

Sensitivity of an Automated Separation Assurance Tool to Trajectory Uncertainty

Andrew C. Cone¹, David Chin²
NASA Ames Research Center, Moffett Field, CA, 94035

The effects of uncertainty are examined on a Center-wide, ground-based automated conflict detection and resolution algorithm at one, two and three times current traffic demand level. The required horizontal separation radius is varied between four and nine nautical miles as a proxy for trajectory uncertainty. Sensitivity of the algorithms' performance to demand and uncertainty is studied in fast-time simulation of the Cleveland Center airspace. Results show the performance of the conflict detection and resolution algorithms is predictable across all cases: when the horizontal separation radius is reduced, performance improves; when the demand level is increased, performance degrades. There is a significant decrease in performance in terms of average delay per resolution in the cases combining the highest demand level and horizontal separation radius. Algorithm performance degradation is first observed in the arrival environment, where constraints are most prevalent. The algorithms show no major performance degradation for conflicts occurring outside of the arrival environment.

I. Introduction

Current forecasts predict a steady growth in the demand for air travel across the National Airspace System (NAS), which in its present form is already impacted by capacity constraints. As one of those en-route capacity constraints is controller cognitive workload, research in both Europe and the U.S. into automated separation assurance systems is being conducted as a means of increasing the en-route capacity to help meet the anticipated growth in demand. Among the inherent challenges is understanding how such systems perform in the presence of uncertainty. Most studies to date focus on the overall performance of automated separation assurance tools using deterministic trajectory models,^{1,2,3} though some recent studies look at the effects of specific sources of uncertainty, such as wind, weight, and pilot response delays in these models.^{4,5} None of these studies, however, is a sensitivity study that can inform system designers of the operating limits of their prototype separation assurance systems. The present work seeks to provide such feedback.

This paper describes an experiment that studies the sensitivity of an automated separation assurance tool to the presence of horizontal trajectory uncertainty. This study is limited to the effects of horizontal uncertainty only, and horizontal separation standards in particular, so as not to overlap with concurrent work focusing on vertical uncertainty.⁶ By evaluating a range of horizontal separation standards and traffic demand levels, a general picture of the cost/benefit landscape in terms of the performance of the automated separation assurance tool could be created. This would help answer some basic questions about what the presence of uncertainty would mean for a system like this. First, assuming five nmi as a nominal standard, what is the benefit of decreasing horizontal separation assurance standards, and how do those benefits scale with traffic demand level? It is logical to assume that lower horizontal separation requirements, which would correspond to less trajectory uncertainty, would produce some benefits to the system. The scope of those benefits, however, is currently unknown. Second, again assuming five nmi as a nominal standard, what is the cost of adding extra buffer to the required horizontal separation standard, and how does it scale with traffic demand level? This would correspond to less stringent requirements for things like aircraft equipage, which might translate to significant cost savings for an airline. However, if the cost in terms of such things as delays and resolution efficiency becomes large enough, then an argument for requiring improved equipage becomes easier to make. Additionally, if the current standard of five nmi remains unchanged, the implementation of an automated separation assurance tool would almost certainly require the addition of some

¹ Aerospace Engineer, Aviation Systems Division, M/S 210-10, and AIAA Member

² Engineering Project Manager, Aviation Systems Division, M/S 243-5

buffer to account for uncertainty effects. But is there a point at which the safety benefit of an expanded buffer is overwhelmed by the attendant performance costs?

This study examines the effects of trajectory uncertainty in the strategic environment on the performance of a specific automated separation assurance tool. “Strategic” is defined as 2 to 20 minutes until predicted loss of separation. The automated separation assurance tool evaluated in this study is the Advanced Airspace Concept, or AAC.⁷ This is a centralized, ground-based concept for a conflict detection and resolution system. It assumes the existence of data communications to support the regular downlink of position and intent information and periodic uplink of trajectory amendments. In this simulation we do not model trajectory uncertainty explicitly, as we are more interested in the collective effect of all uncertainties on the performance of an automated separation assurance tool than a specific source of uncertainty. Instead, we vary the horizontal separation standard as a proxy for all sources of trajectory uncertainty and study the effects those adjustments have on the performance of the algorithms. Performance is defined by safety and efficiency metrics.

II. Technical Approach

The experiment was run in the Airspace Concepts Evaluation System (ACES)⁸ simulation environment, using the Advanced Airspace Concept (AAC) as the ground-based automated separation assurance tool. ACES is a fast-time simulation capable of simulating the entire NAS, though for this experiment it was decided to use a single center, Cleveland Center, for simplicity’s sake. AAC, as mentioned in the introduction, was the automated conflict resolver used in this study. It received a list of conflicts from the conflict detection algorithm in ACES every two minutes, and sent back resolutions in the form of flight plan updates. The version of AAC used in this study utilized what is called the “multiple resolver.” This means that AAC did not take the first successful resolution it found for a specific conflict. Instead it would try to find a variety of resolutions, such as a left turn, right turn, path offset, altitude adjustment, and/or speed adjustment, then pick the “best” of the successful resolutions to send back to ACES as the actual resolution maneuver. “Best,” in this case, meant the least delay resolution. This tended to drive AAC to vertical maneuvers when they were available, as they often have little impact on delay. This version of AAC also had a scheduling algorithm that was used in arrival cases. This algorithm was a “first come, first serve” type that sought to keep aircraft headed into a terminal fix at least one minute apart. This helped ensure a relatively smooth flow of traffic into these fixes. In one sense, the scheduler preconditions the arrival traffic, which makes the separation assurance algorithm’s job much simpler than it would be without the scheduler. However, Traffic Flow Management is a concept that is already seeing acceptance in the field, and it seems a logical assumption that such technology will be available for integration with an automated separation assurance tool when a system such as AAC begins to see use in the real world. With that said, it must be noted that the results of the study, especially in the arrival traffic environment, would be quite different without the presence of some kind of scheduling algorithm.

The conflicts in this study can be broken into two main categories: arrival conflicts and everything else. The non-arrival cases include overflights in addition to cases where one aircraft is passing through an arrival stream. Arrival conflicts, also called “merging arrivals,” are defined by Farley as “projected conflicts between two aircraft on arrival to the same arrival fix and within 20 minutes flying time of that fix.”¹ For these data, the definition of arrival is expanded to include all aircraft heading for the same arrival airport. It might be asked why the arrivals case is treated separately when it involves such a limited portion of the data set. The reason is that the arrival conflict cases are often quite complicated to solve, as not only are aircraft often in a descent for at least part of the conflict, but the fact that they are so near to their final fix can limit the maneuvering options available, as some of them would not allow the aircraft to make it back to its flight plan in time to hit that fix. Additionally, there are often other aircraft in the immediate airspace around the terminal, which increases the likelihood that a trial resolution might be rejected because it would cause a secondary conflict. For those reasons, arrival conflicts are treated as their own subset of conflicts, while everything else is grouped in the non-arrival bin. The conflict detection algorithm also treats these two groups of conflicts differently, scanning ahead 8 minutes for non-arrival conflicts, 20 for arrivals. Finally, aircraft that are maneuvered by the scheduler are not included in either conflict bin. Because the scheduler is integrated with AAC, it can use the conflict resolution logic to check potential scheduling maneuvers to ensure that they are conflict free before sending the maneuver off to ACES and the aircraft in question.

The experiments involved changing the required horizontal separation for the conflict detection and resolution algorithms. The horizontal separation radii used for the detection algorithm in this study were four, five, six, and nine nmi. The case with a horizontal separation radius of five nmi for the conflict detection algorithm was treated as the baseline, as that radius is the current minimum standard. As such, four and six were used to note the effects of variations around this case. Nine nmi was chosen mostly to stress system. We wanted an upper end to our study that was larger than was likely to be seen in a real world implementation, while not being so large as to be completely

beyond the scope of possibility. A two nautical mile buffer was added to the conflict detection radius for the resolution logic in all cases, as in the work by Farley, Kupfer, and Erzberger¹. This was done to account for the real world likelihood that a resolved conflict may reoccur during the course of the resolution due to uncertainties such as wind. For a 10 minute resolution trajectory, two nmi was seen to greatly reduce the chance of this occurring. Adding too many miles to this buffer might adversely affect the performance of AAC, while too few might not adequately reduce the chance of a conflict reoccurring. Therefore, the baseline case, for example, uses five nmi of horizontal separation for detection and seven for resolution. Cases were run in series on the same computer using the same version of ACES and AAC.

Additional conditions included using 960 feet of vertical separation for the conflict detection algorithm for all cases. The vertical separation of just under 1,000 feet was used for coding concerns. There was no buffer added to the vertical separation criterion, so the same value was used for both the detection and resolution algorithms. The traffic data for the 1X case were based on 24 hours of traffic from a single day; April 19, 2007. The traffic demand sets for 2X and 3X cases were created from this initial day using AvDemand.⁹ These demand sets are the same ones used in Kupfer’s recent work². Due to constraints within the simulation, such as the lack of any control at the terminal level, the arrival traffic was manually set to 1.5X for hub airport traffic for both the 2X and 3X cases. This was done by examining all of the aircraft in the 2X and 3X cases that were arriving at hub airports in the NAS, and reducing that number in the input file until it matched a rate of 1.5 times the rate in the 1X case. There were 7,714 total active flights in the 1X case, 15,066 in 2X, and 19,686 in 3X.

The four main metrics of interest were the number of resolutions, the number of unresolved conflicts, the average delay per resolution, and the average number of attempts required to find a resolution. Comparing the number of resolutions and the number of unresolved conflicts helps give us an idea of how complicated the airspace became and how well the system performed overall. The overall point of a separation assurance system is to maintain adequate separation between aircraft, whatever that separation is deemed to be. The average delay helped us understand how efficiently the system ran. Less delay per resolution means less disruption to the originally filed flight plans, which can also help with things such as minimizing fuel burn for a flight. The fourth metric was the number of attempts required to find a resolution. This metric is used to describe “workload,” which in this paper refers to how much effort is required to find a solution for an individual conflict. There are a limited set of possible resolutions available for AAC to use to solve a given conflict. As the number of attempts required to find a successful resolution increases, the likelihood of seeing unresolved conflicts or high delay resolutions also increases. In practical terms, the behavior of this metric can indicate how close AAC is to the edge of its operating envelope.

III. Results

The following table summarizes the general results of this sensitivity study. As mentioned in section II, the two independent variables were required horizontal separation for the conflict detection algorithm and the traffic demand level. The three main metrics of interest were the number of resolutions and unresolved conflicts, the average delay per resolution, and the average number of attempts required to find a resolution.

Table 1. Summary of general results

Traffic Demand Level	Required Horizontal Separation (nmi)	Resolutions	Unresolved Conflicts	Average Delay per Resolution (min:s)	Average Attempts per Resolution
1x	4	547	0	00:07	17.8
2x	4	1822	0	00:11	18.2
3x	4	2317	0	00:29	18.4
1x	5	714	0	00:11	19.8
2x	5	2472	0	00:13	20.1
3x	5	3021	0	00:25	20.2
1x	6	951	0	00:15	21.6
2x	6	3115	0	00:21	21.7
3x	6	4035	1	00:32	22.0
1x	9	1701	0	00:29	25.6
2x	9	5827	2	00:40	28.1
3x	9	7688	2	00:59	28.9

The results in Table 1 clearly show that increasing both density and required horizontal separation results in a greater number of conflicts, a larger average delay per resolution, and more resolution attempts to find a solution.

There are a few interesting points from this table. First would be the non-linear scaling of the number of resolutions with both traffic demand level and the separation criteria. Moving from five nmi to six, for example, increases the number of resolutions issued for all demand levels by roughly 30%. However, changing the radius from five nmi to nine produces an almost 140% increase in the number of conflicts at 1X and 2X, and over 150% at 3X. Holding separation radius fixed while moving traffic demand levels has a similar effect. The point to note from this column is that the horizontal separation radius and traffic demand level have mostly independent effects on the number of resolutions. It was also encouraging to note that the failure cases only appear in the three cases with the highest resolution counts. For the overall numbers, the percentage of successful resolutions from AAC was over 99.97% for all tested demand levels and horizontal separation radii.

The second point is that the average delay per resolution follows a generally upward trend for both increasing horizontal separation standards and traffic demand level. There is a notable exception to this when moving from four nmi to five at three times traffic density. This will be examined in more detail later in this paper. For now, it should be pointed out that the gross effect of increasing both the horizontal separation standards and the traffic demand level is to increase delays, and that increasing both of these factors together can result in some significant delay increases.

The last point on this table is that the number of resolution attempts required to find a suitable resolution is much more sensitive to required horizontal separation than for changes in traffic demand level. However, it should be noted that the nine nautical mile case seems to show slightly more sensitivity to demand than the other cases. This result further supports the idea that the resolution algorithm is relatively stable at lower separation radii, though once the separation radius passes a certain point, the effects of the demand level begin to impact the performance of the algorithm.

A. Non-Arrival Conflicts

The data for the non-arrival conflicts are shown below in Table 2. As mentioned previously, non-arrival conflicts consist of all conflicts that are not arrival conflicts.

Table 2. Summary of non-arrival results

Traffic Demand Level	Required Horizontal Separation (nmi)	Resolutions	Unresolved Conflicts	Average Delay per Resolution (min:s)	Average Attempts per Resolution
1x	4	476	0	-00:01	20.3
2x	4	1675	0	00:03	20.1
3x	4	2140	0	00:03	20.8
1x	5	602	0	00:02	21.3
2x	5	2236	0	00:06	21.5
3x	5	2709	0	00:05	21.7
1x	6	782	0	00:05	22.1
2x	6	2759	0	00:11	22.5
3x	6	3515	0	00:09	22.4
1x	9	1395	0	00:16	24.8
2x	9	4936	1	00:24	25.8
3x	9	6129	0	00:21	25.3

This table shows the general data for the non-arrival conflicts. The first area of interest is the resolution and unresolved conflict count. Firstly, there were a handful of unresolved conflicts that were filtered out of these cases, all of which had less than two minutes to loss of separation when first detected. These cases were caused by boundary issues within the code, such as an aircraft popping into the center airspace from a TRACON and immediately being in conflict with an aircraft descending into its final fix. As AAC was designed and intended to operate solely in a strategic environment, and as conflicts with less than two minutes to loss of separation were seen as below the lower limit of a strategic resolver, those conflicts were considered beyond the scope of this study and were filtered out. The result was a single case in which AAC was unable to resolve a conflict in the non-arrival environment. This one case was difficult as it involved both a climbing and descending flight, and had the climber passing through an arrival stream. The only other point to make about the resolution and unresolved conflict counts is the evident fact that the number of conflict resolutions required increased with increases in both traffic demand

level and horizontal separation. However, the rate at which the numbers increased moving from 2X to 3X was smaller than expected. This could be partially the result of the manual reduction of the arrival traffic to 1.5X. Though the reduced arrival rate was implemented to deal with some artifacts of the simulation near the final fix, the aircraft it removed would have otherwise been flying through the airspace and possibly getting into non-arrival conflicts on their way to their final fix. The fact that the airspace in question was Cleveland Center helps mitigate this somewhat, as that center has a large number of overflights, which needed no reduction. Still, the reduced overall traffic is a major suspect for the reason the number of resolutions is not higher in the 3X, non-arrival cases.

The second area of interest on this chart is the average delay per resolution, which does not vary much for a given horizontal separation radius, though that variation is a little larger for the six and nine nautical mile cases than the four and five. Additionally, for a fixed demand level, every extra mile of required horizontal separation increases the average delay per resolution by about three seconds. This holds true for all demand levels in the non-arrival environment. This shows that, for the non-arrival conflicts, the conflict resolver has a small sensitivity to both demand level and separation radius. While it is true an increase of a few seconds is a huge percentage increase in average delay for some of the lower horizontal separation radius cases, we felt that the absolute change in delay was a better indication of sensitivity. This result confirms the idea that, at least for non-arrival conflicts, AAC should be able to efficiently maintain separation almost regardless of the required horizontal separation and the traffic demand level.

Finally, we looked at the average resolution attempts required to find a successful resolution. These data give an indication of how hard AAC is working to find a solution for a problem. The results show little variation for a fixed horizontal separation radius, though the 1X cases take fewer attempts than the 3X cases for any given radius. Increasing that horizontal separation radius seems to add about one resolution attempt to each conflict per nautical mile added. All of these trends support the assertion that AAC is perfectly capable of handling all of the test cases we ran in the non-arrival environment. From the benefit's point of view, the resolution attempts data do not give much support to the case for smaller separation standards. While decreasing the required horizontal separation radius from five to four nmi results in fewer conflicts, once the conflict gets to the resolution algorithm, a smaller radius is not saving the algorithm much work. This result is most likely due to the multiple resolver logic. This code lets AAC attempt to find a handful of successful resolutions before choosing one. That strategy not only improves the likelihood that for a given conflict AAC will be able to find an efficient solution, but it also tends to even out the number of resolution attempts for conflicts. We believe this is more than a fair trade, as the extra work AAC might go through to find multiple solutions is more than offset by the extra efficiency of the solutions obtained.

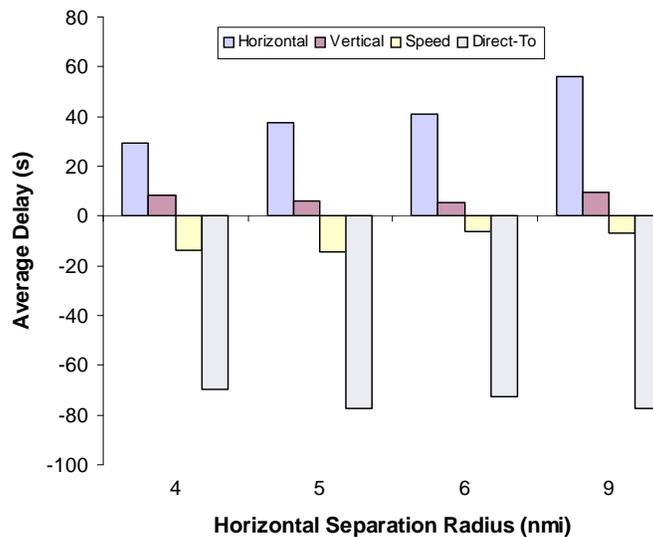


Figure 1. Non-arrival delay breakdown for all horizontal separation radii at 1X traffic demand level

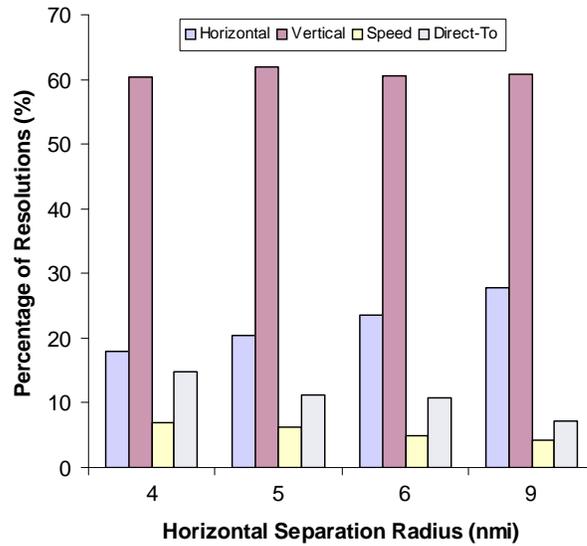


Figure 2. Resolution breakdown by maneuver type for non-arrival resolutions at 1X traffic demand level

Figures 1 and 2 show examples of how the delays break down for a given demand set. In Fig. 1 we see the average delay per resolution maneuver for different classes of maneuvers. It is no surprise that most of the high delay maneuvers are horizontal maneuvers. This is part of the reason that AAC tends to choose vertical maneuvers when it can, as they usually have significantly less delay per resolution than horizontal maneuvers. Resolution maneuvers using speed increases, though limited to an increase of 15 knots or less, are also used to save time when available. However, the number of non-arrival conflicts that are solvable with speed adjustments are quite limited, as evident in Figure 2.

The general trends of Figure 1 are not unexpected. One example would be the delay per resolution maneuver for horizontal resolution maneuvers, which increases by about 5 seconds per nautical mile added or subtracted to the required horizontal separation radius. The average delay per resolution maneuver for vertical maneuvers shows no real trend, though the average values for the 4 radii are within about a 4 second range. The fact that speed resolution maneuvers show less benefit as the horizontal radius increases is interesting, as it implies that increases in separation radius lessen the effectiveness of these maneuvers. However, the difference between the average delays of the highest and lowest cases is only about 6 seconds, which is a small variation for the basis of a conclusion. Finally, the “Direct-To” maneuvers, as with the vertical ones, do not show any definite trend. They average between about 68 and 77 seconds of time savings for each maneuver, with the highest savings per maneuver at nine nmi and the lowest at six.

Those “Direct-To”, or “D2”, maneuvers are one of the more interesting features of these figures. These are resolution maneuvers that send an aircraft to some fix further along in its flight plan. These maneuvers result in both the conflict being resolved and the aircraft saving time by skipping over part of its original flight plan. Technically, these maneuvers take place in the horizontal plane. However, unlike the other horizontal resolution maneuvers, which also take an aircraft off of its original flight, D2 maneuvers do not try to immediately bring an aircraft back to its flight plan once the conflict is cleared. It was this fundamental difference in approach, as well as the vastly differing delay per resolution results, that justified separating D2 maneuvers from other horizontal resolutions. As these D2 maneuvers can be difficult to find, especially in more crowded airspace, there are not many as a percentage of the overall number of resolutions. Nevertheless, even at very high separation radii these maneuvers save a lot of time on a per resolution basis. Of course, as the horizontal separation radius increases, the number of D2s available decreases, so the time savings as applied to an average delay per resolution decreases even though the data show the time savings per D2 is higher at nine nmi than it is at four. These maneuvers are the reason there was a net negative delay per resolution for the 1x, four nautical mile case, as a handful of D2 maneuvers can accumulate a very large time savings. In fact, in every case we ran, D2 maneuvers averaged around a minute of time savings per maneuver.

These two figures also show a few interesting trends in the choices AAC makes about which resolution maneuvers to use. The first would be that the percentage of resolutions that use vertical maneuvers is relatively

constant In addition, the relative simplicity of these maneuvers helps keep the delay per resolution on the scale of around 7 seconds. However, the number of horizontal maneuvers is greatly increased as the horizontal separation radius increases. The increase in these maneuvers is the result of the decrease in the number of D2 and speed change maneuvers. It is this increase in the percentage of horizontal maneuvers as much as the increased delay per horizontal maneuver that is causing the large increases in the average delay per resolution for the overall results of non-arrival conflicts. We expected that increased separation radii would result in more delay for horizontal maneuvers, if only because the aircraft must fly farther to get clear of a conflict. What we did not expect was that AAC would be forced to use so many more traditional horizontal maneuvers as an additional result of increasing horizontal separation radius. In the end, we can say that it is quite probable that the sensitivity in AAC to required horizontal separation is increased as a result of the use of D2 and speed change maneuvers. Without these maneuvers, there would likely be far more horizontal maneuvers in all cases, and the average delay per resolution would be far less sensitive to this parameter. This result also points out that reducing the number of horizontal resolution maneuvers can have a significant impact on the average delay per resolution. While this is a relatively apparent result, it does emphasize that further improvement upon AAC's average delay per resolution will need to successfully reduce the number of resolution maneuvers in the horizontal plane other than D2's. Whether or not this high emphasis on vertical resolution maneuvers in non-arrival conflicts is desirable from viewpoints other than resolution efficiency is beyond the scope of this paper.

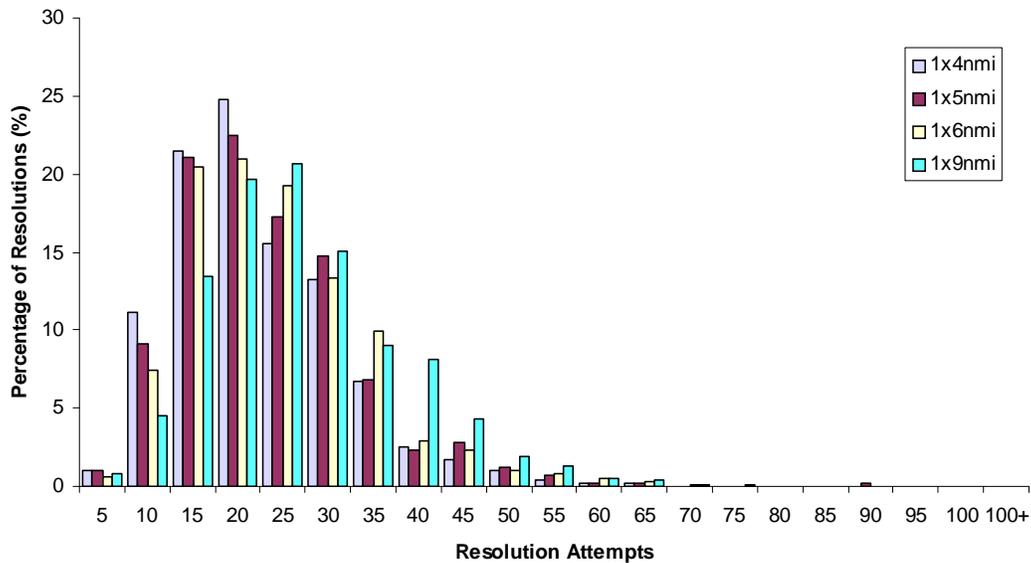


Figure 3. Histogram of resolution attempts required to find a successful resolution for non-arrival conflicts at a 1X traffic demand level

Figure 3 shows a breakdown of the number of resolution attempts required to find a successful resolution for a conflict in the non-arrival environment. Each bin is 5 resolution attempts wide. This chart is a visual confirmation that the shift in the average number of attempts required for a resolution is due more to a shift in the center of the distribution than a shift in the shape of the curve. The resolution attempts curve is certainly flattening out, and would most likely continue to flatten if the horizontal separation radius were further increased. However, as it is unlikely that the required horizontal separation would be increased to over nine nmi in a real world situation, the behavior of AAC beyond that point is of reduced interest. In the case where AAC was beginning to struggle, we would expect the distribution to be much more flattened out, with significantly more cases requiring large numbers of resolution attempts. These curves, then, can be used as a check on how close AAC is to the edge of acceptable performance. As such, we can say that these curves support the assertion that AAC is stable for all of the separation radii tested for cases based on our selected day of traffic.

B. Arrival Conflicts

The arrival conflicts consist of conflicts between two flights that are headed to the same airport. In this area, AAC has some additional functionality to help handle aircraft in this difficult environment. Specifically, it utilizes a first come, first serve arrival scheduler. The scheduler tries to maintain at least one minute of separation between

flights as they pass into the terminal area. This forces a lot of interaction between the conflict detector and the scheduler logic, if for no other reason than to limit the number of cases in which both sets of logic are trying to address the same aircraft at the same time. The results of this from an AAC point of view are much smoother operations when dealing with arrivals. Unfortunately, from an analysis point of view, this makes truly separating the conflict data from the scheduler data difficult if not impossible. Perhaps in later versions of this code there will be ways to more easily separate the cases. As it currently stands, however, the scheduler and arrival conflict logic need to be looked at together, at least in the realm of conflicts and failures.

The way conflicts are defined is important to note, because the scheduler and arrival conflict resolution logic is the same at its core. Thus the general method of addressing a problem is the same. In both cases, the logic will try to satisfy some criteria by using flight plan amendments. In the case of an arrival conflict, the goal is to maneuver so that separation is not lost. The scheduler, on the other hand, is dealing with a single aircraft and seeks to maneuver it in such a way that there is at least a minute between all aircraft headed for the same fix. The scheduler also checks to ensure that all separation standards are maintained through its resolutions. The difference between the arrival conflict and a scheduled aircraft is that a failure of the conflict resolution logic in the arrival environment is certain to result in a loss of separation. For the scheduler, however, a failure simply means that the algorithm could not find a time slot for the aircraft without exceeding some maximum allowable delay. While this type of failure will likely result in a loss of separation, it does not guarantee that there will actually be one. This is all complicated by the interaction between flights that have failed to schedule and the conflict detector logic. At the time of this paper, that aspect of the logic was still being examined.

A last point to make when looking at the arrival data in general is the reminder that the density of arrival traffic was manually lowered to about 1.5X for all of the 2X and 3X cases. As such, though the direction of trends might be accurate for a full 2X or 3X run, the rate of change that results from increasing traffic demand level is most likely underestimated for all of our numbers.

Table 3. Summary of arrival results

Traffic Demand Level	Required Horizontal Separation (nmi)	Resolutions	Unresolved Conflicts	Average Delay per Resolution (min:s)	Average Attempts per Resolution
1x	4	71	0	00:08	24.7
2x	4	147	0	00:22	24.9
3x	4	177	0	00:34	25.3
1x	5	112	0	00:19	27.2
2x	5	236	0	00:28	26.9
3x	5	312	0	00:48	28.7
1x	6	169	0	00:30	29.6
2x	6	356	0	00:42	28.7
3x	6	520	1	00:56	30.1
1x	9	306	0	01:07	35.2
2x	9	891	1	01:22	40.4
3x	9	1559	2	01:59	41.9

The first thing to notice about Table 3 is resolution and unresolved conflict data. As mentioned earlier, the exact resolution count must be taken with a grain of salt because of the complicated interaction between arrival conflicts and aircraft that have been through the scheduler and the reduced number of arrival aircraft in the 2X and 3X demand sets. Still, it is useful to look at the cases that were qualified as arrival conflicts to see how the conflict resolution algorithm performs in this environment. Also, the fact that there were unresolved conflicts in multiple cases should be noted. The fact that all of these unresolved conflicts happen in the most difficult cases implies that AAC might be nearing the edge of its current performance envelope.

The average delay per resolution is where the first few points can be made. In the four, five and six nautical mile horizontal separation cases, there was around a 30 second increase when moving from 1X to 3X for a fixed radii. The nine nautical mile cases showed a 52 second increase in the average delay per resolution moving when comparing 1X to 3X. In addition, every nautical mile increase in the horizontal separation for a given demand level resulted in roughly 10 seconds being added to the average delay per resolution. However, when moving from six to nine nmi the change per nautical mile was closer to 13 seconds for 1X and 2X demand levels, and 20 seconds per nautical mile for 3X. Thus, the efficiency of AAC's conflict resolution maneuvers in the arrival environment is

sensitive to both changes in traffic demand level and required horizontal separation. This sensitivity is expected to be even more pronounced if full 2X and 3X arrival traffic is used. Unfortunately, the near linear trends observed in the data would probably not hold for these cases.

The last point is the average resolution attempts per resolution. Again, this is a metric that lets one get a feel for the workload of the conflict resolution algorithm in terms of the difficulty solving an individual conflict. Cases with higher resolution attempt values are more likely to see high delay resolutions or unresolved conflicts. For these data, except for the nine nautical mile cases, the data showed this parameter was more sensitive to required horizontal separation than traffic demand level. For the nine nautical mile cases, there was some increased sensitivity to demand level. This is not a surprise when we consider the other data for the arrival case, which shows that nearly all of the parameters show increased sensitivity for these cases.

These data point to arrival conflicts being significantly more sensitive to both changes in demand level and required horizontal separation than the non-arrival cases. For the arrivals in general, changing horizontal separation and traffic demand had a noticeable effect on the average delay for a resolution, though only changes in the horizontal separation seemed to affect the average number of resolution attempts required. Additionally, the increased sensitivity shown in the nine nautical mile arrival cases implies that the system might have a limit somewhere above the range of variables we tested. Most likely, increasing the horizontal separation radius beyond nine nmi should produce ever increasing trends until AAC is no longer able to adequately handle the traffic. However, as nine nmi is already a horizontal separation radius that is considerably larger than today's standards, it is unlikely that AAC would be tasked to handle cases like those in the real world. As such, all one can say from these data is that, for the reduced arrival traffic cases which were examined, AAC showed it was quite capable of handling conflicts safely, with the possible exception of the nine nautical mile, 3X case.

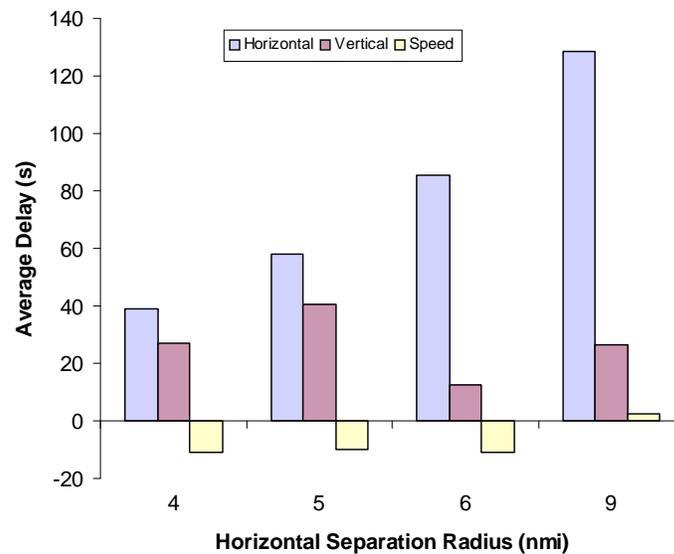


Figure 4. Arrival delay breakdown for all horizontal separation radii at 1X traffic demand level

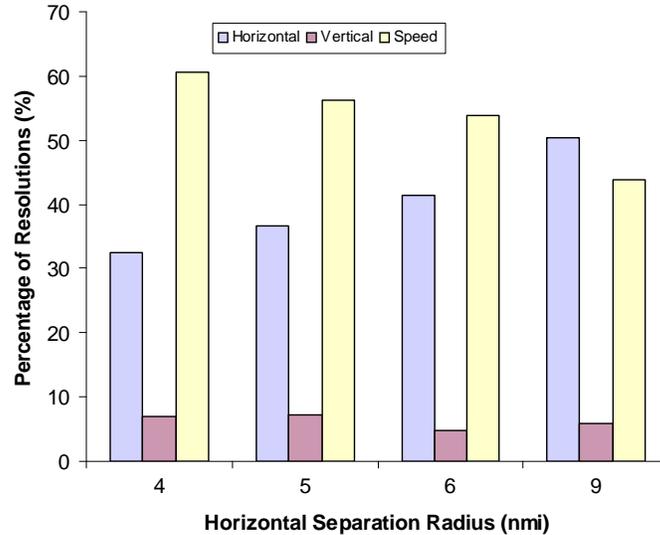


Figure 5. Resolution breakdown by maneuver type for arrival resolutions at 1X traffic demand level

Figures 4 and 5 show the average delay per resolution for the arrival cases and the percentage of resolutions that used each class of maneuver, respectively. The first thing to note is the absence of D2 maneuvers that played such a large role in the non-arrival environment. With aircraft so close to their final fixes in the arrival cases, these D2 maneuvers do not work nearly as well, so they are not used. One of the other things to keep in mind with these charts is that the conflict detection logic is looking ahead 20 minutes to scan for arrival conflicts, which gives AAC more time to find a solution and more options for finding a solution than it would have with the same time window the non-arrival logic uses.

With all this in mind, one of the notable features is the very large increase in the delay of horizontal maneuvers due to increases in horizontal separation radius. Many of these resolutions have large times to first loss, and a resolution algorithm that is trying to minimize delays. Even so, when forced to use a horizontal maneuver in the arrival environment the delay is much more sensitive to increases in horizontal separation than it was in the non-arrival cases. The increase in the arrival case is on the order of 20 seconds per nautical mile, as opposed to the roughly 5 second per nautical mile increase seen in the non-arrival cases. This shift is likely due to the extra complexities involved with moving an aircraft horizontally in the arrival environment while avoiding secondary conflicts from all the other aircraft sharing the airspace. Figure 5 shows that, like the non-arrival cases, increasing the horizontal separation radius drives up the number of horizontal resolutions used.

A second point of note in these figures is the relatively small role that vertical maneuvers play in the arrival cases. With the low number of vertical resolution maneuvers used (Fig. 5), the variation in the average delay per resolution with changes in horizontal separation radius for those maneuvers is easier to understand. With so few vertical maneuvers, a few outliers can significantly affect the average, which is part of what happened here. Even so, the changes in the average delay per resolution for these maneuvers were much larger than anticipated.

Part of the reason there are so few vertical maneuvers in these cases could be because of the number of speed resolutions. It was not surprising that AAC would prefer to use speed resolutions when possible, as they usually have such low delay. However, the fact that AAC found so many successful speed resolutions in the arrival environment was initially unexpected. Some probable explanations include the large amount of time AAC has to deal with conflicts in the arrival environment. With up to 20 minutes to examine a conflict, the time the aircraft has to respond is significantly more than in the non-arrival cases. As such, it is easier to find speed resolutions in these cases, whereas those resolutions might not maneuver an aircraft fast enough in a conflict with only 6 minutes to first loss instead of 16. It should also be remembered that there is a scheduler operating on many of the arriving flights. As such, even though the airspace might be more crowded than in a general, non-arrival case, there should be more order in the arrival cases. Fortunately, this is a situation one could expect in a real world implementation, as traffic flow management tools and arrival schedulers are already in use in some centers. As such, it seems logical to assume that any real world implementation of AAC would be able to take advantage of such tools, as well. It is also notable

that the speed maneuvers have relatively constant delay until the nine nautical mile case, which is the only one where speed maneuvers have positive delay.

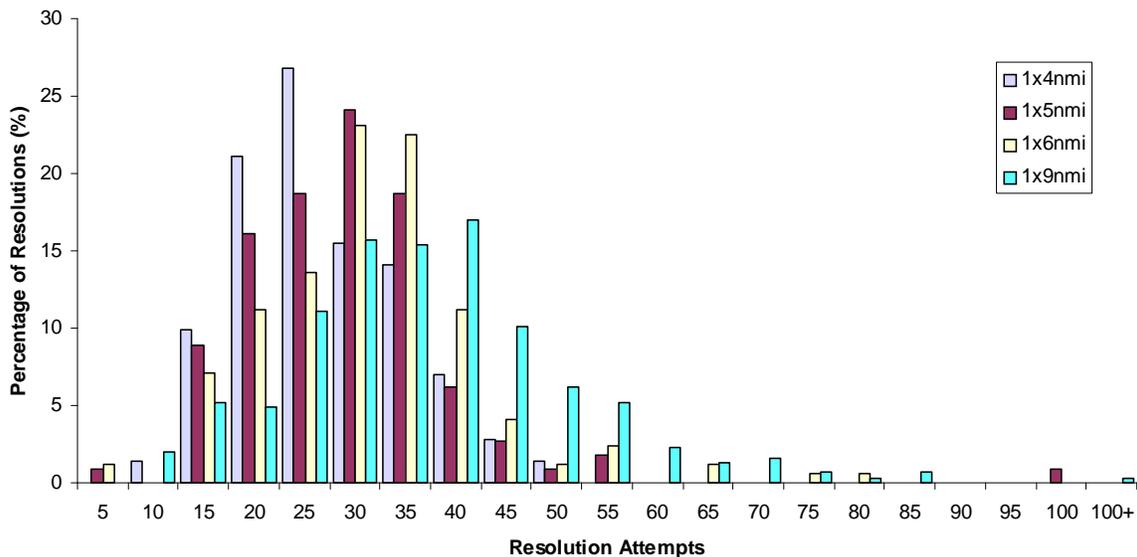


Figure 6. Histogram of resolution attempts required to find a successful resolution for non-arrival conflicts at a 1X traffic demand level

Figure 6 shows what the 1X breakdown of the arrival attempts required to find a resolution looks like for the 1X case. It is immediately evident that the shift in the mean is larger than it was in Fig. 3.

IV. Discussion

The non-arrival cases showed very little sensitivity to either traffic demand levels or required horizontal separation. In the general sense, this implies that AAC is stable, at least for these settings and this day of traffic. This is the kind of behavior one would want from a code being used to help handle a safety critical system with a wide range of possible scenarios. These results also show some of the benefits of reducing required horizontal separation. Though there is not much difference in the average delay per resolution moving from five to four nmi, there is a 20-25% reduction in the number of issued resolutions for all demand levels. That works out to roughly 550 fewer resolutions in the non-arrival environment at 3X. That reduction in the number of resolutions combined with the lower delay per resolution works out to 119 minutes of overall delay savings for moving from five to four nmi.

The arrival case was sensitive to both demand level and required horizontal separation. In these cases there were some unresolved conflicts in the highest demand and separation radius cases. While a strategic resolver is expected to have a tactical, safety-critical backup, it is still desirable to keep the number of unresolved strategic conflicts as low as possible. AAC showed better than a 99.8% success rate in all cases. Of greater impact is the large spread in average delay per resolution, which changes from about 19 seconds for the case most representative of today’s NAS to 1 minute and 59 seconds in the 9 nautical mile, 3X case. While it is unlikely that the real-world would see such a large horizontal separation radius coupled with such a dense traffic level, the data show a definite negative impact to resolution efficiency with increasing demand and/or required horizontal separation. In addition, moving from five to four nmi of horizontal separation appears to have some benefit in the arrival environment, saving over 10 seconds per resolution even at just 1X. Recall, however, that the arrival traffic in the “2X” and “3X” cases were reduced to roughly 1.5X. If a full 2X or 3X scenario were run, it is likely that the differences between all of the horizontal separation radii would increase for the two higher demand levels. That would also increase the benefit of using a smaller radius in arrival cases. Unfortunately, the relatively small number of arrival cases compared to non-arrival cases means that the overall benefit of a reduction in horizontal separation standards is likely to remain small, though it could be significantly more in a center with a greater percentage of arrival traffic.

The average amount of work AAC had to do to find a successful resolution algorithm increased substantially in the worst-case scenarios. This increase in the number of resolution attempts in those scenarios implies that those unresolved conflicts are probably not outliers, and that the number of unresolved conflicts is likely to keep increasing if AAC is stressed with horizontal separation requirements greater than nine nmi in a heavy arrival scenario. This is a concern, as it implies that AAC could have trouble finding resolutions in higher density arrival

scenarios that use demand sets with full arrival traffic as opposed to the 1.5X arrival sets used in this study. In the end, however, even though AAC had to struggle to find resolutions in the most demanding cases and was not able to find particularly efficient conflict resolutions, it was nevertheless able to resolve virtually all of the detected conflicts.

The data raise a number of interesting questions. For example, how can speed maneuvers, which are generally more efficient and more favorable from a flight operations perspective, be used more often in non-arrival cases? If we increased the look-ahead time horizon for the conflict detection algorithm to 20 minutes, as it is for arrival conflicts, would that allow AAC to use more speed resolutions and fewer path stretches? The extra time would certainly come with the cost of increased uncertainty in the trajectory, which would likely require larger separation criteria and increase the number of conflicts detected, but would there be enough reduction in the number of path stretch maneuvers to justify the change?

V. Conclusion

Overall, AAC was able to safely handle conflicts over the entire range of horizontal separation radii and traffic demand levels tested. In non-arrival cases, AAC was also able to consistently find efficient conflict resolutions for a wide range of traffic demand levels and horizontal separation radii. In arrival cases, AAC showed reduced efficiency and increased single conflict workload as both the horizontal separation and traffic demand levels were increased.

In the non-arrival environment, AAC had little sensitivity to either traffic demand levels or horizontal separation radii in regards to the number of unresolved conflicts, the average delay per resolution maneuver, and the number of resolution attempts required by the algorithm to find a successful resolution. Non-arrival delays showed no real trend for demand level increases, though the 3X cases had between 3 and 5 seconds extra delay per resolution. Each increase of one nautical mile of horizontal separation produced around 3 seconds of extra delay and 1 more resolution attempt per resolution in all cases for a given demand level.

In the arrival environment, the average delay per resolution and number of resolution attempts required to successfully resolve a conflict were sensitive to both separation radius and demand levels. The number of unresolved conflicts metric was slightly sensitive to large horizontal separation radii and high demand levels. AAC was still able to safely separate all traffic for every case except the nine nautical mile cases at 2X and 3X and the six nautical mile case at 3X. For those three cases, AAC still resolved over 99.8% of all conflicts. Each step up in demand level increased the average delay per resolution by 10 to 30 seconds, with the larger increases generally appearing in the higher separation radius cases. Each increase of one nautical mile resulted in roughly 10 extra seconds of delay per resolution maneuver except for the 3X case moving from six to nine nmi of separation. In that situation, each extra nautical mile added around 20 seconds of delay. The number of resolution attempts required for a given resolution increased by about 2 per nautical mile added, except in the 2X and 3X cases moving from six to nine nmi which had showed an increase of roughly 4 attempts per resolution.

Overall, these results will support work analyzing the overall costs and benefits of using a certain horizontal separation standard in the NAS. These data should give an approximation of anticipated AAC behavior to researchers who choose to examine something other than the current separation standard. The data should also help those who wish to do trade studies involving horizontal separation to understand the AAC portion of the equation.

Acknowledgments

The authors would like to acknowledge Paul Borchers of NASA, who was instrumental in the initial stages of this experiment.

References

¹Farley, T., Kupfer, M., Erzberger, H., "Automated Conflict Resolution: A Simulation Evaluation Under High Demand Including Merging Arrivals," AIAA-2007-7736, AIAA Aviation Technology Integration and Operations (ATIO) Conference, Belfast, Northern Ireland, Sept 18-20, 2007.

²Kupfer, M., Farley, T., Chu, Y., Erzberger, H., "Automated Conflict Resolution – A Simulation Based Sensitivity Study of Airspace and Demand," 26th International Congress of the Aeronautical Sciences (ICAS), Anchorage, Alaska, Sept 15-19, 2008.

³Consiglio, M., Hoadley, S., Wing, D., Baxley, B., "Safety Performance of Airborne Separation: Preliminary Baseline Testing," AIAA-2007-7739, AIAA Aviation Technology Integration and Operations (ATIO) Conference, Belfast, Northern Ireland, Sept 18-20, 2007.

⁴McNally, D., Thipphavong, D., "Automated Separation Assurance in the Presence of Uncertainty," 26th International Congress of the Aeronautical Sciences (ICAS), Anchorage, Alaska, Sept 15-19, 2008.

⁵Consiglio, M., Hoadley, S., Wing, D., Baxley, B., Allen, D., “Impact of Pilot Delay and Non-Responsiveness on the Safety Performance of Airborne Separation,” AIAA-2008-8882, 26th International Congress of the Aeronautical Sciences (ICAS), Anchorage, Alaska, Sept 15-19, 2008.

⁶Thippavong, D., “Analysis of a Multi-Trajectory Conflict Detection Algorithm for Climbing Flights,” AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Hilton Head, South Carolina, Sept 21-23, 2009. (to be published)

⁷Erzberger, H., “Automated Conflict Resolution for Air Traffic Control,” 25th International Congress of the Aeronautical Sciences, 2006.

⁸Meyn, L., Windhorst, R., Roth, K., Drei, D. V., Kubat, G., Manikonda, V., Roney, S., Hunter, G., and Couluris, G., “Build 4 of the Airspace Concepts Evaluation System,” AIAA Modeling and Simulation Technologies Conference and Exhibit , 2006.

⁹Sensis Corp. AvDemand User Guide, Campbell, California, 2007.