Computer Science 773

Robotics and Real-time Control

INTERFACES FOR PEOPLE

LABVIEW.

(Pictures from M. Santori : "An instrument that isn't really", *IEEE Spectrum* **27#8**, 36 (August, 1990).)

Labview is a Macintosh package with which the Macintosh can be programmed to act as a control and observation interface for a suitable controlled system. It provides a library of screen layouts representing information displays (meters, digital displays, chart recorders) and controls (knobs, buttons, switches, potentiometers), and software facilities to connect these through a suitable interface to real devices. This gives a (comparatively) easy way to construct quite elaborate control interfaces without the pain of actually building them. The result can be used either as the interface which operates the target system, or as a prototype for a more traditional permanent interface. (It remains easier to turn a knob with your fingers than to poke it with a pointer controlled by a mouse.)

AIRCRAFT.

(From T. Helm : “Air crash puts computers in the dock”, *Dominion*, 5 June 1990.)

THE BRITISH Department of Transport report into the Kegworth crash, due to be published later this year, will, according to the principal investigator from the Air Accident Investigation Branch at Farnborough, Edward Trimble, represent “a milestone in air safety”. That pledge, given at the inquest into 47 deaths on a British Midland 737-400 which crashed into the Ml motorway embankment in January last year, is proof that his inquiry is far more than an isolated examination of one more disaster.

Kegworth, it is widely agreed, has highlighted many of the profound anxieties felt by pilots and experts worldwide about the nature and the speed of so-called “progress” in aviation. The advent of ever more sophisticated “glass cockpits” – the computerised flight decks like that on the British Midland Boeing 737 – has raised vital questions about the relationship between humans and computers, between pilots and the increasingly advanced machinery which, it was hoped, would lessen their workload and improve their performance.

When the giant new Boeing 747-400 “glass cockpit” was rolled out in Seattle in 1988, a senior Boeing designer said: “The dark and dirty night is covered by the fact that every system can stand multiple failures which are taken care of by simple procedures. The aircraft gives warnings and cautions; the pilot goes to his quick reference checklist and follows procedures.”

The reality has on many occasions proved alarmingly different. More and more reports of suspected mid-air computer malfunctions, and the apparent difficulties some pilots have had understanding new equipment, are causing profound concern to Britain’s Civil Aviation Authority. The confusion over what actually happened on flight BD 092 from Heathrow to Belfast on January 3 last year is a case in point.

Investigators, as was revealed at the Kegworth inquest, have established that Captain Kevin Hunt and his first officer David McClelland, both experienced pilots, switched off their functioning starboard engine after smelling burning and feeling vibrations on the ill-fated British Midland flight. The fault was, in fact, in the port engine which, having appeared to correct itself after the initial alert, then failed to respond with disastrous consequences when extra thrust was needed as the plane came in to land at East Midlands airport.

Two key questions remain, as yet unanswered. First: why did the pilots shut down a working engine? Second: why when there was a major fault on the port engine, was a warning not seen during 20 minutes of flying after the starboard engine had been switched off?

In the event of an emergency is the increasing prevalence of computerisation lulling pilots into a false sense of security ? Are they less able to make the transition from relative inactivity to full alert? And are they sufficiently well-trained on “glass cockpits” to respond to the ultimate emergency, a total computer systems failure such as that which occurred repeatedly on one holiday flight into London during the bad storms in February?

Senior principal psychologist the RAF Institute of Aviation Medicine Roger Green, who has been involved in the Kegworth inquiry, recently expressed serious reservations about “glass cockpits”. He told the Royal Aeronautical Society: “Modern pilot training methods and digital computer cockpits are distancing the pilot from his aircraft and his

environment. The industry has tried to reduce all behaviour to rule-based rather than knowledge-based action, so pilots are in danger of not being able to handle any situation which requires knowledge or skills."

The design of "glass cockpits" is intended to supply pilots with the best possible information, clearly displayed and easy to interpret, on computer screens. In the event of an emergency or a sudden need for information, the pilot can refer to a computer checklist and call up the relevant details on screen within seconds. But there is concern, some confirmed by information given by investigators at the Kegworth inquest, that certain modern cockpit design features could prove more difficult to interpret, particularly in an emergency.

British aviation officials and pilots believe the following questions should be considered urgently if the lessons of Kegworth and other incidents are to be learned.

- Are the new style primary and secondary flight displays – such as the vibration level gauge – with cursors on the outside edge of smaller dials, as striking and as visible as the larger old electromechanical instruments?

- Should the displays relating to the two engines be more clearly separated, rather than contained next to each other in the same panels?

- Should these displays be positioned well away from the engine throttles to avoid possible wrong association of, for instance, left engine throttle with a right engine dial?

- Could a pilot's vision of a small cursor, perhaps indicating a problem, be impaired if that cursor was the same colour as other digital displays close to it? The yellow vibration cursor on the Kegworh 737-400, when showing maximum reading, would have been directly below a bright yellow digital oil reading .

NUCLEAR REACTORS.

(From K. Fitzgerald : “Shoreham in repose”, *IEEE Spectrum* **27#5**, 46 (May, 1990).)

Shoreham is the name of a nuclear power station in New York state, U.S.A. It has the distinction of being “the first completed U.S. nuclear power plant to close before operating”.

The focus of Shoreham’s control room is a corner section of instrument panel (1) containing the reactor power controls and other nuclear instrumentation. The panel to the right (2) contains the instruments for operating the turbine generators, which produce electricity from the steam from Shoreham’s boiling-water reactor. Also found in this section are the valves and pumps for the steam/water thermal cycle, the plant’s electric system, and the diesel generators that must take over in the event of a primary power outage. Panel (3) contains the controls for emergency shutdown systems, including the core spray system and the low-pressure coolant injection systems. To the right (4) are controls for the high-pressure coolant injection system and auxiliary systems that manage both normal and emergency operations. Stations in the center (5) hold computer consoles that monitor and manage the operation of the plant. The remaining section (6) controls auxiliary ventilating and service water systems.

(A) The central feature of Shoreham’s control room is the panel for controlling reactor power, mainly through positioning of the control rods. These are interspersed among the fuel rods in the configuration evident on both horizontal and vertical panels. The operator selects a group of control rods for positioning by pushing the corresponding button on the horizontal layout. Colored lights on the vertical panel indicate which control rods have been selected and their current positions – either in or out of the fuel core. Two switches at lower right control motors that move the selected rods either in, to reduce power, or out, to increase it.

The rectangular groups of multicolored square indicators at the top, called annunciator panels, alert the operator to unsafe conditions. When a dark red light flashes, an alarm sounds, warning that a safety system or the reactor itself has been shut down. Other colors signify less serious warnings. Shoreham's control room was the first to tilt the indicators so that the operator can easily read them.

(B) Redundant red scram buttons, a pair on either side of the control rod drive panel, let the operator shut down the reactor manually. By rotating the button's outer collar, the operator arms the reactor logic for shutdown, and a white annunciator shines red when the logic has been enabled. The operator then pushes the button to begin the shutdown – an automatic sequence of control rod insertion.

(C) This set of gauges monitors the level of feedwater in the reactor. The operator sets the desired level by turning the black vertical thumbwheel to adjust the leftmost instrument's vertical gauge, which reads out in inches. Comparing the desired to the actual water level obtained from a sensor in the reactor, a control circuit generates a signal proportional to the water flow required to achieve the desired level . This signal reads out on the horizontal gauge as a percentage of the circuit's maximum output voltage (or the maximum water flow). The two instruments at right each control the speed of a turbine that pumps feedwater to the reactor. By pushing the small round buttons at lower left on each instrument, the operator can put the system under automatic feedback control, whereupon the signals from the left instrument feed directly into the speed control circuits of the two turbines. During reactor startup, when water flow must be controlled manually, the operator pushes the button at right on each of the two rightmost instruments to light up the red indicator, and then pushes the black square buttons at the corner of each horizontal gauge to raise or lower turbine speed. This control scheme is employed throughout the control room, not only for water level, but also for temperature and pressure.

(D) This section of the control room operates the steam turbine generators, which take the steam that has been generated in the boiling-water reactor and convert it into electricity. Shoreham's control room designers, who had experience with fossil fuel plants, made extensive use of what is known as "mimicking, " in which controls and indicators are inserted in a flow diagram (center) to give the operators a better feel for how specific actions would affect overall plant operation.

The following three photos highlight examples of generic controls that are applied for various uses throughout the control room:

(E) To operate the turbines, the reactor steam must reach a certain pressure level. The gauge at far left reads out the pressure set point – the minimum pressure level at which valves open to allow steam into the turbines – which the operator adjusts by pushing the square buttons below. The adjacent gauge on the same instrument reads out the actual reactor pressure from a Bourdon tube sensor, a bent metal tube that changes shape – and consequently electric output – in response to changes in pressure within it. The instrument at center is a redundant unit installed for reliability. The square pushbuttons at right adjust the warming of the turbine shell by controlling steam pressure within it, and the right-most vertical gauge gives a reading of the percentage of steam valve opening.

(F) After passing through the turbines, the steam generated in the reactor is condensed, purified and reheated, and then returned to the reactor as feedwater. The vertical gauge at top reads out the water level in the feedwater heater, and the red light above it turns on when the level gets too high, at which point the water could back up and impair turbine operation. The operator can either close the inlet valve to the heat exchanger by pushing the black "CLOSE" button at left or open the feedwater drain valve by pushing the "OPEN" button on the lower switch set. When the green light is on, the valve is fully closed; the red means it is fully open. By pushing the red "STOP" button, the operator can halt the closing or opening of the exchanger steam inlet valve to allow limited flow. In the flow diagram color scheme, the red line feeding into the valve switch signifies steam, while the black line running in line with the gauge represents feedwater.

(G) The steam line valves are very large and have to withstand high pressures and so are motor-operated. The operator actuates them by twisting a pistol grip control open or closed, as shown above for the main steam line drain valve. As is true for all valve controls, the green light means the valve is fully closed, while the red light means it is fully open. The blue light goes out when the valve motor is thermally overloaded, and the valve is rendered inoperable.

GETTING TOO CONFIDENT ? – MORE AIRCRAFT.

(From J. Race : "Computer-encouraged pilot error", *Computer Bulletin Series IV* **2#6**, 13 (August, 1990).)

Since aviation is, for computer systems professionals, a preview of the sort of intelligent systems which will soon become commonplace in less exotic applications such as stock control and medical records, it makes sense for us to see if there are any lessons which we can learn from a number of recent accidents, or near-accidents, that have happened to aircraft with an intelligent onboard computer in the control loop: that is, the pilot commands the aircraft to climb, turn, or accelerate, but a computer listens to these commands, and may modify or even ignore them in order to optimise performance, or to preserve the airframe and engines from stresses outside their design limits.

In this article it is claimed that these new, very competent computers, even when working properly, may encourage pilot-error (or more generally, user-error). A solution is proposed: intelligent computer systems must literally say what they are doing.

Intelligent control systems undoubtedly, on balance, improve economics and safety. Yet we have all read reports such as the account of the Airbus fly-by-wire A320 which crashed into trees at the Habsheim air show in June 1988[1], Boeing 747s whose throttles suddenly close in normal cruise[2], 'glass cockpit' instruments which fail during excessive turbulence[3] or may be misinterpreted[4], and so on. In many cases and in particular the Habsheim crash, where the pilot flew low and slow over a runway and failed to gain sufficient height and speed to climb away before running into trees, it appears that the pilot, not the onboard computer, made a mistake.

Nevertheless, it has been argued that the presence of a highly intelligent computer system in the pilot's control loop may well have led him into a trap, because he did not take full account of the power – or the limitations – of the computer's interventions.

A model of the full, multi-dimensional 'safe performance envelope' is available to the onboard computer in the form of tables and formulae which the computer continuously consults – see Figure 1. The computer flies the aircraft to that point which is closest to the wishes of the pilot, as expressed by his commands (control column deflections etc). but the point must stay within the envelope.

The graph indicates that the aircraft can achieve its maximum speed only at altitude, and its minimum speed (stall) is lowest near the ground. This two-dimensional envelope is, of course, an over-simplification: the aircraft has many more degrees of freedom than just altitude and speed: for example, if the yoke is pulled violently backwards even at a safe altitude and speed, the wing main spar might fail – ie the wings will come off. So the computer will pass on only sensible pilot commands to the actuators that move the controls, and even when sensible, may amend them slightly (for example, to achieve maximum power as soon as possible, the pilot slams the throttles wide open, but the computer may advance them only progressively, to be kinder to the engines).

The computer also knows it is unsafe to try to fly underground. If it believes it is landing at a suitable airport and is in control, it will 'round out' above the runway: but otherwise it will warn the pilot of ground proximity – a warning the pilot can, however, ignore or disable.

Software and hardware errors are another story. But what happens in such systems even when the hardware and software are working well? Unfortunately it seems that a false sense of security may be induced which may lead to ‘pilot error’: what might even be termed ‘computer-encouraged’ pilot error.

Our pilot – or a stock controller, or a family doctor – gets used to the helpful monitoring of a computer system. They have the luxury of taking the aircraft down to minimum speed, or commanding zero widgets, or prescribing maximum dosage of a drug, secure in the knowledge that the computer will not let the aircraft stall, will maintain a minimum safety stock of widgets, will prevent the administration of ten grams rather than ten milligrams of interferon to a baby. It’s as if they have a very senior colleague at their elbow all the time. Unfortunately situations arise in which the safe performance envelope no longer protects the aircraft (or business, or patient). The Habsheim accident is a good example.

Too late

The pilot wished to make use of the A320’s computer to make, in safety, a very low, slow, nose-high approach with undercarriage down, along the line of the runway at a little airfield called Habsheim. Normally, of course, the A320 will only be flying in such a configuration when about to land at a commercial airport, with a runway long enough to brake on, and no obstacles at the far end to prevent a ‘go-around’ if necessary. Habsheim was not such an airport, and had trees at the far end.

Although the computer will prevent the pilot from stalling the aircraft, its ground proximity warnings can be ignored or disabled (as in a normal landing), and this is what the pilot arranged. He finally commanded full power, but too late. He complained later that the computer seemed slow to obey him, but as we have seen, it may have been gentler with the power change than he, in the circumstances, would have liked, and in any case it takes eight seconds to reach full power from idle. He commanded a climb, and the computer did the best it could, but they hit the trees, which the computer knew nothing about until it sensed the compressor blades ingesting branches – see Figure 2.

Of course there are fighters with terrain-following radar and computers that could have coped with the situation, warned the pilot, or even opened up the power without the pilot’s intervention – even against his wishes – when it was apparent to the system that the alternative was a crash in twenty seconds. In fact if Habsheim had been a commercial airport, the computer could have been loaded with its obstacle clearance limits, and initiated a ‘go-around’ automatically when a safe landing was obviously too late, even without radar to see the trees. But the A320 had been placed in a situation, permitted by the logic and formulae of its computer, from which there was no safe escape. The pilot had no business to do what it seems he did, but it seems likely that he was so used to the

system keeping the aircraft safe that he felt it could wave a magic wand and get him out of any problem.

This is the paradox. The better our systems protect our clients, the more likely it is that when a situation occurs outside the system's competence – no doubt, I have to admit, as a result of a questionable decision on their part to overrule it – the client will make a hash of things, still half-believing their now gagged 'senior colleague' will still reach over if necessary and say, 'no, no, don't do that', or make last-minute corrections which will save the day. Maybe we should not permit them to be disabled – but that places a load on us to produce infallible systems that have total world knowledge, and few of us are confident enough to promise that. But the coward's way out – of saying, leave the final responsibility to the human, is not a good option either: a verdict of 'pilot error' only exonerates the computer at one level. At another, our helpful system was indeed to blame, for encouraging over-reliance on its power to retrieve a bad situation.

What we need to do is to think for a moment how groups of human aircrew, surgeons and lawyers get to understand each other's competence, areas of expertise, blind spots: their present appreciation of the situation, what they would do if X happened, and so on. They do this simply by talking – about the present situation and if the present situation is, as usual, fairly boring – about interesting cases, past and hypothetical. Of course there are often some fixed rules determining who does what in an emergency, but the informal procedure helps the team anticipate trouble and allocate contributions optimally in the many cases not covered by the book. It is interesting to read transcriptions of cockpit voice recorders: the crew are talking, conferring, checking, giving views, analogies, proposing actions ... if they get it right, the transcript does not end, 'sound of impact' .

What we need to add to intelligent decision support systems is the ability to chat. As a matter of fact, Hewitt and Winograd as long ago as the early 70s showed an intelligent robot system which would answer questions about why it was doing things, but by means of a teletype. Since surgeons and pilots need to keep their eyes and hands free, by 'chat' we must mean real-time voice input/output, a technology which is currently at long last coming to fruition.

As a fortunate by-product, if expert computer systems have the ability to give an account of their actions and capabilities (future actions), they are accountable: if they do the job of a human who, in the event of an incident, would have been held responsible, they, too, can be held responsible, because they will be able to respond. At present there is an anomaly: past physical states of an aircraft can be replayed from the flight recorder, and the crew's decisions and reasoning recovered from the voice recorder and face-to-face debriefs. But there is no requirement for a computer to explain in detail why it suddenly closed the throttles – yet.

References

1 'After Habsheim', *Flight*, April 11-17, 1990.
2 'Throttle hitch hits 747-400', same issue.
3 'Pilot hotline reveals graphic account of computer failure', *Independent*, April 28, 1990
4 'Kegworth 737 report', *Flight*, May 2-8, 1990.

Alan Creak,
March, 1997.