.00	0.68	-4.30	2.40	7.60	0.00	0.68	-4.30	2.47	7.56	0.00	0.67	-4.30	2.26	7.56	0.00	0.70	SMAW	SAW	SAW	SAW	SAW
70	0.00	3.90	3.80	0.00	0.03	0.00	3.90	3.80	0.00	0.03	0.00	3.95	3.75	0.00	0.05	0.00	3.74	3.75	0.00	0.49	SMAW	SAW	SAW	SAW	SAW
71	-3.31	1.42	6.68	0.00	0.79	-3.31	1.47	6.68	0.00	0.78	-3.31	1.47	6.50	0.00	0.77	3.30	1.26	6.50	0.00	0.49	SMAW	SAW	SAW	SAW	SAW
73	0.00	2.90	2.87	0.00	0.01	0.00	2.90	2.87	0.00	0.01	0.00	2.95	2.75	0.00	0.07	0.00	2.74	2.75	0.00	0.00	SMAW	SAW	SAW	SAW	SAW
72	-3.78	0.21	4.72	1.50	0.68	-3.70	0.28	4.72	1.47	0.69	-3.94	-0.28	4.60	1.47	0.68	-3.90	-0.49	5.14	1.21	0.76	SMAW	SAW	SAW	SAW	SAW
74	-5.35	0.97	5.85	3.76	0.36	-5.30	0.91	5.80	3.73	0.36	-5.50	0.71	5.60	3.73	0.33	-5.50	0.50	6.10	3.47	0.43	SMAW	SAW	SAW	SAW	SAW
73	-5.35	0.97	5.80	3.76	0.35	-5.30	0.91	5.80	3.73	0.36	-5.50	0.71	5.60	3.73	0.33	-5.50	0.50	6.10	3.47	0.43	SMAW	SAW	SAW	SAW	SAW



Fi's Method				10% Rule				20% Rule				30% Rule				Process Selected by various methods												
SAW	GMAW	Saw	Confidence Factor	SAW	GMAW	SMAW	GMAW	GMAW	SMAW	GMAW	SAW	Confidence Factor	SAW	GMAW	SMAW	GMAW	GMAW	SAW	Confidence Factor	Actual	Unpruned Neural Network	Pruned	Fu's Method	10% Rule	20% Rule	30% Rule	Unpruned (20%)	
75	-3.09	6.16	1.35	0.00	0.78	-3.25	6.26	1.62	0.00	0.74	-3.1	6.16	1.62	0	0.74	-3.09	6.15	1.35	0.54	0.78	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
76	-4.77	7.81	1.83	0.00	0.77	-4.93	7.92	2.10	0.00	0.73	-4.8	7.81	2.1	0	0.73	-4.77	7.81	1.83	0.69	0.77	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
77	-4.77	7.81	1.83	0.00	0.77	-4.90	7.90	2.10	0.00	0.73	-4.8	7.82	2.1	0	0.73	-4.77	7.81	1.83	0.69	0.77	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
78	-3.09	6.15	1.35	0.00	0.78	-3.25	6.26	1.62	0.00	0.74	-3.1	6.16	1.62	0	0.74	-3.09	6.15	1.35	0.54	0.78	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
79	-4.77	7.81	1.83	0.00	0.77	-4.93	7.92	2.10	0.00	0.73	-4.8	7.82	2.1	0	0.73	-4.77	7.81	1.83	0.69	0.77	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
80	-4.77	7.81	1.83	0.00	0.77	-4.93	7.92	2.10	0.00	0.73	-4.8	7.82	2.1	0	0.73	-4.77	7.81	1.83	0.69	0.77	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
81	0.33	0.08	2.02	0.00	0.83	0.38	0.02	1.94	0.00	0.80	0.58	0.02	1.94	0.00	0.70	0.58	0.02	1.94	0.00	0.74	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
82	0.48	-0.41	2.09	0.00	0.77	0.56	-0.47	2.08	0.00	0.73	0.58	-0.47	1.94	0.00	0.70	0.58	-0.47	2.19	0.00	0.74	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
83	0.33	0.08	2.02	0.00	0.83	0.38	0.02	1.94	0.00	0.80	0.58	0.02	1.94	0.00	0.70	0.58	0.02	2.19	0.00	0.74	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
84	0.47	-0.40	2.09	0.00	0.77	0.57	-0.47	2.08	0.00	0.73	0.58	-0.47	1.94	0.00	0.70	0.58	-0.47	2.19	0.00	0.74	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
85	0.36	2.78	4.82	-4.10	0.93	0.31	2.77	4.82	-4.02	0.94	0.46	2.77	4.82	-4.02	0.90	0.23	2.77	4.53	-4.02	0.95	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
86	0.23	3.33	5.00	-3.74	0.33	-0.29	3.33	5.00	-3.74	0.33	-0.13	3.33	5.00	-3.74	0.33	-0.36	3.33	5.00	-4.02	0.33	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
87	0.36	2.77	4.82	-4.10	0.43	0.31	2.77	4.82	-4.02	0.43	0.47	2.77	4.82	-4.02	0.43	0.23	2.78	4.53	-4.02	0.39	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
88	-0.23	3.34	5.00	-3.75	0.33	-0.29	3.33	5.00	-3.74	0.33	-0.13	3.34	5.01	-3.75	0.33	-0.36	3.33	5.00	-4.02	0.33	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
89	-2.16	-0.16	5.67	0.00	1.00	2.16	-0.11	5.60	0.60	0.89	-2.30	-0.11	5.50	0.00	1.00	-2.31	-0.32	5.40	0.00	1.00	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
94	-4.65	0.50	5.40	-0.62	1.11	-4.65	0.47	5.43	-0.64	1.12	-4.60	0.47	5.30	0.64	0.88	-4.65	0.26	5.55	-0.90	1.16	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
95	0.00	-0.26	0.18	7.36	0.98	0.00	-0.24	0.18	7.36	0.98	0.00	-0.24	0.06	7.20	0.94	0.00	-0.45	0.90	6.90	0.96	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
96	0.00	0.10	0.81	9.99	0.92	0.00	0.03	0.81	9.96	0.92	0.00	0.16	0.55	9.96	0.99	0.00	-0.37	1.33	9.71	0.86	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
97	0.56	1.20	4.27	0.00	0.55	0.71	1.69	4.27	0.00	0.71	0.56	1.23	4.41	0.00	0.72	0.56	0.81	4.41	0.00	0.82	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
98	3.71	1.67	0.71	0.00	0.55	3.71	1.69	0.71	0.00	0.54	3.70	1.69	0.85	0.00	0.54	3.70	1.52	0.85	0.00	0.96	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
99	3.62	-1.71	0.16	0.60	0.84	3.62	-1.70	0.16	0.66	0.82	3.32	-1.70	0.18	0.52	0.84	3.32	-1.88	0.46	0.00	0.86	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
100	0.00	2.31	2.86	0.00	0.19	0.00	2.37	2.87	0.00	0.18	0.00	2.36	2.74	0.00	0.14	0.00	2.15	2.74	0.00	0.22	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
101	0.00	3.11	2.44	5.21	0.40	0.00	3.16	2.44	5.28	0.40	0.00	3.16	2.44	5.13	0.38	0.00	2.95	2.44	5.13	0.42	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
102	-4.65	0.50	5.44	-0.62	0.91	-4.65	0.47	5.43	-0.65	0.91	-4.60	0.47	5.30	-0.64	0.91	-4.65	0.26	5.59	-0.90	0.99	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
103	-0.35	3.35	3.40	0.00	1.10	-0.35	3.36	3.40	0.00	1.10	-0.17	3.48	3.39	0.00	0.03	-0.17	3.27	3.39	0.00	0.99	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
104	-1.68	1.77	3.66	0.00	0.52	-1.68	1.82	3.66	0.00	0.50	-1.49	1.93	3.50	0.00	0.45	-1.49	1.71	3.54	0.00	0.52	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
105	-2.38	2.24	4.08	0.00	0.45	-2.38	2.26	4.08	0.00	0.45	-2.34	2.17	3.85	0.00	0.44	-2.34	1.95	4.12	0.00	0.53	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW
106	-1.58	0.01	3.49	-0.53	1.00	-1.58	-0.09	3.49	-0.63	1.02	-1.73	-0.08	3.50	-0.63	1.02	-1.73	-0.49	4.04	0.88	1.12	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW	GMAW

Accuracy		68%	78%	56%	58%	64%	55%	80%
SAW	28	13	17	17	17	19	17	16
GMAW	38	44	50	28	26	20	30	46
GMAW	15	21	18	36	36	36	34	16
SAW	17	17	13	18	20	23	18	19
Std. Dev.	10.66	13.89	17.14	9.00	8.38	7.85	8.54	14.57

## **VITA**

Vivek Goel was born in the Sitapur District of Uttar Pradesh province in India. He finished his bachelor's degree in industrial engineering from the prestigious University of Roorkee in May 1989. He joined the master's program in mechanical engineering with specialization in welding in January of 1991. After that he worked at Bharat Petroleum Corporation in Bombay for about a year.

He joined the interdepartmental program in engineering science at Louisiana State University in September of 1991, and started working towards a degree of doctor of philosophy. He earned a master's degree in systems science on his way to the doctorate in philosophy. He joined Acetech, Inc. of New Orleans as a Senior Programmer in 1994 and was promoted to the position of Project Manager in 1996. He kept on working part-time on his doctorate degree. He joined Oracle Corp. in October of 1996 as a Senior Consultant and worked his way up to the position of Technical Manager in the data warehousing group. During his tenure at Oracle, Vivek has worked on several database projects at companies like Anheuser-Busch, Smith and Nephew Medical Products, DELL Corp., United Agro Products, Aventis Crop Science besides working on government projects at US Department of Agriculture and State of Michigan's Department of Consumer Affairs.

His research interests include welding, artificial intelligence, neural networks, data mining and data warehousing. He has written two papers in the journal of Computers in Industry and presented his research in various conferences.

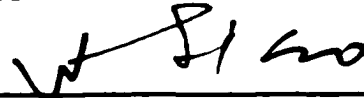
# DOCTORAL EXAMINATION AND DISSERTATION REPORT

**Candidate:** Vivek Goel

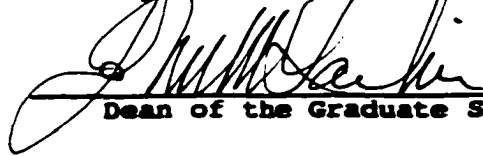
**Major Field:** Engineering Science

**Title of Dissertation:** A Neuro-expert Approach for Decision-making  
in Welding Environment

**Approved:**

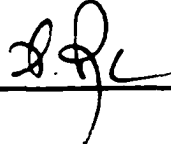


Major Professor and Chairman

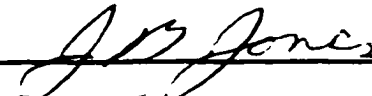


Dean of the Graduate School

**EXAMINING COMMITTEE:**



S. S. S. S.



J. Jones

**Date of Examination:**

1/31/2000