Continuous Safety Risk Evaluation by Example

Peter H Feiler

Software Engineering Institute
Carnegie Mellon University
phf@sei.cmu.edu

DM19-1050

This note demonstrates continuous Safety Risk Evaluation of an aircraft using the SAE International standard Architecture Analysis & Description Language (AADL) for embedded software systems and the safety analysis capability of the Open Source AADL Tool Environment (OSATE). Any similarity of the example use cases to real aircraft incidents is purely accidental.

The objective is exercise is

- To illustrate the importance of including lower Design Assurance Level (DAL) components in a safety risk analysis,
- To include the pilot's role in the overall system safety analysis,
- To perform safety risk analysis throughout the product life cycle – even when aircraft are already in operation.

We proceed by first describing the elements of the model and then discussing several use scenarios for which we assess the safety risk by performing a fault tree analysis that calculates the probability of a catastrophic aircraft incident.

The Model

The model consists of the following elements:

- Angle of Attack (AoA) sensors with implementations coming from different vendors with different quality characteristics. They are the primary sensors used in the operation of ACC.
- An Auxiliary Climb Control (ACC) system: the system of interest. It was considered DAL C with the pilot acting as the backup if the ACC fails. The ACC can be configured to operate with one AoA sensor, or two AoA sensors. When operating with two sensors the ACC goes into standby mode if the two sensor outputs differ.
- An AoA discrepancy detector to indicate to the pilot if the two AoA sensors differ in their output. The AoA discrepancy detector is only used in the aircraft configuration with a single AoA sensor ACC. In the 2 sensor ACC configurations the ACC standby indicator reflects discrepancies in AoA sensors.
- A flight surface that is controlled by the ACC or the pilot.

- Engines as DAL A components: They are highly critical components and are used as context to assess the effect of ACC on the overall aircraft safety risk.
- The pilot as backup to the ACC.

The model is organized as four AADL packages:

- An EMV2 error type library
- An AOADiscrepancy package that contains the above elements and various aircraft configurations dealing with the single AOA sensor operation of the ACC
- An AOAVendors package that contains variants of the AOA from different vendors and various aircraft configurations that use vendor specific AOAs.
- An ACCAOARedundancy package that contains a variant of ACC operating with two AOA sensors and various aircraft configurations utilizing it.

The model is a high-level model to cover the targeted use scenarios. It can be elaborated to take into account knowledge about time sensitive behavior of the system and its components.

Figure 1 shows a configuration of the system with one AoA sensor feeding the ACC, and both the AoA discrepancy indicator taking input from both AoA sensors. The ACC affects the flight surface. The pilot can observe the discrepancy indicator. In a two or three sensor configuration of the ACC the pilot also gets an indication from the ACC when it goes into standby mode. The effect of the flight surface is combined with the effects of the two engines to get a sense of the impact of the ACC in the context of high criticality components.

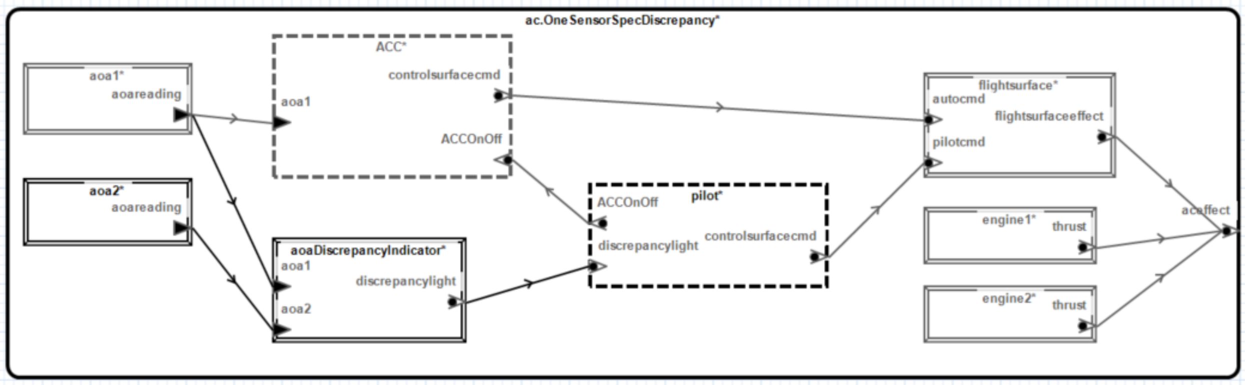


Figure 1: Single Sensor ACC Operation with AoA Discrepancy Indicator

Figure 2 shows a configuration of the system with ACC using redundant AoA input. The pilot is informed of the ACC going into ACC fail mode, either due to AoA failure or bad input, or due to ACC failure.

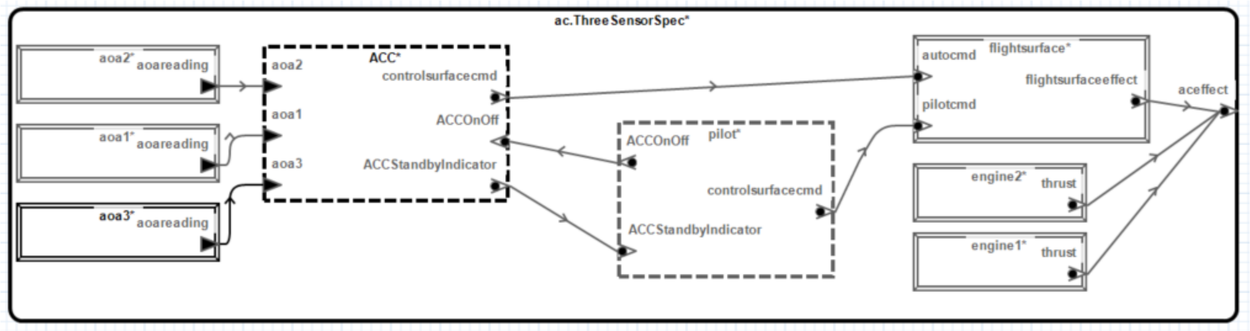


Figure 2: Triple Sensor ACC Operation with ACC Standby Indicator

In the following sections we elaborate on the elements of aircraft system and its configurations.

AoA Sensor

The AADL model has a specification of the Angle of Attack sensor as a device. Its outgoing data port is annotated with ServiceOmission and BadValue as error sources. Service omission represents the sensor failing to produce any output. Bad value represents the sensor producing incorrect values. The occurrence probability is set at 1.0e-5 to reflect DAL C. In addition, we have two implementations of the sensor, one from vendor A[1] and one from vendor B. With this we reflect that AOA sensors from two vendors “appear to have a greater susceptibility to adverse environmental conditions” than sensors from a third vendor[2].

AoA Discrepancy Detector

The discrepancy detector informs the pilot if two AoA sensors produce different output by performing an approximate value comparison. Approximate value comparison is used to filter out minor variation in values.

The AADL specification of the AoA discrepancy detector indicates that a Service Omission is propagated through the discrepancy indicator feature, either when the discrepancy detector fails, or when both sensor outputs have value errors but the values are close to each other such a subtle value error is not detected. Failure occurrence probabilities are associated with both.

There is a variant of the discrepancy detector specification to represent the situation that the detector is configured incorrectly and therefore does not provide a discrepancy indication.

Auxiliary Climb Control (ACC)

The ACC is modeled to accept input from up to two AoA sensors. It controls the flight surface and it provides a standby indicator to the pilot. Furthermore, the pilot can turn the ACC off and on. In order to capture the fault interaction between the ACC and the pilot, we did not have to reflect the nominal ACC on/off behavior through modes.

[1] Typical failure rate of 1 in 70,000 flight hours or 1.5e-5. https://www.bloomberg.com/news/articles/2019-04-11/sensors-linked-to-737-crashes-vulnerable-to-failure-data-show

[2] https://www.heraldnet.com/nation-world/not-just-the-737-angle-of-attack-sensors-have-had-problems/

The ACC specification has two variants:

- Single AoA sensor input: The ACC provides bad control commands when the AoA sensor provides bad values or fails to provide values. The ACC does not relinquish control to the pilot (propagation of *ContinuousBadControl)*. The pilot is required to turn off the ACC to gain control, i.e., *ContinuousBadControl* propagation continues if ACC is not switched off.
- Two AoA sensor input: Here we model operational and standby as explicit error states. If the sensor outputs differ (either different values or one of them fails to provide values) the ACC goes into standby. The standby indicator may fail, which results in a service omission of not reporting standby to the pilot. We also reflect the situation when the two AoA sensor inputs have a subtle value error, i.e., their values are incorrect but close to each other.

The Pilot

The pilot is modeled as an abstract component with input from the AOA discrepancy indicator or the ACC standby indicator and being able to control the flight surface as well as turn off the ACC.

We have several variants of the pilot specification:

- Nominal: The pilot has visual feedback from the AOA discrepancy detector. When the detector fails to provide an indication, the pilot has to deduce from the flight behavior of the aircraft that the ACC provides bad control commands. He takes control by turning off the ACC. Turning off is necessary due to the continuous bad control behavior of the ACC. While in control the pilot may also make flight control mistakes.
- NoACCKnowledge: Here we model that the pilot does not have knowledge of the ACC, therefore, does not turn off the ACC to take control.

The pilot can make mistakes in turning off the ACC when malfunctioning and in controlling the flight surface. The rate at which those mistakes occur changes depending on whether the pilot has knowledge of the ACC, deals with a non-functioning or functioning AOA discrepancy detector

The Flight Surface

The flight surface receives control input from ACC or the pilot. It keeps track of who is in charge of sending control commands through an error state machine.

While ACC is in control, continuous bad control results in an incident effect. While the pilot is in control lack of pilot control commands results in an incident effect.

Control switches from ACC to the pilot when *ServiceOmission* is propagated from the ACC.

As with other components of the aircraft, the flight surface may also encounter a mechanical failure which prevents it to respond to control commands.

The Aircraft Engine

The engine is modeled as a device producing thrust. It has the potential of failing to produce thrust. We chose a failure rate of 1 in 650,000 flight hours[3].

[3] https://en.wikipedia.org/wiki/Turbine_engine_failure and https://en.wikipedia.org/wiki/Pratt_%26_Whitney_Canada_PT6

The Aircraft System

The aircraft consists of two engines, the flight surface, the AoA sensors, the ACC in various configurations, and the pilot. The aircraft as a whole has an incident if both engines fail, or if the flight surface has an incident effect due to ACC or pilot issues.

The aircraft has a number of configurations to support the different use scenarios. Several single AoA sensor ACC configurations include the AoA discrepancy detector and vary to reflect no ACC knowledge, permanent discrepancy detector failure. We also have a two AoA sensor ACC configuration as well as configurations reflecting AoA sensors being sourced from different vendors.

The Use Scenarios

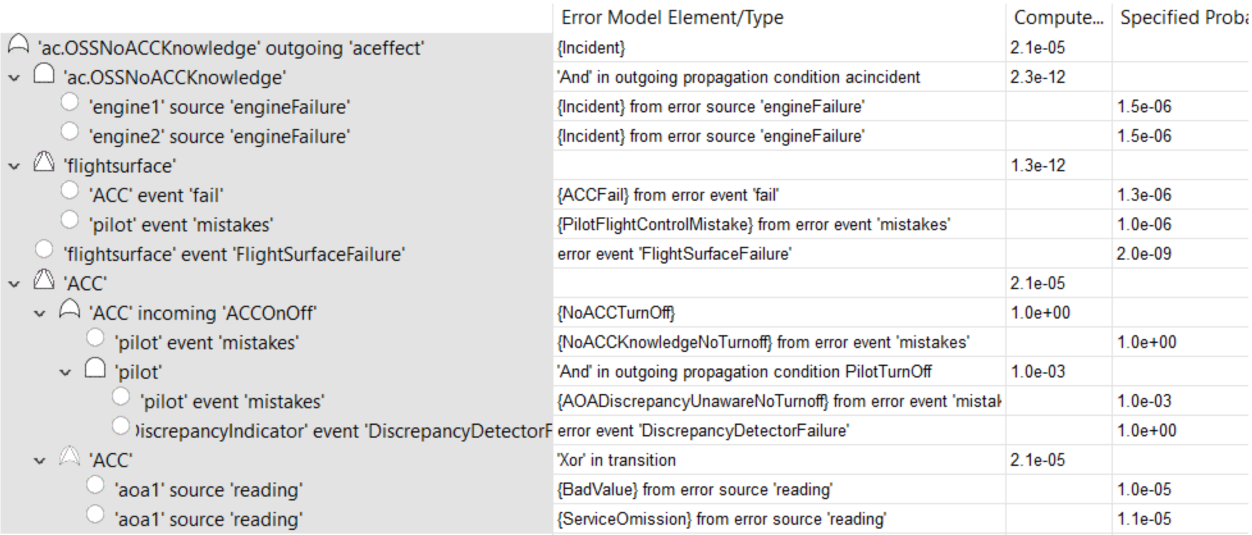
First, we focus scenarios with the aircraft configuration of ACC operating with a single AoA sensor. Then we consider a two sensors configuration. When comparing the safety risk as determined by a probabilistic FTA between a single AoA sensor ACC and two AoA sensor configuration we show that customers can be informed about the quantitative safety risk reduction they receive when expending the additional cost of a two sensors configuration. We also can consider the effect of the AoA supplied by vendor A vs vendor B on the overall safety risk.

Scenario A: Single Sensor ACC Operation Retaining Control with No Pilot ACC Knowledge and Non-working AoA Discrepancy Detector

In the initial scenario the pilot is not aware of the ACC. Furthermore, the discrepancy indicator is not operational. The ACC is configured to operate with a single AoA sensor input. The AoA has the failure occurrence probability derived from the DAL (i.e., the spec value). Finally, the ACC has the problem of repeatedly regaining control from the pilot.

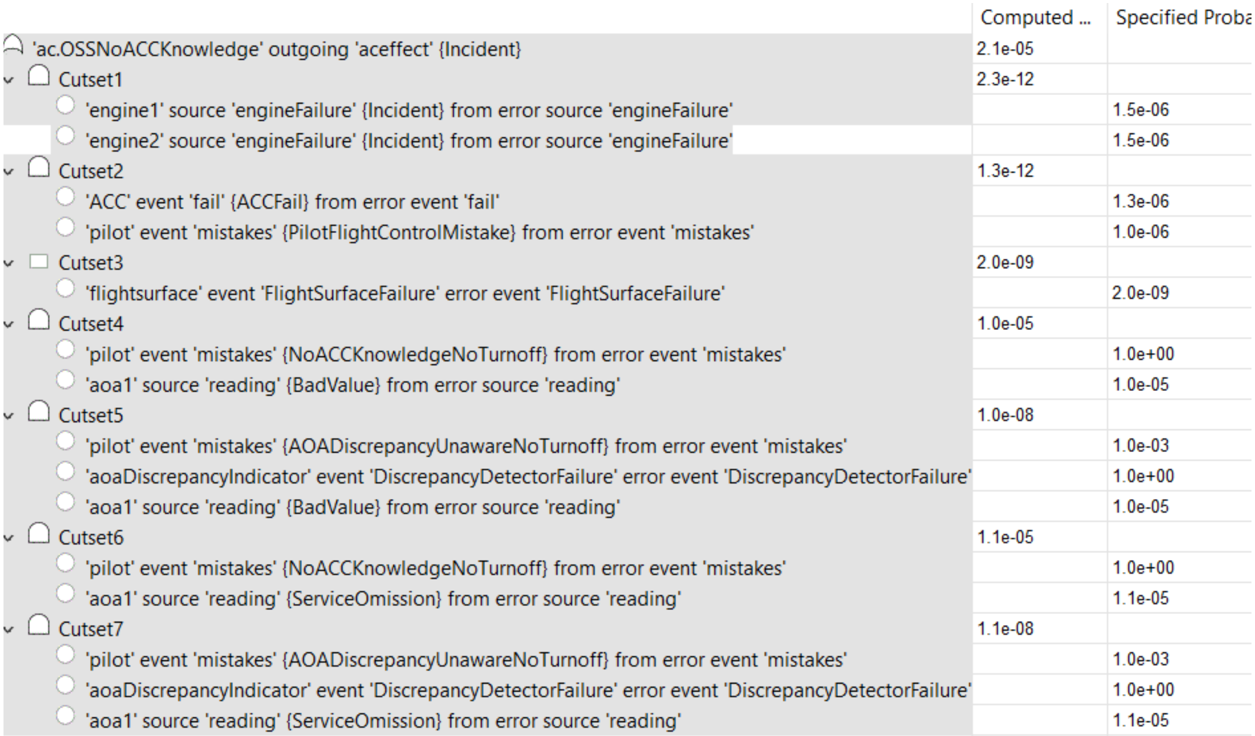
We analyze this configuration by instantiating the aircraft system implementation ac.OSSNoACCKnowledge and running the fault tree analysis (FTA). The result of the FTA is shown in Figure 3. The failure probability is shown as 2.1e-5. This figure is primarily determined by the combination of the ACC and pilot control with the pilot not being aware of the ACC, thus, not disabling it. Disabling the ACC is necessary to overcome it repeatedly regaining control. Other aircraft components contribute a much lower failure probability to the overall aircraft incident probability. Due to their redundancy the two engines come in at 2.3e-12, the flight surface at the specified 2.0e-9, pilot mistakes in flight control when the ACC fails to operate at a rate of 1.0e-3. Between the ACC and the pilot as backup the rate is 2.1e-5, which drives the aircraft failure rate. The ACC contribution is primarily determined by the failure rate of the AoA sensor (2.1e-5).

Figure 4 shows the same computed occurrence probability results, when calculating the minimal cut set for the same aircraft configuration as used for Figure 3. The minimal cut set result also shows the key role of the pilot as backup to the ACC and the impact of lack of knowledge and operational insight.



	Error Model Element/Type	Compute...	Specified Probi
'ac.OSSNoACCKnowledge' outgoing 'aceffect'	{Incident}	2.1e-05	
'ac.OSSNoACCKnowledge'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		2.1e-05	
'ACC' incoming 'ACCOnOff'	{NoACCTurnOff}	1.0e+00	
'pilot' event 'mistakes'	{NoACCKnowledgeNoTurnoff} from error event 'mistakes'		1.0e+00
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mistak		1.0e-03
)iscrepancyIndicator' event 'DiscrepancyDetectorF	error event 'DiscrepancyDetectorFailure'		1.0e+00
'ACC'	'Xor' in transition	2.1e-05	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		1.0e-05
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		1.1e-05

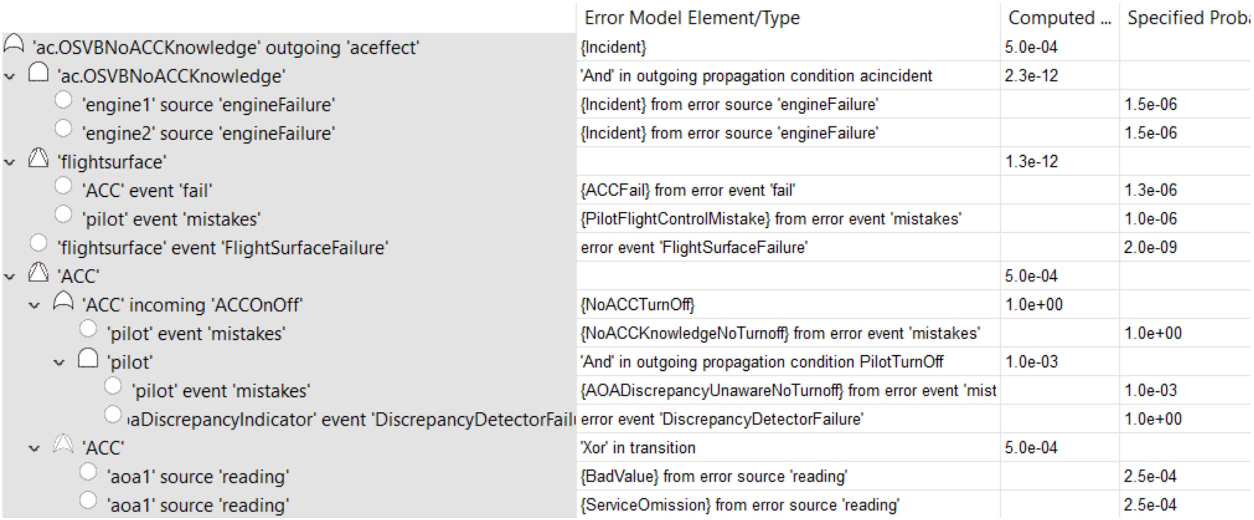
Figure 3: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, no Pilot ACC Knowledge, and non-working AoA Discrepancy Indicator.



	Computed ...	Specified Proba
'ac.OSSNoACCKnowledge' outgoing 'aceffect' {Incident}	2.1e-05	
Cutset1	2.3e-12	
'engine1' source 'engineFailure' {Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure' {Incident} from error source 'engineFailure'		1.5e-06
Cutset2	1.3e-12	
'ACC' event 'fail' {ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes' {PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
Cutset3	2.0e-09	
'flightsurface' event 'FlightSurfaceFailure' error event 'FlightSurfaceFailure'		2.0e-09
Cutset4	1.0e-05	
'pilot' event 'mistakes' {NoACCKnowledgeNoTurnoff} from error event 'mistakes'		1.0e+00
'aoa1' source 'reading' {BadValue} from error source 'reading'		1.0e-05
Cutset5	1.0e-08	
'pilot' event 'mistakes' {AOADiscrepancyUnawareNoTurnoff} from error event 'mistakes'		1.0e-03
'aoaDiscrepancyIndicator' event 'DiscrepancyDetectorFailure' error event 'DiscrepancyDetectorFailure'		1.0e+00
'aoa1' source 'reading' {BadValue} from error source 'reading'		1.0e-05
Cutset6	1.1e-05	
'pilot' event 'mistakes' {NoACCKnowledgeNoTurnoff} from error event 'mistakes'		1.0e+00
'aoa1' source 'reading' {ServiceOmission} from error source 'reading'		1.1e-05
Cutset7	1.1e-08	
'pilot' event 'mistakes' {AOADiscrepancyUnawareNoTurnoff} from error event 'mistakes'		1.0e-03
'aoaDiscrepancyIndicator' event 'DiscrepancyDetectorFailure' error event 'DiscrepancyDetectorFailure'		1.0e+00
'aoa1' source 'reading' {ServiceOmission} from error source 'reading'		1.1e-05

Figure 4: Minimal Cut Set Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, no Pilot ACC Knowledge, and non-working AoA Discrepancy Indicator.

We also configured the above aircraft model with AoA sensors from vendor B (system implementation ac.OSVBNoACCKnowledge), whose product reliability is well below the specified value. The resulting FTA results are shown in Figure 5. The overall aircraft failure probability increases to 5.0e-4. This increase is primarily driven by the reduced quality of the AoA sensor from vendor B. In other words, a small component has significant impact on the overall safety risk of the aircraft.



	Error Model Element/Type	Computed ...	Specified Prob:
'ac.OSVBNoACCKnowledge' outgoing 'aceffect'	{Incident}	5.0e-04	
'ac.OSVBNoACCKnowledge'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		5.0e-04	
'ACC' incoming 'ACCOnOff'	{NoACCTurnOff}	1.0e+00	
'pilot' event 'mistakes'	{NoACCKnowledgeNoTurnoff} from error event 'mistakes'		1.0e+00
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mist		1.0e-03
aDiscrepancyIndicator' event 'DiscrepancyDetectorFail	error event 'DiscrepancyDetectorFailure'		1.0e+00
'ACC'	'Xor' in transition	5.0e-04	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		2.5e-04
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		2.5e-04

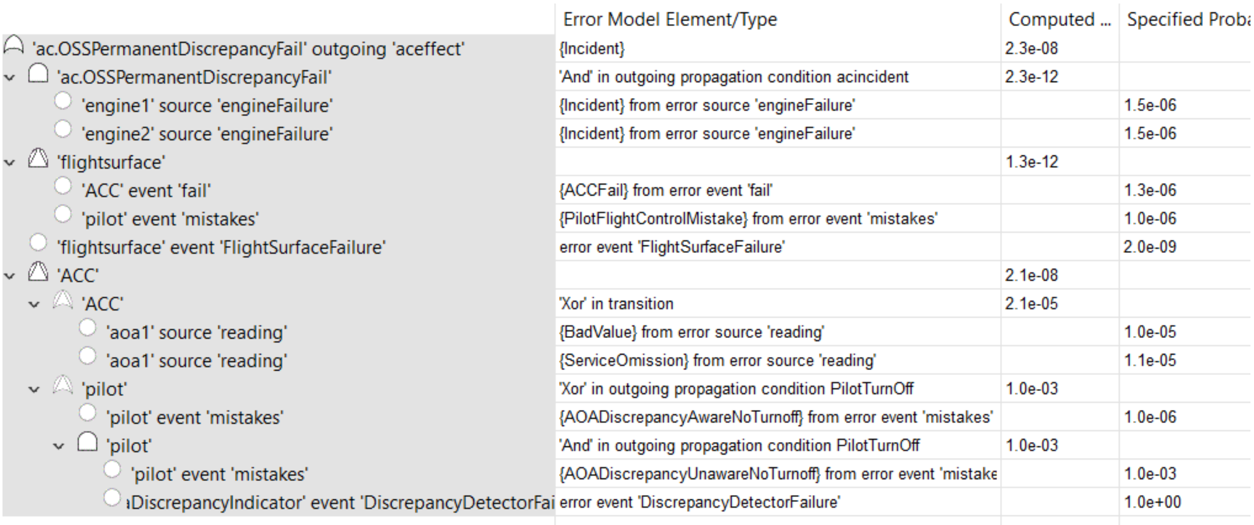
Figure 5: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, no Pilot ACC Knowledge, non-working AoA Discrepancy Indicator, and Vendor B AoA Sensors.

Scenario B: Single Sensor ACC Operation Retaining Control with Non-working AoA Discrepancy Detector

This scenario reflects the situation that the aircraft vendor informs the airlines about the existence of the ACC. The ACC still repeatedly regains control from the pilot. The discrepancy detector is still non-working, i.e., the pilot is not informed about any discrepancy between the two AoA sensors. As a result, the pilot has to infer ACC malfunction from the aircraft flight behavior and turn off the ACC.

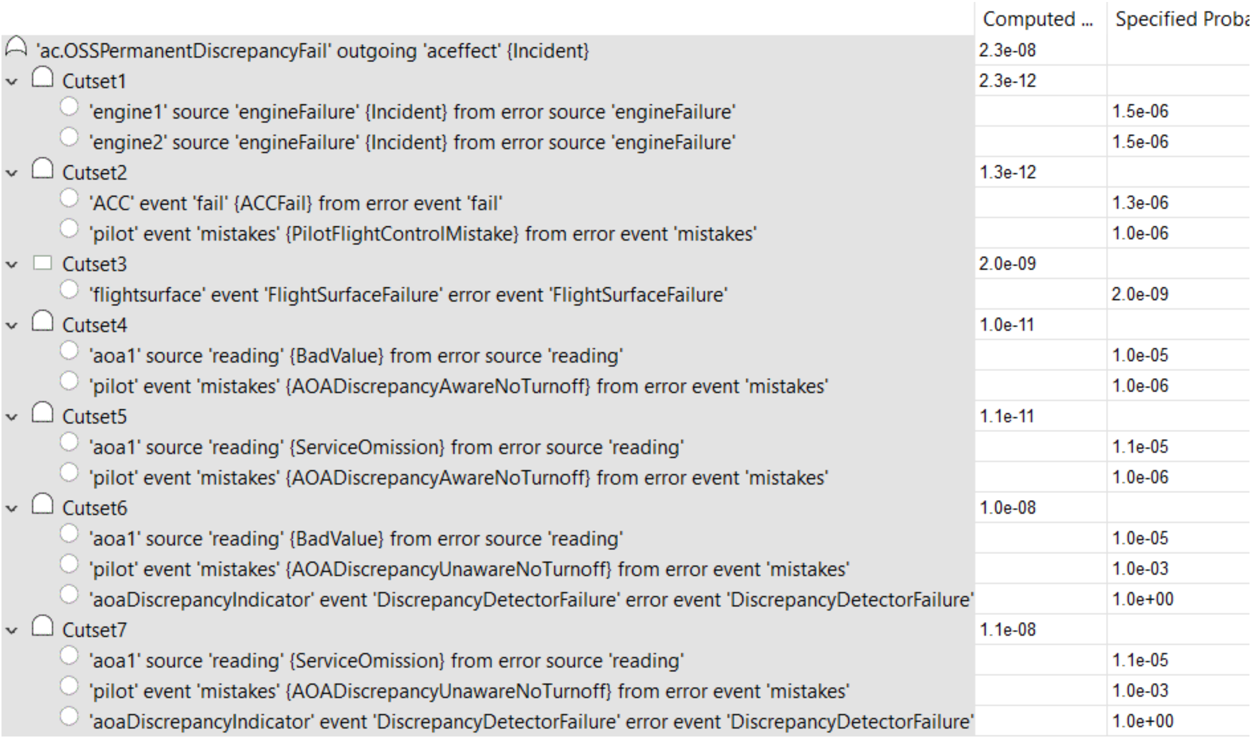
The aircraft configuration ac.OSSPermanentDiscrepancyFail represents the aircraft with non-working AoA discrepancy detector, and AOA sensors with spec-based failure rate. We can see that the pilot is still not informed about the potential misbehavior of the ACC (AOADiscrepancyDetectorFailure at a rate of 1.0). We have assigned a probability of 1.0e-3 of the pilot recognizing the AoA malfunction induced ACC misbehavior, resulting in a failure contribution of 2.1e-8. This figure primarily determines the overall aircraft failure rate. In other words, the AoA sensor malfunction strongly affects the overall aircraft safety. Figure 6 shows the FTA results.

Figure 7 shows the minimal cut set results resulting in the same overall aircraft failure rate. At this time we have reflected the risk of the pilot not recognizing the malfunction of the AoA in the occurrence probability. The AADL model can be expanded to perform latency analysis to take into account the risk of slow response times under high vs. low altitude operational scenarios.



	Error Model Element/Type	Computed ...	Specified Proba
'ac.OSSPermanentDiscrepancyFail' outgoing 'aceffect'	{Incident}	2.3e-08	
'ac.OSSPermanentDiscrepancyFail'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		2.1e-08	
'ACC'	'Xor' in transition	2.1e-05	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		1.0e-05
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		1.1e-05
'pilot'	'Xor' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mistake		1.0e-03
ıDiscrepancyIndicator' event 'DiscrepancyDetectorFai	error event 'DiscrepancyDetectorFailure'		1.0e+00

Figure 6: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, and non-working AoA Discrepancy Indicator.

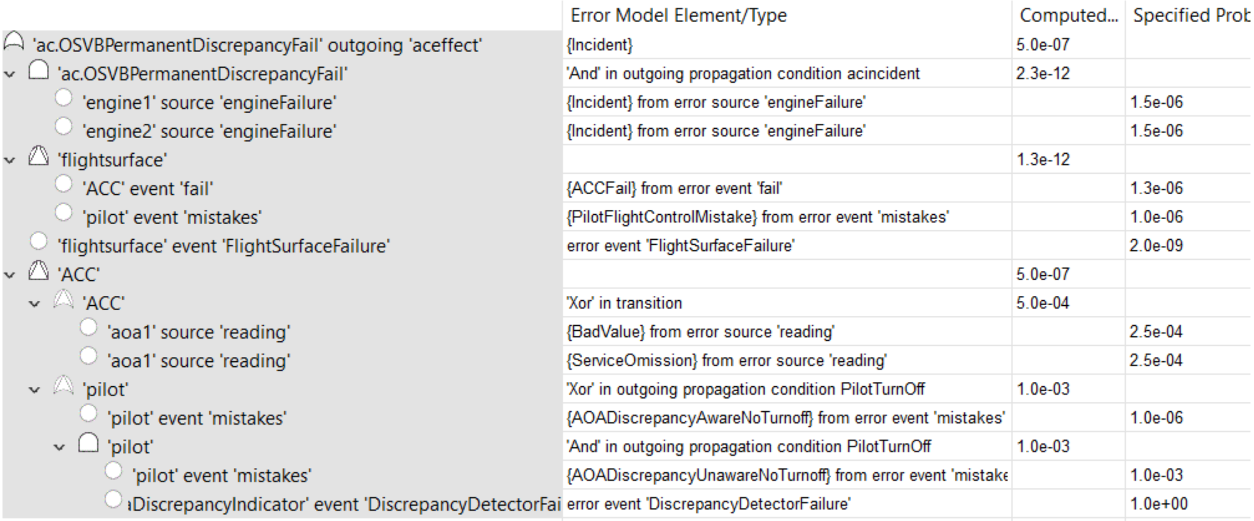


	Computed ...	Specified Proba
'ac.OSSPermanentDiscrepancyFail' outgoing 'aceffect' {Incident}	2.3e-08	
Cutset1	2.3e-12	
'engine1' source 'engineFailure' {Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure' {Incident} from error source 'engineFailure'		1.5e-06
Cutset2	1.3e-12	
'ACC' event 'fail' {ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes' {PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
Cutset3	2.0e-09	
'flightsurface' event 'FlightSurfaceFailure' error event 'FlightSurfaceFailure'		2.0e-09
Cutset4	1.0e-11	
'aoa1' source 'reading' {BadValue} from error source 'reading'		1.0e-05
'pilot' event 'mistakes' {AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
Cutset5	1.1e-11	
'aoa1' source 'reading' {ServiceOmission} from error source 'reading'		1.1e-05
'pilot' event 'mistakes' {AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
Cutset6	1.0e-08	
'aoa1' source 'reading' {BadValue} from error source 'reading'		1.0e-05
'pilot' event 'mistakes' {AOADiscrepancyUnawareNoTurnoff} from error event 'mistakes'		1.0e-03
'aoaDiscrepancyIndicator' event 'DiscrepancyDetectorFailure' error event 'DiscrepancyDetectorFailure'		1.0e+00
Cutset7	1.1e-08	
'aoa1' source 'reading' {ServiceOmission} from error source 'reading'		1.1e-05
'pilot' event 'mistakes' {AOADiscrepancyUnawareNoTurnoff} from error event 'mistakes'		1.0e-03
'aoaDiscrepancyIndicator' event 'DiscrepancyDetectorFailure' error event 'DiscrepancyDetectorFailure'		1.0e+00

Figure 7: Minimal Cut Set Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, and non-working AoA Discrepancy Indicator.

Finally, we also evaluate this aircraft configuration with the AoA sensors from vendor B (system implementation ac.OSVBPermanentDiscrepancyFailure) to determine how much of an impact the lower quality sensor has on the overall aircraft failure rate. As Figure 8 shows the overall aircraft failure rate increases to 5.0e-7 from the original rate of 2.3e-8 based on a specified AoA failure rate according to its

DAL. In other words, the quality of the AoA sensor still has a strongly influence on the safety of the aircraft.



	Error Model Element/Type	Computed...	Specified Prob
'ac.OSVBPermanentDiscrepancyFail' outgoing 'aceffect'	{Incident}	5.0e-07	
'ac.OSVBPermanentDiscrepancyFail'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		5.0e-07	
'ACC'	'Xor' in transition	5.0e-04	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		2.5e-04
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		2.5e-04
'pilot'	'Xor' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-03	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mistake		1.0e-03
ıDiscrepancyIndicator' event 'DiscrepancyDetectorFai	error event 'DiscrepancyDetectorFailure'		1.0e+00

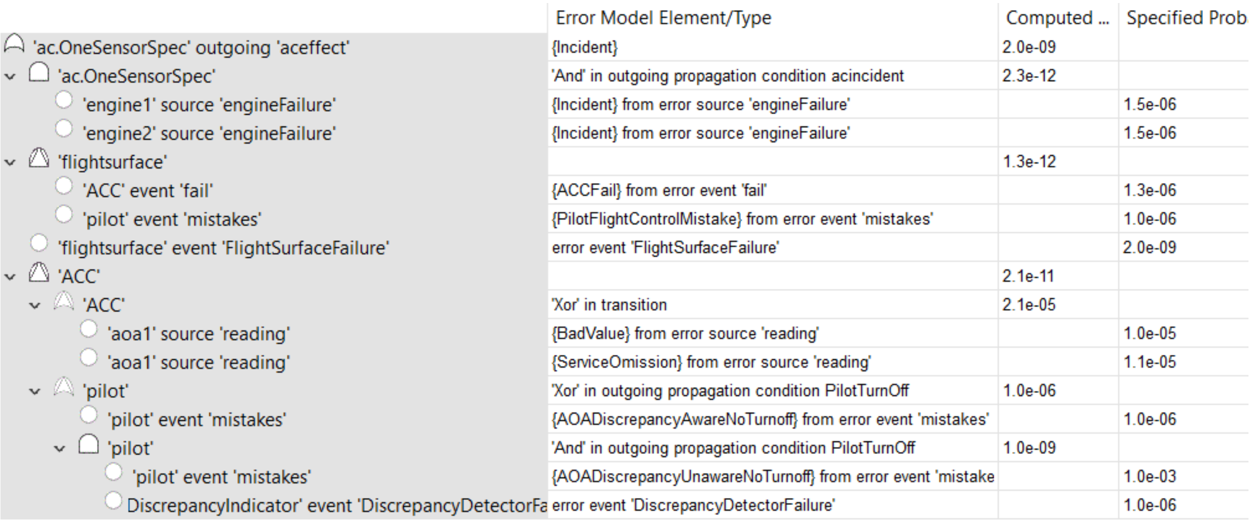
Figure 8: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, non-working AoA Discrepancy Indicator, and Vendor B AoA Sensors.

Scenario C: Working AoA Discrepancy Detector and Single Sensor ACC Operation Retaining Control

This scenario reflects the aircraft with a working AoA discrepancy detector and a single AoA sensor ACC operation that still does not relinquish control on AoA malfunction (system implementation ac.OneSensorSpec). As a result, the pilot has to turn off the ACC after inferring ACC malfunction from the AoA discrepancy indication. Note that we assume an ACC Turn Off failure rate of 1.0e-3 when the discrepancy indicator fails (same as in the previous scenario), and a lower failure rate of turning the ACC off when the discrepancy indicator works. The lower failure rate is intended to reflect the pilot not always drawing the conclusion to turn off the ACC when the discrepancy light comes on.

The results of the FTA analysis are shown in Figure 9. In this configuration we can see that the flight surface failure rate (2.0e-9) becomes the dominant contributor to the overall aircraft failure rate.

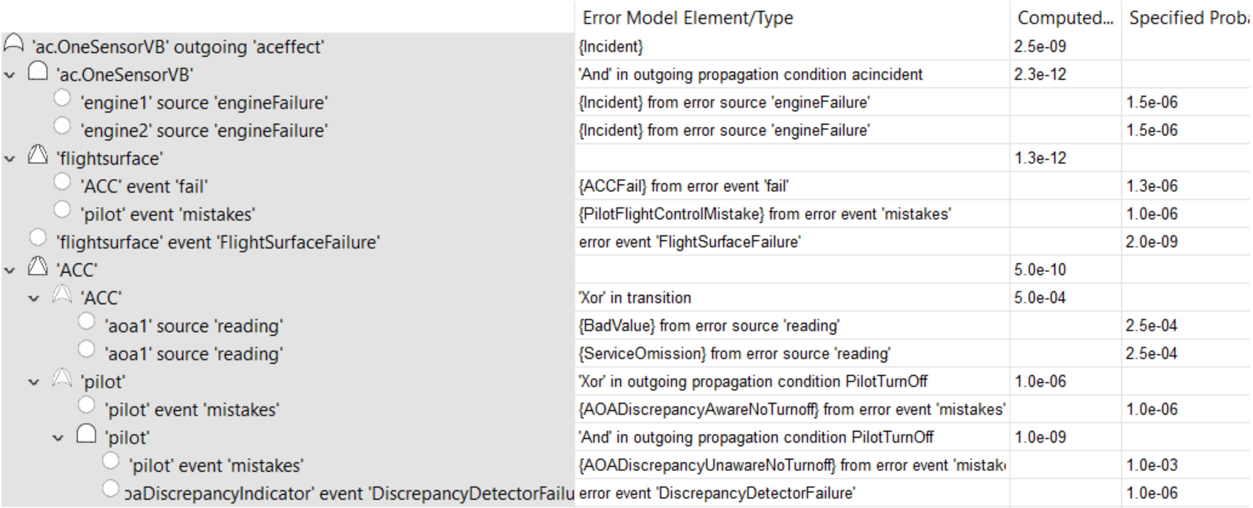
Note that we have not introduced redundancy in the flight surface operation to reduce its failure rate. If that would be done the overall failure rate would change to that of the ACC/Pilot "subsystem" (2.1e-11).



	Error Model Element/Type	Computed ...	Specified Prob
'ac.OneSensorSpec' outgoing 'aceffect'	{Incident}	2.0e-09	
'ac.OneSensorSpec'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		2.1e-11	
'ACC'	'Xor' in transition	2.1e-05	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		1.0e-05
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		1.1e-05
'pilot'	'Xor' in outgoing propagation condition PilotTurnOff	1.0e-06	
'pilot' event 'mistakes'	{AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-09	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mistake		1.0e-03
DiscrepancyIndicator' event 'DiscrepancyDetectorFa	error event 'DiscrepancyDetectorFailure'		1.0e-06

Figure 9: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, and working AoA Discrepancy Indicator.

As before, we will investigate the impact of choosing the AoA sensors from vendor B. This is done through the aircraft configuration ac.OneSensorVB. Note that the overall aircraft failure rate is affected by the AoA sensor quality from vendor B, i.e., the overall aircraft failure rate increases to 2.5e-9 (see Figure 10).



	Error Model Element/Type	Computed...	Specified Prob.
'ac.OneSensorVB' outgoing 'aceffect'	{Incident}	2.5e-09	
'ac.OneSensorVB'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.3e-12	
'ACC' event 'fail'	{ACCFail} from error event 'fail'		1.3e-06
'pilot' event 'mistakes'	{PilotFlightControlMistake} from error event 'mistakes'		1.0e-06
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09
'ACC'		5.0e-10	
'ACC'	'Xor' in transition	5.0e-04	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		2.5e-04
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		2.5e-04
'pilot'	'Xor' in outgoing propagation condition PilotTurnOff	1.0e-06	
'pilot' event 'mistakes'	{AOADiscrepancyAwareNoTurnoff} from error event 'mistakes'		1.0e-06
'pilot'	'And' in outgoing propagation condition PilotTurnOff	1.0e-09	
'pilot' event 'mistakes'	{AOADiscrepancyUnawareNoTurnoff} from error event 'mistak		1.0e-03
ɔaDiscrepancyIndicator' event 'DiscrepancyDetectorFailu	error event 'DiscrepancyDetectorFailure'		1.0e-06

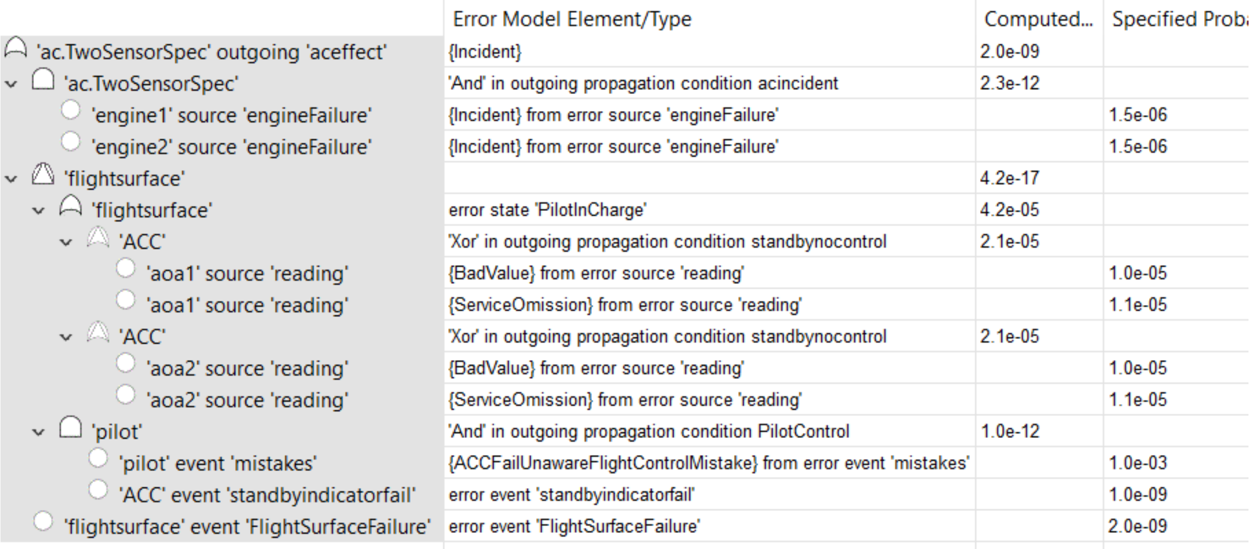
Figure 10: FTA Results of the Aircraft Configuration with Single Sensor ACA Retaining Control, working AoA Discrepancy Indicator, and Vendor B AoA Sensors.

Scenario D: Dual AoA Sensor ACC Operation

This scenario reflects the aircraft with an ACC that operates with two AoA sensors. It goes into standby when it detects AoA discrepancy and informs the pilot of this situation (system implementation ac.TwoSensorSpec). In other words, the ACC relinquishes control to the pilot.

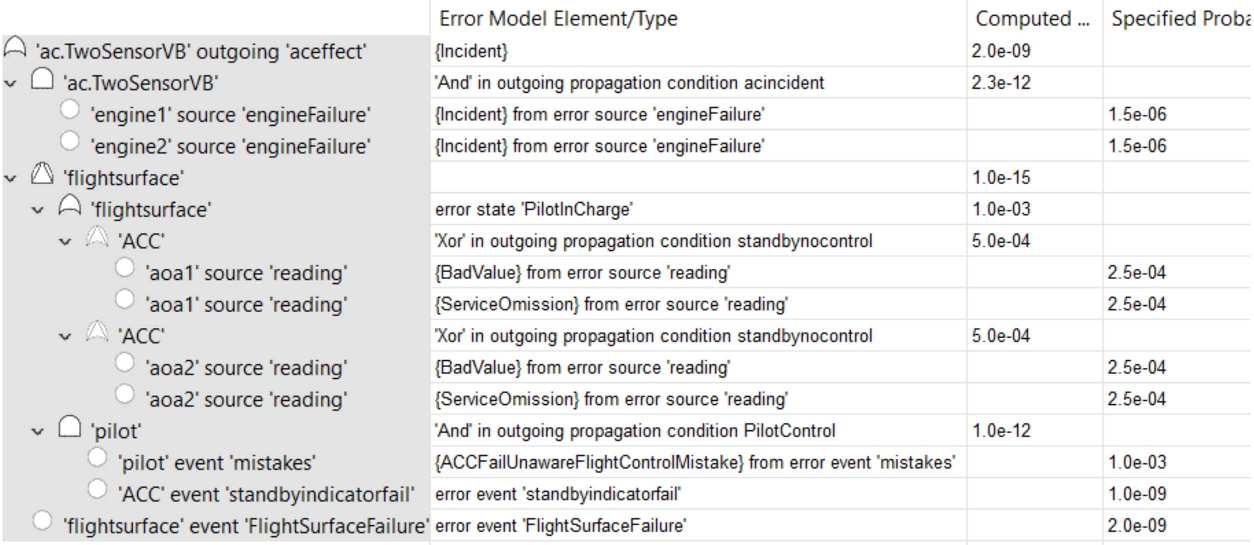
The FTA results are shown in Figure 11. The overall aircraft failure rate of 2.0e-9 is now driven by the flight surface failure rate. The ACC/Pilot climb control "subsystem" failure rate is in the order of 4.2e-17.

Figure 12 shows the FTA results for the same aircraft configuration but with vendor B AoA sensors. The overall aircraft failure rate stays unchanged, but the ACC/Pilot climb control "subsystem" failure rate increases to 1.0e-15, which is still lower than the failure rate of the engine pair (2.3e-12).



	Error Model Element/Type	Computed...	Specified Prob
'ac.TwoSensorSpec' outgoing 'aceffect'	{Incident}	2.0e-09	
'ac.TwoSensorSpec'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		4.2e-17	
'flightsurface'	error state 'PilotInCharge'	4.2e-05	
'ACC'	'Xor' in outgoing propagation condition standbynocontrol	2.1e-05	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		1.0e-05
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		1.1e-05
'ACC'	'Xor' in outgoing propagation condition standbynocontrol	2.1e-05	
'aoa2' source 'reading'	{BadValue} from error source 'reading'		1.0e-05
'aoa2' source 'reading'	{ServiceOmission} from error source 'reading'		1.1e-05
'pilot'	'And' in outgoing propagation condition PilotControl	1.0e-12	
'pilot' event 'mistakes'	{ACCFailUnawareFlightControlMistake} from error event 'mistakes'		1.0e-03
'ACC' event 'standbyindicatorfail'	error event 'standbyindicatorfail'		1.0e-09
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09

Figure 11: FTA Results of the Aircraft Configuration with Two Sensor ACA Operation.



	Error Model Element/Type	Computed ...	Specified Proba
'ac.TwoSensorVB' outgoing 'aceffect'	{Incident}	2.0e-09	
'ac.TwoSensorVB'	'And' in outgoing propagation condition acincident	2.3e-12	
'engine1' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'engine2' source 'engineFailure'	{Incident} from error source 'engineFailure'		1.5e-06
'flightsurface'		1.0e-15	
'flightsurface'	error state 'PilotInCharge'	1.0e-03	
'ACC'	'Xor' in outgoing propagation condition standbynocontrol	5.0e-04	
'aoa1' source 'reading'	{BadValue} from error source 'reading'		2.5e-04
'aoa1' source 'reading'	{ServiceOmission} from error source 'reading'		2.5e-04
'ACC'	'Xor' in outgoing propagation condition standbynocontrol	5.0e-04	
'aoa2' source 'reading'	{BadValue} from error source 'reading'		2.5e-04
'aoa2' source 'reading'	{ServiceOmission} from error source 'reading'		2.5e-04
'pilot'	'And' in outgoing propagation condition PilotControl	1.0e-12	
'pilot' event 'mistakes'	{ACCFailUnawareFlightControlMistake} from error event 'mistakes'		1.0e-03
'ACC' event 'standbyindicatorfail'	error event 'standbyindicatorfail'		1.0e-09
'flightsurface' event 'FlightSurfaceFailure'	error event 'FlightSurfaceFailure'		2.0e-09

Figure 12: FTA Results of the Aircraft Configuration with Two Sensor ACA Operation and Vendor B AoA Sensors.

Summary

The objective of this note is to demonstrate the feasibility of continuous safety risk evaluation of safety critical systems. We wanted to demonstrate the importance of including lower Design Assurance Level (DAL) components in a safety risk analysis as lower criticality component. This issue was highlighted in the NASA Study on Flight Software Complexity led by Daniel L. Dvorak[4]. It draws on work by Charles Perrow[5] wrote about the causes of failure in highly complex systems, concluding that they were virtually inevitable. Perrow argued that when seemingly unrelated parts of a larger system fail in some unforeseen combination, dependencies can become apparent that could not have been accounted for in the original design. In safety critical systems the potential impact of each separate failure is normally studied in detail and remedied by adding backups. But failure combinations are rarely studied exhaustively; there are just too many of them and most of them can be argued to have a very low probability of occurrence. A compelling example in Perrow's book is a description of the events leading up to the partial meltdown of the nuclear reactor at Three Mile Island in 1979.

In our example we wanted to include the role of the pilot as a backup for a component (the ACC) that was assigned a lower DAL. The ACC relied on a single data source, the AoA sensor, which was available from three vendors at different quality levels and relatively easily damaged during operation. During the development and operation of the aircraft various decisions were made without assuring that assumptions about the pilot being a reliable backup were satisfied. With our example we have shown that the overall safety risk of the aircraft is strongly affected by faulty AoA sensors and a poorly designed ACC.

With the example we have demonstrated that it is feasible to perform safety risk analysis throughout the product life cycle – even when aircraft are already in operation. We have used probabilistic fault tree and minimal cut set analysis to determine the overall safety risk of the aircraft under a range of use scenarios. Such an analysis can be easily repeated for different what-if scenarios and applied not only up front during a system safety assessment (SSA), but also throughout the development and operation as decisions are made about changes in operational settings and design.

At this time, we have only modeled the aircraft and its components at an abstract level. As the component in question (the ACC) operates on highly time sensitive data and has restrictions on the degree of control it can exert we can refine the model to illustrate that additional issues with the real ACC could have been raised by using the full error taxonomy of the Error Model V2 Annex standard[6] as has been illustrated in previous work by the SEI[7].

[4] https://www.nasa.gov/pdf/418878main_FSWC_Final_Report.pdf and
https://www.slideshare.net/NASAPMC/dvorakdan

[5] Charles Perrow, Normal Accidents: Living with High Risk Technologies, Princeton University Press, 1984.

[6] https://samprocter.com/wp-content/uploads/2018/12/hilt18-emv2-library.pdf

[7] https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=435051
https://hal.archives-ouvertes.fr/hal-01292322/document

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM19-1050