

# A RULE INDUCTION APPROACH TO THE DEVELOPMENT OF AN ENGINE FAILURE RECOVERY SYSTEM FOR HELICOPTERS

P D Baker<sup>1</sup>,  
Advanced Engineering Department,  
GKN Westland Helicopters,  
Yeovil,  
BA20 2YB

R Bradley,  
Department of Mathematics,  
Glasgow Caledonian University,  
Glasgow,  
G4 0BA

## Summary

The primary objective of the ISSAFE project (Inductive Software Solutions for Applications in the Flight Environment) was to establish a firm basis for the construction and use of rule based expert systems in automating specific flight critical tasks. This paper describes one aspect of the research: the development of an advisory system for recovery from engine failure during helicopter take off from an offshore platform where, following a random engine failure, the pilot has the alternatives of landing back on the pad or continuing with take off. An integral part of the system's functional specification was a set of data from a simulator trial consisting of four pilots carrying out, in total, 162 runs. The associated design task has two parts: the first to deal with the real time monitoring and communication with the simulator, the second to induce advisory rules that will clone the skilled behaviour of the pilots in the recovery manoeuvre. The results of an evaluation of the system on the GKN Westland Advanced Engineering Simulation Facility with audio cueing of the advice are described. They provide validation of significant aspects of the system and indicate areas for future development.

## 1. Introduction

The work reported in this paper stems from the Inductive Software Solutions for Application in the Flight Environment (ISSAFE) project sponsored by the UK Department of Trade and

Industry (DTI) and involving a range of UK industrial and academic partners (Table 1). The aim of the project was to accelerate the transfer of Artificial Intelligence (AI) technology from the area of academic feasibility, where some convincing results had been published, into the area of industrial application and also to address the associated issues of specification, verification and validation that would be required in order to establish this new technology as routine, safe and reliable.

Table 1: ISSAFE Partners

ERA Technology
British Aerospace (Airbus) Ltd
GKN Westland Helicopters
Glasgow Caledonian University Company
Defence Evaluation Research Agency
University of Hertfordshire

The area of helicopter flight control was selected for this research with the specific aim of establishing a firm base for the construction and use of rule based expert systems in improving carefree handling systems and in automating specific helicopter tasks while ensuring that the resulting methodologies were amenable to certification.

Conventional control system design is becoming increasingly complex. Not only does this complexity impose an increased computational load for real-time execution but it also imparts a heavy burden on the verification and certification task. There is a case, therefore,

---

<sup>1</sup> Now at: Avionics Systems, Lear Astronics Corp., Santa Monica, California 90406.

with increasing justification, for exploring the practical application of alternative techniques and methodologies. AI, and in particular rule induction, can provide a rule-base which is structured as simple decision trees which are amenable to fast and exhaustive verification and therefore satisfy the exigencies of software quality procedures and the need for short development times. The rules are induced from data collected from experienced pilots carrying out the required task, so that an explicit underpinning by design theory is eliminated - the skilled behaviour is cloned without a formal understanding of the principles involved. The emphasis then shifts from the integrity of the control system design to the completeness and scope of the measured data.

Three types of piloting task were addressed in the ISSAFE project:

- 1: Instrument Landing System (ILS) guided approach,
- 2: Carefree handling for limit protection,
- 3: Recovery from engine failure,

and it is the last of these, the development of an Engine Failure Recovery System (EFRS) that is the subject of this paper. A full treatment of items 1 and 2 may be found in Refs. 1 and 2 but the lessons learnt in this early work contributed significantly to the development of a system for assisting in the recovery from an engine failure and several of these important issues will feature in this paper.

## 2. Rule Induction Methodology

The type of induction mainly employed in the ISSAFE project has been to approximate a controller function of the type

$$demand = f(x_1, \dots, x_n), \quad (1)$$

where the  $x_1, \dots, x_n$  are state variables measured in flight (or quantities derived from them) by a decision tree consisting of rules of the form

IF *condition* THEN *action*.

In this rule, the *condition* involves the input variables,  $x_1, \dots, x_n$ , and the *action* is the resulting value of the response variable, *demand*. The rules, therefore, approximate the controller

function through a piecewise constant representation as depicted in Fig. 1

The splits in the decision tree are usually made on single variables, which is the default for the Classification and Regression Trees (CART) package (Ref. 3) used for the rule induction (tree building) process. A brief description of the CART package may be found in the Appendix. The regression mode is used to approximate mappings of the kind shown in Eqn. 1. The regression is applied to the presented data in order to achieve an optimum decision tree. If time does not appear explicitly as an input variable, then the controller is static (stationary, or time-independent) and every sample of data in a measured set contributes to the regression process. That is, for example, a regression could be carried out on data from a single run by one pilot since one run would contain many sample data points. If, however, the controller input variables include time explicitly (to reflect a development or sudden change in controller behaviour) then it is necessary to employ samples of several data sets at each time point in order to establish the regression. This would require the same manoeuvre being repeatedly carried out, by either the same or different pilots, in order to obtain a number of samples at any given time.

In the application described here, this increase in the dimensionality of the input space was avoided by explicitly subdividing the task into discrete phases - each of which is addressed by a time independent controller of type shown in Eqn 1.

CART can also be used in the classification mode where the *action* is to designate membership of a class based on the value of the input variables. Fig.1 shows an example of a classification rule which is incorporated into the recovery from engine failure task.

In practice, when Eqn 1 is used to model pilot behaviour, it is recognised that the pilot's demand can depend on prior values of the data and, in particular, there is a finite reaction time before the pilot responds to new information and system delays to account for. Therefore it is appropriate to include previous values of the state values among the input variables. For reasons of economy, it is normal to reduce this information to the mean over a small number of previous samples, or, as is done in the present

work, use delayed values in the induction process. The delay is considered to account for a combination of pilot reaction time and system response. The value for the delay must be found by experimentation since the pilot's anticipation, in critical phases, can vary the perceived reaction time.

### 3. The EFRS task

The recovery from engine failure on take off from an oil-rig is a task of significant importance which requires both decisive action by the pilot and subsequent application of considerable skill. In the normal run of events, the helicopter rises vertically (alternative strategies provide for a certain amount of backing off) to the Critical Decision Point (CDP) and then pitches down in order to accelerate into forward flight while continuing to climb away. When, during this procedure, engine failure is announced then the action taken by the pilot depends on whether or not he is at, beyond, or not yet reached, the CDP.

If he is beyond the CDP, the pilot pitches nose down and reduces collective pitch, thus reducing the power required, and aims to accelerate to a forward speed (typically 40 knots) where the efficiency of the rotor has sufficiently increased to allow a gradual climb out within One Engine Inoperative (OEI) power limits. During this recovery manoeuvre, the pilot must continually adjust the collective pitch to maintain power and rotor speed within limits while minimising the height loss and avoiding the deck edge and any rig projections.

In the case of engine failure before reaching CDP it is necessary to recover to the deck of the platform by a controlled vertical descent. Here the strategy is to maintain a level attitude and, by reducing collective, conserve rotor speed. The descent is arrested by an increase of collective when the deck is approached. The increase in collective is made irrespective of rotor speed - thus trading the kinetic energy of the rotor for a reduction in contact velocity.

### 4. The Advanced Engineering Simulation Facility (AESF)

For the investigation into recovery from engine failure, a high fidelity fixed base simulator (the GKN-WHL Advanced Engineering Simulation Facility) was chosen

rather than the more modest configuration used for earlier ILS and Carefree Handling trials. The AESF comprises a suite of Silicon Graphics workstations connected via ethernet. A three channel 180° x 45° field of view is generated by an ONYX Reality Engine 2 and passed to three SEOS projectors. Fully textured graphics are provided and the Paradigm Simulation product 'Vega', with marine effects included, adds such realism as a dynamic ocean and wake effects for the oil-rig legs. The scene is updated on the basis of positional and rotational data received from the flight model running on a 320VGX Powervision machine.

The cockpit used for the study is based on a Westland 30, and flown with conventional helicopter controls from the right hand seat. A head-down primary flight display is provided, with standard helicopter instruments. A marker was included on the heading display to indicate the heading to fly in order to reach the oil rig. This feature enabled pilots to find the oil rig during task and simulator familiarisation. The dead bands associated with the pitch, roll and yaw axes were tuned for the rate command response type being flown. Audio cueing was also included to advise of engine failure in the initial data gathering trial and subsequently to advise on control actions when evaluating the final system.

The flight model used for these studies was a conceptual model configured around a Westland Lynx Mk. 1 with maximum all up mass. The model incorporated direct rate demand/hold in pitch, roll and yaw but conventional helicopter response characteristics in the vertical (collective) axis. For the EFRS task, the model was modified to allow either or both engines of a twin configuration to be failed and restarted.

### 5. EFRS Requirement Specification

A functional specification for the EFRS (Ref. 4) was produced by GKN-Westland and placed on the system implementers, GCU, in an emulation of contractual situation. The functional specification incorporated the guidelines developed as part of the ISSAFE project. The document was structured as shown in Table 2.

Table 2. Functional Specification Structure

<i>Section</i>	<i>Contents</i>
OVERVIEW	- Intended Operational Requirements - Applicable Aircraft Hazards - Airworthiness Requirements
USER DESCRIPTION	-System Operation - System Goal/Hazard States - Definition of Certified Domain
FUNCTIONAL SPECIFICATION	- Data Specifications - Functional Requirements: Normal Operation - Functional Requirements: Failure Mode Operation
CONSTRAINTS	- Software Design Constraints - Hardware Design Constraints
LIFE CYCLE ASPECTS	- Documentation - Component Testing and Integration - Expansion Requirements
DELIVERABLES	- Software - Operating Instructions - Design - Maintenance
SYSTEM TEST STRATEGY	- Acceptance - Integration - Performance

The concept of defining the domain in terms of goals and sub-goals and relating these to relevant state parameters became a crucial aspect of the design process. An extract from the System Goal/Hazard States section will serve to illustrate this point:

Sub-Goal 1. On confirmation of a failed engine the pilot must manoeuvre the aircraft in such a way to increase airspeed and keep the rotor speed within limits. This can be achieved by lowering the aircraft nose and applying maximum one-engine-inoperative (OEI) torque.

Sub-Goal 2. The act of increasing airspeed through height loss will assist in preserving rotor speed. The aim

of sub-goal 2 is to have a zero rate of descent, achieved by gradually increasing aircraft pitch and reducing torque.

Sub-Goal 3. With the aircraft safely under control a positive climb is initiated, regaining lost height.

An annex to the specification contained data from 162 runs of skilled pilots carrying out a recovery from engine failure. This data is assumed to contain the necessary information to allow the system implementers at GCU to generate an appropriate range and sequence of control actions to recover the vehicle after an engine failure in compliance with the requirements of the specification. With this rule-based approach, no information on vehicle dynamics or performance is given or allowed: all of the necessary information for the production of the system has been captured by the skilled responses of the trials pilots.

## 6. System Design and Implementation

Since the specification includes no dynamic requirements in terms of convention control system design and no information or modelling of the helicopter flight dynamics, the design process cannot follow the conventional route. The system designer should bring no background knowledge to the situation - all of the knowledge to be used should reside in the data measured. He must establish, at the outset, the design framework within which to apply the rule induction process and to recognise that the extraction of rules from the data is a separate scientific study. In essence, the rule induction aspect is handed over as a research task and a specification for appropriate rules is received back. The software design and implementation, a routine software production exercise, is thus kept separate from the more exploratory research task of extracting the rules. This separation of tasks in the overall design process gives a clear definition of roles and responsibilities.

Earlier experience in the ISSAFE applications had suggested that a lack of structure in the design approach had impeded progress. Therefore a structured approach was considered to be essential for the EFRS and a structure based on modelling the pilot's recovery procedure, shown in Fig. 2, was adopted. The

structure diagram shows the sequence of normal operation, followed by a failure, followed by recovery. Recovery is either an abort take off or a continued fly-away. Fly-away is the sequence of sub goals 1-3 described in the specification, whereas the abort is specified by the designer to consist of sub-goals 4 and 5 corresponding to the vertical descent and arrest phases of the abort manoeuvre respectively. The structure shown in Fig. 2 was directly used as the basis for a design using the Jackson System Development (JSD) method. This method is process-oriented and has a strong modelling emphasis, and therefore well suited to an application involving behaviour cloning. Within the framework determined by this structure the rule induction research task is clear. First, the decision to abort or flyaway when engine failure is detected is induced as a classification rule. As expected, the decision turned out to depend solely on whether it occurred before or after a critical height. The rule is shown in Fig. 1 where the value of 285 feet (45 feet above the deck) for the critical height was induced from runs in which the pilots were asked to choose their own CDP. Each sub-goal is then addressed to induce rules that will replicate the associated control strategy. The modelling assumption is that there is a consistent strategy of the form shown in Eqn. 1 being addressed throughout the sub-goal. Clearly the transition through the sequence of sub-goals is supervised by the framework in an explicit fashion. For example, the Fly-away/Abort decision is made at

the time of engine failure and not subject to being revisited subsequently. The point that such sequencing decisions are irrevocable is important - for it is clearly unacceptable for the system to transit from addressing sub-goal 2 to addressing sub-goal 4 in successive time steps.

The research task for rule induction is

- (i) Select the interval of data that is appropriate to the sub-goal (usually done by inspection of the data),
- (ii) Select candidate input variables for the control strategy (this is an exploratory process; a small number of variables should be selected initially),
- (iii) Induce rules from the selected data using the chosen input variables.
- (iv) Supply the induced rules to the system implementer in the form of a decision tree.

With five sub-goals and four demand axes to consider and a range of input variables to explore, this is a considerable task. Table 2 summarises the specification passed to the induction researchers. For each sub-goal there is:

- (i) a hypothesised control strategy,
- (ii) a list of demands for which rules are required
- (iii) a list of possible input variables
- (iv) a terminating condition.

Table 2. System Design Sub-goal Detail

Sub-goal	Strategy description	Response variables	Input variables	Terminating condition
1	Maintain rotor speed within limits and pitch attitude: 5° to -10°	Long. cyclic pitch. Collective pitch.	Rotor speed. Pitch angle.	IAS = 45 knots
2	Pull up and keep rotor speed within limits	Long. cyclic pitch. Collective pitch.	Pitch angle. Rate of climb. Rotor speed.	Rate of climb = 0
3	Maintain rotor speed within limits and achieve positive rate of climb. Maintain airspeed.	Long. cyclic pitch. Collective pitch.	Rotor speed. Rate of climb.	(Height = Final height) .AND. (Rate of climb > 0)
4	Maintain rotor speed within limits and keep level pitch attitude	Long. cyclic pitch. Collective pitch.	Rotor speed. Rate of climb rate. Climb angle.	Height = 10 feet
5	Decelerate and maintain level attitude.	Long. cyclic pitch. Collective pitch.	Height Rotor speed	Height = 0.2 feet

Some of the terminating conditions, such as achieving 45 knots are directly related to the specification but others such as

- (a) the end of the flyaway - 'steady climb established'
- (b) end of the abort descent - 'close to the deck'

are debatable and judged largely on the physical reasonableness of the criteria and inspection of the data. The lateral demand is not included in Table 2 because a simple wings-level criterion was employed based on deviation of roll angle from the trim value.

Pedal demand is not included because heading control was not considered in the initial, simplified, rule set.

Each sub-goal phase is addressed to develop rules which approximate the control behaviour and experience has shown that it is a mistake to include too large a set of the available data variables with a view to letting the inductive software select the most significant. Our experience on the ISSAFE project has been that such an approach is unlikely to succeed and a parsimonious approach of including only key relevant variables is recommended.

Fig 3 shows a typical rule for the normalised longitudinal cyclic pitch demand using input variables pitch attitude and rotor speed. A time delay of 1 sec. is included in the processing to represent pilot reaction time and system delays. The responses include longitudinal cyclic pitch demand, pitch attitude and rotor speed. Two contour plots are shown- one of the presented data and the other of the modest rule induced from it. The decision tree form of the rule is shown in Fig 4.

The complete set of induced rules was passed back as a specification to the system designer to be integrated into the advisory framework. The calculated demands were converted into simple increase/do nothing/decrease commands for transfer to the audio cueing system of the simulator. The complete EFRS system (Ref. 5) included modules for the handling of communication to and from the host simulator and for validating the incoming data. The advisory process was suspended when rogue data were identified.

After testing at GCU, the system was ported, as a set of C routines, to the simulator at GKN Westland for a brief evaluation of the performance criteria.

## 7. Evaluation

The system was integrated as a simple frame time call into the GKN Westland Advanced Engineering Simulation Facility used for the initial data collection. A series of engine failures were given before, after and at, the CDP in order to establish in the first place that the system gave the correct flyaway/abort advice and then sequenced correctly through the sub-goals. The validity of the advice for any given phase was also noted. Both of these aspects were established, the principal difficulty being in conveying advice to the pilot through the sound system. (It was fortunate that no heading advice was being given to control the yaw axis) It was clearly difficult for the pilot to assimilate and respond to advice for several control axes being delivered, interleaved, via a single audio channel.

In the time available, there was no opportunity to pursue the full range of acceptance criteria stipulated in the requirement. Nevertheless, there was satisfaction that this was one application that had gone all of the way through the stages relevant to the ISSAFE project. The task was not actually to develop a functioning system but to emulate the specification, development and implementation stages in order to map out and explore a practical procedure in this new technology.

## 8. Discussion

A significant source of technical difficulty in the ISSAFE work was found to be in inducing rules of practical benefit from the large volume of data collected from experiment - in this case the flight simulator. Some management of the induction process was clearly required. The manner in which this was achieved through a decomposition based on a high level modelling of the task has been described in the previous sections but there are several issues that have arisen as a consequence. One is related to the validity of the model structure (or decomposition). The adoption of an imposed structure places constraints on what can be achieved by the induction process. The ideal situation would be to derive rules for the whole

operation of the system, including a natural evolution of a fly-away/abort and sub-goal sequence in place of the independent treatment followed for this work. There are several considerations here: first, the induction task would be much more complex and possibly involve a mixture of classification and regression -which is outside the scope of current tools and second, it would be necessary to hold all of the data subsequent to engine failure in order to, for example, 'look back' to see whether flyaway or abort was decided upon prior to addressing the current control behaviour. The same applies to the sub-goal sequencing in order to locate the rule in the appropriate phase of the task. In addition, the safety analysis of such rules, whereby advice at a particular point is guaranteed to be relevant to the current position in the task, phasing would be difficult to address (countering one of the stated benefits of the AI approach).

Another issue is the assumption behind the control behaviour of Eqn. 1. Essentially, this mapping is a pilot model and if one assumes small deviations from a reference position then Eqn 1 is a simple linear gain - albeit in several variables - for all of the phases and axes. It is recognised [6] that the pilot behaviour is influenced by the system that is being subject to control and it is interesting to examine briefly what this means in the context of pilot behaviour cloning. The cross over model of pilot behaviour assumes that the open loop transfer function between tracking error and system output, that is across the pilot and the system together, is

$$k \frac{e^{-\tau s}}{s}$$

for gain  $k$  and delay  $\tau$ . A rule that is a simple gain modified by the delay of a pilot reaction time suggests that the system to be controlled should have a transfer function  $\sim 1/s$ . In the roll axis the rotational inertia is small so that the roll rate follows the demand almost instantly so a simple integration gives the roll angle - the error in which was, in fact, used for roll rate control. In the pitch axis, the inertia is larger and so the transfer function between demand and the pitch angle is likely to resemble

$$\sim \frac{1}{s(\tau s + 1)}$$

For small values of the aerodynamic coefficient,  $\epsilon$ , satisfactory modelling should be possible and this would appear to be the case in this work. Nevertheless it may be that the rule induction strategy would benefit from a processing of the data that allows for pilot adaptation to the system dynamics of the output cues that he uses for his behaviour.

An issue particularly relevant to the ILS task and mercifully not a dominant factor here is the problem caused by very skilled pilots. An example of this would be a pilot who, on the announcement of engine failure, threw the helicopter into the correct pitch angle and descent rate by a single, open loop, action of the controls. The consequence is that during the dive phase to gain airspeed there would be no control activity upon which to build an induction of a fully attended control behaviour. While such a 'right first time' control strategy may be commendable it is not within the scope of the structure adopted in this investigation. In fact, no such situation was observed in the EFRS trials although it was commonplace in the ILS exercise and caused significant problems. Finally, avoidance of deck strikes were not part of this study and no attempt to extract a relevant strategy was incorporated.

## 9. Conclusions

- Despite the limitations of the system and its evaluation the main conclusion is that it has been possible to demonstrate the feasibility of design and building a system from a functional specification and a set of data from skilled pilots. It has followed, albeit in a modest way, the guidelines and safety analysis of the ISSAFE project.
- The structure of the task into goals was a key aspect of the specification and system design. This aspect is not well supported by currently available AI tools.
- The assumed form for the pilot model may be unduly restrictive for more complex aspects of the system dynamics.
- Audio advice covering more than one axis is almost impossible to follow. The pilot finds it difficult to coordinate control activity when

received aurally. An improved interface for advice would probably use a head up display with appropriate symbology for cyclic control or active feedback provided through the cyclic in the form of 'stick pushers' and 'shakers' which are being evaluated in on going research at GKN Westland.

## 10 References

- 1 Cussens J, "Using MARS to learn the ILS task", ISSAFE/9039/3/GCUC/3036/R/A\*. August 1995.
- 2 Bradley R, "Box Building Techniques Applied to Limit Protection for Carefree Handling", ISSAFE/9039/3/GCU/3083/R/A\*, December 1996.
- 3 Brieman L, Friedman J, Olshen R A, Stone C J, "Classification and Regression Trees", Chapman Hall 1993.
- 4 Baker P D, "Functional Requirement Specification for A Failure Recovery System", ISSAFE/9039/18/WHL/4048/R/1\*, June 1996.
- 5 Bradley R, "System Design for the Functional Requirement for an Engine Failure Recovery System", ISSAFE/9039/3/GCU/3080/R/1\*, November 1996.
- 6 McRuer D T, Krendel E S, "Mathematical Models of Human Pilot Behaviour", AGARDograph No. 188, January 1974.

\* Details may be obtained from:  
ERA Technology Ltd.  
Cleeve Road,  
Leatherhead,  
Surrey,  
KT22 7SA  
UK

## Appendix

### Classification and Regression Trees

The CART package is used to construct binary decision trees, with the prediction rules for the response variable based on the values of a set of inputs. If the response variable has a finite number of classes then the classification mode is used and the inductive rule obtained will predict the class to which the object belongs based on the value of the inputs. If the response variable is numerical, then the regression option is used and this will predict the numerical value of the response variable based on the inputs.

CART works by building a series of progressively larger trees and then, through a testing process, the optimal tree is found. To obtain good predictive rules, it is important to select the correct size of tree. If the decision tree is too large, it will tend to overfit the data used to construct it and will perform poorly with other data. Conversely, too small a tree could result in certain relationships between variables being ignored. The best way to obtain a good tree depends on the amount of data available. For large data sets the data should be separated into a learning sample (67%) and a testing sample (33%), from which the trees are constructed using the learning sample and the best tree is selected by testing with the testing sample. For smaller data sets, as selected in the current work, cross validation is the preferred estimation method within CART. This randomly separates the data into learning and testing sets and through a process of testing the best decision tree is chosen.



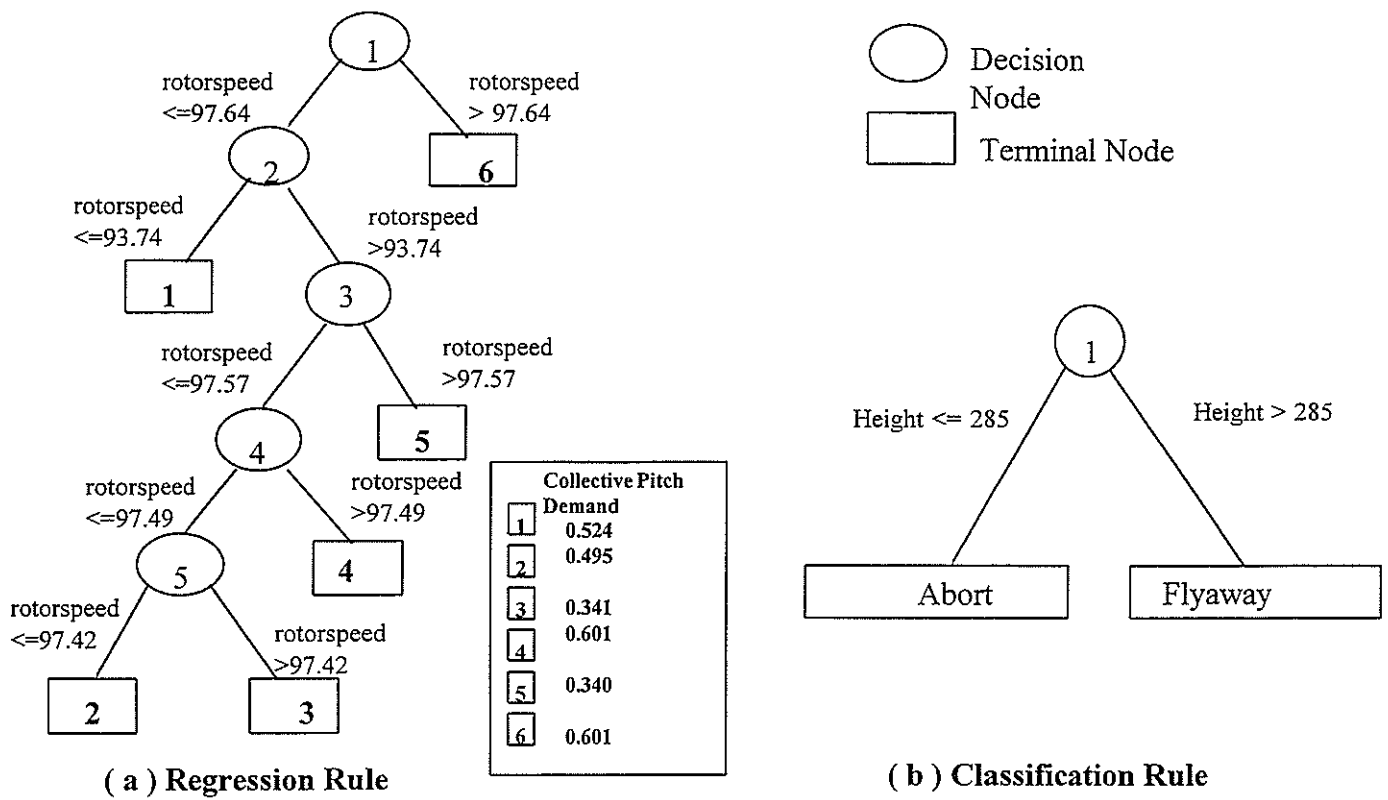


FIGURE 1 : Regression and Classification Decision Rules

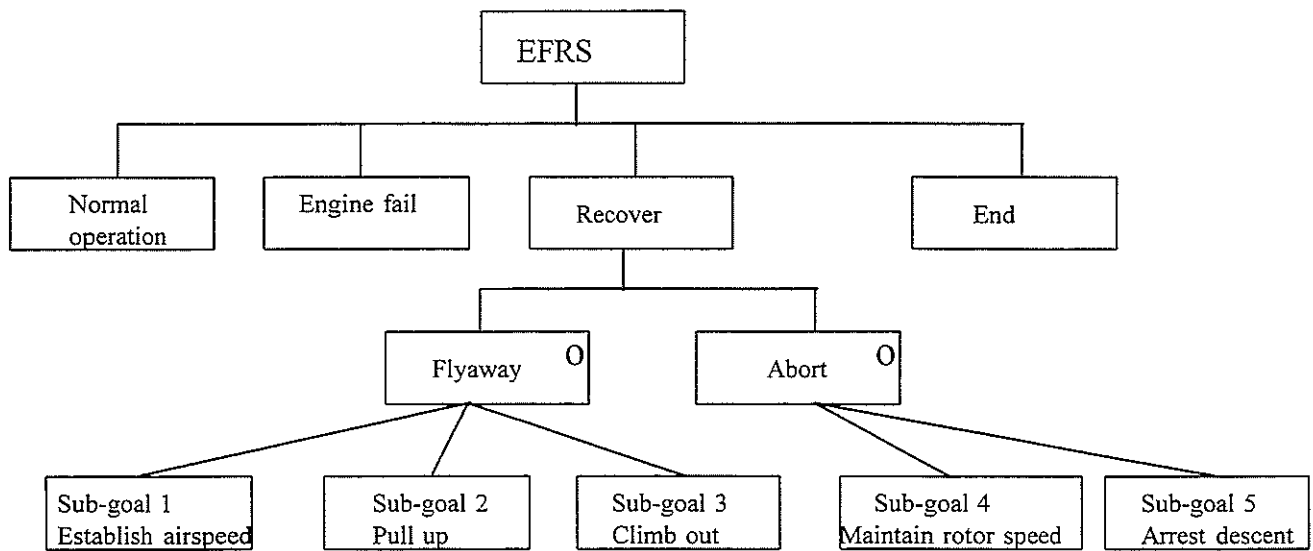
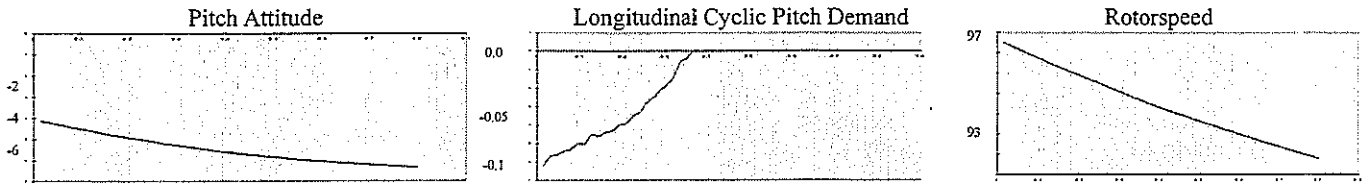


FIGURE 2 : Structure Diagram for Model of Recovery Process



Actual Values of Longitudinal Cyclic Pitch Based on Rotorspeed and Pitch Attitude for Sub-goal 1

CART Values of Longitudinal Cyclic Pitch Based on Rotorspeed and Pitch Attitude for Sub-goal 1

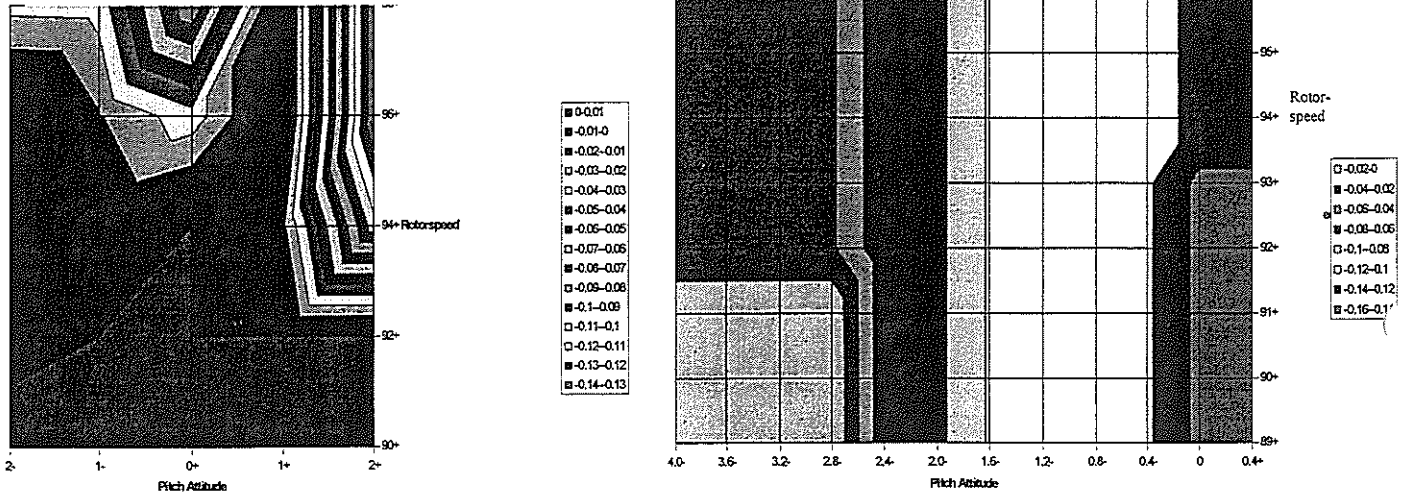


FIGURE 3 : Rule Induction for Sub-Goal 1

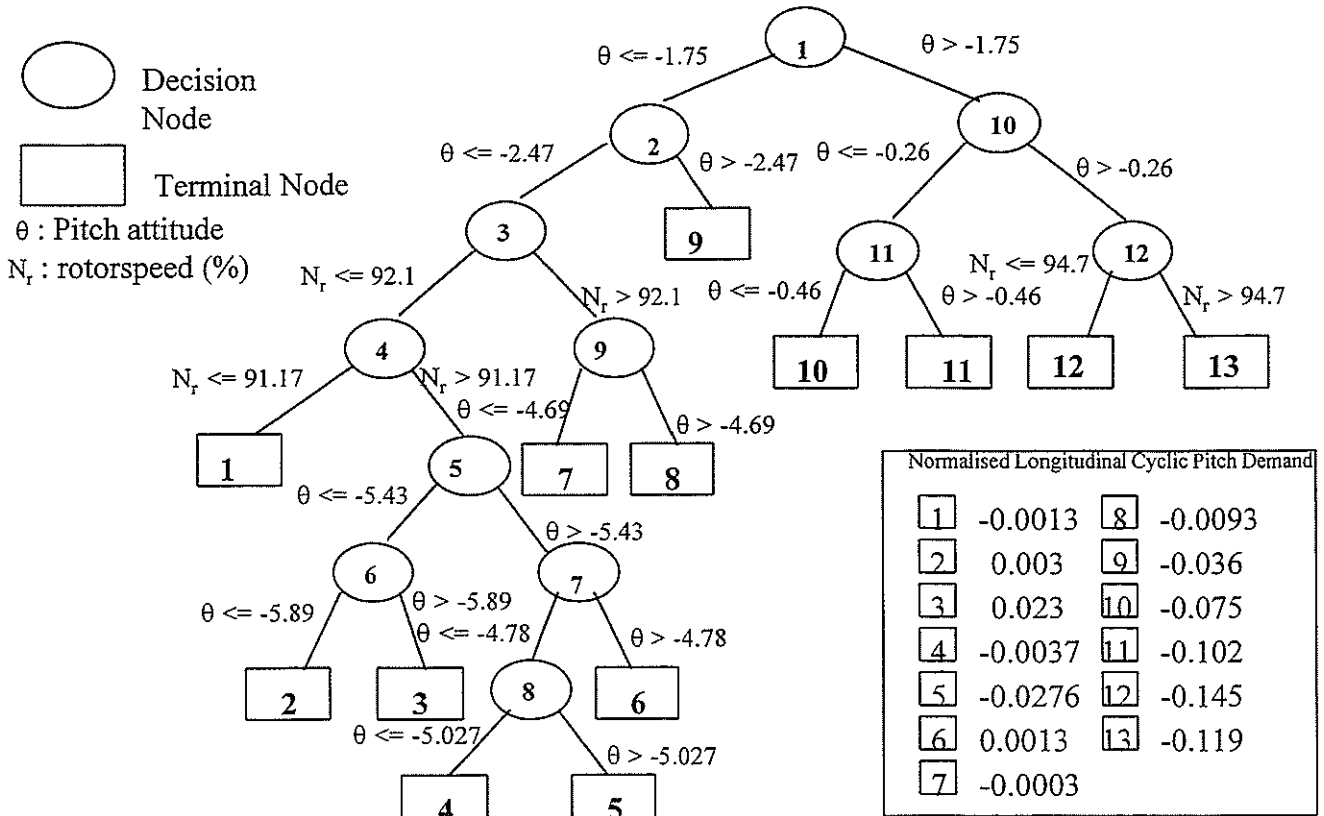


FIGURE 4 : Decision Tree for Sub-Goal 1