Blue Ridge Research and Consulting, LLC

Technical Report

MRNMap Noise Model Improvements

August 26, 2013

Prepared by:
Bruce Ikelheimer, PhD
Micah Downing, PhD

Contract No. W9132T-11-C-0031
BRRC Job No. 110085F

Blue Ridge Research and Consulting
15 W. Walnut St., Suite C
Asheville, NC 28801
(p) 828-252-2209
(f) 831-603-8321
www.BlueRidgeResearch.com

Table of Contents

1. Introduction

The goal of this project was to update the US Air Force's Military Operating Area and Route NoiseMap model (MRNMap), which is the noise model for special use airspaces and routes. This effort updated the computational aspects of MRNMap to improve many of the original simplifying assumptions. Advances in computing power have made such improvements possible, and MRNMap is now equipped with a more robust calculation engine, which has the capacity to produce more accurate results.

Previous work by Fred Wasmer (of Wasmer Consulting) produced the BaseOps Graphical User Interface (GUI) that accesses MRNMap.

Updates to MRNMap are listed below, with details provided in subsequent sections.

- Reconfigured Noise Database: Converted the noise database to conform with the Flight01.dat format.
- Military Operating Areas (MOA) Calculations Improved: Raster analysis and taper calculations, among others.
- Advanced 'Number of Event' Calculations: Implemented a statistical approach.
- BaseOps GUI: MRNMap is operated through BaseOps, although modifications to BaseOps are still necessary to complete this process for seamless operations for a user.
- Standalone Executable File: The Omega10R.exe now runs as a separate executable, and uses updated noise data as input.

2. Reconfigured Noise Database

Two files are required by the new version of MRNMap; the flight reference noise database 'flight02.dat' and the new conversion file 'MRNMap Old-to-New Aircraft Code Table.txt.' The flight02.dat file is an updated version of the original 'NOISE' file, which follows the same format as the NoiseMap 'flight01.dat' file. The MRNMap Old-to-New Aircraft Code Table.txt file provides conversion between the original 'NOISE' file and the flight02.dat file.

The NoiseMap reference noise database file 'Flight01.dat' contains a database of flyover sound power levels for multiple aircraft with a variety of power settings and airspeeds. The Flight02.dat file has been updated to include high speed training route data sets, and two additional aircraft (the C-130J and T-38C). Additionally, the AV-8B aircraft with engine F402-RR-405 was deleted because the data set was estimated and this aircraft and engine combination is no longer in service (the AV-8B with engine F402-RR-408 remains in the database). A list of all modifications made to 'Flight02.dat' is provided in Table 1.

The conversion file named 'MRNMap Old-to-New Aircraft Code Table.txt' comes standard with the Noisemap installation and is located in the 'C:\Noisemap\MRNMap' directory. While attempting to convert sample input files in the old format (V2.2) to the new format (V3.0), it was discovered that several aircraft codes were missing from the conversion file. All of the aircraft codes associated with MRNMap V2.2 are contained in the 'AIRCRAFT.dat' file. Forty-one of these codes, representing mostly training route operating conditions, were missing from the conversion file and have been added (Table 2).

Table 1. List of modifications made to 'Flight01.dat'

No.	Modification Description	Aircraft Name	Engine Name	Entry	Date	Power Setting
1	Added Training Route Entry	A-10A	TF34-GE-100	FM00901211F	10 FEB 89	TRAINING ROUTE
2	Delete Old Aircraft	AV-8B	F402-RR-405	FM00702031V	28 FEB 83	TAKEOFF POWER
3	Delete Old Aircraft	AV-8B	F402-RR-405	FM00702051V	28 FEB 83	APPROACH POWER
4	Delete Old Aircraft	AV-8B	F402-RR-405	FM00702131V	28 FEB 83	TRAFFIC PATTERN
5	Delete Old Aircraft	AV-8B	F402-RR-405	FM00702171V	10 NOV 83	FLIGHT IDLE
6	Added Training Route Entry	B-1	F101-GE-100	FM01201211F	23 MAR 95	TRAINING ROUTE
7	Added Training Route Entry	B-52G	J57-P-43WA	FM01402211F	10 FEB 89	TRAINING ROUTE
8	Added Training Route Entry	B-52H	TF33-P-3	FM01403211F	10 FEB 89	TRAINING ROUTE
9	Add New Aircraft	C-130J	TURBOPROP	FM02904031V	06 MAR 02	TAKEOFF POWER
10	Add New Aircraft	C-130J	TURBOPROP	FM02904061V	06 MAR 02	INTERMEDIATE POWER
11	Add New Aircraft	C-130J	TURBOPROP	FM02904301F	06 MAR 02	HIGH CRU TRAINING RT
12	Added Training Route Entry	C-135A	J57-P-59W	FM03101211F	10 FEB 89	TRAINING ROUTE
13	Added Training Route Entry	C-141A	TF33-P-7	FM03401211F	10 FEB 89	TRAINING ROUTE
14	Correct Interpolation Code from 'P' to 'V'	C-141A	TF33-P-7	FM03401041V	27 DEC 79	CRUISE POWER
15	Added Training Route Entry	C-18A	TF33-PW-102A	FM02101211F	10 FEB 89	TRAINING ROUTE
16	Correct Interpolation Code from 'P' to 'V'	C-5A	TF39-GE-1A	FM01601061V	08 JAN 90	INTERMEDIATE POWER
17	Added Training Route Entry	F-111F	TF30-P-100	FM05203281F	09 SEP 94	LOW CRU TRAINING RT
18	Added Training Route Entry	F-111F	TF30-P-100	FM05203291F	09 SEP 94	MID SPD TRAINING RT
19	Added Training Route Entry	F-111F	TF30-P-100	FM05203301F	09 SEP 94	HIGH CRU TRAINING RT

20	Added Training Route Entry	F-111F	TF30-P-100	FM05203311F	09 SEP 94	HIGH SPD TRAINING RT
21	Added Training Route Entry	F-111F	TF30-P-100	FM05203321F	09 SEP 94	LOW SPD TRAINING RT
22	Added Training Route Entry	F-14A	TF30-P-412A/412	FM04201211F	21 NOV 95	TRAINING ROUTE
23	Added Training Route Entry	F-14A	TF30-P-412A/412	FM04201321F	21 NOV 95	LOW SPD TRAINING RT
24	Added Training Route Entry	F-14B	F110-GE-400	FM04202211F	22 NOV 95	TRAINING ROUTE
25	Added Training Route Entry	F-14B	F110-GE-400	FM04202321F	22 NOV 95	LOW SPD TRAINING RT
26	Added Training Route Entry	F-15A	F100-PW-100	FM04301211F	31 OCT 91	TRAINING ROUTE
27	Added Training Route Entry	F-15A	F100-PW-100	FM04301291F	31 OCT 91	MID SPD TRAINING RT
28	Added Training Route Entry	F-15A	F100-PW-100	FM04301311F	31 OCT 91	HIGH SPD TRAINING RT
29	Added Training Route Entry	F-15A	F100-PW-100	FM04301321F	31 OCT 91	LOW SPD TRAINING RT
30	Added Training Route Entry	F-16A	F100-PW-100	FM04401211F	10 FEB 89	TRAINING ROUTE
31	Added Training Route Entry	F-16A	F100-PW-100	FM04401291F	31 OCT 91	MID SPD TRAINING RT
32	Added Training Route Entry	F-16A	F100-PW-100	FM04401321F	31 OCT 91	LOW SPD TRAINING RT
39	Added Training Route Entry	F-16C	F110-GE-100	FM04404291F	09 SEP 94	MID SPD TRAINING RT
40	Added Training Route Entry	F-16C	F110-GE-100	FM04404301F	09 SEP 94	HIGH CRU TRAINING RT
41	Added Training Route Entry	F-16C	F110-GE-100	FM04404311F	09 SEP 94	HIGH SPD TRAINING RT
42	Added Training Route Entry	F-16C	F110-GE-100	FM04404321F	09 SEP 94	LOW SPD TRAINING RT
48	Added Training Route Entry	F-18A/C	F404-GE-400&402	FM04501211F	01 NOV 95	TRAINING ROUTE
49	Added Training Route Entry	F-18E/F	F414-GE-400	FM04502291F	01 JUL 97	MID SPD TRAINING RT
50	Added Training Route Entry	F-18E/F	F414-GE-400	FM04502311F	01 JUL 97	HIGH SPD TRAINING RT
51	Added Training Route Entry	F-18E/F	F414-GE-400	FM04502321F	01 JUL 97	LOW SPD TRAINING RT
52	Added Training Route Entry	F-4C	J79-GE-15E or -15	FM03901211F	10 FEB 89	TRAINING ROUTE
53	Added Training Route Entry	FB-111A	TF30-P-7	FM05204211F	10 FEB 89	TRAINING ROUTE
54	Change Power Units from '%TORQUE' to '%N1'	JPATS	TP PT6A-68	FM08301031V	19 DEC 96	TAKEOFF POWER
55	Change Power Units from '%TORQUE' to '%N1'	JPATS	TP PT6A-68	FM08301051V	19 DEC 96	APPROACH POWER
56	Add New Aircraft	T-38C	J-85-GE-5	FM06802011F	15 SEP 03	AFTERBURNER POWER

57	Add New Aircraft	T-38C	J-85-GE-5	FM06802031V	27 DEC 79	TAKEOFF POWER
58	Add New Aircraft	T-38C	J-85-GE-5	FM06802041V	15 SEP 03	CRUISE POWER
59	Add New Aircraft	T-38C	J-85-GE-5	FM06802051V	15 SEP 03	APPROACH POWER

Table 2. Aircraft added to 'MRNMap Old-to-New Aircraft Code Table.txt'

Old AC Code	Aircraft	Power Description	Power Setting	Old Airspeed
22	A-10A	TRAINING ROUTE	5333 NF	325
30	B-1B	TRAINING ROUTE	101 % RPM	550
41	B-52G	TRAINING ROUTE	88 % RPM	340
45	B-52H	TRAINING ROUTE	4500 LBS/HR	350
53	FB-111A	TRAINING ROUTE	98 % NC	525
58	C-5A	TRAFFIC PATTERN	3.07 EPR	165
75	C-18A	TRAINING ROUTE	1.10 EPR	240
91	C-135A	TRAINING ROUTE	86 % RPM	250
102	C-141A	INTERMEDIATE POWER	68 % NF	140
104	C-141A	TRAINING ROUTE	80 % NF	200
116	F-4C	TRAFFIC PATTERN	86.5 % RPM	200
117	F-4C	TRAINING ROUTE	98 % RPM	550
129	F-14A	INTERMEDIATE POWER	92 % NC	400
131	F-14A	TRAINING ROUTE	100 % NC	530
132	F-14A	LOW SPD TRAINING RT	96 % NC	460
138	F-14B	TRAINING ROUTE	100 % NC	550
139	F-14B	LOW SPD TRAINING RT	95 % NC	460
144	F-15A	MID SPD TRAINING RT	81 % NC	520
145	F-15A	HIGH SPD TRAINING RT	88 % NC	570
146	F-15A	TRAINING ROUTE	82 % NC	550
147	F-15A	LOW SPD TRAINING RT	77 % NC	450
153	F-16A	TRAINING ROUTE	84 % NC	500
154	F-16A	MAX ENDURANCE	78 % NC	250
155	F-16A	LOW SPD TRAINING RT	82 % NC	370
156	F-16A	MID SPD TRAINING RT	87 % NC	450
162	F-16(G100)	HIGH SPD TRAINING RT	101 % NC	585
163	F-16(G100)	LOW SPD TRAINING RT	94 % NC	465
164	F-16(G100)	MID SPD TRAINING RT	95.4 % NC	500
165	F-16(G100)	HIGH CRU TRAINING RT	99 % NC	540
172	F-18	TRAINING ROUTE	92 % NC	500
178	F-18E/F	MID SPD TRAINING RT	84.5 % N2	400
179	F-18E&F	HIGH SPD TRAINING RT	90.5 % N2	500
180	F-18E/F	LOW SPD TRAINING RT	83.3 % N2	370
201	F-111F	HIGH SPD TRAINING RT	97 % NC	610

202	F-111F	LOW SPD TRAINING RT	88 % NC	450
203	F-111F	LOW CRU TRAINING RT	94 % NC	490
204	F-111F	MID SPD TRAINING RT	90 % NC	500
205	F-111F	HIGH CRU TRAINING RT	93 % NC	540
213	HARRIER	TRAFFIC PATTERN	65 % RPM	313
221	HS748	TRAFFIC PATTERN	71 % RPM	125
295	TORNADO	TRAFFIC PATTERN	82 % RPM	297

3. Updated Noise Calculations

The original version of MRNMap was created at a time when computer memory and processing power were the limiting factors for computations. For this reason, many of the methods used in MRNMap were developed to limit memory and processor requirements. With modern computers, high processing speeds and storage capacity are readily available, so more robust computations can be employed.

One of the new approaches calculates the noise within an area operation in multiple discrete altitude layers versus the original 'Effective Acoustic Altitude' (EAA), which was used in the original version of MRNMap. Aircraft missions are now defined in the initial development of a case by using specific altitude bands with a percentage of the missions assigned to utilize each altitude bands. The process for computing noise starts with these bands as layers, and then adds additional altitude sub-layers to increase the vertical resolution of the analysis. Aircraft operations are apportioned with the sub-layers to represent a linear distribution of aircraft within each mission-defined layer. The altitudes of the sub-layers are spaced such that each successive altitude would be approximately 1.122 times the previous altitude. This ensures that:

$$20 * log10\left({}^{alt2}/_{alt1}\right) = 1$$

This spacing was selected because it provides no more than a one-decibel variation from altitude sub-layer to sub-layer (assuming spherical spreading only). Each mission-defined altitude layer is used, with the additional sub-layers defined in the space between the mission layers. Figure 1 provides an example of the altitude layers used for a particular mission. In this case, the mission is defined with operations between 100 feet and 1,000 feet, 1,000 feet and 5,000 feet, and 5,000 feet and 10,000 feet, showing both logarithmic as well as linear spacing on the altitude bands.

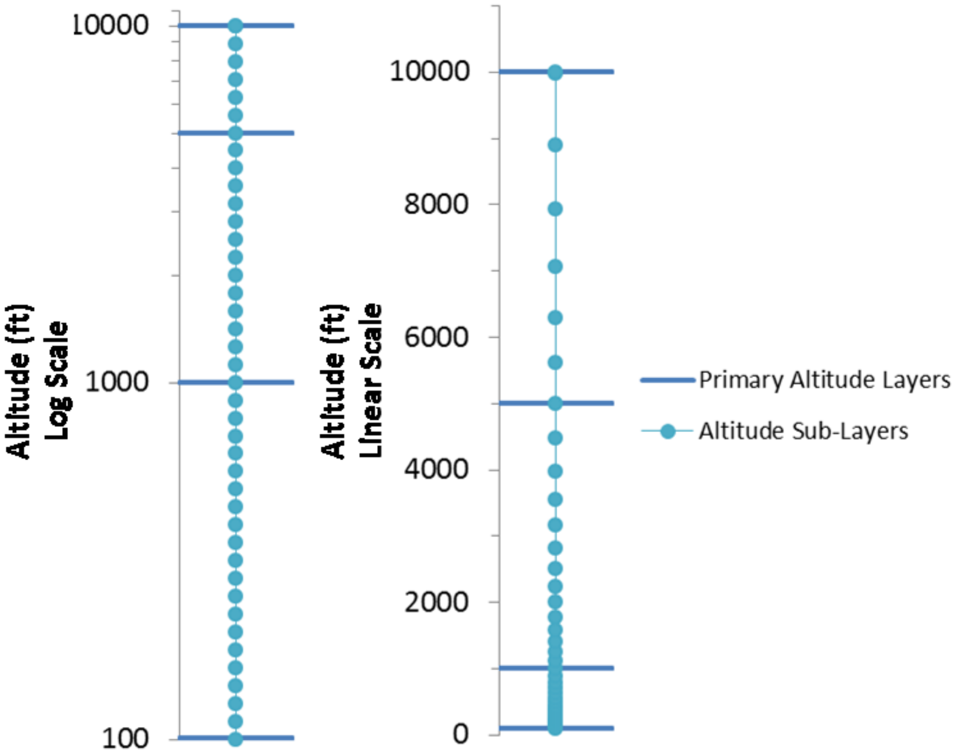


Figure 1. Mission altitude layers and altitude sub-layers.

The advantage of using these finely defined altitude sub-layers provides detailed analysis that can include specifics about each altitude. With the previous EAA system, only one altitude was analyzed. If an Avoidance Area did not happen to occur at that altitude, it would not be considered. Hence, by using multiple altitudes it is possible to ensure that all aspects of the airspace are accounted for accurately.

3.1 Military Operating Area Calculations

One of the crucial aspects of completing these calculations requires the accurate determination of the usable area of each MOA, including the edges affected by the standoff distance and the impacts from avoidance areas. MOAs are analyzed in raster format to avoid the complex mathematical manipulations necessary to analyze intersecting three-dimensional polygons. Each layer and sub-layer is analyzed individually by 'drawing' the MOA and all aspects involved at that altitude. Each sub-layer is then combined to complete the analysis.

At each sub-layer the MOA area is 'drawn' onto a rectangular memory bitmap using windows-based drawing primitives. The memory bitmap is assigned coordinates that are the same as the study area, and the polygons defining the MOA are then drawn onto this bitmap using these defining coordinates. For ease of testing and analyzing this process, the MOAs are drawn in red (number 255 for the memory bitmaps), and avoidance areas that intersect the MOA are 'drawn' in black (number 0 in the memory bitmap). Figure 2 is an image a low altitude sub-layer of the usable airspace for the Townsend Range and Costal MOA Complex at a low altitude, showing the usable areas in red and the areas where aircraft cannot fly in black. The circular cutouts shown are avoidance areas contained within the airspace complex.

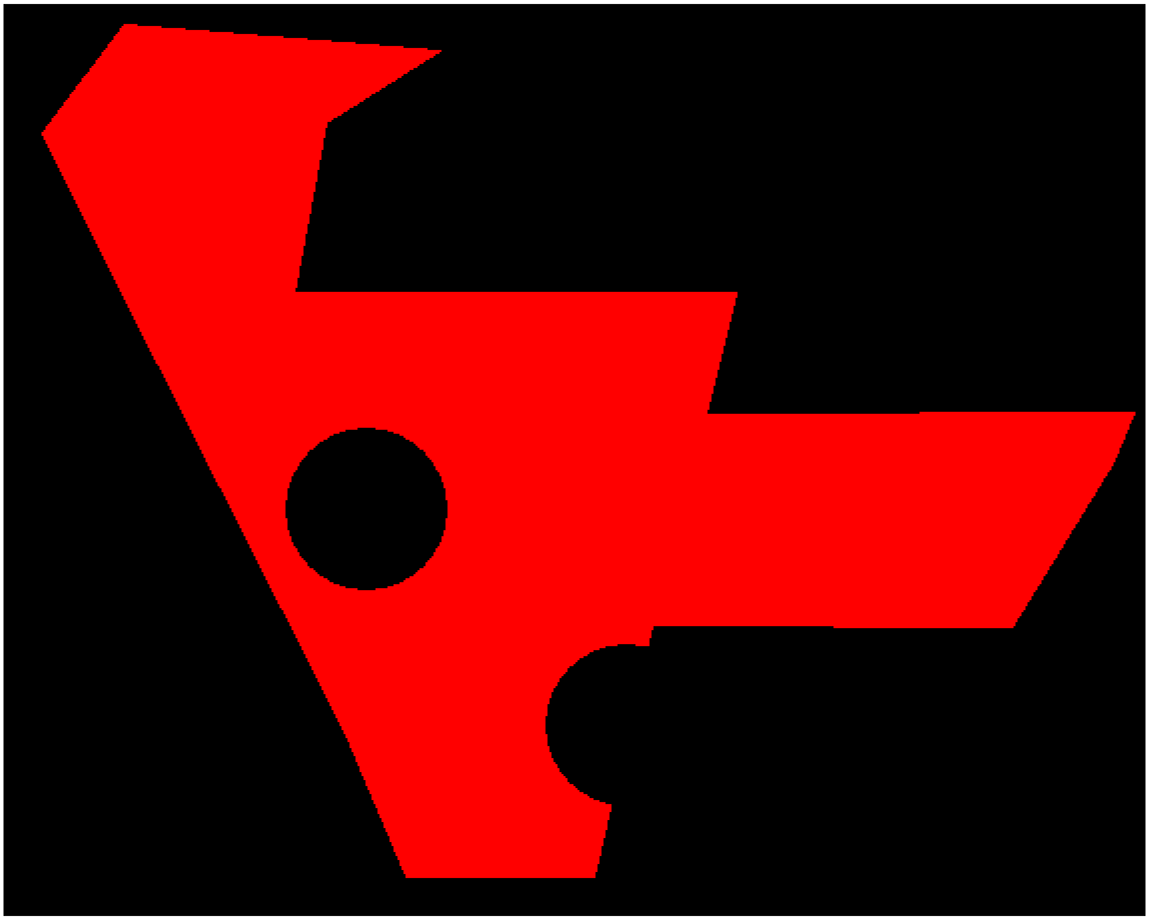


Figure 2. Image of the defined airspace for the Townsend Airspace.

An additional advantage of this system is that the area of a complex polygon, such as the one shown in Figure 2, can be easily computed by summing all non-black/red pixels and multiplying by the pixel size. In this way, multiple nested avoidance areas with different altitudes are not double counted when decrementing.

Once the usable airspace has been identified, the next step is to create a 2-dimensional array of multipliers that show how the different locations within the airspace effect the noise calculations. These multipliers are computed using a taper function.

3.2 Taper Calculations

MRNMap assumes operations within MOAs and ranges are uniformly distributed, which results in the noise being uniformly distributed over the entire area. However, a more complicated calculation is required to compute noise levels at locations near the boundary where the operations may be distributed non-uniformly or near an avoidance area where operations are prohibited. The taper algorithm is designed to have a smooth transition in noise levels from one defined area to the next.

The basic concept of a taper calculation is to reduce the noise toward and beyond the edges of the MOA. Tapering allows the noise to extend beyond the MOA, but only to a level consistent with actual sound propagation. For situations with an offset to the boundary (i.e. an internal buffer area to avoid spilling out of the defined airspace), the distribution within the buffer (or transition zone) is not uniform. The distribution within the transition zone assumes that aircraft will still enter the buffer, but that they will decrease linearly from the start of the offset to zero at the MOA boundary. For example, if a mission has a 3 nautical mile (NM) offset, any internal area that is farther than 3 NM from the MOA boundary will be unaffected. A location that is 1.5 NM from the MOA boundary is assumed to only experience one-half of the total aircraft operations.

The taper values are computed for a set number of points, spaced in 500 feet increments, starting 25,000 feet inside of the MOA boundary and extending to 50,000 feet outside the MOA. For each of these analysis points the noise level is calculated by accumulating the energy received from an array of sources within the MOA boundary. This value is then normalized to provide a range of values from 0 to 1, representing the level of reduction caused by the proximity to the MOA boundary.

Figure 3 shows a graphical representation of the edge of the MOA. The circles represent distributed identical noise sources, all at a given altitude. The diameter of the circle represents the relative number of operations. For example, within the MOA all of the circles are ‘full sized’, meaning that they represent a full complement of operations. As you move through the transition zone between the Mission Offset and the MOA boundary the effective number of operations decreases linearly to zero operations at the MOA boundary.

One representative location has been shown with a red circle on the ground for the analysis where the cumulative effect of all of the sources is combined. At this location, each noise source is propagated to the receiver location, with the only variation being the representative numbers of operations. Once all of the analysis points have been computed, from 25,000 feet within the MOA to 50,000 outside of the MOA, the entire set is normalized by the maximum level, producing an array of numbers representing the impact of the taper on the evenly distributed noise within the MOA. Figure 4 is a plot that demonstrates the impact of the taper as a function of distance to the MOA boundary for an aircraft at 1,500 feet above ground level (AGL).

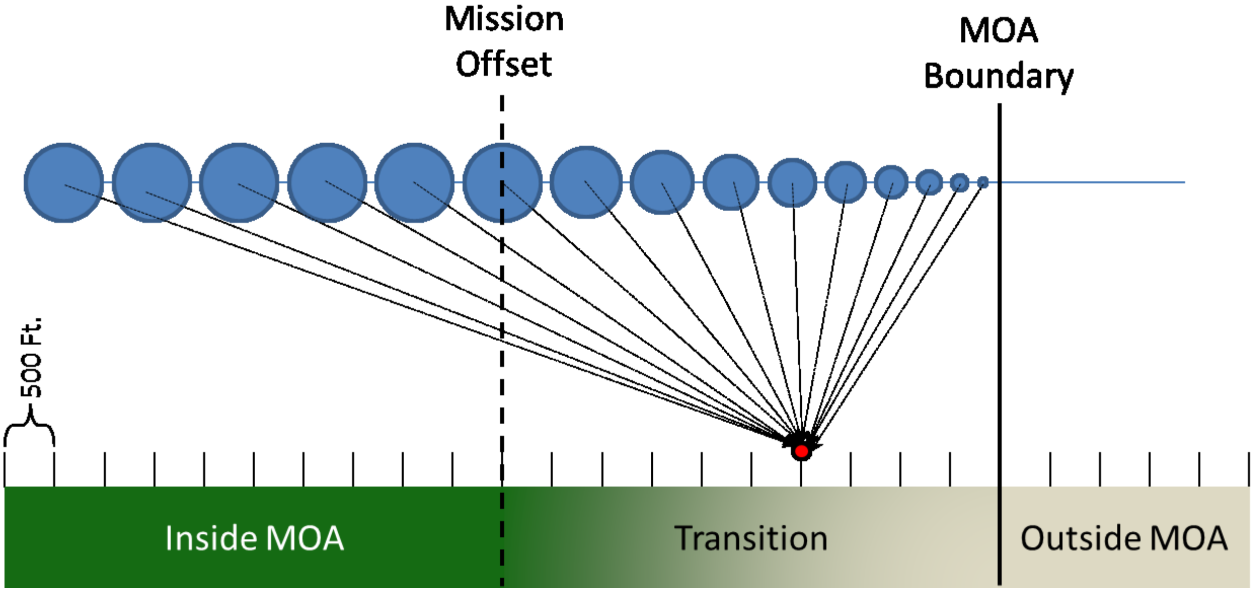


Figure 3. Graphical representation of the taper algorithm.

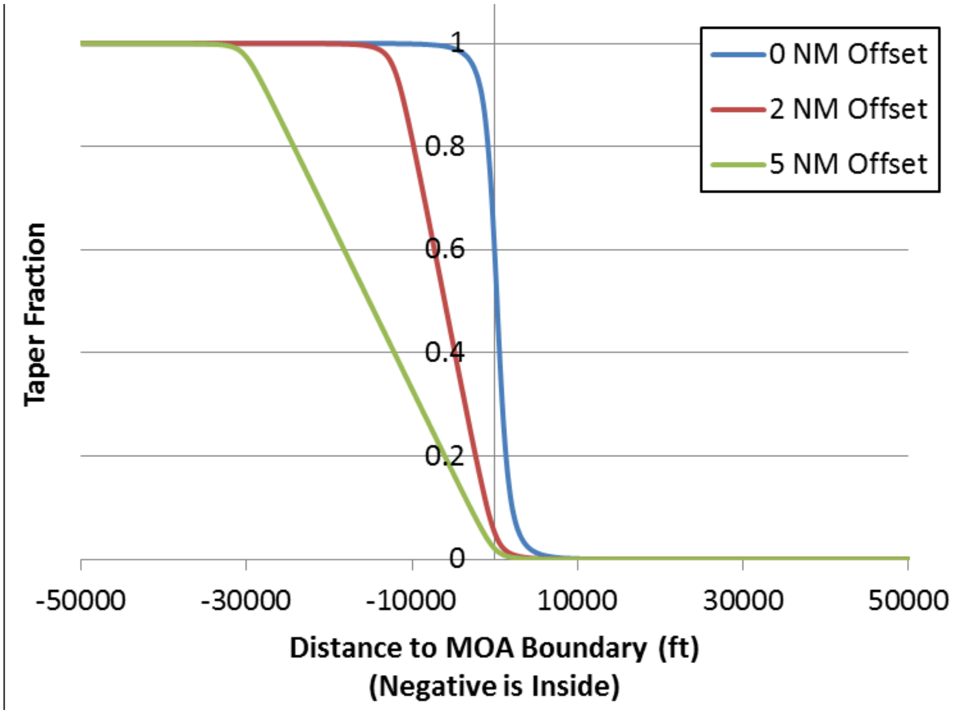


Figure 4. Representative taper fractions for an aircraft at 1,500 feet AGL.

The taper calculation is added to the MOA definition to provide a gradient from the center of the MOA to the boundary. At each point within the MOA, the distance to the MOA boundary is computed and the taper fraction is determined. For most of the interior of the MOA, the taper fraction is 1.0, representing no impact from the MOA edge effects. However, as the boundary is approached the taper fraction decreases, as shown in Figure 4. If a mission has an offset, this offset is taken into consideration. If an avoidance area is encountered, this taper fraction is always calculated with a zero offset. Figure 5 shows an example of the taper fraction applied to the Townsend MOA from Figure 2, where the avoidance areas and the 5 NM mission offset are clearly evident. For this plot, red represents locations with a taper fraction of 0.0 and blue represents locations with a taper fraction of 1.0.

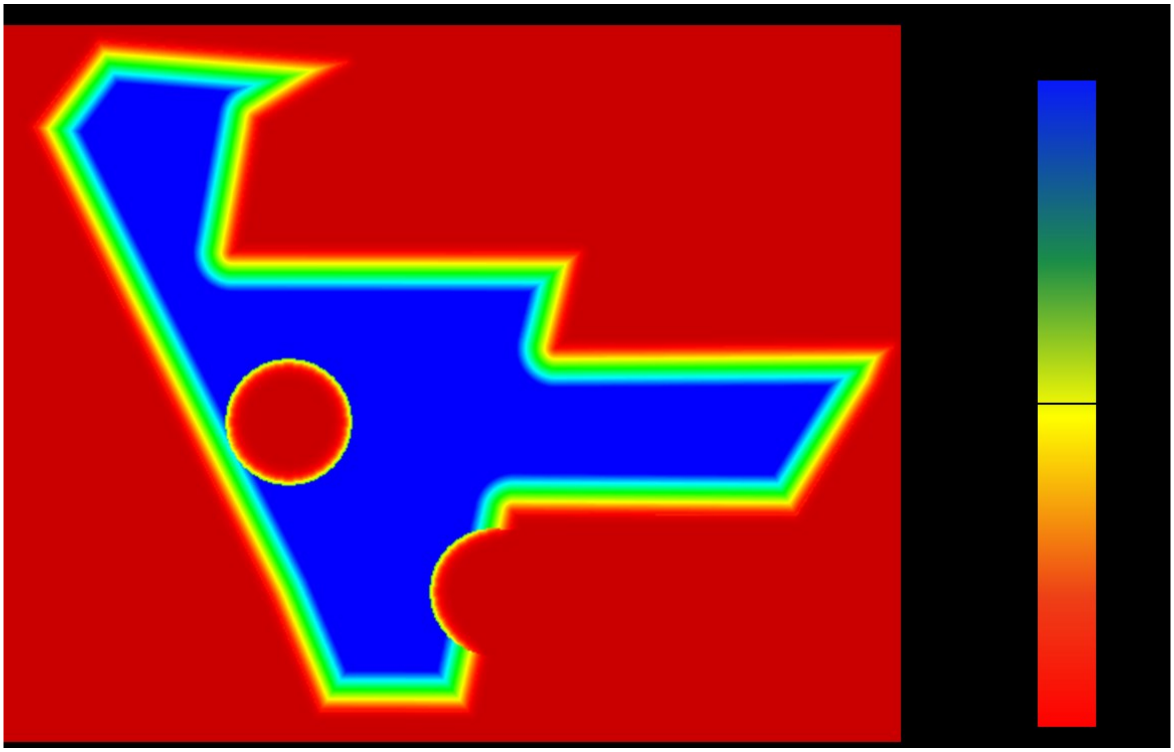


Figure 5. Example of the taper fraction applied to the MOA.

The data shown in Figure 5 are used directly to compute the noise. The averaged noise level for the center of the MOA is multiplied by the grid of taper fractions shown in Figure 5 to produce the final noise grid for this specific mission/altitude combination.

This same methodology is used to determine the noise at any specific point. The distance to the MOA boundary is calculated and the appropriate taper fraction is applied to the average noise level from the center of the MOA.

4. Advanced Number of Events Calculation

The Number of Events calculation is designed to provide the user with an estimate of the number of exceedance events. Exceedance events are times when a specific noise level threshold is exceeded somewhere within a given MOA. This calculation assumes that the aircraft operations are evenly distributed throughout the MOA.

The Number of Events calculation is based on three main concepts: the total numbers of passes that can be expected from a given aircraft, the amount of the MOA each pass would cover, and any impact from the proximity to the MOA boundary or avoidance areas. To begin this analysis a characteristic length for the MOA, L_{MOA} is estimated as the square root of the MOA area:

$$L_{MOA} = \sqrt{A_{MOA}}$$

Then the distance that an aircraft will fly during its mission is computed. This is defined as the velocity of the aircraft multiplied by the time it spends in the MOA:

$$L_{AC} = V_{AC} * t$$

The total number of passes is then calculated as the ratio of L_{MOA} and L_{AC}, defining the number of times the aircraft needs to traverse the MOA given its flight path length and the MOA characteristic length:

$$\#passes = \frac{L_{AC}}{L_{MOA}}$$

Just because the aircraft traversed the MOA does not necessarily mean that it generated sufficient noise to exceed the defined threshold for the entire MOA. The potential to exceed a given level is governed by propagation distance of the noise (L_{ground}). L_{ground} is determined by the Noise-Power-Distance (NPD) data for the aircraft, with the given ground distance calculated to the desired exceedance threshold. Since the noise travels away from both sides of the aircraft, the fraction of the noise that could contribute to an exceedance event is:

$$Frac_{exceed} = \frac{2 * L_{ground}}{L_{MOA}}$$

Therefore, for a single aircraft, the average total number of exceedance events for a given mission is:

$$NumEvents = Ops * \#passes * Frac_{exceed}$$

As an example, consider an aircraft traveling at 300 knots (345 MPH) for 30 minutes in a MOA whose total area is 1,500 square NM. It is desired to estimate the Number of Events that exceed an Lmax value of 65 dB. For calculation purpose, the assumed distance to reach 65 dB Lmax is approximately 3 NM.

The characteristic length for the MOA is $\sqrt{1500}$ = 38.7 miles, and the aircraft will travel 345/2 = 172.5 miles during its 30 minute flight time. Based on the characteristic length of the MOA, the aircraft will transit the MOA approximately four and a half time (172.5 NM/38.8 NM = 4.5). The fraction of the MOA that receives noise exceeding 65 dB Lmax is 2*3/38.7 = 0.155. Thus, for each 30-minute flight, a given location within the MOA will receive, on average, 0.7 events above 65 dB (4.5*0.155 = 0.7). If 75 operations occur per month, that location will receive an average of 52.5 events above 65 dB (75 *0.7 = 52.5). It should be reiterated that this calculation is based on the uniform distribution of operations within the MOA.

5. BaseOps Interface

The updated version of MRNMap runs using BaseOps for the GUI. Specifically, MRNMap now uses a BaseOps generated input file, which offers many advantages over the original MROps GUI. The new GUI allows users to import airspace information directly from Digital Aeronautical Flight Information File (DAFIF) files, and the GUI closely follows NoiseMap methodologies. However, while the GUI works to generate the input data, the seamless integration of MRNMap within BaseOps is not yet completed. Thus, additional effort is required from the user to run MRNMap and generate the appropriate results.

Several steps must be taken before an input file for MRNMap can be created. These steps are detailed in Appendix A. Unfortunately, making the above changes means that BaseOps will not work with NoiseMap during the same work session. It is intended that BaseOps will be updated to allow for seamless integration with MRNMap.

In addition to the issue of integration, several aspects of the BaseOps GUI and MRNMap input file generation will require improvements. Namely, several MRNMap Version 3.0 functions are not accessible with the current BaseOps (Version 7.358). These functions are the following:

1. The SELU (Sound Exposure Level, unweighted) is not a selectable option in BaseOps.
2. BaseOps and the Version 3.0 input file already provide flags that can be used in MRNMap to calculate a grid only, specific points only, or both of these results (using parameters Calculate Grid = Yes/No and Calculate Specific Points = Yes/No located in the CASE section). When a legacy .INP file is converted to the new format in BaseOps, it may contain the keyword ‘ONLY SPECIFIC POINT’, which is used in the old format to generate results only for the specific points (no grid file). For compatibility, BaseOps should scan legacy files for this keyword and set the two parameters above accordingly.
3. Military Training Route (MTR) Specification: The BaseOps MTR definition is currently defined as point-to-point to be consistent with DAFIF inputs. All inputs should remain

the same, with the exception of the Floor Variation input, which should be removed from the MTR entry screen; this parameter has no use in MRNMap. The user will be responsible for changing any related parameters such as the Left Width, Right Width and Width Variation. Only the route Left Width and Right Width will be specified in BaseOps, the parameter sigma will no longer be specified as an input for MTRs or closed range tracks.

4. To be compatible with the previous version of MRNMap, an additional Range Track Specification should be developed. This addition would be a separate category in the BaseOps object type selector for 'Closed Range Tracks' (representing user-defined tracks that could be either closed range tracks, bombing tracks, refueling tracks, strafing tracks, etc.). Tracks built with this new input feature would utilize the same capability that already exists in BaseOps for airfield flight tracks (i.e. permitting the definition of both straight segments and turn segments). However, additional inputs for each segment would include the Left Width, Right Width, and Width Variation values and the airspace floor (not the Floor Variation); straight segments would require a single floor value whereas turn segments would require two floor values (one at the turn entry point and one at the turn exit point) consistent with current MRNMap requirements. This new track specification should be written to the .INS/.INX files using the old track specification format. Legacy Track Flags LW, TW and NW would be included in this specification. Having two track specification formats in the input file may require a new keyword or some other differentiating factor.
5. Several old aircraft codes used in MRNMap Version 2.2 are no longer supported by BaseOps. BaseOps uses a conversion file to go from the old aircraft codes to the new set of parameters (including aircraft type and engine power setting) which reference a specific noise dataset; this is the MRNMap Old-to-New Aircraft Code Table.txt file, located in the Noisemap\MRNMap directory. In cases where an outdated code is used in a legacy file, BaseOps will issue a warning, during the input file conversion process, that not all elements were converted. It is up to the user to select a current aircraft type and operating condition to be used in place of a legacy code not converted by BaseOps. It is highly recommended that conversion of all legacy files be reviewed for proper translation since the legacy codes were not consistent.

6. Standalone Executable File

As part of the integration of MRNMap with BaseOps, the process for generating the input noise data required modifications. In the original version of MRNMap, the processing routines for input data were an internal set of subroutines, Omega10R, that converted the original reference noise data (stored in the 'Noise' file) into usable NPD tables for the program to use. These subroutines have been modified to run as a stand-alone program named Omega10R.exe. This processing program takes in the data from an updated reference noise data file called 'Flight02.dat', which must be co-located in the same directory as Omega10R. BaseOps calls this program and then append the NPD data to the input file for MRNMap 3.0.

Appendix A MRNMap Version 3.0 Input File Format

Written by Fred Wasmer of Wasmer Consulting

Updated 9 July 2007

MRNMap is a computer program developed by the United States Air Force that is used to predict the ground-level noise impacts from aircraft operating in Military Operating Areas (MOAs) and Military Training Routes (MTRs). MRNMap reads an input file describing the scenario being modeled, performs the noise modeling calculations, then writes the results to one or more output files.

This document describes the format of the input file for MRNMap version 3.0.

A.1 File Format Overview

An MRNMap input file is a text file that may be edited by hand using a standard text editor. Lines in the file are separated by either newline characters (ASCII code 10) or carriage return-newline character pairs (ASCII codes 13 and 10).

The first line in the input file must consist of the text "MRNMAP INPUT FILE". Programs can determine if a file is an MRNMAP input file by checking if the file starts with this text.

Comments may be inserted anywhere in the input file after the initial "MRNMAP INPUT FILE" line. A comment is a line beginning with a pound character #. Blank lines may also appear anywhere in the input file. Blank lines and comments are ignored by MRNMap.

An input file consists of sections. Each section begins with the section name, on a line by itself, followed by one or more lines containing the data for that section. The section names are as follows.

MRNMAP INPUT FILE
CASE
SPECIFIC CALCULATION POINTS
AVOIDANCE AREAS
MOAS
MTRS
MISSIONS
MOA OPERATION SETS
MTR OPERATION SETS

FLIGHTNOISE
ENDNOISE

All sections must be present in the order listed.

A section's data is composed of two types of elements.

The first type of element is a name/value pair. This consist of the name of a parameter, optionally followed by spaces and/or tabs, followed by an equals sign, optionally followed by spaces and/or tabs, followed by the value of the parameter.

Here is an example of a name/value pair from the CASE section.

```
Temperature (F) = 59
```

In this example, "Temperature (F)" is the parameter name, and 59 is the value.

The second type of element is a table. Tables are always preceded by a name/value pair that specifies the number of rows in the table. Following that, the table is listed, one file line per table row. Items on a row are separated by one or more spaces and/or tab characters. If an item consists of text with embedded spaces, then it must be enclosed in double-quotes.

The names of table columns are not required to appear in the input file. However, it is suggested that when creating a table, one or more comment lines be added before the first row, listing the column names.

Here is an example of a table from the AVOIDANCE AREAS section.

```
Number of Avoidance Areas = 2

#      Name         X       Y     Radius  Ceiling
#                  (ft)    (ft)    (ft)     (ft)
#-----------------------------------------------
   "Humbolt A"   40000  -20000    5000     4000
   "Humbolt B"   20000  -22000    5000     3500
```

Further in this document, the name/values pairs and tables that comprise each section are described. All name/value pairs and tables must be present in the order they are listed in this document.

Note that the FLIGHTNOISE section is an exception to these formatting rules. See the discussion of the FLIGHTNOISE section below for more information.

A.2 Limits

Unless otherwise specified in this document, MRNMap imposes no limitations on the numeric data in the input file. In particular, no limits are placed on numbers of objects (for example, the number of MOAs), numbers of rows in tables (for example, the number of points in an MTR track), or lengths of items (for example, the number of characters in an avoidance area name).

In practice, these items are limited by your computer's memory. But given the capacity of contemporary computers, it is unlikely that such limits will be of practical importance.

MRNMap places few limitations on physical values (for example, aircraft airspeed). Experience has led MRNMap's developers to believe that users often use models in ways that are unexpected but perfectly legitimate, and that seemingly plausible limits can restrict such users.

A.3 Physical Units

MRNMap input files use English units of measurement: feet for distance, degrees F for temperature, and inches Hg for pressure. The expected units are listed as part of the names of all name/value pairs. For tables, it is recommended that the units for each column be listed in a comment preceding the first table row.

Heights are specified in feet above ground level (AGL).

A.4 Coordinate System

In an MRNMap input file, locations are specified in a Cartesian X/Y coordinate system, with distances measured in feet east and north of the origin. The location of this coordinate system, with respect to the surface of the Earth, is defined by a reference point; both the X/Y and longitude/latitude coordinates of this reference point are specified in the CASE section.

MRNMap does all of its calculations in the X/Y coordinate system. It does not directly use the reference point information. However, it does writes the reference point X/Y and longitude/latitude coordinates to the output grid file.

The grid file can later be read by the NMPlot contouring application. NMPlot can display noise contours on top of a background map. In order for the contours to appear in the proper location on the map, they must be georeferenced. NMPlot uses the reference point information to perform this georeferencing.

In order for the georeferencing to work, the person or computer program who creates a MRNMap input file must use the same method (i.e., the same cartographic projection) as NMPlot uses to convert between longitude/latitude and X/Y coordinates.

NMPlot uses the method used by the United States Federal Aviation Administration's Integrated Noise Model (INM). This method is described in an appendix of the INM User's Guide.

It is expected that most users will create MRNMap input files using the BaseOps preprocessor. BaseOps handles all coordinate system conversion issues, so users of BaseOps will not need to concern themselves with the details.

As a convenience to those users who wish to create an MRNMap input file independently of BaseOps, the INM projection method is summarized here.

Let:

A = 6378137.000 m = radius of equator circle (WGS-84 spheroid)
B = 6356752.314 m = distance from earth center to a pole (WGS-84 spheroid)
$\phi_0\ \lambda_0$ = latitude and longitude of the X/Y coordinate system origin (radians)
$\phi\ \lambda$ = latitude and longitude of a point on the earth (radians)
$x\ y$ = coordinate values of the same point (meters)

$$R_p = A^2 / \mathrm{sqrt}\,(A^2 \cos^2\phi_0 + B^2 \sin^2\phi_0)$$
$$R_m = R_p^3\, B^2 / A^4$$
$$E_0 = 1/2\, \tan\phi_0 / R_p$$

To convert from longitude/latitude to X/Y coordinates, use the following equations:

$$x = (R_p \cos\phi_0 - y_0 \sin\phi_0)\,(\lambda - \lambda_0)$$
$$y = y_0 + E_0 x^2$$

...where...

$$y_0 = R_m\,(\phi - \phi_0)$$

To convert from X/Y to longitude/latitude coordinates, use the following equations:

$$\phi = \phi_0 + y_0 / R_m$$
$$\lambda = \lambda_0 + x / (R_p \cos\phi_0 - y_0 \sin\phi_0)$$

...where...

$$y_0 = y - E_0\, x^2$$

Note that in the above equations, distances are specified in meters. X and Y coordinates will need to be converted to feet for use in an MRNMap input file. Also notice that angles are specified in radians.

A.5 MRNMAP INPUT FILE Section

This section identifies the file as an MRNMap input file. It consists of the following data element.

`Version = <number>`

File format version of this input file. This should be 3.0 for the current version.

A.6 CASE Section

This section contains general information about the scenario being modeled. It consists of the following data elements.

`Noise Metric = <symbol>`

The noise metric that will be calculated. *`<symbol>`* must be one of the following:

`CNEL`	Community noise equivalent level
`CNELR`	Onset rate-adjusted community noise equivalent level
`LDN`	Day-night average sound level
`LDNMR`	Onset rate-adjusted monthly day-night average A-weighted sound level
`LEQ`	Equivalent continuous sound level
`LMAX`	Maximum A-weighted sound level
`SEL`	Sound exposure level
`SELR`	Onset rate-adjusted sound exposure level

```
Reference Point Longitude (deg east) = <number>
Reference Point Latitude (deg north) = <number>
Reference Point X (ft)               = <number>
Reference Point Y (ft)               = <number>
```

```
-180.0 <= Reference Point Longitude <= 180.0
-90.0 < Reference Point Latitude < 90.0
```

The coordinates of a reference point, in both the longitude/latitude and X/Y coordinate systems. Longitude and latitude are specified in decimal degrees. Note that longitude is specified in degrees of east longitude, so for locations in the United States, longitude will be negative.

To minimize distortion, a reference point should be selected near the center of the area being modeled.

```
Grid Left X (ft)  = <number>
Grid Right X (ft) = <number>
Grid Lower Y (ft) = <number>
Grid Upper Y (ft) = <number>
Num X Grid Points = <integer>
Num Y Grid Points = <integer>
```

```
Grid Left X < Grid Right X
Grid Lower Y < Grid Upper Y
```

```
Num X Grid Points >= 2
Num Y Grid Points >= 2
```

The dimensions of the grid of points where noise calculations will be performed. `Grid Left X`, `Grid Right X`, `Grid Lower Y`, and `Grid Upper Y` define the X and Y coordinates of the edges of the grid.

`Num X Grid Points` and `Num Y Grid Points` define the number of grid points in the X and Y directions.

The X coordinates of the first and last columns of grid points will be `Grid Left X` and `Grid Right X`, respectively. The remaining grid point columns are distributed uniformly in between.

Rows of grid points are distributed in the Y direction in a similar fashion.

Beware of one-off errors when calculating the number of grid points. For example, if your grid area is 100,000 feet wide, and you want grid points every 1,000 feet, then `Num X Grid Points` should be 101, not 100

```
Temperature (F)              = <number>
Relative Humidity (%)        = <number>
Atmospheric Pressure (inHg) = <number>
```

The weather conditions that are assumed when calculating atmospheric noise absorption. Realistic meteorological conditions should be supplied, as the equations for calculating absorption break down in extreme circumstances.

The following default values are suggested:

```
Temperature (F)             = 59
Relative Humidity (%)       = 70
Atmospheric Pressure (inHg) = 29.92
```

```
Average Flying Days per Month = <number>
```

```
Average Flying Days per Month > 0
```

The average number of flying days per month. This value is used when the number of daily operations must be calculated from the number of monthly operations.

The following default value is suggested:

```
Average Flying Days per Month = 30
```

```
Noise Calculation Cutoff (dB) = <number>
```

The SEL sound level below which MRNMap will ignore individual noise events.

The following default value is suggested:

`Noise Calculation Cutoff (dB) = 65`

It is suggested that you use the default value unless instructed to do otherwise by MRNMap support staff.

```
Calculate Grid            = YES
Calculate Specific Points = YES
```

Both of these parameters should have the value of either YES or NO. They control whether or not MRNMap calculates the noise level grid and the specific calculation points (sometimes known as points of interest) report, respectively. Note that if both of these parameters are set to NO, then MRNMap will not do anything.

A.7 SPECIFIC CALCULATION POINTS Section

Specific calculation points are locations where MRNMap does an in-depth noise impact analysis. These points are defined in this section, which consists of the following data elements.

```
Number of Specific Calculation Points = <integer>
Table with following columns:
   Name    <text>
   X (ft)  <number>
   Y (ft)  <number>
```

`Number of Specific Calculation Points >= 0`

`Number of Specific Calculation Points` specifies the number of points. The table contains this number of rows. Each row defines one point.

The `Name` column specifies a name for each point. A name can be any arbitrary text. Each point should be given a unique name.

The `X` and `Y` columns specify the location of each point, in X/Y coordinates.

A.8 AVOIDANCE AREAS Section

Avoidance areas are circular-column-shaped portions of airspace where aircraft are prohibited from flying These areas are defined in this section, which consists of the following data elements.

```
Number of Avoidance Areas = <integer>
Table with following columns:
   Name                <text>
   X (ft)              <number>
   Y (ft)              <number>
   Radius (ft)         <number>
   Ceiling (ft AGL)    <number>
```

```
Number of Avoidance Areas >= 0
Radius > 0.0
```

`Number of Avoidance Areas` specifies the number of areas. The table contains this number of rows. Each row defines one area.

The `Name` column specifies a name for each area. A name can be any arbitrary text. Each area should be given a unique name.

The `X` and `Y` columns specify the location of the center of each area, in X/Y coordinates. The `Radius` column specifies the radius of each area.

The `Ceiling` column specifies the height of the top of each area.

A.9 MOAS Section

MOAs (Military Operating Areas) are polygon-shaped areas with defined floor and ceiling altitudes where aircraft engage in dispersed operations. These areas are defined in this section, which consists of the following data elements.

```
Number of MOAs = <integer>
```

`Number of MOAs >= 0`

The number of MOAs specified. The remaining data elements in this section define an individual MOA, and are repeated `Number of MOAs` times.

```
Name = <text>
```

The name of a MOA. This can be any arbitrary text. Each MOA must have a unique name.

```
Floor (ft AGL)   = <number>
Ceiling (ft AGL) = <number>
```

`Floor <= Ceiling`

The heights above ground level of the bottom (`floor`) and top (`ceiling`) of a MOA.

```
Number of Boundary Points = <integer>
Table with following columns:
   X (ft)   <number>
   Y (ft)   <number>
```

`Number of Boundary Points >= 3`

This table defines the vertices of the polygon that defines a MOA's horizontal boundary. The polygon segments connect each adjacent pair of points, and the first and last points. The polygon segments should not cross each other.

A.10 MTRS Section

MTRs (Military Training Routes) are long, narrow volumes of airspace that aircraft following when traveling between two locations. Horizontally, an MTR is defined by a series of line segments denoting the centerline of the MTR, along with a width that can vary with location along the MTR centerline. Vertically, an MTR is defined by a floor height that can vary with location along the MTR centerline.

MTRs are defined in this section, which consist of the following data elements.

`Number of MTRs = <integer>`

`Number of MTRs >= 0`

The number of MTRs specified. The remaining data elements in this section define an individual MTR, and are repeated *`Number of MTRs`* times.

`Name = <text>`

The name of an MTR. This can be any arbitrary text. Each MTR must have a unique name.

```
Number of Points = <integer>
Table with following columns:
   X (ft)              <number>
   Y (ft)              <number>
   Left Width (ft)     <number>
   Right Width (ft)    <number>
   Width Variation     <symbol>
   Floor (ft AGL)      <number>
   Floor Variation     <symbol>
```

`Number of Points >= 2`
`Left Width >= 0`
`Right Width >= 0`

This table defines the centerline, width, and height profile of an MTR.

`X` and *`Y`* define the vertices of the segments that comprise an MTR's centerline. A centerline comprised of n line segments is defined by n+1 points. Zero-length line segments are not allowed: i.e., identical X and Y coordinates must not be repeated in the table.

`Left Width` and *`Right Width`* define the width of an MTR to the left and right of the centerline. Left and right are interpreted from the perspective of the pilot of an aircraft that is flying along the MTR from beginning to end. Zero widths are allowed.

`Width Variation` defines how an MTR's width varies along a line segment between two vertex points. It must have one of the following values

`CONSTANT`	The width is constant along the entire line segment.
`LINEAR`	The width varies linearly with distance along the line segment. By the end of the segment, the width has become equal to the width specified by the table row that defines the start of the next segment.

`Floor` defines the height above ground level of the bottom of an MTR.

`Floor Variation` defines how an MTR's floor varies along a line segment between two vertex points. It must have one of the following values

`CONSTANT`	The floor is constant along the entire line segment.
`LINEAR`	The floor varies linearly with distance along the line segment. By the end of the segment, the floor has become equal to the floor specified by the table row that defines the start of the next segment.

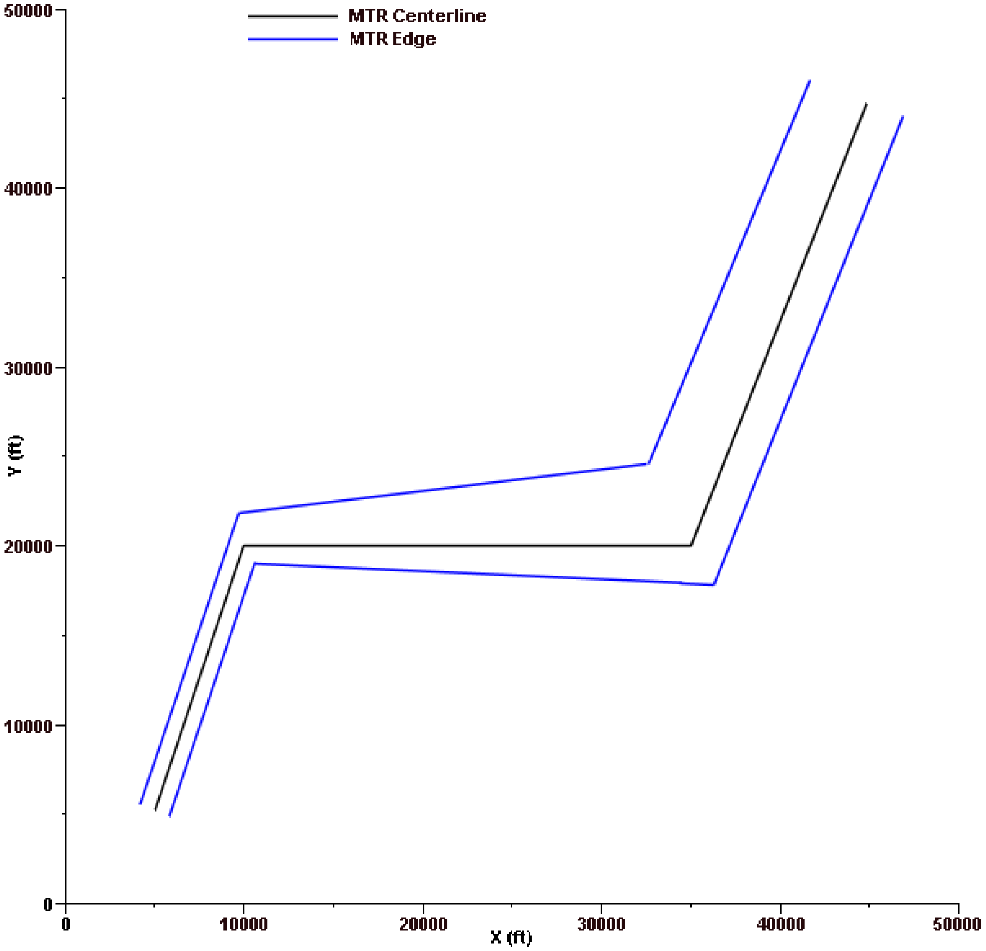
Both *`Width Variation`* and *`Floor Variation`* are ignored in the last row of the table.

As an example, consider the following MTR.

```
Number of Points = 4
#                   Left  Right    Width              Floor
#    X       Y     Width  Width  Variation   Floor   Variation
#   (ft)    (ft)    (ft)   (ft)             (ft AGL)
#--------------------------------------------------------------
    5000    5000    1250   1250   CONSTANT    3000    CONSTANT
   10000   20000    2000   1250    LINEAR     3000    CONSTANT
   35000   20000    4000   2000   CONSTANT    3000    CONSTANT
   45000   45000    4000   2000   CONSTANT    3000    CONSTANT
```

The width variation and floor variation in the last row are ignored. Furthermore, since both the width and floor variation in the second-to-last row are CONSTANT, the widths and floor values in the last row are also ignored.

The following figure shows the centerline and width of this example MTR.



A.11 MISSIONS Section

An MRNMap mission describes the aircraft operating parameters (noise profile, airspeed, and altitude profile) of a particular type of operation (for example, F-16 air combat maneuvers). Missions are defined in this section, which consists of the following data elements.

Note: Some parameters — Aircraft Name, Engine Name, Engine Power Setting, Power Setting Units, Operational Power Code, Operational Power Name, and Noise Interpolation Code — are not used by MRNMap in performing its calculations. MRNMap assume that the OMEGA10 program has already calculated the noise-verses-distance profiles and place them in the FLIGHTNOISE section. When MRNMap runs, it looks up the appropriate noise data using the Flight Noise Profile Id parameter.

MRNMap may, however, read these parameters and use them for reporting purposes. Other programs (for example, BaseOps) may also use these parameters when importing MRNMap input files.

`Number of Missions = <integer>`

`Number of Missions >= 0`

The number of missions specified. The remaining data elements in this section define an individual mission, and are repeated `Number of Missions` times.

`Name = <text>`

The name of a mission. This can be any arbitrary text. Each mission must have a unique name.

`Flight Noise Profile ID = <text>`

The id name of the Omega 10 noise profile representing the flight noise generated by an aircraft operating on this mission. This must match the id name of a flight noise profile in the FLIGHTNOISE section. See the discussion of the FLIGHTNOISE section below for additional information.

`Aircraft Name = <text>`

The name of the aircraft operating on this mission. This must match an aircraft name in NOISEFILE.

`Engine Name = <text>`

The name of the aircraft's engine. This must match an engine name in NOISEFILE.

`Operational Power Code = <integer>`

`Operational Power Code >= 0`

The integer code for the operational power setting used to describe this aircraft's operation on this mission. This must match an operational power code in NOISEFILE, or else be zero to indicate that the operational power code is not known.

`Operational Power Name = <text>`

The descriptive name of the operational power code: for example, "Cruise".

`Noise Interpolation Code = <symbol>`

The NOISEFILE noise interpolation curve used to estimate this aircraft's noise level. This must be one of the following: VARIABLE, PARALLEL, or FIXED.

`Engine Power Setting = <number>`

`Engine Power Setting > 0`

The numeric power setting of the aircraft's engine when operating on this mission.

`Power Setting Units = <text>`

The units used to express the aircraft engine's power setting: for example, "% RPM". This must match a power units in NOISEFILE.

`Airspeed (knots) = <number>`

```
Airspeed > 0.0
```

The airspeed of an aircraft while operating on this mission.

```
Standoff Distance (ft) = <number>
```

```
Standoff Distance >= 0.0
```

The distance from the boundary of a MOA where an aircraft begins to turn while operating on this mission. In this turning zone, the frequency of aircraft operations linearly decreases, reaching zero at the MOA's boundary.

`Standoff Distance` is ignored if this mission applies only to MTRs.

```
Number of Height Pairs = <integer>
Table with the following columns:
   Minimum Height (ft AGL)   <number>
   Maximum Height (ft AGL)   <number>
   Utilization (%)           <number>
```

```
Number of Height Pairs >= 1
Minimum Height <= Maximum Height
0 <= Utilization <= 100
```

This table defines the distribution of heights at which aircraft operate while on this mission. `Minimum Height` and `Maximum Height` define an altitude band, and `Utilization` defines the fraction of the time at which aircraft are operating in that band.

There are no required relationships between the heights of different bands. Bands can be specified in any order, and the heights of different bands can overlap.

The `Utilization` column would typically sum to 100%.

A.12 MOA OPERATION SETS Section

A MOA operation set defines the MOAs utilized by particular missions, and the frequency with which those missions are flown. MOA operation sets are defined in this section, which consists of the following data elements.

```
Number of MOA Operation Sets = <integer>
```

```
Number of MOA Operation Sets >= 0
```

The number of MOA operation sets. The remaining data elements in this section define an individual operation set, and are repeated `Number of MOA Operation Sets` times.

```
Name = <text>
```

The name of a MOA operation set. This can be any arbitrary text. Each operation set must have a unique name.

```
Number of MOAs = <integer>
Table with the following columns:
   MOA Name         <text>
   Utilization (%)  <number>
```

`Number of MOAs >= 1`
`0 <= Utilization <= 100`

This table specifies how the operations defined by a MOA operation set are distributed among one or more MOAs.

`MOA Name` must match the name of a MOA defined in the MOAS section.

The `Utilization` column would typically sum to 100%.

```
Number of Missions = <integer>
Table with the following columns:
   Mission Name         <text>
   Annual Day Ops       <number>
   Annual Evening Ops   <number>
   Annual Night Ops     <number>
   Air Time (min)       <number>
```

`Number of Missions >= 1`
`Annual Day Ops >= 0`
`Annual Evening Ops >= 0`
`Annual Night Ops >= 0`
`Air Time > 0`

This table specifies each of the missions flown as part of a MOA operation set, the number of operations per mission, and the air time per operation.

`Mission Name` is the name of a mission that was specified in the MISSIONS section.

`Annual Day Ops`, `Annual Evening Ops`, and `Annual Night Ops` specify the number of operations of a particular mission that occur per year. The number of monthly operations is calculated by dividing the yearly operations by 12, and the number of daily operations is calculated by dividing the monthly operations by `Average Flying Days per Month`, which is defined in the CASE section.

`Air Time` specifies the length of time, in aircraft-minutes, that an aircraft is airborne and operating in any MOA as part of this mission. MRNMap calculates a MOA's usage airtime for a mission by multiplying the number of operations by `Air Time`, then scaling by the MOA utilization.

A.13 MTR OPERATION SETS Section

An MTR operation set defines the MTRs utilized by particular missions, and the frequency with which those missions are flown. MTR operation sets are defined in this section, which consists of the following data elements.

`Number of MTR Operation Sets = <integer>`

`Number of MTR Operation Sets >= 0`

The number of MTR operation sets. The remaining data elements in this section define an individual operation set, and are repeated `Number of MTR Operation Sets` times.

`Name = <text>`

The name of an MTR operation set. This can be any arbitrary text. Each operation set must have a unique name.

```
Number of MTRs = <integer>
Table with the following columns:
   MTR Name         <text>
   Utilization (%)  <number>
```

```
Number of MTRs >= 1
0 <= Utilization <= 100
```

This table specifies how the operations defined by an MTR operation set are distributed among one or more MTRs.

`MTR Name` must match the name of an MTR defined in the MTRS section.

The `Utilization` column would typically sum to 100%.

```
Number of Missions = <integer>
Table with the following columns:
   Mission Name        <text>
   Annual Day Ops      <number>
   Annual Evening Ops  <number>
   Annual Night Ops    <number>
```

```
Number of Missions >= 1
Annual Day Ops >= 0
Annual Evening Ops >= 0
Annual Night Ops >= 0
```

This table specifies each of the missions flown as part of an MTR operation set, and the number of operations per mission.

`Mission Name` is the name of a mission that was specified in the MISSIONS section.

`Annual Day Ops`, `Annual Evening Ops`, and `Annual Night Ops` specify the number of operations of a particular mission that occur per year. The number of monthly operations is calculated by dividing the yearly operations by 12, and the number of daily operations is calculated by dividing the monthly operations by `Average Flying Days per Month`, which is defined in the CASE section.

A.14 FLIGHTNOISE Section

The FLIGHTNOISE section contains the noise-verses-distances tables used by MRNMap in performing its calculations.

This section is different from other sections, in that its format does not consist of name/value pairs and tables. Instead, it consists of raw Omega 10 output. This section must contain flight profile noise data for each of the flight noise profile IDs listed in the MISSIONS section.

It is expected that most users will create MRNMap input files using the BaseOps preprocessor. BaseOps handles the running of Omega 10 and the inserting of its output into the MRNMap input file, so users of BaseOps will not need to concern themselves with the details.

If you wish to create an MRNMap input file independently of BaseOps, you will be responsible for running Omega 10 and inserting the output into the MRNMap input file. See the Omega 10 documentation for more information.

Note that, unlike the rest of the MRNMap file, comments and blank lines may be not added to the FLIGHTNOISE section. This section must consist entirely of the Omega 10 output, exactly as it was written by Omega 10.

By convention, MRNMap input files are given the extension `.INS` before the Omega 10 output is added, and `.INX` after it is added. This mirrors the NMap convention of naming input files with the extensions `.OPS` and `.OPX`.

A.15 ENDNOISE Section

The ENDNOISE section marks the end of the Omega 10 output listed in the FLIGHTNOISE section. MRNMap does not read the input file past this point.

A.16 Example Input File

```
MRNMAP INPUT FILE

Version = 3.0

# This is an example MRNMap 3.0 input file.




CASE

Noise Metric = LDN

Reference Point Longitude (deg east) = -91.345
Reference Point Latitude (deg north) = 35.221

Reference Point X (ft) = 0
Reference Point Y (ft) = 0

Grid Left X (ft)  = -100000
Grid Right X (ft) =  100000
Grid Lower Y (ft) = -100000
Grid Upper Y (ft) =  100000

Num X Grid Points = 201
Num Y Grid Points = 201
```

```
Temperature (F)             = 59
Relative Humidity (%)       = 70
Atmospheric Pressure (inHg) = 29.92

Average Flying Days per Month = 30

Noise Calculation Cutoff (dB) = 65

Calculate Grid            = YES
Calculate Specific Points = YES




SPECIFIC CALCULATION POINTS

Number of Specific Calculation Points = 2

#       Name           X       Y
#                     (ft)    (ft)
#----------------------------------
  "Mercy Hospital"   40000   10000
  "Hope School"     -40000   50000




AVOIDANCE AREAS

Number of Avoidance Areas = 1

#      Name        X       Y     Radius  Ceiling
#                 (ft)    (ft)    (ft)    (ft AGL)
#-------------------------------------------------
   "Humbolt A"   40000  -20000   10000     5000




MOAS

Number of MOAs = 2


Name = Bulldog A

Floor (ft AGL)   =   500
Ceiling (ft AGL) = 12000
```

```
Number of Boundary Points = 4

#    X       Y
#   (ft)    (ft)
#----------------
  -80000   20000
  -60000   80000
  -20000   80000
  -20000   40000


Name = Bulldog B

Floor (ft AGL)    =  2000
Ceiling (ft AGL) = 10000

Number of Boundary Points = 4

#    X       Y
#   (ft)    (ft)
#----------------
   40000  -20000
   80000  -20000
   80000  -60000
   40000  -60000




MTRS

Number of MTRs = 2


Name = VR-001

Number of Points = 5

#                   Left  Right    Width               Floor
#    X       Y     Width  Width  Variation   Floor   Variation
#   (ft)    (ft)    (ft)   (ft)             (ft AGL)
#--------------------------------------------------------------
   60000   80000   10000  18000   Constant    2000    Constant
   80000   40000    5000  10000   Constant    3000      Linear
   20000   20000   10000  10000     Linear    1500      Linear
  -20000  -80000    8000  10000     Linear    1500    Constant
  -80000   20000   20000  20000   Constant    2000    Constant
```

```
Name = VR-002

Number of Points = 2

#                  Left  Right    Width              Floor
#    X       Y     Width  Width  Variation   Floor   Variation
#   (ft)    (ft)    (ft)   (ft)             (ft AGL)
#--------------------------------------------------------------
 -100000  -50000   10000  10000   Constant    5000    Constant
   40000  -50000   10000  10000   Constant    5000    Constant




MISSIONS

Number of Missions = 1


Name = Mission 1

Flight Noise Profile ID  = FM0200100
Aircraft Name            = C-17
Engine Name              = F117-PW-100
Operational Power Code   = 33
Operational Power Name   = FLIGHT IDLE
Noise Interpolation Code = VARIABLE
Engine Power Setting     = 70
Power Setting Units      = % NC
Airspeed (knots)         = 200
Standoff Distance (ft)   = 5000

Number of Height Pairs = 2

#   Minimum     Maximum
#    Height      Height  Utilization
#   (ft AGL)    (ft AGL)      %
#------------------------------------
      1000        1500       75
      1000        5000       25




MOA OPERATION SETS

Number of MOA Operation Sets = 1
```

```
Name = Training 1

Number of MOAs = 2

#   MOA Name   Utilization
#                  (%)
#--------------------------
  "Bulldog A"      50
  "Bulldog B"      50

Number of Missions = 1

#                     Annual Ops
#   Mission     -------------------     Air
#     Name      Day  Evening  Night    Time
#                                      (min)
#--------------------------------------------
  "Mission 1"   360      0     2.3       30



MTR OPERATION SETS

Number of MTR Operation Sets = 1


Name = Training 1

Number of MTRs = 2

# MTR Name   Utilization
#                (%)
#------------------------
  "VR-002"       75
  "VR-001"       25

Number of Missions = 1

#                     Annual Ops
#   Mission     -------------------
#     Name      Day  Evening  Night
#-----------------------------------
  "Mission 1"   100      0    34.5



FLIGHTNOISE
```

```
                 FLIGHT AIRCRAFT ID: FM02001
               FLIGHT AIRCRAFT NAME: C-17
                        ENGINE NAME: F117-PW-100
                  NUMBER OF ENGINES: 4
 MEASURED FLIGHT NOISE DATA UPDATED: 27 JUN 1996
        SOURCE OF FLIGHT NOISE DATA: U.S.A.F.
 NUMBER OF POWER SETTINGS REQUESTED:  1

PROFILE ID   INTERPOLATION TYPE    POWER SETTING    SPEED (KNOTS)   POWER
DESCRIPTION

FM0200100         VARIABLE          70.00 % NC           160        FLIGHT
IDLE POWER

 Distance    SEL (dB)      EPNL (EPNdB)      ALM (dBA)      PNLT (PNdB)
   (ft)     A-G    G-G      A-G    G-G       A-G    G-G       A-G    G-G
     200    101.1  101.1    105.8  105.7     97.9   97.9    111.5  111.4
     250     99.4   99.4    104.1  104.0     95.7   95.7    109.3  109.2
     315     97.8   97.2    102.3  101.5     93.5   92.9    106.9  106.0
     400     96.1   95.1    100.5   98.8     91.2   90.1    104.5  102.7
     500     94.3   92.6     98.6   96.2     88.8   87.1    102.0   99.6
     630     92.5   90.0     96.5   93.5     86.4   83.9     99.3   96.3
     800     90.6   87.4     94.4   90.6     83.9   80.7     96.5   92.8
    1000     88.6   84.7     92.0   87.7     81.3   77.4     93.6   89.3
    1250     86.5   82.1     89.4   84.5     78.6   74.1     90.4   85.5
    1600     84.3   79.7     86.9   81.6     75.8   71.2     87.3   82.0
    2000     82.1   77.3     84.2   78.8     72.9   68.1     83.9   78.6
    2500     79.7   74.8     81.2   76.0     70.0   65.1     80.4   75.1
    3150     77.2   72.4     78.0   73.0     66.8   62.1     76.6   71.6
    4000     74.5   70.0     74.6   69.8     63.6   59.1     72.8   68.0
    5000     71.7   67.2     71.5   66.1     60.2   55.7     69.3   64.0
    6300     68.8   64.2     68.0   61.8     56.6   52.1     65.5   59.3
    8000     65.7   60.6     64.4   57.0     53.0   47.9     61.5   54.1
   10000     62.4   56.7     60.4   51.3     49.1   43.3     57.2   48.1
   12500     59.0   51.7     56.3   45.2     45.1   37.8     52.5   41.4
   16000     55.4   46.4     51.8   38.0     40.9   31.8     47.4   33.6
   20000     51.7   40.7     46.6   28.7     36.5   25.6     41.6   23.7
   25000     47.7   35.2     40.7   14.7     32.0   19.4     35.1    9.1
ENDNOISE
```

A.17 Installation and Operation Guide

This software is deisgned to be ran directly through BaseOps. However, at this point in the development schedule BaseOps has not been updated yet to accept this new software. In the interim, the following steps can be taken to allow the user to use this software through BaseOps. Many of these steps will impact BaseOps' ability to perform normal NoiseMap calculations, so care must be taken to reverse changes before NoiseMap is used.

Step 1 - Replace 'MRNMap Old-to-New Aircraft Code Table.txt' file.

Within the MRNMap subdirectoy (typically c:\Noisemap\MRNMap) there is a file called 'MRNMap Old-to-New Aircraft Code Table.txt'. Rename this file to something like 'MRNMap Old-to-New Aircraft Code Table.txt~', then copy the provided version of this file into the directory.

Step 2 - Replace 'Omega10.exe'

Within the NMap subdirectory (typically c:\Noisemap\NMap) is the file 'Omega10.exe'. Rename this file to something like 'Omega10.exe~', then move a copy of the new file into the subdirectory.

Step 3 - Replace 'Flight01.dat'

Within the NMap subdirectory (typically c:\Noisemap\NMap) is the file 'Flight01.dat'. Rename this file to something like 'Flight01.dat~', then move a copy of the new file into the subdirectory.

Step 4 - Copy 'Flight02.dat'

Copy the file 'Flight02.dat' into the NMap subdirectory.

You are now able to use BaseOps to generate MRNMap input files. Generate whatever case you want and select 'Run' from BaseOps and select ‘Perform Only The Following Steps’ and check only the steps ‘Create MRNMap Ins File’ and ‘Create MRNMAP INX file.’ If other baxes are slected, errors will occur. The generated INX file is the input file to run MRNMap. Using a command line, navigate to the subdirectory where MRNMap_3.0.exe is stored, then type 'MRNMap_3.0 "filename"', where 'filename' is the full path to the *.inx file. Since files coming out of BaseOps usually have extra spaces in them, make sure to put the filename within quotes.

It is important to remember that the Omega10.exe and the flight01.dat files have been modified to run MRNMap specifically. In order to revert back to a state where BaseOps can run NoiseMap these changes must be reverted.