Model Analysis Technical Report

Document information	
Project Title	ASHiCS
Project Number	E.02.05
Project Manager	Rob Alexander, Rob Alexander, University of York
Deliverable Name	Algorithm Evaluation Technical Report
Deliverable ID	D 3.2
Edition	00.00.03
Template Version	03.00.00
Task contributors	
University of York	

Abstract

Our previous deliverable for the ASHiCS project (D3.1) proposed the introduction of a two-stage process to provide contextual information to the worst case scenario discovered by the first stage of the evolutionary search. This final deliverable for ASHiCS discusses the implementation and results of the two-stage search process.

The technique enables safety analysts to look at the first stage search result in the context of its near neighbourhood of similarly configured scenarios. This was proposed after realising that safety analysts prefer to work with event probabilities using techniques such as fault tree analysis to try to determine the effectiveness of implementing safety barriers against certain outcomes. While automated search can return useful results with respect to simulated hazards or risks, it remains difficult to understand the result in the context of the simulation's configuration space, particularly if that configuration space is too large to search exhaustively.

For example, the first stage of an ASHiCS evolutionary search may discover a scenario configuration with high levels of risk. However, analysts are then faced with the problem that they don't know whether the scenario discovered by the search represents an extremely unlikely input configuration (i.e. a combination

of very rare events) or one of perhaps many variants within a set parameter range that produce similar levels of risk. This makes it difficult for safety analysts to propose barriers to mitigate the risk as it is impossible to quantify how much risk is involved in the wider context of the scenario being modelled.
To address this issue, a second stage search process was proposed that randomly samples from the near neighbourhood of the original result. The sampling of the near neighbourhood permits frequency estimates (i.e. how many high risk input configurations are close to the original?) and gives analysts greater confidence in the first stage search result (i.e. did it find the worst case scenario?). Finally, if the second stage sampling reveals configurations that produce similar or greater levels of risk, analysts can compare them individually against the original to discover what factors have increased the risk measurement.

Authoring & Approval

Prepared By - *Authors of the document.*		
Name & Company	Position & Title	Date
Dr Kester Clegg	Research Associate	17 May 2013
Dr R Alexander	Lecturer	

Reviewed By - *Reviewers internal to the project.*		
Name & Company	Position & Title	Date
Dr Rob Alexander	Lecturer	17 May 2013

Reviewed By - *Other SESAR projects, Airspace Users, staff association, military, Industrial Support, other organisations.*		
Name & Company	Position & Title	Date
Prof Tim Kelly	Professor	17 May 2013

Approved for submission to the SJU By - *Representatives of the company involved in the project.*		
Name & Company	Position & Title	Date
Dr Rob Alexander	Lecturer	17 May 2013

Rejected By - *Representatives of the company involved in the project.*		
Name & Company	Position & Title	Date

Rational for rejection
n/a

Document History

Edition	Date	Status	Author	Justification
00.00.01	21/03/13	Draft	Dr Kester Clegg	Initial draft
00.00.01	21/03/13	Submitted to Eurocontrol before internal reviews.	Dr Kester Clegg	Very little time left on the project to make changes to the deliverable before project ends (01/04/2013).
00.00.02	18 April 2013	Submitted to SJU	Dr Rob Alexander	Revisions based on EEC comments
00.00.03	17 May 2013	Submitted to SJU	Dr Rob Alexander	Added a section on related work

Intellectual Property Rights (foreground)

This deliverable consists of SJU foreground.

Table of Contents

List of tables

List of figures

Executive summary

Our previous deliverable for the ASHiCS project (D3.1) proposed the introduction of a two-stage process to provide contextual information to the worst case scenario discovered by the first stage of the evolutionary search. This final deliverable for ASHiCS discusses the implementation and results of the two-stage search process.

The technique enables safety analysts to look at the first stage search result in the context of its near neighbourhood of similarly configured scenarios. This was proposed after realising that safety analysts prefer to work with event probabilities using techniques such as fault tree analysis to try to determine the effectiveness of implementing safety barriers against certain outcomes. While automated search can return useful results with respect to simulated hazards or risks, it remains difficult to understand the result in the context of the simulation's configuration space, particularly if that configuration space is too large to search exhaustively.

For example, the first stage of an ASHiCS evolutionary search may discover a scenario configuration with high levels of risk. However, analysts are then faced with the problem that they don't know whether the scenario discovered by the search represents an extremely unlikely input configuration (i.e. a combination of very rare events) or one of perhaps many variants within a set parameter range that produce similar levels of risk. This makes it difficult for safety analysts to propose barriers to mitigate the risk as it is impossible to quantify how much risk is involved in the wider context of the scenario being modelled.

To address this issue, a second stage search process was proposed that randomly samples from the near neighborhood of the original result. The sampling of the near neighbourhood permits frequency estimates (i.e. how many high risk input configurations are close to the original?) and gives analysts greater confidence in the first stage search result (i.e. did it find the worst case scenario?). Finally, if the second stage sampling reveals configurations that produce similar or greater levels of risk, analysts can compare them individually against the original to discover what factors have increased the risk measurement.

1 Introduction

Our previous deliverable for the ASHiCS project (D3.1) proposed the introduction of a two-stage search to try and provide some contextual information to the results discovered by the first stage of the search. The introduction of a second stage was felt necessary so that safety analysts could gain a better understanding of the likelihood of a series of events discovered by the search actually occurring in the sector being modelled. However, this posed a problem as the entire search space was far too large to search exhaustively to determine all possible outcomes of the input configurations. This inability to conduct an exhaustive search to verify search performance means that no guarantee could be given that the worst case scenario (in terms of perceived levels of risk or incident) would always be found. Indeed, even with an extensive random sampling of the search space (as we demonstrated in D3.1), it is difficult to understand how much risk is present in the air sector we are modelling due to the sheer size of the input configuration space.

D3.1 looked at some possibilities of reducing the size of input configuration space using dimension reduction techniques. However, the result of these investigations was largely inconclusive. Our proposal to return to randomly sampling configurations was based on the realisation that safety analysts are not generally interested in analysing all possible outcomes of scenario, but tend to focus on variations of particular scenarios and the circumstances that led up to them. An automated equivalent of this could use our evolutionary search to discover high risk scenarios in the first instance, and then use a secondary process of random sampling close to the result to give us an estimate of risk in the near neighbourhood. In effect this gives us a form of dimension reduction, although we arrive via a different route to those proposed in D3.1. As we intensively sample the area around the first stage result we build up a risk profile of input configurations closely associated with the first stage result.

Understanding the risk in a scenario is difficult using an evolutionary search alone, as the search simply returns the highest ranking scenario in terms of a fitness score comprised of a compound risk measurement. Two different input configurations might produce very similar levels of risk with completely different conflicts and traffic distributions. Conversely one might get different levels of risk from very similar conflict patterns. However, if the sampling process could show that any scenarios in the near neighbourhood were essentially minor variants of traffic entry times (i.e. generating same conflicts from a hazard analysis perspective), safety analysts can look at specific conflicts to see how they differ and establish a parameter range for traffic entering the sector within which where they would expect variants of these conflicts to occur.

The second stage process can provide useful information to analysts, but there are still some decisions to be made relating to how widely we should sample around the first stage result. The second stage sampling process gives a window on “what’s close?” to the first stage result. If the size of the sampling window is too large, then the second stage might start to introduce radically different conflicts into the scenarios, making comparisons with the original result difficult. Likewise if it’s too small then we may miss nearby variations that are essentially the same from a hazard analysis perspective but which allow us to see a more accurate picture of the parameter ranges of the aircraft entry times. We look at this in more detail in Section 2.1.

1.1 Purpose of the document

To describe the algorithms used by the ASHiCS project to search for high levels of risk in a complex air traffic scenario.

1.2 Intended readership

This document’s intended readership are ATM planners, modellers and safety analysts interested in automated searches for hazards using fast time ATM simulation software such as RAMS Plus.

1.3 Inputs from other projects

We have had no input from other projects or technical advisors for this deliverable.

1.4 Glossary of terms

Evolutionary search

Form of search algorithm that uses selective pressure and mutation to improve a population of candidate solutions over many generations.

Evolutionary strategy

Pragmatics of evolutionary search relating to rate, range and restrictions of mutation, crossover, combination or other means of furthering good genes, population size, fitness selection policy, number of generations, etc.

Fitness function

Process used to select individuals from the population of candidate solutions by a ranking score assigning to each solution.

Search heuristics

Means of effectively guiding the search algorithm through the search space.

Search Landscape

Imaginary visualisation of a search space in which the fitness of each individual in a set's population is shown as a measure of vertical height with individuals of similar fitness being placed close together. By plotting a curve between the heights of individuals a landscape can be drawn with peaks representing areas in the solution space that contain the fittest individuals. This visualisation is extremely pervasive within the search literature, however it has many theoretical problems: i) there are no horizontal axis which can place the individuals geometrically within a set so the notion of similar solutions lying close to one another is hard to justify; ii) the visualisation breaks down completely in high dimensionality (i.e. where many factors may affect fitness levels), as there are likely to be areas of "impossible" gene combinations that cannot be realised in a solution.

Weighted fitness function

In a multi-objective fitness function, it is possible to assign greater "weight" to certain factors within the fitness evaluation so that the search favours solutions presenting those characteristics over others.

1.5 Acronyms and Terminology

Term	Definition
ANSP	Air Navigation Services Providers
API	Advanced Programming Interface
ARMS	Aviation Risk Management Solutions (working group)
ASHiCS	Automating the Search for Hazards in Complex Systems
ATC	Air Traffic Control
ATCos	Air Traffic Controllers
ATM:	Air Traffic Management
ATOMS	Air Traffic Operations and Management Simulator
CGP	Cartesian Genetic Programming

Term	Definition
CFIT	Controlled flight into terrain
CRT	Computational Red Teaming
CPA	Closest point of approach (between two aircraft)
CSV	Comma separated values
DFS	Deutsche Flugsicherung
EC	Evolutionary Computation
ECAC	European Civil Aviation Conference
eDEP	Early Demonstration & Evaluation Platform
EFT	Evolutionary functional testing
ERC	Event Risk Classification
ESD	Event sequence diagram
FAA	Federal Aviation Administration
FL	Flight level (given in hundreds of feet)
FTS	Fast time simulation
IRP	Integrated Risk Picture
ISA Software	Innovation for Sustainable Aviation Software
MOGA	Multi-Objective Genetic Algorithms
NASA	National Aeronautics and Space Administration
NATS	National Air Traffic Service (UK)
NSGA	Non-dominated sorting genetic algorithm
PUMA	Performance and Usability Modelling
RAMS	Re-organized ATC Mathematical Simulator
RTS	Real time simulation
SDAT	Sector Design and Analysis Tool (FAA)
SESAR	Single European Sky ATM Research Programme
SID	Standard instrument departure
SoS	System of Systems

Term	Definition
SJU	SESAR Joint Undertaking (Agency of the European Commission)
SJU Work Programme	The programme which addresses all activities of the SESAR Joint Undertaking Agency.
SESAR Programme	The programme which defines the Research and Development activities and Projects for the SJU.
SSE	Safety significant event
SSMT	System Safety Management Transformation (internal program of FAA)
STAR	Standard arrival
TAAM	Total Airspace and Airport Modeller
TMA	Terminal Area

2 The two stage search process

We wished to provide safety analysts with a tool that could not only discover risk within scenarios but would also provide some context to that discovery. Providing context in this sense means giving the analysts some information about how many more scenarios are similar to this one with respect to the sources of risk being investigated. The previous single stage evolutionary search could discover high risk scenarios but we had no way of knowing if the search result represents the only scenario configuration that could generate that risk score, or whether those risk levels are likely to be generated by any scenario containing similar grouping of aircraft entering the sector within a set time of one another. For example, if three aircraft conflict and have entry times of 10, 15 and 20 minutes, and we search in the near neighbourhood of a two minute window, then we would expect to find the same conflict at 8, 13 and 18 minutes, and at 11, 16 and 21 minutes, and so on. To the search each of these scenarios is a different scenario configuration, but to a safety analyst each of the configurations would generate more or less the same conflict.

There are several ways to try to solve this. If the scenarios were less complicated, it might be possible to write an algorithm that would compare the inputs of high scoring scenarios for similarity (in fact, this would not be dissimilar to the dimension reduction attempts we discussed in our earlier reports). Alternatively, it would be possible to use a sensitivity analysis that records the effects of mutation on the near neighbours of selected scenarios. By recording the fitness scores of the mutations, we could track whether the mutations are causing a significant drop in the fitness level of the scenario, particularly in the latter part of stage one when a high ranking scenario can remain as the best in a population for many generations. If mutation to a high ranking scenario results in a significant loss of fitness, it indicates that the scenario's risk levels are very sensitive to small changes in the input configuration. By gauging the average fitness of mutations and tracking the distance between them and the best in a generation, we can gain an understanding of the solution landscape – i.e. whether the peaks of high fitness are very narrow (therefore difficult to find, D3.1 has an in-depth explanation of this). The plots in Figure 1 and Figure 9 show typical sensitivity analyses that record the five best scenarios in the population (see [1] for a detailed analysis).

However a sensitivity analysis gives a picture only of the mutations to a small number of high scoring scenarios in each generation and so uses a very small sample size of "near neighbours". In a high dimensional solution space (even over many generations) it may simply miss configurations that are close to the best. It cannot provide the sort of detailed frequency information that is useful to safety analysts in trying to determine how likely a particular input configuration is to occur, or whether there are many configurations that would generate similar levels of risk. For this we need the closest we can get to an exhaustive search of the area around the final result of the stage one result. Unfortunately, an exhaustive search would still take too long to be practical, as there is a lower limit to how small you can make the sample size before you risk missing high scoring variants.

The following sections describe how we extended the idea of a sensitivity analysis so that we could get an understanding of the context of a search result by extensively sampling the near neighbourhood of the final result. The sampling process is simply a continuation of the search, albeit over a fraction of the original search space. However, rather than continue to employ evolutionary search, we randomly sample within the near neighbourhood of the best scenario. As we sample many thousands of scenarios within a relatively small parameter range we are able to achieve coverage levels far in excess of those of the original search.

2.1 Deciding near neighbourhood sizes

From our limited experience of analysing second stage variants of a stage one result, it appears that there is no consistent rule we can apply to determine the sampling size of the near neighbourhood. Safety analysts must decide what represents a suitable parameter range for the aircraft entry times to the sector being modelled. In our case, as we model an en route sector where the aircraft are travelling fast and there are relatively few conflicts, we give ourselves a sampling window size of 120 seconds either side of the original entry times of the aircraft in the stage one result. No other parameters are changed from the first stage scenario. This 120 second range was determined experimentally by comparing the scenarios discovered by the second stage with the original from the

first stage. If they differed too radically we reduced the range, if the second stage failed to find any close matches to the original fitness score we extended the range.

Our initial tests used a near neighbourhood of 180 seconds either side of the original entry times, but as this gives up to 6 minutes variability (assuming the aircraft are not stuck in queues, see D2.2) we found that it created scenarios that contained conflicts which were substantially different from the stage one scenario. Conversely, we also tried making the sampling window size 90s and 60s. These were on occasion as good as 120s (in the sense that no higher ranking scenarios were found than the stage one result, see Section 3.1.3). However, as one of the principal functions of the second stage is to give a degree of confidence that the first stage found the worst case or near to the worst case scenario. By setting the sample window size too low, you lose this confidence as it impossible to know if you have missed some variants that lie outside the window but which still resemble the original search result. We decided given our sector's characteristics that a near neighbourhood of 120s was a good compromise.

We suspect that each scenario model will need to be taken on its own merits in order to determine a suitable sample window size. In our en route sector, this is relatively easy to experiment with. However we can envisage it being much more difficult in a mixed use sector such as a terminal area. For example, where there are likely to be aircraft climbing and descending during take-off and landing, with changing speed and degrees of proximity allowed. In this type of scenario it may make no sense to sample outside the take-off or landing separation times (for example), as there is often very little freedom to alter these slots and this window size is likely to cover all the variants of any significance.

2.2 Confidence levels for sample sizes

Determining whether you have achieved sufficient sampling to get an adequate coverage of the input configuration possibilities in the near neighbourhood is difficult. While it is possible to use statistical methods to show that a normal distribution of sample configuration has been achieved during the second stage, we have no way of knowing in advance whether a normal distribution should be expected in the solution space we are sampling, particularly as we are already targeting a small area around a relatively rare input configuration.

One of the problems we encountered with this issue is that the first stage search result is often already very close to the worst case scenario that can be said to exist in the near neighbourhood. Subsequent sampling may reveal just a few worse cases, or even none; of those that are found most have only marginally worse levels of risk (although this does not mean that the conflicts and workloads are necessarily similar to the first stage result).

Although we have had insufficient time to explore the behaviour of the second stage sampling in detail, a typical discovery rate for higher ranking scenarios in terms of their risk seems to be around 2 or 3 scenarios out of the 5000 sampled in the near neighbourhood, or about 0.04 - 0.06%. This appears to be very low, and increasing the intensity of the sampling to 10,000 did not increase the discovery rate sufficiently to justify the increase in time required to acquire the extra samples (rates remained roughly equal or increased slightly to 3 or 4 from 10,000). These discovery rates for the second stage may be scenario specific; we were unable to test them on scenarios modelling different types of air sector, traffic types or ATCo workload.

It should be noted that perhaps 2 out of 5 second stage searches failed to produce any instances of scenarios with higher fitness scores than the first stage result. This might be an indication that the total number of samples could be increased. If there were more time available for the project, we would recommend a simple test in the code that records the number of results from the second stage that have a greater fitness score than the first stage result. If no results are obtained from successive second stage samples, then the code could continue to step up the number of samples it makes to some predetermined maximum. This would automate the sampling process of the second stage to a degree; however the sample size would need to be tested against different near neighbourhood sizes to give confidence that a suitable number of samples had been arrived at. There are also pragmatic considerations for resource allocation. Increasing the number of samples in the second stage can significantly increase the total length of the process, leaving less time for experimentation with other parameters.

3 Analysis of Results

3.1 Example of near neighbourhood variants

3.1.1 Significant variants

Example 1

Figure 1 shows the sensitivity analysis for our storm scenario (by plotting the fitness of the five best individuals at each generation), with scenario fitness score on the vertical axis and number of individuals on the horizontal. Due an increased length of processing time required to calculate the NASA complexity score [2] we reduce these runs to 250 generations (our experiments suggest that improvement is rarely achieved after this point), and thus have 1250 “top 5 individuals”. The figure shows a typical run for the first stage search. Note that the “plateaus” (horizontal lines) in the plot are caused by no mutations improving on the previous best scenario which is carried over to the following generation.

For the second stage we take the best scenario of the final generation and start to randomly sample variant traffic entry times within its near neighbourhood. For our sector, we decided that the near neighbourhood of the original result could not be any larger than a 120 second range of the original entry times. Our rationale is that greater than this (180 seconds) sometimes resulted in the creation of high scoring scenarios that were substantially different to the original and therefore difficult to compare, e.g. a conflict may appear, disappear or radically change. Figure 2 shows 5000 samples of the near neighbourhood of the best scenario of the final generation shown in Figure 1. The fitness score of the original is shown as a continuous horizontal line just below the fitness score of 2500 (vertical axis).

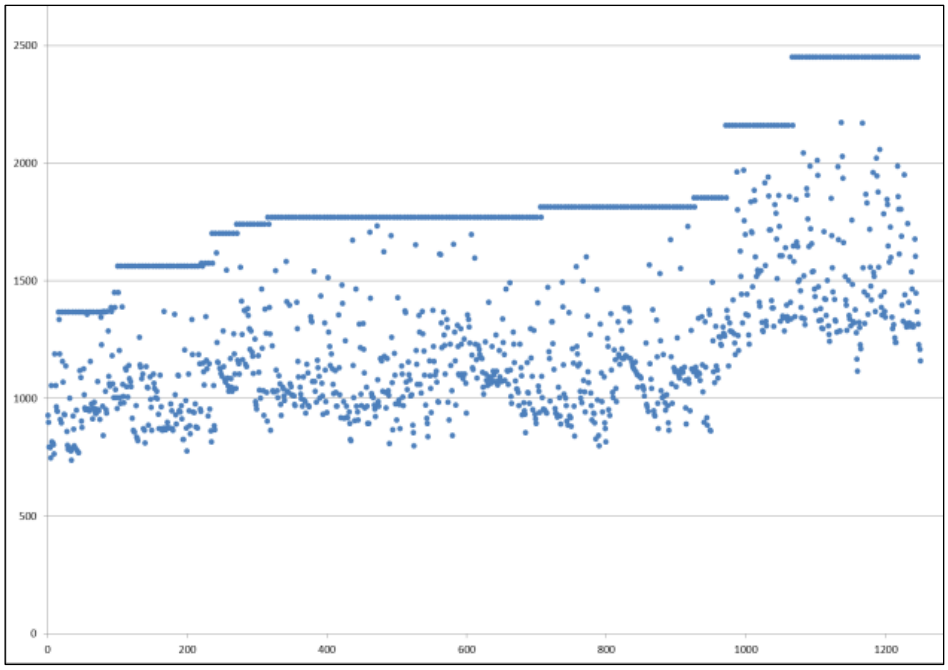


Figure 1: First stage search showing sensitivity analysis with the five highest ranked scenarios per generation plotted vertically against their fitness score. The best scenario in each generation is carried over unchanged to the next generation.

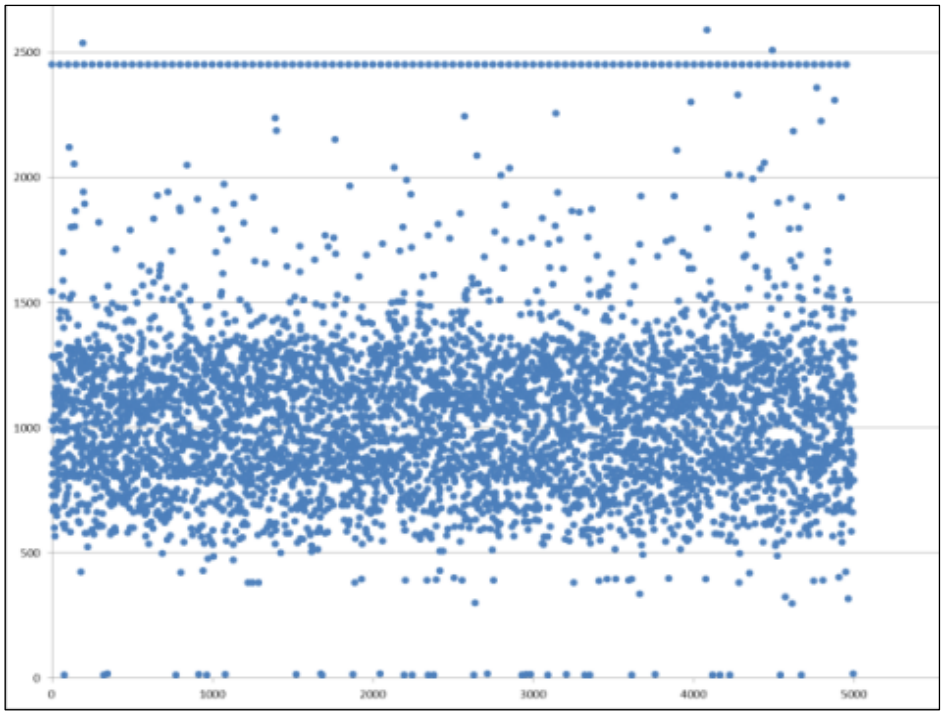
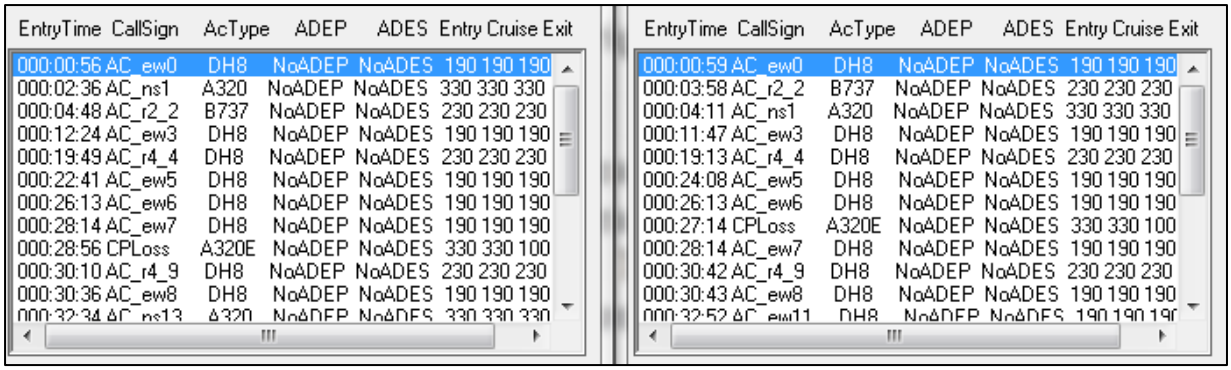


Figure 2: Near neighbour sampling of solution space around final result of Figure 1. The best solution from stage one is shown as a horizontal bar. Just 3 scenarios outrank the best solution from the stage one search. Note the zero ranked scenarios are an artefact of file locks happening as a result of delayed processes which raises an error in our code, so we rank all such incidents as a zero fitness score.

What is immediately interesting about the right-hand plot is that we can see that the first stage of the search failed to find the worst case scenario. By extensively sampling the near neighbourhood in the second stage, we were able to uncover 3 variants whose entry times improved the fitness score of the original scenario (as indicated by the dots above the horizontal line). However, it is also apparent that the original search result is very close to the worst cases found in the second stage, something which we have confirmed by careful comparison of the aircraft entry times involved in conflicts. We can also see that the vast majority of variants, even within this narrowly defined near neighbourhood of a high ranking scenario, do not come anywhere near the original fitness score. This suggests that there is a relatively narrow parameter band for aircraft entry times that generated the original high scoring scenario and its close variants.

We conducted a manual analysis of the variant entry times for these scenarios, and it appears that slight variants of the entry times of just 3 out of 20 aircraft are responsible for all the reported conflicts. Figure 3 shows the entry times for the original scenario discovered by the search and one of the near neighbourhood variants that out-scored it. From the conflict summary log file produced by RAMS Plus, we can see that the main aircraft involved in conflicts are AC_ew3, AC_ew7 and CPLoss.

When we look at the entry times for these aircraft, we can see that AC_ew7 is “trapped” between two other aircraft (see D2.2 for an explanation into this feature of aircraft “queued” on a flight path), so that it can’t move forward or backwards along the ew flight path without generating a wake turbulence conflict with following or preceding aircraft. The search is therefore not allowed to vary its position. However, CPLoss (the aircraft that suffers the emergency cabin pressure loss) is free to mutate its entry time on the nw flight path and this can cause a greater loss of separation when it conflicts with AC_ew3. It is CPLoss’s *initial* conflict, rather than the later conflicts from “bunched up” aircraft on the ew flight path, that creates a higher risk scenario. By comparing the high scoring variants in this manner, safety analysts can check what changes to entry times to increase the risk levels. In this instance, they can see that the critical factor is the entry time of CPLoss in relation to AC_ew3, rather than changes to AC_ew7 or the fact that aircraft are severely “bunched up” on the ew flight path.



EntryTime	CallSign	AcType	ADEP	ADES	Entry	Cruise	Exit
000:00:56	AC_ew0	DH8	NoADEP	NoADES	190	190	190
000:02:36	AC_ns1	A320	NoADEP	NoADES	330	330	330
000:04:48	AC_r2_2	B737	NoADEP	NoADES	230	230	230
000:12:24	AC_ew3	DH8	NoADEP	NoADES	190	190	190
000:19:49	AC_r4_4	DH8	NoADEP	NoADES	230	230	230
000:22:41	AC_ew5	DH8	NoADEP	NoADES	190	190	190
000:26:13	AC_ew6	DH8	NoADEP	NoADES	190	190	190
000:28:14	AC_ew7	DH8	NoADEP	NoADES	190	190	190
000:28:56	CPLoss	A320E	NoADEP	NoADES	330	330	100
000:30:10	AC_r4_9	DH8	NoADEP	NoADES	230	230	230
000:30:36	AC_ew8	DH8	NoADEP	NoADES	190	190	190
000:32:34	AC_ns13	A320	NoADEP	NoADES	330	330	330

EntryTime	CallSign	AcType	ADEP	ADES	Entry	Cruise	Exit
000:00:59	AC_ew0	DH8	NoADEP	NoADES	190	190	190
000:03:58	AC_r2_2	B737	NoADEP	NoADES	230	230	230
000:04:11	AC_ns1	A320	NoADEP	NoADES	330	330	330
000:11:47	AC_ew3	DH8	NoADEP	NoADES	190	190	190
000:19:13	AC_r4_4	DH8	NoADEP	NoADES	230	230	230
000:24:08	AC_ew5	DH8	NoADEP	NoADES	190	190	190
000:26:13	AC_ew6	DH8	NoADEP	NoADES	190	190	190
000:27:14	CPLoss	A320E	NoADEP	NoADES	330	330	100
000:28:14	AC_ew7	DH8	NoADEP	NoADES	190	190	190
000:30:42	AC_r4_9	DH8	NoADEP	NoADES	230	230	230
000:30:43	AC_ew8	DH8	NoADEP	NoADES	190	190	190
000:32:52	AC_ew11	DH8	NoADEP	NoADES	190	190	190

Figure 3: Left: Traffic entry times for best scenario in first stage search. Right: Traffic times for highest scoring scenario in second stage near neighbourhood sample.

This contextual information about variants of the original stage one search result can be valuable to safety analysts. First, it allows some confidence that the original search result was close to the worst case that could be found. Second, it allows analysis of near variants to see which aircraft are generating the worst conflicts within a relatively narrow parameter range. Finally, it shows that most variants within 120 seconds of the original entry times do not create higher risk scenarios. This last piece of information means that safety analysts can look at the scenario and make an estimate of the likelihood of aircraft entry times falling within the parameter band of the worst case scenarios based on historical records for the sector being modelled. They can study the conflicts and the near variants to see whether they form a consistent pattern, and can investigate the options for creating safety barriers to prevent future hazards from occurring within that sector.

Example 2

Our second example looks at the effects of a severe disruption caused by a storm. The storm trajectory moves towards the oncoming traffic in the ns and ew flight paths (Figure 4); this trajectory means that diversions around the storm run the risk of coming into conflict with aircraft from the nearby converging flight path (Figure 5). Just prior to the point where the two flight paths cross, the first stage result shows CPLoss commencing its emergency descent (Figure 6).

Clearly this scenario contains bad timing for CPLoss combined with the worst storm trajectory, and as a result we would expect a high fitness score reflecting the overall levels of risk. From the stage one sensitivity analysis (Figure 7) we can see that very few mutations are getting close to the best ranked scenario and this pattern is repeated in the stage two analysis where random samples in the near neighbourhood of the best stage one scenario show just two cases out of 5000 that marginally outscore the original stage one result.

Perhaps surprisingly, when we examine the conflict summary tables output to compare the worst case scenario in stage one (Table 1) and stage two (Table 2), the first thing we notice is that one of the conflicts between two incidental aircraft has been removed completely in the stage two scenario, despite the stage two scenario having a greater fitness score indicating higher risk. This means that the increased risk must either be coming from a change to the remaining conflicts, a change to the NASA complexity score or changes related to the ATCo workload. As we can see the number of conflicts has reduced in the second stage scenario, we must conclude that an increase of workload for the ATCo is unlikely, and we can make a similar assumption about the sector complexity score.

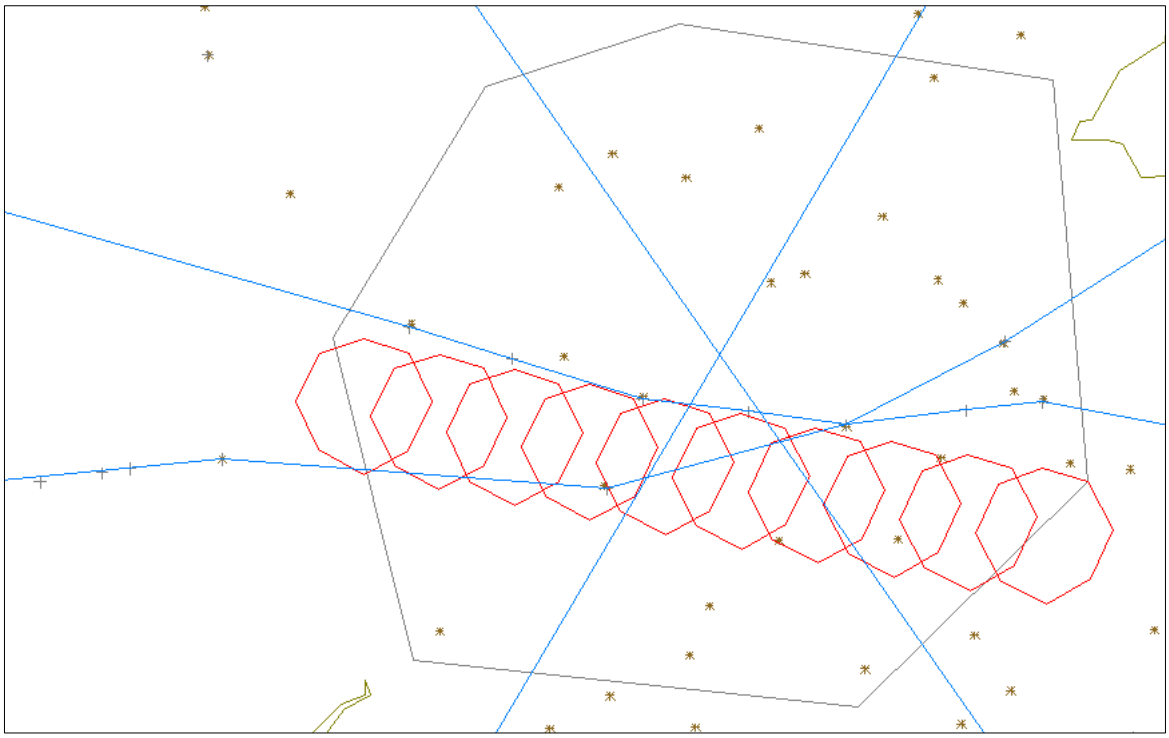


Figure 4: Storm trajectory that causes the greatest disruption is when it moves against the prevailing traffic on the ew and ns flight paths (right to left).

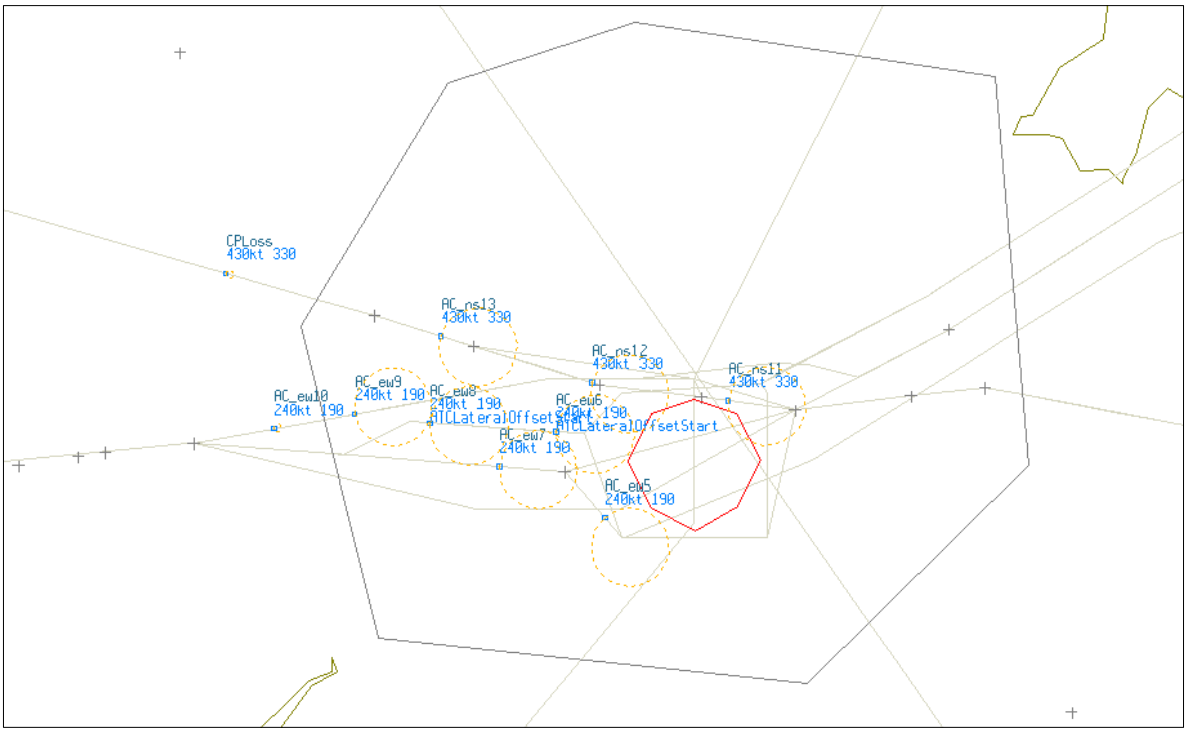


Figure 5: Storm moving across sector, CPLoss about to enter top left.

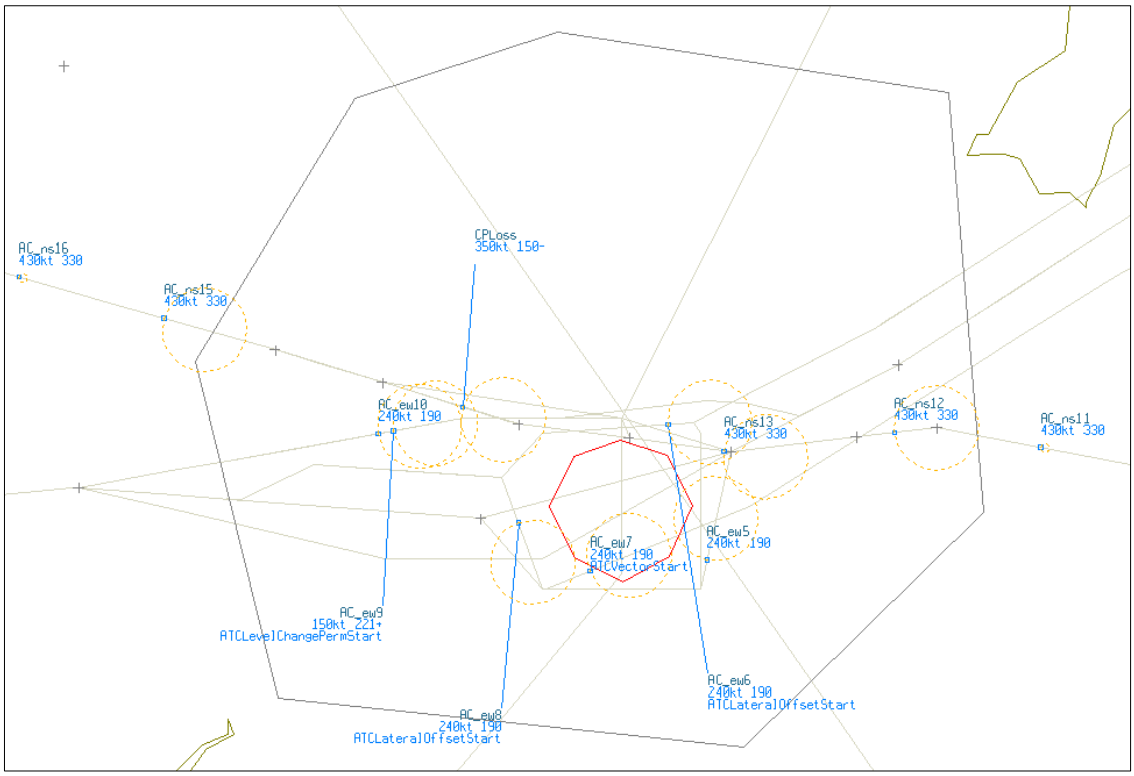


Figure 6: Detailed view after CPLoss has started emergency descent.

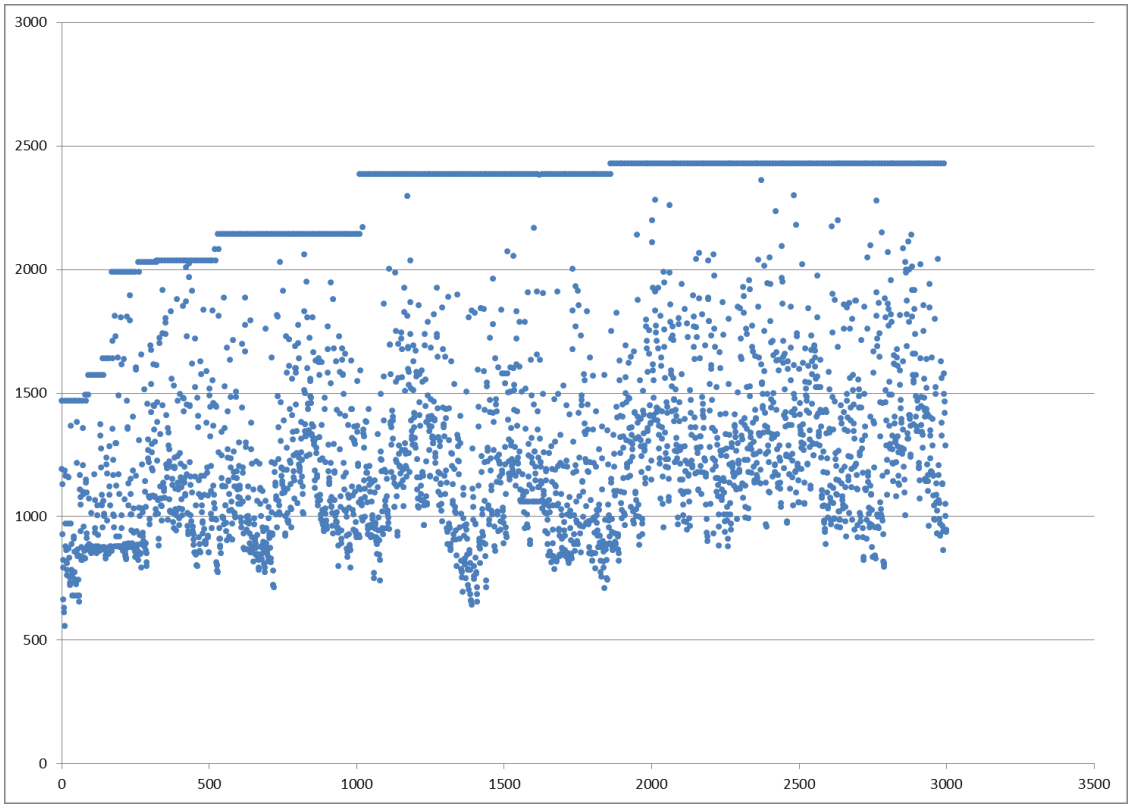


Figure 7: First stage search. Sensitivity analysis showing ten highest ranked scenarios of each generation, with the best in each generation carried over unchanged into next generation.

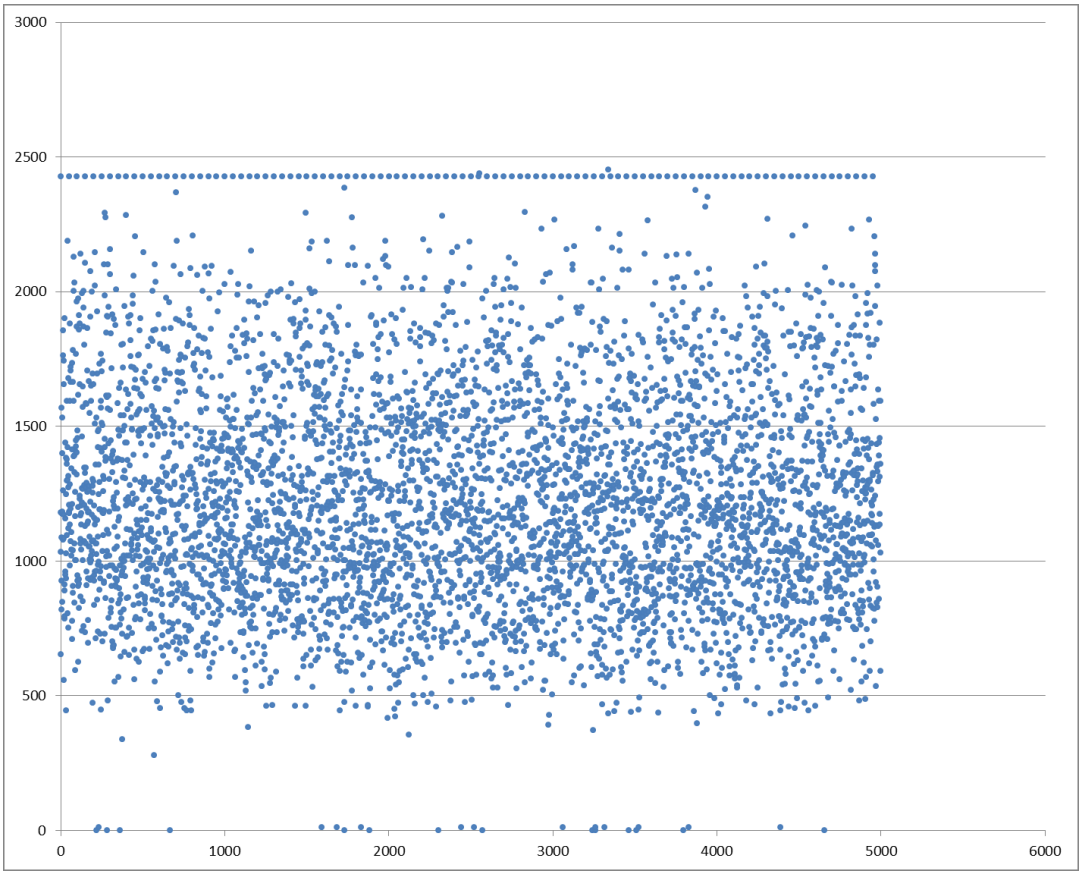


Figure 8: Second stage search. Random sampling around final result of stage one shown in Figure 7.

However, one of the most important weightings in the fitness score for risk is to have some measure of conflict severity. This measure looks the amount of available separation as a percentage, and as the loss of separation becomes very severe the percentage rapidly diminishes to almost zero, reflecting the reality that a resolution between the aircraft to avoid a collision is probably unlikely. The second stage scenario has such a figure (highlighted in the final column). A “severity of available separation” figure this low indicates resolution did not happen, something which is extremely unlikely.

Sure enough, after carefully following the two aircraft closely to monitor their conflict, it becomes clear that the available separation percentage is an anomaly that occurs because of how aircraft are set up to enter and leave the sector. We modelled all aircraft as appearing on their flight paths at cruise height outside the sector and leaving at sector at the same height (with the exception of CPLoss). The aircraft appear at the start of the flight path and fly to the end point of that flight path: they do not land or take off at an airport. The very low figure of available separation percentage comes about as the aircraft make their way in parallel (due to earlier diversions around the storm) towards the endpoint of the flight path. The endpoint is outside the sector and therefore there is no ATCo to resolve the conflict that occurs as the aircraft reach the endpoint at the same time, resulting in either a collision or near collision. This is still reported in the conflict summary table, despite lying outside the sector we were modelling. Prior to this result, we were unaware of this reporting behaviour in RAMS Plus and this sort of “out of sector” conflict is very rare, as most aircraft can make their way to the flight path endpoint with plenty of separation, but it serves to illustrate how careful we must be when designing our fitness functions and demonstrates the benefits that a detailed comparison between the stage one and stage two scenarios as it can quickly uncover errors such as this. After a check of the other two high scoring scenarios in the stage two samples, we concluded that none of the variations represented a significant increase in the level of risk.

Table 1: Conflict summary table of highest ranked scenario from stage one search (Figure 7).

Flight1	Detection Time	AC Model	Category	CP AAlt	Cf Attitude	Cf Speed	Flight2	CP AAlt	Cf Attitude	Cf Speed	Dist To Cf	Cf Start	Cf End	Severity Avail Sep
AC_r2_2	00:18:58	B737	Normal	230	Cruise	425	**AC_r4_1**	230	Cruise	235	33.75	00:27:35	00:28:31	0.087
AC_ew6	00:39:22	DH8	Normal	190	Cruise	240	**AC_ew5**	190	Cruise	240	74.08	00:57:53	00:57:58	1.249
AC_ew8	00:43:25	DH8	Normal	190	Cruise	240	**AC_ew7**	190	Cruise	240	74.08	01:01:56	01:02:01	1.249
AC_ew9	00:46:00	DH8	Normal	190	Cruise	240	**AC_ew7**	190	Cruise	240	74.42	01:04:37	01:17:54	0.497
AC_ew10	00:48:38	DH8	Normal	190	Cruise	240	**AC_ew8**	190	Cruise	240	75.91	01:07:38	01:08:50	0.678
CPLoss	00:50:29	A320E	Emergency	180	Descent	350	**AC_ew9**	190	Climb	150	25.36	00:56:49	00:57:13	0.6

Table 2: Conflict summary table of highest ranked scenario from stage two sampling (Figure 8).

Flight1	Detection Time	AC Model	Category	CP AAlt	Cf Attitude	Cf Speed	Flight2	CP AAlt	Cf Attitude	Cf Speed	Dist To Cf	Cf Start	Cf End	Severity Avail Sep
AC_ew6	00:38:55	DH8	Normal	190	Cruise	240	**AC_ew5**	190	Cruise	240	74.1	00:57:27	00:57:30	1.259
AC_ew8	00:43:25	DH8	Normal	190	Cruise	240	**AC_ew7**	190	Cruise	240	74.11	01:01:57	01:01:59	1.27
AC_ew9	00:45:43	DH8	Normal	190	Cruise	240	**AC_ew7**	190	Cruise	240	74.49	01:04:21	01:17:52	0.312
AC_ew10	00:50:04	DH8	Normal	190	Cruise	240	**AC_ew8**	190	Cruise	240	71.57	01:07:58	01:09:51	0.002
CPLoss	00:50:35	A320E	Emergency	180	Descent	350	**AC_ew9**	190	Cruise	240	25.36	00:56:55	00:57:19	0.745

3.1.2 Non-significant variants

Example 3

In this example, we take a case (Figure 9) where the second stage of the search (Figure 10) finds a variant with a higher fitness score, but on analysis we discover that the increased score is due to how the NASA complexity measure is calculated and thus does not indicate a significant worsening of risk levels in the sector. Our detailed analysis showed that although the fitness score rose slightly, this was not due to an increase in the number of conflicts or a decreasing conflict separation percentage, therefore there appears to be no increase in the loss of separation between any of the aircraft.

(We can note that Figure 10 shows some degree of stratification in the second-stage fitness scores. This suggests that there are step changes in risk levels along certain lines in the near neighbourhood, with the stratum at 500-1000 fitness occupying a large part of the neighbourhood. We don't have enough data, however, to know whether this is common or uncommon.)

On closer inspection of the RAMS Plus output logs, we discover that an additional task has been added to the ATCo workload. When we analyze the traffic times between the stage one result and the highest ranked scenario in stage two, we can see that most aircraft change their start time by a negligible amount. However, a group of aircraft after the 00:40:00 mark do have their times altered to almost the full extent of the neighborhood limit (120 seconds of the original start time).

When we looked at the running order of these aircraft entering the sector (most of which are bunched together on the same flight path, see Figure 11) we observed that the change in start times means that around the 01:09:00 mark, the second stage scenario has a greater number of aircraft in the sector than the original stage one scenario. The total number of aircraft in the sector at any minute is one of the major factors used in the NASA complexity measure as it tries to assess any increases in complexity that might initiate dynamic re-sectoring by the multi-sector planner. While it is true that more aircraft can increase the workload or difficulty of managing the sector, in this case the aircraft are only being handed over or received from adjacent sectors. The increased number of aircraft lasts just a few minutes and therefore we conclude that no additional risk is created in this variant of the original scenario.

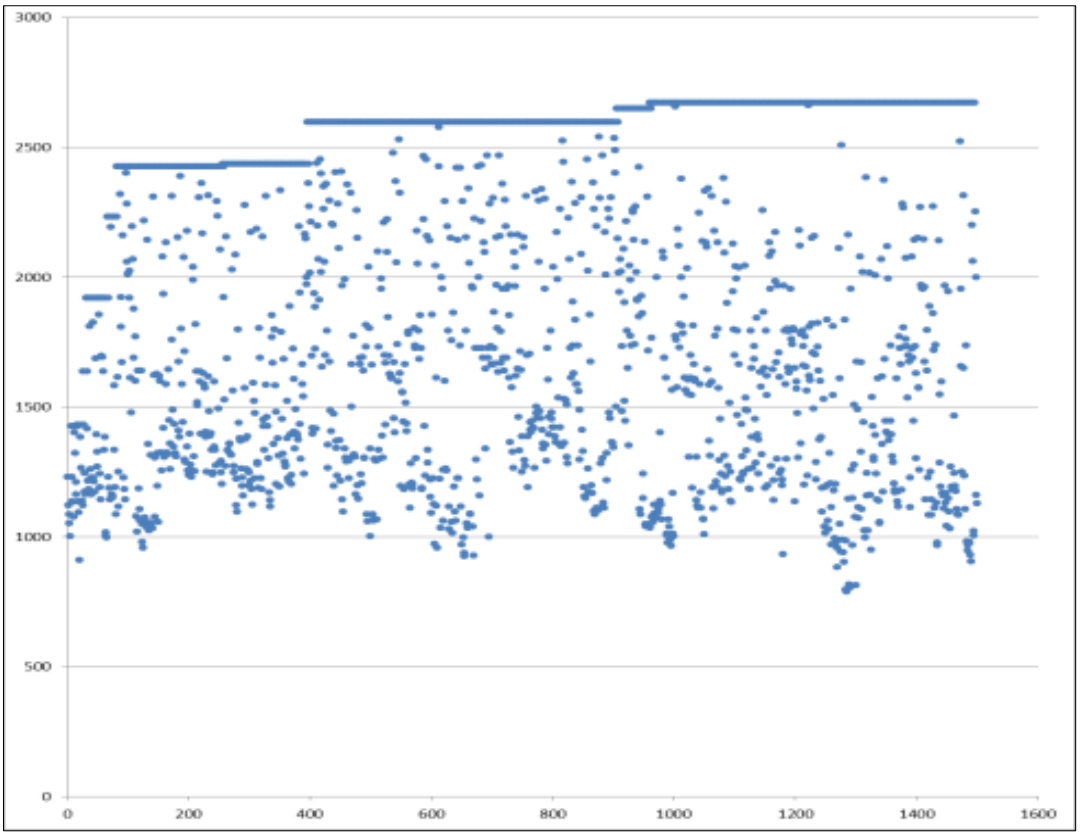


Figure 9: First stage search. Sensitivity analysis showing ten highest ranked scenarios of each generation, with the best in each generation carried over unchanged into next generation.

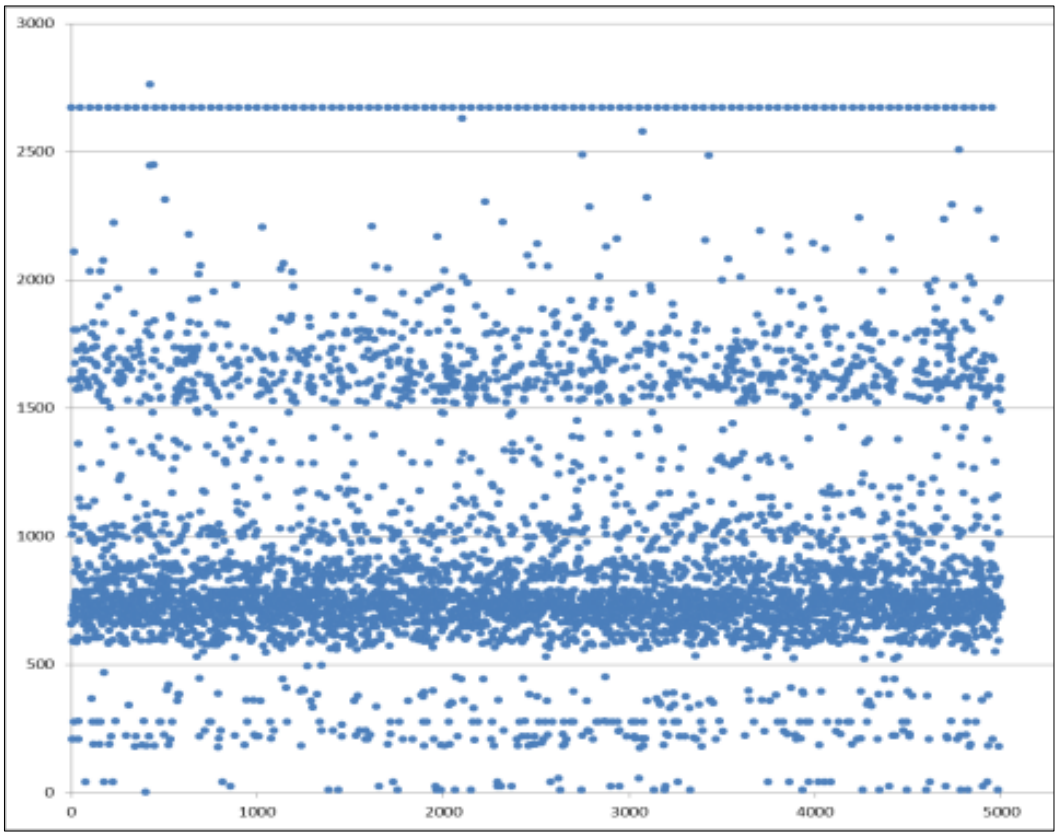
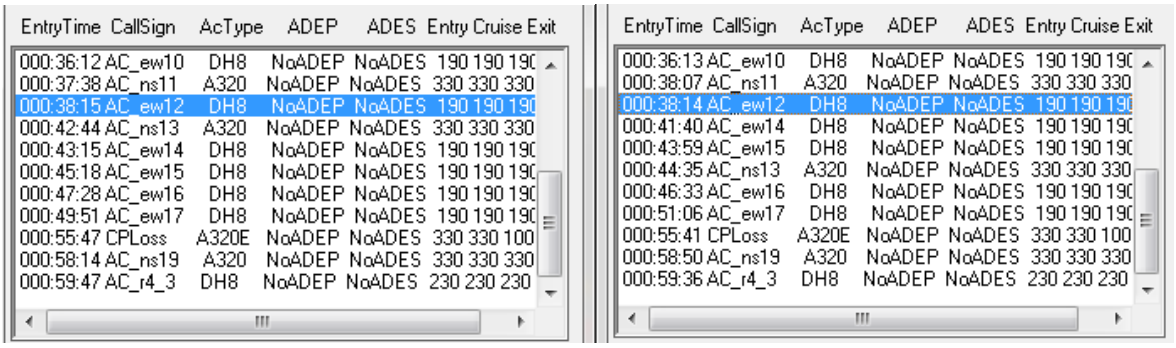


Figure 10: Second stage search. Random sampling around final result of stage one shown in Figure 9.



EntryTime	CallSign	AcType	ADEP	ADES	Entry	Cruise	Exit
000:36:12	AC_ew10	DH8	NoADEP	NoADES	190	190	190
000:37:38	AC_ns11	A320	NoADEP	NoADES	330	330	330
000:38:15	AC_ew12	DH8	NoADEP	NoADES	190	190	190
000:42:44	AC_ns13	A320	NoADEP	NoADES	330	330	330
000:43:15	AC_ew14	DH8	NoADEP	NoADES	190	190	190
000:45:18	AC_ew15	DH8	NoADEP	NoADES	190	190	190
000:47:28	AC_ew16	DH8	NoADEP	NoADES	190	190	190
000:49:51	AC_ew17	DH8	NoADEP	NoADES	190	190	190
000:55:47	CPLoss	A320E	NoADEP	NoADES	330	330	100
000:58:14	AC_ns19	A320	NoADEP	NoADES	330	330	330
000:59:47	AC_r4_3	DH8	NoADEP	NoADES	230	230	230

EntryTime	CallSign	AcType	ADEP	ADES	Entry	Cruise	Exit
000:36:13	AC_ew10	DH8	NoADEP	NoADES	190	190	190
000:38:07	AC_ns11	A320	NoADEP	NoADES	330	330	330
000:38:14	AC_ew12	DH8	NoADEP	NoADES	190	190	190
000:41:40	AC_ew14	DH8	NoADEP	NoADES	190	190	190
000:43:59	AC_ew15	DH8	NoADEP	NoADES	190	190	190
000:44:35	AC_ns13	A320	NoADEP	NoADES	330	330	330
000:46:33	AC_ew16	DH8	NoADEP	NoADES	190	190	190
000:51:06	AC_ew17	DH8	NoADEP	NoADES	190	190	190
000:55:41	CPLoss	A320E	NoADEP	NoADES	330	330	100
000:58:50	AC_ns19	A320	NoADEP	NoADES	330	330	330
000:59:36	AC_r4_3	DH8	NoADEP	NoADES	230	230	230

Figure 11: Traffic times for original scenario in Figure 9 (right) and highest ranked scenario from near neighbourhood sample in Figure 10 (left).

Example 4

In this example, we look at how the stage one evolutionary search can occasionally get stuck on local optima (in the global sense). In the stage one search, this can happen whenever we use a small population size (50 scenarios) in order to decrease the run time of the stage one search. The reason is due to the fact that first population is entirely randomly generated across the whole search space, whereas only a proportion (40%) of subsequent populations are randomly generated. After the first generation, the evolutionary algorithm starts to select scenarios based on the fitness function. This means that from that point onwards, 60% of the population is generated from the top fifth selected from the previous generation. This has implications, as small populations sizes give poor coverage of the search space on that initial generation, there is always a danger that a poor scenario (from a risk profile perspective) gets selected to be passed onto the next generation.

Once selected as a high ranking scenario, the mutation operator is only allowed to change aircraft entry times into the sector in order to increase levels of risk. It cannot change the original distribution of aircraft on the flight paths, or alter the start time or direction of the storm. A small population size means that if the initial generation of scenarios did not include any high ranking scenarios, there is a risk that a relatively low scoring scenario will be evolved and will continue to dominate the scenario rankings over many generations. We have decided to include an example of this type of result, even though it occurs quite rarely (perhaps 1 in 15-20).[1]

In a situation such as this, the evolutionary search performance initially looks disappointing (in the sense that little improvement to the fitness score occurs as a result of mutation). Quite often this is the result of a storm track that is not particularly disruptive to the flight paths. As a result, the only additional conflicts or workload that the mutation to aircraft entry times can achieve is principally through aircraft coming into conflict with CPLoss (the aircraft that undergoes emergency cabin pressure loss, see D1.2 and D2.1) as these conflicts are weighted to give higher scores. We show an example of a low scoring scenario from a stage one search that has the storm trajectory crossing the western edge of the sector (Figure 12). As the storm progresses, we can see how the aircraft are able to divert around it without causing major disruption to the other flight paths (Figure 13 and Figure 14). In fact by the time the aircraft re-join their flight paths the only conflicts that occur are not as result of the storm at all, but such that would occur if the storm were not there (notwithstanding the delays caused by previous diversions around the storm).

Clearly scenarios such as this offer little opportunity for the evolutionary algorithm to improve the fitness score other than altering entry times to increase the chances of coming into conflict with other aircraft (in particular CPLoss). We can see this reflected in the stage one search plot (Figure 15) that shows rapid initial improvement that soon plateaus and barely improves. The sensitivity analysis that tracks the scores of the ten best scenarios of each generation also shows that few mutants get close to the best scenario's fitness score. When we examine the number of conflicts, we can see that half are with CPLoss, therefore weighted to score highly (Table 3).

[1] We have relatively little data from the three months of running experiments for D3.2, so unfortunately we are only able to give approximate estimates for the outcomes of search runs such as this.

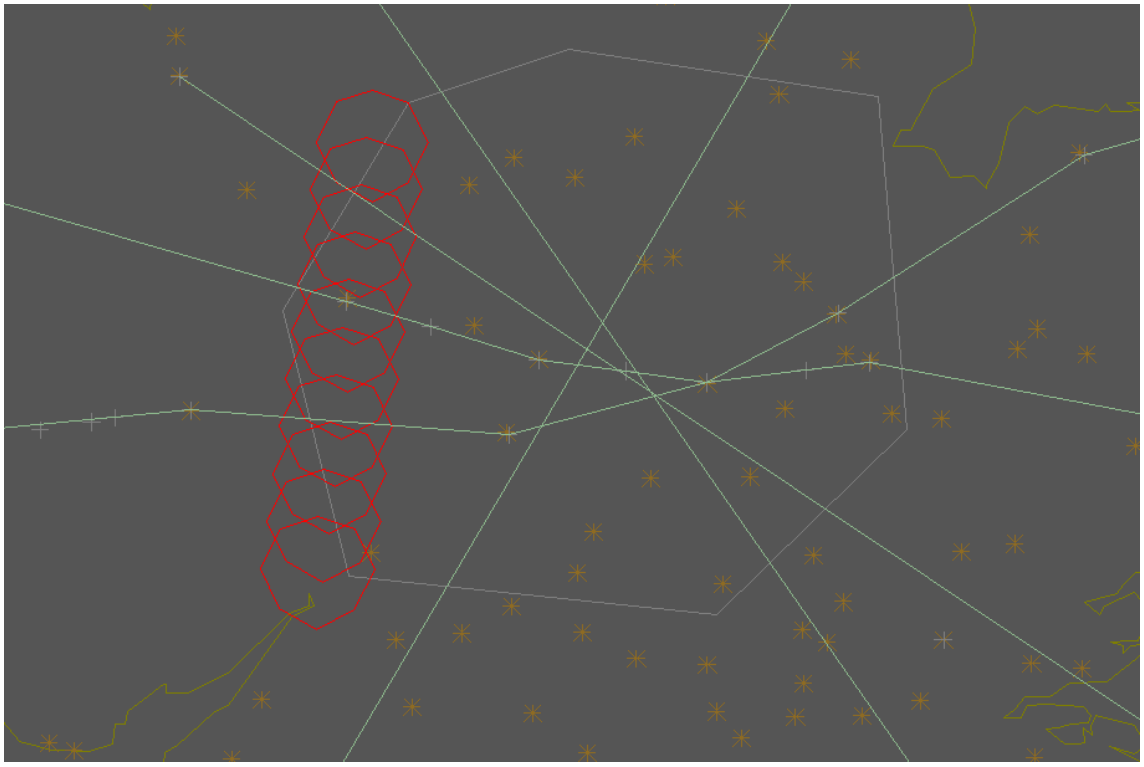


Figure 12: Storm trajectory (red polygons) for low scoring scenario in Example 4.

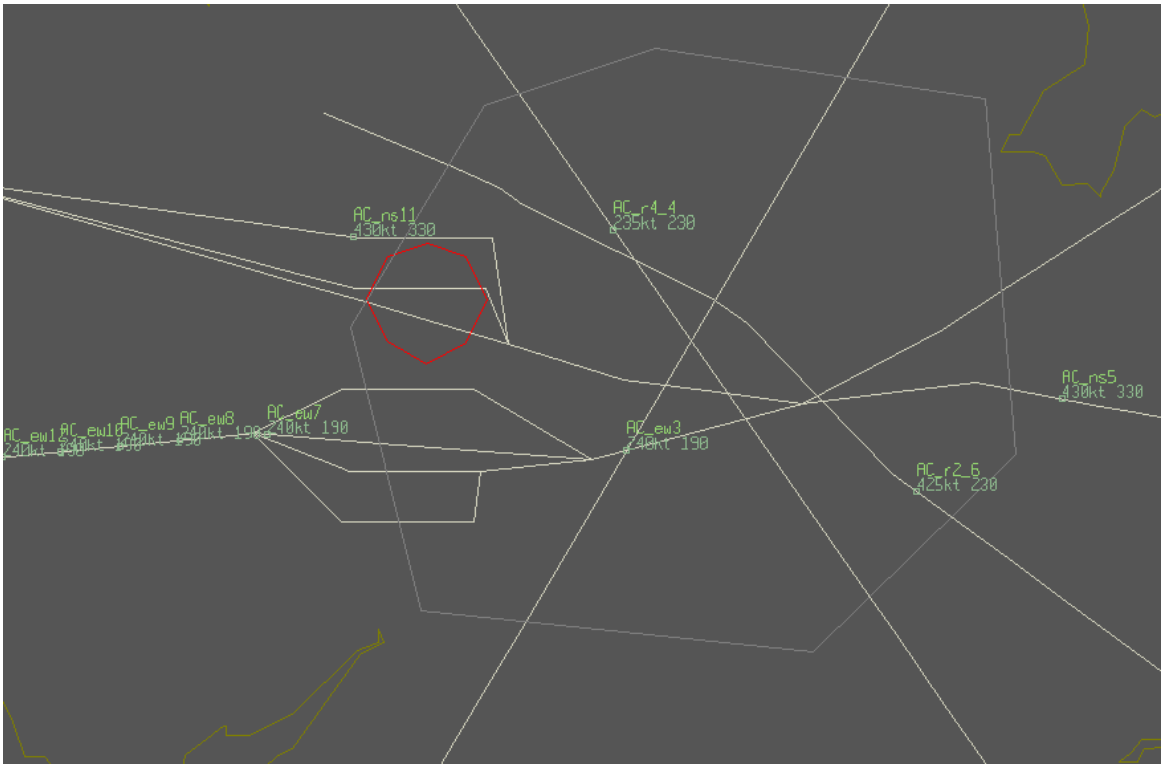


Figure 13: Storm crosses ns flight path.

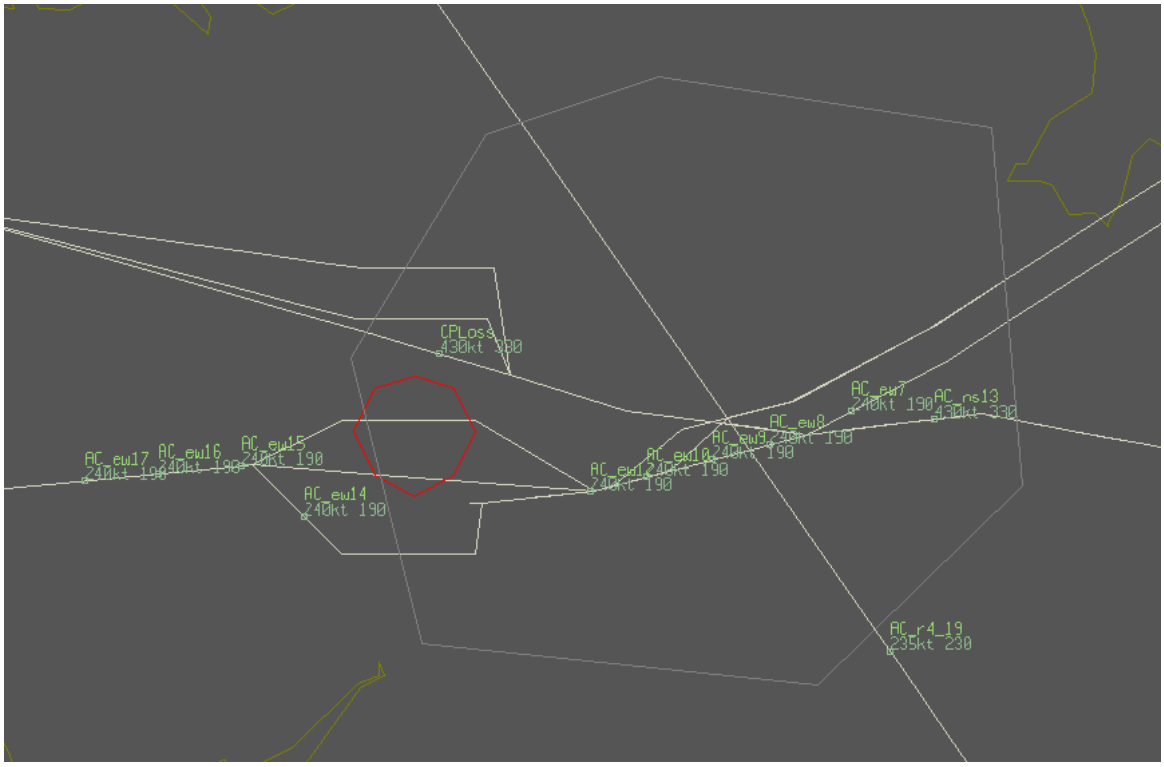


Figure 14: Storm cross ew flight path.

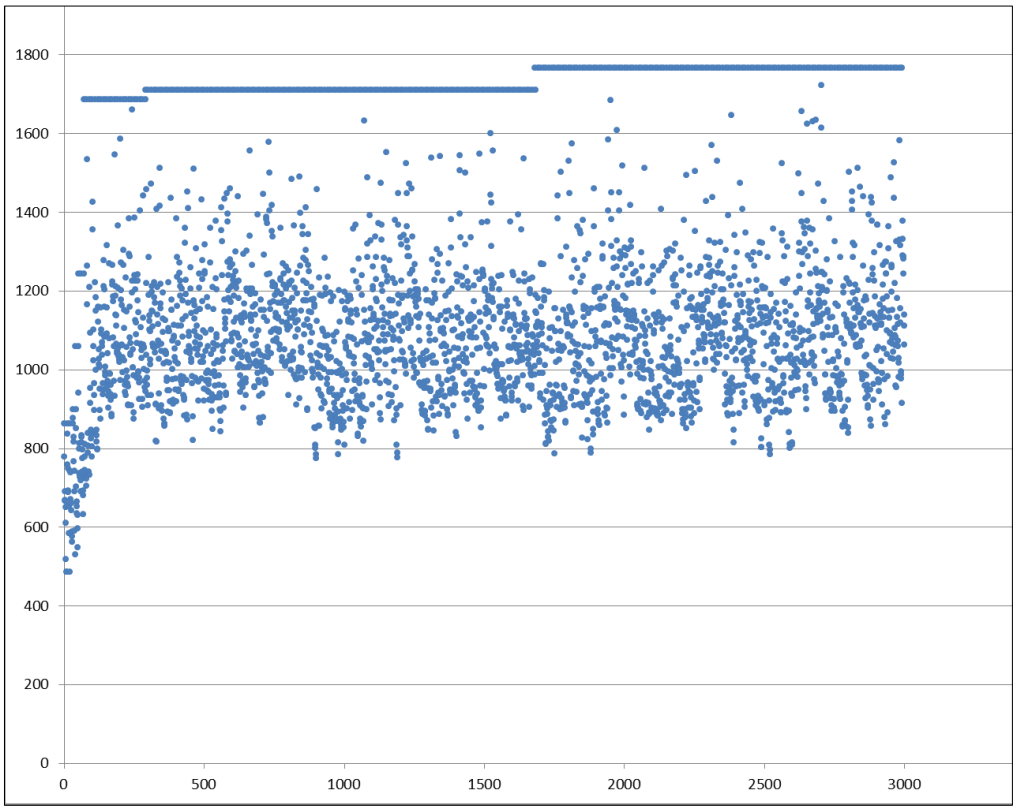


Figure 15: Stage one plot showing sensitivity scores for ten best in generation in Example 4.

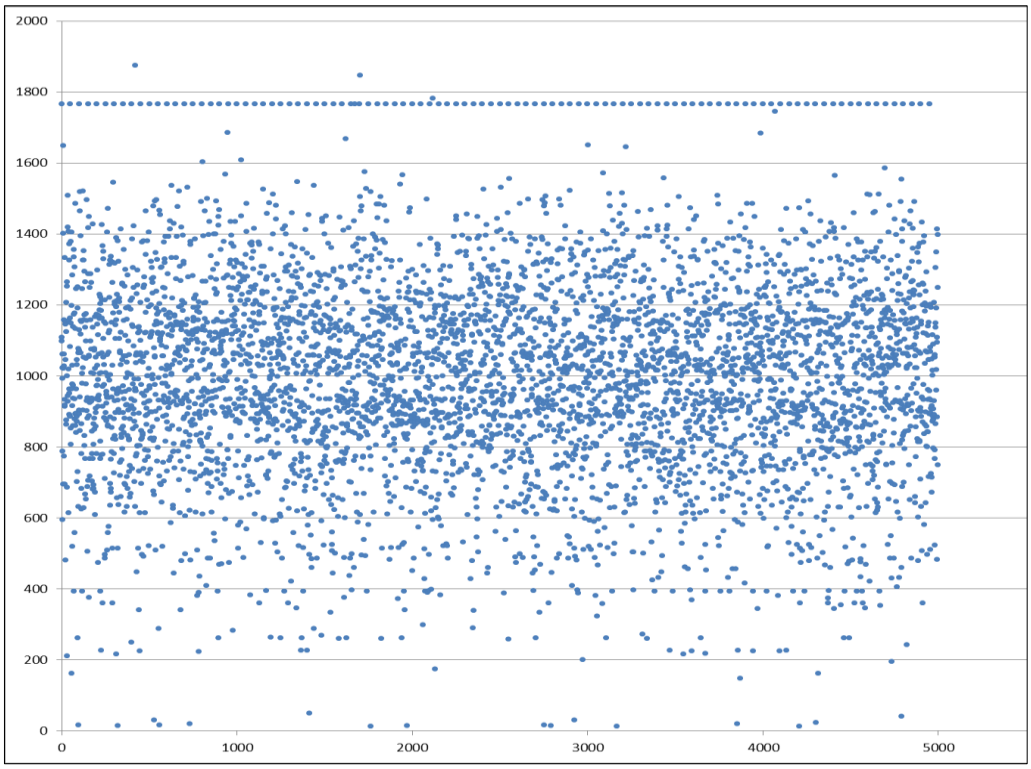


Figure 16: Second stage sampling around near neighbourhood of low scoring scenario from Example 4.

Table 3: Conflict summary table of highest ranked scenario from stage one search (Figure 15).

Flight1	Detection Time	AC Model	Category	CP AAlt	Cf Attitude	Cf Speed	Flight2	CP AAlt	Cf Attitude	Cf Speed	Dist To Cf	Cf Start	Cf End	Severity Avail Sep
AC_r2_6	00:33:11	B737	Normal	230	Cruise	425	**AC_r4_4**	230	Cruise	235	32.78	00:41:36	00:42:29	0.408
CPLoss	01:03:55	A320E	Emergency	180	Descent	350	**AC_ew10**	190	Cruise	240	49.49	01:16:17	01:16:41	1.017
CPLoss	01:03:55	A320E	Emergency	200	Descent	390	**AC_ew12**	190	Cruise	240	49.49	01:16:17	01:16:41	0.711
AC_ew15	01:08:14	DH8	Normal	190	Cruise	240	**AC_ew14**	190	Cruise	240	44.64	01:19:24	01:40:40	0.794

Table 4: Conflict summary table of highest ranked scenario from stage two sampling (Figure 16).

Flight1	Detection Time	AC Model	Category	CP AAlt	Cf Attitude	Cf Speed	Flight2	CP AAlt	Cf Attitude	Cf Speed	Dist To Cf	Cf Start	Cf End	Severity Avail Sep
AC_r2_6	00:33:06	B737	Normal	230	Cruise	425	**AC_r4_4**	230	Cruise	235	31.5	00:41:10	00:42:04	0.251
CPLoss	01:03:43	A320E	Emergency	200	Descent	390	**AC_ew12**	190	Cruise	240	49.49	01:16:05	01:16:28	0.829
CPLoss	01:03:43	A320E	Emergency	180	Descent	350	**AC_ew10**	190	Cruise	240	49.49	01:16:05	01:16:29	0.961
AC_ew15	01:08:46	DH8	Normal	190	Cruise	240	**AC_ew14**	190	Cruise	240	48.36	01:20:52	01:21:35	0.991

We can now look at the second stage plot of this example shown in Figure 16. This shows there were just 3 scenarios out of 5000 sampled in the near neighbourhood with a higher fitness score (i.e. having a greater risk profile) than the stage one result. We will take for comparison the highest scoring of these, but it should be noted that no scenario scores significantly more than the stage one result.

When we compare the conflict summary outputs for the two scenarios (Table 3 and Table 4), we can see that the slight variation in traffic entry times for those aircraft in conflict has made a marginal difference to the severity of available separation (final column, highlighted in yellow – a smaller value is more severe). However, the difference in traffic entry times has not increased the total number of conflicts, neither has it changed the nature of the existing conflicts. It would appear that CPLoss comes into conflict with both AC_ew10 and AC_ew12 in both scenarios at the same time, as the two aircraft on the ew flight path are minimally separated. The single factor responsible for the difference in fitness scores is probably the reduced percentage of available separation between AC_r2_6 and AC_r4_4. As this factor is weighted in the heuristics (by applying a logarithmic function), a smaller figure can account for the difference in fitness score between the stage one and stage two scenarios. This is backed up by examining the ATCo's total number of tasks and the NASA complexity rating of the two scenarios, both of which remain identical. So while the increased severity of the conflict between AC_r2_6 and AC_r4_4 might be something the safety analysts would want to examine more closely, overall the slight increase in risk between the stage one and stage two scenarios does not seem to be serious, as the resolution of the conflict remains the same and the ATCo's workload is unaffected by any external factors.

3.1.3 No higher ranking variants found

Beyond the four example runs discussed above, we performed a number of runs in which we did not find any scenarios with higher fitness scores in the stage two sampling part of the process. We have not described these in detail as there seems little to say, other than that the first stage search may have managed to find the worst case scenario (at least in the size of neighbourhood covered by the second stage samples).

However, as discussed in Section 2.1 and 2.2, this could be because for these instances we needed to use either a larger neighbourhood size or a higher number of samples in the second stage, but we made the decision to keep both of these parameters constant as we had little time to conduct the final series of experiments and changing the size would have made comparisons difficult between runs. If there had been more time on the project this is a hypothesis we would have tested.

4 Issues with the two-stage search process

The two stage search process introduced by ASHiCS brings with it several benefits that could not be achieved using evolutionary search alone in such a large, complex solution space. Our desire to create a technique of practical benefit to safety analysts was the principal driver to include the second stage sampling around the near neighbourhood of the first stage result. As the examples we have given show, the second stage serves both to give a level of assurance over the evolutionary search's performance in finding the worst case scenario, but it also permits a comparison of near variants. This second feature is likely to be of most benefit to domain experts, as they have the knowledge to set up scenarios with a view to constrain the total solution space to those hazards of particular interest to themselves and they are then able to make rapid comparisons between high risk variants using their domain knowledge.

However, generating the high risk variants of the stage one search result comes with a cost. The two stages are almost like running one search after another (though this depends on how detailed the sampling process is in stage 2), and so the length of time to come to the results nearly doubles. We found that using a PC running Windows 7 64-bit with Intel Core 2 Duo CPU at 3GHz with 4GB RAM took between 36-40 hours to complete a run, with our smaller machine that ran in parallel taking a little longer (~ 48 hours).

A second issue illustrated by Example 2 is the need to be familiar with how RAMS Plus reports conflicts and other outputs that can be used to measure levels of risk in the sector. The design of scenarios, flight paths and aircraft must be done so that (in particular) the first stage evolutionary search is not misled by figures that are not relevant to the hazards or areas of risk being modelled. We would assume that domain experts would be sufficiently familiar with RAMS Plus to avoid this type of error during the design of their fitness functions. We are aware that many of the reporting capabilities of RAMS Plus can be customised to suit the analyst, but we do not have the level of expertise to investigate this option (other than for the workload assessments in the compound fitness function, discussed in D2.1).

Finally, there is the issue of how large the near neighbourhood should be for the second stage sampling around the stage one result and the number of samples that should be taken. As discussed in Sections 2.1 and 2.2, part of this will require some domain expertise to analyse the results of a trial run of the second stage to look at any high risk variants that are found and decide whether the near neighbourhood size was so large that the nature of the scenario, aircraft timings or conflicts could be radically changed as a result. This requires not only the domain experts to go through trial runs in detail, but it also means that there is unlikely to be a formula or rule that can determine the ideal size of near neighbourhood for all types of air sector. Deciding the number of samples is easier, as this can be increased until variants of similar or greater levels of risk are found (see final paragraph of Section 2.2 for suggestions on automating this).

5 Contrast with Closely-Related Work

There are two areas of work that are closely related to in ASHiCS. First, there is a large body of work using some form of Monte Carlo simulation for quantitative risk analysis. The most prominent researchers in this are at the National Aerospace Laboratory of the Netherlands (NLR). This differs from ASHiCS in its objectives and methods. Second, there is a smaller but growing body of work on evolutionary safety analysis, dominated by Hussein Abbass and Sameer Alam at the University of New South Wales in Australia (although much of the recent work has been in tandem with Eurocontrol). This work is very similar to ASHiCS in terms of basic goal and approach, differing only in variations of technique.

5.1 Monte Carlo Simulation for Risk Analysis

Researchers at NLR have spent many years developing their TOPAZ safety analysis methodology which has, at its core, a heavy reliance on agent-based simulation using Monte Carlo (MC) analysis to derive quantitative results. These results include both headline risk figures (e.g. collision probability per hour in the airspace sector modelled) and contributors to that (e.g. how much of that is contributed to that by failures of the TCAS system). A recent paper that explains the basic approach well (including how they adapt MC analysis to deal with rare events) and demonstrates its advantages over a simpler event-tree approach is Stroeve et al [18].

ASHiCS has slightly different goals to MC analysis, and on the surface less ambitious ones. While MC aims to produce an overall assessment of how dangerous the system is (and how various things contribute to that), ASHiCS merely aims to discover all the ways in which high-risk situations can occur. We say “merely”, but in practice MC analysis risks missing some rare causes and hence some rare events – the space being sampled from is enormous, and MC can only (stochastically) sample a subset of it.

Will ASHiCS-like methods consistently discover hazards that MC analysis misses? That is an empirical question, and one that we have not been able to answer in this short study. Modern MC analysis is a complex and nuanced activity, and to do it well requires significant expertise. Ideally, any experiments on this issue would be conducted by a consortium involving NLR and a team of heuristic search experts.

One way that MC and ASHiCS could be combined would be to use MC repetition of individual runs. In this project, each individual in an ASHiCS population has been run once, with RAMS configured to give exactly the same results each time. If stochastic variation points were added within a run (for example, as a random variation in the time a controller takes to perform a given action, or the precise route he puts an aircraft on) then each individual could have its fitness scored based on the distribution of outcomes. This could improve the representativeness of the scores assigned to individuals. This would avoid one weakness of the work in this project, in that our deterministic scenarios may totally exclude certain dynamics because they’re always “possible” but never the “most likely” that we’ve set up our deterministic situation for. It would, of course, be very computationally expensive.

A second way to combine the two would be to search over coarse-grained deterministic simulations in an ASHiCS vein, then zoom into identified high risk areas and perform detailed MC analysis. This is of course the two-stage variant of the ASHiCS process as described in the report, but there is a wealth of existing MC work that could be drawn on to make the second stage more thorough than we have done here. Potentially, this would allow the ASHiCS search to identify risk hot-spots and the MC to thoroughly explore them.

This second approach, however, abandons one of the great advantages of the MC approach – that they can make a wide variety of statistical claims about how well they have explored the whole space and therefore can give overall risk statistics. These statistics are extremely valuable in any safety-decision-making situations. There are of course a wide range of ways in with quantitative risk analysis can go wrong [19] but using skilled people and good techniques it is possible to get around many of these (one of the authors (Alexander) is helping to develop a maturity model for quantitative risk assessment, which should be published in Autumn 2013).

A third approach would be to invert the previous suggestion and use MC over the whole search space followed by search from selected points within that space. Such a search may reveal that had the MC

been configured slightly differently it could have produced an appreciably higher risk estimate, which of course would raise doubts about the validity of the MC's original results (it would reduce our assurance of its validity). Conversely, if the search *cannot* find additional risk then we have a little extra assurance that the original risk estimate is valid. Now, we cannot easily *quantify* the increase or decrease in assurance, and it is not obvious how we should select the precise starting points for the search, but this combination has potential for the future.

5.2 Multiobjective Evolutionary-Based Risk Assessment (MEBRA)

Hussein Abbass and Sameer Alam at the University of New South Wales at Canberra have developed an approach to using heuristic search for risk assessment[2]. Like ASHiCS, one of their application areas has been ATM. In a 2009 paper [20] they call this class of activities Multiobjective Evolutionary-Based Risk Assessment (MEBRA) – by their definition, the ASHiCS work is a form of MEBRA. More recently, they appear to have focussed on a variant known as Computational Red Teaming (CRT) which co-evolves the state of a system (in particular, the vulnerabilities of that system e.g. an airport having some runways unavailable) with a set of threats or demands from its environment (e.g. aircraft arriving at an airport and wanting to land).

Specific work by Abbass et al includes identifying patterns in arrival traffic and ground events that led to delays in dynamic continuous descent arrivals scenarios [13] and exploring how controller actions can lead to risk [12] (in the latter they directly evolve sequences of controller actions that lead to high-risk situations). The latter work could be useful for determining when changes to systems or procedures have left controllers within a few minor errors of very bad situations.

Overall, the MEBRA/CRT work is very similar to ASHICS in general approach, although specific case studies used are different. In terms of technique, there is a slight difference in that they favour co-evolution whereas we have used a conventional heuristic search with a single fitness function.

In the MEBRA overview paper [20] they authors explicitly distance themselves from quantitative risk analysis (e.g. deriving overall probabilities of catastrophic events) and position their work as a means to identify qualitative ways in which risks and occur – to identify sources of risk. They justify this as a response to a modern perception of "deep uncertainty" in complex systems; they contrast this with an earlier assumption that systems could be modelled by "branching points" with knowable probabilities. The arguments they present are not entirely convincing – it is certainly valuable to identify many diverse sources of risk in a system, but the questions that we must answer still have the same forms: "Is this new system design safe enough to justify deploying?" or "How much effort should we spend on risk analysis?". This is unlikely to change, but it is these questions that motivate safety analysis in the first place. The question MEBRA (and ASHiCS) asks, "What are the distinct qualitative ways in which high-risk situations can occur in the system?" must be answered as part of answering those other questions, and has value beyond that (e.g. its answer allows us to improve safety by fixing or managing the causal networks we find), but answering it is not sufficient in itself. In ASHiCS we have attempted to make some progress on this (through the second stage of our process), and it remains a crucial area for further development.

The MEBRA overview paper remains a good introduction to the general field in which ASHiCS sits. (Their account of the computational needs of MEBRA/CRT work is overly pessimistic, however – their example costing ignores a 3000-fold opportunity for parallelism.)

The MEBRA team at New South Wales have not yet published any results in the same precise area as ASHiCS, but their approach is highly applicable here and there are probably many synergies between the the two approaches that could be explored.

[2] Within in Eurocontrol, this is often referred to as "the Canberra work"

6 Conclusions

The ASHiCS project has presented a two stage search technique that allows safety analysts to use evolutionary search to automatically discover hazards within ATC simulations. One of the problems of using search to discover hazards in high dimensional solution spaces is the inability to provide guarantees related to the search performance (i.e. did it find the worst case scenario?) or frequency information (i.e. how many other similarly 'bad' scenarios are there?).

This lack of context to search results means that traditional safety assessment of those hazards (using fault trees and event probabilities) is not possible. However, contextual information that provides both simulated event frequency (by extrapolating from the random samples) and variant severity can be retrieved by extensive random sampling within the near neighbourhood of the original result. The second stage of the ASHiCS search process provides some insight to the nature of the risk landscape in terms of how other high risk variants vary from the original result. By analysing the variants, safety analysts can focus on the parameter range that generates the worst cases and can then investigate how to prevent that configuration of inputs leading to a hazard in the air sector being modelled.

Some of the results reported in this deliverable were written up and accepted as a poster and short paper in the Industrial Experience track at GECCO 2013.

7 References

[1] K. Clegg and R. Alexander, “Searching for Risk in Large Complex Spaces,” in *EvoStar 2013, Vienna, Austria*, 2013.

[2] P. Kopardekar and S. Magyarits, “Dynamic density: measuring and predicting sector complexity [ATC],” 2002.

[3] K. De Jong and W. Spears, “An analysis of the interacting roles of population size and crossover in genetic algorithms,” in *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 1991, pp. 38-47.

[4] D. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley, 1989, pp. Goldberg, D. (1989). Genetic algorithms in search, optimization, and machine learning. Addison-Wesley.

[5] J. R. Koza, M. Keane and M. Streeter, “Evolving inventions.,” *Scientific American,* p. 52–59., 2003.

[6] C. F. Lima, D. E. Goldberg, K. Sastry and F. G. Lobo, “Combining competent crossover and mutation operators: A probabilistic model building approach,” in *Proceedings of the 2005 conference on Genetic and evolutionary computation (GECCO '05)*, New York, 2005.

[7] S. Borenor, “Innaxis Complex World - Key note presentation: Complexity and safety performance - Will circular accident causality result from NextGen and SESAR?,” 2011. [Online]. Available: http://complexworld.innaxis.org/presentations/borener.pdf.

[8] D. White and S. Poulding, “A rigorous evaluation of crossover and mutation in genetic programming,” in *12th European Conference, EuroGP 2009*, Tübingen, Germany, 2009.

[9] A. Yousefi, R. Hoffman, M. Lowther, B. Khorrami and H. Hackney, “Trigger Metrics for Dynamic Airspace Configuration,” in *9th AIAA ATIO Conference*, 2009.

[10] ISA Software, *RAMS Plus User Manual Version 5.36,* 2011.

[11] E. Perrin, B. Kirwan and R. Stroup, “A Systemic model of ATM Safety: the integrated risk picture,” in *7th ATM Seminar*, Barcelona, 2007.

[12] S. Alam, C. Lokan and H. Abbass, “What can make an airspace unsafe? characterizing collision risk using multi-objective optimization,” in *IEEE Congress on Evolutionary Computation (CEC), 2012*, 2012.

[13] S. Alam, W. Zhao and J. Tang, “Discovering Delay Patterns in Arrival Traffic with Dynamic Continuous Descent Approaches using Co-Evolutionary Red Teaming,” in *9th ATM Seminar*, Berlin, 2011.

[14] ARMS Working Group, 2007-2010, “The ARMS Methodology for Operational Risk Assessment in Aviation Organisations,” ARMS Working Group, v 4.1, March 2010.

[15] K. Clegg and R. Alexander, “ASHiCS E.02.05 Method Description Technical Report,” Available online at: http://www-users.cs.york.ac.uk/~kester/ashics-deliverables/E.02.05-ASHiCS-D2.2-Method%20Description%20Technical%20Report-V3.pdf, 2012.

[16] K. Beyer, J. Goldstein, R. Ramakrishnan and U. Shaft, “When Is “Nearest Neighbor” Meaningful?,” in *Database Theory — ICDT’99: Lecture Notes in Computer Science*, Berlin , Springer , 1999, pp. 217-235.

[17] P. Merz and B. Freisleben, “On the effectiveness of evolutionary search in high-dimensional NK-landscapes,” in *Evolutionary Computation Proceedings, IEEE World Congress on Computational Intelligence*, 1998.

[18] Stroeve, S.H., Blom, H.A.P., Bakker, G.J.: Contrasting safety assessments of a runway incursion scenario: Event sequence analysis versus multi-agent dynamic risk modelling. Reliability Engineering & System Safety 109, 133-149 (2012)

[19] Rae, A., McDermid, J., Alexander, R.: The Science and Superstition of Quantitative Risk Assessment Proceedings of PSAM 11 & ESREL 2012, June 2012 (2012)

[20] Abbass, H., Alam, S., Bender, A.: MEBRA: multiobjective evolutionary-based risk assessment. IEEE Computational Intelligence Magazine 4, 29-36 (2009)

-END OF DOCUMENT-