AN AI SITUATIONAL PILOT MODEL FOR REAL-TIME APPLICATIONS

I Y Burdun*

College of Aeronautics, Cranfield University, United Kingdom

Abstract

Automated flight control systems can create problems in flight because they do not possess a self-preservation instinct and cannot predict critical events in the manner of the human pilot. On the other hand, human pilots can make serious errors because they are unable to compute the effect of a complex combination of factors and do not function with either the accuracy or the speed of a machine. The proposed solution is to develop a real-time AI situational pilot model which combines the advantages of both sides. The purpose of the model is to identify and rectify very complex and potentially hazardous situations before they are allowed to become critical. The model's function is based on two main structures for representing knowledge regarding the flight. These are the flight situation scenario (the micro-structure), and the situational tree-network (the macro-structure of flight). This knowledge is specially constructed to describe various flight situations and possible excursions at the flight constraints under the influence of various key factors and control inputs. The results of the model development and testing are presented here. Flight scenarios and simulation results are demonstrated for FLA prototype take-off under complex conditions. An example of the situational tree-network prototype construction and application for modelling and optimisation of high-altitude hypersonic manoeuvring of an aerospace vehicle is shown. The potential areas for model application are also discussed. These applications include pilot assistance, individualised pilot-vehicle interface, automated flight envelope protection, robotic piloting, and knowledge production for complex flight domains. There may also be other real systems or virtual environments which require the use of pilot models of the anthropomorphic-mathematical type.

Introduction

The problem

Two main groups of factors shape flight safety and mission success with aircraft. These factors are the complicated features of the flight, and the shortcomings of human-vehicle interface during emergencies.

Complicated features of flight. The following features of flight contribute to the problem:

- *The chain reaction of accident development.* Transition from the normal situation to the potentially catastrophic situation with a modern vehicle often develops like a chain reaction. One event or one process may irreversibly initiate a serious incident or mission failure some 15-30 sec later.
- *Complex flight situations occur when several factors overlap.* A complex flight situation is normally the result of the interaction of several unusual factors. Every new factor further complicates and disfigures the situation, making its true character ('portrait') more difficult to identify.
- *The proximity of constraints and the rapid development of emergencies.* Broad use of the flight modes at or close to the constraints is becoming normal even in civil aviation. This may dull the vigilance of both pilots and designers. Modern control systems can increase the speed at which the vehicle approaches the constraints. This makes recognition of a critical point difficult and thereafter the situation may become irreversible.
- *Unsteadiness and variability of flight envelopes.* Flight envelopes of modern air vehicles are broadening. The shape of the flight envelope depends on a combination of various factors, as operational limits become more complex and variable. The combined effects of these factors on the flight envelope are not always clear.
- *More precise and synchronised flight control is required.* During the more demanding phases of flight and in busy ATC environments more precise performance of flight control scenarios is required. In military aviation there are growing requirements for closely synchronised behaviour between a group of co-operating vehicles.
- *4-D continuum is only a part of the flight space.* Geometric co-ordinates and time are usually considered as the only space where flight is developing. However, the success or failure of a flight in an emergency depends heavily on other, invisible co-ordinates, such as the distance to the constraints, and the alternatives and chances for following safe or unsafe flight paths. There is a shortage of this vital information in the cockpit. Nor is the human pilot able to produce it when required. In this sense, flight under the more complex conditions can be compared to driving a car at night with the headlights off.

Shortcomings of the human-vehicle interface in emergencies. Reliability of the pilot-vehicle interface,

---

* Cand.Sc. (ex-USSR); PhD Candidate.

particularly in emergencies, requires enhancement. The following points characterise the problem:

- *Avionics may fail to warn the pilot in time about the developing emergency.* Automated systems do not feel the approach to a critical situation. They may remain 'calm' and inactive when the danger is becoming evident and this delay is intolerable. Systems are generally incapable of advising the pilot of the recovery action. A little support is available in the cockpit to help the pilot estimate the nature of the future flight taking into account the key factors. As a result, the pilot remains unassisted in an emergency.
- *The techniques for avoiding critical flight situations are already well known.* The modelling of flight accidents after the event demonstrates that the techniques for recovery from many of these or similar accidents were already known to the specialists. The unfortunate point here is that this information is often not available to the pilot during a particular emergency.
- *'Who has control: the pilot or the avionics'? - this is a question without a yes/no answer.* Neither answer to this question has been sufficiently justified to be the right one. Probably, the solution does not exist in the binary domain. Certainly, cockpit automation has not brought harmony into the pilot-avionics interface.
- *Degradation of a human pilot's skills.* A pilot's skills important for coping with flight complications are subject to decay, and this process is progressing faster in the over-automated cockpit.
- *Isolated flight safety devices may generate a danger.* Automated systems are becoming more sophisticated and intelligent. However, under non-standard flight conditions the system's logic may be unclear to the pilot or be in conflict with the logic of the human pilot. In addition, isolated flight safety devices may also conflict one with another and thus create an additional source of danger to the flight. It is becoming increasingly more difficult for the pilot to limit the faulty functioning of an excessively autonomous flight control system in order to rectify the situation manually.
- *Non-protection of the aircraft flight control systems.* Even the latest aircraft are still not protected against incompetent or deliberately dangerous control inputs.

The solution

Real-time artificial intelligence (AI) and flight modelling techniques have a substantial potential to provide improved flight safety and mission success for future aerospace vehicles.

The pilot needs external intelligent support to enhance his decision making abilities in complex conditions (11, 15). Cockpit innovations are focused on providing broader information and improving visual representation of the *current* state of flight. Yet the pilot receives very little information from instruments regarding *future* flight trends. What developments will dominate the situation in, say, 15 seconds? What factors may be dangerous, and what are safe and when will they occur? What are the current chances of a catastrophic outcome (or mission failure) and under what circumstances? Neither the pilot, nor avionics can produce quantitative answers to these vital questions during flight.

There is also an emerging demand for advanced AI based systems for autonomous and remote control for unmanned vehicles ('flying robots'), etc. These systems are required to perform critical flight missions in highly complex (unknown, hostile or dynamically changing) operational environments. Robustness, learning capability and other basic human's qualities are becoming key features of such systems.

Solution implementation method

In order to help the pilot in emergencies or to copy the human pilots performance in an AI system, three questions are to be answered first: • What is the normal situation, the complex situation and the critical situation of flight? • What mechanisms does the pilot use to learn, retain and apply his experience in complex and critical situations? • What are the strongest and the weakest features of the human pilot and of a computing machine in emergencies?

1. Complex and critical flight situations. The *flight situation* is a portion of flight lasting from several seconds to, say, two minutes, depending on the vehicle type, flight mission, etc. The situation consists of a set of inter-related events and processes. These elements characterise flight control techniques, airborne systems functions, current objectives and constraints of the flight and the external factors.

In the *complex flight situation* several strong factors can overlap, and the flight is close or rapidly approaching the constraints. Situations of this type are infrequent in practice and when they occur they require mobilisation of the pilot's knowledge and skills. The *critical flight situation* is a complex situation with a potentially catastrophic result (or mission failure) if a serious control error is made or a strong negative factor is added.

In very complex situations (emergencies) the reliability of even an experienced pilot may be severely tested and may prove inadequate. Modern avionics are not reliable enough under these conditions. In flight operations, however, there have been cases when the pilot has intuitively made the correct decisions in complex, and even critical situations. The main problem is how to obtain the exhaustive list of such situations.

Critical flight situations are much more difficult to tackle. Neither the pilot, nor an automated system is able to recognise and rectify such situations in adequate time.

Reliable identification of critical situations is therefore a prerequisite for safe flight.

2. Pilot's situational decision making mechanism. A pilot's decision making mechanism includes, at least, three levels. These levels are: perceptual and motor functions to maintain the aircraft's present state, the situational (tactical) decision making functions to identify and resolve a particular situation, and the strategic planning functions to perform the general analysis of the flight and to develop/update the required control policy.

Situational decision making is of crucial importance in emergencies. These functions are based on the pilot's practical knowledge of the aircraft dynamics and its control. Situational decision making fuses the lower, reactive, and the upper, proactive, levels of the pilot's activity. The quality of these decisions determines the outcome of a particular flight mission. However, these functions are difficult to formalise, because the spectrum of possible situations is broad and their relationships are too complex.

3. Important qualities of the human pilot and of the machine. The qualities of the human pilot and the machine (computer and algorithms) which are essential in emergencies are compared in Table 1.

Critical quality (CQ_i)	Pilot	Machine
1. Self-preservation instincts	yes	no
2. Learning capability	yes	no
3. Real-time forecasting capability	yes	no
4. Capability for approximate evaluation of flight state	yes	no
5. Intuition- and insight-based action	yes	no
6. Capability to systematically cover broad operational domain	no	yes
7. Non-shadowed and error-free retention of information	no	yes
8. Absence of panic and errors, stability of response	no	yes
9. Capability for evaluation of multi-factor effects	no	yes
10. Fast response to a well detectable event	no	yes

Table 1. Qualities of the human pilot and the machine important in emergencies

The pilot's knowledge of the aircraft's flight dynamics and control is fragmentary (CQ_6) and a human's piloting skills are exposed to decay (shadowing in memory) (CQ_7). Under stress, the pilot is subject to tension, even panic; his response slows down and contains more mistakes (CQ_8). The human pilot is unable to recall the entire spectrum of situations and evaluate quantitatively the respective weight (the results of combined action) of the factors (CQ_9). Finally, the pilot is behind the machine in responding to a rapidly developing situation (CQ_{10}).

The shortcomings CQ_1, ..., CQ_5 of an automated control system in emergencies are almost the mirror image of the human pilot's advantages.

Therefore, in order to minimise the shortcomings and to combine the advantages of both sides, an AI model which integrates the human's and formal features useful in emergencies would be helpful. The prime task of such a system is to identify and avoid the complex and critical flight situations. Therefore, the model should produce its automated 'decisions' on the basis of a comprehensive knowledge of the vehicle's possible excursions in complex situations under various flight conditions. Studying the methods of how this knowledge should be organised, acquired and applied is one of the main tasks of this work.

Structure of the study

The aim and background. This paper relates to the development and testing on a computer of an applied AI model of the human pilot's situational decision making functions. This work is a continuation of the author's many years' research into modelling of the behaviour of the 'pilot - vehicle - operational conditions' system during complex flight situations.

The problem under investigation is two-fold. In the first instance, there is a lack of computational techniques to adequately model the pilot's knowledge structures and situational decision making mechanisms. In the second instance, there is a lack of the on-board intelligent support to the pilot and automated control systems in emergencies.

The subject of study. Methods for organisation, accumulation and application of the pilot's situational knowledge, and computational techniques appropriate for real-time AI pilot modelling are being studied.

The main tasks are as follows:

1. To develop concepts, data structures, algorithms and software for modelling complex flight situations (the micro-structural model of flight).
2. To develop a structure for representing a pilot's situational knowledge and a mechanism for decision making based on flight safety or mission efficiency criteria (the macro-structural model of flight).
3. To implement the micro-model on a computer and demonstrate its performance on complex and critical flight situations.
4. To implement a prototype of the macro-model (the situational tree-network) and to demonstrate its potential using modelled and hypothetical flight situations.
5. To identify areas for model application and discuss its potential for enhancement of the human pilot performance,

flight accident prevention, intelligent avionics ('intellonics') development, etc.

Techniques and tools used in this study include the following: flight dynamics, non-linear six-degree-of-freedom flight models (1,2); fourth-order prediction-correction techniques for numeric integration; theory of situational control (3), situational networks (4); general *M*-way balanced trees and other dynamic data structures (5,6,7); fuzzy sets techniques for flight state description (8,9); modified Bellman-Zadeh's technique for path optimisation using transition networks (10); computer simulation techniques; artificial intelligence (11); IBM PC, MS Windows; FORTRAN, Modula-2.

Introduction to the model

Informal definition. The *situational AI pilot model* is a system of data structures and algorithms which imitates a limited set of a human pilot's knowledge and situational decision making functions. The main feature of the model is the union of positive anthropomorphic and mathematical qualities helpful in emergencies.

Object and aim. The *object* for the model's application is the behaviour of the *'pilot - vehicle - operational conditions' system* (system) in complex and critical situations. They represent multi-factor operational conditions and flight regimes at the constraints. The overall aim is to ensure safe flight (mission success) in these situations and prevent the vehicle entering zones of irreversible (inefficient) flight paths.

The main tasks of the model are knowledge provision and functional support. The *knowledge provision* is the real-time supply of the information required to rectify the current situation. It includes flight forecast data, the distance to the nearest constraints, and the effects of the selected key factors. This information is produced automatically in the form of messages (warnings, instructions, prohibitions or explanations). It can be delivered automatically or on request.

The *functional support* is given to the standard operator (the pilot or the autopilot). It includes partial, temporary, or complete substitution of the operator if the vehicle is approaching a zone of critical (irreversible) situations. This kind of support can also be arranged on request or automatically (under agreed conditions, or when a critical situation is unavoidable).

Formal definition. The *AI situational pilot model* is a set which includes two components: the knowledge base and the decision making ('reasoning') mechanism, i.e.:

$$\mathcal{M} = \{\mathcal{KB}; \mathcal{RM}\}. \quad (1)$$

The *knowledge base* $\mathcal{KB}$ is represented by the following set of main data structures and relationships:

$$\mathcal{KB} = \{\mathbf{X}, \mathbf{U}, \mathbf{W}, \mathcal{V}, \underline{\mathbf{X}}, \underline{\mathbf{U}}, \underline{\mathbf{W}}, \Sigma^{sit}, \Sigma^{sem}, \mathbf{M}^{s}\}. \quad (2)$$

Knowledge base contents. There are three main components in the knowledge base: a tree-network of fuzzy flight situations and transitions (situational tree-network), a library of flight situation scenarios, and a limited network of semantic relationships of flight.

The *situational tree-network* (STN), Σ^{sit}, describes the behaviour of the system as a set of inter-related fuzzy flight situations and transitions. It imitates the knowledge experienced by the pilot in various flight situations. The STN structure is opened. The *flight scenarios library*, $\mathbf{M}^{s}$, is a collection of formalised plans of various flight situations composed of events and processes. The *semantic network*, Σ^{sem}, contains pseudo-physical semantic relationships of flight. This is a mixture of the cause-effect, time, space, instrumental, informational, comparison, and other relationships (3) used to describe the flight domain. The network Σ^{sem} is static. Its role is, in particular, to define the STN's structure and to plan flight scenarios.

The decision making mechanism ($\mathcal{RM}$) is a set of algorithms (mappings) which implement the automated decision making process on board the aircraft in the current situation (Fig. 1). These functions support the answers to the following groups of common-sense questions: • What is happening in the current situation and what is expected next? What are the chances of an accident (success)? • What are the current and future flight objectives? • What are the nearest flight constraints, and how close is the vehicle to them? • What control technique is to be used and which decision is the best (worst) one? • What control resources are available and what is their status and role? • Why does this situation occur? What is the reason for a particular control input? What factor is the key one? The automated decision making is considered as a multi-stage repetitive process of providing answers to these questions.

Assumptions. A comprehensive non-linear mathematical model of flight which describes complex and critical flight modes is available. Flight state parameters required for decision making can be obtained from this model or from the on-board database. A human pilot's strategic functions are not modelled (i.e. the construction and change of the control policy, learning and use of pseudo-physical relationships of flight).

Micro- and macro-structure of flight. In the model, the situational knowledge of flight is represented and processed at two levels. These are called the micro-structure and the macro-structure of flight. The *micro-structure of flight* describes a particular situation as a set of inter-related events and processes. This description is

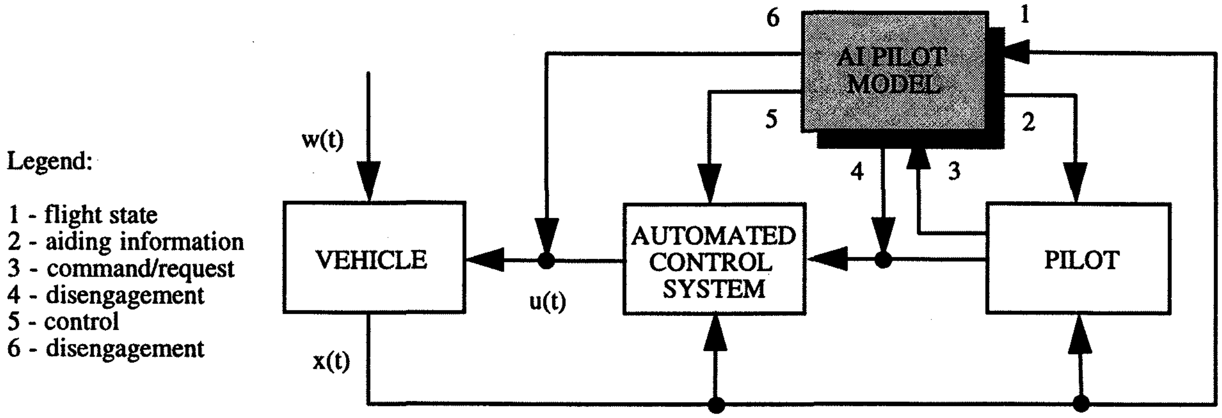
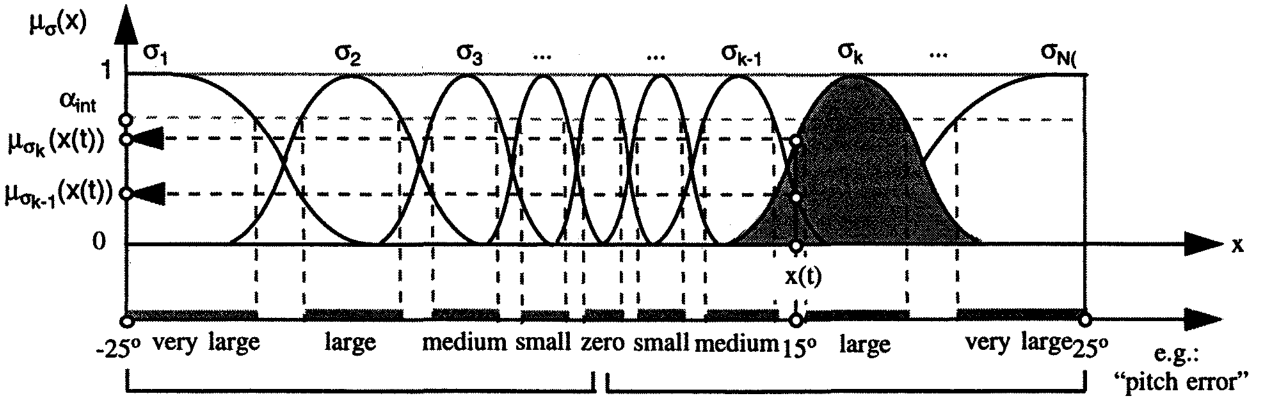
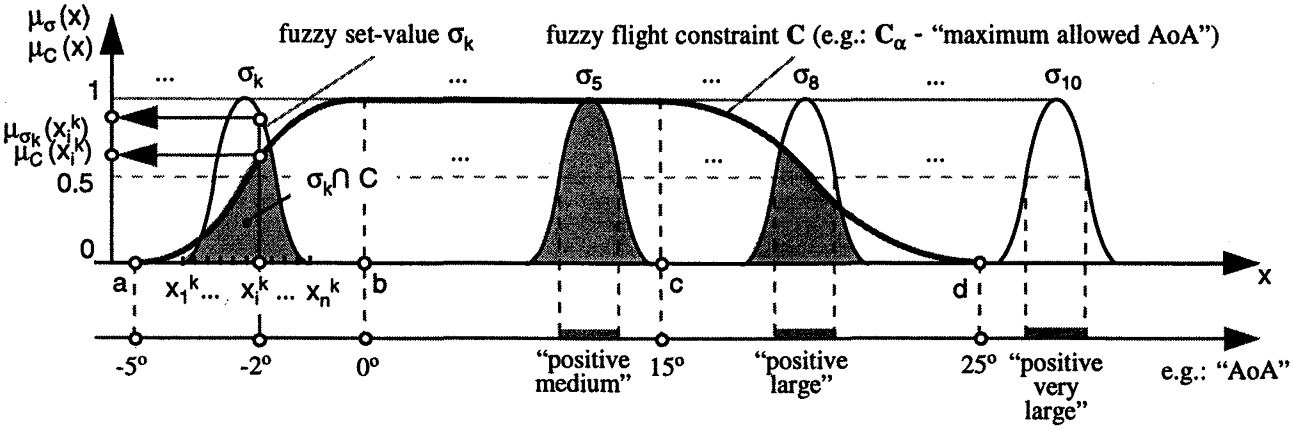


Fig. 1. AI pilot model on board the aircraft



$$(\forall x(t))\ (x(t)\in [x_{inf};\ x_{sup}])\ (\exists \sigma_k)(\sigma_k \in \underline{X})\ (\mu_{\sigma_k}(x(t)) = \max \mu_{\sigma_i}(x(t)),\ i=1,\ \ldots,\ N(\underline{X})) \Rightarrow \underline{x}(t) = \sigma_k)$$

Fig. 2. Fuzzy measurement scale and criterion for fuzzy set-value identification



$$\mu_C(\sigma_k) = \frac{\text{card}(\sigma_k \cap C)}{\text{card}(\sigma_k)} \qquad \text{card}(\sigma_k \cap C) = \sum_{i=1}^{n} \min[\mu_{\sigma_k}(x_i^k),\ \mu_C(x_i^k)], \qquad \text{card}(\sigma_k) = \sum_{i=1}^{n} \mu_{\sigma_k}(x_i^k).$$

Fig. 3. Measurement of the compatibility between a fuzzy constraint and fuzzy values of a flight variable

called the flight situation scenario. The *macro-structure* of flight is a more general store of situational knowledge. It has the form of a tree-network and describes excursions of the flight situations under various control inputs and factors. The main elements here are the fuzzy flight situation, the transition, and the branch. At this level the details of a flight situation are not considered. More important is the direction in which the flight situation is moving and its current position in relation to the constraints.

In the next two sections these two inter-related knowledge structures will be described in more detail.

Flight situation scenario (micro-structure of flight)

The following concepts are used to specify the micro-structure of flight. These are: the flight variable, the fuzzy measurement scale, the flight event, the flight process, the elementary flight situation and the flight situation scenario. The flight event and process are the main components. These concepts are introduced below with reference to the real situations.

1. Flight variable. The *flight variable* is a time-dependent parameter which describes a certain aspect of the system's state. It is also called the *model variable.*

The main classes of the flight variable are as follows: numeric, symbolic, fuzzy, linguistic; discrete and continuous; aircraft dynamics and control; airborne systems functions, and external environment. Linguistic variables take values from fuzzy measurement scales (9,10). They serve for approximate characterisation of the aircraft's states and control. The aircraft's states, and external conditions are described by numeric vectors $\boldsymbol{x}$, $\boldsymbol{u}$ and $\boldsymbol{w}$, together with their fuzzy analogues, i.e.:

$$\boldsymbol{x} = (x^1, ..., x^p), \qquad (3)$$
$$\boldsymbol{u} = (u^1, ..., u^q), \qquad (4)$$
$$\boldsymbol{w} = (w^1, ..., w^r).$$

Each model variable has its frame-specification. A list of specifications for all model variables, $\mathcal{V} = \{v^1, ..., v^k, ..., v^{N(V)}\}$, is called the *model vocabulary* $\mathcal{L}[\mathcal{V}]$, $\mathcal{L}[\mathcal{V}] = \{R[v^1], ..., R[v^k], ..., R[v^{N(\mathcal{V})}]\}$. For example, the frame $R[x^{12}]$ describing the bank angle variable, $x^{12} \equiv \gamma \equiv$ `Bank`, may have the following structure: $R[x^{12}]$ = `{12, Bank, 'bank angle', angular, body axes, degr., rad., [-60, +60], CvxPar9/1.5, RollRate, 4.0}`. That means, this angular parameter is calculated in body axes and its derivative is the variable `RollRate`. The bank variable is produced in radians but used in degrees. The variable's practical domain is [-60°; +60°]. Fuzzy values of x^{12} are measured using the fuzzy scale `CvxPar9/1.5`.

2. Fuzzy measurement scale. The *fuzzy measurement scale* (*fuzzy scale*) $\underline{X}^j$ of a state variable x^j is a mapping $\underline{X}^j$: $\mathbf{X}^j \rightarrow \underline{\mathbf{X}}^j$. It converts a numeric state value $x^j(t)$ to its approximate, or fuzzy, equivalent $\underline{x}^j(t)$. Fuzzy measurement scales in the control space are defined in a similar way, i.e. $\underline{U}^k$: $\mathbf{U}^k \rightarrow \underline{\mathbf{U}}^k$. Fig. 2 illustrates the technique which implements the mappings $\underline{X}$ and $\underline{U}^k$ developed for aerospace applications (10). The general criterion shown in Fig. 2 defines the algorithm for recognition of the current fuzzy state and fuzzy control input. The *fuzzy set index* $\boldsymbol{I}(\underline{x})$ refers to an approximate analogue (fuzzy set) $\underline{x} \equiv \sigma_k$, of a numeric state $x^i(t)$ on $\underline{X}^i$, i.e. $\boldsymbol{I}(\underline{x}^i(t)) \equiv k$.

Fuzzy scales are defined using special input specifications, for example: $R[\underline{X}]$ = `{Cvvx9/1.5, 'convex scale', 9, 1.5, [-15, +15], parabolic, 0.75, {-VL, -L, -M, -S, 0, +S, +M, +L, +VL}}`. This is a convex scale (10) with the irregularity parameter 1.5 used to measure angular errors within a [-15°; +15°] range. There are nine basic fuzzy values on $\underline{X}$ with a parabolic shape of the membership function (Zadeh's function (9)). The level of linguistic interpretation of fuzzy values is 0.75, and their names are taken from the set `{-VL, -L, ..., +L, +VL}`.

3. Flight event. The *flight event* (**E**) is a special state of the flight, a distinctive moment in the system's behaviour. It has a special meaning to the model. Events stand for noticeable changes in the flight situation (aircraft's state or control, airborne system functioning or failures, etc.). The following examples represent the flight events: $\mathbf{E}_1$: `'altitude 120 m'`, $\mathbf{E}_3$: `'left engine failure'`, $\mathbf{E}_5$: `'airspeed is low'`, $\mathbf{E}_6$: `'on the glideslope'`, $\mathbf{E}_7$: `'go-around decision'`, $\mathbf{E}_9$: `'flaps 25°'`, $\mathbf{E}_1$: `'reliable wheel-runway contact at touchdown'`, $\mathbf{E}_{11}$: `'time 00:68'`, $\mathbf{E}_{22}$: `'ailerons neutral'`, $\mathbf{E}_{17}$: `'bank ~25°'`.

A flight event becomes active in the model if its recognition criterion is met. The *event recognition criterion* has the following general form:

$$(x \nabla \mathfrak{R})_1 \; l_{12} \; (x \nabla \mathfrak{R})_2 \; l_{23} \; (x \nabla \mathfrak{R})_3, ..., \qquad (5)$$

where $x \in \mathcal{V}$ and ij $\in$ {12, 23, ...}. For example, some event $\mathbf{E}_4$: `'at circuit altitude'` can be modelled using the criterion: `(Altitude ≈ 400 m) AND (VerticalSpeed ∈ [-0.5; 0.5] m/s)`.

The following *flight event types* are useful: *independent* and *dependent* (in the latter case the event-prerequisite, or *if-event*, should be checked first); *simple* and *compound* (i.e. the number of criteria $(x \nabla \mathfrak{R})_j$ in (5)); *'precise'* and *fuzzy* (specified by the variable x in (5)); *momentarily recognisable* and *delayed*; *unique* and *periodical*; *single* and *serial* (non-zero increment of $\mathfrak{R}$). These pairs may have non-empty intersections.

Every event is provided with its input specification, for example: $R[\mathbf{E}_2]$ = `{2, speed=300, 'speed 300 km/h IAS', descent, (SpeedIAS ≈ 300 km/h AND (Acceleration ∈[-VS; +VS]),1.5,10,0}`. This event has the if-event `descent`. $\mathbf{E}_2$ is recognised using a compound (two-fold) recognition criterion and $l_{12}=\wedge$. This is also a periodic event which may appear in a situation every 10 seconds; it also has 1.5 sec delay in recognition. $\mathbf{E}_2$ is a single event ($\delta=0$).

The list of events which may occur in a situation is called the *flight events calendar* £(E). The calendar can be considered as a discrete framework to which various flight processes are attached. Logical completeness and synchronisation of a flight scenario depend on the events calendar. The delayed recognition or non-recognition of flight events can seriously distort the functioning of the automated control system or the performance of the pilot.

4. Flight process. The *flight process* (Π) is a time-history of one or several model variables which characterise a certain aspect of the system's behaviour in a particular situation. This may relate to the aircraft dynamics, pilot's control, weather conditions, airborne system functioning or failures. Every flight process has its specific purpose in the logical structure of the situation. Unlike the events, the processes are continuous components of flight.

Flight processes can be organised by their nature and purpose into the following groups: vehicle dynamics (**D**), flight control processes (**T**, **O**, **P**), airborne systems functioning and failures (**B**, **F**), external flight conditions (**A**, **R**, **W**, **Y**, ...), flight objectives (**G**) and flight constraints (**C**). Examples of system functioning and flight dynamics processes are: $\mathbf{B}_8$: `'autothrottle function'`, $\mathbf{D}_1$: `'lateral phugoid motion'`, $\mathbf{D}_3$: `'longitudinal short-period motion'`.

The main group constitutes the *flight control processes*, i.e. piloting tasks, flight 'state observers', and control procedures.

Piloting task. The *piloting task* (**T**), or the *task*, is the main manual control process. It is carried out using the vehicle's primary controls (elevator, ailerons, rudder, power controls or equivalent systems). Piloting tasks represent control with feedback. Every piloting task requires observation of the current flight state ('state observers' - see below) and flight objectives. The examples of $\mathbf{T}_j$ are as follows: $\mathbf{T}_1$: `'maintain required pitch after lift-off'`, $\mathbf{T}_4$: `'keep to the centreline during groundroll'`, $\mathbf{T}_5$: `'make co-ordinated turn at bank +15°'`, $\mathbf{T}_8$: `'maintain pitch at 10° with zero bank during climb'`.

Specification of the piloting task $\mathbf{T}_j$ includes its reference number, names, a set of controls in use, and vectors of 'state observers' and flight objectives; for example: $R[\mathbf{T}_1]$ = `{1, PitchControl, 'maintain pitch', Elevator, (PitchObs, PitchRateObs, PitchAccelObs), GoalPitch+10}`.

'State observer'. The *'state observer'* (**O**) is the process for evaluating the system's current state and comparing of this state with the one required (objective). The aim is to detect an error between these two states sufficient to change the performance of the relating piloting task. For example, the piloting task $\mathbf{T}_1$ is provided with a 'state observer' $\mathbf{O}_1$ to monitor the vehicle pitch motion. It may consist of three components (elementary 'state observers') used to control pitch angle, pitch rate and pitch acceleration: $\mathbf{O}_1$ = `(PitchObs, PitchRateObs, PitchAccelObs)`. Another example is $\mathbf{O}_4$: `'altitude monitoring in circuit flight'`.

An *elementary 'state observer'* is specified by the following frame (example): $R[\mathbf{O}^1_1]$ = `{1, o1:pitch, 'PitchObs', pitch, Sym/1.5, [-30, +30], degr., 0.5, small, very_small, 0.25}`. Here, $\mathbf{O}^1_1$ is a part of the 'state observer' vector `o1:pitch` which monitors `pitch` variable. A fuzzy scale `Sym/1.5` is used to measure observation errors within the range [-30°; +30°]. The error feedback gain is 0.5 (12). The process for pitch angle observation is triggered on (off) when the error is `small` (`very_small`). The observation time increment is 0.25 sec.

Control procedure. The use of secondary controls (flaps, spoilers, etc.), as well as single movements with the primary controls, are described by the process type called *control procedure* (**P**). For example, $\mathbf{P}_1$: `'wheels - up'`, $\mathbf{P}_2$: `'unstick'`, $\mathbf{P}_3$: `'flap 30°→15°'`, $\mathbf{P}_6$: `'engines - to MCPR'`. Control procedures are specified using the following frame (example): $R[\mathbf{P}_2]$ = `{2, p2:unstick, 'move stick to rotate', elevator, g4:elev(-8), rel, 0.5}`. This particular frame describes an unstick procedure at V_R. It is modelled as a moderate (`0.5` of the normal rate) change of the variable `elevator` by -8° (the objective is `g4:elev(-8)`) from its current position (`rel`).

Failure. An *airborne system's failure* is a process which imitates abnormal function of an on-board system. The examples are: $\mathbf{F}_2$: `'left engine failure'`, $\mathbf{F}_8$: `'uncommanded deployment of thrust-reverser'`, $\mathbf{F}_{27}$: `'elevator jammed in 7.5°'`. In the model, failures are formally described as artificial control procedures.

Flight objective. The *objective-process* (**G**) defines some target value(s) of the aircraft's state or control variables as a function of time. The objectives are used in manual control processes, for example: $\mathbf{G}_8$: `'10° pitch during initial climb'`, $\mathbf{G}_4$: `'400 m altitude'`, $\mathbf{G}_{12}$: `'zero sideslip and bank in en-route'`. For example, the frame $R\ [\mathbf{G}_7]$ = `{7,` `g7:MCPR, 'maximum continuous PR', throttles, (85, 85, 85, 85), %, abs}` specifies the demanded en-route position of the aircraft power levers (85%). The attribute `abs` stands for the absolute goal value and `throttles = (thr_1, ..., thr_4)`.

Flight constraint. The purpose of the *flight constraint* process type (**C**) is to uniformly describe various limitations which should be taken into account in situational decision making. Together with the flight objective, the flight constraint is another artificial process which is important for the construction and analysis of the situational tree-network. Examples of flight constraints are as follows: $\mathbf{C}_2$: `"bank = 30°"`, $\mathbf{C}_1$: `"obstacle elevation"`, $\mathbf{C}_3$: "V_z = -5m/s", $\mathbf{C}_5$: `"aileron limit"`.

The technique for measuring the compatibility between a fuzzy state and a fuzzy constraint is illustrated in Fig. 3. The characteristic points of the fuzzy constraint **C** define the intervals of 'fuzzification' of the precisely specified constraints. The expression for calculation of the compatibility measure, $\mu_C(\sigma_k)$, between the fuzzy value σ_k of x, $\sigma_k \equiv \underline{x}(t)$, and the fuzzy constraint **C** is also shown.

An example of constraint specification is as follows: $R[\mathbf{C}_7]$ = `{7, c7:AoA, 'critical AoA', AoA, LE, (max_AoA.tof flap), degr., 0.8}`. This structure defines the maximum permitted (`LE` is the relation) angle of attack at take-off. $\mathbf{C}_6$ is modelled as a fuzzy set and derived from a file-table `max_AoA.tof`. It contains the `AoA` variable in degrees as a function of flap setting. The minimum acceptable compatibility between the variable and the constraint is 0.8.

Wind. The *wind process* type (**W**) is used to define various wind effects upon the vehicle dynamics. It belongs to the group of environmental processes. The examples are: $\mathbf{W}_1$: `'strong crosswind of 15 m/s'`; $\mathbf{W}_5$: `'3-D microburst with the intensity of 4.5 m per 30 m of H'`. An example of wind specification is as follows: $R[\mathbf{W}_5]$ = `{1, w1:microburst, 'Accident of dd.mm.yy', (horiz_wind, vert_wind, cross_wind), m/s, ddmmyy.dat altitude m, abs}`. It specifies three wind components in m/s, identified from an accident (the flight recorder data table `ddmmyy.dat`), as a function of the variable `altitude` in meters.

Other types of environmental processes. Atmospheric conditions (**A**), rain (**R**), and runway surface condition (**Y**) external process types have also been developed in the model. The examples are as follows: $\mathbf{A}_1$: `'pressure +15 mm Hg and temperature +20°'`, $\mathbf{R}_{11}$: `'heavy rain, intensity 500 mm/h'`; $\mathbf{Y}_3$: `'wet runway, coefficient of friction 0.3'`.

5. Elementary flight situation. This concept is introduced to describe the logic of a flight situation. The *elementary flight situation* specifies the primary cause-effect link between two events and a process, or a set of homogeneous processes (Fig. 4), i.e.:

$$\mathbf{S} = (\mathbf{E}_i, \mathbf{E}_k, \{\Pi_1, \ldots, \Pi_{N(\Pi)}\}). \tag{6}$$

The elementary situation **S** begins at the source event $\mathbf{E}_i$ and ends at its target event $\mathbf{E}_k$. It incorporates a set of homogeneous processes Π_j running between $\mathbf{E}_i$ and $\mathbf{E}_k$. The event $\mathbf{E}_i$ ($\mathbf{E}_k$) is called the *opening (closing) event* as it triggers the process Π_j on (off). An example of frame-specification for an elementary situation $\mathbf{S}_7$ is as follows: $R[\mathbf{S}_7]$ = `{7, obs_in_climb, 'observers in climb', airborne, altitude=400, (o1:pitch, o2:bank, o3:sideslip, o4:vert_speed, o5:altitude), observer, 0}`. This structure defines an elementary situation with the 'state observers' used for the airborne phase of take-off. The event `airborne` (after lift-off) opens the situation and the event `altitude=400` (400 m) closes it. The five 'observers' referred to are modelled 0.5 sec after the source event has been recognised.

The criterion for controlling flight processes is as follows:

$$\begin{aligned}&(\forall \mathbf{S})(\mathbf{S} = (\mathbf{E}_i, \mathbf{E}_k, \Pi_j)\ (((\mathbf{E}_i \in \Omega^R(\mathbf{E}) \wedge \mathbf{E}_k \notin \Omega^R(\mathbf{E})\\ &\wedge \Pi_j \notin \Omega^{CL}(\Pi)) \wedge (t \geq t\ [\mathbf{E}_i \in \Omega^R(\mathbf{E})] + \tau))\\ &\Rightarrow \Pi_j \in \Omega^A(\Pi)) \vee ((\mathbf{E}_k \in \Omega^R(\mathbf{E}) \Rightarrow \Pi_j \in \Omega^{CL}(\Pi)).\end{aligned} \tag{7}$$

Together with the algorithms which implement flight events and flight process types, the relationship (7) defines the main algorithm of the micro-model.

6. Flight situation scenario. The scenario is a compact and uniform specification of the flight situation. Its purpose is to plan the content and the logic of a particular situation. The scenario specifies the flight events and processes which are expected (or unexpected) to occur in the situation, and the conditions (elementary situations) for linking them together. The *flight situation scenario* $\mathcal{S}$ may be therefore defined as a set of the following lists of specifications: flight events, flight processes and elementary situations, i.e.:

$$\mathcal{S} = \{\mathcal{L}(\mathbf{E}); \mathcal{L}(\Pi); \mathcal{L}(\mathbf{S})\}, \tag{8}$$

where

$$\mathcal{L}(\Pi) = \mathcal{L}(\mathbf{D}) \cup \mathcal{L}(\mathbf{CON}) \cup \mathcal{L}(\mathbf{SYS}) \cup \mathcal{L}(\mathbf{ENV}), \tag{9}$$

$$\mathcal{L}(\mathbf{CON}) = \mathcal{L}(\mathbf{T}) \cup \mathcal{L}(\mathbf{O}) \cup \mathcal{L}(\mathbf{P}) \cup \mathcal{L}(\mathbf{G}) \cup \mathcal{L}(\mathbf{C}), \tag{10}$$

$$\mathcal{L}(\mathbf{SYS}) = \mathcal{L}(\mathbf{B}) \cup \mathcal{L}(\mathbf{F}), \tag{11}$$

$$\mathcal{L}(\mathbf{ENV}) = \mathcal{L}(\mathbf{W}) \cup \mathcal{L}(\mathbf{R}) \cup \mathcal{L}(\mathbf{A}) \cup \mathcal{L}(\mathbf{Y}) \cup \ldots . \tag{12}$$

The *flight control scenario* $\mathcal{S}_U$ is defined as a subset of $\mathcal{S}$, $\mathcal{S}_U \subset \mathcal{S}$, i.e.:

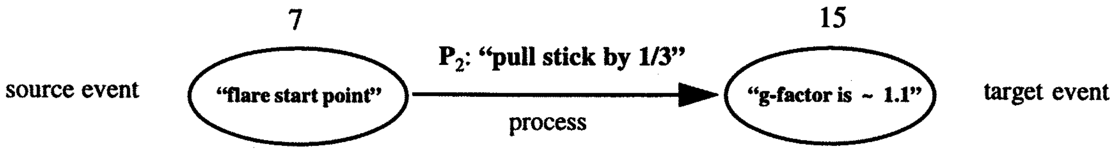


Fig. 4. Elementary flight situation

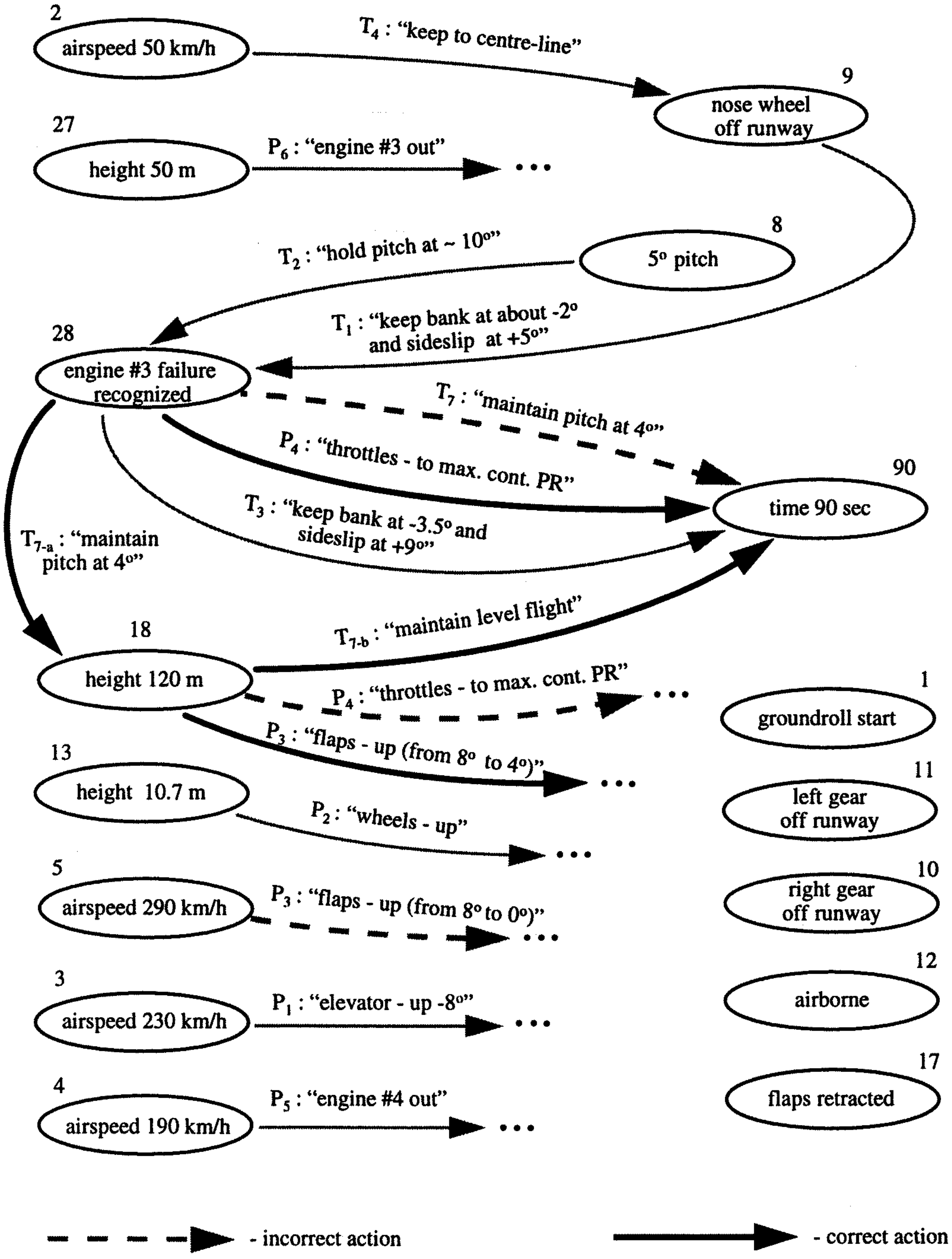


Fig. 5. Scenario $\mathcal{S}_2$: "FLA prototype F-93A takeoff with two right-hand engines out (incorrect and correct piloting)"

$$\mathcal{S}_U = \{\mathcal{L}(\mathbf{E}); \mathcal{L}(\mathbf{T}); \mathcal{L}(\mathbf{O}); \mathcal{L}(\mathbf{P}); \mathcal{L}(\mathbf{G}); \mathcal{L}(\mathbf{C}); \mathcal{L}'(\mathbf{S})\}, \quad (13)$$

where $\mathcal{L}'(\mathbf{S}) \subset \mathcal{L}(\mathbf{S})$ with $\Pi \in \{\mathbf{T}, \mathbf{O}, \mathbf{P}, \mathbf{G}, \mathbf{C}\}$.
Directed graphs are a convenient form for flight scenarios representation. Fig. 5 depicts a realistic scenario of the complex flight situation scenario $\mathcal{S}_2$: `'FLA prototype F-93A take-off with two right-hand engines failure'`. The vertices (ellipses) stand for the events from $\mathcal{L}(\mathbf{E})$ and the directed arcs depict the processes from $\mathcal{L}(\Pi)$. The process-arrow begins at its opening event-vertex and normally ends at the closing event-vertex. The flight situation scenario can be specified by the following general frame:

$$\begin{aligned} R[\mathcal{S}] = \{&\mathrm{i}, \mathrm{N}, (et_1, \mathcal{L}(et_1), (\mathrm{j}_{1(1)}, \ldots, \mathrm{j}_{n(1)})), \\ &\ldots, (et_k, \mathcal{L}(et_k), (\mathrm{j}_{1(k)}), \ldots, \mathrm{j}_{n(k)})), \ldots, \\ &(et_{N(\mathcal{S})}, \mathcal{L}(et_{N(\mathcal{S})}), (\mathrm{j}_{1(N(\mathcal{S}))}, \ldots, \mathrm{j}_{n(N(\mathcal{S}))}))\}. \end{aligned} \quad (14)$$

The structure (14) is flexible. For example, in order to derive from $\mathcal{S}_2$ a modified scenario $\mathcal{S}'_2$ (see Fig. 5) without the retraction of wheels and flaps (i.e. the procedures $\mathbf{P}_2$ and $\mathbf{P}_3$ are not performed), the following description may be used: $R\ [\mathcal{S}'_2] = \{2,$ `'take-off without retraction of wheels and flaps'`, $(E, \mathcal{L}(\mathbf{E}))$, $(T, \mathcal{L}(\mathbf{T}))$, $(P, \mathcal{L}(\mathbf{P}), (-\mathbf{P}_2, -\mathbf{P}_3))$, ...}, where the lists $\mathcal{L}(\Pi)$ correspond to $\mathcal{S}_2$.

Advantages of the flight scenario technique. Planning of a complex flight situation for both manual and automatic control modelling requires dynamic linking and synchronisation of various events and processes which may have hidden inter-relationships. The flight scenario concept helps perform this task more reliably.

Simulation practice demonstrates that the majority of practical flight situations can be formalised by clear structures containing only two basic element types - events and processes (the lists (8)-(14)). This technique allows the separation of the details (i.e. the content and logic) of a particular situation from the scenario processing algorithm (7). Therefore, the control scenarios can be stored, modified and loaded *independently* from the flight control or flight modelling software. Using this technique, the following structural and parametric modifications to a scenario can be made as well: • set operations over scenarios or scenario components; • shifting and 'freezing' (i.e. making temporarily inactive) some events and processes along the time scale; • changing the parameters of events and processes without changing the scenario structure, etc.

The flight scenario concept helps better understanding of the flight control logic under complex conditions. Various real and hypothetical complex flight cases can be modelled (and repeated) in detail without the pilot in the loop, using the control scenario (13). The events-processes descriptive language is close to practical lexicons of flight experts. It can be used to formalise Flight Manuals, flight incidents and their variations, flight test and pilot training programmes, etc.

Situational tree-network (macro-structure of flight)

Real flight domain. Real flight is far from matching the ideal scenarios. How does one account for possible variations of flight control techniques, pilot errors, deterioration of weather conditions, or systems malfunctioning? What happens if, for instance, the pilot has delayed recognition of the critical engine failure or forgets to conduct some important control procedure? What are the consequences of non-recognition of the event `'aircraft weight on wheels'` for the success of landing on a wet runway? Which constraint is the nearest one and what is the distance to it under specific conditions? The lack of systematic answers to these and similar questions on the flightdeck is the fundamental reason for many incidents.

To account for this sort of flight complication, in addition to the micro-structural model (flight scenario), the situational tree-network (STN) of flight is introduced. It represents the upper level of the model's knowledge of situations called the *macro-structure of flight*. In brief, STN is a collection of inter-related fuzzy flight situations and transitions which develop under various control inputs and factors. Its purpose is to describe possible deviations from a standard flight scenario depending on various factors. The main requirement to STN is that these excursions are to be stored in a compact (economical), systematic (specially designed) and approximate (fuzzy) form and accessible in real time.

STN implementation technique. General M-way balanced fuzzy situational trees and linked lists meet this condition. General fuzzy situational trees for aerospace applications have been introduced and tested in (4, 13). Algorithms for construction and processing M-way balanced trees and associated data structures are derived from (5-7). Special techniques have also been developed to modify flight scenarios to plant STN. The construction, analysis and application of a fuzzy STN prototype for an aeroplane have been studied in (10) based on the modified Bellman-Zadeh's dynamic programming method.

Factor → non-standard flightpath → critical situation. Formally, the *factor* (Φ) is a new, missed, or modified process or event which may occur during actual flight. The factor is characterised by its type and the power (level) of influence it has upon the flight control scenario and flight quality (safety). The variety of factors and their combinations make systematic examination of complex flight situations difficult. It is almost impossible to specify the relationships between these factors in the Flight Manual or in avionics system's logic. Nor can they be remembered by the human pilot.

STN is designed to make this knowledge accessible to the pilot/autopilot in emergencies. Modelling of non-standard flightpaths under various conditions (factors) creates STN branching in the form of arranged excursions of the vehicle (flight model) towards the constraints and back to return within the flight envelope. The goal is to construct STN which threads these boundary flight modes. This will help identify zones of critical situations which, in fact, define the flight envelope under multi-factor conditions.

Factor types. The flight factors are modelled as changes in flight scenario specifications (8)-(14). There are four *factor types* in the model: structural or parametric changes of the flight events calendar, manual control errors and variations, demanding external conditions of flight, and failures of airborne systems. The examples are as follows: Φ_1: `'thrust reverser deployment in flight'`, Φ_4: `'go-around decision'`, Φ_7: `'large pitch at take-off'`, Φ_3: `'non-observation of sideslip'`, Φ_{11}: `'0°C...+15°C deviations in the air temperature'`, Φ_8: `0°...15° flap asymmetry'`.

Factor level. The *characteristic level (power) of a factor* Φ_j, $\Phi_j\,[\Lambda]$, is the deviation of the associated flight event or flight process from the standard. These levels are described and measured using fuzzy scales. Examples are as follows: Φ_1`[-VL] - 'very strong windshear'` (~ -4.5 m/s per 30 m of altitude), Φ_3 `[+S] - 'small error in maintaining optimum pitch'`.

Factor specification. The *factor of flight* (Φ_j), or flight factor, can therefore be defined as a set of deviations in value of the model variable, which represents the factor, in relation to some standard level. An example of factor specification is as follows: $R[\Phi_2]$ = `{2, 'pitch errors in take-off', manual, objective, pitch, [-10°; +15°], NormPar7}`. This frame describes Φ_2 as a series of manual control errors or variations which result in a series of the goal pitch angles after lift-off (`objective` is the affected process type). These errors are examined in relation to the optimum pitch level (e.g.: 12°) within the range [-10°; +15°]. That means, a series of flightpaths is modelled with the goal pitch angles from [+2°; +27°]. To interpret and measure this factor, a regular fuzzy scale `NormPar7` with seven basic fuzzy values is used.

Main concepts of STN. Three main concepts describe STN, these are: the fuzzy flight situation, the transition between two situations (fuzzy transition), and the branch.

1. Fuzzy flight situation. The fuzzy flight situation is the formalisation of some imprecise image (or a 'snapshot') of the current flight situation reflected in the pilot's consciousness. In addition to the aircraft's state and flight control, it is characterised by the sets of important events and processes. The *fuzzy flight situation*, **S**(*t*), is defined as follows:

$$\mathbf{S}(t) = \{\underline{\boldsymbol{x}}(t), \underline{\boldsymbol{u}}(t), \boldsymbol{x}(t), \boldsymbol{u}(t), \mathcal{L}(\mathbf{E}), \mathcal{L}(\Pi), \mathcal{L}(\mathrm{Q})\}. \qquad (15)$$

Frame-specifications (15) are generated automatically for all **S**(*t*) during STN construction.

2. Fuzzy transition. The *fuzzy transition*, **T**(*t*; *t*+Δ), is an approximate discrete representation of the system evolution from *t* to *t*+Δ. Its purpose is to specify various dependencies and differences between two neighbouring fuzzy flight situations in STN (i.e. cause and effect, time, technological (control), and others). The transition is the result of the processes which run on Δ; it begins in the initial situation **S**(*t*) and ends in the final situation **S**(*t*+Δ). In general, the transition can be defined as follows:

$$\mathbf{T}(t;t+\Delta) = \{\mathbf{S}(t), \mathbf{S}(t+\Delta), \mathcal{L}(\Pi(t)), \mathcal{L}^{\mathrm{N}}(\Pi), \Delta, \mathcal{L}(\mathrm{Q})\}, \qquad (16)$$

where $\mathcal{L}^{\mathrm{N}}(\Pi) = \mathcal{L}(\Pi(t+\Delta))/\mathcal{L}(\Pi(t))$ is the difference between the sets of flight processes running in **S**(*t*+Δ) and **S**(*t*). Special frame-specifications are used to retain data on fuzzy transitions in STN.

3. Branch. The *branch* **B** is a chain of several inter-related fuzzy situations and transitions which occur according to some flight scenario. In other words, this is a special structural representation of a flightpath from t_0 to t_N, i.e.:

$$\mathbf{B}(t_0; t_N) = \{\mathbf{S}(t_0), \mathbf{T}(t_0; t_0+\Delta_1), \mathbf{S}(t_1), \ldots, \mathbf{S}(t_{N-1}),$$
$$\mathbf{T}(t_{N-1}+\mathrm{k}\Delta_{N-1}; t+\mathrm{k}\Delta_N), \mathbf{S}(t_N)\}. \qquad (17)$$

Note, that $\Delta_1 + \ldots + \Delta_N$ is the total length of $\mathbf{B}(t_0; t_N)$ by time. The identification code of a branch within STN includes its order (i), parent's branch number (j), and its number (k) on the parent's branch, i.e.: $\mathbf{B}_{\mathrm{k}}^{\mathrm{i(j)}}$.

Formal definition of STN. The *situational tree-network of flight* (phase of flight) is a directed graph Σ^{sit}:

$$\Sigma^{\mathrm{sit}} = \{\Omega(\mathbf{S}), \Omega(\mathbf{T})\}. \qquad (18)$$

This is an ordered set of fuzzy flight situations (nodes, or vertices) and transitions (arcs) between them. Σ^{sit} exemplifies possible paths, or excursions, in the system's behaviour at the constraints under the action of various factors. The examined flight factors are called *anticipated flight conditions*. The STN construction technique is based on the systematic exploration of the effects of a combination of the most probable factors and their levels.

STN components (introduction). STN has the following main components (Fig. 6): the root situation, the main branch, the leaf situation, the derivative branch, the top, flight objectives and flight constraints.

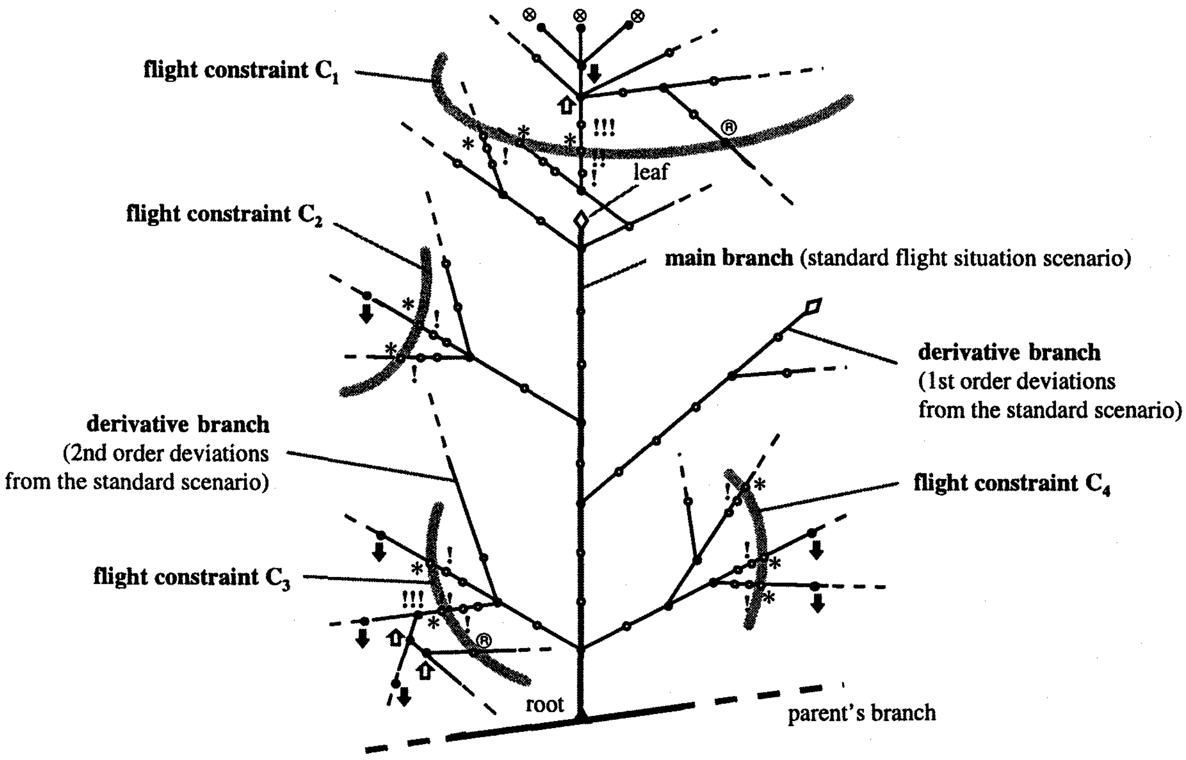


Fig. 6. Situational tree-network and its main components

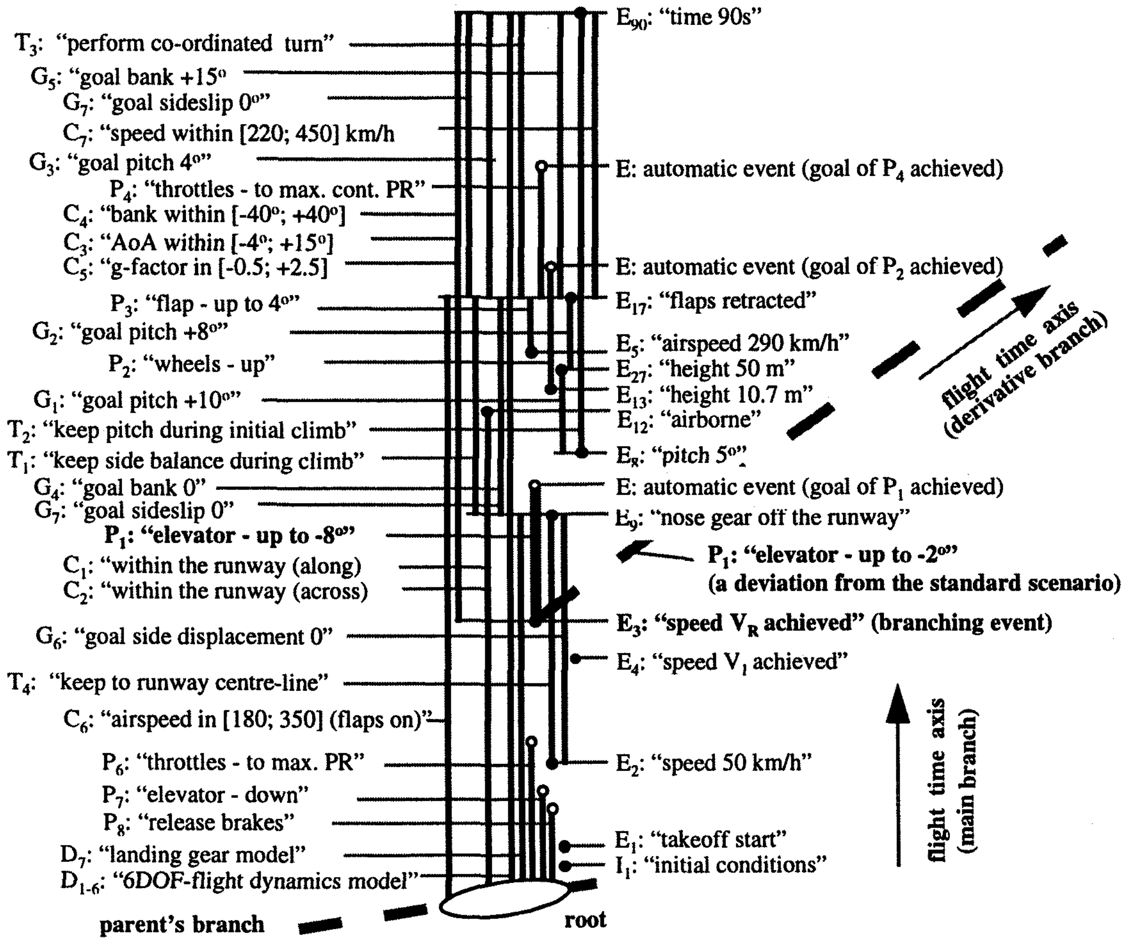


Fig. 7. Events-processes structure of the main branch of a situational tree-network (standard takeoff scenario)

The *root situation* (root) is the initial (number one) fuzzy situation of some phase of flight. By means of the root, STN, or its subtree, is linked to the previous phase of flight (parent's branch or subtree), i.e. STN grows from the root. The *main branch* (trunk) is a chain of situations and transitions developing under the standard scenario (ref. definition). Its end (the *leaf situation*), which is opposite to the root, stands for an objective situation to be achieved during the specific phase of flight. The *first-order derivative (secondary) branch* is the result of implementation of a non-standard scenario (i.e. as the effect of some factor(s)). It begins at the point (situation) where the change in the main scenario has occurred. The derivative branch ends if the normal scenario is restored, or some flight constraint has been infringed.

Higher-order derivative branches shape the STN's *top* (crown). They stand for more complex and rare combinations (overlapping) of flight factors. The STN's top is designed to cover the zones of flight constraints infringement including the irreversible situations marked by the leaf-type situations. These branches also represent the recovery paths to restore safe flight.

Relationship between the micro- and the macro-structure of flight. The knowledge of the micro-structure of flight (flight scenario) is used to construct compact and comprehensive STNs. The take-off scenario which is very similar to Fig. 5 is shown in Fig. 7 in the form of the STN main branch (trunk). This transformation is made if the flight events are ordered along the time axis and the associated processes are lined up in the upward direction. A first-order derivative branch is shown here by the thick dashed line (the branch 'planted' in a bud situation with the event $\mathbf{E}_3$: `'speed` $\mathrm{V_R}$ `achieved'`. This branch is the result of the action of a new factor - the non-standard procedure $\mathbf{P}_1$: `'elevator - up to -2°'`, which diverts the remaining part of the flight scenario. Therefore, given the main scenario and the strategy for planting derivative branches, STN with the required properties can be constructed.

Virtual flight space. The *virtual flight space* (**VFS**) is an abstract 2-, 3-, or 4-D dynamic image which depicts a subtree, or a slice from STN. This part of STN is projected onto the colour display (windscreen, HUD, FMS display, etc.). **VFS** is considered as the basis for implementation of the model-based intelligent pilot-vehicle interface. In particular, this picture can be used for examination of the future flightpaths and for dynamic synthesis of the flight control policy.

STN genotype. The STN *genotype* $\mathcal{G}$ is a set of semantic pseudo-physical relationships which specify the structure of STN (branch, or subtree). In general, the following parameters determine this structure: the main flight scenario, the length of a fuzzy transition, the examined flight factors and their levels, together with the rule(s) used to combine these factors and levels, i.e.:

$$\mathcal{G} = \{\mathcal{S}, \Delta, \{\Phi_1, ..., \Phi_N\}, \{\Phi_1[\Lambda], ..., \Phi_N[\Lambda]\}, \mathcal{R}(\Phi_1, ..., \Phi_N)\}. \tag{19}$$

STNs and natural trees. The model performance largely depends on the shape and quality (density) of its STN. The STN's structure is also determined by the model purpose. Nature provides a perfect clue to the structure of artificial STNs. Many properties and techniques useful for the construction and analysis of STNs can be learned from it. In particular, the following patterns of natural trees may be helpful when designing STN for specific applications: flight envelope protection (Fig. 8), guided missile control (Fig. 9), dynamic exploration of the unknown flight conditions (Fig. 10), basic pilot training/anti-missile control (Fig. 11).

Fuzzy objectives and constraints. The precise objectives and constraints are, in fact, numeric approximations of actual goals and limitations to flight. The aircraft's state and control inputs located just beyond the precisely specified limits are only slightly less acceptable than elements which are close to them, but fit the allowed range. This inherent feature of real flight objectives and constraints is described by fuzzy sets (9, 10).

Flight constraints within STN. The flight constraints look like external objects attached to STN branches (see Fig. 6). They separate safe (efficient) and dangerous (prohibited, unsuccessful) flight situations. Flight constraints are used in STN to measure distance between situations and constraints, to assign safety levels to flight situation, to evaluate the probability of safe and unsafe developments of future flight, and to specify safety margins for decision making. It is important that the position of the flight constraints within STN can be identified during its construction. The situations located on the branches before the constraints (within the flight envelope) are then provided with the distances to the nearest constraint.

Constraints and critical situations. In general, two fundamental causes create the chain reaction development of flight incidents. The first cause is when the pilot (or automated system) fails to identify in time the closest constraint(s) which the vehicle is approaching. This is called the *critical constraint* $\mathbf{C}_{\mathrm{crit}}$, and it may be subject to change during these excursions. The second cause, which normally follows the first one, is that the vehicle may enter the zone of situations with a dominating proportion of unsafe outcomes, whatever the factors and control inputs follow. These situations are called *critical situations* $\mathbf{S}_{\mathrm{crit}}$.

Therefore, the prime task of the model should be: • to advise the pilot (avionics) of the constraints which are currently critical to the flight and of the expected zone of critical situations (*knowledge provision*), and • to prevent

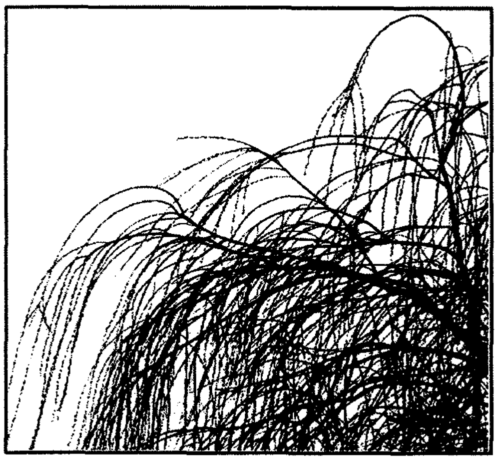


Fig. 8. Pattern of STN for flight envelope protection

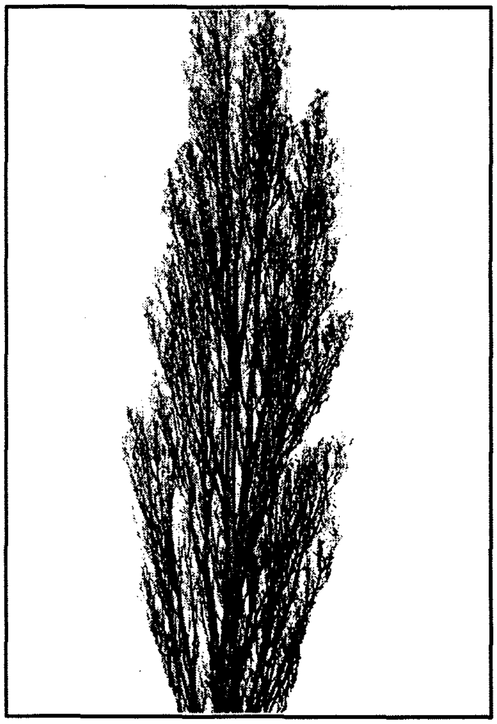


Fig. 9. Pattern of STN for guided missile control

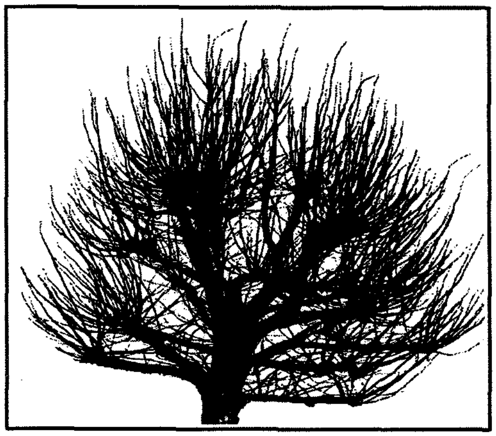


Fig. 10. Pattern of dynamically constructed STN

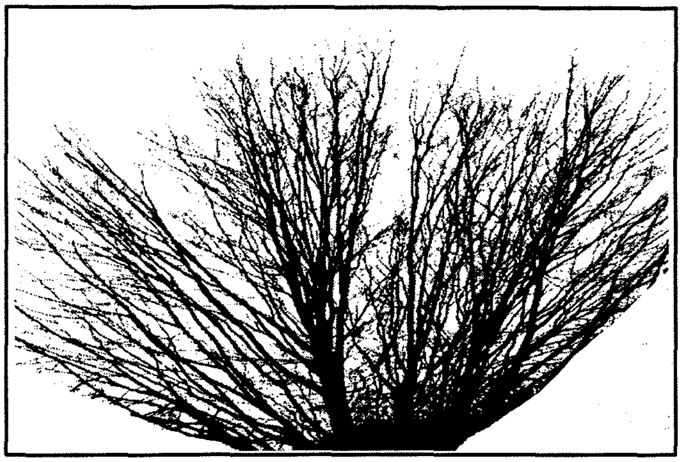
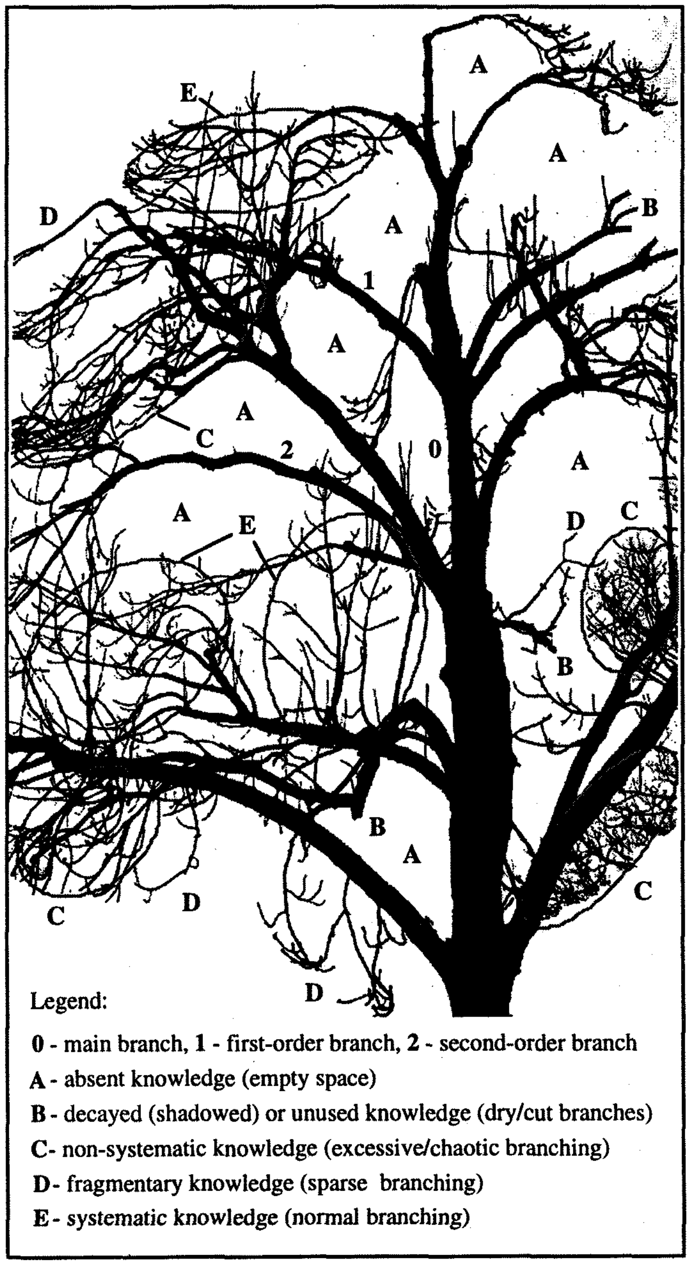


Fig. 11. Pattern of STN for basic pilot training/anti-missile control



Legend:

0 - main branch, **1** - first-order branch, **2** - second-order branch
A - absent knowledge (empty space)
B - decayed (shadowed) or unused knowledge (dry/cut branches)
C- non-systematic knowledge (excessive/chaotic branching)
D- fragmentary knowledge (sparse branching)
E - systematic knowledge (normal branching)

Fig. 12. Pattern of 'internal STN' of the human pilot

the vehicle from entering such a zone (*functional support*). In the first case, the time margin for making the pilot's decision and the instructions for manual recovery are also produced.

STN and Flight Manual. STN first-order branches describe main deviations from the standard scenario, i.e. they represent the *main factors of flight* which are the most important to the pilot. The Flight Manual (and the automated flight control systems' logic) normally accounts for the most probable of these factors. The Manual may also cover the scenarios which represent some of the second-order branches, i.e. several 'typical' multi-factor situations. However, due to the multiplicity of these factors in combination, the main burden of retaining this specific knowledge rests with the human pilot.

STN and pilot skills. Given an emergency, the pilot may fail to estimate the effects of various control inputs and overlapping factors in time. The hypothesis is as follows. The pilot's skills to cope with emergencies are mainly stored in the second- and higher-order derivative branches of his 'internal STN'. For various reasons, these flight complexities may have not been explored (memorised) as well as for the main scenario or its first derivatives. Hence, this knowledge is subject to decay or damage. Given a significantly deteriorated flight condition and stress, the pilot is unable to produce a decision of the required quality. The deeper and faster the pilot attempts to refer to his 'internal STN', the more probable it is that he will pick up the wrong branch (if one is identified at all) from it. A pattern of the natural tree which may help explain these deficiencies in the human pilot is shown in Fig. 12.

Therefore, the purpose of the model is to take over the routine of systematic coverage and non-decaying retention of situational knowledge. Having provided the cockpit with a sufficiently comprehensive artificial STN, the pilot will probably not need to trace rapidly, under conditions of stress, his own 'STN'. The pilot's potential can be applied at a more creative level leaving these resource-consuming functions to the model. The pilot could help the model focus *its* search within STN applying his strategic ('deep') knowledge. He may also select the desired flightpath from the **VFS** picture for automated realisation.

Complexity and safety measures of the flight situation. In flight safety applications, the model must be capable of measuring the complexity and safety of a flight situation. This can be done through examination of the STN topology. The following parameters are useful here: the number of factors, distance to the constraints, gradient of the situation excursion towards the constraints, consumption of control resources, etc. From the point of view of flight safety the graph Σ^{sit} (its vertex-set $\Omega(\mathbf{S})$) can be divided into two parts:

$$\Sigma^{\mathrm{sit}} = \Omega^{+}(\mathbf{S}) \cup \Omega^{-}(\mathbf{S}), \tag{20}$$

To assign the general safety status to a situation (i.e. safe or unsafe), the following criterion is proposed:

$$\begin{aligned}&(\forall \mathbf{S}, \underline{\boldsymbol{x}})\ (\mathbf{S}\in\Sigma^{\mathrm{sit}} \wedge \underline{\boldsymbol{x}} = (\underline{x}^{1}, \ldots, \underline{x}^{k}, \ldots, \underline{x}^{p}) \wedge \underline{\boldsymbol{x}}\in\mathbf{S})\\ &(\exists k)\ (k\in\{1, \ldots, p\})\ ((\exists\mathbf{C})\ (\mathbf{C}\in\{\mathbf{C}_1, \ldots, \mathbf{C}_{N(\mathbf{C})}\})\\ &(\mu_{\mathrm{C}}(\underline{x}^{k}(t)\leq\alpha_{\mathrm{C}}) \Rightarrow (\mathbf{S}\in\Omega^{-}(\mathbf{S}))\vee((\forall\mathbf{C})\\ &(\mathbf{C}\in\{\mathbf{C}_1, \ldots, \mathbf{C}_{N(\mathbf{C})}\})\ (\mu_{\mathrm{C}}(\underline{x}^{k}(t)>\alpha_{\mathrm{C}}) \Rightarrow (\mathbf{S}\in\Omega^{+}(\mathbf{S}))).\end{aligned} \tag{21}$$

In (21), the measure of the compatibility $\mu_c\ (\underline{x}^k(\mathrm{t})$ between the fuzzy constraint **C** and the aircraft's fuzzy state $\underline{\boldsymbol{x}}(t)$ is used (see Fig. 3). More detailed gradation of safety levels requires further analysis of flight situation types.

Main situation types. In STN, all situations can be categorised by their genetic role and safety status. These properties of situations are used for the construction, analysis and application of STN. The following *genetic types* and *safety status types* of flight situations are defined (Fig. 13; see also Fig. 6):

$$\{\mathbf{S}_0(\mathbf{S});\ \mathrm{S}_{\bullet};\ \mathbf{S}_{\mathrm{G}}(\mathbf{G});\ \mathbf{S}_{\lozenge};\ \mathbf{S}_{\Delta}\}, \tag{22}$$

$$\{\mathbf{S}_{\mathbf{OK}};\ \mathbf{S}_{\mathbf{M}};\ \mathbf{S}_{!},\ \mathbf{S}_{!!},\ \mathbf{S}_{!!!};\ \mathrm{S}_{*};\ \mathbf{S}_{\Uparrow};\ \mathbf{S}_{®};\ \mathbf{S}_{\downarrow};\ \mathbf{S}_{\otimes}\}. \tag{23}$$

The *ordinary situations* ($\mathbf{S}_0$ or **S**) are used as reference points for the recognition of real flight situations. That situation $\mathbf{S}_0$ which matches the real one is identified as the current situation. The ordinary situations constitute the majority in STN. The *'bud'-situation* ($\mathbf{S}_{\bullet}$) type may be used to implant a new derivative branch at any time during STN life. The information stored in the frame-specification for $\mathbf{S}_{\bullet}$ is sufficient to restore or repeat the branch outgoing from $\mathbf{S}_{\bullet}$. This provides the model with the capability for autonomous repair of damaged STN.

A subset of *objective situations* $\mathbf{S}_{\mathrm{G}}$, $\mathbf{S}_{\mathrm{G}}\in\Omega^{+}(\mathbf{S})$, is used as the target to bring the vehicle to the flight envelope during the recovery path optimisation. In flight safety applications, the *main flight objective* and the *alternative flight objective* are defined. The latter type is used when, due to an emergency, it is impossible to carry on the normal control scenario to achieve the objective of flight. This may occur if the links between the current situation and the main objective cannot be maintained via STN.

The *'leaf' situation* ($\mathbf{S}_{\lozenge}$) completes any STN branch. The branch ends when the vehicle enters a zone of irreversible or catastrophic situations (i.e. $\mathrm{S}_{\lozenge}\equiv\mathrm{S}_{\downarrow}$, or $\mathrm{S}_{\lozenge}\equiv\mathrm{S}_{\oplus}$), or the vehicle returns to the flight envelope ($\mathbf{S}\in\Omega^{+}(\mathbf{S})$ or $\mathbf{S}\in\Omega^{\mathrm{OK}}(\mathbf{S})$), or if the flight scenario has been completed. The *'root' situation* ($\mathbf{S}_{\Delta}$) is the first situation of some phase of flight; it links the tree (subtree) with a previous part of flight. From $\mathbf{S}_{\Delta}$ STN begins growing. For any subtree $\Sigma^{'\mathrm{sit}}$, $\Sigma^{'\mathrm{sit}}\subset\Sigma^{\mathrm{sit}}$, its basic situation $\mathbf{S}_{\bullet}$ is also its root situation.

The *pilot's 'comfort' situation* ($\mathbf{S}_{\mathbf{OK}}$) is any situation from $\Omega^{\mathrm{OK}}(\mathbf{S})$ where the pilot (autopilot) can demonstrate the

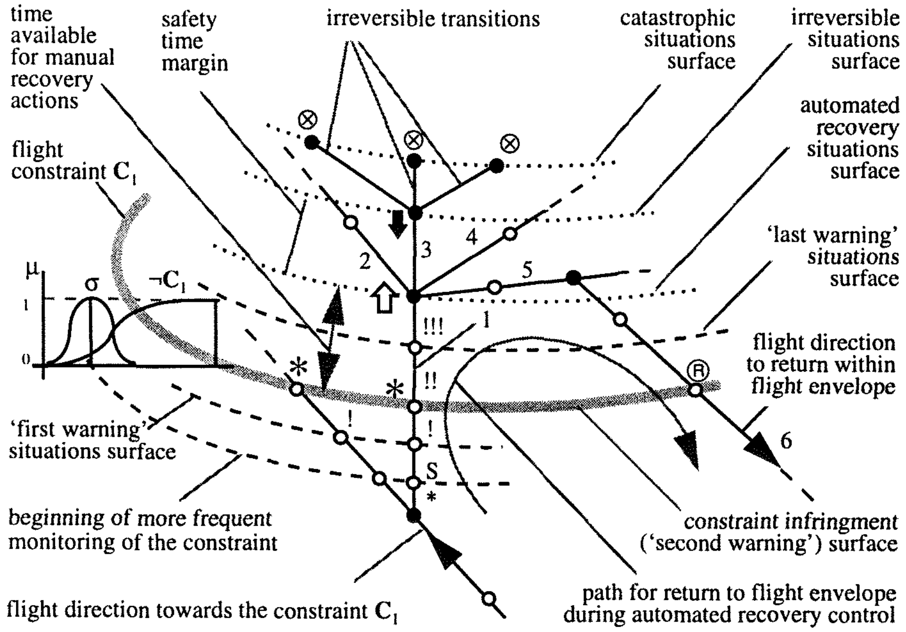


Fig. 13. Situational tree-network topology at a flight constraint

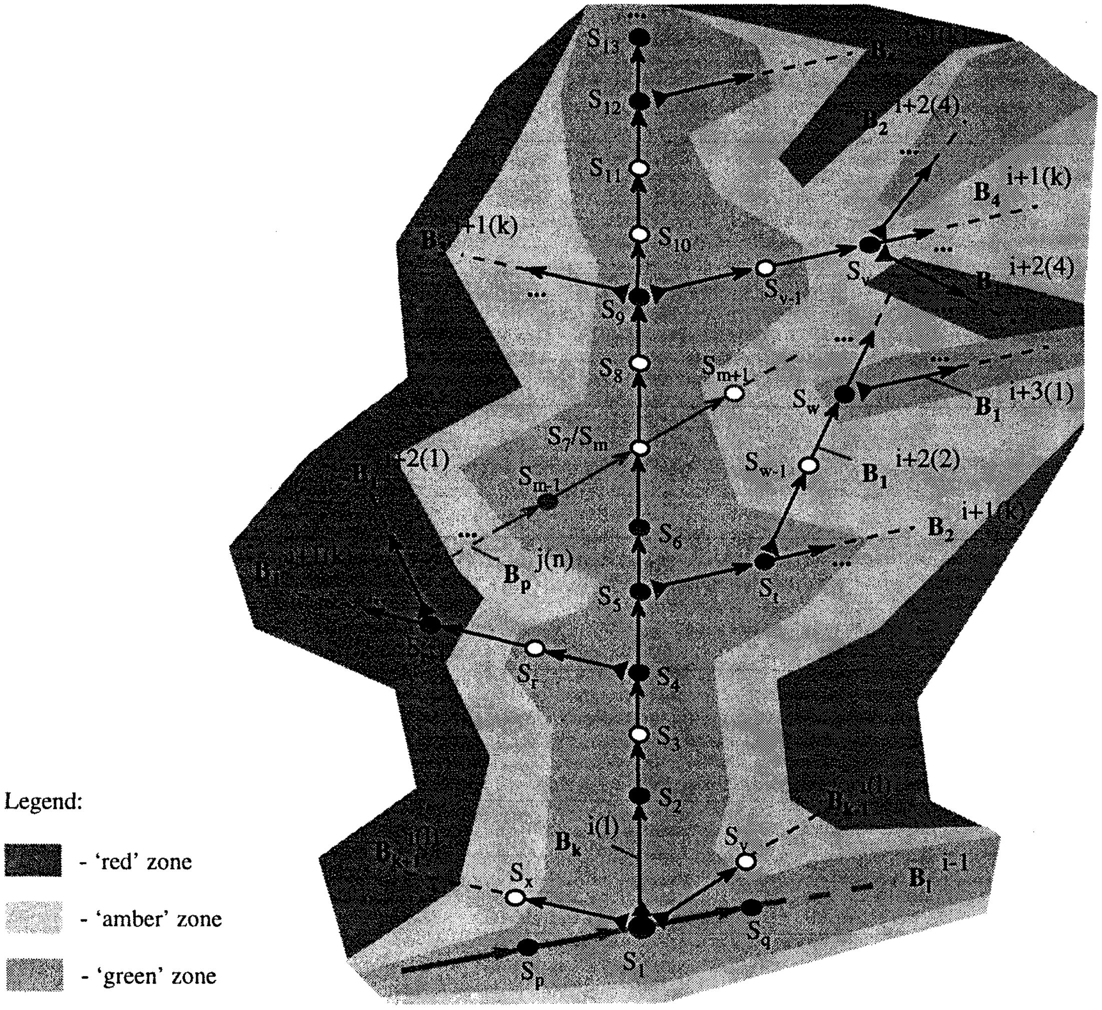


Fig. 14. Example of situational tree-network colouring (flight safety grades)

maximum performance with the minimum workload (see also below). The *constraint monitoring start situation* ($\mathbf{S}_M$) triggers the procedure for monitoring the nearest flight constraints. It is close to **C**, but still $\mathbf{S}_M \in \Omega^+(\mathbf{S})$. A general criterion for $\mathbf{S}_M$ identification is as follows:

$$((\mathbf{S}(t) \in \Omega^+(\mathbf{S})) \wedge (\mu_C(\underline{x}^k(t) \leq \alpha_M) \wedge (\mu`_C(\underline{x}^k(t) < 0)) \\ \Rightarrow (\mathbf{S}(t) = \mathbf{S}_M)). \qquad (24)$$

The danger warning situations ($\mathbf{S}_{!}$, $\mathbf{S}_{!!}$ and $\mathbf{S}_{!!!}$ - *first, second and third warning situations*, respectively) serve to warn the pilot about the dangerously close distance between the aircraft's state and the critical flight constraint. These distances (by time or by other key variable) are measured only when the constraint has been infringed. The situations $\mathbf{S}_{!}$, $\mathbf{S}_{!!}$ and $\mathbf{S}_{!!!}$ are ordered as follows (Fig. 13):

$$\mathbf{S}_M \rightarrow \mathbf{S}_{!} \rightarrow \mathbf{S}_{!!} (\rightarrow \mathbf{C}_{crit}) \rightarrow \mathbf{S}_{!!!}, \qquad (25)$$

where $\mathbf{S}_{!!} \equiv \mathbf{S}_*$ is possible. The *last warning situation* ($\mathbf{S}_{!!!}$) belongs to the unsafe flight modes zone ($\mathbf{S}_{!!!} \in \Omega^-(\mathbf{S})$). The *constraint infringement situation* ($\mathbf{S}_*$) characterises the point when the vehicle just leaves the safe zone $\Omega^+(\mathbf{S})$. These situations mark the borders of the specified flight envelope. The criterion for recognition of the $\mathbf{S}_*$-type situations is defined by (21).

In the *automated recovery situation* ($\mathbf{S}_{\Uparrow}$) special control inputs must be applied by the model to prevent the vehicle from entering irreversible situations $\mathbf{S}_{\downarrow}$ and to bring it back to $\Omega^+(\mathbf{S})$ across the $\mathbf{S}_*$ surface. Together with the use of multi-level warning situations, this constitutes the method for imitation of the human's *self-preservation instincts*, or rational, *knowledge-based*, sensation of fear. In terms of decision making in emergencies, the situations of the $\mathbf{S}_{\Uparrow}$-type are called *critical situations*.

To assign status $\Uparrow$ to a situation **S**, the analysis of its possible consequences is required. The ratio between safe and unsafe developments of $\mathbf{S}_{\Uparrow}$ which occur as the result of action of the flight factors is calculated. The distance to the surface of irreversible situations (*safety time margin*) is also used. The criterion for assigning a situation from $\Omega^-(\mathbf{S})$ the automated recovery status is as follows:

$$(\mathrm{P}[\mathbf{S}(t+\Delta) = \mathbf{S}_{\downarrow} \vee \mathbf{S}(t+\Delta) = \mathbf{S}_{\otimes}] \leq \alpha_{min}) \Rightarrow (\mathbf{S}(t) = \mathbf{S}_{\Uparrow}). \qquad (26)$$

The criterion (26) means that the automated recovery procedure starts when the proportion of dangerous transitions (leading to $\mathbf{S}_{\downarrow}$ or $\mathbf{S}_{\circledR}$) from $\mathbf{S}_{\Uparrow}$ via STN is not less than some threshold α_{min}. (Note: the actual chances for recovery may differ, but the model, like the human pilot, conducts control based on the available knowledge, i.e. on STN.) The *successful return situation* ($\mathbf{S}_{\circledR}$) is the situation when the vehicle first enters the safe zone $\Omega^+(\mathbf{S})$ arriving from $\Omega^-(\mathbf{S})$. The *irreversible situation* ($\mathbf{S}_{\downarrow}$) is the situation **S**, $\mathbf{S} \in \Omega^-(\mathbf{S})$, from which an inappropriately large percentage of transitions lead to catastrophic situations:

$$(\mathrm{P}[\mathbf{S}(t+\Delta) = \mathbf{S}_{\otimes}] > \alpha_{max}) \Rightarrow (\mathbf{S}(t) = \mathbf{S}_{\downarrow}). \qquad (27)$$

The *catastrophic (fatal) flight situation* ($\mathbf{S}_{\otimes}$) is defined as a physical condition when flight is impossible.

Pilot's 'comfort' zone. *A zone of a pilot's (autopilot's) 'comfort' flight modes* $\Omega^{OK}(\mathbf{S})$, $\Omega^{OK}(\mathbf{S}) \subset \Omega^+(\mathbf{S})$, are the situations which the pilot (autopilot) performs in an absolutely reliable and goal-oriented control made with the minimum workload. Evidently, this zone should be the inner portion of the flight envelope. $\Omega^{OK}(\mathbf{S})$ can be defined as the situations which correspond to relatively 'quiet' parts of the standard control scenario, or as balanced straight-and-level, or gentle ascending (descending) flight.

STN's topology at the constraints. Characteristic surfaces and zones. The STN's topology at the constraints is unfolded during the STN construction. Fig. 13 depicts a branch leading towards a flight constraint $\mathbf{C}_1$. The following surfaces which separate characteristic situations introduced above are shown:

$$\{\mathbf{OK}, \mathbf{M}, !, !!, *, !!!, \Uparrow, \circledR, \downarrow, \otimes\}. \qquad (28)$$

The surfaces (28) split STN into the following *flight safety (mission success) zones*:

- *green zone* [**OK**...**M**] - safe flight modes (transitions and situations) including the pilot's 'comfort' zone;
- *amber zone* [**M**...*] - *flight modes* between the constraint monitoring surface and the constraint;
- *red zone*]*...↓(⊗)] - flight modes between the constraint and the irreversible (catastrophic) surface.

Therefore, all flightpaths at the constraints can be arranged into two groups (Fig. 13: the *unsafe paths* leading the vehicle to the irreversible $\mathbf{S}_{\downarrow}$ or catastrophic $\mathbf{S}_{\otimes}$ situations (the paths 1-2, 1-3, and 1-4), and the *safe paths* which bring the vehicle to $\Omega^+(\mathbf{S})$ (the path 1-5-6).

The proposed 'green-amber-red' scheme is useful for colouring STN by the flight safety, or mission success, grades (Fig. 14). The 'green-amber-red' coloured subtree displayed on the **VFS** screen provides the pilot with the integral picture of safety of the future flight paths during the next, say, 10-30 seconds. This method matches ergonomic criteria and behavioural psychology of the human pilot. The surfaces (28), together with safe and unsafe flightpaths, also represent the conceptual basis for the *anthropomorphic-mathematical mechanism for automated protection of the flight envelope*. This mechanism takes into account the integral characteristics of STN components.

Integral characteristics of STN and its components. To assess the level of knowledge in STN (subtree), the following *integral characteristics of STN* (subtree) can be used: the flight scenario, STN's power by situations, STN's power by transitions, total numbers of branches in STN, total flight time of STN, lists of factors and factor levels examined in STN.

The *flight scenario* is the reference to the scenario of the main branch. The *STN's power by situations*, $N(\Omega(\mathbf{S}))$, is the total number of flight situations stored in STN (the number of vertices in graph Σ^{sit}). $N(\Omega(\mathbf{S}))$ is used to characterise the *competence level* of STN at the particular constraint. The *STN's power by transitions*, $N(\Omega(\mathbf{T}))$, is the total number of transitions in STN (the number of arcs in Σ^{sit}). The *total numbers of branches, or density of STN's top*, $N(\Omega(\mathbf{B}))$, may be defined as the sum of all branches which thread a particular region of the flight space. This characteristic is useful, in particular, for estimation of the time required to retrieve data from STN.

The *total flight time of STN*, $TT(\Sigma^{sit})$, is the sum of the length (by time) of all transitions from STN:

$$TT(\Sigma^{sit}) = \Sigma\Delta_i,\ i = 1, \ldots, N(\Omega(\mathbf{T})). \qquad (29)$$

The characteristic $TT(\Sigma^{sit})$ is most important. It shows the 'competence' and the quality of STN. Together with the factors and their combinations, it may be considered as the equivalent of the pilot's total flight time and his piloting experience. The following time characteristics are also useful: flight time within the zone of unsafe flight modes - $TT(\Omega^-(\mathbf{S}))$, total flight time on safe (unsafe) paths outgoing from a situation, $TT(\Omega^-(\mathbf{S}(t) \mid \mathbf{S}(t+\Delta) \in \Omega^+(\mathbf{S}))$ and $TT(\Omega^-(\mathbf{S}(t) \mid \mathbf{S}(t+\Delta) = \mathbf{S}_{\downarrow}$ or $\mathbf{S}(t+\Delta) = \mathbf{S}_{\otimes}))$, total flight time with the presence of a specific factor (combination of factors), and total flight time at a particular constraint.

Note. Comparison of the competence levels of the pilot and the model provides the objective basis for the identification of the flight modes with the better or worse performance of the model and the pilot. This will help in resolving ethical and legal problems arising when the control take-over from the pilot (autopilot) to the model or back has to be made.

Characteristics of situations from amber and red zones. For a situation **S**, located on the surface **M** or behind it towards the constraint, the following *flight safety characteristics* are calculated: • *total number of unsafe situations* (i.e. located on the surfaces ↓ and ⊗) and *safe situations* (on ® and **OK**) which can be reached from **S**; • *total number of unsafe* (i.e. leading to the surfaces ↓ or ⊗) and *safe* (which bring the vehicle back to ® and **OK**) *transitions* outgoing from **S**; • probability of recovery from **S** via STN; • distances to the characteristic situations from $\Omega^+(\mathbf{S})$ and $\Omega^-(\mathbf{S})$; • safety time margin for manual recovery before the automated procedure is initiated; etc.

Transition characteristics. The characteristics of transitions from STN are defined by analogy with situations. These are the differences or maximum/minimum of characteristics calculated for situations $\mathbf{S}(t)$ and $\mathbf{S}(t+\Delta)$, e.g.: gradients $\mu'_C(\underline{x}^k(t)$ (see (24)), increments of main variables (e.g.: altitude, speed, vertical speed), etc.

Addressing techniques in STN. The technique for addressing objects within STN is based on the information of fuzzy states and factors of the current flight situation $\mathbf{S}(t)$. It implements the following mappings:

$$A^{\underline{x}}: \underline{x}(t) \rightarrow (\mathbf{B}, \mathbf{S}), \text{ and } A^{\Phi}: \Phi[\Lambda] \rightarrow (\mathbf{B}, \mathbf{S}). \qquad (30)$$

Mappings (30) provide the fast access to STN records on disk, as well as the economical use of core memory for processing the loaded subtree(s). This subtree contains future flightpaths which begin (or include) $\mathbf{S}(t)$ and examines short-term propagation of the required flight factors and their levels. The mapping $A^{\underline{x}}$ is implemented as a special multi-level hierarchical structure which includes two-dimensional arrays and general *M*-way balanced trees. Only those subtrees which cover the current situation $\mathbf{S}(t)$ in terms of fuzzy affinity (10) are loaded for processing.

The principles of automated decision-making using STN. The following three principles form the basis for automated decision making in an emergency: • monitoring of the current flight situation position in relation to the constraints; • short-term forecast of flight; and • the use of the 'self-preservation imperative'.

1. Monitoring of the relative position of a current flight situation and constraints. Compatibility of the current situation and the constraints, the distance to the constraints and the rate of this change are evaluated. A surface, or zone, which the current situation belongs to, is identified (**M**, **!**, **!!**, *****, **!!!**, ⇑, etc.).

2. Short-term forecast of the flightpath-alternatives. The transitions which constitute the loaded subtree are used to make dynamic forecasts of the future paths outgoing from the situation. Both dangerous and safe alternatives are investigated. The key factors which are *controllable* (manageable) and *uncontrollable* (weather, failures, etc.) associated with these excursions are identified.

3. The use of the 'self-preservation imperative' for flight envelope protection. In order to secure the human-like responsibility for safe outcome of the automated decisions, the model is to imitate the human pilot's *self-preservation instincts* (a reasonable, *knowledge-based*, sensation of fear). The following components and functions help implement this feature: • awareness of the currently critical constraint C_{crit}, the distance to C_{crit} and the rate of its change; • knowledge of the key factors, their effects and available counter-measures; • knowledge of safe and unsafe paths from the situation in the presence of the key factors and control inputs; • multi-level defensive measures (warnings, prevention, counter-actions) designed to protect the flight envelope.

Automated flight envelope protection mechanism. When the vehicle crosses one of the warning surfaces (!, !!, *(!!), or !!!), the model generates information imperatives (warnings, instructions, prohibitions, explanations) to advise of the vehicle approach to the zone of critical situations. In the automated recovery situation (on ⇑), the model has a sufficient justification derived from the inspection of STN to disengage the pilot (autopilot) and apply automated control inputs to restore flight (Fig. 1). To find the optimum (rational) path which brings the vehicle back to $\Omega^{OK}(S)$, the characteristics of interim situations and transitions are examined.

The following criteria for recovery path optimisation may be used (Fig. 15): • *minimum time* to return within the comfort zone $\Omega^{OK}(S)$, • *maximum compatibility* with the constraints of intermediate situations (between $S_{\Uparrow}$ and $S_{\circledR}$), or • *minimum consumption* of control resources, or non-use of some (e.g.: inoperative) controls.

Main types of information imperatives. With the aim of real-time knowledge provision the following types of *information imperatives* (messages) can be used: *warnings* (**WNG**), *prohibitions* (**PRHB**), *instructions* or *recommendations* (**INSTR**), and *explanations* or *motivations* (**WHY**). The examples are as follows: **WNG**$_5$: `'windshear rate -2.5m/s'`, **PRHB**$_9$: `'do not fully retract flap'`, **INSTR**$_8$: `'reduce engine#4 power to 85%'`, **INSTR**$_{13}$: `'keep pitch at 6°'`, **WHY**$_7$: `'bank 30° in 4s due to asymmetry'`, **WHY**$_8$: `'unable to counter asymmetrical power in 6s'`.

Together with the situational forecast displayed on the **VFS** screen (similar to Fig. 14), these imperatives constitute the *information output* of the model. Automated real-time production of such messages is possible based on the inspection of STN.

Model's operational modes. The following *operational modes* of the model are outlined: • *green mode* - monitoring flight situations and constraints if the vehicle is approaching the constraints; • *amber mode* - flight monitoring and knowledge provision when flight is within $\Omega^{+}(S)$ but close to the constraints; • *red mode* - flight monitoring, knowledge provision and performance of the automated recovery procedure to restore flight when the vehicle enters the zone of critical situations; • *simulation mode* - flight modelling on pilot's request conducted when flight is within $\Omega^{OK}(S)$); • *post-training mode* - automated post-training of STN when flight is in $\Omega^{OK}(S)$ and spare computer resources are available.

The principles of model functioning in the amber and red modes (knowledge provision in an emergency) are illustrated in Fig. 16. This is a hypothetical example of an airliner's take-off under strong windshear conditions combined with a critical engine failure. It demonstrates the outlined mechanism for automated flight envelope protection based on the self-preservation imperative.

Examples of model testing

Both the micro- and macro-components of the model have been implemented and tested separately on a computer.

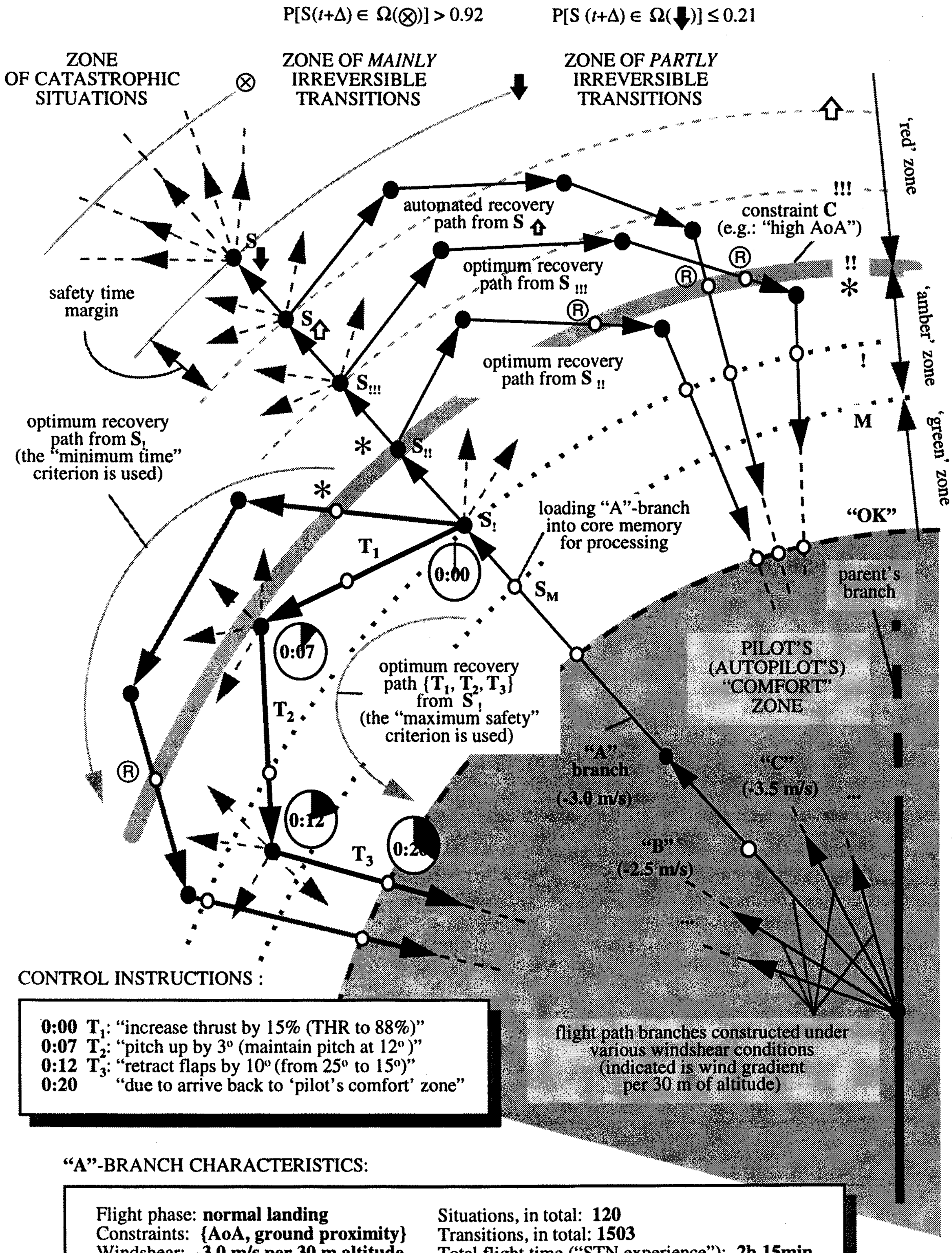
The micro-model prototype has demonstrated its outstanding performance in numerous applications. Over thirty problems in the areas of flight safety, aircraft piloting and flight tests have been studied. Some 150 complex flight situation scenarios have been constructed and modelled for fifteen aircraft types including aeroplanes and helicopters. The model has also been tested as a flight scenario tool and the autonomous piloting model within a flight simulator of an aerospace vehicle.

The macro-model prototype (STN) has been tested as the technique for analysis, optimisation and modelling of the high-altitude hypersonic manoeuvring of an airspace plane during an aerodynamic turn of its orbital plane under uncertain flight conditions (4, 13). Its initial variant has also been tried as the fuzzy model of the aeroplane's flight dynamics and control on landing in severe weather conditions (10).

The following test examples are demonstrated: FLA prototype F-93A (14) take-off under complex conditions (two right-hand engines failure), airliner's take-off accident under strong windshear and heavy shower conditions ('microburst'), the results of STN prototype construction flight path optimisation, and high-altitude hypersonic manoeuvring of an aerospace vehicle under uncertain conditions (aerodynamic turn of its orbital plane).

Example 1. A realistic scenario of a complex flight situation **S**$_2$: `'Take-off of an FLA prototype F-93A with two engines out'` is represented in Fig. 5. The graph combines, in fact, scenarios of two very close situations of the S_2 type: one with pilot errors and another with the correct piloting after the second engine failure. This situation starts from the event **E**$_1$ and finishes at **E**$_{90}$. The details of the situation are clear from the graph. (Note: some processes have no specified target events because the latter ones are recognised automatically; some events are also introduced only for the purpose of information.) The results of modelling this complex situation with correct piloting in an emergency are shown in Fig. 17. In total, over ten various flight scenarios have been examined, including take-off, landing and en-route modes.

Example 2. The results of modelling of a flight accident **S**: `"airliner's take-off under severe microburst and shower conditions"` are



CONTROL INSTRUCTIONS :

0:00 T_1: "increase thrust by 15% (THR to 88%)"
0:07 T_2: "pitch up by 3° (maintain pitch at 12°)"
0:12 T_3: "retract flaps by 10° (from 25° to 15°)"
0:20 "due to arrive back to 'pilot's comfort' zone"

"A"-BRANCH CHARACTERISTICS:

Flight phase: **normal landing**
Constraints: **{AoA, ground proximity}**
Windshear: **-3.0 m/s per 30 m altitude**
Thrust rating: **{55% ... 100%}**
Goal pitch range: **{-5° ... +15°}**
Goal bank range: **{-15° ... +15°}**
Flap control variations: **{0° ... 25°}**

Situations, in total: **120**
Transitions, in total: **1503**
Total flight time ("STN experience"): **2h 15min**
Safety time margin: **5 sec**
Accident probability (at ⬇ level): **0.92**
Recovery probability (at ⇧ level): **0.79**
...

Fig. 15. Example of multi-level flight envelope protection and recovery path optimisation using STN

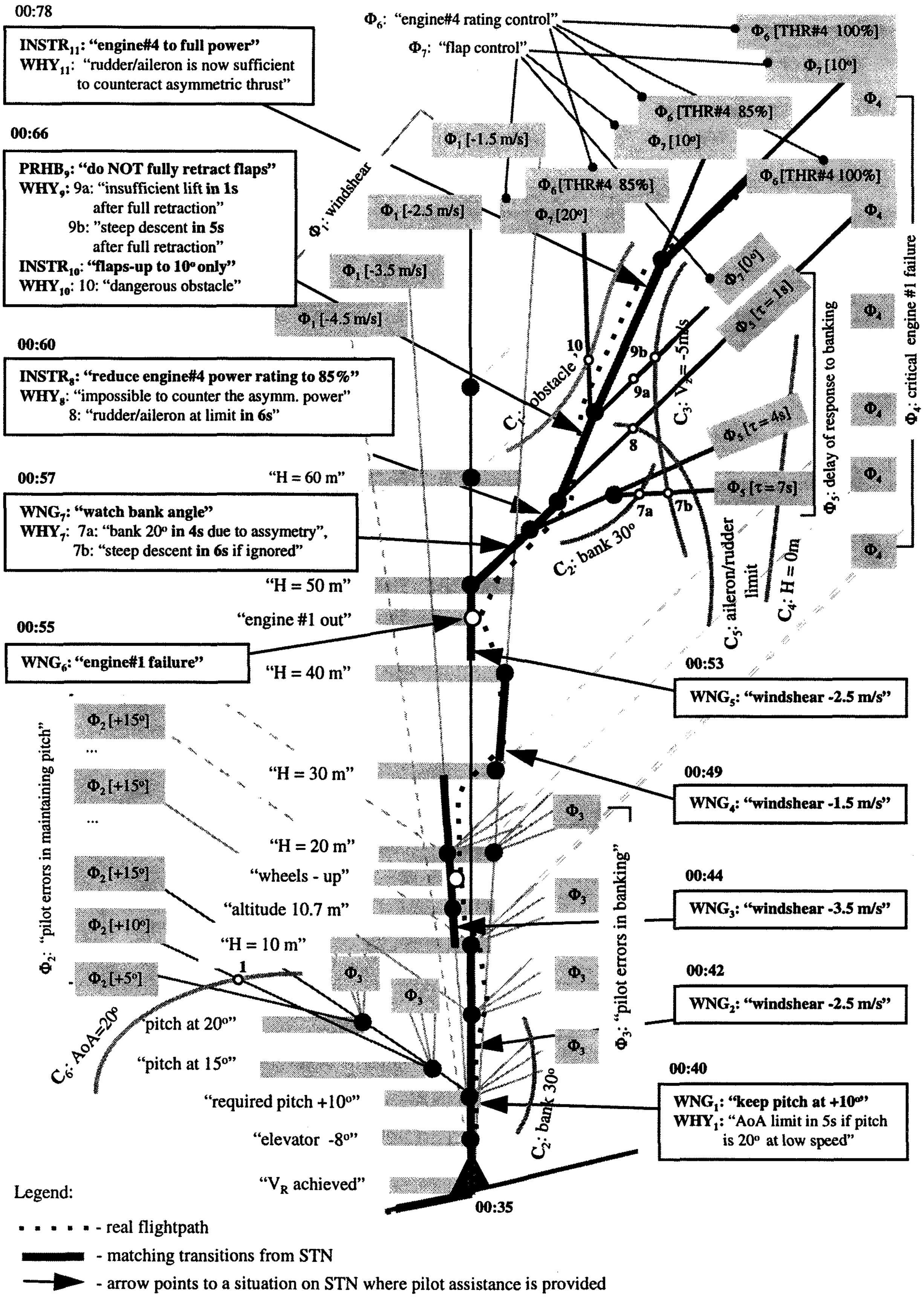


Fig. 16. Hypothetical example of model functioning during take-off with windshear and critical engine failure

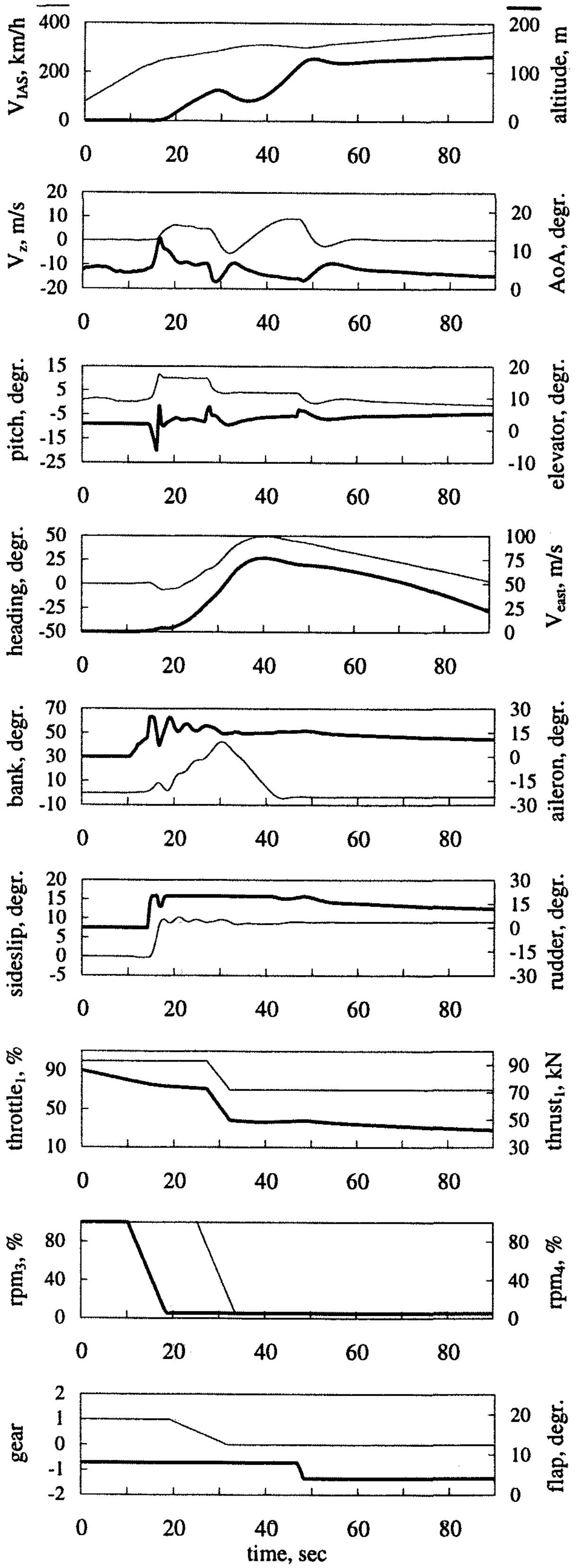
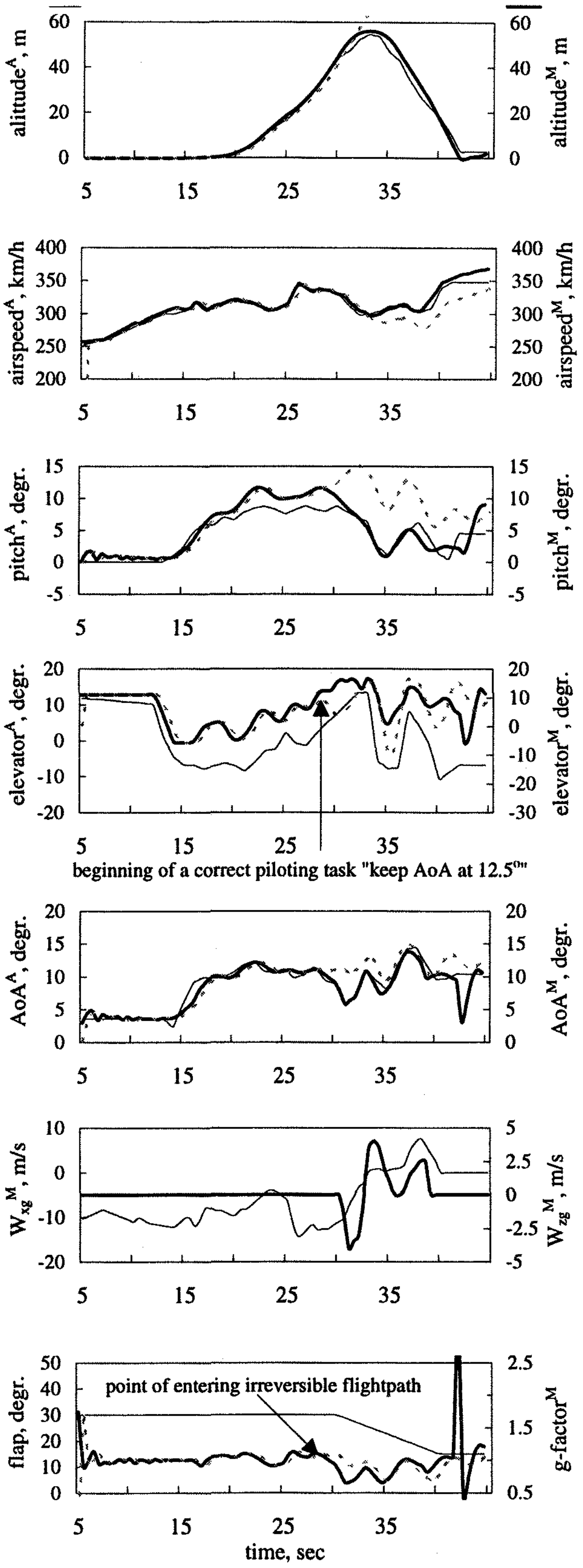


Fig. 17. FLA prototype F-93A take-off with two right-hand engines out



beginning of a correct piloting task "keep AoA at 12.5°"

point of entering irreversible flightpath

Fig. 18. Take-off under windshear and shower conditions (microburst)

shown in Fig. 18. Combined with the pilot's incorrect response to the stall alarm (excessive stick move forward at t = ~28 sec), this situation has resulted in the vehicle entering a zone of irreversible flightpaths. Note that the flightpath has become irreversible 15 seconds before the crash. This is the critical situation.

Modelling also demonstrates that a safe outcome was possible under these conditions. The correct piloting would be to keep the aircraft's angle of attack at the maximum limit of about 12^o-13^o (dashed curves). The use of the micro-model was crucial for this particular investigation. The model helped to construct and examine in detail various possible piloting techniques, as well as to identify windshear conditions of the accident. However, an attempt to model actual control inputs failed (to repeat this complex situation and to investigate its alternatives).

Example 3. The results of the construction of STN prototype for an aerospace vehicle are shown in Fig. 19-20. This STN has been constructed to cover various flight modes of an aerospace vehicle during high-altitude hypersonic manoeuvring in an aerodynamic turn of the orbital plane (4). The factors examined in STN are variations in the command angle of attack and bank angle. STN takes into account uncertainty as to the atmospheric density at high altitudes as variations within [-30%; +30%]; this is the third factor. The appropriate flight constraints have been also implemented.

Example 4. Application of STN for optimisation and modelling of an aerospace vehicle flight and its control under uncertain conditions is demonstrated in Fig. 21. In this 'flight' the model performs reliable control of the vehicle with its weight 30% heavier than one used for STN construction. During simulation, the atmospheric conditions of flight were varied within [-20%; +20%]. A special series of simulation experiments has also been conducted to justify the STN performance under other non-standard conditions and variations of the model parameters.

Model application areas

The following three techniques are proposed (but there is no restriction on these) for model implementation: the tool for automated/autonomous flight control scenario planning, the situational forecast display, and the flight safety (mission success) indicator. The first technique has already been demonstrated (see Fig. 5).

Situational forecast display (SFD). The **VFS** concept can be useful for the development of an *adjustable intelligent pilot-vehicle interface*. SFD is aimed at providing the pilot with a dynamic integral picture of future (10-30 seconds) flight alternatives, taking into account the key factors and control inputs. SFD has green-amber-red colouring of the main safety (mission success) zones. It will also show the distance to critical situations and the key factors.

A structure of SFD is much like that shown in Fig. 14-16, provided with semantic legends for branches and with messages produced by an audio system. The pilot may examine (activate, alter) the required flightpath (control scenario) by a finger touch control applied to the appropriate branch segment. Alternatively, a laser scanner would be used to pinpoint the pilot's optical focus point on the display to activate the desired route within the subtree depicted on SFD. The maximum safety (mission success), the minimum time or other criteria can be used to find the paths which will determine the sequence of the model's control actions.

Flight safety (mission success) indicator (FSI/MSI). A draft of FSI/MSI is shown in Fig. 22. The display depicts the current distances to the nearest constraints and chances for safe and unsafe developments of the flight (sector diagram). It also contains messages reproduced by an audio system on the currently important key events and processes (both manageable and uncontrollable).

Application areas. The model can be used as a prototype for AI systems in the following areas: • automated flight envelope protection, • pilot assistance and in-flight pilot training, • autonomous, robotic or automated flight control under complex (unknown, multi-factor, rapidly changing, hostile) conditions, • automated identification of the flight envelope, • production of situational rules for semantic knowledge generators, etc.

Conclusions

An AI pilot model concept for studying pilot's decision making processes in complex flight situations has been developed. The model imitates a human pilot's self-preservation instincts and capability for short-term prediction of flight. This mechanism is fused with the complementary artificial properties useful in emergencies. The model's function is based on the tree-network of situational knowledge derived from the aircraft's flight dynamics model. It is specially constructed to examine (thread) the potentially critical zones at the constraints where the flight may develop irreversibly.

The micro-model has been tested in various tasks in the areas of flight safety and flight control for several aircraft types, including aeroplanes, helicopters and an aerospace vehicle. The macro-model prototype (STN) has been tested in two applications for analysis and optimisation of aircraft's behaviour under uncertain and complex conditions. Both models have demonstrated their potential.

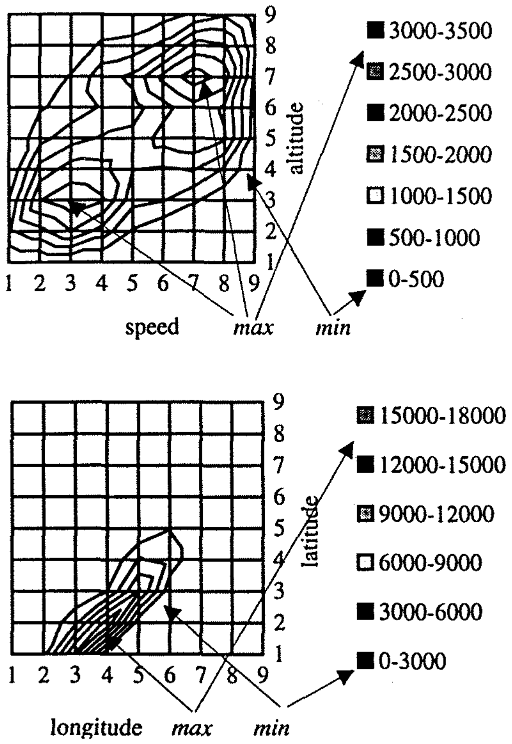


Fig. 19. Projection of fuzzy situations from STN (on planes "altitude - speed" and "latitude - longitude" of the vehicle's fuzzy state space (N($\underline{X}^i$) = 9, i = 1, 2)

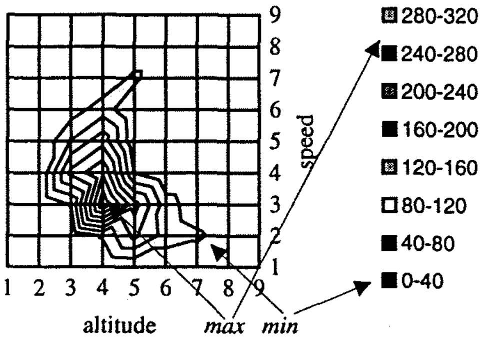


Fig. 20. Projection of optimised fuzzy situations (i.e. those which have been linked with the objective subset by transitions from STN) on the "speed - altitude" plane of the vehicle's fuzzy state space (N($\underline{X}^i$) = 9, i=1, 2)

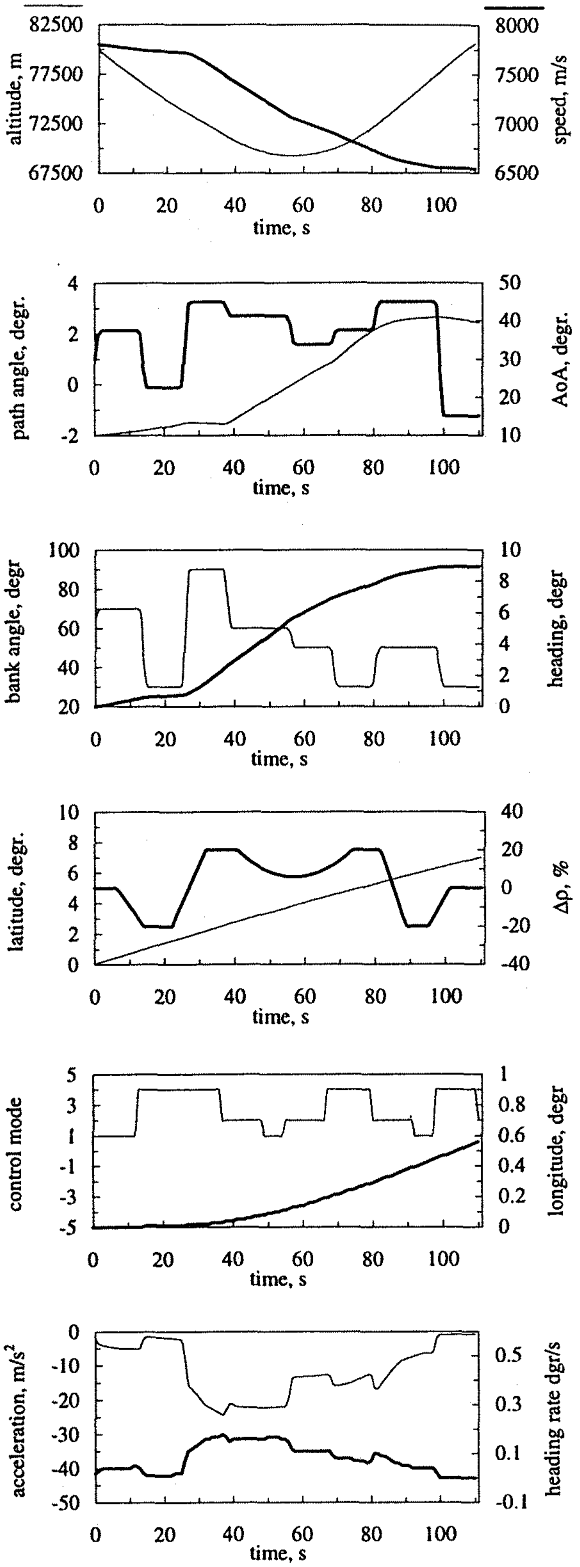
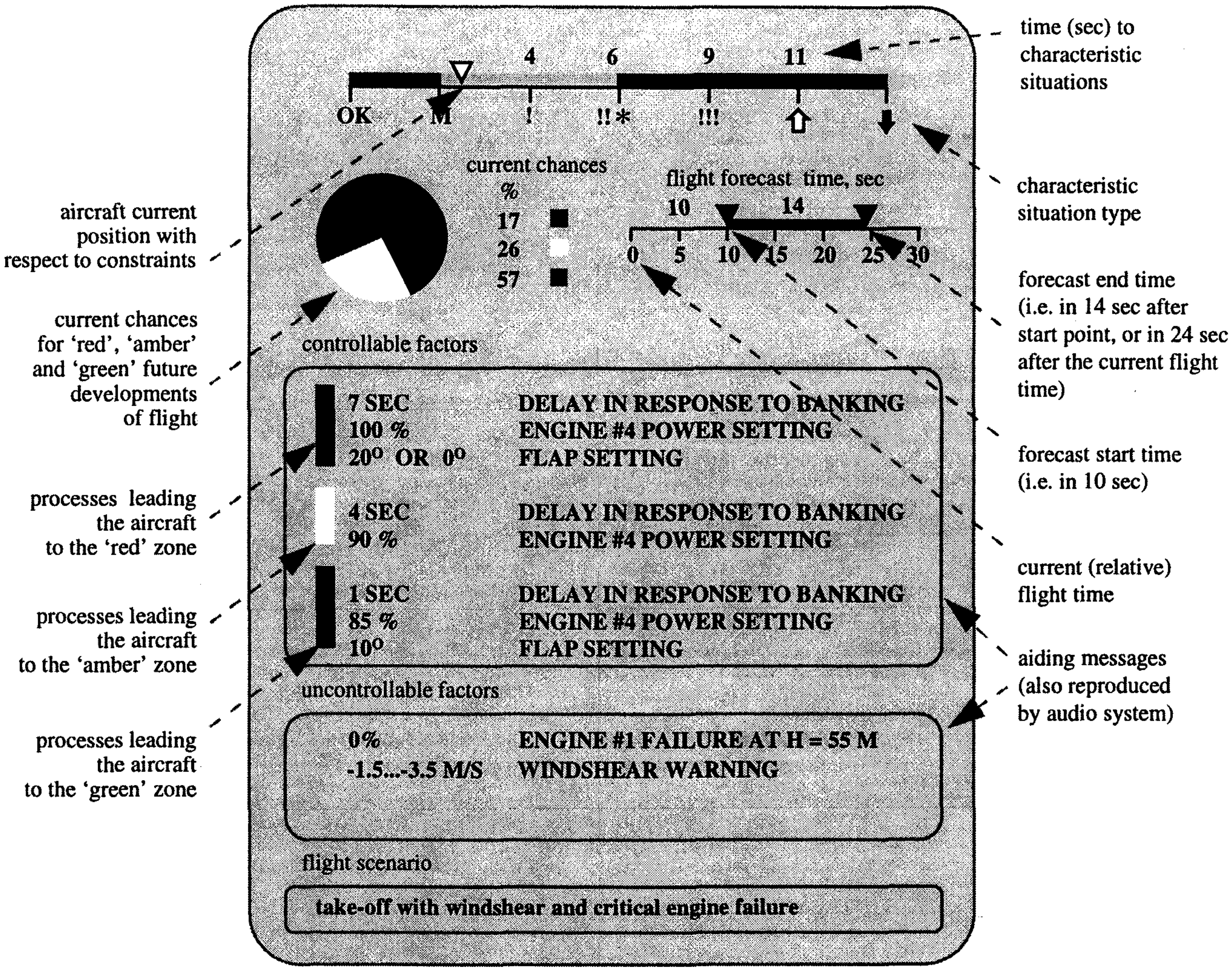


Fig. 21. STN-based control of an aerospace vehicle during hypersonic high-altitude maneuvering under uncertain operational conditions (atmospheric density variations and non-standard (+30%) weight of the vehicle)



Legend:

CONTROLLABLE (MANAGEABLE) FLIGHT FACTORS:

Φ_5 [+L]: "delay in response to banking - **positive large** level"
Φ_6 [0]: "engine #4 power setting - **standard** (take-off) level"
Φ_7 [+M] and Φ_7 [-M]: "flap setting - **positive medium** and **negative medium** levels"

Φ_5 [+M]: "delay in response to banking - **positive medium** level"
Φ_6 [-S]: "engine #4 power setting - **negative small** level"

Φ_5 [0]: "delay in response to banking - **standard** level"
Φ_6 [-M]: "engine #4 power setting - **negative medium** level"
Φ_7 [0]: "flap setting - **standard** level"

UNCONTROLLABLE FLIGHT FACTORS:

Φ_4 [-VL]: "critical engine #1 failure - **negative very large** level"
Φ_1 [-S, -M, -L]: "windshear warning" - **positive small, medium and large** levels"

Fig. 22. Flight safety indicator (example)

Using the micro-model, flight control scenarios can be loaded as external data structures into the flight control software. The events-processes language can be used for specification of both manual and automated flight control.

The human pilot's experience is represented in the form of a fuzzy situational tree-network. The main advantage of this knowledge structure is the capability for compact and comprehensive coverage of various non-standard flight conditions. The tree-network is provided with the integral characteristics of its competence similar to those used to measure a human pilot's experience. This provides the objective basis for the development of criteria to share and take over the responsibility for flight control between the pilot and the model in emergencies.

The situational tree-network concept can be used to study properties of a human pilot's knowledge and decision making, identify causes of a pilot's skills decay and errors in emergencies, etc. The model provides hypotheses for explanation of a pilot's inability to cope with critical flight situations (the action of chain reaction mechanisms during flight incident development, combined with insufficient chances for recovery).

In emergencies, external intelligent support is required to prevent the vehicle from entering the zone of critical situations. An anthropomorphic-mathematical mechanism for flight envelope protection is proposed. It is based on the identification of the approach to critical situations and evaluation of their outcomes. The key events and processes which may divert flight to either unsafe or safe flightpaths can be derived automatically from the flight scenario. The model may be used as the basis for the development of intelligent pilot assistance systems.

Potential areas for model application include real systems and virtual environments which require dynamic planning and performance of situational flight control under complex (multi-factor, unknown, hostile, dynamically changing) conditions. In particular, the concepts of the situational forecast display and the flight safety (mission success) indicator based on the model are outlined.

Future work will be concentrated on the model prototyping for one of its applications. Research into SFD and FSI, dynamic STNs and direct pilot-model communication techniques will be continued. However, the techniques for the automated construction and analysis of situational tree-networks are most important.

Acknowledgements

This work is a part of a PhD research course funded by an ORS Award (Ref. 93089005) from the CVCP of the Universities of the United Kingdom. The study is partly supported by the College of Aeronautics of Cranfield University. The author would like to acknowledge the contribution of Air Foyle Limited (UK). At the initial stage of this research substantial funds have been received from Cand.Sc. Yu.Pribylsky and Mr V.Golubev (Riga). A travel grant for attending the 20th ICAS Congress has been kindly provided by the Royal Aeronautical Society (UK). Many thanks to all of these parties.

The author conveys his sincere gratitude to Prof. D.Allerton, Head of the Department of Avionics of the College of Aeronautics, for providing a stimulating research environment and support for this work. Special thanks are extended to Mr D.Lowe, Director of Lion Aviation Limited (UK), for his invaluable multi-aspect help during this study. The author is grateful to Dr O.Parfentyev (Riga) for useful discussions and for sharing several valuable ideas contributed to the model.

The author wishes to thank Prof. V.Goriatchev, Director of the Russian State Civil Aviation Research Institute (GosNII GA, Moscow), and Prof. E.Kuklev (Russian Academy of Civil Aviation, St. Petersburg) for their support at the initial stage of this research course. Special thanks are due to the former colleagues at Riga Experimental Centre for Aeronautical Research (Latvia), the former branch of GosNII GA, for their creative help with model prototype application.

References

1. P.G.Thomasson, “Flight Dynamics Simulation”, Cranfield University, Cranfield, 1993.
2. S.A.Gorbatenko, etc., “Mechanics of Flight” (in Russian: С.А.Горбатенко и др. “Механика полета”. М., Машиностр., 1969, 420 с.).
3. D.A.Pospelov, “Situational Control. Theory and practice” (in Russian: Д.А. Поспелов, “Ситуационное управление. Теория и практика”, М., Наука, 1986г., 288 с.).
4. O.M.Parfentyev, “The algorithms of an AI system for controlling 3-D motion of an aerospace vehicle”, Cand.Sc. Thesis, Riga, 1993 (in Russian: О.М.Парфентьев, “Алгоритмическое обеспечение интеллектуальной системы управления пространственным движением космического аппарата самолетного типа”. Диссерт. на соиск. уч. степени канд. техн. наук, РВВАИУ, Рига, 1993г., 196 с.).
5. K.Bamford, P.Curran, “Data Structures, Files and Databases”, Macmillan, 1991.
6. M.Loomis, “Data Management and Files Structures”, Prentice-Hall, 1989.
7. S.Lipshutz, “Data Structures”, McGraw-Hill, 1986.
8. L.A.Zadeh, “Fuzzy Sets”, *Information and Control*, v.8, No.1, 1965, pp. 338-353.

9. L.A.Zadeh, “A Fuzzy-algorithmic Approach to the Definition of Complex and Imprecise Concepts”, *Int. J. Man-Mach. Stud*, 1976, vol. 8, No. 3, pp. 249-291.
10. I.Y.Burdun, “The Development of the Fuzzy-Sets-Theory-based Techniques and Their Application to Computer Modelling of Aircraft Dynamics and Control in Complex Flight Conditions”, Cand.Sc. Thesis, Riga, 1982 (in Russian: И.Е. Бурдун, “Разработка и применение методов, основанных на теории нечетких множеств, для исследования управляемого движения воздушных судов в ожидаемых условиях эксплуатации”. Диссерт. на соиск. уч. степени канд. техн. наук, РИИ ГА, Рига, 1982г., 272 с.).
11. S.Baron, C.Feehrer, “An Analysis of the Application of AI to the Development of Intelligent Aids for Flight Crew Tasks”, NASA Contractor Report 3944, NASA, 1985.
12. I.Y.Burdun, “Some Methodological Issues of Computer-aided Modelling of the Behaviour of the ‘Pilot - Aircraft - External Flight Conditions’ System” (in Russian: И.Е.Бурдун, “Некоторые вопросы методики вычислительного эксперимента в исследованиях поведения системы “пилот-ЛА-внешняя среда”, Сб. *“Вопросы кибернетики. Выпуск ‘Проблемы теории биотехнических систем эргатического типа’ (ВК-153)”*, НСК “Кибернетика”, АН СССР, М., 1989г., с. 45-58).
13. I.Y.Burdun, O.M.Parfentyev, “A Fuzzy-Sets-Theoretic Approach to the Optimisation of Control of an Aerospace Vehicle under Uncertainty” (in Russian: И.Е.Бурдун, О.М.Парфентьев, “Нечетко-множественный подход к оптимизации управления пространственным движением космического аппарата самолетного типа в условиях неопределенности”. Сб. *“Научно-методические материалы по проблемам обеспечения безопасности полета авиации МО СССР”*, ВВИА, ИВВАИУ, Иркутск, 1991г., с. 47-52).
14. D.Howe, “Military General Purpose Aircraft (Turbofan Version) F-93A”, Project Description, Cranfield University, 1993.
15. K.W.Alter and D.M.Regal, “Definition of the 2005 Flight Deck Environment”, NASA Contractor Report 4479, NASA, 1992.

Nomenclature

Symbols

Σ	knowledge tree-network/network
σ	standard fuzzy set-value from a fuzzy scale
∇	relation in the event recognition criterion, $\nabla \in \{>, \geq, <, \leq, =, \neq, \approx, \neg\approx\} \cup \{\in; \notin\}$
δ	events series increment in $\Re$
$\Re$	right part of the event recognition criterion, $\Re \in \{a; a+\delta\} \cup \{[a; b]\}$, $a < b$
$\vee$	logical disjunction (or)
$\wedge$	logical conjunction (and)
τ	time delay
Φ	flight factor (operational factor)
Λ	factor level name
Δ	transition length (by time)
μ	membership/compatibility function (8)
α	threshold value (of μ function or $N(\Omega(\mathbf{T}))$
$\Delta\rho$	atmospheric density variation
$\Omega(...)$	set of elements of ... type
$\Omega^{+}(\mathbf{S})$	subset of safe situations
$\Omega^{-}(\mathbf{S})$	subset of unsafe situations
$\Phi[\Lambda]$	level of the flight factor Φ
$\mu^{`}$	time derivative of the membership function
$\underline{...}$	fuzzy (vector, variable, component, scale, etc.)
$...(t)$	specific value of ... variable at time t
`<data>`	specific value of an attribute in a frame
A	‘atmospheric state’ type process
A	addressing mapping
B	‘airborne system function’ type process
$\mathbf{B}(t_0; t_N)$	branch in STN from $\mathbf{S}(t)$ to $\mathbf{S}(t_N)$
C	flight constraint
D	‘vehicle dynamics’ type process
E	flight event
E	‘event’ type
et	scenario element type, $et \in \{E, T, P, W, ...\}$
F	‘airborne system failure’ type process
$\mathcal{G}$	STN genotype
G	flight goal
H	altitude
П	flight process, $\Pi \in \{\mathbf{D}, \mathbf{B}, \mathbf{F}, \mathbf{T}, \mathbf{O}, \mathbf{P}, \mathbf{G}, \mathbf{C}, \mathbf{W}, \mathbf{A}, \mathbf{R}, \mathbf{Y}, ...\}$
$I(\underline{...})$	index of a fuzzy set-value $\underline{...}$ on a fuzzy scale
j	used (>0) or ignored (<0) element in a list $\mathcal{L}$
l	link in the recongition criterion, $l \in \{\vee; \wedge\}$
M	library
N	name, identifier
$N(\Omega(...))$	number of element in $\Omega(...)$ set
$N(...)$	number of elements in ... set
O	flight ‘state observer’
P	control procedure
P	‘control procedure’ type
P [...]	probability of ...
p, p	dimension (**x** vector)
Q	vector of quality (safety) measures
q, q	dimension (**u** vector)
R	‘rain’ type process
r, r	dimension (**w** vector)
$R[...]$	frame-specification (record) of ...-type object
S	flight situation
$\mathbf{S}(t)$	fuzzy flight situation at t
t	flight time
T	piloting task
T	‘piloting task’ type
$\mathbf{T}(t; t+\Delta)$	transition from situation $\mathbf{S}(t)$ to $\mathbf{S}(t+\Delta)$
$t[...]$	time when ...

TT(...)	total flight time in STN (subtree) ...
U	control space
$\underline{\mathbf{U}}$	fuzzy measurement scales in **U**
$\underline{U}$, $\underline{\mathrm{U}}$	fuzzy scale in **U**
$\boldsymbol{u}$, $\mathbf{u}$	numeric control vector
u, u	**u** vector component
v	model variable
V	airspeed
W	[external] flight conditions space
$\underline{\mathbf{W}}$	fuzzy measurement scales in **W** space
W	'wind' type process
W	'wind-process' type
$\boldsymbol{w}$, $\mathbf{w}$	numeric vector of external flight conditions
X	aircraft's state space
$\underline{\mathbf{X}}$	fuzzy measurement scales in **X**
$\underline{X}$, $\underline{\mathrm{X}}$	fuzzy scale in **X**
$\boldsymbol{x}$, $\mathbf{x}$	numeric state vector
x, x	**x** vector component
Y	'runway surface state' type process

Graphic symbols

!	first warning situations surface
!!	second warning situations surface
!!!	last warning situations surface
°	ordinary situation
•	'bud' situation
◊	'leaf' situation
*	constraint infringement situations surface
⇑	automated recovery situations surface
®	'return to the flight envelope' situations surface
↓	irreversible situations surface
⊗	catastrophic situations surface
M	'constraint monitoring' situations surface
OK	pilot's/autopilot's comfort zone

Superscripts

+	safe
-	unsafe
A	'active' state of a process
A	data from a flight accident
CL	'closed' state of a process
i, j, k	element number in a set, vector, list, or array
M	modelled data
OK	'pilot comfort's' zone (**OK**-zone)
R	'recognised' state of an event
s	scenario
sem	semantic
sit	situational

Note: see also main and graphic symbols.

Subscripts

crit	critical
G	goal
g	earth axes
i, j, k	element number in a set, vector, list, or array
R	rotate
U	control
z	vertical axis in a co-ordinate system

Note: see also main and graphic symbols.

Abbreviations

+prefix	positive
−prefix	negative
0	approximately equal to zero
AI	artificial intelligence
AoA	angle of attack
ATC	Air Traffic Control
Cand.Sc.	'Candidate of Sciences' academic degree
CON	control
CQ	critical quality
CVCP	Committee of Vice-Chancellors and Principals
degr.	degrees
ENV	environment (external flight conditions)
F-93A	FLA prototype (14)
FLA	Future Large Aircraft (14)
FMS	flight management system
FSI/MSI	Flight Safety (Mission Success) Indicator
HUD	head-up display
INSTR	instruction/recommendation
L	large
LE	less or equal
M	medium
MCPR	maximum continuous power rating
ORS	Overseas Research Scheme Award
PR	power rating
PRHB	prohibition
rad.	radians
S	small
SFD	Situational Forecast Display
STN	situational tree-network
SYS	airborne systems
V	very
VFS	virtual flight space
WHY	explanation/motivation
WNG	warning