GHOST: experimenting conflicts countermeasures in the pilot's activity

Frédéric Dehais, Catherine Tessier and Laurent Chaudron
Onera-DCSD
31055 Toulouse cedex 04 - France
{dehais,tessier,chaudron}@cert.fr

Abstract

An approach for designing countermeasures to cure conflict in aircraft pilots' activities is presented, both based on Artificial Intelligence and Human Factors concepts.
The first step is to track the pilot's activity, i.e. to reconstruct what he has actually done thanks to the flight parameters and reference models describing the mission and procedures. The second step is to detect conflict in the pilot's activity, and this is linked to what really matters to the achievement of the mission. The third step is to design accurate countermeasures which are likely to do better than the existing onboard devices. The three steps are presented and supported by experimental results obtained from private and professional pilots.

1 Introduction

The review of civilian and military reports reveals that a conflictual situation is a precursor to the loss of aircrews' situation awareness and is a major cause of air accidents. Conflict may occur through different forms: open conflict between the different operators (Air Philippines crash, April 1999), representation conflict about the aircraft position (Korean air, August 1997), resource conflict (Streamline, May 2000), knowledge conflict (Crossair, January 2000), or more recently, a conflict between automated systems (TCAS) and human operators (air collision between a Tupolev 154 and a Boeing 757 over Switzerland, July 2002). Therefore, the idea is to detect conflict so as to propose accurate on-line countermeasures to pilots.
The first step is to track the pilot's activity, i.e. to reconstruct what he has actually done thanks to the flight parameters and reference models describing the mission and procedures. The second step is to detect conflict in the pilot's activity, and this is linked to what really matters to the achievement of the mission. The third step is to design accurate countermeasures which are likely to do better than the existing onboard devices. The three steps are presented and supported by experimental results obtained both from private and military pilots.

2 Track the pilot's activity

2.1 Reference model

The approach to capture the pilot's activity is based on situation tracking classically used for system diagnosis, monitoring and intelligence [Ghallab, 1996; Chaudron *et al.*, 1997; Rota and Thonnat, 2000]. Whatever the formalism used to represent knowledge, situation tracking rests on a matching between observed facts and known situations (reference models), and its object is to carry out postdictions, predictions or to diagnose the current state of the system.
To represent knowledge, we use the concept of *propositional attitude*, *"which covers the notions of belief, desire, intention, [...] which have in common the fact of being identified by their propositional contents"* [Tessier *et al.*, 2000].
<u>Definition</u>: a propositional attitude (PA) is a pair (K, T) where K is a logical cube, i.e. a conjunction of first order literals [Chaudron *et al.*, 2003], and T is the time interval within which the properties captured by K are true.
The pilot's knowledge is divided into three categories of PAs, which constitute the reference model:
• Propositional attitude *goal* describes a goal of the mission.
Ex: `pa_goal(goal(5),<Landing(francazal)>,T)`
means that *the fifth goal of the mission is to land at Francazal during time interval T.*
• Propositional attitude *plan* describes the properties the pilot is committed to satisfy to reach a PA-goal as long as he carries out this goal.
Ex: `pa_plan(goal(5),<VOR_frequency(116), VOR_heading_deflection(true),gear_down(true), flaps_down(2),visibility(true)>,T)`
means that *to satisfy goal 5, the pilot has to select the accurate radio frequency, fly a correct route, lower gears and flap deflection 2 and have a good visibility during time interval T.*
• A *crucial* propositional attitude describes the hard constraints associated to one or several goals.
Ex: `pa_crucial(goal(5),<visibility(true)>,T)`
means that to succeed in landing at Francazal, the pilot must have a correct visibility during T.
<u>Remark</u>: the model does not implement deontic logic [Traum and Allen, 1994] to capture the pilot's obligations; at each time step, the system will check whether the pilot meets the different constraints on the PA-goals.

2.2 Observed activity

The pilot's activity is observed and analyzed through his actions on the airplane interface: during the experiments, the flight parameters are recorded every second. The parameters are then translated into symbolic knowledge and aggregated to build *observation propositional attitudes*.

The numerical thresholds that are used for numerical to symbolic translation are adjusted during interviews with the pilots to built an accurate model for each pilot. For example, if the pilot has to fly 2500 ft, he defines his own flight envelope where he feels secure [Amalberti and Wioland, 1997]. After numerical to symbolic translation, parameter altitude is characterized as altitude(false) *(i.e. out of flight envelope)* or altitude(true) *(i.e. within flight envelope)*.

The same kind of analysis is performed with parameters such as the route, the fuel level, etc.

Therefore, propositional attitude *observation* describes perceived and deduced facts from the pilot's activity:

Ex: pa_obs(<VOR_frequency(116), VOR_heading_deflection(true),gear_down(true), flaps_down(3),visibility(false)>,<1500,1500>) means that *at time 1500, the pilot has selected the accurate radio frequency, is flying a correct route, has lowered gears and flap deflection 3 and has a bad visibility*.

3 Detect conflict

3.1 Conflict

Considering research in the field of A.I. and particularly on multi-agent systems, the focus has been brought much on how to avoid, solve or get rid of conflicts, especially *via* negotiation [Sycara, 1989; Rosenschein and Zlotkin, 1994], argumentation [Jung and Tambe, 2000] or constraint satisfaction [Conry *et al.*, 1991; Khedro and Genesereth, 1994; Hannebauer, 2000]. Such approaches, where (i) conflict is likened to formal inconsistency and (ii) the internal knowledge of the agents is clearly identified, are not suited to our purpose. Indeed in aeronautics, uncertainty on data is high and the human behavior is extremely variable and unpredictable. Moreover, while in air, aircrews frequently face incoherent situations due to the uncertain environment (e.g. weather, failures) or because of their own errors. Nevertheless, these situations are not necessarily conflictual, and pilots are used to going on with their missions with such inconsistencies. An inconsistency becomes conflictual only if it leads to a critical flight phase and *matters* to the mission or to the aircrew's safety.

Therefore we are very close to Easterbrook's definition of conflict [Easterbrook, 1991]: "*[...] viewpoints are free to differ, and only conflict when that difference matters for some reason, leading to interference. [...] A conflict, then, is simply a difference that matters.*" This point of view has been emphasized in social sciences [Festinger, 1957; Lewin *et al.*, 1939] and in A.I. applied to social modeling [Castelfranchi, 2000], where human conflict is seen as the result of an impossibility to succeed in a goal that matters.

Definition: a *conflict* is a set of propositional attitudes that is not coherent[1] and the fact that it is not coherent matters.

The inconsistency of a set of PAs *matters* either because there is a need:

- to satisfy the rationality[2] of the agent(s) involved (pilot, copilot ...);
- or to build a coherent set of knowledge;
- or to decide on further goals.

All these different reasons are implemented in the reference model as *crucial* propositional attitudes (see section 2.1).

Therefore a *conflict* is an incoherent set of knowledge that matters through one or several *crucial propositional attitudes*.

3.2 Conflict detection

Conflict detection is a tracking process that monitors the pilot's PAs at each time step t. It is a two-step process: (1) detect the sets of PAs that are not coherent at time t and (2) detect the conflicts given the crucial PAs that concern time t. One could have designed a single-step process, however it is of great interest to track PAs that are not coherent and do not conflict: indeed, contrary to the minimal inconsistent sets approach [De Kleer, 1986], we aim at capturing all the inconsistencies and analyzing how they are managed by the pilots for "lessons learned" and further man-artefact interaction model tuning.

(1) Detect the sets of PAs that are not coherent

Predicate not_hold_together detects incoherent sets of PAs Inc_t at time t.

Ex: given the set of PAs

- pa_goal(goal(5),<Landing(francazal)>,<1000,t1>)
- pa_plan(goal(5),<VOR_frequency(116), VOR_heading_deflection(true),gear_down(true), flaps_down(2),visibility(true)>,<1000,t1>)
- pa_obs(<VOR_frequency(116), VOR_heading_deflection(true),gear_down(true), flaps_down(3),visibility(false)>,<1500,1500>),

two incoherent sets of PAs are detected:

→ not_hold_together(pa_plan(5,flaps_down(2)), pa_obs(flaps_down(3),<1500,1500>);

→ not_hold_together(pa_plan(5,visibility(true), pa_obs(visibility(false)),<1500,1500>).

(2) Detect the conflicts

The crucial PAs that concern time t are then considered.

Definition: let Inc_t an incoherent set of PAs at time t. If Inc_t includes PAs or properties that matter through one or several crucial PAs, then Inc_t is a conflict.

Predicate matters screens sets Inc_t thanks to the crucial PAs, so as to determine whether they are conflicts or not.

Ex: for the previous example, the crucial PA that holds at time 1500 is:

- pa_crucial(goal(5),<visibility(true)>, <1000,t1>)

Predicate matters detects that one of the two incoherent

[1] in relation to the formalism used to represent the properties in the PAs.

[2] According to Festinger [Festinger, 1957] who postulates that an individual always aims at being coherent when interacting with his environment.

sets of PAs is conflictual:
`matters(pa_plan(goal(5),visibility(true)), pa_obs(visibility(false),<1500,1500>)`
The other set of PAs relative to flaps deflection is not a conflict because a landing with flap deflection 3 is possible even though less recommended.

3.3 Preliminary experiment and results

A preliminary experiment designed to test conflict tracking [Dehais, 2002] was conducted with ten pilots on a research flight simulator. The pilots were proposed a flight scenario whose aim was to follow a flight plan at a constant altitude and then land with instruments. The difficult point in this scenario was to manage the fuel consumption correctly considering refuelling areas.
This experiment has validated conflict detection and interestingly enough has shown that a psychological *perseveration* phenomenon appeared with conflict : the pilots who faced conflictual situations did not come to the right decision to solve their conflict, on the contrary they got themselves tangled up in it, and all the more so since the temporal pressure was increasing. The most surprising thing was that all the necessary data to come to the accurate decision were available and displayed on the onboard equipments.
Ex: one of the pilots flew toward waypoint 5 instead of flying first over waypoint 6 for refuelling, and started circling around, until he ran out of fuel (see figure 1).

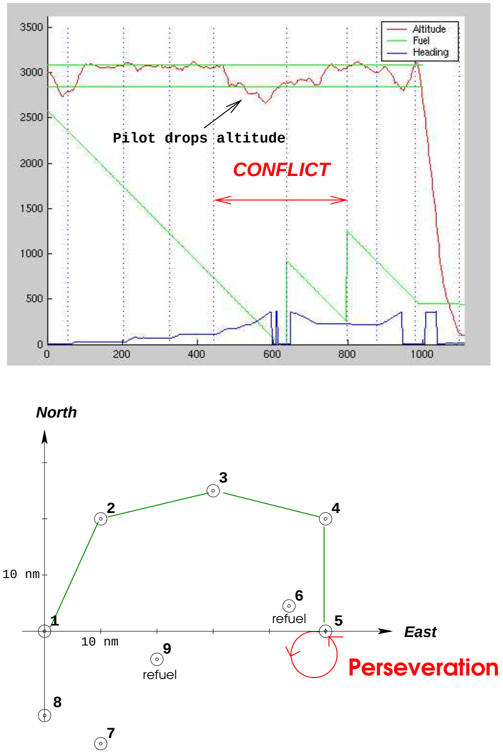

Figure 1: Conflict and perseveration

During the debriefing, the pilot explained that he was convinced that waypoint 5 was the refuelling area (in spite of the map displayed in the cockpit); he reached it and noticed that the fuel level did not increase. Thinking he had missed waypoint 5, he decided to fly back to waypoint 5. He did this manoeuvre again and again till he ran out of fuel, getting more and more stressed.

4 Send accurate countermeasures

4.1 Guidelines

The findings of the preliminary experiments are akin to a recently published report of the BEA[3] (the French national institute for air accident analysis) that reveals that pilots' erroneous attitudes of perseveration have been responsible for more than 40 percent of casualties in air crash (in civilian aeronautics). This behavior, called the "perseveration syndrome", is studied in neuropsychology [Vand Der Kolk, 1994; Pastor, 2000] and psychology [Beauvois and Joule, 1999]: it is known to summon up all the pilot's mental efforts toward a unique objective (excessive focus on a single display or focus of the pilot's reasoning on a single task). Once entangled in perseveration, the pilot does anything to succeed in his objective even if it is dangerous in terms of security, and worse, he neglects any kind of information that could question his reasoning (like alarms or data on displays).

Therefore the idea is to design countermeasures in order to break this mechanism and get the pilot out of the conflict. The design of the countermeasures is grounded on the following theoretical and empirical results:
• conflictual situations lead pilots to persevere;
• additional information (alarms, data on displays) designed to warn pilots are often unnoticed when pilots persevere.
Our conjectures are then:
→ with an accurate on-line conflict detection, it is possible to analyze and predict the subject of the pilot's perseveration;
→ instead of adding information (classical alarms), it is more efficient to *remove the information* on which the pilot is focused and which makes him persevere, and to display a message to explain his error instead.
Ex: in the experiment described afterwards, pilots persevere in trying a dangerous landing at Francazal despite the bad visibility, with a particular focus on an instrument called the H.S.I.[4] to locate Francazal airport. The countermeasures will consist in removing the H.S.I. during a few seconds, to display two short messages instead, one after the other ("*Landing is dangerous*"... "*look at central display*") and next to send an explanation of the conflict on the central display ("*Low visibility on Francazal, fly back to Blagnac*").
The idea here is to shock the pilot's attentional mechanisms with the short disappearance of the H.S.I., to introduce a cognitive conflict ("*if I land, I crash*") to affect the pilot's reasoning, and to propose a solution on the central display.

[3] http://www.bea-fr.org
[4] The H.S.I is a display that gives the route to follow and indicates any discrepancy from the selected route.

4.2 GHOST

Ghost is an experimental environment designed to test countermeasures to cure the pilot's perseveration. It is composed of:
• Flightgear[5], an open source flight simulator, which means that many modifications can be made, e.g. implementing new displays. To fly the airplane, pilots have a stick, rudder and a keyboard. Almost all the airports and beacons (NDB and VOR[6]) of the world are modeled, and the air traffic control is simulated. During the experiment, the flight simulator interface is displayed on a giant screen;
• Atlas[7], a freeware designed to follow the pilot's route. The airplane trajectory, cities, airports and beacons are displayed with an adjustable focus;
• a wizard of Oz interface (see figure2) we have designed and implemented, which allows a human operator to trigger events (failures, weather alteration and the countermeasures) from an external computer *via* a local connection (TCP/IP protocol, sockets communication).
As far as the countermeasures are concerned, several actions are available to the wizard of Oz:
• *replace* a display on which the pilot is focused by a black one (time of reappearance is adjustable);
• *blink* a display (frequency and time of blinking is adjustable);
• *fade* a display and *brighten* it again;
• *send a message* to a blinked, removed or faded display;
• *send a message* to the central display for explicit conflict solving.

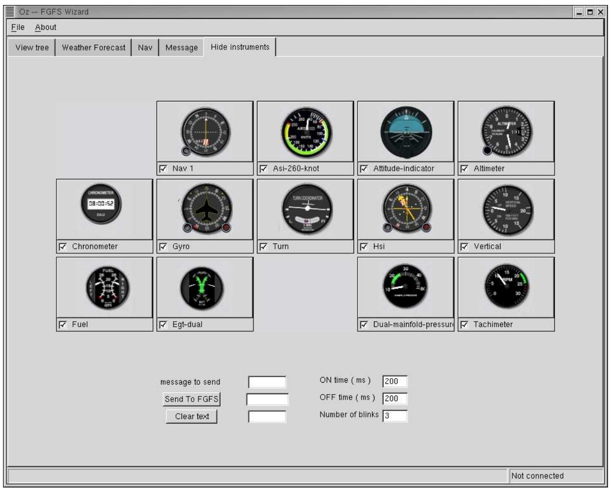

Figure 2: The wizard of Oz interface - countermeasures

4.3 Experimental scenarios

As conflict appears when a desired goal cannot be reached, we have designed three experimental scenarios where a crucial goal of the mission cannot be achieved:
• scenario 1 is a navigation task from Toulouse-Blagnac airport to Francazal airport including three waypoints (VOR 117.70, NDB 331 and NDB 423). An atmospheric depression is positioned in such a way that the pilot cannot see the landing ground but at the last moment when it is too late to land on it.
• scenario 2 is a navigation task from Toulouse-Blagnac airport to Francazal airport including three waypoints (VOR 117.70, NDB 415 and NDB 423). The visibility is decreased from the runway threshold: from far away, the landing ground is visible but as the pilot gets closer to it, it disappears totally in a thick fog.
• scenario 3 is a navigation from Toulouse-Blagnac back to Toulouse-Blagnac including three waypoints (VOR 117.70, NDB 331 and NDB 423). An atmospheric depression is positioned over waypoint 2 (NDB 331), and as the pilot flies over this waypoint, the left engine fails.
In scenarios 1 and 2 the pilot's conflict can be summarized as "*should I try a dangerous landing at Francazal or should I fly back to Blagnac for a safer landing?*". If our conjectures hold, pilots will persevere and try to land at Francazal.
In scenario 3, the pilot's conflict can be summarized as "*should I go on with the mission despite the failure or should I land at Francazal and therefore abort the mission?*"

4.4 Experimental Results

21 experiments were conducted with Ghost in December 2002 (see figure 3). The pilots' experiences ranged from novice (5 hours, small aircraft) to very experienced (3500 hours, military aircraft). Conflicts where detected with the conflict tracker (described above) and were inputs for the wizard of Oz to elaborate the countermeasures.

Results for scenarios 1 and 2: "impossible landing"

In these scenarios, the pilots faced the decision of mission abortion, i.e. not to land at Francazal and fly back to Blagnac. Both scenarios were tested within two different contexts: without countermeasures and with countermeasures.
• Context 1: no countermeasures
7 pilots tested scenario 1 or 2 without any countermeasures (see next table). "Circuits" corresponds to the number of circuits performed by the pilot round Francazal before crashing or landing.

Pilot	Scenario	Circuits	Results
Pilot1	1	3	crash
Pilot2	1	3	crash
Pilot3	1	1	chance landing
Pilot4	1	1	chance landing
Pilot5	2	1	crash
Pilot6	2	1	chance landing
Pilot7	2	1	crash

The results suggest that without any countermeasures, none of the pilots came to the right decision (fly back to Blagnac): they all persevered at trying to land at Francazal. Four of them hit the ground, and the other three had a "chance landing", which means that while they were flying round Francazal, the runway appeared between two fog banks and

[5] http://www.flightgear.org/
[6] NDB and VOR are two kind of radionavigation beacons used to guide pilots.
[7] http://atlas.sf.net

they succeeded in achieving a quick landing. During the debriefing, all of them admitted they had come to an erroneous and dangerous decision.
• Context 2: with countermeasures
12 pilots tested scenario 1 or 2 with countermeasures (see next table). "Circuits" corresponds to the number of circuits performed by the pilot round Francazal before a countermeasures is triggered by the wizard of Oz.

Pilot	Scenario	Circuits	Results
Pilot8	1	3	crash on Francazal
Pilot9	1	2	back to Blagnac
Pilot10	1	2	back to Blagnac
Pilot11	1	2	back to Blagnac
Pilot12	2	3	back to Blagnac
Pilot13	2	2	back to Blagnac
Pilot14	2	2	back to Blagnac
Pilot15	2	2	back to Blagnac
Pilot16	1	2	chance landing Francazal
Pilot17	1	2	back to Blagnac
Pilot18	1	2	back to Blagnac
Pilot19	1	5	crash on Francazal

The results show the efficiency of the countermeasures to cure perseveration: 9 pilots out of 12 changed their minds thanks to the countermeasures, and flew back safely to Blagnac. During the debriefing, all the pilots confirmed that the countermeasures were immediately responsible for their change of mind. Moreover, the short disappearance of the data due to the countermeasures did not cause any stress to them. 4 military pilots found that the solutions proposed by the countermeasures were close to what a human co-pilot would have proposed.
The results with Pilot19 suggest that the more a pilot perseveres, the more difficult it is to get him out of perseveration. During the debriefing, Pilot19 told us that he was obsessed by the idea of landing, and that he became more and more stressed as long as he was flying round Francazal. He then declared that he did not notice any countermeasures. Pilot16 and Pilot8 also persevered despite the countermeasures: they declared they knew they were not flying properly and that they had done that on purpose because they wanted to test the flight simulator.

Results for scenario 3: "failure"

Only 2 pilots tested scenario 3. One pilot experimented this scenario without any countermeasures: he did not notice the failure and hit the ground. The second pilot was warned through the countermeasures that he had a failure and that there was a landing ground at vector 80: the pilot immediately performed an emergency landing on this landing ground.
Other experiments are currently being conducted with this scenario.

Other results

During the experiments, 9 pilots made errors, e.g. selection of a wrong radio frequency, erroneous altitude, omission to retract flaps and landing gear. To warn them, the wizard of Oz blinked the display on which they were focusing and displayed the error (e.g.: "gear still down"). In each case, the countermeasure was successful: the pilots performed the correct action at once. During the debriefing, the pilots declared that this kind of alarm was very interesting, and much more efficient and stressless than a classical audio or visual alarm because they could identify at once what the problem was.

These experiments have shown also the significance of the message contents on the blinked display. For example, as far as the erroneous landing at Francazal is concerned, the initial message was "Don't land", but the pilots did not take it into account, thinking it was a bug of the simulator. When the message was changed to "Fly back to Blagnac" or "Immediate overshoot" it was understood and taken into account at once.

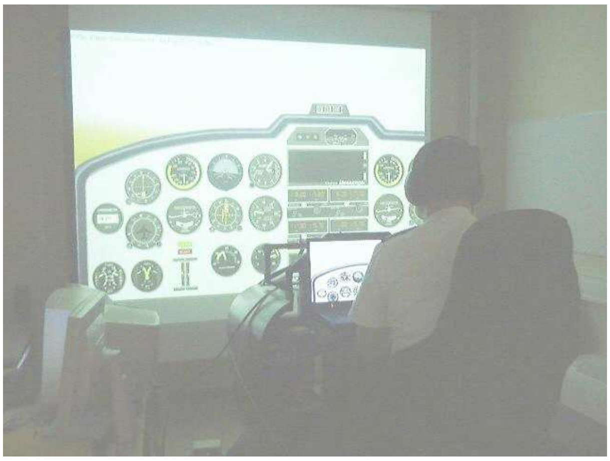

Figure 3: GHOST

5 Conclusion and perspectives

We have presented an approach to detect and cure conflicts in aircraft pilots' activities. A first experiment has validated the conflict detection tool and has shown that pilots facing conflicts have a trend to persevere in erroneous behaviors. This agrees with the observations made by experts of the pilots' behaviors in real air accidents. These first results have led us to design GHOST, an experimental environment dedicated to test countermeasures designed to get pilots out of perseveration. A second experiment was conducted with twenty-one pilots in this new experimental environment: it has proved the efficiency of the countermeasures to cure perseveration and has also confirmed the relevance of conflict detection. Further experiments are to be conducted to go on tuning the countermeasures according to the pilots' feedback.

The next step is to design and implement the whole countermeasures closed-loop, i.e to remove the wizard of Oz and perform an automated conflict and perseveration management. This is currently done thanks to (see figure 4):
• the conflict detection tool (see section 3).
• Kalmansymbo(aéro), a situation tracker and predictor [Tessier, 2003] which is based on predicate/transition timed Petri nets for procedure and pilot's activity modeling. Kalmansymbo assesses the current situation thanks to the pa-

rameters received in real-time from Flightgear and predicts what is likely to happen. Both the current situation and the predictions are inputs for the conflict detection tool, which allows possible conflicts to be better anticipated.

• a tool for building and sending accurate countermeasures back to the pilot *via* the Flightgear cockpit.

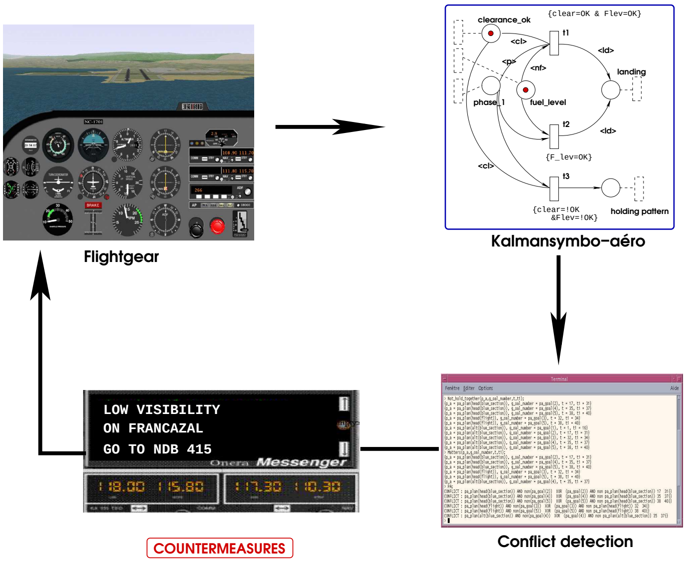

Figure 4: The countermeasures closed-loop

Acknowledgments

Many thanks to Olivier Grisel and Fabrice Schwach for their wonderful work on Flightgear and the wizard of Oz.

References

[Amalberti and Wioland, 1997] R. Amalberti and L. Wioland. Human error in aviation. In *Aviation safety conference*, Rotterdam, August 1997.

[Beauvois and Joule, 1999] J.L. Beauvois and R.V. Joule. A radical point of view on dissonance theory. In Eddie Harmon-Jones and Judson Mills, editors, *Cognitive Dissonance : Progress on a Pivotal Theory in Social Psychology*. 1999.

[Castelfranchi, 2000] C. Castelfranchi. Conflict ontology. In H.-J. Müller and R. Dieng, editors, *Computational conflicts - Conflict modelling for distributed intelligent systems*, pages 21–40. Springer Verlag, 2000.

[Chaudron *et al.*, 1997] L. Chaudron, C. Cossart, N. Maille, and C. Tessier. A purely symbolic model for dynamic scene interpretation. *International Journal on Artificial Intelligence Tools*, 6(4):635–664, 1997.

[Chaudron *et al.*, 2003] L. Chaudron, N. Maille, and M. Boyer. The CUBE lattice model and its applications. *Applied Artificial Intelligence*, 17(3):207–242, March 2003.

[Conry *et al.*, 1991] S.E. Conry, K. Kuwabara, V.R. Lesser, and R.A. Meyer. Multistage negotiation for distributed constraint satisfaction. *IEEE Transactions on SMC*, 21(6):1462–1477, 1991.

[De Kleer, 1986] J. De Kleer. An assumption-based truth maintenance system. *Artificial intelligence*, 28, 1986.

[Dehais, 2002] F. Dehais. Modelling cognitive conflict in pilot's activity. In *STAIRS 2002: Starting Artificial Intelligence Researchers Symposium*, pages 45–54, July 2002.

[Easterbrook, 1991] S. M. Easterbrook. Handling conflict between domain descriptions with computer supported negotiation. *Knowledge Acquisition: An International Journal*, 3(4):255–289, 1991.

[Festinger, 1957] L. Festinger. *A theory of cognitive dissonance*. CA : Stanford University Press, 1957.

[Ghallab, 1996] M. Ghallab. On chronicles: representation, on-line recognition and learning. In L.C. Aiello, J. Doyle, and S. Shapiro, editors, *Proceedings of the Fifth National Conference on Artificial Intelligence*, San Francisco, California, 1996.

[Hannebauer, 2000] M. Hannebauer. Their problems are my problems, the transition between internal and external conflicts. In C. Tessier, L. Chaudron, and H.-J. Müller, editors, *Conflicting agents - Conflict management in multi-agent systems*, Multiagent systems, artificial societies and simulated organizations 1, pages 63–109. Kluwer Academic Publishers, 2000.

[Jung and Tambe, 2000] H. Jung and M. Tambe. Conflicts in agent teams. In C. Tessier, L. Chaudron, and H.-J. Müller, editors, *Conflicting agents - Conflict management in multi-agent systems*, Multiagent systems, artificial societies and simulated organizations 1, pages 153–167. Kluwer Academic Publishers, 2000.

[Khedro and Genesereth, 1994] T. Khedro and M. Genesereth. Modeling multiagent cooperation as distributed constraint satisfaction problem solving. In *11th ECAI*, pages 249–253, 1994.

[Lewin *et al.*, 1939] K. Lewin, R. Lippitt, and R. K. White. Patterns of aggression behavior in experimentally created "social climates". *Journal of Social Psychology*, 10:271–299, 1939.

[Pastor, 2000] J. Pastor. Cognitive performance modeling - Assessing reasoning with the EARTH methodology. In *COOP'2000*, Sophia Antipolis, May 2000.

[Rosenschein and Zlotkin, 1994] J. S. Rosenschein and G. Zlotkin. *Rules of encounter. Designing convention for automated negotiation among computers*. Artificial Intelligence. MIT Press, 1994.

[Rota and Thonnat, 2000] N. Rota and M. Thonnat. Activity recognition from video sequences using declarative models. In *ECAI'00*, pages 673–677, Berlin, 2000.

[Sycara, 1989] K. Sycara. Multiagent compromise via negotiation. In Les Gasser and Michael N. Huhns, editors, *Distributed Artificial Intelligence*, volume 2, pages 119–137. Pitman Publishing, 1989.

[Tessier *et al.*, 2000] C. Tessier, L. Chaudron, H. Fiorino, and H.-J. Müller. Agents' conflicts: new issues. In C. Tessier, L. Chaudron, and H.-J. Müller, editors, *Conflicting agents - Conflict management in multi-agent systems*, Multiagent systems, artificial societies and simulated organizations 1, pages 1–30. Kluwer Academic Publishers, 2000.

[Tessier, 2003] C. Tessier. Towards a commonsense estimator for activity tracking. In *AAAI Spring symposium*, Stanford University, Palo Alto CA, USA, March 2003.

[Traum and Allen, 1994] D.R. Traum and J.F. Allen. Discourse obligations in dialogue processing. In James Pustejovsky, editor, *Proceedings of the Thirty-Second Meeting of the Association for Computational Linguistics*, pages 1–8, San Francisco, 1994. Morgan Kaufmann.

[Vand Der Kolk, 1994] B. Vand Der Kolk. The body keeps the score: memory and the evolving psychobiology of post traumatic stress disorder. *Harvard Review of Psychiatry*, 1:253–265, 1994.