[EN-010] Facilitating Free Flight Conflict Resolution using Nautical Minute Discretisation

(EIWAC 2010)

+P. Simon*, C. Bil*

* School of Aerospace, Mechanical and Manufacturing Engineering
Royal Melbourne Institute of Technology
Melbourne, Australia
[paul.simon | c.bil]@rmit.edu.au

Abstract: Nautical Minute Discretisation (NMD) refers to the use of airspace, divided into minutes of longitude and latitude, to map out aircraft trajectories and conflicts. NMD methodology provides a conceptually different approach to handling current conflict avoidance, detection, resolution, and complexity computations, despite being based on the same concepts. While NMD methods were developed for simulation of Free Flight, simulation of highly dense and complex scenarios experienced today, such as Advanced Flexible Use of Airspace and Dynamic Sectorisation efforts could also use these methods.

Heuristic and MatLab data transfer algorithms that support NMD inside the current framework for air traffic control have been developed, and will enable NMD methodology to provide holistic assessments of potential conflicts and overall air traffic complexity. Computation time trials using NMD have been performed, and indicate efficient and expandable functionality in some areas. Where inefficient code was found, efficient variations have been developed, and their inclusion into the NMD framework is underway. NMD methodology appears to be a viable means of speeding up calculation of conflict detection, resolution, and complexity.

Keywords: Capacity, Complexity, Conflict Detection, Conflict Avoidance, Conflict Resolution, Conflict Characterization.

1. INTRODUCTION

Ensuring safety, and optimizing efficiency, within the framework of Free Flight is a difficult task wherever air traffic complexity is high. Even on the assumption that Free Flight is simply the allowance for aircraft to fly their preferred route, the significantly large range of possible interactions between aircraft forces the creation of pre-established routes by Air Traffic Management (ATM), and the use of direct intervention by Air Traffic Control (ATC). The true definition of Free Flight, i.e. pilots can select their preferred route, for example based on minimum distance or favorable winds, etc, is potentially even more complicated, as it implies that the entity responsible for safety (be it pilot, airline or ATC) can instantaneously allocate sufficient airspace to ensure safe separation for all interactions that aircraft has till it lands.

These complications imply technological intervention: it would be unsafe to rely on only isolated sectors and limited human capability to ensure correct separation standards in such a complex environment. However, while the technological capability to facilitate this may exist, Air Navigation Service Provider (ANSP) approved methodologies, that could use such technology, do not. Thus it becomes important for the underlying methodologies that can support these technologies to be changed to provide ANSP, and relevant stakeholders, sufficient information and capability to facilitate Free Flight.

A potential methodology is Nautical Minute Discretisation (NMD) using finite regions of airspace defined by the minutes of longitude and latitude as a reference. NMD can turn the complexities of high capacity free flight into a space allocation problem based on the properties of the finite regions that define it as well as the aircraft going through them. Furthermore, as the methodology is discrete in nature, it benefits from efficiencies gained from improvements in numerical and parallel computation.

This paper describes the efforts and results from the application of an NMD based system that ensures safety and timely separation in any high capacity scenario, not just Free Flight. Given these improvements, NMD based systems may become useful for current applications; potentially speeding up dynamic sectorisation or flexible use of airspace capacity.

2. BASIC CONCEPT

ATC and ATM simulators use one or two airspace calculation methodologies to primarily understand and manipulate air traffic; path mapping [1] & conflict detection [2], or aircraft clustering [3] & sectorisation [4]. The former follows the dynamics and limitations of aircraft relative to other airspace users.. The latter follows the limits of airspace itself in view of other situational limitations like dynamic airspace capacity and controller workload. It is preferable for complete situational awareness, that both methods are employed in any ATM system program [5]. Furthermore, the limitations that these programs encounter come from the need for increased computing power to handle the systems' level of variable fidelity, and the need for increased computational processing speed to handle the large number of complex iterations to safely manage multi aircraft pathway interactions on an entire ATM system scale.

With increased maturity of finite element methods, attention is now on discretizing three dimensional volumes, and using the properties gained from doing so to assist in speeding up computational calculations [6]. Physical variable discretisation is common in numerical and simulation studies, even in ATM improvement studies. Sectors are an example of physical area discretisation as they divide an area to evenly distribute ATC workload. Further sector sub-division into common shapes has been performed to allow iterative optimization of ATC workload distribution [7]. Area discretisation therefore, facilitates the two means of understanding and manipulating air traffic.

However the base theory that supports these two means, i.e. intersections of great circle arcs, requires assessment of the flight path of each aircraft with reference to each other aircraft within the same airspace to define regions of intersection. Furthermore, these intersections need to be checked for overlap with other intersections to determine if an intersection that involves more than two aircraft will actually occur.

This process is highly iterative and, unless assumptions on intersection interaction are made early, leads to the computational limitations as experienced by ATC [8]. However, determination of flight path intersection is not the only means of determining conflict; if two aircraft enter a sufficiently small enough area within a sufficiently small time frame then it could be said that the two are in conflict.

Aircraft, due to their wake turbulence profile, have a volume of effect defined by their speed and the wake's rate of dispersion and dissipation. Assuming no wind, and general wake drop rates, this volume can be safely contained within an area thousand feet high and one nautical mile wide, with a length defined by aircraft speed. If you exclude the presence of the aircraft, a nautical mile becomes the base unit for lateral separation. In essence, if two cruising aircraft are within 1000 vertical feet (reduced vertical separation minima) and five minutes (minimum time separation standard for cruising aircraft with potential conflict [9]) of each other when they enter the same square nautical minute, then they are in conflict. Determination of conflict in Terminal Maneuvering Airspace can be done the same way, but will probably require square nautical seconds.

3. CONCEPTUAL EXPANSION

The basic NMD core concept is simple, but it does have significant ramifications on how aspects of traditional conflict recognition and resolution programs should be written. For simplicity, the necessary modifications to traditional conflict recognition concepts can be categorized in terms of trajectory mapping, conflict assessment, and conflict resolution.

3.1 Trajectory Mapping

Currently, ANSP can use a variety of methods to map a particular trajectory. Generally speaking however, planned routes between destinations are largely static (e.g. [10]). Simulators that assess situational complexities will of go a little further, and map regions so as to define the intersections, and intersections of intersections, mentioned earlier. Once a route is known, pilot, airline or ANSP simulator programs can define a desired profile using their choice of aerodynamic, range or endurance equations, and applying heuristic or experience based methods to optimize the flight path where possible. Where the aircraft experiences minimal interruption to its desired plan, these efforts lead to an optimal flight, given ANSP requirements.

Where wind is known to have a significantly variable effect on a flight, an optimal route can be determined from a selection of potential routes with calculated flight times derived from the flight's desired profile and known meteorological data. Generally, wind is not largely variable, and simplistic forms of the above process will define, with sufficient accuracy, predicted flight times, and therefore flight trajectory.

The relevance of this to NMD trajectory mapping is two fold. Firstly, NMD needs to define routes in terms of cells that are entered by an aircraft; an NMD route allocation algorithm is therefore needed. Secondly, given the interaction between wind, flight profile characteristics, and the NMD cell's ability to determine conflict, a means of transferring data between aircraft trajectory and the NMD cells must also be defined.

3.1.1 Allocation of NMD Cells to a specified trajectory

To allocated NMD cells to a particular trajectory, a method to determine which NMD aircraft actually enters the cell needs to be found. If all cells in a sufficiently sized NMD field, continental sized NMD fields usually comprise greater than six million cells, have to check their cross track distance from each aircraft's path to see if it is less than the maximum cross track deviation allowed, then the resulting number of iterations, even assuming parallel processing capability, may be significant. To avoid this, a heuristic algorithm was developed that searches for potentially entered NMD cells based on the longitudes that it passes whilst in the NMD field. A summary of the method follows the steps below with Figure 1, Figure 2, Figure 3, and Figure 4 giving a visual description.

1. Define/Nominate the NMD field. This is largely superfluous, except where aircraft are likely to exit the field, in which case NMD field limits are used as default values where longitude and value calculations exceed these limits.

2. Place all discrete longitude values between the start and end point (including them, and some outside of them to form a buffer) and place them in a vector array.

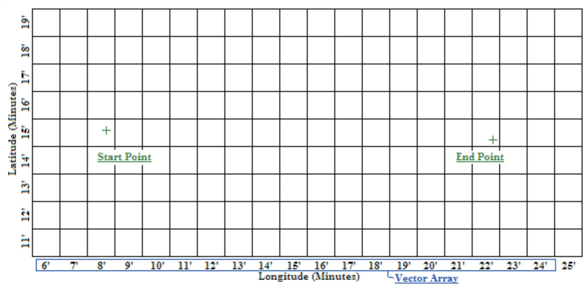

Figure 1 Step 2 – Collecting Longitudes

3. For each element of this array, determine the corresponding latitude (unrounded) and aircraft bearing at that latitude, using the start point and its bearing to the end point.

4. For each bearing, determine the along track distance of a polar path intersecting a path of that bearing assuming a known allowable cross track deviation (usually 5 NM minimum [9] plus buffer). This distance can be calculated using a flat surface approximation as it is defined along a meridian and does not suffer the spherical warp to coordinate position that distances in other directions would incur.

5. Centre this distance on the corresponding latitude for each bearing to determine the maximum and minimum latitude experienced for each discrete longitude.

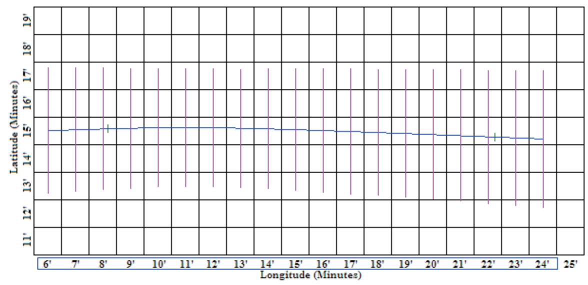

Figure 2 Step 3-5 – Determine Relevant Discrete Areas by Latitude

6. For each discrete longitude, place the cell index of all cells between and on the maximum and minimum latitude for that longitude into a cell list.

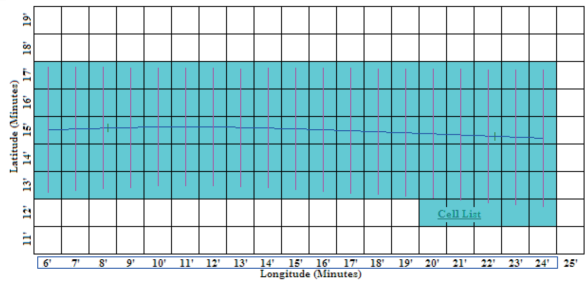

Figure 3 Step 6 – Collecting Relevant Discrete Areas

7. Calculate the cross track distance and along track distance of all NMD cells in the list, and remove any that are sufficiently far away for them not to be involved in the aircraft's path.

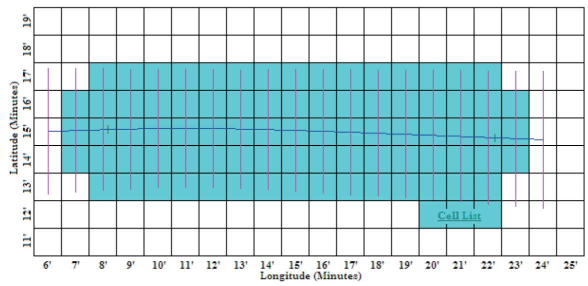

Figure 4 Step 7-9 – Correcting for start and end points.

While not shown here, all buffers have analytical equations to determine their size to ensure that all relevant cells are included in the vector array.

3.1.2 Data Transfer between NMD & Aircraft Profile

In order to facilitate trajectory prediction, an aircraft profile must be applied over known weather conditions, which are usually defined within a weather grid which itself is divided by regions on the ground surface, and a particular flight level. Transition of weather data to NMD only requires appropriate discretisation of an already discretized data source. Traditionally, such weather grids define regions in terms of five or ten nautical degree squares; this common factor with NMD can therefore be used.

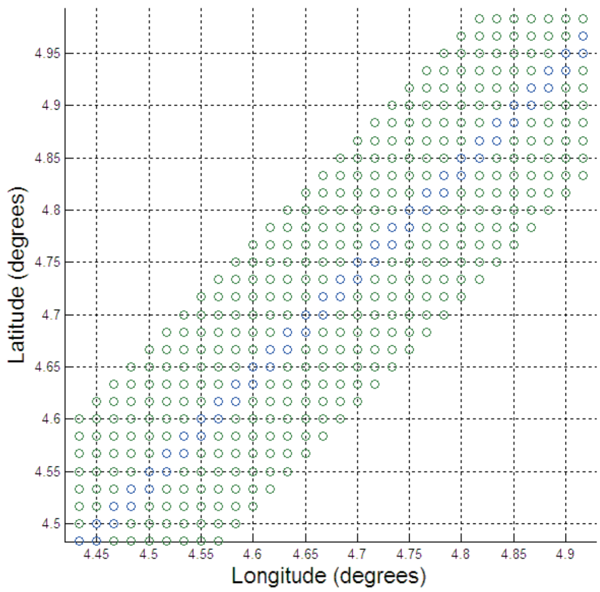

Figure 5 Aircraft Field (green) and Actually Entered (blue only) NMD

To transfer data from NMD to aircraft trajectory requires a distinction between the NMD cells that are within maximum cross track deviation of a route, and the NMD cells that the aircraft actually entered. The rationale being that an aircraft will only experience the wind of the cells it actually enters. To perform this separation each cell's absolute cross track deviation from the route is checked to see if it is less than the maximum half diagonal of a cell at that latitude. The cells for which this is true are then given a tag indicating they represent the actual path of the aircraft. Figure 5 gives a visual example of this separation.

Once actual path cells have been tagged, their wind magnitude and direction are stored with the along track distance (as defined in the allocation of NMD cells) of the cell they come from. This data is then interpolated against all cells within the trajectory's route, and used to determine the time that an aircraft will pass by that particular cell.

3.2 Conflict Assessment

Also known as conflict detection, conflict avoidance, etc, this stage is merely the determination of whether or not any two aircraft are likely to interfere with each other.

The previously mentioned method regarding the intersection of two great circle arcs is likely the most used form of conflict assessment for the purposes of ANSP; in localized settings, it is simple, quick and accurate, and all known separation methods and minima [9] can be applied to determine where two aircraft will intersect and therefore use as a reference to apply separation from. When written as a program, it only requires that other objects that can be interacted with are also defined with the great circle theory. Again, this is correct in a localized setting, but when airspace density or complexity increases, an alternative may be needed.

Another method frequently seen in conflict assessment simulations is what is commonly referred to as the "Puck" envelope; the aircraft's field of influence would be a "puck" shaped envelope, i.e. 1000ft above and below, and 5 NMI laterally around, centered on the aircraft as it follows its path. Conflict detection occurs when any aircraft enters the puck envelope of another aircraft. The method works well wherever an aircraft's path is discretized in terms of time; the aircraft's general position in global coordinates, with respect to time, becomes known and the envelope becomes the check system for conflict detection. The only perceivable weakness in how it is used currently is that it does not readily cater for wake field definition behind an aircraft. However, if altered to allow a time aspect to the puck (i.e. 1000ft above, below, 5NMI around and 5 minutes ahead, or behind) then there should be no reason for it not to be used in such a capacity.

NMD assessment of conflict is fairly similar to the puck envelope; however it has two noticeable differences. The first is that it only considers the altitude and time when an aircraft passes a cell; if it is near enough to be included in the cell, then there is no need to consider the lateral differences between any aircraft in the cell. The second difference is that conflict assessment in puck is usually done iteratively to simulate the horizon perspective of the aircraft, whereas NMD conflict assessment can be performed without cell order and is therefore more suitable to complexity assessment and not just conflict assessment, both of which can be done in parallel.

Figure 6, Figure 7, Figure 8, show the perspectives of consideration of two conflicting aircraft. Figure 9 shows how NMD interprets these data and suggests an alternative altitude that is correct for the cell that it is suggested for. NMD does not consider, for the purposes of conflict assessment of a particular cell, any data outside of that cell.

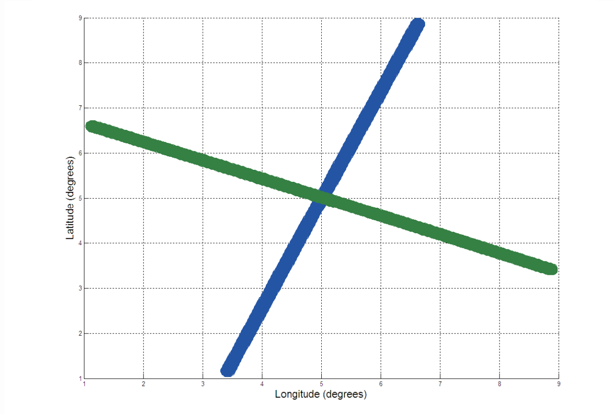

Figure 6 Scenario A: Two intersecting flight routes

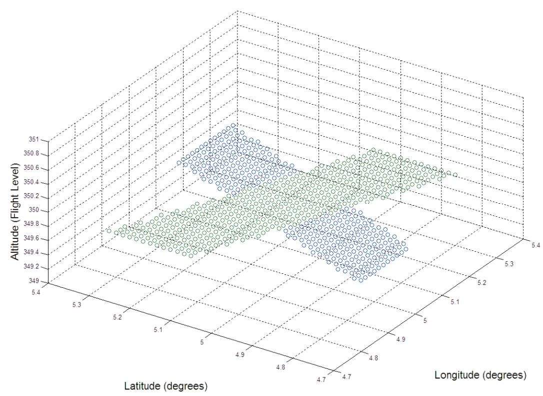

Figure 7 Scenario A: Closer Inspection of Intersecting Fields

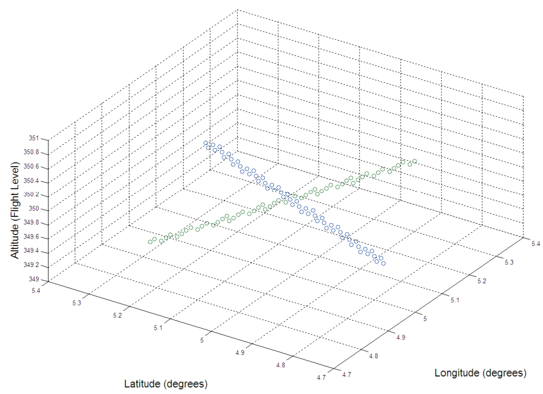

Figure 8 Scenario A: Actually Entered NMD within Figure 7

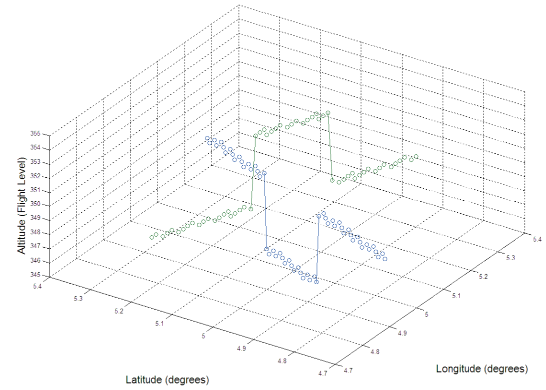

Figure 9 Scenario A: Suggested Altitudes for cells from Figure 8, given the influence of the field cells from Figure 7 (lines added for clarity)

Figure 10 and Figure 11, characterize another scenario, the key result to be indicated is within Figure 12; note the distribution of aircraft counts in the intersection. The ability to do this suggests a high resolution in defining complexity assessment for a particular airspace.

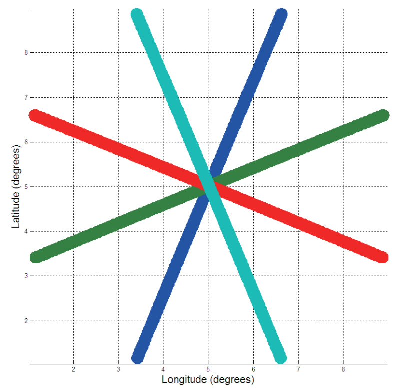

Figure 10 Scenario B: Four intersecting flight routes

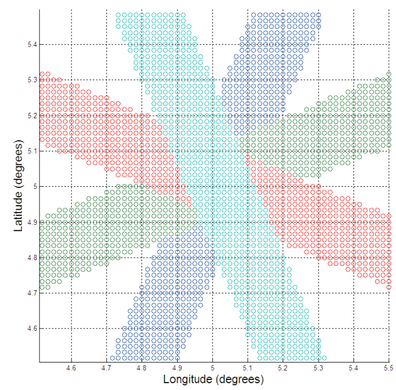

Figure 11 Scenario B: Closer Inspection of Intersecting Fields

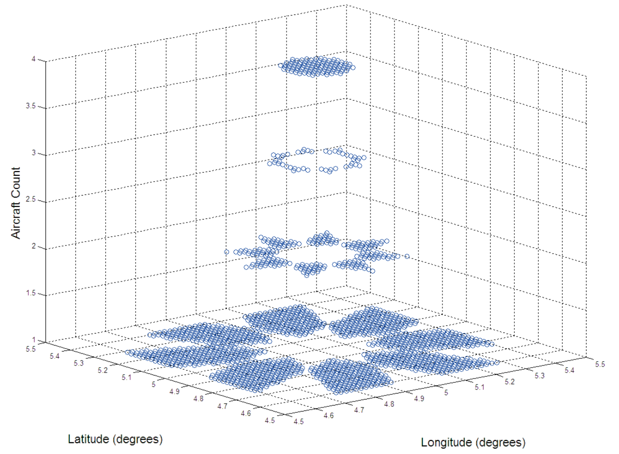

Figure 12 Scenario B: Aircraft Counts for each NMD cell in Intersection

3.3 Conflict Resolution

The resolution of conflict is, at the simplest level, whatever action is needed to ensure that an apparent conflict does not occur. ANSP achieve this by developing procedural structures to support and limit tactical separation of aircraft. Free Flight wants to achieve this by setting up similar processes but as implemented by aircrew. The exact method of how to achieve it, whilst maintaining an optimal trajectory causes considerable debate due to the many airspace complexities that can occur. As these methods are still being explored, and a means of optimizing trajectory whilst allocating space is still being sought, NMD has no set way of resolving a conflict.

However, the features present in NMD allow for certain possibilities. In addition to suggesting an alternative altitude, NMD methods can also suggest alternative speeds and lateral deviations to clear a conflict. The provision of alternative altitudes and speeds stems from an assessment of the cell's aircrafts' desired, minimum and maximum speeds and altitudes, and available times within the cell. Lateral deviation is determined by the relative bearing of conflicting aircraft within an NMD cell. In essence, NMD has the ability to create suggestions for maneuvers in any dimension. There are three implications to this.

The first is that assessment of alternatives currently occurs at the conflict determination level, i.e. that existence of conflict automatically calls up solutions. This suggests synergy with an automatic re-optimization process in the event of a conflict that just appeared. Provided airspace complexity is low, and that aircraft therein usually fly an altered path, the automatic determination of a maneuver to an optimized trajectory would be appealing.

The second is that the selection of alternatives can be done in a holistic manner, using potential guides or desirables, such as reduced fuel emissions, to define manipulation of the entire system. This is powered by the NMD ability to characterize conflict for the whole airspace, and should lend it self readily to conceptual flight rules that lack the ability to handle high density airspace complexity.

The third, as an aircraft's minimum and maximum speeds and altitudes are interpreted as unbreakable boundary conditions; it is possible for them to be set as values other than the actual physical limitations of the aircraft, provided they are still within the physical limitations of the aircraft. This can work with either of the two implications; as ANSP and airline interpretation of what an aircraft's minimum and maximum speeds and altitudes should be could cause niche development for airlines. This development could, if designed properly, reflect a holistic guide for air traffic, or become a standard to which an automatic re-optimization could be applied.

4. COMPUTATION TIME TRIALS

4.1 Rationale

For the purpose of conflict detection the use of NMD is practically a brute force method. It can be done simpler by using the great circle theory with some assumptions on the proximity of involvement of intersections. However, as mentioned previously, accuracy of information required for resolution of highly complex conflict scenarios may be insufficient. As a brute force method, the pertinent interest would therefore be in terms of processing time. The costs in terms of additional processing time, or the equivalent hardware required to support it, has to be leveraged against the benefits it gives in terms of additional situational assessment and minimized data transfer.

In order to facilitate such an assessment, computation time trials were run on two airspace scenarios. However as current performance characteristics of software processed conflict detection and resolution are only known to organizations that use such software, trying to define an industry standard for the sake of comparison is difficult. Furthermore, differences in airspace complexity, hardware, software, and code interpretation of great circle theory cause such software to have variable performance characteristics and would make comparisons difficult. Instead, the two airspace scenarios simply recorded times of computation, and numbers of expected computation sessions (ECS) for assessment.

4.2 Potential Issues

Two issues have to be mentioned regarding these tests. The first regards the validity of using MatLab as representative of actual computation times. The second regards how the computation times are interpreted.

4.2.1 MatLab as indicative of true computation speeds.

The experiment assumes that MatLab is representative of actual computation speeds. MatLab is indeed representative of processing capability currently available, and particularly in the area of numerical computation and use of parallel processors. Furthermore, it does use IEEE standards wherever applicable. Its use in time trials made sense and would be indicative of likely computation times in a crowded data environment. Just to make sure that any unintentional code was utilized in the program, the NMD program was written entirely without toolboxes, and should therefore be transferrable to other software platforms.

There is the possibility that MatLab can cause unintended software acceleration, but that is indicative of current numerical computation methodology, and therefore a positive inclusion for testing on MatLab. In addition, the simulation times excluded time for graphics processing which is a negative consideration for potential real time use, however as the NMD data layers inside use a linear index indicative of the actual airspace field, creating an image representing the data therein is not as time consuming as using plot functions available in MatLab.

Overall however, MatLab does adhere to the limitations of the computer system, so whatever computation times are achieved, should allow appropriate indication of computation times on other systems.

4.2.2 Logical Assessment of Computation Times

As no industrial comparison can be made with the computation times recorded, an internal measurement system was used to see if the NMD program was being processed faster or slower than expected. Computation times for MatLab functions are usually proportional to an aspect of the data the function handles. If what aspect therein is known, it can be used to define what can be called an expected computation session (ECS). As time should be directly proportional to the number of expected computation sessions, one can determine the state of the computation process from the way computation time per ECS changes as number of computations increases.

If time per ECS is constant, it implies that significant variation will only occur at higher numbers of computations; further testing would still be required to ensure that Time per ECS is not increasing, however as an indicator of performance, time predictions, using it as a modifier on the predicted number of ECS, should yield relatively accurate results.

If time per ECS is increasing it implies that unintended data processing is occurring and slowing the program down. It suggests that inefficient code is still present in the program and needs to be removed. Time modifiers calculated in this scenario should not be used for prediction of computation times.

If time per ECS is decreasing, it implies that data acceleration is occurring and speeding the program up. While time modifiers gained in this scenario can successfully predict computation times, care should be taken to ensure that the values for ECS are correct..

4.3 Results and Discussion

The results in Table 1 are the Time Modifiers, and Time per ECS trends for computation times achieved using MatLab R2009a (32 bit) on an Intel® Core™ i7 3.60 Ghz computer running Windows Vista (Note: Parallel Distribution requires a MatLab Toolbox, and was avoided in these trials. Times below are indicative of the program run in sequence as opposed to parallel).

Table 1 Margins and Column Width

Function	Time Modifier (sec) (@ Max #ECS)	Time Per ECS Variation
NMD Cell Allocation	2.28E-06	Constant
Profile and Trajectory Determination	1.71E-04	Increasing
Profile to NMD Data Distribution	6.49E-04	Increasing
Conflict Assessment	3.34E-06	Decreasing

As NMD Cell Allocation and Conflict Assessment experienced either constant or decreasing time per ECS values, their time modifiers can be used to predict computation times under other scenarios. For example:

NMD Cell Allocation's ECS value was determined by the number of times a cell had been included as part of an aircraft's path. Therefore allocation of cells for a 600nmi journey with a 10nmi allowable cross track deviation would take 13.7ms to calculate.

Conflict Assessment's ECS value was determined by the number of aircraft in all cells that were checked for conflict. Conflict assessment for a continent with 6 million cells, averaging 7000 aircraft per cell, would take 38.97 hours to calculate, memory conditions permitting.

These times are for the computer that these values were tested on; however the trials disallowed parallel computation, which is allowed by the methods, just not by MatLab without Toolbox input, and could therefore be significantly faster using parallel super computers.

Unfortunately, as the other two functions experienced increasing time per ECS values, they could not be used to predict other scenarios in their current form. Additional data processing was found in the profiling function with a save function that stores already made NMD profiles within a storage format (cell array) for later use. The cause in the distribution function was the growing data size of the NMD storage layers. In both cases a sufficiently sized predefined matrix would remove the increasing Time per ECS values; however the complexity for such would be significant and would require further assessment of likely data usage.

5. CONCLUSION

NMD methodology, despite being based on the same concepts used in current conflict avoidance, detection, resolution, and complexity computations, provides a conceptually different approach to handling these roles. While NMD methods were developed for simulation of Free Flight, simulation of highly dense and complex scenarios experienced today, can use these methods.

Search and data transfer algorithms that support NMD inside the current framework for air traffic manipulation have been developed, and will the enable NMD methodology to provide holistic assessments of an potential conflicts and overall complexity in a given airspace.

Computation time trials using NMD were performed, and indicate efficient and expandable functionality in some areas. Where inefficient code was found, potentially efficient variations were developed and their inclusion into the NMD framework is underway. NMD methodology appears to be a viable means of speeding up the calculation of conflict detection, resolution, and complexity.

6. REFERENCES

[1] Xue M, and Atkins EM. “Three-Dimensional Segmented Trajectory Optimization for Runway-Independent Aircraft.” 43rd AIAA Aerospace Sciences Meeting and Exhibit, USA, AIAA-2005-909, 2005.

[2] Chaugule V, and Pant RS. “Investigation of Strategies for Optimal Collaborative Airborne Separation in Free Flight,” 7th AIAA Aviation, Technology, Integration, and Operations Conference (ATIO) Proceedings, Northern Ireland, AIAA-2007-7730, 2007.

[3] Bilimoria KD and Jastrzebski M. Aircraft “Clustering Based on Airspace Complexity”, 7th AIAA Aviation, Technology, Integration, and Operations Conference (ATIO) Proceedings, Northern Ireland, AIAA-2007-7728, 2007

[4] Tandale MD, and Menon PK. “Genetic Algorithm Based Ground Delay Program Computations for Sector Density Control,” AIAA Guidance, Navigation, and Control Conference Proceedings, South Carolina, AIAA-2007-6358, 2007.

[5] Alam S, Abbass HA, and Barlow M. “ATOMS: Air Traffic Operations and Management Simulator,” IEEE Transactions on Intelligent Transportation Systems, June, 2008.

[6] Massink M, and DeFrancesco N. “Modelling Free Flight with Collision Avoidance.” Proceedings for the Seventh IEEE International Conference on Engineering of Complex Computer Systems, 2001.

[7] Ehrmanntraut R, and McMillan S. "Airspace Design Process for Dynamic Sectorisation" IEEE Transactions on 26th Digital Avionics Systems Conference, October, 2007

[8] Lope, J, Vilaplana M, Bayraktutar I, Klooster J, Asensio JM, McDonald G, Kappertz P. “Towards an open test bed for the study of trajectory synchronization in the future ATM system: The ASIS initiative.” Integrated Communications, Navigation and Surveillance Conference, ICNS '09, 2009

[9] International Civil Aviation Organization. “Procedures for Air Navigation Services in Air Traffic Management,” 15th Edition, International Civil Aviation Organization, 2007.

[10] Federal Aviation Administration, NextGen Implementation Plan, U.S. Department of Transportation, Federal Aviation Administration, Washington, 2008.

7. COPYRIGHT