J. Indian Inst. Sci., May–June 1990, 70, 213–229.
© Indian Institute of Science

An expert system pilot aid for remote control of an unmanned vehicle—ESPA

KANAKA KRUPAKARAN* AND V. V. S. SARMA
Department of Computer Science and Automation, Indian Institute of Science, Bangalore 560 012, India.

Received on January 12, 1990; Revised on March 1, 1990.

Abstract

This paper proposes the use of an expert system as a pilot aid to guide a pilot remotely controlling an unmanned vehicle. It outlines the development of the proposed expert system, justifies the need of such an intelligent pilot aid and brings out its main features.

Key words: Expert system, pilot aid, remote control, unmanned vehicle.

1. Introduction

The Defence Advanced Research Projects Agency (DARPA) of the USA has been involved in the development of 'pilot's associate' project as part of its strategic computing initiative program. The project is directed towards providing the pilot of a single place fighter aircraft with the support and expertise of a 'phantom flight crew'. Using the concept of an integrated cockpit, the pilot is provided with the support of four interactive expert systems: a situation assessment manager, a tactical planning manager, a mission planning manager and a system status manager. While the authors have initially considered the study of such a system for the light combat aircraft (LCA) being developed in India presently, as a first step they have chosen to develop such a pilot-aid system for a remotely piloted vehicle, as this is also one of the major projects being pursued by the Defence Research and Development Organisation (DRDO).

A remotely piloted vehicle (RPV) belongs to the class of unmanned aircraft, extensively used for the following military applications:

1. Photo reconnaissance,
2. Target acquisition/designation/damage assessment, and
3. Electronic counter-measures.

* Present address: Flight Test, Telecommand and Tracking Division, Aeronautical Development Establishment (ADE), C. V. Raman Nagar, Bangalore 560 093.

This vehicle presents an ideal solution to many defence problems[1]. The small size and reduced design constraints make it less vulnerable and lead to better survival in military operations. Deeper penetration into enemy territory is possible since there is no element of risk to human life. Its mobility, lower cost and reduced training investment make it a very useful tool for defence applications. However, there are a few serious limitations. Firstly, the range of operation is restricted to about half a kilometer from the pilot as navigation and control is by visual observation of the air-vehicle and its attitude, heading, etc. Secondly, the missions are limited to daylight under good visibility as set by the control requirements.

Full utilisation of an RPV can be made possible only if the above limitations are overcome. With increasing demands made on the pilot's visual perception, cognitive processing and manual control, it is desirable to have an alternative system that performs in parallel the flight tasks of a pilot and offers expert advice regarding situation assessment, mission planning, system status, etc. Hence a feasible solution is to have a dynamic, heuristic model of the pilot's decision-making process as applied to command and control of the RPV[2]. This enables control beyond visual range. The requirement of a consultant system which is competent enough to match the decision-making capacity of a human pilot calls for an expert system. The environment of an RPV is more suited for developing an expert system as timing is less critical and the consequences of wrong decisions are less catastrophic. The pilot of an RPV is relatively more comfortable as he is in a benign environment unimpaired by vibration and noise. Hence the system invokes more confidence in the pilot so that under critical conditions he has no second thoughts in accepting the advice offered by the system. Thus the pilot aid provides a safe test-bed for assessing the performance of expert systems for real-time applications.

2. Expert systems: An overview

An expert system is a program that displays a comparable level of intelligence to that of a human specialist in a narrow domain. Sufficient expertise should be built into the system so that it simulates the expert's behaviour in the chosen domain. Figure 1 shows the basic structure and the various components that constitute an expert system. The domain knowledge has to be extracted from the expert and developed into a conceptual model which is a logical representation of the system. Hence the primary concern is acquisition of domain knowledge and this task is performed by the knowledge engineer who maps the expert's knowledge into the program. The knowledge engineer has the following important tasks:

(i) Identifying one of the several experts.
(ii) Interviewing the experts or entering into an interaction with the experts by any other means.
(iii) Encoding the expert's approach to problem solving in a given situation.
(iv) Providing a validated and sound end-user system.

The success of the system depends on how accurately the following interfaces are done[3]:

1. Expert's interface with the knowledge engineer.
2. Knowledge engineer's interface with the computer.
3. Expert's interface with the computer.

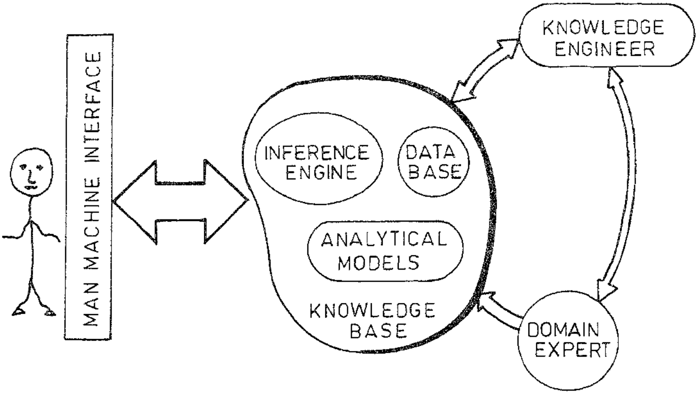


FIG. 1. Various components of an expert system.

Implementation level follows once the domain concepts have been formulated. The expert system differs from conventional programs by way of data-driven mode of operation and use of heuristics. The latter is used to code knowledge used by experts to solve problems intuitively.

The knowledge in the knowledge base is organised on three levels: data, knowledge and control. On the data level is the declarative knowledge about the particular problem being solved and the current state of the parameters involved. On the knowledge-base level, the knowledge specific to the problem area is considered. It has facts and relationships about an application area. At the control structure level, there are instructions as to how the knowledge should be organised and processed. Real-world problems have imprecision and uncertainty associated with the knowledge. Constraint management is responsible for the evolution of database management system into expert database system (fig. 2).

Thus the success of the expert system primarily depends on the sound selection of the experts, on how well the knowledge engineer translates the expert's knowledge into a program and finally on the validation of the system.

3. Role of the intelligent pilot aid for RPVs

In the control of an RPV two levels of operations are involved[4]:

1. The pilot is in complete control (within the line-of-sight) and is aware of all the range of available choices enabling him to decide the appropriate action.
2. Fully automated decision-making process in which a sequence of intelligent decisions made by the computer without human intervention results in an automatic action.

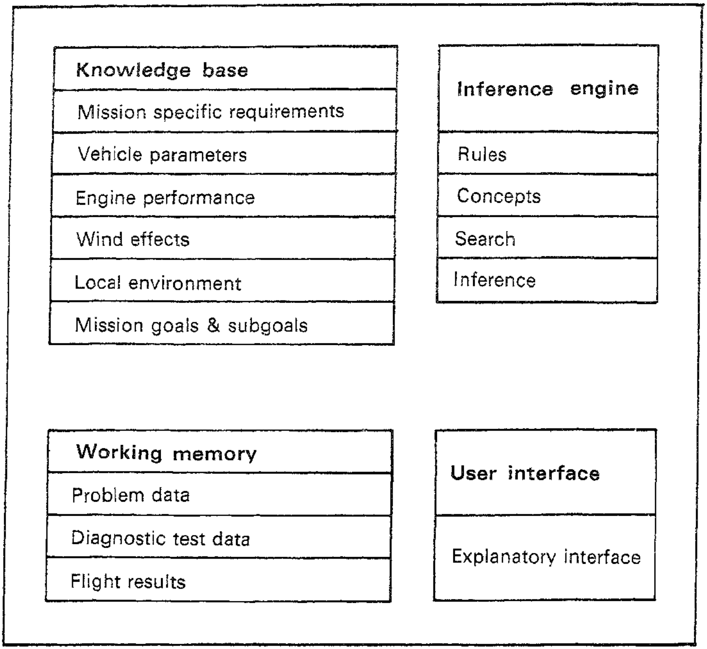


FIG. 2. System configuration.

In the latter operational phase the pilot aid has the following important tasks[5]:

(i) Mission assessment will assess tactical environment (*i.e.*, weather, terrain, targets, threats) and the capabilities of onboard systems to meet the demands of the mission and then recommend the best course of action in the present tactical situation.

(ii) Mission status assessment will identify malfunctioning of the systems and advise the initiation of appropriate action depending upon the severity of the faults.

(iii) Situation assessment will keep a watch on the critical parameters and when the limiting values are exceeded the pilot is warned of the impending danger and given proper advice.

(iv) On many occasions the pilot does not limit his actions to those points covered in the flight manual[6]. While making decisions he must be concerned whether he is ahead or behind the FEBA (forward edge of the battle area) and the possible threats. Automation of the above decisions is one of the aims of the ESPA.

4. Main features of the system

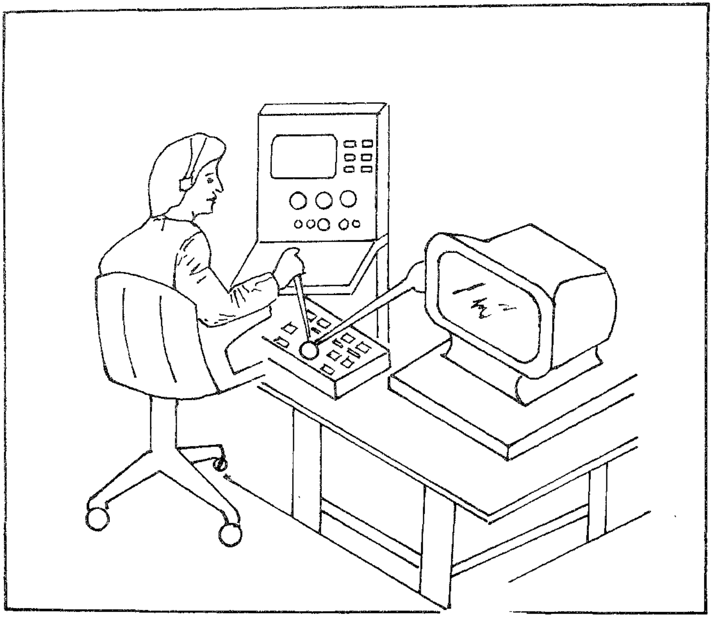
Figure 3 shows the ground station with the pilot and his assistant. The user, *i.e.*, the pilot, communicates with the computer *via* graphic displays plus question and answers in natural language. Voice commands could be used to alert the pilot during emergencies. The system receives information from the sensors and other onboard systems through the communication links which are connected to both the system and ground-command panel.

4.1. *System configuration*

Figure 2 is a schematic of the total system configuration. There are two levels of execution: 1. knowledge level, and 2. implementation level.

4.1.1. *Knowledge level*

This involves the identification of an expert and subsequent knowledge acquisition by the knowledge engineer.



.FIG. 3. Ground station with the pilot and his assistant.

4.1.1.1. *Expert identification*

Acquiring domain knowledge from the expert is the central task in building a sound system. It is required to identify an expert whose depth of knowledge in the field is well recognised. The present system involves experts in the following fields: 1. launch and recovery, 2. controls, 3. engine performance, 4. navigation, and 5. payload and mission planning.

No specific guidelines are available for determining expertise in an area totally new to the knowledge engineer. This is a very important and critical aspect as the success of the resulting system depends on the sound selection of so many experts. In general, this can be done in two ways.

Personal experience: The knowledge engineer knows the experts personally and has consulted them and found that they give sound advice.

Recommendation: The experts may have been recommended by other people like the Project Director*.

4.1.1.2. *Knowledge acquisition*

'An expert system is only as good as the knowledge which is in it'[7]. The knowledge engineer has to work with experts and the pilot (user) has to be in the picture right from the early stages. With the knowledge of the experts the knowledge engineer has to reconcile the needs of the user and map this to a computer system. So the knowledge engineer has to interact with two classes of people with different roles. The pilot is not aware of the technical details of the hardware and software and can only specify what he expects the system to do while the experts with their knowledge can throw light on the pilot's responses in a given situation. It is generally felt that experts should not perform the knowledge engineering task themselves for the following reasons[7]:

1. They will usually have insufficient knowledge about programming and expert system techniques.
2. They will find it difficult to describe and translate their knowledge in a machine-usable form. All the knowledge must be acquired before implementation and this is a challenging task. The knowledge engineer has to prepare himself, before his interview with the experts. He should have some background reading about the types of problems encountered, the terminology, accepted methods and tools. This is very important to make full use of the expert, as the expert will be interacting at his level of versatility and may not necessarily understand the problems of the knowledge engineer.

There are two possibilities:

1. The expert can tailor the information to meet specific needs of the system.
2. The knowledge engineer has to distinguish between the irrelevant and the relevant.

*In the work reported in this paper, the first author working in the field since four years could identify the experts easily through her personal experience. It is felt that for a project of this nature, problem assessment, expert selection and identification for a novice would take considerable time, effort and patience.

Both of the above are not easy tasks as the former requires that the expert has to take extra initiative to find the requirements of system development which rarely happens. The latter seems more feasible but is time consuming as the knowledge engineer has to have a clear picture of the context by a suitable task analysis of the user's and expert's domains. Thus building a knowledge base in itself is a problem-solving activity in its problem space. The domain knowledge could be both deterministic and subjective in nature which calls for both algorithmic and heuristic approaches.

The knowledge elicited from the experts is classified into four types:

1. *Declarative knowledge*: This is the knowledge represented by static symbolic expressions leading to a precise description of a concept. Frame-base representation is adopted where an object is represented by a data structure containing a number of slots, with each slot filled with one or more values^{8}.

Example

{SYSTEM

Is-a	:	RPV
All-up-wt	:	65.0 kg
Endurance	:	2h
Max-alt.	:	700 m
Max-speed	:	85 knots
Max-fuel	:	10 litres
Max-payload	:	18.0 kg}

2. *Procedural knowledge*: This is the knowledge in the form of a program which details the sequence of actions to perform. Production rules are useful for representing knowledge of the following type:

IF ⟨situation⟩ THEN ⟨action⟩

situation ⟶ action.

The situation is any state that could arise during the performance of the task and the action specifies an appropriate response.

Example

IF: Wind direction is headwind AND
wind speed is >25 knots

THEN: Call-off-flight.

3. *Causal knowledge*: This is the knowledge at a theoretical level expressed as mathematical models, for example, those used in the calculation of range and bearing of the vehicle.

4. *Heuristic knowledge*: This is the procedural knowledge in the form of decision trees combined with the more intuitive knowledge of the expert. Hence the knowledge engineer has to precisely understand the decision-making process of the expert. The knowledge base

is composed of a considerable amount of dynamically changing information and real-time data. High degrees of uncertainty are associated with both the data and reasoning processes. This is achieved by assigning a confidence level in each of the inference mechanisms as shown below:

IF:

1. Range of waypoint $w1$ is less than permissible range
2. AND Threat factor is less than .30
3. AND Okay-weather condition

THEN:

There is suggestive evidence (0.6) that waypoint $w1$ can be traversed.

4.1.2. *Implementation level*

Table I illustrates the different AI techniques involved in the different system functions. The knowledge engineer must take into account both AI methodologies and human performance to provide an adequate integration of the above knowledge levels.

AI languages: Several computer languages such as LISP and PROLOG have been specially adapted to artificial intelligence since they aim at representing data at the semantic level. Thus they differ from the conventional languages and provide a highly interactive and flexible programming environment. This offers a sound consultation facility with the pilot and through windowed displays the system can offer explanations to its reasoning process. Another alternative is to use an expert system shell. A shell is an expert system emptied of its knowledge base and provided with inference mechanisms and user interface facility. But it can be used only for restrictive domains where the inference mechanisms are almost identical.

The examples illustrated in the following paragraphs bring out the different implementation techniques to perform intelligent processing of information stored or retrieved from the data base.

Control structures: The three main control structures are sequence, selection and iteration. In rule-based system, control mechanism has to be embedded in the rules. The sequence of rules has to be in such a way that it leads to efficient and quick-search algorithms. For example, in mission planning:

```
IF    : The mission range exceeds possible range
THEN  : Confirm = 1
IF    : Confirm = 1 AND available fuel = Max. fuel capacity
THEN  : Confirm = 2
IF    : Confirm = 2 THEN:Start new mission profile AND confirm = 0.
```

To develop a more efficient algorithm the rules have to be arranged in the order in which they are to be applied and where certainty factors are present they are arranged in the decreasing order of the strength of the conclusions[10] The most critical of the rules should be fired first. It is required to give priority to those rules that are used most frequently.

When the number of rules becomes very large, in a given situation it is not required to adopt a blind search through all the possibilities. The term 'conflict resolution' refers to how the system behaves in a situation where several rules could be applied. The system should decide which set of rules is applicable at any given stage thus avoiding the necessity of going through the entire set of rules. For example, in the fault diagnostic algorithm the presence of a fault in any of the subsystems triggers rule 1. Only if rule 1 is triggered, it is required to scan the set of rules to find which subsystem has problems and suggests the appropriate action to be taken.

Constraint management: In the automatic flight path planning, there are many constraints on the feasible path. The search problem is to find a suitable route from launch to recovery *via* the chosen waypoints satisfying the following constraints: 1. keep up time, 2. enemy threat, 3. not to exceed the permissible range limit, and 4. avoid rough weather.

It is required to keep track of each point in the route. If the point is accessible from its nearest neighbours, then the decision to follow a certain route depends upon the constraint value contributed by each of the above factors.

The feasibility factor between points *i.e.*, FEASIBLE (source, destination, constraint factor) decides the route. The constraint factor indicates how difficult it is to travel from point *source* to point *destination*. Thus in the specified constraint space the decision to follow a specific path is taken.

Dealing with uncertainty: The feasibility factor $FE1$ of the vehicle V passing through waypoint $W1$ is obtained as shown in fig. 4.

$$FE1 = \text{Pos-range}\,(V, W1)^{*}\ \text{Threat-factor}\,(V, W1)^{*}\ \text{Weather}\,(W1)$$

Figure 5 illustrates a fault tree. If E is a fault event connected to fault events $E1, E2 \ldots En$ by a logic gate, then the probability of occurrence of E is defined as[11]:

$P(E) = \prod_i P(Ei)$ if the logic gate is AND.

$= \sum_i P(Ei)$ if the logic gate is OR provided the basic events are independent.

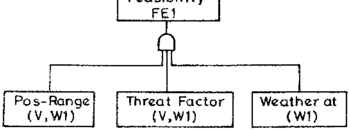


FIG. 4. Calculation of feasibility factor.

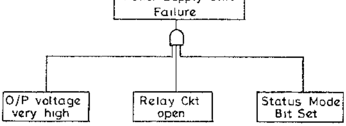
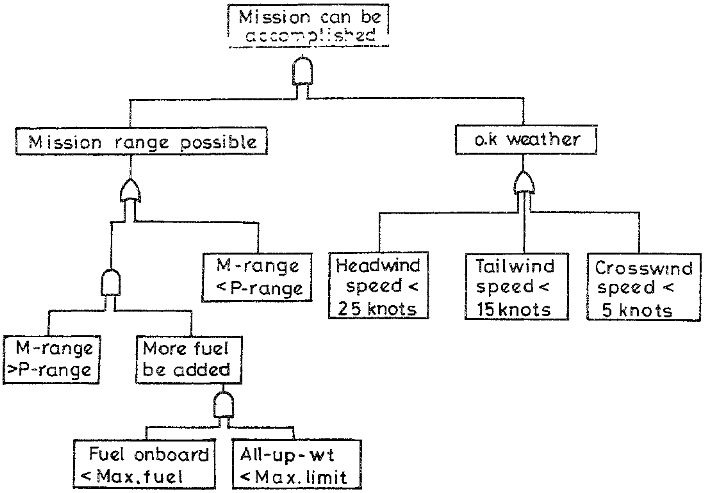


FIG. 5. Fault tree illustration.



M-range – Maximum range planned in the current mission; P-range – Possible range in the current mission, and Gate – Output event occurs if all events occur simultaneously; or Gate – Output event occurs if any one of the events occurs.

FIG. 6. Relationship between IF-THEN rules used to determine whether mission can be accomplished.

4. *Rules*: Figure 6 is an example of IF-THEN rules used to determine whether mission can be accomplished.

IF: there is evidence that *A* and *B* are true

THEN: conclude *C* is true.

Example

1. IF: mission range possible AND OK—weather

   THEN: mission can be accomplished.

2. IF: mission range <possible range OR

   [mission range >possible range AND more fuel be added]

   THEN: mission range possible.

3. IF: all-up-wt <max limit AND

   onboard fuel <max fuel

THEN: more fuel can be added.

4. IF: headwind AND windspeed < 25 knots
OR tailwind AND windspeed < 15 knots
OR cross wind AND windspeed < 5 knots
THEN: OK—weather.

4.2. *System validation*

Testing the system for completeness and correctness is a challenging job. Mistakes could occur due to the following:

(i) The possibility of the experts committing mistakes.
(ii) The lack of proper communication or misinterpretation of ideas transferred between the experts and the knowledge engineer.

The important point is the realisation of system faults as this can be done only if the knowledge engineer knows what it means to state that ESPA is complete in all aspects. Subjecting ESPA for validation test necessitates close association between the knowledge engineer, experts and the pilot. For such applications, validation has to be real-time as unforeseen situations may arise during the actual flight.

5. Conclusions

Pilot aid is directed at exploring AI techniques as a means to aid the pilot in control of remotely piloted vehicles. It combines the advantages of a conventional algorithmic controllers and sophisticated reasoning ability of knowledge-based systems. The system has to respond to dynamic environmental changes. The system response time and accuracy should meet mission requirements. There are apprehensions as to how well and complete the expert ideas can be formulated into programs. But system validation is possible by subjecting it to thorough real-time analysis. The success will induce more confidence in using AI techniques for more complex applications like pilot aid for a fighter aircraft.

References

1. AUSTIN, R. G. AND ROBERTS, C. J. Horses for courses in RPV operations, *Second Inter. Conf. on RPVs*, Bristol, UK, April 6–8, 1981.

2. JOE, K. C., RICHARD, V. AND ALHAND, C. Expert systems for real-time applications, *Proc. NAECON*, 1985, pp. 1322–1329.

3. REISS, G. Knowledge acquisition and the oleophilic adviser, *Proc. WES TEX-86*, IEEE Computer Society Press, 1986, pp. 102–104.

4. ALLAN, M. D. Back to the future—AI in military decision making, Defence Electronics and Computing, Suppl. 4, *Int. Def. Sci. Rev.*, 1989, 30–32.

5. Silin, K J D — Development of pilot's associate program. *Aviat Week Space Technol.*, 1985, **122**, 69–74

6. John, R J and Richard, E F — *Expert system pilot aid*, 1985, pp 1583–1590, IEEE Computer Society Press

7. Hart, A — *Knowledge acquisition for expert systems*, 1989, Kogan Page, London.

8. Michael, D R — *Expert systems for engineering design*, 1988, pp. 20–27, Academic Press.

9. Munson, K — RPVs—Who are the real remote pilots?, *Jane's Def Weekly*, Aug 24, 1985, **4**, 360–413

10 Alain, B. — *Artificial intelligence–promise and performance*, 1985, pp 108–119, Prentice-Hall.

11. Viswanadham, N., Sarma, V. V. S and Singh, M. G — *Reliability of computer and control systems*, 1987, pp 115–125, North-Holland

12. Chambers, A. B. and Nagel, D C. — Pilots of the future: human or computer, *IEEE Computer*, 1989, **18**, 74–87.

Appendix I

Test results

This is divided into two sessions. *Results 1* and *2* correspond to the preflight and the flight sessions.

Result 1.1 indicates the response of ESPA to changes in onboard fuel availability and the waypoints chosen. *Result* 1.2 shows the dynamic planning of a new mission profile and the results are made available to the pilot.

Result 2.1 After launch command is given, control is handed over to the pilot after the vehicle has reached an altitude of 1000 ft. ESPA then lists down the parameters of the approaching waypoint before setting course.

The position of the vehicle is displayed with reference to time. At 29 min 48 sec, ESPA makes an observation to increase the cruise speed. In the very next cycle ESPA offers advice to correct the altitude.

Result 2.2 shows yet another situation when ESPA warns about the vehicle descending below safe limits.

Result 2.3 is an indication of an onboard power supply failure and ESPA's suggestions.

Result 3.1 is a post-flight analysis report.

Let us proceed to the ⇒MISSION-ASSESSMENT⇐ session

FLIGHT PROFILE -{Enter the following data}

Sl no.	Parameters	Data
1.	Identification no. of the waypoints. Enter as (*W*1 *W*2 *W*3 ...)	(*W*1 *W*2 *W*3 *W*4)
2.	Fuel available on board (in lts)	1
3	Min. altitude (AGL) for safe recovery (in ft)	600
4.	Wind parameters:	
	a Speed (knots)	
	b Direction (indicate as headwind/tailwind/crosswind)	HEADWIND
	c. Angle (degrees)	45
5.	Takeoff wt (in kg)	115

〉〉〉– Press *C* to continue –〈〈〈*C*

Result 1.1

Phase 1:- {Mission altitude selection}

The altitude chosen for the mission is :2500 ft

〉〉〉– Press *E* or *C* –〈〈〈

E – explanation.

C – continue.

E

⇒ The above altitude has been arrived at after accounting for the safe recovery of the vehicle at the highest WAYPOINT to be traversed. ⇐

///– Press *C* to continue –///

C

Phase 2:- {Effect of external disturbances- WIND}

⇒ In the prevailing headwind speed conditions the IAS needs to be increased by 10% to maintain the selected speed. ⇐

*** Press *C* to continue ***

C

Phase 3:- {Cruise-speed selection}

The desired cruise speed for the mission has been found to be 73.92 knots.

///– press *E* or *C* –///
E – explanation.
C – continue.

E

⇒ The above optimal value of cruise speed has been arrived at after the following consideration:

a. to ensure the optimal performance of the engine.
b. to have maximum range.
c. existing wind conditions.

///– Press *C* to continue –///

C

Phase 4:- {Flight planning}

!!! MISSION ASSESSMENT COMPLETED !!!

Comments

The desired waypoints cannot be traversed with the available fuel.

It has been found that if an additional 5 kg fuel is added, the Take-Off-Wt equals Max-All-Up-Wt (120 kg), and the addition of this fuel does not exceed the maximum fuel capacity; hence the mission can be accomplished. It is suggested that the fuel be added and the mission assessment recommenced.

⇒ Press *F* to enter flight data ⇐*F*

Result 2.1

PARAMETERS !!!

The parameters of the approaching waypoint are:

```
Range                  : 9.36522 km
Bearing                : 60.0 deg
Coordinate X           : 4.6825
Coordinate Y           : 8.1109
Altitude               : 1000 ft
Time required to reach: 226.919 sec
```

⇒ Press *Y* to confirm course SET ⇐
and HOLD ON !!!

Y

- CURRENT VEHICLE POSITION -

```
Range                  : 225.864 m
Azimuth                : 54.3356 deg
Heading                : 45 deg
```

```
Altitude              : 2500 ft
Distance travelled    : 225.864 m

28 min 6 sec

                    *- CURRENT VEHICLE POSITION -*

Range                 : 451.727 m
Azimuth               : 54.3356 deg
Heading               : 45 deg
Altitude              : 2510 m
Distance travelled    : 451.727 m

28 min 14 sec

                    *- CURRENT VEHICLE POSITION -*

Range                 : 3250.97 m
Azimuth               : 47.5917 deg
Heading               : 48 deg
Altitude              : 2610 ft
Distance travelled    : 3250.97 m

29 min 48 sec

                        ////// OBSERVATION //////

It has been found that under the prevailing flight conditions, the present cruise speed is
inadequate to reach the approaching waypoint at the stipulated time.

It is advised to increase the speed by 0.97962 F + 01 knots.

                    *- CURRENT VEHICLE POSITION -*

Range                 : 3485.15 m
Azimuth               : 47.0135 deg
Heading               : 46 deg
Altitude              : 2610 ft
Distance travelled    : 3485.15 m

29 min 59 sec

                            ///// ERROR /////

                      Deviation from the set altitude.

ADVICE:- Correct the error of 110 ft in altitude.
```

Result 2.2

```
                    PARAMETERS !!!

The parameters of the approaching waypoint are:

Range                 : 5.82306 km
Bearing               : 60.0 deg
Coordinate X          : 2.9115
Coordinate Y          : 4.6935
Altitude              : 1000 ft
Time required to reach: 153.127 sec

          ⇒ Press Y to confirm course SET ⇐
            and HOLD ON !!!
Y

             *-CURRENT VEHICLE POSITION-*

Range                 : 225.963 m
Azimuth               : 54.1002 deg
Heading               : 60 deg
Altitude              : 1500 ft
Distance travelled    : 225.963 m

5 min 31 sec

                     WARNING !!!

Vehicle descending below safe altitude limits.
Increase the altitude IMMEDIATELY.
```

Result 2.3

```
                    PARAMETERS !!!

The parameters of the approaching waypoint are:

Range                 : 8.62922 km
Bearing               : 60.0 deg
Coordinate X          : 4.3092
Coordinate Y          : 7.1477
Altitude              : 1000 ft
Time required to reach: 226.919 sec

          ⇒ Press Y to confirm course SET ⇐
            and HOLD ON !!!
Y
```

```
                    *-CURRENT VEHICLE POSITION-*

Range               : 912.562 m
Azimuth             : 48.8086 deg
Heading             : 55 deg
Altitude            : 1510 ft
Distance travelled  : 912.562 m

38 min 18 sec

                    *-CURRENT VEHICLE POSITION-*

Range               : 1139.96 m
Azimuth             : 52.469 deg
Heading             : 48 deg
Altitude            : 1490 ft
Distance travelled  : 1139.96 m

38 min 26 sec

                    *-CURRENT VEHICLE POSITION-*

Range               : 1364.97 m
Azimuth             : 55.6633 deg
Heading             : 43 deg
Altitude            : 1505 feet
Distance travelled  : 1364.97 m

38 min 33 sec

                      **- ELECTRICAL FAILURE -**

Power supply unit failure. Voltage beyond permissible limits.

        Confirm the following:
          —Supply switched off.
          —Payload switched off.
          —Battery switched ON.

Since the nav-computer has failed, the vehicle has to be remotely controlled. It is possible
to travel a distance of 53.635 m before recovering it.
```

Result 3.1

POST-FLIGHT ANALYSIS

At time (19 21 6 18 12 1989) [sec-min-h-date-month-year] severe RPM fluctuations were observed. After ascertaining the continued presence of the above observation it was concluded that the engine has failed. Hence resorted to emergency recovery.