NUMERICAL PROPAGATION OF AIRCRAFT EN-ROUTE NOISE

A Thesis in
Acoustics
by
Kieran Poulain

© 2011 Kieran Poulain

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

December 2011
The thesis of Kieran Poulain was reviewed and approved* by the following:

Victor W. Sparrow  
Interim Chair of the Graduate Program in Acoustics, Professor of Acoustics  
Thesis Advisor

Thomas B. Gabrielson  
Professor of Acoustics, Senior Scientist

Kenneth S. Brentner  
Professor of Aerospace Engineering

*Signatures are on file in the Graduate School.
Abstract

The impact of aircraft noise has traditionally been oriented towards airport community noise issues. However, the emerging concern of noise issues in quiet natural areas such as National Parks has lately shifted some of the research efforts towards the propagation of noise from high altitude aircraft (over 18,000 feet), technically defined as aircraft en route noise. Indeed, the future re-emerging of open-rotor propulsion systems which would reduce the fuel emissions by as much as 30%, could also potentially increase the emitted noise. Such a configuration calls for new methods in order to determine the transmission loss between source and receiver, compared with a traditional community noise issue. Additionally, any numerical prediction should include a realistic and altitude-dependent weather profile along such long propagation paths, which directly affects how wave fronts bend as well as the total atmospheric absorption. AERNOM, which stands for “Advanced En Route NOise Model”, is a ray tracing numerical prediction scheme that was designed to handle such a geometry in order to account for the atmospheric effects in predicting en route noise ground contours. The performance of AERNOM was further compared to FAA’s Integrated Noise Model, Penn State’s Hybrid Propagation Model, as well as Delta’s NORD 2000 for simple test cases representing a stable en route overflight path.
# Table of Contents

List of Figures
- x

List of Tables
- xv

List of Symbols
- xvi

Acknowledgments
- xix

## Chapter 1

### Introduction

1.1 Background and Motivation ........................................... 1
1.2 Characterization of en route noise experimental data ......... 2
  1.2.1 Challenges faced by en route noise data collection ......... 2
  1.2.2 Historical review of en route noise data .................... 4
    1.2.2.1 The FAA/NASA en route noise symposium ............. 4
    1.2.2.2 The FICAN report .................................... 5
    1.2.2.3 The BANOERAC project ............................... 6
1.3 Noise propagation numerical methods: state of the art and application to en route noise prediction .................. 8
  1.3.1 Requirements for the prediction of en route noise ........ 8
  1.3.2 Ray-tracing for en route noise propagation ............... 10
    1.3.2.1 Mathematical derivation of the ray-tracing equations in a stable medium ............................ 10
    1.3.2.2 Development of the ray-equations ...................... 12
    1.3.2.3 Integration of the ray-equations ...................... 14
    1.3.2.4 Benchmarking of the ray solution for en route geometry: stable medium case ....................... 15
1.3.2.5 Mathematical derivation of the ray-tracing equations in a moving medium .......................... 19
1.3.2.6 Benchmarking of the ray solution for en route geometry: moving medium case ...................... 21
1.3.2.7 Ray tracing high frequency approximation for en route aircraft noise ............................... 22
1.4 Overview of A.E.R.N.O.M. .............................................. 23

Chapter 2

Ray-tracing transmission loss calculations 25
2.1 Noise Metrics and Outdoor Sound Propagation ..................... 25
  2.1.1 What is Sound? ............................................. 25
  2.1.2 Outdoor Sound Propagation ................................. 26
  2.1.3 Atmospheric refraction ..................................... 27
2.2 Outdoor Sound Loss Mechanisms in the Ray-Tracing Solution ... 28
  2.2.1 Geometrical Spreading and Refraction Losses ................. 28
  2.2.2 Lambancusa’s method: predict spreading losses along curved rays .................................. 30
  2.2.3 Atmospheric Absorption .................................... 32
      2.2.3.1 Absorption mechanisms in detail ....................... 32
      2.2.3.2 Comparison of two available algorithms for pure-tone frequencies ........................... 33
  2.2.3.3 Impact of humidity on atmospheric absorption losses 34
  2.2.3.4 Propagation of a 1/3 octave-band spectrum .......... 35
  2.2.3.5 Other attenuation mechanisms in the atmosphere . 38
  2.2.4 Excess Ground Attenuation .................................. 38
      2.2.4.1 Ground impedance models ....................... 38
      2.2.4.2 Plane-wave reflection coefficient .................. 39
      2.2.4.3 Excess Ground Attenuation formulation ......... 40
      2.2.4.4 EGA benchmarking for short-range propagation 41
      2.2.4.5 Adapting the ray-tracing integration for en route EGA predictions ................. 42
      2.2.4.6 Direct/reflected ray numerical construction .... 43
      2.2.4.7 EGA benchmarking for en route geometry ...... 43
  2.2.5 Shadow zone diffracted field .................................. 44
      2.2.5.1 The residue series solution ......................... 45
      2.2.5.2 Benchmarking of the residue series solution: theoretical approach ..................... 47
      2.2.5.3 Benchmarking of the residue series solution: experimental approach ............ 48
2.2.5.4 Extension of the residue series solution to en route geometry ........................................ 48
2.2.5.5 Other methods to predict acoustic shadow zone fields ........................................ 49
2.2.6 Turbulence effects in the ray-tracing context ......................................................... 50
2.2.6.1 Turbulence effects on EGA .................................................................................. 51
2.2.6.2 Scattering in the shadow zone due to turbulent scattering .................................. 52
2.2.7 Summary of transmission loss calculations ............................................................ 55

Chapter 3

The effects of meteorology on long-range vertical propagation ........................................ 56
3.1 NOAA weather balloon data ....................................................................................... 56
  3.1.1 The limitations of using weather balloon data ....................................................... 57
  3.1.2 NOAA data treatment ............................................................................................ 57
    3.1.2.1 First step: data collection ................................................................................ 58
    3.1.2.2 Second step: data smoothing and profile generation .................................... 58
  3.2 Example effects of meteorological variability on outdoor sound propagation ................... 59
    3.2.1 Effects of the atmospheric data scatter on the ground sound pressure level profile ................................................................. 59
    3.2.2 Ray-refraction seasonal variability ..................................................................... 59
    3.2.3 Atmospheric absorption seasonal variability .................................................... 60
      3.2.3.1 From dew point temperature to water vapor content .................................. 60
      3.2.3.2 The effects of humidity variability ............................................................. 61
    3.2.4 Geographical meteorological variability ............................................................ 61

Chapter 4

AERNOM to predict aircraft en route noise ground contours ........................................ 63
4.1 Stationary point source ground sound pressure level contours .................................. 63
  4.1.1 Pseudo-three-dimensional methodology to produce ground contours ......................... 64
  4.1.2 Limiting angle for upward refracting two-dimensional planes .................................. 65
  4.1.3 Example wind speed and temperature gradient effects on ground contours ................ 66
  4.1.4 En route noise ground contours .............................................................................. 68
4.2 En route overflight ground sound pressure level contours ........................................ 70
  4.2.1 From a point source to a line source representation ................................................ 70
  4.2.2 Seasonal variability of overflight ground noise level contours .................................. 71
  4.2.3 Overflight noise as a function of slant-range distance ............................................. 71
4.3 Moving flight effects in AERNOM ................................................................................. 73
4.3.1 Theoretical developments ........................................... 73
4.3.2 Moving flight effects in slant-range SEL decay .................. 75
4.3.3 Time series sound pressure level variation ....................... 75
4.4 General quantification of the impact of en route noise on the ground 76
  4.4.1 General trends for the propagation of en route noise ............. 76
  4.4.2 Extensive meteorological data treatment and slant-range level statistics .................. 77

Chapter 5

Software comparison for en route noise prediction 82

  5.1 Presentation of each model ........................................... 83
    5.1.1 Presentation of INM ........................................... 83
    5.1.2 Presentation of NORD 2000 ................................... 84
    5.1.3 Presentation of HPM .......................................... 85

  5.2 Software modification for long-range vertical propagation ........... 85
    5.2.1 Modifications in NORD 2000 .................................. 86
      5.2.1.1 Absorption losses ...................................... 86
      5.2.1.2 Excess Ground Attenuation [EGA] ......................... 87
      5.2.1.3 Refraction Losses [added module] ......................... 88
      5.2.1.4 Improving the linear sound speed approximation .......... 88
    5.2.2 Modifications in HPM .......................................... 89
      5.2.2.1 Meteorological data ..................................... 89
      5.2.2.2 Absorption losses ...................................... 90
      5.2.2.3 Propagation of a 1/3 Octave Band Spectrum .............. 90
      5.2.2.4 FFP method for en route overflight ...................... 90
      5.2.2.5 FFP grid spacing for en route noise ..................... 91
      5.2.2.6 Effects of density and wind in normal velocity boundary conditions (FFP) ............ 91
    5.2.2.7 Benchmarking the FFP for en route geometry and modification of FFT window ............. 92

  5.3 Software comparison setup ........................................... 92
      5.3.1 NPD curves retrofitting .................................... 92
      5.3.2 Noise output ............................................... 95
      5.3.3 General setup .............................................. 95

  5.4 AERNOM vs. NORD 2000 vs. HPM (FFP) ................................ 96
      5.4.1 Comparison of sound pressure level decay in downwind plane 96
      5.4.2 Comparison of loss mechanisms contribution in AERNOM, NORD 2000 “linear” and NORD 2000 “finer” ............... 97

  5.5 En Route Noise prediction comparison ................................ 98
      5.5.1 Overflight En Route Noise prediction using INM .............. 98
A.1 From the NOAA balloon data to the production of meteorological polynomial fits ........................................ 114  
A.1.1 Initial process ................................................. 114  
A.1.2 The computation of Weather polynomial fits ........... 119  
A.2 AEROM's code .................................................. 136

Appendix B
Additional codes .................................................. 168
B.1 The NORD 2000 “finer” code ..................................... 168
B.2 HPM (FFP only) for en route noise propagation ............. 184
B.3 The TSSR code .................................................. 203

Appendix C
Example of runs .................................................... 210
C.1 Running the NOAA atmospheric data code .................. 210
C.2 Running AEROM ................................................ 211
C.3 Running NORD 2000 .......................................... 212

Bibliography ....................................................... 213
List of Figures

1.1 Flight tracks recorded over Grand Canyon National Park on August 31st, 2003 .................................................. 2
1.2 1/3 octave band level time-histories of Metro III flyover .............. 4
1.3 Example of data collected using an ensemble-averaging technique ... 5
1.4 Estimated refraction effects on sound propagation upwind from a source at 10 km ..................................................... 5
1.5 $L_A$ vs. $L_{A_{1kHz}}$ metric for a typical en route overflight measured event in the time-domain ............................................. 7
1.6 Rays and wave fronts in the ray theory .................................. 11
1.7 Schematic of 2-dimensional ray geometry ................................ 14
1.8 Benchmarking of 2-dimensional ray tracing code in a stable atmosphere ................................................................. 17
1.9 Description of a ray path in a windy environment ...................... 19
1.10 Benchmarking of 2-dimensional ray tracing code in a moving atmosphere ................................................................. 22
1.11 Overview of the Advance En Route NOise Model (AERNOM) .... 24

2.1 Temperature effects on sound refraction ................................ 27
2.2 Wind effects on sound refraction .......................................... 27
2.3 Ray geometry for the prediction of spreading losses along curved and non-curved rays .................................................. 30
2.4 Refraction effects on spreading loss as a function of range .......... 31
2.5 Comparison of the 2-dimensional cumulative absorption losses (dB) distribution for two different absorption algorithms, in a stable yet refractive atmosphere from a point source located 10 km above the ground ........................................................................ 34
2.6 Seasonal averaged humidity profiles above Huntsville, AL ......... 35
2.7 Evaluation of the effects of humidity on the cumulative absorption losses (in dB) the 2-dimensional range-altitude plane, in a stable atmosphere from a point source located 10 km above the ground .. 35
2.8 Comparison of corrected versus non-corrected third-octave band cumulative absorption losses ........................................... 36
2.9 Consideration of 2 rays to compare the effect of two distinct absorption algorithms on the predicted ground noise level .......... 37
2.10 Sketch showing the ray-interaction for a receiver located above an impedant ground .......................................................... 38
2.11 Benchmarking of the Excess Ground Attenuation routine: transmission loss as a function of range for a frequency of 100 Hz over grassland ................................................................. 42
2.12 Benchmark: implementation of the Excess Ground Attenuation routine in AERNOM for a negative sound speed gradient .......... 44
2.13 Benchmark: implementation of the Excess Ground Attenuation routine in AERNOM for a positive sound speed gradient .......... 44
2.14 Sketch representing the limiting ray in an upward refracting atmosphere ................................................................. 45
2.15 Benchmark of the residue series solution for different radii of curvature, over a soft ground ............................................ 47
2.16 Benchmark of the residue series solution for different receiver ranges, over a soft ground ............................................. 47
2.17 Comparison of residue series implementation with experimental data over hard ground .................................................. 48
2.18 Comparison of residue series implementation with experimental data over soft ground .................................................. 49
2.19 Analogy between the propagation of sound in a refractive atmosphere over a flat terrain and the propagation of sound in a homogeneous atmosphere over a curved terrain ......................... 50
2.20 Comparison of available methods to compute the shadow zone transmission loss for Attenborough’s upward refracting benchmark .... 50
2.21 Illustration of the turbulence effects on the excess ground attenuation in a homogeneous atmosphere ........................................ 52
2.22 Sketch showing the setup for the TSSR method .................................................. 53
2.23 Sketch showing the setup for the Menounou edge diffraction calculation .......................................................... 54
2.24 Benchmarking and comparison of Maekawa and Menounou’s diffraction methods .................................................. 55

3.1 Winter 2010 wind speed profile above Pittsburgh, PA: from the original data to single-altitude data ........................................... 58
3.2 Winter 2010 temperature profile above Pittsburgh, PA: from the single-altitude data to smoothed data .................................. 58
3.3 Effects of atmospheric scatter on predicted ground sound pressure level profile .................................................. 59
3.4 2010 Seasonal atmospheric variability above Pittsburgh, PA. ........... 60
3.5 The effects of seasonal variability on ray refraction ......................... 60
3.6 Humidity profiles for winter and summer 2010 seasonal averages above Pittsburgh, PA. ........................................ 61
3.7 Seasonal variability for the cumulative absorption losses from a high altitude source to the ground above Pittsburgh, PA. for several pure-tone frequencies .................................................. 62
4.1 From 2-dimensional runs to pseudo 3-dimensional ground contours . 64
4.2 From receiver grids aligned from the radial runs to the Cartesian ground contour grid .................................................. 65
4.3 Benchmark: effects of temperature and wind speed gradients on transmission loss ground contours, for a positive temperature lapse rate .................................................. 67
4.4 Benchmark: effects of temperature and wind speed gradients on transmission loss ground contours, for a negative temperature lapse rate .................................................. 67
4.5 2010 seasonal noise level ground contours from a high altitude stationary point source ........................................ 69
4.6 Difference in 2010 seasonal averaged noise level ground contours from a high altitude stationary point source ....................... 69
4.7 Example of geographical noise level ground contour variability from a high altitude stationary point source ....................... 69
4.8 Difference in 2010 winter averaged noise level ground contours from a high altitude stationery point source at two different locations .. 70
4.9 2010 seasonal noise level ground contours from a high altitude over-flight at Pittsburgh, PA. ........................................ 72
4.10 Geometry from an overflight track ........................................... 72
4.11 SEL decay as a function of slant-range distance for various seasonal atmospheric conditions in 2010. ........................................ 72
4.12 SEL decay as a function of slant-range distance for various angles between the wind direction and the flight track, for the 2010 Pittsburgh summer average weather ........................................ 73
4.13 Impact of random changes of NOAA balloon data on SEL decay with slant-range distance for atmospheric data above Pittsburgh, PA 73
4.14 Moving source effects on the slant-range SEL decay as a function of aircraft Mach number ........................................ 75
4.15 Moving source effects on the slant-range SEL decay as a function of the flight track versus wind direction angle ........................................ 75
4.16 Source-receiver geometry for time series prediction. ......................... 76
4.17 Time series sound pressure level variation for head, cross and tail wind situations for an en route overflight ................................. 76
4.18 Jet and UDF spectrum shapes corresponding to the same overall sound power level. ......................................................... 78
4.19 NOAA weather stations scattered around the continental United States .......................................................... 78
4.20 $L_{A_{\text{max}}}$ decay with slant-range distance for **summer** average weather conditions in 2011 above 16 different locations within the continental United States, from a jet type and a UDF/ATP theoretical source spectrum .......................................................... 79
4.21 $L_{A_{\text{max}}}$ decay with slant-range distance for **winter** average weather conditions in 2011 above 16 different locations within the continental United States from a jet type and a UDF/ATP theoretical source spectrum .......................................................... 79
4.22 Logarithmic fits for 2011 seasonal averages over 16 different locations within the continental United States, assuming jet and rotor source signatures of the same unweighted overall sound power level. 80

5.1 Horizontal versus vertical propagation geometry for outdoor sound propagation .......................................................... 83
5.2 Division of the atmosphere into layers of constant thickness ................ 87
5.3 Impact of the refined semi-analytical method on the reproduction of nonlinear sound speed profiles ............................................. 88
5.4 Sketch showing the construction of semi-analytical rays using a layered atmosphere ......................................................... 89
5.5 Repartition of the noise propagated from an en route overflight using the Hybrid Propagation Model ........................................... 90
5.6 Issues with steep-angle propagation using the Fast Field Program for en route geometry .................................................... 93
5.7 B777-300 1/3 octave-band spectrum retrofitted from the NPD data 95
5.8 Averaged realistic weather profiles .............................................. 96
5.9 Sound Pressure Level ($dB_A$) decay in a downwind refracting 2-dimensional vertical plane for AERNOM, the two NORD 2000 models and HPM (FFP) .............................................. 96
5.10 Comparison of geometrical losses for AERNOM and both NORD 2000 models in a particular 2-dimensional vertical plane ............ 97
5.11 Comparison of atmospheric absorption losses for AERNOM and both NORD 2000 models in a particular 2-dimensional vertical plane
5.12 Comparison of the excess ground attenuation for AERNOM and both NORD 2000 models in a particular 2-dimensional vertical plane
5.13 SEL ground contour using INM
5.14 SEL ground contour using the NORD 2000 “linear fit” model
5.15 SEL ground contour using the NORD 2000 “finer fit” model
5.16 SEL ground contour using the FFP from HPM
5.17 SEL ground contour AERNOM
5.18 SEL decay with sideline distance with AERNOM, INM, NORD 2000 “linear”, NORD 2000 “finer” (2 layers) and the FFP from HPM
5.19 Comparison of the original with the refined B777-300 NPD data for a specific weather model
5.20 Impact of seasonal weather averages on the refined B777-300 NPD data
5.21 Comparison of the original with the refined NPD data for two distinct spectral classes and a specific weather model
5.22 Refined NPD data for 9 different aircraft associated with 5 distinct spectral classes and various power settings
5.23 Refined NPD data computed from the B777-300 NPD data for various atmospheric models
5.24 Effects of wind direction on ground contour shape

B.1 Example of input text file for use in HPM (FFP only)
### List of Tables

1.1 Pros and cons of available numerical methods for aircraft en route noise propagation ........................................... 9
1.2 Ray travelling times for benchmark purposes, using a 2-dimensional ray tracing method in a stable atmosphere .................. 18

2.1 Comparison of atmospheric absorption losses for two different algorithms ......................................................... 38
2.2 Typical values for the ground effective flow resistivity ........... 39

4.1 A-weighted ground level aircraft sound power level third octave-band spectrum ................................................. 68
4.2 Theoretical 8-blade UDF/ATP A-weighted spectrum ............. 78
4.3 Coefficients of the logarithmic fit providing $L_{A,\text{max}}$ values as a function of slant-range distance for seasonal weather average conditions taken from 16 different NOAA balloon stations .................. 80

5.1 Absorption losses as a function of range at 1 kHz, for various atmospheric layer widths using NORD 2000 .................. 87
5.2 NPD data for a B777-300 (departure settings) .................... 94
5.3 Example of retrofitted NPD data ..................................... 94
5.4 NPD refinement process: difference in cumulative absorption between the SAE and the ANSI standard ...................... 104
5.5 NPD refinement process: difference in overall sound pressure level between the original and the refined NPD data .......... 105
5.6 Spectral content for spectral classes 104 and 105, normalized at 70 dB at 1 kHz and corrected for SAE absorption losses at 305 meters from the source .................................................. 106
5.7 Refined NPD data for possible use in en route noise propagation .. 108

C.1 Polynomial coefficients for the 2010 winter atmospheric averaged data above Pittsburgh, Pa. ........................................ 211
List of Symbols

\( f \) Frequency of interest [Hz], p. 4
\( L_A \) Overall A-weighted sound pressure level [dB\(_A\)], p. 8
\( L_{A,1kHz} \) Overall A-weighted sound pressure level cut-off at 1 kHz [dB\(_A\)], p. 8
\( p \) Acoustic pressure [Pa], p. 12
\( \omega \) Radial frequency [rad/s], p. 12
\( c \) Thermodynamic sound speed [m/s], p. 12
\( i \) Imaginary number defined as \( \sqrt{-1} \), p. 12
\( \theta_{\text{launch}} \) Ray launching angle relative to the horizontal [deg], p. 15
\( c_0 \) Ground thermodynamic sound speed [m/s], p. 17
\( g \) Thermodynamic sound speed gradient [/s], p. 17
\( T \) Atmospheric temperature [K], p. 17
\( \vec{v}_{\text{ray}} \) Ray velocity vector, p. 20
\( \vec{v} \) Wind vector, p. 20
\( \vec{s} \) Slowness vector, p. 20
\( \Omega \) Ratio of the sound speed in a still medium and the sound speed in the corresponding moving medium, p. 20
\( c_{\text{eff}} \) Effective sound speed defined as \( c_{\text{eff}} = c + v \) [m/s], p. 22
\( SPL \) Sound pressure level [dB], p. 28
\( L_w \)  Sound power level [dB], p. 28
\( \rho \)  Air density \([\text{kg/m}^3]\), p. 30
\( \gamma \)  Separation distance for a pair of rays \([\text{m}]\), p. 32
\( d \)  Distance between source and receiver \([\text{m}]\), p. 32
\( \Delta \Phi \)  Ray separation angle at receiver location \([\text{radians}]\), p. 32
\( a_{\text{refraction}} \)  Loss factor due to atmospheric refraction, p. 33
\( Z_g \)  Ground complex impedance \([\text{rayls}]\), p. 42
\( \sigma \)  Ground effective flow resistivity \([\text{cgs rayls or kPa.s.m}^{-2}]\), p. 43
\( R_p \)  Plane wave reflection coefficient, p. 44
\( x_{H_2O} \)  Volume mixing ratio of water, p. 70
\( X_{H_2O} \)  Percentage of water vapor \([\%]\), p. 70
\( \phi \)  Angle between the 2-dimensional vertical plane and the wind direction \([\text{radians}]\), p. 74
\( \zeta \)  Shadow boundary angle \([\text{deg}]\), p. 77
\( \beta \)  Atmospheric lapse rate defined as the negative value of the temperature gradient \([\text{K/km}]\), p. 78
\( \tau \)  Overflight duration \([\text{s}]\), p. 83
\( \varphi \)  Angle between the en route overflight track and the wind direction \([\text{deg}]\), p. 84
\( f_s \)  Emitted frequency before Doppler shift \([\text{Hz}]\), p. 87
\( f_r \)  Received frequency after Doppler shift \([\text{Hz}]\), p. 87
\( V_{SG} \)  Aircraft speed relative to the ground \([\text{m/s}]\), p. 87
\( f_a \)  Frequency at which the atmospheric absorption loss is computed \([\text{Hz}]\), p. 88
\( M_c \)  Aircraft Mach number, p. 88
\( d_{\text{SL}} \)  Slant-range distance \([\text{km}]\), p. 94
$h_{\text{layer}}$  Atmospheric layer thickness [m], p. 102

$\Delta z$  Vertical grid spacing in the FFP method [m], p. 107

$\Delta r$  Horizontal grid spacing in the FFP method [m], p. 107
Acknowledgments

First and foremost, I want to thank my advisor Dr. Victor Sparrow as well as the graduate program in acoustics for having giving me the opportunity to experience innovative research in a passionate field. I feel very fortunate to have been a student of the Penn State graduate program in acoustics and am now proud to enter the circle of its alumni. Then, I am very grateful to the PARTNER Center of Excellence, the FAA and the Volpe National Transportation Systems Center for having supported my work. It has been a great pleasure to work with Dr. Bill He (FAA), M. Eric Boeker and M. Noah Schulz (VOLPE). I am furthermore grateful to M. Lou Sutherland for his vision regarding my first publication and to Dr. Birger Ploving of Delta, Inc. for his cooperation.

I have had the opportunity to learn from extraordinary mentors and from an international student’s perspective to experience an enriching mentor-student relationship. I want to personally thank my advisor Dr. Victor Sparrow for his guidance, trust and professionalism. I also want to thank Dr. Steven Garrett not only for his suggestions on the best east coast restaurants, but also for being so passionate and communicative with students.

Many thanks to Dr. Kenneth Brentner and Dr. Thomas Gabrielson for haven taking the time and energy to be on my committee.

I also want to thank the students that surrounded me during these two years in Research West: Alexandre (from the French contingent), Joyce and Joe for their help and insightful comments, as well as Whitney, Sang, Beomsoo, Denise, Kim, Amanda and Andy. It was a pleasure to work in the same office.

Lastly, I want to thank my wife Marie-Felicite for having been patient with me at some times, it is not always easy to live with a young researcher. I am very proud of us and look forward to living next to you for many more years.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of PARTNER sponsoring organizations.
Dedication

This work is dedicated to my parents, for their everlasting support since my education started. Dedication to my mother, for her patience and hard work in raising me in the best condition she could. Dedication to my father, for having shown me that anything is possible provided one has a goal in life.
Chapter 1

Introduction

1.1 Background and Motivation

The context of aircraft noise has traditionally been oriented towards airport community noise issues. This concern originates mainly from the continuous expansion of airports, as a result of economic growth, often facing local opposition for environmental reasons. The Federal Aviation Administration (F.A.A.) has been managing the development of numerical prediction tools such as the Integrated Noise Model (I.N.M. [1]) to account for annual average noise level ground contours around airports. Its goal is to implement a neutral decision-making process as far as protecting inhabitants. One of the limitations of I.N.M. is its inability to consider realistic weather profiles, for the purpose of limiting the computation time. More recently, advanced models such as the Hybrid Propagation Model (H.P.M. [2]) developed at Penn State include terrain, ground impedance discontinuities, weather variability and aircraft directivity. In the end, the modelling capabilities of aircraft community noise have reached a good level of comprehension, while the common tools such as I.N.M. are widespread among airport authorities. With the re-emerging of new propulsion systems which could help reduce the emissions by 30% while increasing the noise emissions, some of the research efforts in aircraft noise modelling have lately shifted to high altitude flyovers above 18,000 feet, which can be detected in natural areas where the background noise can be as low as 20 dB [3]. Figure (1.1) shows all the recorded day time en route overflight events on a peak day over the Grand Canyon National Park in 2003.
In 1987, President Reagan signed the National Parks Overflights Act of 1987 (Public Law 100-91), which set the requirement that at least half of the Grand Canyon surface area be quiet for at least 75% of the day. Following that initiative, a joint FAA/NASA symposium was held in 1989 [5], during which scientists and officials discussed various topics related to aircraft en route noise: experimental data, subjective test responses and numerical predictions. One of the main conclusions to take away from that colloquium is that the measurements emphasize large day-to-day variabilities, which the numerical prediction methods fail to fully explain. Since then, efforts have mainly been oriented towards aircraft en route noise data measurement in order to characterize and define the contribution from high altitude flyovers. With higher computational power nowadays it appears essential to re-think en route noise numerical prediction schemes, by evaluating the available software and by building a prediction code. Chapter 1 will focus on an overview of available experimental data that define aircraft en route noise, then the choice of an appropriate numerical method will be discussed, before elaborating into the details of the ray tracing technique applied to long-range vertical propagation.
1.2 Characterization of en route noise experimental data

A brief overview of available civilian jet aircraft en route noise data will be presented in this section, after having discussed some of the experimental challenges linked with the collection of such data.

1.2.1 Challenges faced by en route noise data collection

Collecting en route noise data is a difficult task. Several external factors make the measurement a highly technical effort. First, no specific noise metric exists to define en route noise yet, although a few attempts have been made [5]. The low-frequency noise component from new propulsion systems could potentially have a large impact given the community response to wind turbine noise. Recently [3], it was proposed to apply a low pass filter at 1 kHz for ground measurement data as higher frequencies are attenuated by the atmosphere and the noise insects produce often contaminate the measurement anyway at higher frequencies. Further, the atmosphere through which the noise propagates is highly unstable, resulting in fast changing wind speeds and boundary layer atmospheric turbulence. This partially explains the observed scatter in the collected data over time [3, 5]. Other possible explanations of this observed scatter are the presence of insects during measurements [3], thus leading to the recommendation of using two microphones per test site. For turboprops, one possible explanation of the observed phenomena is the emission of beat frequencies [6] due to the superposition of sound signatures originating from two noise sources radiating at slightly different frequencies ($\omega_1$ and $\omega_2$ corresponding to two different rotational speeds). According to reference [6], the time period of pressure fluctuations is calculated as:

$$t_s = \frac{2 \times \pi}{\Delta \omega} = \frac{2}{\frac{\Delta N}{60} \times BLN \times HN} \quad (1.1)$$

where $\Delta \omega = (\omega_1 - \omega_2)$, $\Delta N$ is the corresponding difference in rotational speed, $BLN$ is the common rotor blade number and $HN$ is the harmonic number. Equation (1.1) shows that as the harmonic number increases, the time between pressure
maxima and minima decreases, this being the basis of the observed scatter, once the band levels are recombined into an overall sound pressure level. Figure (1.2) provides the time variation of the fundamental and first harmonic frequencies (including Doppler frequency shifts) from a Metro III flyover. As observed, the first harmonic $f \approx 200$ Hz exhibits twice the beat frequency value as is observed for the fundamental frequency, thus proving that level fluctuations could potentially be a result of beats due to slightly different rotational speeds of both propellers.

1.2.2 Historical review of en route noise data

1.2.2.1 The FAA/NASA en route noise symposium

The FAA/NASA en route symposium is an extensive collection of collaborative work [5] that lies as the basis of en route noise noise literature. The main interesting conclusions have been summarized by He and Boeker in a recent paper [7]. A major experimental study consisted of a series of flight tests of the Gulfstream\textsuperscript{1} Propfan

\textsuperscript{1}www.gulfstream.com
Turbofan Assessment (PTA) mounted with a engine developed by NASA Langley Research Center. Acoustical and weather data were collected, and a chase plane was utilized to measure the en route noise emission of the PTA during the overflight portions over a microphone array. The tests were run at two different locations during different seasons, during which various flight Mach numbers were applied. While the chase plane indicated similar noise levels, the measured noise on the ground did vary up to 12 dB from one day to another, thus indicating that the observed scatter (see section (1.2.1)) is largely due to propagation mechanisms, in particular to atmospheric variability. A ray-tracing model was created to explain this observed variability; however, it could only explain a 7 dB variability, mainly due to atmospheric absorption (defined in Chapter 2), and was unable to predict the observed short-term variability. The remaining variation might be due to other factors already mentioned in section (1.2.1). A very complete report [8] provides additional details regarding the experimental methods and ray-tracing characteristics. Figure (1.3) shows the result of the microphone array ensemble-averaging, with the corresponding 80% confidence intervals, and gives an idea of the typical en route noise signature in the time domain. Another essential conclusion of this study has to do with the effects of high altitude atmospheric conditions on the source characteristics. Indeed, the ray tracing predictions had to be scaled about 10 dB down to the measured data in order to find an acceptable agreement for a 35,000 feet overflight, thus enhancing probable effects of low density and low temperature on the noise source itself. The high altitude characterization of
aircraft noise emissions and aerodynamics is still an ongoing research topic.

One other essential analysis was reported by Sutherland and Wesler [9], with an emphasis on the influence of atmospheric parameters on the propagation of en route noise: the atmosphere is known to cut the high frequencies from a high altitude source (which is why it is often referenced as a low-pass filter), and the temperature and wind speed gradients act on the refraction of sound, this effect being directly proportional to the gradient strengths as shown in Figure (1.4). Consequently, only a portion of the noise emitted effectively reaches the ground in certain directions.

1.2.2.2 The FICAN report

In 2003 and 2004, the Federal Interagency Committee on Aircraft Noise (F.I.C.A.N.) worked conjointly with the FAA and the National Park Service (NPS) to conduct an elaborate study on the status of aircraft en route noise in the Grand Canyon National Park (GCNP) [4]. Volpe\(^2\) and Wyle\(^3\) joined to compare the prediction capabilities of FAA/Volpe’s Integrated Noise Model (INM, [1]) and Wyle’s

\(^2\)http://www.volpe.dot.gov/
\(^3\)www.wylelabs.com
NoiseMap Simulation Model (NMSim, based on the ray tracing method), with field data collected during the summer of 2004 in GCNP. Both models predict similar noise levels within a few dB, with a tendency for INM to over predict the peak levels and audibility. Also, in an effort to translate the ground aircraft sound power level to higher altitudes, attachments 1 and 2 of reference [4] provide some insightful details on how to correct the jet component of the total aircraft noise from a low-altitude to a high-altitude flyover, depending on the en route altitude atmospheric characteristics. This method might not be suited to en route noise settings as some believe that shock-cell noise is the primary noise source in such a situation [10]. Furthermore, an important observation from this field test is that a higher overflight altitude leads to higher noise levels on the ground, which was thought contrary to expectations (since the larger the travelled distance the larger the geometrical and absorption losses), but this observation is consistent with other field tests [11].

1.2.2.3 The BANOERAC project

As mentioned in a recent report [12], the BANOERAC project [3] has been initiated by the European Safety Agency in 2009, and has two major goals: the first one being the creation of a background noise map for most of Europe (mostly based on population density data), and a few data collection campaigns for some selected sites in order to assess the numerical prediction tools. The second focus lied in modern aircraft en route noise measurement, overlapping a 6-month time period which included both the summer and winter seasons, for a variety of aircraft (propeller, jet, military). The state-of-the-art recording equipment and the real-time aircraft tracking system make the collected data probably the most recent and most reliable in the literature to date. Unfortunately, no weather balloon data were acquired so that the atmospheric structure is not precisely known. Figure (1.5) shows an en route measured event as a function of time, for two different metrics. The green and black curves represent the unfiltered and the low-pass filtered \( f \leq 1 \text{ kHz} \) noise variation respectively, in dB\(_A\). As discussed in section (1.2.1), filtering the measured noise cuts some of the insect background noise, and consequently leads to a sharper detection of en route overflight events. Results en-

\(^4\)BAckground NOise level and noise levels from En Route airCraft
hance very little differences in the noise levels between different classes of turbofan aircraft, while turboprop aircraft were usually 10 dB_A louder than the turbofans. Compared with previous field data [11], the levels were substantially lower since these were taken at a time before high-bypass ratio turbofans came into service. A scatter of 5 to 7 dB, typical of en route noise data, was observed during these field tests. Lastly, the study says that I.N.M. [1] substantially under predicts en route noise, thus opening the discussion to the ongoing research on high altitude aircraft noise signature (see also [4]).

Section (1.2) has provided some insightful information regarding the experimental characteristics of en route noise overflight events across the years, which brings us to the following summary:

- Aircraft en route noise is clearly audible in National Parks.
- Turboprop aircraft are louder than turbofan aircraft.
- The atmosphere plays an essential role on the propagation of en route noise.
- There is a large scatter in measured levels possibly due to atmospheric
variability, measurement conditions and/or beat frequencies for rotor-type propulsion systems.

The following section will present the various numerical methods generally used for outdoor sound propagation purposes, with an emphasis on en route noise, which is the main goal of this thesis.

1.3 Noise propagation numerical methods: state of the art and application to en route noise prediction

1.3.1 Requirements for the prediction of en route noise

With growing computational capabilities, the numerical prediction of noise propagation has been of soaring interest since the 1980’s; it helps reduce the cost of field measurements, and has become a decision-maker in the industry as well as amongst communities. Nevertheless, each numerical method comes with a set of assumptions which make it applicable to a particular type of problem only. Also, no numerical method can fully and realistically model common futures such as the ground or the atmosphere. In the end, the goal of a numerical prediction scheme is to compute noise levels that are the closest possible to those measured or expected. The following requirements help defining an appropriate numerical method in the context of aircraft en route noise:

- The propagation distances involved are large ($d \geq 10$ km).
- The atmospheric refraction (by wind speed and temperature gradients) calls for a directional source representation.
- An altitude-dependent weather must be taken into account as an input.
- A frequency-independent method would be preferred, in order to efficiently propagate a 1/3 octave-band spectrum (which stands as the standard for aircraft noise certification).
Four main numerical methods are available for underwater and outdoor sound propagation: the Fast Field Program (FFP), the Parabolic Equation (PE), the Boundary Element Method (BEM) and the ray-tracing method. The first two are wave-based models as they are based on the full wave equation solution and consequently propagate the noise with a frequency-dependent computation time (for these models, the computation time is directly proportional to the frequency). The BEM is generally applied in fluid/structure interaction problems and is not efficient for long-range propagation. The ray-tracing formulation is based on a high-frequency approximation of the initial wave equation. Its computation time is therefore independent of the frequency of interest and it is therefore suitable for long-range propagation. Table (1.1) references the pros and cons of the FFP, the PE and the ray-tracing methods, commonly used as tools to predict outdoor sound propagation, in the context of aircraft en route noise propagation. More details regarding these methods are presented in Chapter 5. The ray-tracing method is the most appropriate to predict en route noise. In fact, many references discuss the application of ray-tracing for aircraft noise [8, 9, 13, 14, 15, 16] and section (1.3.2) will provide some details into the construction and benchmarking of a ray-tracing numerical code to predict long-range vertical noise propagation, such as the type of noise propagated by an en route aircraft overflight.

1.3.2 Ray-tracing for en route noise propagation

The ray-tracing technique has extensively been studied at Penn State [17], used in underwater acoustic propagation [18] and outdoor sound propagation as mentioned in section (1.3.1) and its foundations are well established and limits of application completely framed. Few references, however, mention the construction of a ray-tracing code to predict en route aircraft noise, most of them being directly related to the FAA/NASA symposium, presented in section (1.2.2.1). The propagation distances involved are much greater than those accounted for in a typical community noise issue, which sets a goal to limit the computation time. Furthermore, the major noise contribution propagates in a vertical direction towards the ground, thus the necessity to propagate the noise through an altitude-dependent atmosphere. Section (1.3.2.1) will describe the ray equations before section (1.3.2.3).
Table 1.1: Pros and cons of the FFP, the PE and the ray-tracing methods applied to aircraft en route noise propagation.

<table>
<thead>
<tr>
<th>Numerical method</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
</table>
| **FFP** | • No frequency limitation  
• Fast run (get all range-dependent outputs for given receiver height)  
• Can include atmospheric turbulence | • Slow at mid to high frequencies \( f \geq 250 \text{ Hz} \)  
• Unable to predict vertical propagation  
• Unable to take range-dependent weather and terrain into account  
• Only propagates omnidirectional point sources |
| **PE** | • No frequency limitation  
• Can include range-dependent weather and terrain  
• Can include atmospheric turbulence | • Very slow at mid to high frequencies \( f \geq 250 \text{ Hz} \)  
• Angle-limitation: inaccuracy of the predictions at angles larger than 35° with respect to the vertical  
• Only propagates omnidirectional point sources (one exception: see PhD dissertation of Rosenbaum [2]) |
| **Ray-tracing** | • Fast solution (CPU time determined by size-step)  
• Can propagate directional sources | • High frequency approximation  
• Inability to fully predict shadow-zone levels  
• Partial and computationally expensive methods for turbulence  
• Often need to launch many rays |

will discuss the different methods to integrate these equations, leading to the construction of ray paths. Finally, the ray solution will be benchmarked for an en route geometry in section (1.3.2.4).

1.3.2.1 Mathematical derivation of the ray-tracing equations in a stable medium

The goal of this section is to introduce the ray equations from the full wave-solution that governs the propagation of acoustic waves, in a stationary medium (no wind). The method and equations are very well presented in [18]. Section (1.3.2.5) presents the equations corresponding to a realistic, windy environnement.
Let us start with the Helmholtz equation expressed in Cartesian coordinates in which \( \mathbf{x} = (x, y, z) \):

\[
\nabla^2 p + \frac{\omega^2}{c^2(x)} p = -\partial(\mathbf{x} - \mathbf{x}_s)
\]

(1.2)

where the source is located at position \( \mathbf{x}_s \). In order to obtain the ray equations, one seeks a solution to (1.2) in the form of a ray series (1.3).

\[
p(x) = e^{i\omega \tau(x)} \sum_{j=0}^{+\infty} \frac{A_j(x)}{(i\omega)^j}
\]

(1.3)

where \( A_j \) and \( \tau \) are functions to be determined. Now, taking successive derivatives of (1.3) with respect to the \( x \) coordinate:

\[
\frac{dp}{dx} = e^{i\omega \tau} \left[ i\omega \frac{d\tau}{dx} \sum_{j=0}^{+\infty} \frac{A_j}{(i\omega)^j} + \sum_{j=0}^{+\infty} \frac{dA_j}{(i\omega)^j} \right]
\]

\[
\frac{d^2p}{dx^2} = e^{i\omega \tau} \left\{ -\omega^2 \left[ \frac{d\tau}{dx} \right]^2 + i\omega \frac{d^2\tau}{dx^2} \sum_{j=0}^{+\infty} \frac{A_j}{(i\omega)^j} + 2i\omega \frac{d\tau}{dx} \sum_{j=0}^{+\infty} \frac{dA_j}{(i\omega)^j} + \sum_{j=0}^{+\infty} \frac{d^2A_j}{(i\omega)^j} \right\}
\]

(1.4)

Applying (1.4) to the \( y \) and \( z \) coordinates and recombining the equations leads to Equation (1.5) expressed in 3-dimensional form.

\[
\nabla^2 p = e^{i\omega \tau} \left\{ -\omega^2 |\nabla \tau|^2 + i\omega \nabla^2 \tau \sum_{j=0}^{+\infty} \frac{A_j}{(i\omega)^j} + 2i\omega \nabla \tau \sum_{j=0}^{+\infty} \frac{\nabla A_j}{(i\omega)^j} + \sum_{j=0}^{+\infty} \nabla^2 A_j \right\}
\]

(1.5)

Equation (1.5) is substituted in (1.2), then terms of equal order in \( \omega \) are equated to obtain the following infinite sequence of equations for the functions \( \tau(x) \) and \( A_j(x) \) in descending order of power:

\[
O(\omega^2) : |\nabla \tau|^2 = c^{-2}(x)
\]

\[
O(\omega) : 2\nabla \tau \cdot \nabla A_0 + (\nabla^2 \tau) A_0 = 0
\]

(1.6)

\[
O(\omega^{1-j}) : 2\nabla \tau \cdot \nabla A_j + (\nabla^2 \tau) A_j = -\nabla^2 A_{j-1}, \quad j = 1, 2, ...
\]
The first equation of (1.6) for $\tau(\mathbf{x})$ is called the Eikonal equation, while the remaining equations are referenced as the Transport equations. While the Eikonal equation provides information about the ray trajectory, the Transport equations govern the amplitude along the ray. The first two equations of (1.6) can be integrated and solved quite easily. The standard assumption at this point is to discard all but the first term in the ray series, making the ray solution a high-frequency approximation (further details regarding this approximation in the context of en route noise will be given in section (1.3.2.7)). Section (1.3.2.3) provides some guidance as far as integrating the Eikonal equation. It should be noted that the Transport equations will not be integrated in the context of this thesis. Although the first Transport equation can be integrated to track the amplitude along each ray, another method was used instead and will be presented in Chapter 2.

1.3.2.2 Development of the ray-equations

Equation (1.7) is a first-order non-linear partial differential equation. By introducing a family of curves (commonly called rays) which are perpendicular to the level lines of $\tau(\mathbf{x})$, the Eikonal can be expressed in terms of ray coordinates leading to an easier mathematical integration. The wave fronts and corresponding rays are shown on Figure (1.6).

$$|\nabla \tau|^2 = \frac{1}{c^2(\mathbf{x})}$$  \hfill (1.7)
\( \nabla \tau \) is by definition perpendicular to the wave fronts, so that the ray trajectory \( \mathbf{x}(s) \) can be defined by Equation (1.8) defined such as \( \frac{d\mathbf{x}}{ds} = c \nabla \tau \)\\

\[
\frac{d\mathbf{x}}{ds} = c \nabla \tau \tag{1.8}
\]

Going back to the Eikonal (1.7), the right-side of Equation (1.8) squared is equal to unity, thus \( \left| \frac{d\mathbf{x}}{ds} \right| = 1 \), and therefore \( s \) is the arclength along the ray. Further mathematical manipulations lead to the expression of the Eikonal in a form only involving \( c(\mathbf{x}) \) from the differentiation of Equation (1.7) with respect to \( s \). This is shown in development (1.9).

\[
\frac{d}{ds} \left( \frac{1}{c} \frac{dx}{ds} \right) = \frac{d}{ds} \left( \frac{\partial \tau}{\partial x} \right) = \frac{\partial^2 \tau}{\partial x^2} \frac{\partial x}{\partial s} + \frac{\partial^2 \tau}{\partial x \partial y} \frac{\partial y}{\partial s} = c \left( \frac{\partial^2 \tau}{\partial x^2} \frac{\partial x}{\partial s} + \frac{\partial^2 \tau}{\partial x \partial y} \frac{\partial y}{\partial s} \right) = c \left( \frac{\partial \tau}{\partial x} \right)^2 + \left( \frac{\partial \tau}{\partial y} \right)^2
\]

\[
\frac{d}{ds} \left( \frac{1}{c} \right) = \frac{1}{c^2} \frac{\partial c}{\partial x} = -\frac{1}{c^2} \nabla c
\]

By expanding (1.9) to the \( y \) and \( z \) coordinates, the 3-dimensional representation of the ray trajectory is written as:

\[
\frac{d}{ds} \left( \frac{1}{c} \frac{dx}{ds} \right) = -\frac{1}{c^2} \nabla c
\]

A 2-dimensional ray-tracing code was written in the context of this thesis. In such a situation, the ray equations are expressed in the range-altitude plane \((r, z)\) (see Figure 1.7), leading to the form shown on Equations (1.10a) and (1.10b), where two auxiliary variables \( \xi, \zeta \) are introduced to write the equations as a system of
four coupled first-order differential equations.

\[
\begin{align*}
\frac{dr}{ds} &= c\xi(s), & \frac{d\xi}{ds} &= -\frac{1}{c^2} \frac{dc}{dr}; \\
\frac{dz}{ds} &= c\zeta(s), & \frac{d\zeta}{ds} &= -\frac{1}{c^2} \frac{dc}{dz}.
\end{align*}
\]

Equations (1.10a) and (1.10b) are numerically integrated for a given set of initial conditions (see Equations (1.11a) and (1.11b)). As shown in Figure (1.7), these are the source position \((r_s, z_s)\) as well as the launching angle \(\theta_{\text{launch}}\). This angle is taken from the horizontal line of sight, so that a \(\theta = 90^\circ\) launching angle corresponds to the vertical ray.

\[
\begin{align*}
\frac{r}{r_s} &= 1, & \frac{\xi}{\xi_s} &= \frac{\cos \theta_{\text{launch}}}{c(r_s, z_s)}, \\
\frac{z}{z_s} &= 1, & \frac{\zeta}{\zeta_s} &= \frac{\sin \theta_{\text{launch}}}{c(r_s, z_s)}.
\end{align*}
\]

The notation \(\bar{y}' = \bar{f}(s, \bar{y})\) will be used in order to move forward with the integration of these equations where \(\bar{y} = \begin{pmatrix} r \\ z \\ \xi \\ \zeta \end{pmatrix}\).

### 1.3.2.3 Integration of the ray-equations

Several integration techniques are available to numerically construct the ray paths [19]:

![Figure 1.7: Schematic of 2-dimensional ray geometry [18].](image)
• Euler’s method (1st order, not recommended)

• Runge-Kutta methods (2nd and 4th order, more integration steps but more accurate)

After a brief presentation of these integration techniques, section (1.3.2.4) provides some initial results in the context of long-range propagation. Let us call \( h = s_{i+1} - s_i \) the spatial step size along the ray path, from step \( i \) to step \( i + 1 \). The Euler integration scheme is a one-step marching process:

\[
\vec{y}_{i+1} = \vec{y}(ih) + hf(s_i, \vec{y}_i)
\]

The 2nd order Runge-Kutta is also called the midpoint method, as the integration path is divided into a two-step process:

\[
\begin{align*}
\vec{y}_{i+1/2} &= \vec{y}_i + \frac{h}{2} \vec{f}(s_i, \vec{y}_i) \\
\vec{y}_{i+1} &= \vec{y}_i + h \vec{f}(s_{i+1/2}, \vec{y}_{i+1/2})
\end{align*}
\]

which can be rewritten as

\[
\begin{align*}
k_1 &= hf(s_i, \vec{y}_i) \\
k_2 &= hf(s_{i+1/2}, \vec{y}_i + \frac{1}{2}k_1) \\
\vec{y}_{i+1} &= \vec{y}_i + k_2
\end{align*}
\]

This step marks the end of the 2nd order Runge-Kutta integration method. From \( k_2 \), the method can be expanded to the 4th order Runge-Kutta integration method by adding the following steps:

\[
\begin{align*}
k_3 &= hf(s_{i+1/2}, \vec{y}_i + \frac{1}{2}k_2) \\
k_4 &= hf(s_{i+1}, \vec{y}_i + k_3) \\
\vec{y}_{i+1} &= \vec{y}_i + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}
\end{align*}
\]

The accuracy and applicability of these methods depend on the size step \( h \), on the launching angle \( \theta \) and more generally on the impact of the sound speed gradient on the ray paths refraction. Section (1.3.2.4) provides a discussion regarding the
choice of an appropriate combination of size-step and integration method, when applied to long-range vertical propagation in a stable atmosphere.

1.3.2.4 Benchmarking of the ray solution for en route geometry: stable medium case

Simplification of the ray equations in an altitude-dependent atmosphere

As illustrated in section (1.3.2.3), the sound speed variation along the ray path is a major governing factor of the ray paths construction. The sound speed gradient \( g \), defined as \( g = \frac{dc}{ds} \) is therefore a critical parameter. For the remaining part of this thesis, the weather will be assumed altitude-dependent only which is supported by the following facts:

- Most of the noise propagates in the vertical direction.
- The atmospheric altitude variations are generally larger than their range variations above flat and even ground.
- This assumption greatly simplifies the ray equations.

Under this reasonable approximation, \( g = \frac{dc}{dz} \), and Equations (1.10a) and (1.10b) are transformed to Equations (1.12a) and (1.12b).

\[
\frac{dr}{ds} = c\xi(s), \quad \frac{d\xi}{ds} = 0, \quad \text{(1.12a)}
\]

\[
\frac{dz}{ds} = c\zeta(s), \quad \frac{d\zeta}{ds} = -\frac{1}{c^2} \frac{dc}{dz}, \quad \text{(1.12b)}
\]

Pierce [20] demonstrates that if \( g \) is constant (so the sound speed is linear: \( c(z) = c_0 + g \times z \)), then the rays are circular arcs defined by:

- an arc center \( \Gamma \) with a z coordinate \( z_\Gamma = \frac{c_0}{g} \)
- an arc radius \( R_c = \frac{c_0}{g \times \cos \theta} \)

Application to a realistic atmosphere

The U.S. 1976 standard atmosphere models the atmosphere temperature by a linear decrease with height up to an altitude of 11 km: \( \frac{dT}{dz} = -6.5 \text{ K/km} \). Given that
the sound speed is only a function of $T$ : $c \simeq 20 \times \sqrt{T}$, if the effects of humidity are neglected, $g \simeq 4 \text{ s}^{-1}$ for this particular standard atmosphere. This value of $g$ was considered to benchmark the ray solution for large propagation distances.

The goal of the following sections is to discuss the best computational time over accuracy ratio, by varying the Runge-Kutta integration methods and the size steps. Some comparisons will also be made with the Euler integration method, to emphasize its inapplicability to long-range propagation. First, the analytical solution is computed using the equation for a circle in the cylindrical coordinate system $(r, z)$: $(r - r_\Omega)^2 + (z - z_\Omega)^2 = R_c^2$. Given a set of altitudes $z_{\text{analytical}}$, the only unknown are the $r_{\text{analytical}}$ values which are computed using: $r_{\text{analytical}} = \sqrt{R_c^2 - (z_{\text{analytical}} - z_\Omega)^2}$. The corresponding coordinates $(r_{\text{analytical}}, z_{\text{analytical}})$ constitute the analytical ray trajectory. Then the numerical solution ray trajectory is plotted on top of the analytical solution (see Figure (1.8)); the coordinates of the last point along the trajectory and the total computation time are recorded, and an error is defined by $e = 100 \times \frac{|d_{\text{ray}} - d_{\text{analytical}}|}{d_{\text{analytical}}}$, where $d_{\text{ray}}$ and $d_{\text{analytical}}$ are the straight distances between the origin and the last point along the numerical ray and the analytical ray, respectively. These runs span all three integration methods presented in section (1.3.2.3), with step sizes which vary from 10 m to 10 km. Figure (1.8) shows that with a slight computation time increase, the 2nd order Runge-Kutta integration method predicts much more accurate ray trajectories than does the Euler integration method. Consequently, the Euler integration method is discarded for long-range propagation. Further tests involving both Runge-Kutta (RK) possibilities quickly revealed that the following configurations led to roughly the same, small error ($e \simeq 1.10^{-7}\%$):

- (1) 2nd order RK with $h=0.01 \text{ km}$ - CPU time $= 87 \text{ ms}$
- (2) 4th order RK with $h=1 \text{ km}$ - CPU time $= 2.1 \text{ ms}$

Using combination (2) would lead to a huge potential gain in computation time, given that many rays will be launched from the en route aircraft. This integration setting was consequently implemented in the en route ray tracing trajectory code. An initial step size $h=1 \text{ km}$ was considered, and an adaptive step size algorithm was introduced, as the ray altitude decreases in order to precisely treat the ray/ground interaction (this will further be discussed in Chapter 2).
Figure 1.8: Example of 2-dimensional ray benchmarking. On the left, 2\textsuperscript{nd} order Runge-Kutta integration with a $h=1$ km step size versus analytical solution. CPU time=1.97 ms. Error=0.01 %. On the right, Euler integration with a $h=1$ km step size versus analytical solution. CPU time=1.28 ms. Error=1.04 %. Although the CPU time is slight increased, the 2\textsuperscript{nd} order Runge-Kutta integration method leads to much more accurate ray trajectories for a given step size.

An alternate way of benchmarking the numerical code is by computing the travel time along the ray, and verifying the output against the analytical solution for a linear sound speed profile in a stable atmosphere [21]. If $\theta_0$ and $\theta$ are respectively the launching angle and the angle at a specific location along the ray, then the corresponding travel time is expressed in seconds as:

\[
T = \frac{1}{g} \ln \left| \frac{\tan(\theta/2)}{\tan(\theta_0/2)} \right|
\]

provided the angles are taken from the $Z$ axis. The propagation time along each incremental step is stored and the total travel time is computed using: $T_{\text{total}} = \sum_j T_j$. With an altitude-dependent sound speed, the integration step size plays an essential role in the accurate prediction of the ray travel time, this being critical to the generation of the ground effects, as will be discussed in Chapter 2. Consequently, the total travel time was computed using various step sizes for all three integration techniques, for a ray horizontally launched from the origin, and travelling 50 km upwards in a linear sound speed gradient as shown on Figure (1.8); the results are exposed in Table (1.2). Both Runge-Kutta integration methods provide accurate
Table 1.2: Comparison of travelling time for a ray travelling upwards from the origin \((0,0)\) with a \(\theta_{\text{launch}} = 0^\circ\) launching angle. The analytical solution provides a travel time of \(193.3\) s. For the Euler and \(2^{\text{nd}}\) order RK (RK2) methods, each segment travel time was stored at the end of the integration step. For the \(4^{\text{th}}\) order RK (RK4) method, the segment travel time was stored at the end of the \(3^{\text{rd}}\) integration step, which proved to be the best solution.

<table>
<thead>
<tr>
<th>Integration method</th>
<th>step size [km]</th>
<th>(T_{\text{output}}) [s]</th>
<th>CPU time [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euler</td>
<td>10</td>
<td>185.3</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>192.3</td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>193.2</td>
<td>4.4</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>193.3</td>
<td>45.7</td>
</tr>
<tr>
<td>RK2</td>
<td>10</td>
<td>192.7</td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>193.3</td>
<td>1.6</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>193.3</td>
<td>8.2</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>193.3</td>
<td>66.8</td>
</tr>
<tr>
<td>RK4</td>
<td>10</td>
<td>193.4</td>
<td>1.1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>193.3</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>193.3</td>
<td>14.0</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>193.3</td>
<td>168.5</td>
</tr>
</tbody>
</table>

Given the conclusions of previous comparisons, the best approach to trace rays for en route noise propagation is to use the \(4^{\text{th}}\) order Runge-Kutta integration scheme with a starting step size of \(h=1\) km.

1.3.2.5 Mathematical derivation of the ray-tracing equations in a moving medium

This section will present the ray-equations in a realistic, moving atmosphere. In a windy atmosphere, the ray velocity is the time derivative of the ray trajectory:

\[
\frac{d\vec{r}_{\text{ray}}}{dt} = \vec{v}(\vec{x}_p, t) + \vec{n}(\vec{c}_p, t)c(\vec{x}_p, t),
\]

where \(\vec{v}\) defines the wind vector and \(\vec{n}\) is the normal vector to the acoustic wave fronts, pointing in the ray direction, see Figure (1.9). The wind effects are taken into account by the introduction of the slowness vector \(\vec{s}(\vec{x}) = \nabla \tau(\vec{x})\) [20]. Pierce shows that \(\vec{n} = \frac{c\vec{s}}{\Omega}\) where \(\Omega = 1 - \vec{v} \cdot \vec{n} = \frac{c}{c + \vec{v} \cdot \vec{n}}\) is the ratio between the sound speed in a still medium and the sound speed in the
The Eikonal equation therefore becomes $s^2 = \frac{\Omega^2}{c^2}$ or $(\tau)^2 = \frac{\Omega^2}{c^2}$. Pierce also provides the ray equations in a moving medium, which are parametrized in time (see Equations (1.13a) and (1.13b)).

\[
\frac{d\vec{x}_p}{dt} = \frac{c^2\vec{s}}{\Omega} + \vec{v}
\]  \hspace{1cm} (1.13a)

\[
\frac{d\vec{s}}{dt} = -\frac{\Omega}{c} \nabla c - \vec{s} \times (\nabla \times \vec{v}) - (\vec{s} \cdot \nabla)\vec{v}
\]  \hspace{1cm} (1.13b)

**Application to a stratified atmosphere**

As mentioned in section (1.3.2.4), the atmosphere is modelled as a vertically stratified medium. Furthermore, the vertical wind component is generally neglected compared with its horizontal component: $v_z \ll v_x, v_y$ so that $c = c(z)$ and $\vec{v} = v_x(z)\vec{e}_x + v_y(z)\vec{e}_y$ where $\vec{e}_x$ and $\vec{e}_y$ are unit vectors in the $x$ and $y$ directions, respectively. Under these assumptions, the ray-equations are simplified for $j \in (x, y, z)$:

\[
\begin{cases}
\frac{\partial x_j}{\partial t} = \frac{c_j^2}{\Omega} + v_j \\
\frac{\partial s_j}{\partial t} = \frac{\Omega}{c} \frac{\partial c}{\partial x_j} - \sum_{i \in x, y, z} s_i \frac{\partial v_i}{\partial x_j}
\end{cases}
\]
Since $\frac{\partial c}{\partial x} = \frac{\partial c}{\partial y} = 0$ and $\frac{\partial v_j}{\partial x} = \frac{\partial v_j}{\partial y} = 0$, $\frac{\partial s_x}{\partial t} = \frac{\partial s_y}{\partial t} = 0$. The components of the slowness vector along the $X$ and $Y$ axis are constant along any given ray. This is used to predict the limiting angle, as discussed in Chapter 4. The $z$ component of the slowness vector is computed using the partial differential equations:

$$\frac{\partial s_z}{\partial t} = -\frac{\Omega}{c} \frac{\partial c}{\partial z} - s_x \frac{\partial v_x}{\partial z} - s_y \frac{\partial v_y}{\partial z}$$

or in cylindrical coordinates, parametrized in distance:

$$\frac{1}{v_{\text{ray}}} \frac{\partial s_z}{\partial h} = -\frac{\Omega}{c} \frac{\partial c}{\partial z} - s_r \frac{dv}{dz}$$

Finally, Equations (1.10a) and (1.10b) are transformed to Equations (1.14) and (1.15) to take the wind effects into account.

$$\frac{dr}{dh} = \frac{c_{\text{eff}} \times c^2 \times s_r}{\Omega} + v, \quad s_r = \text{constant} = \frac{\cos \theta_{\text{launch}}}{c(z_{\text{source}}) + v(z_{\text{source}}) \cos \theta_{\text{launch}}}$$

$$\frac{dz}{dh} = \frac{c_{\text{eff}} \times c^2 \times s_r}{\Omega}, \quad \frac{ds_z}{dh} = -\frac{\Omega \times c_{\text{eff}}}{c} \frac{dc}{dz} - s_r \times c_{\text{eff}} \frac{dv}{dz}$$

Equations (1.14) and (1.15) are integrated using the 4th order Runge-Kutta integration marching step, and the complete Matlab code is shown in Appendix A.

1.3.2.6 Benchmarking of the ray solution for en route geometry: moving medium case

Pierce [20] derives an analytical solution for the ray radii of curvature in the presence of wind under the following assumptions:

- The wind speed is much weaker than the sound speed ($|\vec{v}| \ll c$).
- Or the ray direction is parallel to the wind speed vector $\vec{v}$, considered horizontal.
- His analytical solution neglects the side drift caused by winds.
Under these strong approximations (especially for long-range propagation), it is possible to express the ray radii of curvature:

\[ R_{\text{curvature}} = \frac{c_0 + 2n_0 \cdot v_0 + \frac{v_0^2}{c_0}}{\left| \frac{dc}{dz} \times \cos \theta + \frac{dv}{dz} \right|} \approx \frac{c_0}{\left| \frac{dc}{dz} \times \cos \theta + \frac{dv}{dz} \right|} \]

where the subscript “(0)” denotes the ray origin altitude. A linear sound speed \( c(z) = c_0 - 4 \times z, \ z \ [\text{km}] \) and wind speed profile \( v(z) = z, \ z \ [\text{km}] \) were considered to construct a ray following the derivation presented in section (1.3.2.5). The same approach described in section (1.3.2.4) allows the evaluation of the integration method/step size combination to accurately predict ray trajectories in a moving atmosphere. The assumptions being extremely limiting for long-range propagation, a small ray-trajectory deviation is observed after a propagation distance of 50 km, see Figure (1.10), explained by the fact that as the ray bends upwards, the angle between the ray direction and the wind direction increases, thus progressively breaking the 2nd bullet point assumption. Overall the ray tracing numerical code is validated, both in a stable and moving medium. The 4th order Runge-Kutta integration process associated with a starting step size of 1 km was found to produce the best computation time over accuracy ratio. Chapter 2 will go over the ray tracing routine more in depth, and will present the methods to compute the transmission loss between source and receiver using the ray tracing solution, all this in the context of long-range vertical noise propagation.

1.3.2.7 Ray tracing high frequency approximation for en route aircraft noise

For long-range propagation, one of the main loss factors is the atmospheric absorption which cuts the higher frequencies. Consequently, the new unducted fan propulsion systems with larger low-frequency content could potentially be louder on the ground. Given that the ray-tracing method lies on a high-frequency approximation, it is essential to clarify what this assumption states, before moving to Chapter 2.

In the development of the ray tracing equations, the high-frequency approxi-
Figure 1.10: Example of 2-dimensional ray benchmarking in a moving medium. 4th order Runge-Kutta integration with a $h=1$ km step size versus analytical solution. CPU time=7.0 ms. Error=0.86% The same error is obtained with a 2nd order Runge-Kutta integration scheme and a step size of $h=10$ m, with an associated CPU time of 1.2 s.

Information essentially states that:

$$\left| \frac{\nabla^2 p}{p} \right| \ll \left( \frac{\omega}{c} \right)^2$$

so that

$$\omega \gg \frac{c}{L}$$

where $L$ is a typical atmospheric scale length. In other terms, the angular frequency should be much larger (typically three times larger) than the sound speed gradient. Chapter 3 will cover the atmospheric characteristics. Generally, the strongest atmospheric refraction occurs for gradients $g \simeq \pm 8$ s$^{-1}$. Consequently, the ray-tracing solution is valid for frequencies down to 25 Hz, which roughly corresponds to the human hearing threshold. This information sufficiently demonstrates that the ray tracing solution can accurately propagate low frequencies, thus completely
supporting its choice as the best numerical method to propagate aircraft en route noise.

1.4 Overview of A.E.R.N.O.M.

This thesis will primarily present an advanced ray-tracing model, AERNOM\textsuperscript{5}, suited for the prediction of aircraft en route noise and that was entirely developed and tested from its foundations. Figure (1.11) highlights the process involved with the computation of noise ground contours: AERNOM computes the Transmission Loss (TL) between a high-altitude source and receivers on the ground through an atmospheric model (Pressure $P$, Temperature $T$, Wind speed $W_{\text{speed}}$, Wind direction $W_{\text{direction}}$ and humidity $h$). As mentioned on Figure (1.11), AERNOM includes several modules (refraction, absorption, excess ground attenuation and shadow zone diffracted field calculation) which will be presented in Chapter 2. Chapter 3 will focus on the available weather profiles, via the N.O.A.A.\textsuperscript{6} database. Then, Chapter 4 will discuss the results of the combination of these data with AERNOM to produce noise ground contours, as well as the various prediction possibilities it offers. Chapter 5 will present a comparison of available software to predict aircraft en route noise ground contours, and Chapter 6 will provide some conclusions and open up the scope of this research to future work.

\begin{footnotesize}
\begin{itemize}
  \item \textsuperscript{5}Advanced En Route NOise Model
  \item \textsuperscript{6}National Oceanic and Atmospheric Administration
\end{itemize}
\end{footnotesize}
Figure 1.11: Overview of AERNOM: from a noise source model and weather characteristics, AERNOM computes noise ground contours from a high altitude aircraft.
Chapter 2

Ray-tracing transmission loss calculations

2.1 Noise Metrics and Outdoor Sound Propagation

Extensive publications and research has been conducted for short-range propagation issues, and for these types of situations, the outdoor sound propagation mechanisms are well understood [22]. However very few publications specifically tackle the geometry of long-range vertical propagation. This chapter will therefore focus on the methods used within the ray-tracing context to predict the noise attenuation from a high altitude source to a low altitude receiver, after a short description of some noise metrics and definitions is presented.

2.1.1 What is Sound?

Sound propagates outwards from a source, by a succession of vibrational air particle transmission through air outdoors. The motion of air particles about their equilibrium position engenders a local compression-rarefaction cycle. The sound pressure \( p(r, t) \) is the measure of this fluctuating disturbance, from which one de-
finishes the sound pressure level as:

\[
SPL = 20 \times \log_{10} \left( \frac{p_{\text{peak}}}{p_{\text{ref}}} \right)
\]

where \( p_{\text{peak}} \) and \( p_{\text{ref}} \) are respectively the peak pressure level and the reference pressure level in air (\( p_{\text{ref}} = 20\mu\text{Pa} \)). The dB scale was introduced to quantify noise levels because of its ability to treat the large dynamic range of the human ear.

However, the human ear reacts accordingly to the frequency content of signals, which is essential to mention in the context of en route noise propagation. Consequently, the A-weighting filter was introduced to account for the response of the human ear to noise at different frequencies: lower frequencies (\( f \leq 250 \text{ Hz} \)) are considerably reduced while high frequencies (\( 1 \text{ kHz} \leq f \leq 5 \text{ kHz} \)) are emphasized due to the high sensibility of the ear in this particular frequency range. Each 1/3 octave-band is filtered using the appropriate band correction \( L_A(f) = SPL(f) + \Delta L_A(f) \), then these corrected band levels are logarithmically added to produce an overall A-weighted sound pressure level.

### 2.1.2 Outdoor Sound Propagation

The propagation of sound through air is governed by various effects: geometrical spreading, refraction effects, atmospheric absorption, ground and turbulence effects. All these loss mechanisms link the source sound power level \( L_w = L_p(r = 1 \text{ m}) \) to the receiver sound pressure level by:

\[
L_{p,\text{receiver}} = L_w - \Delta L_{\text{geom}} - \Delta L_{\text{refraction}} - \Delta L_{\text{absorption}} - \Delta L_{\text{ground}} - \Delta L_{\text{other}}
\]

In the ray-tracing solution, unlike any wave-based solutions (such as the P.E. or the F.F.P.), these contributions are computed once the ray trajectories are determined. The weather plays an essential role in the determination of these loss contributions; temperature and wind speed gradients affect the ray front curvature thus governing the ray trajectories, and humidity has a major impact on the atmospheric absorption as introduced in section (2.2.3.3). Amongst the atmospheric effects on sound propagation, atmospheric refraction is the most important for en route noise propagation, as mentioned by Sutherland and Weisler [9]. These are
emphasized in section (2.1.3).

### 2.1.3 Atmospheric refraction

Sound does not travel as straight lines, it bends towards the regions of lower sound speed in a stable atmosphere. The speed of sound speed is directly related to the temperature by $c \simeq 20\sqrt{T}$; thus, sound bends towards regions of lower temperature, as shown in Figure (2.1). On the right of Figure (2.1), the situation is one of a downward refracting atmosphere, in which rays get concentrated as they reach a receiver, compared with an upward refracting atmosphere in which they spread out away from the source as shown on the left of Figure (2.1). Consequently, noise levels are often louder in a downward refracting atmosphere. Typical upward and downward refraction situations happen respectively during and after a warm and clear summer day. During such a day, the solar radiation warms the ground quicker than the air which consequently creates an upward refracting atmosphere. At night, the ground cools off quicker than the air, and the atmosphere then refracts the sound downward. This phenomenon, associated with lower background noise levels during night time explains why people living next to an airport increasingly notice the noise at night time. Upward and downward refraction both lead to different features: there is a point where rays do not reach the ground in an upward refracting atmosphere, precisely when the launching angle is larger than the limiting angle. The region beneath the limiting angle, see left of Figure (2.1), is called a shadow zone, in which ray acoustics fails to predict the sound level. More
details regarding the shadow zone sound field in the context of en route noise will be discussed in section (2.2.5). In a downward refracting atmosphere, a multiple [ray/ground] bounce phenomenon often occurs for horizontal propagation (defined by low altitude source and receivers). For a high altitude source however, it is reasonable to neglect the effects of multiple bouncing.

Wind also contributes to the refraction characteristics of a given atmosphere, as shown in Figure (2.2). In the direction facing the wind, sound is bent upwards (upwind refraction) as the result of the moving medium effects presented in section (1.3.2.5). Rays are bent downwards in the direction of the wind (downwind refraction), and straight propagation (no refraction) occurs when the sound propagates perpendicularly to the wind direction in a homogeneous atmosphere.

This section has summarized the effects of refraction, opening the discussion to the more specific situation of en route noise propagation. The following sections will provide a description and implementation methods for each of the loss mechanisms mentioned, and which are optimized for long-range vertical propagation.

## 2.2 Outdoor Sound Loss Mechanisms in the Ray-Tracing Solution

### 2.2.1 Geometrical Spreading and Refraction Losses

Sound spreads in space as it propagates outwards from a source, at a rate determined by its characteristics relative to the size of the sound wavelength. Assuming that the sound field is spherically symmetric with respect to the source, the peak intensity $I_{\text{peak}}$ and acoustic power $W_{\text{peak}}$ are related at a distance $r$ from the source by $I_{\text{peak}} = \frac{W_{\text{peak}}}{4\pi r^2}$. Since $I_{\text{peak}}$ is defined as $I_{\text{peak}} = \frac{p_{\text{peak}}^2}{\rho c}$, $p_{\text{peak}}^2 = \rho c \frac{W_{\text{peak}}}{4\pi r^2}$. The sound pressure level was defined in section (2.1.1) and is also expressed as:

$$L_p = 10 \times \log_{10} \left( \frac{p_{\text{peak}}^2}{p_{\text{ref}}^2} \right)$$
The sound power level is mathematically defined as:

\[ L_w = 10 \times \log_{10} \left( \frac{W_{\text{peak}}}{W_{\text{ref}}} \right) \]

where \( p_{\text{ref}}^2 \approx \rho c_{\text{ref}} \). Consequently,

\[ L_p = L_w - 10 \times \log_{10}(4\pi r^2) \]

which leads to Equation (2.1).

\[ \Delta L_{\text{geometrical}} = 10 \times \log_{10}(4\pi r^2) \quad (2.1) \]

Equation (2.1) represents the geometrical attenuation from a point source in 3-dimensional space, where \( r \) is the distance between source and receiver, expressed in meters. The shortest distance from a source located at an altitude of 10 km above the ground being 10 km, the smallest corresponding geometrical attenuation is equal to 91 dB. This value therefore corresponds to the minimum attenuation level value.

Equation (2.1) provides the spherical spreading in a homogeneous medium in which rays propagate as straight lines. In a realistic medium however, rays are refracted and this relation does not hold any more. Several solutions are possible: one is to solve the Transport Equation, as presented in section (1.3.2.1), which provides the amplitude variation (thus including spherical spreading) along any given ray; another way of handling geometrical losses along refracted rays is by considering the straight line spherical loss Equation (2.1) for curved rays and assuming the total travelled distance represents a straight line distance. The latter method is suitable for short-range propagation and is used in available ray-based models such as NORD 2000 [24]; nevertheless it neglects the additional effects of refraction presented in section (2.1.3) which become important for long-range propagation.

Lamancusa and Daroux [25] published an efficient implementation for ray-tracing applications, which provides the total geometrical loss along a curved ray. The introduction of this method is the purpose of the next section.
2.2.2 Lamancusa’s method: predict spreading losses along curved rays

This formulation is based on the conservation of wave action along a ray tube, as mentioned by Pierce [20], and physically represented by the Blokhintsev invariant:

$$\frac{p^2 |v_{ray}| A_{ray}}{\rho c^2 \Omega} = \text{constant}$$

where $p$, $|v_{ray}|$ and $A_{ray}$ are respectively the pressure, velocity magnitude and ray tube area associated with a given ray tube. $\Omega$ was introduced in section (1.3.2.5). By evaluating the separation distance along a wave front between two closely spaced rays, and comparing the value with equivalent geometry in a homogeneous medium, it is possible to compute the additional effects of refraction. Assuming 2-dimensional ray propagation (thus neglecting wind side drifts), ray tube areas of refracted $A(r)$ and non-refracted $A(r')$ rays are respectively:

$$A(r) = \gamma d \Delta \Phi$$
$$A(r') = \gamma'd' \Delta \Phi$$
as shown in Figure (2.3). The separation distances in the pair at any location are given by:

$$\gamma' = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

for the refracted ray and

$$\gamma = 2d \tan(\frac{\Delta\theta}{2}) \simeq d\Delta\theta$$

for the straight ray, $\Delta\theta$ being the infinitesimal launching angle increment between both launched rays. The ray tube area is computed in the direction of energy propagation as $A_{\text{ray}} = A(r') \cos(\theta - \theta_{\text{ray}})$, $\theta$ and $\theta_{\text{ray}}$ being the launching and ground receiving angle respectively in the context of this thesis since the main interest lies in the total spreading loss obtained on the ground from a high altitude source. The additional loss linked with refraction is finally calculated after substitution of the latter equation in Equation (2.2):

$$a_{\text{refraction}} = \frac{p^2(r)}{p^2(r')} = \frac{A_{\text{ray}} |v_{\text{ray}}| \rho c}{A(r) \Omega \rho' c'^2}$$

By considering straight line propagation in a homogeneous atmosphere with a density $\rho'$ and a sound speed $c'$ equal to the ground values $\rho$ and $c$, and assuming the wind speed near the ground is negligible, the definition of $a_{\text{refraction}}$ further simplifies to:

$$a_{\text{refraction}} = \frac{A_{\text{ray}}}{A(r)}$$

since $|v_{\text{ray}}| = c$ and $\Omega = 1$ close to the ground. The attenuation due to refraction is defined as $\Delta L_{\text{refraction}} = 10 \times \log_{10}(a_{\text{refraction}})$, and the total spreading loss in a refracting atmosphere is shown in Equation (2.3):

$$\Delta L_{\text{spreading}} = \Delta L_{\text{refraction}} + 10 \times \log_{10}(4\pi d^2)$$

(2.3)

where $d$ is the straight distance between source and receiver. Figure (2.4) illustrates the effects of refraction on the total spreading loss in the context of en route noise propagation, for a stationary point source and receivers located 10 km and 1.2 m respectively above the ground, for constant sound speed gradients. The refraction effects are negligible at short-ranges (up to 10 km on the ground) but become significant at larger ranges, especially in upward refracting atmospheres.
as the receiver approaches the cut-off location defined as the range reached by the limiting ray. Overall, the implementation of this method is critical to the appropriate evaluation of the total spreading losses in a realistic atmosphere for the geometry involved with en route noise propagation.

2.2.3 Atmospheric Absorption

Equation (2.3) represents the transmission loss between a source and a receiver in an unbounded, lossless, refracting medium. However in a realistic atmosphere the amplitude of the travelling wave decreases, due to viscous friction, heat conduction losses and vibrational internal molecular processes. The total associated loss is frequency and range-dependent, which leads to the updated transmission loss Equation (2.4)

\[
TL(f, r) = \Delta L_{\text{refraction}}(r) + 10 \times \log_{10}(4\pi d^2)(r) + \alpha(f) \times r
\]  

(2.4)
where \( d \) and \( r \) are respectively the straight distance and the distance along the curved ray between source and receiver, \( \alpha \) being the frequency-dependent absorption coefficient expressed in dB/m. The absorption losses being range-dependent, atmospheric absorption plays a key role for long-range sound propagation. Obtaining accurate absorption coefficient values is therefore critical.

### 2.2.3.1 Absorption mechanisms in detail

Atmospheric attenuation is divided into viscous and heat conduction losses (referred to as the classical absorption coefficient) and losses associated with internal molecular processes (rotational and/or vibrational relaxation). Details regarding the mechanisms involved in the attenuation of a travelling wave in a dissipative medium are provided in the literature ([26] with errata [27]). The combined classical plus rotational absorption coefficient \( \alpha_{cr} \) depends on atmospheric parameters such as viscosity \( \eta \), mean pressure \( P \), speed of sound \( c \) and on the frequency of the travelling wave \( f \), but is independent of the atmospheric composition at altitudes below 76 km, much greater than the 10 km of concern herein. The vibrational relaxation coefficient, \( \alpha_{vib} \), on the other hand, is highly dependent on the molecular composition of the atmosphere (which includes humidity) down to altitudes below 10 km, as well as on other atmospheric parameters and the vibrational relaxation frequencies. The total absorption coefficient \( \alpha = \alpha_{cr} + \alpha_{vib} \) therefore relies on altitude dependent atmospheric parameters (mean pressure \( P \), temperature \( T \), ratio of specific heats \( c_P/c_v \), viscosity \( \eta \)), an altitude dependent atmospheric composition \((X_{O_2}, X_{O_3}, X_{CO_2}, X_{N_2}, X_O, X_N \) and \( X_{H_2O} \) (humidity) expressed in \%), and the vibrational relaxation frequencies for \( O_2, O_3, CO_2, \) and \( N_2 \). Consequently, the complete description of atmospheric absorption emanates from a set of equations defining an absorption algorithm coupled with a model describing the atmosphere composition and weather vertical distributions.

### 2.2.3.2 Comparison of two available algorithms for pure-tone frequencies

The purpose of this section is to summarize the content of a recent publication [28] that compares two available atmospheric absorption algorithms in the context of en
route noise: the ANSI S1.26 algorithm [29] and a refined algorithm ([30] with errata [31]). The ANSI standard has been widely used and referenced in the outdoor sound propagation community for horizontal propagation. The refined algorithm is more recent, is meant to predict the absorption coefficient up to 160 km above the ground, and includes additional terms related to the vibrational relaxation of CO$_2$ and O$_3$. It appears essential to compare both these algorithms in the context of long-range vertical propagation, by initially considering a vertical ray from a point source located 10 km above the ground, associated with a single frequency (also referenced as a pure-tone frequency). Since the absorption coefficient is altitude-dependent, it is necessary to divide the atmosphere into layers, and consider the mean absorption coefficient $\alpha_{\text{mean}}$ within each layer to compute the total absorption loss:

$$\Delta L_{\text{absorption}}(f) = \sum_i \alpha_i(f) \times s_i = \sum_i \alpha_i(f)$$

provided the atmospheric layer thickness $s$ is constant and equal to 1 m. The method and results are detailed in [28], the main conclusion being that both algorithms predict different pure-tone attenuations (3 dB for $f=1$ kHz), the departure being larger for higher frequencies. Figure (2.5) shows the difference in the cumulative absorption loss for a pure frequency of $f=500$ Hz between the refined algorithm and the ANSI S1.26 standard. A total of thirteen rays are launched in a refractive atmosphere, using the numerical process introduced in Chapter 1, and the attenuation losses are cumulatively added as the ray travels from the high altitude source to the ground. As the travelled distance increases, the departure between the predictions from both algorithms gets larger.

This section introduced some of the observed differences between these two available definitions, without modifying the atmospheric properties. It turns out that humidity has a large impact on atmospheric absorption; this is addressed in the next section.

### 2.2.3.3 Impact of humidity on atmospheric absorption losses

The chemical composition of the troposphere (up to 11 km) generally remains uniform, except for the humidity content taken as the percentage of water vapor ($\%H_2O$), which varies from one season to another at a specific location. The effects
of humidity on the total attenuation were assessed [28], by comparing two seasonal humidity profiles, presented in Figure (2.6), for a specific absorption algorithm (here the ANSI S1.26 standard was used), leaving all the other atmospheric parameters unchanged. A 2-dimensional attenuation comparison plot was produced in order to evaluate the impact of humidity, using the same settings as the ones used to generate Figure (2.5). The result is shown on Figure (2.7); it is clear that the vertical structure of humidity content in the atmosphere has a huge impact on the attenuation due to absorption. Salomons [32] explains that water molecules (hence humidity) play the role of a catalyst in the internal molecular processes of excitation and relaxation after the passage of a sound wave. That is the reason...
why the absorption of sound is generally lower during summer seasons because of higher water vapor content in the atmosphere, thus the noise levels are generally louder on the ground during summer seasons. Overall, these results emphasize the importance of the atmospheric vertical structure on the reliable prediction of aircraft en route noise propagation towards the ground. This has been referenced in the literature [33].

2.2.3.4 Propagation of a 1/3 octave-band spectrum

The comparison of the two absorption algorithms for pure-tone frequencies was presented in section (2.2.3.2). However, the noise spectrum of an aircraft is constituted by many frequencies, and represented as a 1/3 octave-band spectrum. Several techniques are available to compute the attenuation of 1/3 octave-band frequencies:

- The exact method requires the knowledge of both the narrow-band characteristics of the sound source and the frequency response characteristics of the one-third octave-band filters used in the analysis. For these reasons, it is not applicable to certification.

- The approximate method does not require knowledge of the narrow-band characteristics of the sound source. It uses a fairly simplistic equation to
Figure 2.7: Evaluation of the effects of the seasonal variation of atmospheric humidity content for $f=500$ Hz at Huntsville, AL: 2-dimensional (in the range-altitude plane) absorption loss [dB] difference distribution ($\Delta L_{\text{absorption \ summer}} - \Delta L_{\text{absorption \ winter}}$) in a stable yet refractive atmosphere from a point source located 10 km above the ground.

approximate one-third-octave band-level attenuation values, based on the frequency response characteristics of a third-order Butterworth filter which stands as the standard for 1/3 octave band filters. The corrected attenuation is a theoretically founded, experimentally validated non-linear function of the pure-tone attenuation and the normalized filter bandwidth. It is easy to implement but the attenuation must be less than 50 dB or else the solution diverges compared to the exact method. This technique is well described in Annex E of the ANSI S1.26 standard [29].

- The Volpe method [34]: this method was developed to face the limitations of the approximate method. It is as simple to implement as the approximate
method but is more accurate (with respect to the exact method) and there are no restrictions as far as the value of attenuation, so it is more adapted to long-range propagation. Furthermore, it was originally developed for aircraft noise certification.

Figure (2.8) illustrates the effects of correcting the absorption losses, using the Volpe method. Overall, the correction attenuates the absorption losses especially at high frequencies, this difference being larger further away from the source. The predicted 1/3 octave-band levels are consequently higher on the ground after the correction is applied. The Volpe method was chosen as the best solution to compute the absorption loss from a high altitude aircraft. It was used to compare the outputs of both absorption algorithms for the attenuation of a ground-level aircraft spectrum provided by Salomons [32]. The same atmospheric model was taken from [30] and the total atmospheric attenuation along rays 1 (the vertical ray) and 10 were computed, see Figure (2.9), by using the following process:

- The absorption for each 1/3 octave-band frequency is computed along each ray, assuming it is a pure-tone frequency.
- A correction terms is applied to total absorption, per 1/3 octave-band fre-
Table 2.1: Attenuation differences between both noise spectrum absorption algorithms with the same atmospheric model (2nd line) and between two seasonal humidity profiles with the same ANSI S1.26 algorithm (3rd line). The effect of humidity is of greater consequence.

<table>
<thead>
<tr>
<th></th>
<th>Ray</th>
<th>1</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>ΔL_{algorithm} [dB]</td>
<td></td>
<td>0.6</td>
<td>0.8</td>
</tr>
<tr>
<td>ΔL_{humidity} [dB]</td>
<td></td>
<td>6.0</td>
<td>8.2</td>
</tr>
</tbody>
</table>

- The attenuated 1/3 octave-band levels are recombined using a logarithmic summation

\[
\Delta L_{\text{absorption spectrum}} = 10 \times \log_{10} \left( \sum_i 10^{\frac{\Delta L_{\text{abs}}(f_i)}{10}} \right)
\]

to produce an overall attenuation loss.

The output of both algorithms were compared, then the ANSI S1.26 standard was used with Huntsville winter and summer average humidity, as shown on Figure (2.6), to assess the impact of humidity on the propagation of a broadband spectrum. The results are presented in Table (2.1). Both algorithms predict roughly the same attenuation losses for a broadband spectrum, with differences less than 1 dB. On the other hand, alternate humidity profiles lead to large differences in the prediction for the same absorption algorithm.

Two available absorption algorithms were compared in the context of en route noise propagation. Both algorithms predict different pure-tone attenuation values, but similar octave-band attenuations. The impact of humidity was further assessed and was introduced as being critical and of greater importance compared with the choice of an algorithm. Given its simplicity, the ANSI S1.26 standard was implemented in AERNOM.

2.2.3.5 Other attenuation mechanisms in the atmosphere

There are other references to sound absorption in the atmosphere, due to fog [35] and dust [36]. However the contribution of these mechanisms is negligible for
The total attenuation was computed for both absorption algorithms, using the Volpe method along the vertical ray (ray 1) and ray 10, in a stable refractive atmosphere.

long-range propagation at audible frequencies.

2.2.4 Excess Ground Attenuation

In the previous sections, the propagation of sound was treated in an unbounded medium. In reality however, the ground plays an essential role in the received noise on the ground. Any receiver receives a direct ray that does not interact with the ground, and a reflected ray that bounces off the surface. The combination of these two rays is the source of an interference phenomenon which can be destructive or constructive for pure-tone frequencies. Figure (2.10) illustrates the combination of the direct and reflected ray at the receiver location.

2.2.4.1 Ground impedance models

The interaction effects are significantly affected by the properties of the ground surface, modelled by the complex impedance of the ground surface $Z_g$ defined as $Z_g = \frac{\hat{p}}{\hat{v}_n}$ where $\hat{p}$ and $\hat{v}_n$ are the complex pressure and normal component of the

Figure 2.9: The total attenuation was computed for both absorption algorithms, using the Volpe method along the vertical ray (ray 1) and ray 10, in a stable refractive atmosphere.
Figure 2.10: Two distinct rays reach the receiver: the direct ray (in red) and the reflected ray (in green). $\theta_{\text{graz}}$ is the reflection angle with respect to the horizontal.

<table>
<thead>
<tr>
<th>Surface type</th>
<th>$\sigma$ [cgs rayls]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dry snow</td>
<td>10-30</td>
</tr>
<tr>
<td>Forest cover</td>
<td>20-80</td>
</tr>
<tr>
<td>Grassland</td>
<td>150-300</td>
</tr>
<tr>
<td>Roadside dirt</td>
<td>300-800</td>
</tr>
<tr>
<td>Asphalt</td>
<td>$\sim$ 30000</td>
</tr>
</tbody>
</table>

Table 2.2: Typical values for $\sigma$, from [38].

Complex velocity amplitude respectively. In the local reacting ground approximation, $Z_g$ is independent of the ray incident angle [32]. Several ground impedance models exist in the literature. Delaney and Bazley [37] empirically determined the ground impedance of a variety of fibrous material, using the effective flow resistivity $\sigma$ (in cgs rayls where 1 cgs rayls $= 1000$ Pa.s.m$^{-2}$) as a single parameter, see Equation (2.5). This work was later extended to outdoor ground surfaces, providing a simple and efficient way of producing ground impedances. Table (2.2) provides some values for $\sigma$, typically encountered outdoors. A snow cover attenuates the sound compared to propagation over a hard surface, as a result of the flow resistivity decrease. Other ground impedance models have been since introduced; the more realistic four-parameter model developed by Attenborough requires the knowledge of the ground flow resistivity, grain and pore shape factor as well as tortuosity [38].

$$\frac{Z_g}{\rho c} = 1 + 9.08 \left(\frac{\sigma}{f}\right)^{0.75} + i11.9 \left(\frac{\sigma}{f}\right)^{0.73} \quad (2.5)$$
Given its simplicity, the ground impedance model implemented in AERNOM follows the Delany and Bazley formulation, see Equation (2.5).

### 2.2.4.2 Plane-wave reflection coefficient

The quantification of the ground effects originates from the reflection coefficient, which for plane-waves is defined as

\[
R_p = \frac{\hat{p}_{\text{reflected}}}{\hat{p}_{\text{incident}}} = \frac{Z_g \sin \theta_{\text{graz}} - 1}{Z_g \sin \theta_{\text{graz}} + 1}
\]

Sound propagates as plane waves in the far-field; it is therefore appropriate to use \(R_p\) in the ground interaction module for a high altitude source. A more complicated formulation handles the interaction of a spherical wave with an impedant surface, valid for short-range propagation. Detailed derivations are omitted for brevity, and they can be found in reference [32].

### 2.2.4.3 Excess Ground Attenuation formulation

The following section describes the mathematical derivation of the ground effects on the received noise level, also called Excess Ground Attenuation (EGA). This value can be positive or negative as a result of the constructive or destructive interferences respectively.

Let us consider the propagation from a spherical source in a homogeneous medium above a ground surface. The resulting pressure field at the receiver location is the combination of a direct and a reflected ray

\[
\hat{p}_{\text{tot}} = S e^{ikr_1/r_1} + R_p S e^{ikr_2/r_2}
\]

where \(r_1\) and \(r_2\) are the direct and reflected propagation distances respectively, \(S\) is the spherical source’s monopole amplitude [20], \(k\) is the complex wave-number (including atmospheric absorption) and \(R_p\) is the plane-wave reflection coefficient.
Further reductions lead to

\[
\hat{p}_{\text{tot}} = \hat{p}_{\text{freefield}} \times \left( 1 + R_p \frac{r_1}{r_2} e^{ikr_2 - ikr_1} \right) \\
= \hat{p}_{\text{freefield}} \times \left( 1 + R_p \frac{r_1}{r_2} e^{i\omega \Delta t} \right)
\]

where \( \Delta t \) is the difference in arrival time between the reflected ray and the direct ray, and \( \omega \) is the radial frequency [rad/s]. These equations lead to the definition of the Excess Ground Attenuation (2.6).

\[
EGA = 20 \times \log_{10} \left| \frac{\hat{p}_{\text{freefield}}}{\hat{p}_{\text{tot}}} \right| = 20 \times \log_{10} \left| 1 + R_p \frac{r_1}{r_2} e^{i\omega \Delta t} \right|
\]

Equation (2.6) is valid for pure-tone frequencies. Chessell [39] introduced a formula to compute the EGA for 1/3 octave-band frequencies that takes the frequency-averaging per 1/3 octave-band into account, provided that \( R_p = |R_p| e^{i\theta} \).

\[
EGA = 10 \times \log_{10} \left( 1 + \frac{|R_{p,i}|}{r'} + 2|R_{p,i}| \sin \left( \frac{\mu \Delta t}{f_i} \right) \cos \left( \frac{\eta \Delta t}{f_i} + \theta_i \right) \right)
\]

\[
\left\{ \begin{array}{l}
\mu = \frac{2\pi \Delta f_i}{2f_i} \\
\eta = 2\pi \sqrt{1 + \left( \frac{\Delta f_i}{2f_i} \right)^2}
\end{array} \right.
\]

\[
(2.7)
\]

where \( R_p \) is evaluated at each 1/3 octave-band frequency \( f_i \), \( \Delta f_i \) is the corresponding 1/3 octave-band width and \( r' = \frac{r_2}{r_1} \). Equation (2.7) will be used in the EGA routine of AERNOM. Its use requires the knowledge of the user-defined flow resistivity as well as the reflection angle and arrival time difference between the direct and reflected rays at a specific receiver. The latter two parameters are computed via the ray-tracing numerical solution presented in Chapter 1. The precision of the difference in arrival time is critical to the accurate determination of interferences, especially at high frequencies.
2.2.4.4 EGA benchmarking for short-range propagation

Before elaborating into methods adapted to long-range vertical propagation, the implementation of the EGA was benchmarked for short-range propagation, using one of Attenborough et al’s outdoor sound propagation benchmarks [40]. Results are shown on Figure (2.11), and provide different initial settings (launching angle, receiver separation) to produce Transmission Loss (TL) values defined by 

\[
TL(r) = -20 \log_{10} \left( \frac{p_{\text{tot}}(r)}{p_{\text{tot}}(r = 1m)} \right)
\]

which includes geometrical and absorption losses, as well as the ground effects. The launching angle increment appears to be critical to the accurate transcription of the ground effects for short-range propagation. Overall, the implementation is validated, and the next sections will discuss the particular case of an en route geometry.

2.2.4.5 Adapting the ray-tracing integration for en route EGA predictions

Chapter 1 introduced the best combination of integration method and step size to integrate the ray equations for en route noise. Section (2.2.4.3) highlights the importance of two parameters (\(\Delta t\) and \(\theta_{\text{grazing}}\)) in the determination of the EGA. The starting step size is set to be \(h=1\) km. Then, as the ray approaches the ground, a secant method is employed to accurately compute the ray trajectory. For altitudes less than 1 km, the step size is modified at each integration step, following the
process \( h_i = \frac{z_{\text{ray}_{i-1}}}{2} \). That means that for rays below an altitude of 1 km, the step size successively decreases to values that are small enough to accurately trace the ground reflection, and to capture the difference in travel time in a direct/reflected ray interaction, even at high frequencies. The ray is reflected when the ray reaches an altitude less than 1 \( \mu \)m. The reflection process is automatically accompanied by the storage of the grazing angle from geometrical principles:

\[
\theta_{\text{grazing}} = \tan^{-1}\left(\frac{|z_{\text{ground}} - z_{i-1}|}{r_{\text{ground}} - r_{i-1}}\right)
\]

\[
= \tan^{-1}\left(\frac{|z_{i-1}|}{r_{\text{ground}} - r_{i-1}}\right)
\]

where \( r_{\text{ground}}, r_{i-1} \) and \( z_{i-1} \) are respectively the range reached on the ground, the range and the ray altitude at the previous integration step.

Once the ray trajectory is accurately computed via the ray-tracing technique, its trajectory is interpolated in order to track its altitude every 100 meters, thus forming 100 meter width atmospheric chunks in which the mean sound speed is considered to calculate the travel time in each layer. These are added to compute the total travel time \( \tau_{\text{total}} = \sum \tau_i \). Great attention was put to properly compute the reflection angle and travel time in AERNOM. The next section will present the method used to numerically construct the direct/reflected ray interaction at a specific receiver height.

### 2.2.4.6 Direct/reflected ray numerical construction

This section provides some information on the method used in AERNOM to produce the interaction between the direct and reflected rays. The starting field in the ray-tracing method is determined by the ray launching angle \( \theta_{\text{launch}} \), instead of being the receiver range for wave-based models. Rays are traced with incremental launching angles and each ray reaches the ground at a different range. An interpolation process then leads to the sound level at a specific range, if needed. The EGA calculation routine implemented in AERNOM includes the following steps:

- A receiver height is specified as an input. For each launching angle, a ray is launched and its propagation is stopped once the ray altitude reaches the receiver height: this corresponds to the direct ray. The range at which the
propagation is stopped defines the position of the receiver where the EGA term has to be calculated.

- The critical step in the EGA computation process lies in the determination of the correct launching angle which defines the reflected ray that reaches the receiver specified by the previous step. Two rays are launched with different launching angles $\theta_1 = \theta_0 + \Delta\theta_1$ and $\theta_2 = \theta_0 + \Delta\theta_2$ where $\theta_0$ is the direct ray launching angle. After their reflection off the ground, their propagation is stopped at the receiver altitude, and the corresponding range is recorded and compared to the direct ray final range $r_0$, by respectively forming $\Delta r_1 = r_0 - r_1$ and $\Delta r_2 = r_0 - r_2$.

- From the associations $[\theta_1, \Delta r_1]$ and $[\theta_1, \Delta r_2]$, a spline interpolation provides the reflected ray launching angle $\theta_{\text{reflected}}$ defined by $\Delta r = 0$.

2.2.4.7 EGA benchmarking for en route geometry

The solution was benchmarked for en route geometry against the linear sound speed analytical solution presented in Chapter 1, for upward and downward refracting atmospheres characterized by $g = \mp 4 \text{ s}^{-1}$, over a hard ground. Under the linear sound speed approximation, rays are circular arcs and the reflection point is defined by a 3rd order polynomial equation (see [20],[41]). Figures (2.12) and (2.13) show the comparison between AERNOM’s output and the benchmark, for an upward and a downward refracting atmosphere respectively.

The launching angle increment in AERNOM was chosen such as $\Delta\theta_{\text{launch}} = 1^\circ$, which explains the fact that the interference dips are partially reproduced. It is possible to improve the negative interference dip prediction by decreasing $\Delta\theta_{\text{launch}}$, which is automatically associated with an increase in computation time. Nevertheless, the general shape of the EGA as a function of range is well represented for all the frequencies that were tested, which validates the implementation of the EGA routine in AERNOM. For realistic, turbulent atmospheres, the depth of the negative interference dips is greatly reduced, as discussed in section (2.2.6.1), and therefore the constraint on $\Delta\theta_{\text{launch}}$ is not as restrictive.
Figure 2.12: Benchmark of the EGA value computed with AERNOM (left) vs. the analytical solution (right) in a stable atmosphere characterized by a negative sound speed gradient $g=-4$ s$^{-1}$, for pure-tone frequencies from 125 Hz to 2 kHz. In such an upward refracting situation, rays turn upward.

2.2.5 Shadow zone diffracted field

At this point, AERNOM’s spreading and absorption losses, as well as the excess ground attenuation routines in a refractive non-turbulent atmosphere have been presented. These are sufficient to predict the full transmission loss in a downwind refracting atmosphere or within the illuminated zone of an upwind refracting atmosphere, defined as the region reached by rays launched at angles $\theta_{\text{launch}} \leq \theta_{\text{limit}}$, where the limiting angle $\theta_{\text{limit}}$ was introduced in section (2.1.3) as the launching angle associated with the last ray that reaches the ground at the cut-off range, see Figure (2.14). For receivers further away from this point, located in the so-called shadow zone, the geometrical acoustics solution fails to predict the received sound; the purpose of the next sections is to present some of the available methods to predict the sound diffracted inside the shadow zone in the context of en route noise propagation.

2.2.5.1 The residue series solution

Pierce [20] was the first to find a way to predict the field in the shadow zone over soft and hard boundaries, provided the limiting ray is a circular arc. A significant amount of acoustic energy penetrates in the shadow zone via the creeping wave,
Figure 2.13: Benchmark of the EGA value computed with AERNOM (left) vs. the analytical solution (right) in a stable atmosphere characterized by a positive sound speed gradient $g=4 \text{ s}^{-1}$, for pure-tone frequencies from 125 Hz to 2 kHz. In such a downward refracting situation, rays turn downward.

Figure 2.14: Sketch representing the limiting ray in an upward refracting atmosphere. Receivers on the left of the cut-off location are in the illuminated zone, while receivers on the right of the cut-off location are in the shadow zone. In the linear sound speed approximation, rays are arcs of circles, and the cut-off range is analytically determined. Figure originally printed in [42].

which propagates near the ground and sheds energy upwards. The creeping wave solution is expressed in terms of residue series and is valid under the linear sound speed approximation. It has been improved and extended to finite impedance boundaries by Berry and Daigle [43], as mentioned in an excellent tutorial on shadow zone prediction in the context of ray-tracing applications [44]. Others
have tried to come up with a solution for more realistic sound speed profiles without any success.

The derivation of the residue series solution starts with the inhomogeneous Helmholtz equation representing the sound field from a point source located above a locally reacting surface in a stratified atmosphere:

$$\nabla^2 p + \left( \frac{\omega^2}{c^2(z)} \right) \left( \frac{\partial^2 p}{\partial z^2} \right) = -4\pi S \delta(x) \delta(y) \delta(z - z_s)$$

where \( \omega \) is the angular frequency, \( S \) the spherical monopole amplitude and the source position is \((0, 0, z_s)\). The solution for \( p(r, z) \) in the range-altitude plane is expressed as a Hankel transform:

$$p(r, z) = -S \int_{-\infty}^{\infty} H_0^{(1)}(kr) P(z, k) dk$$

where \( H_0^{(1)} \) represents the Hankel function of the first kind and order 0. Berry and Daigle evaluate this integral by contour deformation, and end up with the residue theorem:

$$p(r, z) = \frac{\pi e^{i\pi/6}}{l} S \sum_n H_0^{(1)} k_n r \frac{\text{Ai}[b_n - \frac{1}{l} e^{2i\pi/3}] \text{Ai}[b_n - \frac{Z}{l} e^{2i\pi/3}]}{[\text{Ai}'(b_n)]^2 - b_n[\text{Ai}(b_n)]^2}$$

where \( \text{Ai} \) represents the Airy function and

$$b_n = \tau e^{2i\pi/3} = (k_n^2 - k_0^2) l^2 e^{2i\pi/3}$$

are the zeros of the expression:

$$\text{Ai}'(b_n) + q[e^{i\pi/3}] \text{Ai}(b_n) = 0$$

The abbreviations used are: \( q = \frac{ik_0 \rho l c}{Z} \), \( l = \left( \frac{R}{2k_0^2} \right)^{1/3} \), \( \tau = (k_0^2 - k_n^2) l^2 \) and \( k_0 = \omega/c(0) \) where \( R \) is the radius of the limiting ray. Later, Berthelot [46] explains that the creeping wave solution can be improved if the distance \( r \) in the argument of the Hankel function is interpreted as the shortest path between the source and the receiver that does not intersect the boundary, rather than the distance projected
as an arc length along the surface boundary. This theory was developed in a linear sound speed approximation for which the limiting ray is a perfect circular arc. In reality however, nonlinear temperature and wind speed gradients near the ground and up to high altitude prevent the use of this method for en route noise shadow zone fields. West et al. [44] came up with a technique to use the residue series in realistic atmospheres: the atmosphere is divided into atmospheric layers in which the effective sound speed is linear. A radius of curvature is computed in each of these layers, and an averaged radius of curvature is then calculated by:

$$R_{AV} = \frac{\sum_i R_i^2 \Delta \theta_i}{\sum_i R_i \Delta \theta_i}$$

where $\Delta \theta_i$ is the angle subtended by arc $i$ at its centre. $R_{AV}$ represents the radius of curvature of a circle arc and can therefore be used in the residue series solution.

2.2.5.2 Benchmarking of the residue series solution: theoretical approach

Details regarding the numerical implementation of the residue series, and the determination of the roots $b_n$ are provided in reference [44], which also contains useful benchmarks: the first one shows the transmission loss at a fixed receiver in the shadow zone for various limiting ray radii, see Figure (2.15). Then, the second benchmark provides the transmission loss for a fixed radii of curvature, at different receiver ranges, see Figure (2.16). These benchmark results validate the numerical implementation of the residue series solution, for short-range horizontal propagation.

2.2.5.3 Benchmarking of the residue series solution: experimental approach

The residue series solution was also tested against experimental data taken by Embleton [22], for different atmospheric conditions and over hard and soft grounds. Although weather data or ground characteristics are not provided by Embleton, these comparisons are another way of verifying that the numerical implementation of the residue series solution is accurate. Results are shown in Figures (2.17) and
Figure 2.15: Benchmark of the residue series solution for different radii of curvature, over a soft ground ($\sigma = 120$ kRayls/m). Left: benchmark from [44]. Right: residue series numerical implementation. The point source and receiver are both located 1.5 meters above the ground. The receiver range is $r=200$ meters.

Figure 2.16: Benchmark of the residue series solution for different receiver ranges, over a soft ground ($\sigma = 120$ kRayls/m). Left: benchmark from [44]. Right: residue series numerical implementation. The point source and receiver are respectively located 2 and 1.5 meters above the ground. The limiting ray radii of curvature is equal to 1925.19 m.

(2.18). The close match for both test cases at all pure-tone frequencies validate the residue series implementation. Sections (2.2.5.3) and (2.2.5.2) demonstrate the correct implementation of the numerical solution of the residue series solution. This method is an addition to the ray tracing technique in shadow zones.

2.2.5.4 Extension of the residue series solution to en route geometry

The residue series solution works for horizontal propagation, in an environment in which source and receivers are close to the ground. A simple approach for a high
Figure 2.17: Comparison of residue series implementation with experimental data over hard ground. Left: benchmark from [22]. Right: Residue series numerical implementation. Sound pressure level relative to free field for a source on the ground, and receivers at a range=200 meters, from the ground to 8 meters above the ground, for hard boundary conditions (asphalt). Pure-tone frequencies of 250, 2000 and 4000 Hz are propagated, in a refractive atmosphere characterized by an effective sound speed gradient $g=-0.35/s$. In the absence of any meteorological data, a temperature $T=285$ K and relative humidity $r_h=80\%$ were used to compute the absorption losses. These data were determined by considering the output at $f=4000$ Hz as they mainly impact the high frequency absorption losses.

An altitude source is to compute the cut-off field along the limiting ray (which does not include an excess ground attenuation term), then set a secondary source 1 meter above the ground and compute the diffracted pressure field from this secondary source position.

Furthermore, for en route noise, the cut-off field would be comparable to background noise in many cases. The detectability of noise in acoustic shadow zones is likely to be negligible, especially beneath an overflight section or generally for any situation linked with the propagation of a continuous noise source.

2.2.5.5 Other methods to predict acoustic shadow zone fields

The residue series method is computationally expensive, which mainly originates from the determination of the poles $b_n$. Other methods have been introduced in
Figure 2.18: Comparison of residue series implementation with experimental data over soft ground. Left: benchmark from [22]. Right: Residue series numerical implementation. Sound pressure level relative to free field for a source on the ground, and receivers at a range—230 meters, from the ground to 8 meters above the ground, for hard boundary conditions (asphalt). Pure-tone frequencies of 250, 500 and 1000 Hz are propagated, in a refractive atmosphere characterized by an effective sound speed gradient $g=0.30/s$. In the absence of any meteorological data, a temperature $T=285$ K and relative humidity $r_h=70\%$ were used to compute the absorption losses. These data were determined by considering the output at $f=4000$ Hz as they mainly impact the high frequency absorption losses. In the absence of any information regarding the ground flow resistivity, a value of $\sigma=190$ kRayls/m was utilized to compute the excess ground attenuation. This specific value was assessed by evaluating the low frequency output as it is particularly sensitive to $\sigma$.

available software such as NORD 2000. They tend to be less accurate, but quicker, which is why they are used in engineering applications. These alternate concepts are all based on the analogy shown in Figure (2.19). The curved ray is replaced by a curved ground, coarsely modelled as an edge or a wedge depending on the methods. The resulting shadow zone field is the combination of a diffraction term and a ground effect. Lam [47] proposed a calculation based on edge diffraction theory, whereas NORD 2000’s implementation considers Pierce and Haiden’s wedge diffraction theory, see [38]. The ground related term is respectively computed for flat and wedge-shaped ground in Lam’s method and in NORD 2000. Figure (2.20)
Figure 2.19: Analogy between the propagation of sound in a refractive atmosphere over a flat terrain and the propagation of sound in a homogeneous atmosphere over a curved terrain [43].

highlights the comparison of three available methods to compute the shadow zone field for a specific case. The PE solution [2] is the most accurate. The residue series solution, presented in section (2.2.5.1) matches closely to the PE output for short-to mid ranges within the shadow zone, whereas the wedge diffraction solution used in NORD 2000 only approaches the PE levels close to the cut-off location, where the limiting ray grazes the ground. These observations are sufficient to justify the consideration of the residue series in AERNOM’s shadow zone routine.

2.2.6 Turbulence effects in the ray-tracing context

Up to this point, although the moving medium effects have been taken into account, no turbulence effects have been accounted for. This section discusses the implementation of the effects of atmospheric turbulence in the ray-tracing context. These are quantified in realistic atmospheres by the fluctuation of the thermodynamic sound speed and wind speed due respectively to temperature and wind
speed variations around their average value. These fluctuations in the boundary layer have multiple effects on sound propagation:

- Sound scattering due to sound speed gradient variations.
- Reduction of interference dips in excess ground attenuation due to the loss of coherence between two adjacent rays.
- Sound scattering in the shadow zone.

An acoustic refractive index \( n = \frac{c_0}{c_{\text{eff}}} \) is introduced in order to mathematically quantify the effects of turbulence. \( n \) is divided into two contributions: the average value \( \overline{n} \) and the fluctuation \( \mu \) such as \( n = \overline{n} + \mu \). \( \mu \) is constrained by \( \mu \ll n \) and \( \overline{\mu} = 0 \). The turbulence strength is measured by the square root of the square of the standard deviation of the fluctuating part of the index of refraction for a Gaussian...
turbulence model, $\sqrt{\mu^2}$. Typical values for $\mu^2$ range from $1.10^{-6}$ to $5.10^{-4}$ for weak to strong turbulence [38].

Modelling turbulent ray scattering with the ray-tracing method is computationally expensive: the wind speed and temperature gradient fluctuations lead to many different ray trajectories along which sound pressure level variations are computed. Then, a logarithmic summation provides the averaged turbulent level at a specific location. Additionally, most of the travelled distance from an en route source occurs outside the boundary layer, which generally corresponds to the lowest 200 meters of our atmosphere, so this phenomenon is discarded in AERONOM. On the other hand, the effects of turbulence on the ground effects and the shadow zone scattering are relatively well modelled. The following two sections will provide some guidance regarding the numerical implementation of these two phenomena. Both methods consider Gaussian spatial correlation coefficient

2.2.6.1 Turbulence effects on EGA

Chessell [39] and Daigle [48] both introduced some useful concepts to include turbulence effects in the excess ground attenuation formulation shown in section (2.2.4.3). A turbulence coherent factor $C$ is defined by:

$C = e^{-\sigma^2(1 - \rho)}$

where

$\sigma^2 = A\sqrt{\pi} \langle \mu^2 \rangle k^2 RL_0$

and

$\rho = \frac{\sqrt{\pi} L_0}{2h} \text{erf} \left( \frac{h}{L_0} \right)$

In these definitions $A$, $\langle \mu^2 \rangle$, $k$, $R$, $L_0$ and $h$ are respectively a range-dependent constant, the squared-averaged local turbulent strength, the wave number for a given frequency, the path length, the correlation length ($L_0=1.1 \text{ m}$) and the maximum transverse path separation between the direct and reflected ray.

In a turbulent atmosphere, the excess ground attenuation is modified from
Figure 2.21: Illustration of the turbulence effects in a homogeneous atmosphere. Source height = 5 m, receiver height = 1 m, receiver range = 600 m, propagation over grassland ($\sigma = 300$ kPa.s.m$^{-2}$). As the turbulence gets stronger, the interference dip is diminished and shifts to lower frequencies.

Equation (2.7) and is expressed as:

$$EGA = 10 \times \log_{10} \left( 1 + \left( \frac{|R_{p,i}|}{r'} \right)^2 + \frac{2|R_{p,i}|}{r'} \sin \left( \frac{\mu \Delta t}{f_i} \right) \cos \left( \frac{\eta \Delta t}{f_i} + \theta_i \right) \times C \right)$$

Atmospheric turbulence reduces the destructive interference dips as shown in Figure (2.21), these effects being stronger over soft grounds at higher frequencies.

### 2.2.6.2 Scattering in the shadow zone due to turbulent scattering

For short-range horizontal sound propagation, the inherent problem of the ray tracing method to be unable to determine the sound pressure level in the shadow zone is a serious issue. As presented in Figure (2.20), the residue series solution generally provides accurate shadow zone pressure levels in a non-turbulent atmosphere. However, the issue of sound scattering in a turbulent atmosphere remains open in
Figure 2.22: Setup for the TSSR method from [47]. The turbulence effects are modelled by ray height variations which in turn act on the diffraction term. Each snapshot corresponds to different pressure levels, and their energetic average leads to turbulent pressure levels.

The ray-tracing community. Quick engineering type calculations usually consider a constant excess sound pressure level of -20 dB in the shadow zone, with respect to homogeneous atmospheric conditions. Lam [47] recently published a new method to predict shadow zone levels in turbulent atmosphere using a Gaussian turbulence spectrum.

The Turbulent Scattered Sound Rays (TSSR) method introduced by Lam is based on the analogy presented in Figure (2.19) and therefore uses an edge diffraction term and a flat ground effect term to compute the shadow zone field. The effects of turbulence are taken into account by statistically modifying the ray height as shown in Figure (2.22) which engenders a variation of the geometrical properties in the diffraction term. Diffraction losses are quantified by the Insertion Loss (IL) defined by:

$$IL = 20 \log_{10} \left( \frac{P_{\text{without barrier}}}{P_{\text{with barrier}}} \right)$$

The transmission loss in the shadow zone is then defined by:

$$TL_{\text{shadow}} = IL + EGA$$
Figure 2.23: Setup for the Menounou edge diffraction calculation; The source theoretically located at $-\infty$ produces a straight ray that gets diffracted by the curved ground. $r$ is the straight distance between the edge and the receiver located in the shadow zone.

Lam uses Maekawa’s edge diffraction theory, which has recently been improved by Menounou [49], by introducing two Fresnel numbers $N_1$ and $N_2$ defined as:

$$N_1 = 2 \frac{kr}{\pi} \cos^2 \left( \frac{\phi - \phi_0}{2} \right)$$

and

$$N_2 = 2 \frac{kr}{\pi} \cos^2 \left( \frac{\phi + \phi_0}{2} \right)$$

where $k$ is the wave number and the other parameters are defined in Figure (2.23) Menounou’s diffraction term is more accurate than Maekawa’s older definition for a number of situations. The next paragraphs will compare the use of both definitions in the implementation of the TSSR method, after having presented the successive steps that build up the shadow zone field in a turbulent atmosphere:

1. A range and turbulent strength specific ray height standard deviation $\Delta z_{\text{STD}}$ is computed, from a pre-defined regression curve.

2. Random ray height values are then generated, using a Gaussian distribution with zero mean and standard deviation $\Delta z_{\text{STD}}$.

3. A diffraction term is computed using each ray height corresponding to a snapshot (for $N$ realizations).

4. The mutual coherence derivation presented in section (2.2.6.1) is used to compute the ground effects in a turbulent atmosphere for a flat ground situation.
Figure 2.24: Benchmarking and comparison of Maekawa and Menounou’s diffraction methods: excess attenuation [dB] with respect to homogeneous atmosphere propagation, as a function of range, from [47]. The settings are taken from Attenborough et al’s upward refracting benchmark [40]: limiting ray radii of curvature $R=3440$ meters ($g=-0.1$/s), receivers 1.5 meters above the ground for ranges up to 10 km, source located 5 meters above the ground, a turbulence strength $\mu_0^2=3.10^{-6}$ and a frequency $f=1000$ Hz. The thin dotted line shows the PE output in a non-turbulent atmosphere.

5. $N\ IL$ terms are added to the turbulent EGA term.

6. These $N\ TL$ values are averaged energetically to form the turbulent $TL$.

The numerical implementation of the TSSR method was benchmarked against some of the figures published by Lam. As noted by the author, the solution works well except close to the cut-off location, and generally underpredicts the shadow zone transmission loss compared with the PE output. Nevertheless, the solution is very fast at high frequencies compared with a PE type implementation, which makes it attractive to use for community noise issues. Figure (2.24) shows the impact of choosing one edge diffraction solution rather than the other. After having tested various configurations, it seems like the Maekawa and Menounou theories are best suited for high and low frequencies respectively. Overall, the TSSR method is an appropriate technique to predict the shadow zone transmission loss in a turbulent atmosphere, given its low computational time.
2.2.7 Summary of transmission loss calculations

After having elaborated on each of the transmission loss contributions, this section summarizes the prediction of receiver sound pressure levels depending on their position:

in the illuminated zone, each receiver sound pressure third octave band level is:

\[ SPL_{\text{receiver}}(f) = L_w(f) - \Delta L_{\text{geom}} - \Delta L_{\text{absorption}}(f) + EGA(f)[+\text{turbulence}(f)] \]

in the shadow zone, each receiver sound pressure third octave band level is:

\[ SPL_{\text{receiver}}(f) = SPL_{\text{cut-off}}(f) - \Delta L_{\text{shadow}}[+\text{turbulence}(f)] \]

where \( SPL_{\text{cut-off}} \) is the cut-off field, without any ground term. The third-octave band levels are then logarithmically added to compute an overall sound pressure level.

After having presented all the loss mechanisms using the ray-tracing method, Chapter 3 will address the issue of weather inputs, in the context of en route noise propagation. In particular, the effects of atmospheric data on long-range sound propagation will be discussed.
Chapter 3

The effects of meteorology on long-range vertical propagation

3.1 NOAA weather balloon data

Chapter 2 introduced AERNOM’s modules to compute the full transmission loss from a high altitude source to a receiver. They characterize the transfer function between a source power level and a receiver sound pressure level, as presented in section (1.4) as the step 2 in the full process of en route noise ground contour generation. The meteorology plays an essential role in long-range sound propagation. Furthermore, for vertical propagation, it is essential to consider the altitude stratification of our atmosphere. The meteorological profiles are specified as inputs, see step 1 of Figure (1.11), and govern the ray refraction as well as the associated losses.

The 1976 US standard atmosphere provides average, piece-wise continuous, mid-latitude temperature, pressure and density profiles. The standard temperature profile is characterized by a 6.5 K/km constant lapse rate up to 10 km. There are however no standard profiles for humidity and wind speed. The propagation of en route noise therefore calls for more sophisticated meteorological models, that handle geographical and temporal variability if possible. The National Oceanic and Atmospheric Administration (N.O.A.A.) weather balloon data was chosen to be the most efficient way to gather meteorological inputs. Weather balloons are
launched on 102 different sites within the continental United States, on a daily basis. These meteorological stations provide all the atmospheric parameters that are needed for the prediction of aircraft en route noise propagation: temperature, pressure, dew point, wind speed and wind direction stratified profiles. The dew point and the temperature are used to compute the percentage of water vapor ($\%H_2O$) which impacts on the frequency-dependent absorption coefficients.

### 3.1.1 The limitations of using weather balloon data

Although the NOAA data are extensive and made available on a public database, using balloon data for outdoor sound propagation comes with certain limitations that are worth mentioning:

- Weather balloon data represent a snapshot of the atmosphere at a particular time and a specific geometrical altitude, in a so-called *frozen atmosphere* approximation. This means that the wind gusts and rapid changes in temperature and humidity are not accounted for, and therefore neglected.

- The weather balloon naturally drifts sideways due to high altitude winds. This typically means that the captured weather does not exactly represent a vertical profile per location, but rather an averaged vertical profile around a given location. This drift can reach 30 km for balloons that typically reach altitudes of 30 to 40 km. High altitudes are hard to access because of the pressure decrease which makes the balloons burst.

Despite these limitations, the NOAA data provide a reliable mean to assess en route ground contour variability should it be geographical or seasonal, as detailed in a section to follow.

### 3.1.2 NOAA data treatment

This section presents the analysis of the meteorological treatment considered in the context of this thesis, from the actual measurements to the generation of smoothed, averaged, altitude-dependent profiles. The end result is a set of polynomial coefficients for each of the atmospheric parameters.
Figure 3.1: Winter 2010 wind speed profile above Pittsburgh, PA. From the original data to single-altitude data (open circles). The final step produces the smoothed profile (solid line). Notice the large spread in the data; also represented are the solid dashed lines representing the wind speed profiles corresponding to same day measurements leading to the minimum and maximum wind speed at 10 km altitude.

3.1.2.1 First step: data collection

The initial step is constituted by the data collection at a specific location for a given time frame, which can vary from a day to a season. Multiple data sets are gathered and averaged to produce meteorological data for each altitude. Figure (3.1) illustrates this process. The smoothed profile is obtained after the final step which will be presented next.

3.1.2.2 Second step: data smoothing and profile generation

The single-altitude data are smoothed using a moving-point averaging process. This reduces the scatter shown in Figure (3.2). A polynomial fit is then applied to the smoothed profile. The order of the polynomial fit is determined by an
error analysis: the order of the polynomial is increased until the error between the smoothed data and fit is minimized. An alternative method is to use a spline interpolation between successive altitudes.

3.2 Example effects of meteorological variability on outdoor sound propagation

3.2.1 Effects of the atmospheric data scatter on the ground sound pressure level profile

The atmospheric data corresponding to the wind speed profiles highlighted on Figure (3.1) as black thick dashed lines were used as inputs to AERNOM as well as the smoothed, averaged profiles. The B777 retrofitted NPD data were used as a noise input, see Table (5.3), and the sound pressure levels were computed in the downwind refracting plane for which the propagation and the wind speed...
Figure 3.3: Effects of atmospheric scatter on predicted ground sound pressure level profile, for the 2010 winter season above Pittsburgh, PA. The red solid thick line is AEROM’s output with smoothed, averaged atmospheric profiles; the black dashed thick lines represent the output for which the atmospheric input data correspond to extreme wind speed profiles.

direction are aligned. The results are shown on Figure (3.3). Overall, there is a 5 dB maximum variability between the predicted sound pressure levels. More results regarding the effects of weather variability will be presented throughout Chapter 4.

3.2.2 Ray-refraction seasonal variability

Using the data treatment method described in section (3.1.2) emphasizes seasonal variability at a particular location, as shown in Figure (3.4). General observations lead to the following observations: the temperature gradients are usually larger in the summer, and high-altitude winds are greater in the winter. These facts directly
Figure 3.4: 2010 Seasonal atmospheric variability above Pittsburgh, PA. Seasonal temperature profiles (left). Seasonal wind speed profiles (right).

Figure 3.5: Seasonal variability for ray refraction in the downwind plane and a launching angle $\theta_{\text{launch}} = -25^\circ$ above Pittsburgh, PA. Here the ray bends more in the summer season.

impact the ray refraction as highlighted in Figure (3.5), for a specific launching angle. Here the ray bends more in the summer season.

3.2.3 Atmospheric absorption seasonal variability

The humidity seasonal variability on the other hand largely contributes to the seasonal variability of ground sound pressure levels. A short mathematical description first explains the steps to transition from a dew point temperature to a percentage
of water vapor which constitutes the input of the absorption coefficient algorithm.

### 3.2.3.1 From dew point temperature to water vapor content

The NOAA data provide the dew point temperature as a function of altitude. In order to get to the corresponding percentage of water vapor, the dew point temperature is first converted to a relative humidity using the Magnus approximation:

\[
r_h = 100 \times e^{\left( T_{\text{dew}} \times \frac{\beta_T T}{\lambda_T + T} - \frac{\lambda_T \beta_T T}{\lambda_T + T} \right)}
\]

where \( T_{\text{dew}}, T, \beta_T \) and \( \lambda_T \) are respectively the dew point temperature, the atmospheric temperature, and two temperature-dependent coefficients. Then, the temperature-dependent vapor pressure of water \( e \) is computed from the derivation by Lowe and Ficke [50]. The volume mixing ratio of water is then \( x_{\text{H}_2\text{O}} = \frac{r_h \times e}{100 \times P} \) where \( P \) is the atmospheric pressure from which the mass concentration of water \( c_{\text{H}_2\text{O}} \) is calculated using \( c_{\text{H}_2\text{O}} = x_{\text{H}_2\text{O}} \rho_{\text{air}} M \) where \( \rho_{\text{air}} \) and \( M \) are respectively the air density and molar mass of water. Finally, the percentage of water vapor \( X_{\text{H}_2\text{O}} \) is derived using equation (D.15) of the ANSI S1.26-1978 standard:

\[
X_{\text{H}_2\text{O}} = 4.55e^{-4} \times c_{\text{H}_2\text{O}} \times T \times \frac{P}{P_{\text{std}}}
\]

where \( P_{\text{std}} = 101.325 \) kPa is the standard atmospheric pressure.

### 3.2.3.2 The effects of humidity variability

The derivation presented in section (3.2.3.1) was used to produce polynomial humidity profiles for both the winter and summer seasons above Pittsburgh, PA, see Figure (3.6). Following work published by Larsson [33], it appears interesting to use the NOAA weather balloon data to assess the variability of the cumulative absorption losses along the vertical ray from a point source located at 10 km altitude to the ground. The results are shown on Figure (3.7). The greatest cumulative losses occur at mid-altitudes in the 4 to 6 km range, this being mainly due to the relaxation losses associated with oxygen. Figure (3.7) highlights several important results:
Most of the absorption losses occur at altitudes below 6 km for all the frequencies.

Although the humidity is higher during summer seasons, this does not imply that absorption losses are lower for all the frequency bands. In fact, summer seasons could lead to lower absorption at low to mid-frequencies but higher absorption at mid to high-frequencies, as shown on Figure (3.7).

### 3.2.4 Geographical meteorological variability

The previous sections presented seasonal, or more generally temporal, meteorological variability for a specific location. The same comparative study at different locations for the same season would further increase our understanding. Chapter 4 will address some results of geographical meteorological variability. Above all, temporal and geographical meteorological data are critical to the complete treatment of long-range vertical propagation.
Figure 3.7: Seasonal variability for the cumulative absorption losses from a high altitude source to the ground above Pittsburgh, PA. for several pure-tone frequencies. Blue: winter ; Red: summer. Solid thick line: 125 Hz ; thick dash-dotted line: 250 Hz ; thin solid line: 500 Hz ; thin dotted line: 1000 Hz.
AERNOM to predict aircraft en route noise ground contours

Chapter 2 introduced AERNOM’s loss mechanism routines and Chapter 3 the role played by meteorology on long-range propagation. In particular, the consideration of geographical and temporal averaged atmospheric data was presented. This chapter will present some results of the association of the ray-tracing propagation method with input meteorological data to compute ground noise contours. Stationary point source ground contours will first be addressed, then the effects of a high altitude overflight will be discussed. The effects of turbulence are discarded in the remainder of this thesis.

4.1 Stationary point source ground sound pressure level contours

These sections will introduce the propagation of noise from a high altitude stationary source. This particular situation is not realistic but carries the advantage of emphasizing the wind effects in the resulting ground contours, as well as the impact of atmospheric variability on the noise levels.

A pseudo-three-dimensional methodology to compute ground contours will first be presented. Then, the prediction of the limiting angle in upwind refracting planes will be derived. This section will finally open up to the effects of an overflight,
Figure 4.1: AEROM is run radially from the source in each two-dimensional plane around the Z axis. The green and dark red planes are examples of two-dimensional planes. The propagation stops at the limiting angle in upwind refracting planes or at the horizontal launching angle in downwind refracting planes.

modelled as a line source.

4.1.1 Pseudo-three-dimensional methodology to produce ground contours

AEROM uses the two-dimensional ray equations. By rotating around the vertical Z axis, it is possible to consider noise propagation in three-dimensional space. This method has already efficiently been handled [2] and provides a way of numerically propagating noise in three-dimensional space using two-dimensional formulations that are more computationally friendly. The methodology is illustrated on Figure (4.1). Noise levels are then computed using two-dimensional vertical polar grids and reproduced on a two-dimensional horizontal Cartesian receiver grid. The transfer between these distinct grids takes place in an interpolation step [51], as shown on Figure (4.2). The angle increment \( \phi \) around the vertical Z axis is therefore not
Figure 4.2: From receiver grids aligned from the radial runs to the Cartesian ground contour grid. For each contour grid point, the interpolation process considers two closest receiver grid points to compute the corresponding noise level.

restricted to small values and \( \phi = \frac{\pi}{5} \) has been found to be a typical good value to use.

In the context of this thesis the following approximations are assumed in the generation of noise ground contours:

- The wind direction is assumed to be constant with altitude.
- Propagation occurs above flat and even terrain.

Assuming the wind direction corresponds to the vertical plane defined by \( \phi = 0 \), the propagation starts in the downwind plane and spans incremental \( \phi \) values up to the upwind plane for which \( \phi = \pi \). In a specific two-dimensional plane, the altitude-dependent wind speed profile becomes \( v_{2D}(z) = v_{\text{NOAA}}(z) \times \cos(\phi) \). The flat and even ground approximations are used to symmetrically construct the ground contour for \( \pi < \phi \leq 2\pi \) based on the levels computed for \( 0 \leq \phi \leq \pi \).
4.1.2 Limiting angle for upward refracting two-dimensional planes

The accurate definition of the limiting angle $\theta_{\text{limit}}$ in an upward refracting two-dimensional plane is critical to the transition between the illuminated and shadow zone for which the solutions are different as presented in section (2.2.7). The slowness vector was introduced in section (1.3.2.5); by neglecting any atmospheric range-dependence, the range-component of the slowness vector is constant along any given ray:

$$S_r = \frac{\cos(\theta)}{c + v \times \cos(\theta)} = \text{constant}$$

where $\theta$ is the angle between the horizontal and the ray direction. This property is used as a starting point in the prediction of the limiting angle in an upward refracting atmosphere.

First, the maximum effective sound speed $c_{\text{eff}}(z) = c_{\text{thermo}}(z) + v(z) \times \cos(\phi)$ and its corresponding altitude are recorded. This altitude corresponds to the ray turning point. A grazing ray at the turning point will never refract downwards, thus justifying that the calculation starts at this specific altitude. Then, considering the grazing ray for which $\cos(\theta_{\text{grazing}}) = 1$, a recursive algorithm from the turning point to the source altitude leads to the limiting angle prediction. The first step is to apply the property of $S_r$ between the turning point for a grazing ray approach and a slightly higher altitude:

$$S_r(z_{\text{turn}}) = S_r(z_{\text{turn}} + \Delta z)$$

$$\frac{1}{(c + v)(z_{\text{turn}})} = \frac{\cos(\theta(z_{\text{turn}} + \Delta z))}{(c + v)(z_{\text{turn}} + \Delta z)}$$

$$(c + v)(z_{\text{turn}}) \times \cos(\theta(z_{\text{turn}} + \Delta z)) = c(z_{\text{turn}} + \Delta z) + v(z_{\text{turn}} + \Delta z) \times \cos(\theta(z_{\text{turn}} + \Delta z))$$

$$\cos(\theta(z_{\text{turn}} + \Delta z)) = \frac{c(z_{\text{turn}} + \Delta z)}{(c + v)(z_{\text{turn}}) - v(z_{\text{turn}} + \Delta z)}$$

Then, the same process is applied in atmospheric layers in which the sound speed and wind speed are assumed to be constant up to the source altitude, see Appendix A for additional details. A typical atmospheric layer width for en route noise propagation was found to be $\Delta z = 1$ m. Although larger values decrease the computational time, they lead to deviated limiting angle predictions for non-linear
atmospheric profiles.

If the turning point corresponds to the source altitude, then propagation occurs in a downward refracting plane and launching angles span from $\theta_{\text{launch}} = -\frac{\pi}{2}$ to $\theta_{\text{launch}} = 0$. On the other hand, if the turning point corresponds to any other altitude value, then the propagation is characterized by upward refraction and in that case, $-\frac{\pi}{2} \leq \theta_{\text{launch}} \leq \theta_{\text{limit}}$. 

4.1.3 Example wind speed and temperature gradient effects on ground contours

The ground contours from a stationnary source would be circles in a stable atmosphere. Indeed, each vertical two-dimensional plane around the Z axis would be characterized by the same temperature gradients. In a windy atmosphere, the resulting ground contours are more complicated. As the two-dimensional runs sweep over the $\phi$ angles around the Z axis, the wind contributions vary and the refraction switches from downwind to upwind refraction. If the atmospheric temperature is constant, downwind refracting planes are defined by $0 \leq \phi < \frac{\pi}{2}$ whereas $\frac{\pi}{2} < \phi \leq \pi$ characterize upwind refracting planes. Straight line propagation occurs for $\phi = \frac{\pi}{2}$.

The temperature and wind speed gradient effects were benchmarked in AER-NOM against a recent publication [14] in which the following settings were used:

- The point source is located 1,500 meters above the ground.
- The wind direction is assumed to be constant along the positive X direction.
- The contours show the geometrical and refraction losses only.
- The benchmark considers constant wind speed and temperature gradients.
- Standard (day time) and inverted (night time) temperature lapse rate contours are generated.

Figures (4.3) and (4.4) show the comparison of AER-NOM’s output compared with the benchmark for standard and inverted temperature lapse rates, respectively. The general shape of the ground contours is accurately reproduced and the transmission loss values are predicted within 2 dB. The angle $\zeta$ defines the shadow
boundary angle which delimits the shadow zone, and is analytically defined by:

\[ dv \times \cos(\zeta) \leq \frac{c_0 \times \beta}{2T_0} \]

where \( dv \), \( c_0 \), \( \beta \) and \( T_0 \) are respectively the wind speed gradient, the ground sound speed, the temperature lapse rate and the ground temperature.

4.1.4 En route noise ground contours

This section shows the sound pressure level ground contours from a high-altitude omnidirectional stationary point source located 10 km above the ground using an A-weighted ground level measured aircraft sound power level [32] with a 2 kHz cut-off frequency (higher frequencies are highly attenuated by atmospheric absorption as discussed in Chapter 2). Table (4.1) provides the corresponding A-weighted third octave-band sound power levels. The ground contours are generated with the following settings:

- The ground is modelled as grassland with an associated flow resistivity \( \sigma = 300 \) kRayls.
- The wind direction is assumed constant in the positive X-direction.
Figure 4.4: Benchmark 2: Negative temperature lapse in windy atmosphere, leading to a shadow boundary angle \( \zeta = 112^\circ \) Transmission loss [dB] representing geometrical and refraction losses only. Left: benchmark from [14]. Right: AER-NOM output.

Table 4.1: A-weighted ground level aircraft sound power level third octave-band spectrum [32].

<table>
<thead>
<tr>
<th>( f_{1/3, \text{Oct}} ) [Hz]</th>
<th>50</th>
<th>63</th>
<th>80</th>
<th>100</th>
<th>125</th>
<th>160</th>
<th>200</th>
<th>250</th>
<th>315</th>
<th>400</th>
<th>500</th>
<th>630</th>
<th>800</th>
<th>1000</th>
<th>1250</th>
<th>1600</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL [dB]</td>
<td>120</td>
<td>125</td>
<td>128</td>
<td>131</td>
<td>135</td>
<td>139</td>
<td>142</td>
<td>143</td>
<td>145</td>
<td>146</td>
<td>148</td>
<td>148.5</td>
<td>149.5</td>
<td>149</td>
<td>148</td>
<td>148</td>
<td></td>
</tr>
</tbody>
</table>

- The angle increment around the vertical Z-axis is \( \phi = \frac{\pi}{5} \).
- The launching angle increment in each two-dimensional plane is \( \Delta \theta_{\text{launch}} = 2^\circ \).
- Receivers are located 1 meter above the ground.
- Boundary layer turbulence is neglected.
- Forward flight effects are not accounted for.
- The receiver grid size is 80 km by 80 km.

Several runs for alternate averaged atmospheric data illustrate the effects of wind speed and temperature gradients on the ground contour shapes, and the variability associated with seasonal and geographical meteorological data. The
Pittsburgh 2010 seasonal data presented in Chapter 3 were used as inputs to generate winter and summer average noise ground contours, see Figure (4.5). As expected, summer averaged ground levels are generally higher due to higher water vapor atmospheric content. This statement is illustrated on Figure (4.6).

Additionally, 2010 winter average weather data above Denver, CO were considered to compare the resulting ground contour against the 2010 winter average Pittsburgh data, see Figure (4.7). For the same temporal averages, dryer environments such as the atmosphere above Denver lead to lower ground noise levels. This statement is illustrated on Figure (4.8).

After having given examples of stationary point source resulting contours, the following section will discuss the impact of en route altitude overflights on the ground.

4.2 En route overflight ground sound pressure level contours

4.2.1 From a point source to a line source representation

More realistic situations involve the overall impact of an overflight. Like any moving and continuous noise source such as a highway, an aircraft overflight can be modelled as a succession of point sources. The impact of a flight track is
Figure 4.6: 2010 summer minus winter averaged ground sound pressure level differences \[\text{dB}_A\] at Pittsburgh, PA for a wind blowing in the \(+X\) direction. Summer averaged ground noise levels are overall higher than winter averaged ground noise levels, especially in the direction aligned with the wind speed direction and corresponding to downwind refraction. Thus, the summer minus winter sound pressure level differences are mostly positive.

Figure 4.7: 2010 winter average noise level ground contours \[\text{dB}_A\] at Denver, CO (left) and Pittsburgh, PA (right). Both contours show different noise levels, thus illustrating geographical variability given the same time frame. Computation time between 7 and 8 minutes.
Figure 4.8: Ground sound pressure level differences between 2010 winter average noise level ground contours [dB$_A$] at Pittsburgh, PA and Denver, CO for a wind blowing in the $+X$ direction. Dryer environments such as the atmosphere above Denver lead to lower ground noise levels. Hence, the levels at Pittsburgh are higher, and there is a positive difference between Pittsburgh and Denver averaged ground sound pressure levels.

represented by the effect of $N$ point sources with associated weighted sound power levels:

$$L_{w_{\text{line}}} = 10 \times \log_{10} \left( \frac{10^{L_{w_{\text{point}}}/10}}{N} \right)$$

where $L_{w_{\text{line}}}$ and $L_{w_{\text{point}}}$ are sound power levels associated respectively with the line source and point source representations and $N$ is the number of point sources that model the line source effects. In the flat and even ground approximation, two distinct point sources located on the same axis and separated by a distance $d$ produce the exact same noise levels on the ground translated a distance $d$. Consequently, the line source ground contour generation starts with the computation of ground noise levels from a stationary point source $S_0$ located above the $(X = 0, Y = 0)$ grid point. The impact of a point source $S$ located a distance $d$ from $S_0$ along the over-flight portion is taken from the levels $S_0$ generates with receiver point coordinates translated a distance $d$ from the receiver points associated with $S_0$. Finally the contour grid point levels are generated using an interpolation technique mentioned in section (4.1.1).
4.2.2 Seasonal variability of overflight ground noise level contours

A 150 km long en route overflight track located 10 km above the ground was numerically constructed by considering 50 point sources equally spaced from each to another. The settings are similar than those used to generate point source contours in the previous section. Some additional parameters are the following:

- The noise metric used for an overflight is the Sound Exposure Level (SEL) which takes the overflight duration $\tau$ into account: $SEL = L_p + 10 \times \log_{10} \tau$.

- The overflight duration time is calculated using a Mach number $M_c=0.8$.

- The en route flight track is aligned with the positive $X$ axis.

Winter and summer meteorological averaged overflight SEL ground contours were generated using the 2010 NOAA data above Pittsburgh. The results are shown on Figure (4.9). Moving flight effects have not yet been included. Lower shadow zone levels from a particular point source are dominated by higher levels from other point sources located along the flight track. Consequently, line source contours call for the quantification of noise perpendicular to the flight track along the so-called sideline distance which defines the receiver distance to the perpendicular projection of the flight track on a horizontal plane. Instead of using the sideline distance to characterize the receiver position with respect to en route overflights, the slant-range distance parameter is a better choice since it allows its use for operations with a certain lateral position with respect to the microphone. The overflight geometry is shown on Figure (4.10): the slant-range represents the straight-line distance between a receiver and the en route overflight and consequently starts from 10 km, corresponding to vertical propagation. Furthermore it appears essential to study the noise impact along slant-range distances to potentially tackle the potential issue of en route flight re-routing over preserved areas such as National Parks.

4.2.3 Overflight noise as a function of slant-range distance

This section presents the SEL decay as a function of slant-range distance for a high altitude overflight. As highlighted in Figure (4.10), this quantity varies with
Figure 4.9: 2010 winter (left) and summer (right) average SEL ground contours [dB$_A$] at Pittsburgh, PA. from an en route overflight. Computation time just below 7 minutes. Moving flights effects not yet included.

Figure 4.10: Geometry from an overflight track [1]. $\beta$ is called the elevation angle. The slant-range distance is the shortest distance between a receiver and the flight track.

seasons, but also with geographical location and angle $\varphi$ between the wind direction and the flight track. Figure (4.11) provides the SEL decay as a function of slant-range distance for the 2010 winter and summer average weather above Pittsburgh, and for the 2010 winter average weather above Denver. Figure (4.12) shows the effect of angle between wind direction and flight track for the 2010 summer weather average above Pittsburgh, PA. As expected, the crosswind situation with $\varphi = 90^\circ$ leads to the higher levels because the direction perpendicular to the flight track is
then characterized by downwind refraction.

Small, random changes in meteorology were further assessed by generating 1%, 5% and 10% independent random changes in temperature, wind speed, dew point and pressure from the 2010 summer averaged profile measured above Pittsburgh. These variations were introduced by multiplying the original NOAA data by random sets of numbers between 0 and 1 associated with 1%, 5% and 10% maximum variation, respectively. Random changes only affect the large slant-range values when the wind direction and the flight track are aligned as shown on Figure (4.13). The maximum predicted deviation is 3 dB compared with the original balloon data. Random meteorological changes could therefore be one possible explanation of the observed scatter in measured ground noise levels for large slant-range distances.

Figure 4.11: SEL decay as a function of slant-range distance for various seasonal atmospheric conditions in 2010.
Figure 4.12: SEL decay as a function of slant-range distance for various angles between the wind direction and the flight track, for the 2010 Pittsburgh summer average atmosphere. The angle $\varphi$ in degrees represents the angle between the wind direction (assumed constant for all altitudes) and the flight track. For example, $\varphi = 90^\circ$ represents a crosswind situation.

Figure 4.13: Impact of random changes of NOAA balloon data on SEL decay with slant-range distance. The original data is the 2010 summer atmospheric data measured above Pittsburgh.
4.3 Moving flight effects in AERNOM

4.3.1 Theoretical developments

The contours and results showed up to now did not take the moving flight effects into account. These act in two different ways:

- The well-known Doppler effect acts on the received frequency for a moving source passing by a fixed receiver. As the moving source approaches the static receiver, the received frequency rises and then decreases as the moving source moves away from the receiver.

- Convective amplification occurs as a result of the motion of a pulsating body and will be simply modelled in the context of this thesis.

These moving effects are perceptible around airports: as the flight approaches a receiver, the spectrum shifts to higher frequencies and higher associated sound pressure levels. On the other hand, once the flight moves away from the receiver, the spectrum shifts to lower frequencies with lower associated levels. There are however no references in the literature for the geometry involved with en route noise propagation. Given the large propagation distances and the effects of atmospheric absorption, the moving flight effects are essential for reliable ground contour prediction.

The Doppler effects were implemented in AERNOM using a method described by Roy [52]. The frequency shift occurs at the source altitude and remains constant along a ray tube provided the dispersion is neglected which is a fair approximation. This shift is shown in Equation (4.1) for a given ray defined by a specific launching angle.

\[
\frac{f_r}{f_s} = \frac{1}{1 - \frac{V_{SG} \times \cos(\theta_{launch})}{c(z_s) + v(z_s)}}
\]

(4.1)

where \(f_r\), \(f_s\), \(V_{SG}\), \(\theta_{launch}\), \(c(z_s)\) and \(v(z_s)\) are respectively the received frequency, the emitted frequency, the aircraft speed relative to the ground, the ray launching angle, the sound speed and wind speed at the source altitude. The frequency at which the sound wave excites a moving particle of air and which is used to compute
the atmospheric absorption losses is then given by:

\[ f_\alpha = f_r (1 - \vec{v} \cdot \vec{s}) \]

where \( \vec{v} \) and \( \vec{s} \) represent respectively the wind speed and slowness vectors.

Dowling’s theoretical developments [53] were applied in AERNOM to model the convective amplification effects. Her solution assumes a finite length source region and is valid for frequencies below 2 kHz which corresponds to the frequency range of interest in en route noise propagation. For a given ray defined by a specific launching angle, Equation (4.2) shows the minimum ratio between emitted and stationary sound power level as a result of the source motion, where \( M_c = \frac{V_{SG}}{c(z_s)} \) is the Mach number referenced to the source altitude and \( \theta_{\text{launch}} \) is the ray launching angle.

\[ \frac{L_{w,\text{emitted}}}{L_{w,\text{stationary}}} = \frac{1}{(1 - M_c \times \cos(\theta_{\text{launch}}))^3} \]  

\[ \text{(4.2)} \]

### 4.3.2 Moving flight effects in slant-range SEL decay

The moving source effects were implemented in AERNOM and slant-range SEL decays were produced for alternate test cases. Results are provided on Figure (4.14) which shows that higher aircraft Mach numbers could potentially lead to lower SEL received noise levels on the ground due to the smaller overflight duration.

Figure (4.15) illustrates the effects of considering alternate angles between the wind and the flight track. With the moving flight effects, the crosswind situation, for which \( \varphi = 90^\circ \), appears to change the levels the most.

### 4.3.3 Time series sound pressure level variation

AERNOM also has the ability to produce time series sound pressure level variations for a fixed receiver located right beneath the flight track. The chosen source-receiver geometry is shown on Figure (4.16). These results indicate an asymmetry of the SPL as a function of time. In addition, the duration of the noise above a certain SPL is seen to be a function of wind direction. The noise at the receiver lasts the longest above a certain SPL when aircraft flies into a head wind.
Figure 4.14: Moving source effects on the slant-range SEL decay as a function of aircraft Mach number, for the 2010 summer average atmosphere above Pittsburgh when the flight track and the wind direction are aligned. The red curve provides the SEL decay when the moving effects are discarded.

The wind direction hardly impacts on the peak received level; however when the wind and flight track directions are aligned, a head wind situation would lead longer signal duration in a low background noise environment.

4.4 General quantification of the impact of en route noise on the ground

4.4.1 General trends for the propagation of en route noise

The previous sections provide big picture results for en route ground contours and slant-range predicted noise levels. From these observations it is possible to address the issue of trends for en route noise propagation:

- Summer seasons associated with high water vapor content generally lead to higher ground noise levels up to mid slant-range values (20-25 km).
Figure 4.15: Moving source effects on the slant-range SEL decay as a function of the flight track versus wind direction angle $\varphi$, for the 2010 summer average weather above Pittsburgh when the aircraft Mach number is equal to $M_c=0.8$. The red curve provides the SEL decay when the moving effects are discarded.

Figure 4.16: Source-receiver geometry for time series prediction.

- Winter seasons associated with larger high-altitude wind speeds generally lead to higher ground noise levels for large slant-range distances ($\geq 30$ km).
- The higher slant-range ground noise level values occur for crosswind propagation, when the flight track and the wind direction are perpendicular.
- Higher aircraft Mach numbers could lead to lower received SEL levels, due
to shorter overflight durations.

All these results were calculated with the same source power level spectrum. Other frequency spectra, and in particular other propulsion types such as open-rotors, might lead to different observations.

### 4.4.2 Extensive meteorological data treatment and slant-range level statistics

The aim of this section is to compute a general law that provides the maximum A-weighted sound pressure level (or any other metric) value as a function of slant-range distance by means of a fit to many NOAA weather data inputs in the continental United States. A total of 16 different locations scattered around the United States from coast to coast were considered. Their geographical location is shown on Figure (4.19). These locations represent extreme atmospheric conditions encountered in the continental United States, from dry to humid environments. Seasonal
Table 4.2: Theoretical 8-blade UDF/ATP A-weighted spectrum. Most of the energy is concentrated in the low frequency bands and therefore the overall A-weighted sound power level $L_{w,A}=154.1\, \text{dB}_A$ is lower than that for the jet source.

<table>
<thead>
<tr>
<th>$f$ [Hz]</th>
<th>200</th>
<th>400</th>
<th>600</th>
<th>800</th>
<th>1000</th>
<th>1200</th>
<th>1400</th>
<th>1600</th>
<th>1800</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL [dB$_A$]</td>
<td>151</td>
<td>147</td>
<td>144</td>
<td>141</td>
<td>139</td>
<td>137</td>
<td>135</td>
<td>133</td>
<td>132</td>
<td>130</td>
</tr>
</tbody>
</table>

weather averages were taken as inputs to predict slant-range maximum A-weighted noise level decays. This metric has previously been used to quantify the impact of en route noise [3], and does not take the overflight duration into account.

Turbojet and rotor type propulsion system source spectral content were considered for comparison purposes. The B777 retrofitted NPD data, shown in Table (4.1), is associated with an overall A-weighted sound power level $L_{w,A}=157.9\, \text{dB}_A$ which corresponds to a sound power level $L_w = 162.8\, \text{dB}$. An 8-blade UDF$^1$/ATP$^2$ source power level spectrum was theoretically constructed using the relative strength of harmonics relative to the fundamental [9] corresponding to the blade passage frequency (BPF):

$$L_{w,N} - L_{w,1} = -33.2 \times \log_{10} N$$

where the “1” denotes the BPF frequency and “N” the $N^{th}$ harmonic number. Table (4.2) provides the theoretical A-weighted spectrum with an associated BPF=200 Hz and corresponding sound power level of 162.8 dB. Both spectrums are shown on Figure (4.18). The results are presented in Figures (4.20) and (4.21) for the winter and summer seasons of year 2011 respectively, and show that there can be as much as 10 dB difference from a location to another for both seasons.

The pure-tone to octave-band correction term in the absorption term was discarded in the absorption loss routine for the propagation of the UDF/ATP spectrum as it is mainly tonal. It is fundamental to note that although the UDF/ATP A-weighted source spectrum is lower than the A-weighted jet source spectrum, the associated noise levels are statistically higher for short sideline distances because low frequencies are less attenuated due to atmospheric absorption.

---

$^1$UnDucted Fan
$^2$Advance TurboProp
Figure 4.18: Jet and UDF spectrum shapes corresponding to the same overall sound power level.

Figure 4.19: NOAA weather stations scattered around the continental United States, from www.maps.google.com. The blue shapes show the locations of the 16 considered weather stations.
Figure 4.20: $L_{A,\text{max}}$ decay with slant-range distance for **summer** average weather conditions in 2011 above 16 different locations within the continental United States, from a jet type source spectrum (left) and a UDF/ATP theoretical source spectrum (right). The black dotted lines represent the maximum (top) and minimum (bottom) predicted levels, the red line shows the mean levels and the green curve shows the logarithmic fit to the scattered results.

Figure 4.21: $L_{A,\text{max}}$ decay with slant-range distance for **winter** average weather conditions in 2011 above 16 different locations within the continental United States, from a jet type source spectrum (left) and a UDF/ATP theoretical source spectrum (right). The black dotted lines represent the maximum (top) and minimum (bottom) predicted levels, the red line shows the mean levels and the green curve shows the logarithmic fit to the scattered results.

The BANOERAC report provides logarithmic fits to observed en route noise levels:

$$L_{A,\text{max}} = A - B \times \log_{10}[d_{\text{SL}}]$$

where $d_{\text{SL}}$ is the slant-range distance expressed in km. Table (4.3) provides the $A$ and $B$ coefficients for the 2011 winter and summer average taking the 16 NOAA
Table 4.3: Logarithmic fit coefficients providing $L_{A,max}$ values as a function of slant-range distance in km for noise from the turbojet and rotor source sound power level through the 2011 winter and summer seasons averaged over 16 different NOAA weather stations.

<table>
<thead>
<tr>
<th>Season</th>
<th>Propulsion</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winter</td>
<td>Jet</td>
<td>67.3</td>
<td>35.4</td>
</tr>
<tr>
<td></td>
<td>Rotor</td>
<td>79.2</td>
<td>44.7</td>
</tr>
<tr>
<td>Summer</td>
<td>Jet</td>
<td>73.5</td>
<td>39.5</td>
</tr>
<tr>
<td></td>
<td>Rotor</td>
<td>85.1</td>
<td>48.5</td>
</tr>
</tbody>
</table>

weather balloon data into account. These expressions provide quick noise level estimates for extreme seasonal conditions.

These fits are grouped on Figure (4.22) and are helpful in providing a first-pass in the comparison between broadband jet and tonal rotor source signatures, assuming omnidirectional directivity:

- UDF/ATP sources could potentially be louder up to slant-range distances of 20 km. Beyond that range, jet sources could be noisier on the ground.

- Most of the seasonal variability could occur close to the flight track, for slant-range distances lower than 20 km.

This chapter has introduced the benefits of using AERNOM to study the propagation of aircraft en route noise and has led to interesting prediction trends. AERNOM can produce noise ground contours from a high-altitude overflight using various inputs: atmospheric data, aircraft Mach number, angle between the wind direction and the flight direction, ground type and the receiver height. Some interesting conclusions are:

- The noise on the ground is generally louder during summer seasons.

- The noise levels are increased along the sideline direction for a crosswind situation.

- The angle between the overflight track and the wind direction impacts the duration of the received noise signal in the time domain.
Figure 4.22: Logarithmic fits for 2011 seasonal averages over 16 different locations within the continental United States, assuming jet and rotor source signatures of the same unweighted overall sound power level.

Chapter 5 will discuss other available numerical methods to propagate noise from a high altitude overflight, in comparison to AERNOM.
Software comparison for en route noise prediction

This chapter focuses on the comparison of available software in the context of aircraft en route noise. A comparative study [54] was conducted regarding the propagation of aircraft en route noise from a high altitude overflight portion, involving FAA’s Integrated Noise Model (INM), Delta¹’s NORD 2000 (N2K) and Penn State University’s Hybrid Propagation Model (HPM). Each of these models were assessed for various test cases involving realistic weather profiles, and their outputs were compared against Penn State University’s Advanced En Route NOise Model (AERNOM) which stands as a reference in the absence of published benchmarks or available software for the case of long range vertical propagation.

In the absence of a specific tool to predict the impact of aircraft en route noise on the ground, it appears interesting to study available software in a different context compared with a situation for which they are meant to be work efficiently. FAA’s INM and Penn State University’s HPM are specifically dedicated to the prediction of airport community noise, and Delta’s NORD 2000 is an engineering noise propagation tool for environmental noise. These models all deal with short-range and primarily horizontal propagation, whereas the propagation of aircraft en route noise calls for a long-range vertical propagation path. The transition from one geometry to another requires the consideration of an altitude-stratified atmosphere as well as other modifications linked with the loss mechanisms. Therefore, the

¹Danish Electronics, Light & Acoustics
available models were adjusted when possible, this will be discussed in section (5.2).

5.1 Presentation of each model

All the models discussed in this chapter are constructed based on various approximations, which are essential to mention before going further with the comparison itself.

5.1.1 Presentation of INM

Developed by the Volpe National Transportation Systems Center, INM [1] has been the standard airport community noise prediction tool used by the FAA since 1978. It is being used in over 50 countries by over 700 organizations. INM is a segmented noise model in which all the aircraft noise sources are integrated into the so-called Noise Power Distance (NPD) curves, which are operational and aircraft dependent. These NPD curves provide the noise received from a particular flight track at a certain range from the source. A spectral class adjustment is applied on the NPD curves for aircraft that have a similar spectral content for the same operation. INM’s noise predictions include:
• A ground effect, different for soft and hard grounds

• Spherical straight line attenuation

• Atmospheric absorption losses referenced to a standard day (the absorption does not vary with height, range, or time)

These last two mechanisms are incorporated in the NPD curves before the computation starts.

5.1.2 Presentation of NORD 2000

NORD 2000 [24] has been developed since 2001 by a joint Nordic project group in the context of environmental short-range horizontal noise propagation (mainly rail and road noise). It is based on a semi-analytical ray formulation that enables one to compute noise levels with two built-in approximations:

1. A linear sound speed profile approximation

2. The effective sound speed approximation

The sound rays are circular arcs in the linear sound speed approximation, which enables one to analytically construct the sound ray paths from the source to the receiver. The effective sound speed approximation has to do with wind effects. In this approximation, these are added to the temperature effects so that the sound speed is the summation of the thermodynamic sound speed with the horizontal component of the wind speed projected in the two-dimensional plane of interest:

\[ c_{\text{eff}}(z) = c(z) + v(z) \times \cos(\phi) \]

As a result, a realistic, moving atmosphere is modelled as an effective, still atmosphere. NORD 2000’s noise predictions include:

• Spherical spreading along the curved rays

• Atmospheric absorption losses for 1/3 Octave Band frequencies corrected from pure-tone to octave band frequencies [ISO method]

• Excess ground attenuation
• Scattering by vegetation
• Diffraction from buildings and/or noise barriers
• Irregular terrain and uneven ground impedance
• Shadow zone predictions, however there is no turbulent scattering
• Atmospheric turbulence

5.1.3 Presentation of HPM

Dr. Joyce Rosenbaum, Dr. Anthony Atchley and Dr. Victor Sparrow from Penn State University have been developing HPM [2] since 2006. HPM is specifically dedicated to airport community noise prediction, in a realistic atmosphere [55]. It is based on the combination of two full wave equation solutions: the Parabolic Equation (PE) and the Fast Field Program (FFP). Each of these numerical methods are handled up to their limits (narrow-angle limitation for the PE, inability to treat range-dependent information for the FFP) and combined in order to compute sound levels. Being based on the full acoustic wave solution, these methods are very accurate. Their shortcoming lies in their use of the effective sound speed approximation as well as their frequency-dependent computation time: when low frequency predictions can take a few seconds, higher frequency runs may take hours. HPM’s noise predictions include:

• Full losses from source to receiver (wave based method)
• Irregular terrain and uneven ground impedance
• Atmospheric turbulence
• Simplistic atmospheric profiles in the effective sound speed approximation

5.2 Software modification for long-range vertical propagation

Figure (5.1) illustrates the differences between short-range horizontal propagation (urban community noise) and long-range vertical propagation (en route noise).
Section (5.2) details the transformations that were applied to NORD 2000 and HPM. INM was not modified as a part of this research.

The meteorology plays an essential role on the loss mechanisms from a high altitude aircraft fly over, and particularly on the atmospheric absorption losses [28]. Generally, the models cited in this report assume constant absorption coefficient values per frequency. For a given frequency, the absorption coefficient depends on atmospheric parameters such as temperature, humidity and ambient pressure. Given that these are a varying function of altitude, realistic absorption coefficients are a function of altitude and therefore it is necessary to account for this altitude dependency for long-range vertical propagation. Thus, altitude dependent absorption coefficients were implemented in NORD 2000 and HPM, which will be discussed in the next paragraphs. To be consistent regarding the absorption losses, the ANSI S1.26 algorithm is considered in NORD 2000, HPM and AERNOM.

5.2.1 Modifications in NORD 2000

NORD 2000’s original code does not account for a stratified atmosphere, so the prediction of absorption losses and excess ground attenuation for en routes noise are not possible with the official version. Consequently, a Matlab code was written, based on the same foundation from which NORD 2000 was built, which incorporates the loss mechanisms from the high altitude source to the near ground receiver: spherical spreading, absorption loss, excess ground attenuation. This initial version, called NORD 2000 “linear fit”, considers a linear fit of the effective sound speed from the ground to the source altitude. The routines for the scattering zones and shielding effects were discarded, as they are not the main focus of this comparison task.

5.2.1.1 Absorption losses

An altitude dependant absorption loss routine was implemented; each ray is subdivided in portions defined by a constant atmospheric width layer as shown in Figure (5.2). The corresponding mean atmospheric layer height defines a frequency-dependent absorption coefficient, which gets multiplied by the total distance travelled along this ray portion to produce the frequency dependent absorption loss per
layer. Summing these losses over all the layers then provides the total absorption loss. That being stated, it is obvious that the atmospheric layer thickness $h_{\text{layer}}$ is critical in regards to the computation time over accuracy ratio. Several values for $h_{\text{layer}}$ were tested for moderate atmospheric refraction conditions. Finer atmospheric layers lead to precise predictions; consequently, a value of 1 meter for $h_{\text{layer}}$ was considered as a reference. The atmospheric thickness varied from 1 meter to 1 km, and the total absorption attenuation at 1 kHz was computed, using the ANSI absorption algorithm [29]. Results are presented in Table (5.1).

The prediction accuracy lies within 0.4 dB up to $h_{\text{layer}}=200$ meters. Above that value, the accuracy degrades, dropping to 2 dB for 1 km thick layers. The computation time for these runs is inversely proportional to the atmospheric layer thickness. For example, the computation for $h_{\text{layer}}=10$ meters takes 20 times longer
than the computation for \( h_{\text{layer}} = 200 \) meters. Based on this, an optimal layer thickness lies in between 100 and 200 meters. Consequently, it is essential to gather weather data in such a stratified atmosphere, tracking the atmospheric parameters at least every 100 to 200 meters from the ground to 10 km.

5.2.1.2 Excess Ground Attenuation [EGA]

The original code assumes a constant sound speed to compute direct and reflected ray arrival times, directly impacting the ground interferences. In a stratified atmosphere however, the sound speed is a function of altitude, so that the travel time varies with altitude for a fixed travelled distance. The layered atmosphere method was again considered in the EGA context: in layer \( i \), the time propagation \( \tau_i \) is calculated using the sound speed at the mean altitude \( z_{\text{mean},i} \) and the distance travelled along the corresponding ray portion \( s_i \) according to

\[
\tau_i = \frac{s_i}{c(z = z_{\text{mean},i})}.
\]

Then, the total travel time \( \tau \) is the summation of all the travel times in each atmospheric layer:

\[
\tau = \sum \tau_i.
\]

This method is used to compute the difference in arrival time between direct and reflected ray. The reflection point is determined analytically using a 3rd degree polynomial [20], this analytical solution being part of the original NORD 2000 code.

5.2.1.3 Refraction Losses [added module]

For propagation in a stratified media, such as the atmosphere, the sound rays curve towards the regions of lower sound speed, and in the context of NORD 2000, to regions of lower effective sound speed. NORD 2000 computes the geometrical loss as a spherical spreading loss along the curved ray. This method is not suitable for long-range propagation as discussed in section (2.2.2); the Lamancusa method was included in NORD 2000’s prediction scheme.

5.2.1.4 Improving the linear sound speed approximation

As mentioned in section (5.1.2), two built-in approximations exist in the formulation on which NORD 2000 is based: the linear sound speed approximation and the effective sound speed approximation. While it is not possible to improve the latter, it is possible to use piecewise linear fitting to successively construct the rays
Figure 5.3: The layered method accurately reproduces a realistic effective sound speed profiles, here in a downwind refraction atmosphere.

from the source to the ground, thus improving the former. Instead of applying a linear fit on the effective sound speed from the source to the ground, the fit is realized in each atmospheric layer. For a realistic effective sound speed, the result is a more accurate reproduction of any realistic effective sound speed, as shown in Figure (5.3) [56].

This refined version of NORD 2000 is referenced as the NORD 2000 “finer” model. Rays are constructed differently in this revised model, as illustrated in Figure (5.4): the initial conditions are linked with the ray launching angle $\theta_{\text{launch}}$ rather than to the receiver range. The computation time rises as a result of the atmosphere layering; the optimal layer width to use is directly dependent upon the effective sound speed profile of interest. More specifically, for night-time boundary layer inversions for example, the atmospheric width must be very small near the inversion (on the order of 1 meter). For typical atmospheric conditions, an atmospheric width of 1 km may be sufficient. Concerning the EGA in this revised version, the method employed was based on the AERNOM implementation as discussed in section (2.2.4); there is no analytical solution to predict the reflected ray launching angle to produce an interaction at the receiver location. Instead, two
closely separated rays are launched, their range on the ground are recorded and compared with the required range, and finally an interpolation method leads to the correct launching angle. The full Matlab code of the refined version of NORD 2000 for illuminated zone predictions is presented in Appendix B.

5.2.2 Modifications in HPM

HPM was originally built to predict airport community noise, in a realistic atmosphere (including turbulence) and over realistic ground (including terrain). Given the different geometry required by en route noise predictions, a few modifications were applied to HPM. This section discusses these changes, as well as the motivation to discard the PE prediction for en route noise and the failure of the FFP to predict short-range noise levels directly beneath the source, along the vertical ray.

5.2.2.1 Meteorological data

HPM only treats simplistic atmospheric profiles, using the logarithmic effective sound speed profile: \( c_{\text{eff}}(z) = c_0 + b \times \ln \left( \frac{z}{z_0} + 1 \right) \), where \( b \) varies from \(-1\), \(0\) to \(1\) for an upwind, a homogeneous and a downwind refracting atmosphere respectively.
The input file and some portions of the code were modified, in order to include temperature, wind speed, humidity and pressure dependence as input data. Consequently, any realistic atmospheric profile can be treated by the revised version of HPM.

5.2.2.2 Absorption losses

Just like NORD 2000, HPM does not account for altitude dependent absorption coefficients. HPM being a wave-based model, the modifications are different. Instead of calling for constant absorption coefficients before the code is run, the revised version of HPM calls for altitude dependent absorption coefficients stored in a vector. The PE and FFP being recursive, the initial propagation step was the only one to be modified.

5.2.2.3 Propagation of a 1/3 Octave Band Spectrum

HPM enables the propagation of a pure tone or a flat spectrum (with the same levels at 1 meter for all octave band frequencies). Modifications were made in order to propagate a 1/3 octave-band spectrum, which is the standard for aircraft noise source definitions.

5.2.2.4 FFP method for en route overflight

For an en route overflight situation characterized by a high altitude source, the FFP output covers a larger ground surface area compared to the PE output. This originates from the angle limitation of the PE, as presented in Figure (5.5). Furthermore, one FFP run provides the output at all ranges for a single receiver height, which makes the method a much faster one compared to the PE for long-range noise propagation. Consequently, only FFP outputs from HPM will be considered. However, due to very large computation times especially at high frequencies, the corresponding ground contour will be restricted in space so that the runs end at a range of 40 km from the source, even in the downwind plane where rays can hit the ground at larger ranges.
Figure 5.5: For an en route overflight, the major noise contribution hitting the ground is propagated via the FFP method along the flight track. Hence, HPM was modified to only use the FFP module for en route noise prediction.

### 5.2.2.5 FFP grid spacing for en route noise

HPM matches the PE and FFP outputs on the same grid so that the FFP grid spacing is the same as the PE grid spacing in the horizontal and vertical direction, determined by $\frac{\lambda}{10}$ where $\lambda$ is the wavelength of the acoustic wave. The FFP method itself does not generally require such a fine grid. Indeed, the grid spacing in the vertical direction $\Delta z$ needs to ensure a smooth transition of the effective sound speed with height, as the FFP formulation is based on a constant wave number approximation per atmospheric layer. The horizontal grid spacing $\Delta r$ has to satisfy conditions linked with the Fourier Transforms, such as $\Delta r = \frac{\lambda}{2}$ to avoid aliasing. Therefore, a safe value for the horizontal grid spacing is $\Delta r = \frac{\lambda}{3}$; the vertical grid spacing values depend on the effective sound speed gradients: thick layers are generally acceptable for slowly changing gradients whereas thinner layers are needed where the gradients change quickly. For en route propagation $\Delta r$ was set to $\frac{\lambda}{3}$ and $\Delta z$ was varied from 10 to 200 meters. This is much less restrictive than in the original HPM implementation, and greatly speeds the computation.
5.2.2.6 Effects of density and wind in normal velocity boundary conditions (FFP)

For low altitude propagation, the original focus of HPM, the density and wind speed variations from layer to layer are neglected in the FFP formulation. This is not the case for en route noise propagation, so these effects were included in the revised version of HPM following work carried out by Salomons [32].

5.2.2.7 Benchmarking the FFP for en route geometry and modification of FFT window

The revised FFP was re-benchmarked against Attenborough’s benchmark test cases [40], then against the analytical solution for a stable homogeneous atmosphere over a hard ground. As a source gets higher, the range from which the FFP output matches the benchmark increases, thus emphasizing the fact that the FFP code as it is written in HPM cannot predict the noise emitted at steep angles from a high altitude source. As a final step in the FFP solution, a Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) are applied to transfer the solution from the wavenumber domain to the spatial domain. In the original HPM implementation, a tapered window is utilized to eliminate the small, rapid oscillations that are due to the truncation of the integration interval at the maximum wave number [32]. The short-range predictions can be improved for a high-altitude source by removing this window. Figure (5.6) illustrates this observed phenomenon: FFP Transmission Loss (TL) values are compared with the analytical solution in a homogeneous atmosphere over a hard ground. The short-range predictions are wiggly and the first predicted value is higher than expected by 10 to 15 dB for all tested frequencies. Given that the red curves provide better short-range predictions, the window was removed in the en route version of the FFP code in HPM. The full Matlab code of the modified version of HPM is presented in Appendix B.
Figure 5.6: For en route geometry, the FFP cannot predict the Transmission Loss at short ranges corresponding to steep launching angles unless the window is removed before applying the FFT and IFFT. Left plot: TL for ranges up to 30 km for a pure-tone frequency $f=250$ Hz. Right plot: zoom for short ranges. Removing the window leads to accurate predictions except for a short range with over predictions.

5.3 Software comparison setup

Comparing INM, NORD 2000, and HPM requires the consideration of the same noise source (input) to propagate through each model, as well as a consistent noise metric to quantify the noise ground contours (output).

5.3.1 NPD curves retrofitting

In order to be consistent in the comparative study, a 1/3 octave-band spectrum was retrofitted from an INM NPD curve representing a B777-300 with departure type settings (80,000 pounds of thrust), thus enabling the representation of the worst case scenario in the absence of high altitude aircraft source modelling capabilities. This reverse engineering method has been utilized in the context of the Imagine project [57] for which a specific Excel spreadsheet [58] was created. Its goal is to estimate by trial and error, an A-weighted sound power level spectrum that takes the noise source directivity into account and approaches the NPD Sound Exposure Level (SEL) at the ten slant-range distances (200, 400, 630, 1000, 2000, 4000, 6300, 10000, 16000, and 25000 feet). The result is a 1/3 octave-band spectrum that is very close to the original NPD curve, once theoretically constructed using
the standard NPD construction methodology. The following steps summarize the retrofitting process:

1. After having specified the NPD data [dB], the spectral class content is adjusted for A-weighting and atmospheric absorption.

2. The aircraft-type dependent directivity is applied to estimate frequency and angle-dependent noise power levels.

3. For each of the ten slant-distances associated with different launching angles, the sound pressure levels are computed using geometrical spreading and SAE absorption losses. A logarithmic summation process is then used to calculate the overall sound pressure level which depends on the slant distance and the propagation angle.

4. The overflight time duration $\tau$ is computed then the sound pressure levels are converted to SEL values for each distance using $SEL = L_{eq} + 10 \times \log_{10} \tau$.

5. The original SEL values and compared and eventually manually re-adjusted to match the output, by including a scaling factor prior to step 2.

6. Once the SEL values approach the NPD values for all distances, a sound power level spectrum is constructed by using equation (5.1), where $\text{constant}_{\text{shift}}$ is the manually adjusted scaling factor added prior to step 2 to closer match the targetted original NPD data.

$$L_w(f) = L_{w,\text{estimated}}(f) + \text{constant}_{\text{shift}} - A_{\text{weighting}}(f) + EGA + 11 + 10 \times \log_{10} N_{\text{engines}}$$

(5.1)

The retrofitted NPD data were used as input data for NORD 2000 and AER-NOM up to 2 kHz (assuming the atmosphere acts as a low-pass filter for higher frequencies) and up to 1 kHz as an input to the FFP for computation time reasons. The characteristics of the input data for all models are shown in Table (5.2) and Table (5.3).

### 5.3.2 Noise output

As presented in Chapter 4, the Sound Exposure Level (SEL) noise metric was chosen to reproduce the noise propagated from the high altitude overflight portion
<table>
<thead>
<tr>
<th>Slant distance [ft]</th>
<th>200</th>
<th>400</th>
<th>600</th>
<th>1000</th>
<th>2000</th>
<th>4000</th>
<th>6000</th>
<th>10000</th>
<th>16000</th>
<th>25000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEL [dB]</td>
<td>111.6</td>
<td>108.6</td>
<td>105.4</td>
<td>102.7</td>
<td>98.7</td>
<td>92.7</td>
<td>88.7</td>
<td>84.2</td>
<td>78.7</td>
<td>72</td>
</tr>
</tbody>
</table>

Table 5.2: NPD data for a B777-300 (departure settings) from [59].

<table>
<thead>
<tr>
<th>$f_{1/3}$ [Hz]</th>
<th>50</th>
<th>63</th>
<th>80</th>
<th>100</th>
<th>125</th>
<th>160</th>
<th>200</th>
<th>250</th>
<th>315</th>
<th>400</th>
<th>500</th>
<th>630</th>
<th>800</th>
<th>1000</th>
<th>1250</th>
<th>1600</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL [dB]</td>
<td>116</td>
<td>121</td>
<td>130</td>
<td>134</td>
<td>136</td>
<td>140</td>
<td>141</td>
<td>145</td>
<td>147</td>
<td>147</td>
<td>148</td>
<td>148</td>
<td>149</td>
<td>150</td>
<td>130</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.3: 1/3 octave-band spectrum retrofitted from NPD data shown in Table (5.2). The resulting spectrum is represented in Figure (5.7).

5.3.3 General setup

In terms of geometry, the settings are the following:

- A 140 km overflight track at a constant altitude of 10,000 meters (32,808 feet)
- Flat and even terrain [Grassland: flow resistivity $\sigma$ of 300 cgs Rayls]

![B777-300 1/3 Octave band spectrum](image)

Figure 5.7: B777-300 1/3 octave-band spectrum retrofitted from the NPD data, with the corresponding octave bands considered in each model.
• Receivers located 1.2 meter above the ground (4 feet), separated 500 meters one from each other

• A ground grid of $80 \times 80$ km with one axis aligned with the flight track, levels associated in $\text{dB}_A$

• 1 degree increment in the polar ray launching angle per 2D-plane (AERNOM and NORD 2000 “finer”)

• $\pi/4$ increment in the azimuthal angle around the vertical $Z$ axis

In terms of meteorological profiles, the settings are the following:

• Realistic temperature, pressure and humidity averaged profiles from [30]

• Linear wind speed profile of moderate strength $[w(z) = 3 \times z, z \text{ in km}]$, not applicable in INM (because INM assumes constant headwind only)

• Constant wind direction with altitude, taken in the positive $X$-axis

• The effects of turbulence are discarded

In section (5.4), a sound pressure level decay in $\text{dB}_A$ from both NORD 2000 versions, AERNOM and HPM, then the loss contributions predicted by the ray-based models will be presented in the downwind refracting 2D-plane (along the positive $X$ axis), then section (5.5) will present a SEL ground contour for each model.

5.4 AERNOM vs. NORD 2000 vs. HPM (FFP)

This section focuses on the comparison of AERNOM with both versions of NORD 2000, in particular on the improvements of the NORD 2000 “finer” method.

5.4.1 Comparison of sound pressure level decay in downwind plane

A point source sound pressure level decay in $\text{dB}_A$ was extracted from AERNOM, NORD 2000 “linear”, NORD 2000 “finer” and HPM (FFP only) and is shown on Figure (5.9) which illustrates the following:
Figure 5.8: Averaged realistic weather profiles [30] (temperature, humidity and pressure) used in this chapter. The wind speed profile is assumed linear of moderate strength.

- As expected, the predictions of HPM (FFP only) are inaccurate at short ranges. The FFP overpredicts the levels right beneath the source.

- Breaking down the atmosphere into layers in which the effective sound speed is linear (thus improving the linear effective sound speed approximation built in NORD 2000, leading to NORD 2000 “finer”) proves to be efficient since it reduces the prediction departure observed with NORD 2000 “linear”, generally keeping the predictions within 2 dB of AERNOM.

- However, the inability of NORD 2000 “finer” to consider wind effects separately leads to erroneous predictions of shadow zone locations in upwind
Figure 5.9: Sound Pressure Level [dB_A] decay in the downwind plane for AERNOM, both NORD 2000 models and HPM (FFP). Propagation stopped at $\theta_{\text{launch}} = 0$ degrees for the ray based models.

refracting planes [56].

- Generally speaking, all models provide the same predictions within a few dB.

5.4.2 Comparison of loss mechanisms contribution in AERNOM, NORD 2000 “linear” and NORD 2000 “finer”

It is possible in ray-based models to extract the contribution of each loss mechanism (excess ground attenuation, refraction and geometrical losses, absorption losses), thus enabling to track the origin of the departure of the NORD 2000 “linear” model predictions.

Geometrical (+ refraction) losses: Figure (5.10) shows that the geometrical and refraction losses are fairly well predicted by both NORD 2000 models, compared with AERNOM’s predictions, even at large ranges.
Figure 5.10: Comparison of geometrical losses for AERNOM and both NORD 2000 models.

Figure 5.11: Comparison of absorption losses for AERNOM and both NORD 2000 models. Top figure: $f=125$ Hz. Bottom figure: $f=1000$ Hz.

Absorption losses: Figure (5.11) emphasizes the limitation of both NORD 2000 models when it comes to consider wind effects (effective sound speed approximation). The ray refraction is different, which therefore predicts different ray trajectories and consequently different attenuation levels, which is especially visible for higher frequencies. Nevertheless, the differences between both NORD 2000 models are minor.

Excess Ground Attenuation: Figure (5.12) strongly illustrates the benefit of
using NORD 2000 “finer”, which closely reproduces the excess ground attenuation shape, the minor differences with AERNOM being due to the moving media effects. NORD 2000 “linear” cannot accurately predict the ground effects, which significantly affects the overall sound pressure level. For example, at 125 Hz around 11 km from the source, the attenuation due to the ground is about 20 dB, which is not accounted for in NORD 2000 “linear”.

### 5.5 En Route Noise prediction comparison

This section presents the SEL ground contours obtained with each of the software presented in section (5.1).

#### 5.5.1 Overflight En Route Noise prediction using INM

First, an overflight track was constructed from a reference altitude of 0 meters. The case and scenario set-up are standard. The operations on this flight track are:

- Altitude: 10,000 meters
- Indicated Air Speed [IAS]: 520 km/h
Figure 5.13: SEL ground contour using INM. Levels in dB_A. CPU time: 0.5 min.

- Number of flights per day: 1
- Thrust: 800,000 pounds (departure type setting for worst case scenario)

The noise was then propagated through a standard atmosphere (temperature: 15° C, pressure: 101.325 kPa) on a manually constructed grid. The ground noise values were then exported to Matlab for graphical processing.

5.5.2 Overflight En Route Noise prediction using NORD 2000 “linear”

A linear fit is applied to the effective sound speed profile in all of the 2D-planes in order to determine $g$ and $c_0$, these parameters being the basis of the analytical development of NORD 2000. After rotation around the Z axis, a pseudo-3D ground contour is generated.
Figure 5.14: SEL ground contour using the NORD 2000 “linear fit” model. Levels in dB$_{A}$. CPU time: 4.2 min.

5.5.3 Overflight En Route Noise prediction using NORD 2000 “finer”

A piecewise linear fit is applied to the effective sound speed profile in all of the 2D-planes in order to successively construct the ray trajectories. After rotation around the Z axis, a pseudo-3D ground contour is generated. Various atmospheric thickness layers were tested. Generally, thinner layers are preferred if the effective sound speed gradients are large. Furthermore, the choice of $h_{layer}$ directly impacts the computation time. For a moderate wind speed profile, the effective sound speed gradients being moderate, the atmospheric layer thickness was set to 5 km (2 layers).
5.5.4 Overflight En Route Noise prediction using HPM (FFP)

No frequencies above 1 kHz were propagated due to the very large computation time involved. Only the FFP portion of the HPM was used to propagate the spectrum.

5.5.5 Overflight En Route Noise prediction using AERNOM

As for both NORD 2000 models, the sound pressure levels on the ground are computed in each plane around the Z axis, then a pseudo-3D contour is generated.
Figure 5.16: SEL ground contour using the FFP from HPM. Levels in dBA. CPU time: 44 minutes.

Figure 5.17: SEL ground contour AERNOM. Levels in dBA. CPU time: 639 s.
Comparison discussion

5.6.1 Comparison of SEL decay with sideline distance

Figure (5.18) shows the SEL decay with sideline distance from the runs presented in section (5.5), which provides another perspective at the software comparison discussion.

5.6.2 AERNOM is suitable for fast and accurate predictions

The main conclusion is that Penn State’s AERNOM is the most accurate solution in the context of en route noise propagation; it distinguishes the wind speed effects from the temperature effects in terms of ray refraction (which none of the other models considered can do) and runs as fast as semi-analytical models such as NORD 2000. Comparisons against experimental data would be helpful to prove
the realistic efficiency of AERNOM.

5.6.3 HPM is suitable for short-range propagation

Penn State’s HPM is very accurate for short-range propagation, and has the ability to incorporate terrain and turbulence effects. However, for long-range vertical propagation, the PE is not user-friendly mainly because of the grid size restrictions, and the FFP fails to predict the levels right beneath the flight track. The FFP in HPM was modified for en route noise propagation. Generally, for short-ranges, the FFP underpredicts AERNOM’s outputs and at larger ranges, it over predicts AERNOM’s outputs.

5.6.4 NORD 2000, an engineering tool with possible use for en route noise propagation

The original NORD 2000 model was modified in order to replace the unrealistic effective sound speed linear fit between the en route source altitude and the ground by a finer semi-analytical model that successively constructs the ray paths through layers in which the effective sound speed is approximated by constant values. This improvement enables NORD 2000 to predict sound pressure levels within 2 dB generally of AERNOM’s outputs, which makes it a reasonable prediction tool. However, the effective sound speed approximation built in NORD 2000 leads to erroneous predictions of the illuminated zone boundaries. Nevertheless, this issue is not essential to predict the noise propagated from an overflight, so that the revised version of NORD 2000 “finer” is a reliable en route ground noise contour prediction tool, assuming an omnidirectional source.

5.6.5 Improving the NPD curves in INM for en route noise propagation

As it is handled for airport community noise, INM cannot be used to predict en route noise ground contours. One of the main technical limitations built in INM concerns the NPD curves, which were mainly introduced for low altitude horizontal propagation predictions. The atmospheric absorption coefficients do not
vary close to the ground, and a constant value per frequency is acceptable. For long-range vertical propagation, it is necessary to include the altitude dependency of the absorption coefficients, which originates from the altitude dependency of our atmosphere. This will be discussed in the next section.

5.7 Improving the NPD data for long-range vertical propagation

5.7.1 Review of the NPD curves construction principles

The goal of this section is to summarize the methodology that governs the construction of the NPD curves, in order to stress the potential improvements for use in long-range vertical propagation predictions.

NPD curves are based on experimental data, and provide the noise levels expressed as a function of engine power and distance, and are corrected for standard atmospheric absorption \((T = 25^\circ C, \text{relative humidity } rh = 70\%\) using the SAE standard) among other mechanisms as shown in Equation (5.2).

\[
\begin{align*}
L_p(f, d) & \approx \alpha_{SAE}(f, T_0, rh_0) \times d - \Delta L_{other} \\
NPD(d) & = 10 \times \log_{10} \left( \sum_i 10^{\frac{L_p(f_i, d)}{10}} \right) + 10 \times \log_{10} \tau
\end{align*}
\]

Equation (5.2)

Forsyth et al. [60] have compared various atmospheric absorption algorithms to construct NPD curves, but they considered constant absorption coefficient values, even for high altitude flyovers. Others [61] have tackled the limitations of using current NPD data for altitudes greater than certification distances. The purpose of the next section is to introduce a potential method to take into account the altitude-dependency of atmospheric absorption for vertical propagation.

5.7.2 Including an atmospheric-dependant absorption loss

One possible way of improving NPD curves for en route noise propagation is by removing the built-in absorption losses taken in a homogeneous atmosphere, then
add altitude-dependent absorption losses, assuming straight-line propagation as a first approximation. The NPD data being associated with a spectral class representing a specific signal spectral shape normalized at 70 dB at 1 kHz measured 305 m from the source, the first step is to translate the spectrum to the source position. The transition to vertical propagation in a stratified atmosphere is then conducted by gradually increasing the virtual source altitude, computing the cumulative losses along the vertical ray [28] (assuming straight line propagation) and applying an appropriate pure-tone to octave-band correction factor [34] for every 1/3 octave-band frequency of the spectral class data. The spreading and absorption losses are then removed from the spectral shape for both cases (SAE coefficients in a homogeneous atmosphere: $T = 25^\circ$, $rh = 70\%$ and ANSI S1.26 absorption coefficients in a stratified medium). Finally, the 1/3-octave band levels are logarithmically added, and the range-dependent overall resulting algebraic levels are added to the original NPD data to constitute the refined NPD data. This is an efficient way of comparing the absorption losses only. Other parameters such as the lateral attenuation or the over-flight duration remain unchanged.

### 5.7.3 Refinement of B777-300 NPD data in stratified media

This section presents the specific example of the B777-300 NPD refinement process for the Bass & Sutherland atmospheric model [30]:

1. The spectral class data (id reference 105) is A-weighted and translated to realistic values, assuming a source power level at 1 kHz $L_{w,1kHz} = 145$ dB. This value is arbitrary. In practice, a value of 150 dB will provide the same conclusions.

2. For each slant distance, the spreading and attenuation losses are added to the spectrum 1/3 octave-band levels: first, the SAE constant absorption coefficients taken at standard atmospheric conditions are considered, then cumulative losses are computed along a vertical propagation ray, using the ANSI absorption algorithm and a pure-tone to 1/3 octave-band correction term. Tables (5.4) and (5.5) provide some comparison results.

3. Overall levels are then produced at each slant distance and the differences
Table 5.4: Difference in predicted absorption losses at 1 kHz as a function of vertical distance: $\Delta L_{\text{abs}} = \delta L_{\text{SAE}} - \delta L_{\text{ANSI}}$ [dB]. The pure-tone to 1/3 octave-band correction attenuates the absorption losses, see section (2.2.3.4).

<table>
<thead>
<tr>
<th>Distance [feet]</th>
<th>200</th>
<th>400</th>
<th>630</th>
<th>1000</th>
<th>2000</th>
<th>4000</th>
<th>6300</th>
<th>10000</th>
<th>16000</th>
<th>25000</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta L_{\text{abs}}$ [dB]</td>
<td>0.1</td>
<td>0.2</td>
<td>0.4</td>
<td>0.6</td>
<td>1.2</td>
<td>2.7</td>
<td>4.7</td>
<td>8.2</td>
<td>8.8</td>
<td>5.1</td>
</tr>
</tbody>
</table>

Table 5.5: Difference in overall sound pressure level: $\Delta L = \delta L_{\text{overall,SAE}} - \delta L_{\text{overall,ANSI}}$ [dB].

<table>
<thead>
<tr>
<th>Distance [feet]</th>
<th>200</th>
<th>400</th>
<th>630</th>
<th>1000</th>
<th>2000</th>
<th>4000</th>
<th>6300</th>
<th>10000</th>
<th>16000</th>
<th>25000</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta L$ [dB]</td>
<td>0.2</td>
<td>0.4</td>
<td>0.6</td>
<td>0.8</td>
<td>1.3</td>
<td>1.8</td>
<td>2.3</td>
<td>3.1</td>
<td>3.8</td>
<td>2.2</td>
</tr>
</tbody>
</table>

between both methods are stored in order to later be added to the original NPD data.

The results of this process are illustrated in Figure (5.19).

Figure 5.19: Comparison of original with the refined NPD data corresponding to the B777-300 [departure settings, 80000 thrust] for a specific atmospheric model.
5.7.4 Influence of the atmospheric model on the refined NPD data

The following section illustrates the impact of the atmosphere on absorption losses along a vertical ray, directly impacting the refined NPD data. In particular, seasonal variability will be addressed. Humidity plays an essential role on atmospheric absorption as discussed in section (2.2.3.3). The modified NPD curves should therefore include, when possible, any weather variability. Figure (5.20) shows the seasonal variability of NPD curves for Pittsburgh, PA.

5.7.5 Extension to alternate spectral classes

The previous sections introduced a potential method to refine NPD curves for long-range vertical propagation for a specific spectral class. INM V7.0 includes a total of 72 distinct spectral classes, none of them representing a high altitude stable flyover. The goal of this section is to present some results linked with the consideration of alternate spectral classes. The absorption losses being frequency-dependent, two different spectral class aircraft should emphasize variable modified
Table 5.6: Spectral content for spectral classes 104 and 105, normalized at 70 dB at 1 kHz and corrected for SAE absorption losses at 305 meters from the source.

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>50</th>
<th>63</th>
<th>80</th>
<th>100</th>
<th>125</th>
<th>160</th>
<th>200</th>
<th>250</th>
<th>315</th>
<th>400</th>
<th>500</th>
<th>630</th>
<th>800</th>
<th>1250</th>
<th>1600</th>
<th>2000</th>
<th>3150</th>
<th>4000</th>
<th>6300</th>
<th>8000</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectral class 104</td>
<td>57.3</td>
<td>56.3</td>
<td>61.5</td>
<td>67.7</td>
<td>71.4</td>
<td>73.7</td>
<td>67</td>
<td>72.1</td>
<td>73.8</td>
<td>74.1</td>
<td>71.3</td>
<td>70.4</td>
<td>70.9</td>
<td>70</td>
<td>60.2</td>
<td>67.1</td>
<td>63.4</td>
<td>60.9</td>
<td>56.6</td>
<td>53.2</td>
<td>47.8</td>
</tr>
<tr>
<td>Spectral class 105</td>
<td>60.5</td>
<td>60.4</td>
<td>62.1</td>
<td>63</td>
<td>70.2</td>
<td>78.2</td>
<td>31.5</td>
<td>76.7</td>
<td>74.4</td>
<td>74.6</td>
<td>72.3</td>
<td>71.8</td>
<td>71.1</td>
<td>70</td>
<td>69</td>
<td>68.8</td>
<td>67</td>
<td>65.3</td>
<td>63.4</td>
<td>59.2</td>
<td>53.8</td>
</tr>
</tbody>
</table>

Figure 5.21: Comparison of the original with the refined NPD data for two distinct spectral classes with the Bass & Sutherland atmospheric model [30]. The shape of the NPD curve translation is similar, however the incremental values are different depending on the spectral class.

The spectral classes associated with the MD83 (spectral class id 104) and the B777-300 (spectral class id 105) for departure settings were considered and are presented in Table (5.6). Figure (5.21) illustrates the effects of refining two distinct spectral classes: the process is spectral class dependent. For each spectral class, the variation between the original and the refined NPD data is indeed different. These results highlight the necessity to consider INM spectral class data in the computation process of refining NPD curves for en route noise propagation.
5.7.6 Extension to larger slant distances and en route noise propagation

The shortest distance from an en route flyover is 35,000 feet, which is much larger than the last slant distance NPD value, set to 25,000 feet. For greater propagation distances, INM uses a logarithmic interpolation method that takes the 15,000 and 25,000 feet noise metric values into account. This is valid for vertical propagation providing the atmosphere is homogeneous, which is unrealistic.

In an inhomogeneous atmosphere, it is possible to expand the NPD curves to 35,000 feet by computing the additional spreading and absorption losses along the vertical ray from the 25,000 feet sound metric value. Then, larger slant distances for a 35,000 feet overflight represent receiver positions that are characterized by non-zero sideline distances and for such positions, straight-line propagation occurs at an angle with respect to the horizontal, as shown in Figure (4.10). The additional spreading and absorption losses along the straight lines are added from the 35,000 feet interpolation noise metric value to construct sideline receiver noise metric values.

**Example: extension of the B777-300 NPD curves to en route slant distances**

The following steps show how to extend available NPD curves to larger slant distances, after having refined the data to take the altitude-stratified atmosphere into account:

1. Translation of the 25,000 feet sound metric to 35,000 feet along the vertical ray: the additional spreading loss is computed using \( \delta L_{\text{spreading}} = 20 \times \log_{10} \frac{35}{25} \approx 2.9 \text{ dB} \). Then, the extra absorption loss is calculated for each frequency. These losses are added per 1/3 octave-band frequency to the 25,000 feet spectrum, which leads to an overall sound pressure level (SPL) value after logarithmic summation.

2. For the SEL metric, the difference in overall SPL between 25,000 and 35,000 feet is added to the 25,000 feet SEL to compute the 35,000 feet SEL value. For the B777-300 NPD data propagated through the Bass & Sutherland atmosphere: \( SPL\_{25,000}=58.3 \text{ dB}_A \) and \( SPL\_{35,000}=54.1 \text{ dB}_A \). Since \( SEL\_{25,000}=72 \text{ dB}_A \), the SEL for 35,000 feet is calculated as:

\[
SEL\_{35,000} = SEL\_{25,000} + \delta L_{\text{spreading}}
\]

\[
SEL\_{35,000} = 72 + 2.9 = 74.9 \text{ dB}_A
\]

Thus, the SEL for a 35,000 feet overflight is 74.9 dB(A).
Vertical propagation is considered up to 35,000 feet. For larger slant distances, up to 135,000 feet, the propagation takes place along a straight line that reaches the ground with a certain elevation angle. The corresponding range is called the sideline distance. In this context, a 135,000 feet slant distance leads to a 40 km sideline distance which roughly represents the typical en route noise footprint on the ground.

<table>
<thead>
<tr>
<th>Distance [ft]</th>
<th>200</th>
<th>400</th>
<th>600</th>
<th>800</th>
<th>1000</th>
<th>2000</th>
<th>4000</th>
<th>6000</th>
<th>10000</th>
<th>16000</th>
<th>25000</th>
<th>35000</th>
<th>50000</th>
<th>75000</th>
<th>100000</th>
<th>135000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sideline distance [km]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>10.9</td>
<td>20.2</td>
<td>28.6</td>
<td>39.7</td>
<td></td>
</tr>
<tr>
<td>Original NPD [SEL]</td>
<td>112</td>
<td>108</td>
<td>105.4</td>
<td>102.7</td>
<td>98</td>
<td>92.7</td>
<td>88.7</td>
<td>84.2</td>
<td>78.7</td>
<td>72</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Refined NPD [SEL]</td>
<td>111.1</td>
<td>108.4</td>
<td>106.0</td>
<td>103.5</td>
<td>99.3</td>
<td>94.5</td>
<td>91.0</td>
<td>87.3</td>
<td>82.5</td>
<td>74.2</td>
<td>70.0</td>
<td>63.0</td>
<td>44.9</td>
<td>11.6</td>
<td>-44.2</td>
<td></td>
</tr>
</tbody>
</table>

For a 35,000 feet overflight, $SEL_{35,000} = 68 \text{ dBA}$ in an approximation that neglects the difference in overflight duration from an altitude to another. This approximation in fact underestimates the SEL levels, because higher-altitude overflights are characterized by longer durations for a fixed receiver. For other metrics such as the $L_{A,\text{max}}$ metric there are no approximations in using this method.

3. Considering a maximum sideline distance of 40 km for en route noise, the corresponding slant distance is approximately equal to 135,000 feet. This would be the maximum slant distance considered for en route noise adapted NPD curves. For slant distances up to 135,000 feet, the extra cumulative absorption and spreading losses are computed along straight lines that arrive at the ground with a specific angle $\beta$. Then, the band levels are logarithmically added to form an overall sound pressure level (SPL), which is compared with the 35,000 feet SPL. For each slant distance, the difference is subtracted from the 35,000 feet noise metric value. The additional slant distances that were considered throughout this study are: 50,000 ; 75,000 ; 100,000 and 135,000 [feet]. To get the noise metric value in between these data points, a logarithmic interpolation method can be applied.

This outline was applied to the B777-300 original NPD curves with the Bass & Sutherland atmospheric model. Results are shown in Table (5.7).
5.7.7 En route NPD data: comparison of different spectral classes & weather data

This section will use the extension of available NPD data to en route slant distances to discuss the comparison of different spectral classes for the same atmospheric model, and to assess the effects of including altitude-dependent weather in the construction of NPD data.

5.7.7.1 Comparison of the extension process for various spectral class aircraft

A total of 9 mid-range to long-range aircrafts associated with 5 distinct departure type spectral classes were used to construct refined NPD curves for en route slant distances with the Bass & Sutherland atmospheric model.

- DC10-10: spectral class 101
- A330: spectral class 102
- B757-300, B767-300 and MD11-GE: spectral class 103
- B777-300: spectral class 105
- A340, B707 and B747-100: spectral class 107

In the absence of $L_{A,\text{max}}$ data for some of the aircraft, the comparison uses SEL data for departure settings at maximum available power level. The results are provided on Figure (5.22).

5.7.7.2 Comparison of the extension process for various atmospheric conditions

The B777-300 NPD data for departure settings and 80,000 thrust of power is refined for en route noise in this section, using different seasonal atmospheric data measured above Pittsburgh, PA and Denver, CO in 2010. The results are shown in Figure (5.23) and emphasize the importance of including location specific atmospheric data in the generation of NPD curves, the observed variability being mainly due to variation in the atmospheric water vapor content.
Figure 5.22: Refined NPD data for 9 different aircrafts associated with 5 distinct spectral classes and various power settings. Distances up to 10,000 meters represent SEL values along the vertical ray for a gradually increasing overflight altitude. Larger distances correspond to non-zero sideline distances for a 10,000 meter overflight. Legend: [aircraft reference; kpounds of thrust]; Blue: spectral class 101; Black: spectral class 102; Red: spectral class 103; Brown: spectral class 105; Green: spectral class 107.

Figure 5.23: Refined NPD data computed from the B777-300 NPD data, with departure settings and 80,000 pounds of thrust, for various atmospheric models. The linear decrease in the logarithmic scale as the sideline distance increases and well as the higher SEL levels for summer atmospheric conditions are essential observations.
5.7.8 Summary of NPD refinement process for en route noise propagation and extension to future work

This section introduced a potential improvement of NPD curves for use in en route noise propagation using INM or AEDT, which stands as a first step in these software to predict aircraft en route noise. Location specific weather atmospheric and spectral class characteristics are sufficient to refine the original NPD data for use with an inhomogeneous atmosphere for vertical propagation. However, these modifications neglect the atmospheric refraction since they assume straight-line propagation. One could consider trying to include curved ray propagation in the NPD database, but the database would grow to an unmanageable size. Instead, the best solution would be to modify the propagation algorithms in INM/AEDT to include curved rays directly. Indeed, the propagation algorithms built-in INM/AEDT do not account for ray refraction. The wind direction, for example, plays an essential role in the shape of the noise contours, as highlighted by Figure (5.24). If the wind direction and the overflight track are aligned, then the ground contour is symmetrical with respect to this common direction; however, for a crosswind situation the noise contour is shifted in the direction corresponding to the wind direction (perpendicular to the overflight track). This is not currently handled by INM/AEDT, which shows that there is room for further improvements.
Figure 5.24: Ground contours [dB$_A$] from an enroute overflight, highlighting the effect of wind direction on the general shape of the contour. On the left, the wind direction is in the $+X$ direction (head or tailwind in the absence of moving source effects). On the right, the wind direction is in the $+Y$ direction (crosswind). INM does not currently handle the effects of the angle between the flight track and the wind direction.
Chapter 6

Conclusions

6.1 Research summary

This thesis has introduced alternate numerical methods to predict long-range vertical noise propagation. Most notably, a numerical ray-tracing method was fully implemented and tested in order to predict the noise impact on the ground from a high altitude overflight, using weather balloon data under the frozen atmosphere approximation. Other existing software such as NORD 2000 and the Hybrid Propagation Model were further modified to handle long-range vertical propagation geometries.

AERNOM which stands as “Advanced En Route NOise Model” is the only method in this thesis that considers the wind and temperature effects on refraction as distinct mechanisms and is consequently the most reliable tool to compute aircraft en route noise ground contours. In particular, AERNOM reliably tracks the transition between the illuminated and shadow zones where the received noise levels dramatically drop.

6.2 General trends for the impact of aircraft en route noise

This thesis emphasized various noise prediction trends as far as seasonal variation, showing that summer seasons generally lead to higher noise levels on the ground
due to the excess of water vapor in the atmosphere compared with winter seasons. Additionally the impacts from a rotor and jet type aircraft of same overall sound power level were assessed: the low frequency energy inherently part of future unducted fan propulsion source signatures could significantly impact on slant-range distance noise levels, up to 20 km.

6.3 Some answers to the initial questions

One of the initial goals of this research was to explain and to predict the observed day-to-day ground noise level variability due to en route overflights and highlighted in past publications. Three possible factors are: the presence of insects during the measurements, the beat frequencies produced by two rotors with slightly different rotational speeds, and atmospheric effects. While the two former facts are explicitly mentioned in the literature, the latter has yet to be explored. Randomizing the weather balloon data led to a maximum 3 dB departure and could therefore constitute a plausible cause of noise variability. The frozen atmosphere approximation might not be reliable for use in long-range vertical prediction schemes, and this calls for future research.

6.4 The need for future research and data collection

Future research on en route noise propagation could potentially prove to be extremely useful. First, the effects of high altitude atmospheric conditions (low temperature and density) and of cruising conditions on the radiated noise are unknown to date. This thesis has considered ground level measurements as input data, and does not attempt to provide realistic predictions. Only a coupled en route at altitude and ground level measurement campaign associated with a GPS tracking system and detailed atmospheric parameters would help in benchmarking the outputs of AERNOM or other en route prediction approaches. These two gaps should stand as top priorities for future research on en route noise propagation.
Appendix A

NOAA meteorology and AERNOM code

This appendix presents the Advanced En Route NOise Model (AERNOM) Matlab code. All the routines used in the main code are presented in published Matlab form. Each routine contained in the main code is presented in the order of appearance. First, the NOAA balloon data treatment code is shown, then the complete AERNOM code is given to predict ground contours from a high altitude overflight.

A.1 From the NOAA balloon data to the production of meteorological polynomial fits

This section provides the code to produce meteorological polynomial fits from a NOAA “.dat” file representing the atmospheric data at a particular weather station.

A.1.1 Initial process

Contents

- Open and read the file
- Extract and store the data
- Loop through the data for the "rlevels" lines
- Save data in Matlab format
% February 2010: code that reads through the NOAA ftp .dat weather station
% file and stores the data in a .mat file for further treatment as weather
% inputs to AERONOM.

% Original version sent by Joe Salomone in January 2010, thank you!

function []=NOAA_process(fname,fsave)

Open and read the file

open up file as read-only

fid=fopen(fname,’r’);

% read in the first line of file
ftmp=fopen(fid);
jj=1;

Extract and store the data

while ftmp~=-1

% -------------------
% Header Record Format:
% -------------------
%
% Variable Name Columns Description
% ------------------- ----- ------------
%
% Header Record Indicator 1- 1 # character
%
% Station Number 2- 6 WMO station number
%
% Year 7- 10
%
% Month
% 11-12
%
% Day
% 13-14
%
% Observation Hour
% 15-16 00-23 UTC
%
% Release Time
% 17-20 0000-2359 UTC, 9999 = missing
%
% Number of levels
% 21-24 number of subsequent data records

data(jj).ryear =str2num(ftmp(7:10));
data(jj).rmonth=str2num(ftmp(11:12));
data(jj).rday =str2num(ftmp(13:14));
data(jj).rhour =str2num(ftmp(15:16));
data(jj).rtime =str2num(ftmp(17:20));
data(jj).ohour=str2num(ftmp(15:16));
data(jj).rlevels=str2num(ftmp(21:24));

% initialize weather storage variables
pressure =zeros(data(jj).rlevels,1);
geoheight=zeros(data(jj).rlevels,1);
tempdegc =zeros(data(jj).rlevels,1);
dewptdegc=zeros(data(jj).rlevels,1);
winddir =zeros(data(jj).rlevels,1);
windspeed=zeros(data(jj).rlevels,1);

Loop through the data for the "rlevels" lines

for kk=1:data(jj).rlevels
    rtmp=fgetl(fid);
% -------------------
% Data Record Format:
% -------------------
%  Variable Name       Columns Description
%  ------------------- -------  ------------
% Major Level Type     1-1     1 = standard pressure level
%                      2       2 = significant thermodynamic level
%                      3       3 = additional wind level
% Minor Level Type     2-2     1 = surface, 2 = tropopause, 0 = other
% Pressure             3-8     units of Pa (mb * 100)
% Pressure Flag        9-9     A, B, or blank (see note 4 above)
% Geopotential Height  10-14  units of meters
% Geopotential Height Flag 15-15 A, B, or blank (see note 4 above)
% Temperature          16-20  units of degrees C * 10
% Temperature Flag     21-21  A, B, or blank (see note 4 above)
% Dewpoint Depression  22-26  units of degrees C * 10
% Wind Direction       27-31  units of degrees (0-360, inclusive)
% Wind Speed           32-36  units of (m/s)*10

pressure1(kk) = str2num(tmp(3:8));
geoheight1(kk)=str2num(rtmp(10:14));
tempdegc1(kk) =str2num(rtmp(16:20))/10;
dewptdegc1(kk)=str2num(rtmp(22:26))/10;
winddir1(kk) =str2num(rtmp(27:31));
windspeed1(kk)=str2num(rtmp(32:36))/10;
end

% Include treatment line to delete the unknown values (-999.9 or -9999... ...depending on the atmospheric parameter)

l5=length(geoheight1);
e=find(geoheight1(1:15)^=-9999);

l1=length(winddir1);
a=find(winddir1(1:11)^=-9999);

l2=length(windspeed1);
b=find(windspeed1(1:12)^=-999.9);

l3=length(tempdegc1);
c=find(tempdegc1(1:13)^=-999.9);

l4=length(pressure1);
d=find(pressure1(1:14)^=-9999);

l6=length(dewptdegc1);
f=find(dewptdegc1(1:16)^=-999.9);
g=intersect(f,c);

data(jj).pressure=pressure1(intersect(d,e));
data(jj).geoheightP=geoheight1(intersect(d,e));
data(jj).geoheightT=geoheight1(intersect(c,e));
data(jj).geoheightWs=geoheight1(intersect(b,e));
data(jj).geoheightWd=geoheight1(intersect(a,e));
data(jj).geoheightH=geoheight1(intersect(e,g));
data(jj).tempdegc=tempdegc1(intersect(e,c));
data(jj).dewptdegc=dewptdegc1(intersect(e,g));
data(jj).winddir=winddir1(intersect(e,a));
data(jj).windspeed=windspeed1(intersect(e,b));

clear pressure 1 geoheight1 tempdegc1 dewptdegc1 winddir1 windspeed1

ftmp=fgetl(fid);
jj=jj+1;
end

Save data in Matlab format

save (fsave)
end

A.1.2 The computation of Weather polynomial fits

Contents

- INPUTS
- OUTPUTS
  - Extract weather data for given time frame
  - Compute humidity polynomial fit in two-step process
  - Compute pressure polynomial fit
  - Compute wind speed and wind direction polynomial fits
  - Compute temperature polynomial fit
  - Extract weather for year(s), month(s) and time
• Processing NOAA data for humidity
• Code to compute the molecular concentration of water molecules from dew point values (cf NOAA files)
• Compute polynomial fit for humidity
• Compute polynomial fit for pressure
• Compute polynomial fit for wind speed
• Compute polynomial fit for temperature

% From NOAA data, compute weather parameter polynomial fits to use as inputs in AER Nom

% Successive developments between February 2010 to August 2011

INPUTS:

• year: year of interest
• month: month(s) of interest
• day: day(s) of interest
• time: 1 (morning), 2 (afternoon) or 3 (night time) releases
• fname: Matlab .mat NOAA weather data file

OUTPUTS:

• T: polynomial coefficients for Temperature [K]
• h: polynomial coefficients for humidity [% water vapor]
• P: polynomial coefficients for Pressure [Pa]
• W: polynomial coefficients for wind speed [m/s]
function[T,h,P,W]=Weather_pfit(year,month,day,time,fname)

% Extract weather data for given time frame
[Wspeed,geoZT,geoZH,geoZW,geoZP,tempZ,pressZ,Dp]=Extract_Weather...
    (year,month,time,fname);

% Compute humidity polynomial fit in two-step process
[XH2O]=NOAAh(pressZ,tempZ,Dp);
[h]=hfit(XH2O,geoZH,day);

% Compute pressure polynomial fit
[P]=Pressfit(day,geoZP,pressZ);

% Compute wind speed and wind direction polynomial fits
[W]=Wspeedfit(day,geoZW,Wspeed);

% Compute temperature polynomial fit
[T]=Tempfit(day,geoZT,tempZ);
end

Extract weather for year(s), month(s) and time

function[Wspeed,geoZT,geoZH,geoZW,geoZP,tempZ,pressZ,Dp]=Extract_Weather...
    (year,month,Time,time,fname)

if Time==1
    time=600:1000;
elseif Time==2
    time=1100:1500;
elseif Time==3
    time=[0001:0200 2200:2359];
end

load(fname, 'mat')

% Extract the data for given year(s)
for k=1:length(data)
    if ismember(data(k).ryear,year)==1
        ind(k)=k;
    end
end
ind1=find(ind~=0);

% Extract the data for given month(s)
for k=1:length(ind1)
    if ismember(data(ind1(k)).rmonth,month)==1
        test(k)=ind1(k);
    end
end
end
end
end
end
end
end
end
if isempty(find(test3~=0, 1))
    error('no such data!')
end
end
end
end
end
end
end
end
end
end
end
end
end
end
end
end
end
end
end
% print the corresponding data
data1=data(ind3);
pressure=zeros(60,length(ind3));
  geoheightT=zeros(60,length(ind3));
  geoheightW=zeros(60,length(ind3));
  geoheightH=zeros(60,length(ind3));
  geoheightP=zeros(60,length(ind3));
  tempdegc=zeros(60,length(ind3));
  windspeed=zeros(60,length(ind3));
  dp=zeros(60,length(ind3));

for k=1:length(ind3)
  pressure(1:length(data1(k).pressure),k)=data1(k).pressure;
  geoheightT(1:length(data1(k).geoheightT),k)=0.001*...
    data1(k).geoheightT;
  geoheightW(1:length(data1(k).geoheightWs),k)=0.001*...
    data1(k).geoheightWs;
  geoheightH(1:length(data1(k).geoheightH),k)=0.001*...
    data1(k).geoheightH;
  geoheightP(1:length(data1(k).geoheightP),k)=0.001*...
    data1(k).geoheightP;
  tempdegc(1:length(data1(k).tempdegc),k)=data1(k).tempdegc;
  windspeed(1:length(data1(k).windspeed),k)=data1(k).windspeed;
  Ldp=length(data1(k).dewptdegc);
  LT=length(data1(k).tempdegc);
  Tpp=data1(k).tempdegc;
  if Ldp==LT
    dp(1:Ldp,k)=data1(k).tempdegc-data1(k).dewptdegc;
  elseif Ldp<LT
    dp(1:Ldp,k)=Tpp(1:Ldp)-data1(k).dewptdegc;
  end
end
% Eliminate values of height that are above 12 km

for k=1:length(ind3)
a=find(geoheightT(:,k)<=15);
b=find(geoheightW(:,k)<=15);
c=find(geoheightH(:,k)<=15);
d=find(geoheightP(:,k)<=15);

Wspeed(1:length(b),k)=windspeed(b,k);
Wdir(1:length(b),k)=winddir(b,k);
geoZT(1:length(a),k)=geoheightT(a,k);
geoZW(1:length(b),k)=geoheightW(b,k);
geoZH(1:length(c),k)=geoheightH(c,k);
geoZP(1:length(d),k)=geoheightP(d,k);
tempZ(1:length(a),k)=tempdegc(a,k);
pressZ(1:length(d),k)=pressure(d,k);
Dp(1:length(c),k)=dp(c,k);
end

[aa bb]=size(geoZT);
geoZH=geoZH(1:aa,:);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Processing NOAA data for humidity

function [XH20]=NOAAh(pressZ,tempZ,Dp)
[a b]=size(pressZ);
XH20=zeros(a,b);
for k=1:b
Code to compute the molecular concentration of water molecules (humidity) from dew point values (cf NOAA files)

```matlab
function [XH20] = DewTOh(P,D,T)

% gas constants
R = 8.314;
M = 18.01;
Pso = 101.325e3;

% Use the Magnus approximations which is temperature dependant
if T > -45
    beta = 17.62;
    lambda = 243.12;
else
    beta = 22.46;
    lambda = 272.63;
end
```
% Use formula to get relative humidity, see Ref [51]
RH=100.*exp((Dp.*(beta-(beta.*T.)/(lambda+T)))-lambda*beta.*
  T.)/(lambda+T))/(lambda+Dp));

% From relative humidity, go back to %X20
% first step compute the air density
nair=P.*100./(R.*T);

% Get vapor pressure of water (and ice) which depends on the temperature
aWater=[6.107799961 4.436518521e-1 1.428945805e-2 2.650648471e-4 ...
  3.031240396e-6 2.034080948e-8 6.136820929e-11];
aIce=[6.109177956 5.034698970e-1 1.886013408e-2 4.176223716e-4 ...
  4.824720280e-6 4.838803174e-8 1.838826904e-10 ];

e=zeros(length(T),1);
for k=1:length(T)
  if T(k)>=0
    e(k,1)=aWater(1,1)+T(k)*(aWater(1,2)+T(k)*(aWater(1,3)+T(k)*...
      (aWater(1,4)+T(k)*aWater(1,5)+T(k)*(aWater(1,6)+T(k)*
        aWater(1,7)))));
  else
    e(k,1)=aIce(1,1)+T(k)*(aIce(1,2)+T(k)*(aIce(1,3)+T(k)*...
      (aIce(1,4)+T(k)*aIce(1,5)+T(k)*(aIce(1,6)+T(k)*
        aIce(1,7)))));
  end
end

% Know compute xH2O
xH20=RH.*e./(100.*P);

% Get to cH20 [g/m^3]
cH20=xH20.*nair.*M;
% Apply formula from ANSI Standard S1.26 Appendix D
XH20=4.55e-4.*cH20.*T.*P./Pso;
end

Compute polynomial fit for humidity

function [h]=hfit(XH20,geoZH,day)

% Storage vector initialization

[a b]=size(geoZH);
X0=reshape(geoZH(:,day),a*length(day),1);
[Z,ind]=sort(X0);

h=reshape(XH20(:,day),a*length(day),1);
h=h(ind);
ind1=find(h~=0);
h=h(ind1);
Z=Z(ind1);
Z=Z-min(Z); % normalizing against station altitude

% Filter the altitude vector so to keep only singular values, and specify...
...the corresponding indices in order to extract the corresponding values...
...for Wspeed // m1 gives the "start" indices and m2 the "stop" ,...
...indices. That means that for each row we need to extract the ...
...values from m1 to m2 then compute a mean (trend).

[Zsingle, m1, n1] = unique(Z,'first');
[Zsingle, m2, n2] = unique(Z,'last');

Avh=zeros(length(m1),1);
for k=1:length(m1)
    Avh(k,1)=mean(h(m1(k):m2(k)),1);
end

% Apply smoothing process
N=length(h);
hs=smooth(Avh,N,'lowess');

% Build up a loop in order to gain accuracy for the profile. Goal: compute
% the trend that suits the best the profile (by minimizing the error due to
% norm of residuals but on the same time which matches the profile smoothly
% No need to go very up in polynomial order for ex).
zfit = linspace(Zsingle(1),Zsingle(end),1000);
e_stor=0;

for k=1:10; % stop at 10th order polynomial degree
  % Compute conditioned fit
  [h,S] = polyfit(Zsingle,hs,k);
  [hBIS,SBIS] = polyfit(Zsingle,hs,k+1);
  % Work on the correlation coefficient
  norm=S.normr;
  dev=std(hs);
  R=1-norm^2/((length(hs)-1)*dev^2);
  % could even work on the error
  e=100*sqrt((1/R-1)/(length(hs)-2));

  % Same technique for (k+1) order polynome
  normBIS=SBIS.normr;
  dev=std(hs);
  RBIS=1-normBIS^2/((length(hs)-1)*dev^2);
eBIS=100*sqrt((1/RBIS-1)/(length(hs)-2));
if abs(eBIS-e)<e_stor
    break
else
end

e_stor=abs(eBIS-e);
end


Compute polynomial fit for pressure

function [P]=Pressfit(day,geoZP,pressZ)

% Storage vector initialization

[a b]=size(geoZP);

X0=reshape(geoZP(:,day),a*length(day),1);
Y1=reshape(pressZ(:,day),a*length(day),1);

% First step, eliminate -999.9 entries in tempZ matrix
ind=find(Y1==-999.9);
ind1=find(Y1==0);
ind2=intersect(ind,ind1);
P=Y1(ind2);

% Extract corresponding geometric altitudes
X0=X0(ind2);
X0=X0-min(X0); % getting back to normalizing against source altitude
[Z,ind]=sort(X0);
P=P(ind);
% Filter the altitude vector so to keep only singular values

[Zsingle, m1, n1] = unique(Z,'first');
[Zsingle, m2, n2] = unique(Z,'last');

AvPz=zeros(length(m1),1);

for k=1:length(m1)
    AvPz(k,1)=mean(P(m1(k):m2(k)),1);
end

% Apply a smoothing technique
N=length(AvPz);
Ps=smooth(AvPz,N,'lowess');

% Build up a loop in order to gain accuracy for the profile. Goal: compute
% the trend that suits the best the profile (by minimizing the error due to
% norm of residuals but on the same time which matches the profile smoothly
% No need to go very up in polynomial order).
zfit = linspace(Zsingle(1),Zsingle(end),1000);
e_stor=0;

for k=1:10; % stop at 10th order polynomial degree
    % Compute conditioned fit
    [P,S] = polyfit(Zsingle,Ps,k);
    [PBIS,SBIS] = polyfit(Zsingle,Ps,k+1);

    % Work on the correlation coefficient
    norm=S.normr;
    dev=std(Ps);
    R=1-norm^2/(((length(Ps)-1)*dev)^2);
    e=100*sqrt((1/R-1)/((length(Ps)-2)));
% Same for (k+1) order polynome
normBIS=SBIS.normr;
dev=std(Ps);
RBIS=1-normBIS^2/((length(Ps)-1)*dev^2);
eBIS=100*sqrt((1/RBIS-1)/(length(Ps)-2));

if abs(eBIS-e)<e_stor
    break
else
    e_stor=abs(eBIS-e);
end

P=PBIS;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Compute polynomial fit for wind speed

function [W]=Wspeedfit(day,geoZW,Wspeed)

% Storage vector initialization

[a b]=size(geoZW);

X0=reshape(geoZW(:,day),a*length(day),1);
Y1=reshape(Wspeed(:,day),a*length(day),1);

% getting rid of the zero altitude values
ind=find(X0~=0);


X0=X0(ind);

% Normalize the altitudes to station altitude = min (geoZW)
X0=X0-min(X0);
Y1=Y1(ind);

[Z,ind]=sort(X0);
WSPEED=Y1(ind);

% Filter the altitude vector so to keep only singular values
[Zsingle, m1, n1] = unique(Z,'first');
[Zsingle, m2, n2] = unique(Z,'last');

AvWspeed=zeros(length(m1),1);

for k=1:length(m1)
    AvWspeed(k,1)=mean(WSPEED(m1(k):m2(k)),1);
end

% Apply a smoothing technique
N=length(WSPEED);
Ws=smooth(AvWspeed,N,'lowess');
Wds=smooth(AvWdir,N,'lowess');

% Build up a loop in order to gain accuracy for the profile. Goal: compute
% the trend that suits the best the profile (by minimizing the error due to
% norm of residuals but on the same time which matches the profile smoothly
% No need to go very up in polynomial order).
zfit = linspace(Zsingle(1),Zsingle(end),1000);
e_stor=0;

for k=1:10; % stop at 10th order polynomial degree
% Compute conditioned fit
[pSPEED,S] = polyfit(Zsingle,Ws,k);
[pSPEEDBIS,SBIS] = polyfit(Zsingle,Ws,k+1);
% Work on the correlation coefficient
norm=S.normr;
dev=std(Ws);
R=1-norm^2/((length(Ws)-1)*dev^2);
% could even work on the error
e=100*sqrt((1/R-1)/(length(Ws)-2));

% Same for (k+1) order polynome
normBIS=SBIS.normr;
dev=std(Ws);
RBIS=1-normBIS^2/((length(Ws)-1)*dev^2);
% could even work on the error
eBIS=100*sqrt((1/RBIS-1)/(length(Ws)-2));

if abs(eBIS-e)<e_stor
    break
else
    end
    e_stor=abs(eBIS-e);
end

W=pSPEEDBIS;
end

Compute polynomial fit for temperature

function [T]=Tempfit(day,geoZT,tempZ)

% Storage vector initialization
[a b]=size(geoZT);

X0=reshape(geoZT(:,day),a*length(day),1);
Y1=reshape(tempZ(:,day),a*length(day),1);

% First step, eliminate -999.9 entries in tempZ matrix
ind=find(Y1~=-999.9);
ind1=find(Y1~=0);
ind2=intersect(ind,ind1);
T=Y1(ind2);

% Extract corresponding geometric altitudes
X0=X0(ind2);
% normalize altitude vector to station altitude
X0=X0-min(X0);
[Z,ind]=sort(X0);
% getting back to Kelvins
Tz=273.15+T(ind);

% Filter the altitude vector so to keep only singular values, and specify
% the corresponding indices in order to extract the corresponding values for
% Wspeed and Wdirection // m1 gives the "start" indices and m2 the "stop"
% indices. That means that for each row we need to extract the values from
% m1 to m2 then compute a mean (trend).

[Zsingle, m1, n1] = unique(Z,'first');
[Zsingle, m2, n2] = unique(Z,'last');

AvTz=zeros(length(m1),1);

for k=1:length(m1)
    AvTz(k,1)=mean(Tz(m1(k):m2(k)),1);
end
% Apply smoothing technique
N=length(AvTz);
Ts=smooth(AvTz,N,'lowess');

% Build up a loop in order to gain accuracy for the profile. Goal: compute
% the trend that suits the best the profile (by minimizing the error due to
% norm of residuals but on the same time which matches the profile smoothly
% No need to go very up in polynomial order).
zfit = linspace(Zsingle(1),Zsingle(end),1000);
e_stor=0;

for k=1:10; % stop at 10th order polynomial degree
% Compute conditioned fit
[pT,S] = polyfit(Zsingle,Ts,k);
[pTBIS,SBIS] = polyfit(Zsingle,Ts,k+1);
% Work on the correlation coefficient
norm=S.normr;
dev=std(Ts);
R=1-norm^2/((length(Ts)-1)*dev^2);
e=100*sqrt(((1/R-1)/(length(Ts)-2)));

% Same for (k+1) order polynome
normBIS=SBIS.normr;
dev=std(Ts);
RBIS=1-normBIS^2/((length(Ts)-1)*dev^2);
eBIS=100*sqrt(((1/RBIS-1)/(length(Ts)-2)));

if abs(eBIS-e)<e_stor
    break
else
end
e_stor=abs(eBIS-e);
end

T=pTBIS;
end

A.2 AERNOM’s code

This section provides the full AERNOM code to produce ground SEL contours from a high-altitude overflight, including the forward flight effects in a non-turbulent atmosphere.

Contents

- Inputs
- Outputs
- Computation of sound and wind speed values and gradients
- Limiting angle prediction routine
- Routine to compute the turning point altitude
- 2-dimensional sound pressure level decay routine
- 4th order Runge-Kutta integration ray-tracing code
- Moving effects routine
- Routine to compute the separation launching angle
- Sub-routine to compute the Excess Ground Attenuation (EGA)
- Function that provides the plane-wave coefficient
- Sub-routine to compute the geometrical and refraction losses
- Sub-routine to compute the absorption losses
- ANSI S1.26 absorption algorithm
- Sub-routine to compute the shadow zone diffracted pressure field
- Function that generate an average limiting ray radii of curvature
- Function to find poles in residue series solution

% Full code to produce ground contours using AERNOM, using the flat and % even ground assumption and rotation around the vertical axis to speed the % computation
Developed from October 2009 to June 2011

Inputs:

- $L_w$: third-octave band source sound power level [dBA].
- $\text{Dir}$: angle between the flight track and the wind direction.
- $\text{Mach}$: aircraft Mach number (<1).
- $zs$: flight track mean altitude [km].
- $zr$: receivers altitude [km].
- $\delta R$: receiver grid range increment [km].
- $\delta R_{\text{grid}}$: contour grid range increment [km].
- $\text{Xmax}$: maximum grid X coordinate [km].
- $\text{Ymax}$: maximum grid Y coordinate [km].
- $\delta \phi$: azimuthal angle increment in the rotation around the vertical axis [rd]
- $\delta \theta$: polar angle increment in each 2-dimensional vertical planes
- $\sigma$: ground flow resistivity [rayls]
- $N_{\text{pts source}}$: number of point sources to represent line source
- $T$: temperature polynomial fit [K] from NOAA balloon data
- $W$: wind speed polynomial fit [m/s] from NOAA balloon data
- $h$: humidity polynomial fit [% water vapor] from NOAA balloon data
- $P$: pressure polynomial fit [Pa] from NOAA balloon data
Outputs:

- (Xgrid,Ygrid): ground contour grid coordinates
- levelTotal: corresponding SEL values [dBA]

```matlab
function [levelTotal Xgrid Ygrid]=AEROMN_contour(Lw,DIR,Mach,zs,zr,deltaR,...
    deltaR_grid,Xmax,Ymax,deltaphi,deltatheta,sigma,N_ptssource,T,W,h,P)

% 1/3 Octave-band frequencies from 50 Hz to 2 kHz
freq=[50 63 80 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000];

% settings for aircraft flight speed
[c_top dc w_top dw]=Validprof_AEROM(zs,T,W,DIR);
Ceff_top=c_top+w_top;
V=Mach*Ceff_top;

% setting up contour horizontal square grid
Xvals=-Xmax:deltaR_grid:Xmax;
Lx=length(Xvals);

Yvals=0:deltaR_grid:Ymax;
Ly=length(Yvals);

[Xgrid,Ygrid]= meshgrid(Xvals,Yvals);

% initializing
LevelAll= zeros(length(Ygrid),length(Xgrid),N_ptssource);

% setting up receiver grid
phi=0:deltaphi:pi;

% Span all 2-dimensional vertical planes, determined by values of phi
r_interp=0:deltaR:max(Xmax,Ymax);
LpFINAL=zeros(length(phi),length(r_interp));
```
for m=1:length(phi)
  % Know where the computation is
  pourc=m/length(phi)*100;
  display(num2str(pourc),’ % done’);

  % Limiting angle prediction prior to running ray-tracing code
  thetalim=RayWlim(zs,0.1,T,W,phi(m)-DIR);

  if isreal(thetalim)==0
    thetaLim=0; % limit to horizontal vertical launching angle for ... 
    ...downwind refraction
  else
    thetaLim=thetalim-0.5; % this will prevent from computing...
    ...unrealistic rays
  end

  % Generate polar launching angle vector
  N_angle=floor((90+thetaLim)/deltatheta);
  theta=linspace(-90,thetaLim-1,N_angle);

  % Initializing prior to run
  Lp=zeros(1,length(theta));
  range=zeros(1,length(theta));

  % Running the ray-tracing code for each 2-dimensional plane
  for p=1:length(theta)
    theta_launch=theta(p);
    [R L Lcutoff rlimit zlimit]=AERNO2D(Lw,zs,zr,freq,sigma,phi(m)-DIR,...
      theta_launch,T,W,h,P,N_ptsSource,V);
    Lp(p)=L;
    range(p)=R;
  end
% get results at ranges of interest using the spline function
Lp_fitting_ill=spline(range,Lp,r_interp(r_interp<max(range)));  
Lp_final=Lp_fitting_ill;

% Determine if there is need for a shadow zone calculation
if thetalim<0 && R<max(r_interp);
  % Create receiver range vector in shadow zone
  r_cutoff=R;
  indr=mod(r_cutoff,deltaR);
  Drec_sdz=deltaR-indr:deltaR:max(r_interp)-r_cutoff;
  Lp_cutoff=L_cutoff;
  [c0 dc ws dw]=Validprof_AERONOM(0,T,W,phi);
  Lp_sdz=shadow_press_enroute(sigma,rlimit,ylimt,Drec_sdz,Lp_cutoff,c0);
  Lp_final=horzcat(Lp_fitting_ill,Lp_sdz);

elseif thetalim==0 && R<max(r_interp); % downwind refraction
  indr=mod(R,deltaR);
  L_extra=length(deltaR-indr:deltaR:max(r_interp)-R);
  % no sound reaches extra receiver points: allocate -150 dB
  Lp_extra=-150*ones(1,L_extra);
  Lp_final=horzcat(Lp_final,Lp_extra);
end

LpFINAL(m,:)=Lp_final;
end
Lp=LpFINAL;

% Add contribution of overflight time
[c_top dc w_top dw]=Validprof_AERONOM(zs,T,W,0);
Ceff_top=c_top+w_top;
v_aircraft=Mach*Ceff_top;
% time spent above grid
tau_ovf=(2*Xmax+1)*1000/v_aircraft;
% extra dL to preduce SEL due to time spent on overflight portion
dL_SEL=10*log10(tau_ovf);

% impact on levels
Lp=Lp+dL_SEL;

% Step to translate noise levels from the receiver to the ground contour...
...grid: interpolation technique (initially written by Joyce Rosenbaum)

Level= zeros(Ly,Lx);

% The following example represents a 301 km flight track centered above the
% Y-axis.
Xflight=linspace(-150,150,N_pts source);
Yflight=zeros(1,length(Xflight));

% create rGrid and thetaGrid matrices - different for each run
for p=1:length(Xflight)
    [thetaGrid,rGrid]= cart2pol(Xgrid-Xflight(p),Ygrid-Yflight(p));
end

Lp_receiver=Lp;

    for mm= 1:Ly %represent the y axis.
        for n= 1:Lx %represent the x axis
            % first things first, discard any values outside of receiver
            % grid
            if abs(rGrid(mm,n))<=max(Xmax,Ymax)

            % for matching receiver grid/contour grid values, proceed to
            % interpolation technique:
if rem(rGrid(mm,n),deltaR)==0

if rem(thetaGrid(mm,n),deltaphi)==0
    %if the cartesian grid point falls exactly on a polar gridpoint, then the level at that cartesian gridpoint equals the level at the corresponding polar gridpoint
    Level(mm,n)=Lp_receiver(round(thetaGrid(mm,n)... /deltaphi+1),round(rGrid(mm,n)/deltaR+1));

else

    %otherwise, if the cartesian gridpoint falls exactly ...
    ...
on the r value, but between two theta values of the ...
    ...
polar coordinate system:
    floorTheta=deltaphi*floor(thetaGrid(mm,n)/deltaphi);
    ceilTheta=floorTheta+deltaphi;
    %find the coordinates A and B where A and B can be ...
    ...found in the Numerical Recipes book, p. 107
    Ainterp= (ceilTheta-thetaGrid(mm,n))/deltaphi;
    Binterp= 1-Ainterp;
    %interpolate the level at the r value between to two ...
    ...values of theta
    Level(mm,n)= 10*log10(Ainterp*10^((Lp_receiver...
    (round(floorTheta/deltaphi+1),...
    round(rGrid(mm,n)/deltaR+1))/10)+...
    Binterp*10^((Lp_receiver(round(ceilTheta/deltaphi+1)... , (round(rGrid(mm,n)/deltaR+1))/10));
end

elseif rem(thetaGrid(mm,n),deltaphi)==0
%if the cartesian gridpoint falls between two r values in ...
...the polar coordinate system but exactly on a theta value:
floorR= deltaR*floor(rGrid(mm,n)/deltaR);
ceilR= floorR+deltaR;
%find the coordinates A and B where A and B can be found in ...
...the Numerical Recipes book, p. 107
Ainterp= (ceilR-rGrid(mm,n))/deltaR;
Binterp= 1-Ainterp;
%interpolate the level at the theta value between to two
%values of r

Level(mm,n)= 10*log10(Ainterp*10^(-Lp_receiver)...  
(round(thetaGrid(mm,n)/deltaphi+1),...  
round(floorR/deltaR+1))/10)+...  
Binterp*10^(-Lp_receiver...  
(round(thetaGrid(mm,n)/deltaphi+1),...  
round(ceilR/deltaR+1))/10));

else

%if the cartesian gridpoint falls between both two r values...
...and two theta values in the polar coordinate system:
floorTheta= deltaphi*floor(thetaGrid(mm,n)/deltaphi);
ceilTheta= floorTheta+deltaphi;

floorR= deltaR*floor(rGrid(mm,n)/deltaR);
ceilR= floorR+deltaR;

%find the coordinates A and B along an r where A and B ...  
...can be found in the Numerical Recipes book, p. 107
AinterpR= (ceilTheta-thetaGrid(mm,n))/deltaphi;
BinterpR= 1-AinterpR;
%interpolate the level at the lower r value between to two...
...values of theta
LevelLR = 10*log10(AinterpR*10^(Lp_receiver...)
  (round(floorTheta/deltaphi+1),... 
  round(floorR/deltaR+1))/10)+...
  BinterpR*10^(Lp_receiver...
  (round(ceilTheta/deltaphi+1),round(floorR/deltaR+1))/10));

%interpolate the level at the upper r value between to
%two values of theta
LevelUR = 10*log10(AinterpR*10^(Lp_receiver...)
  (round(floorTheta/deltaphi+1),... 
  round(ceilR/deltaR+1))/10)+...
  BinterpR*10^(Lp_receiver(round(ceilTheta/deltaphi+1)... 
  ,round(ceilR/deltaR+1))/10));

%find the coordinates A and B between the r's at the...
...correct theta value where A and B can be found in the ...
...Numerical Recipes book, p. 107
AinterpTheta = (ceilR-rGrid(mm,n))/deltaR;
BinterpTheta = 1-AinterpTheta;
%interpolate the level at the correct theta value between
%to two values of r
Level(mm,n) = 10*log10(AinterpTheta*10^(LevelLR/10)+...
  BinterpTheta*10^(LevelUR/10));

end
else continue
end
end

LevelAll(1:Ly,:,p) = Level;

% add a line to apply symmetry (axisymmetric approximation) from the X
Computation of sound and wind speed values and gradients

given an altitude $z$ [km] and an angle between the wind direction and a particular 2-dimensional vertical plane, $\phi$.

function [c dc w dw] = Validprof_AEROM(z,T,W,phi)

% Constants
M=28.964; % for altitudes below 90 km, Mair is assumed to be a constant ...
...[kg/mol]
R=8314.48; % the universal gas constant
gamma_air=1.4;

dT=polyder(T);
c=sqrt(gamma_air*R*polyval(T,z)/M);
dc=0.5*sqrt(gamma_air*R/(polyval(T,z)*M))*dT;

w0=polyval(W,z);
dw0=polyval(polyder(W),z);
w=w0*cos(phi);
dw=dw0*cos(phi);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Limiting angle prediction routine

in the presence of horizontal wind vector, using the fact that for a ...

...z-stratified atmosphere the r-component of the slowness vector is...
...constant.

function[thetalim]=RayWlim(z,dz,T,W,phi)

% Determination of the turning point altitude
[Zturn]=turn_point(T,W,phi);

if length(Zturn)==1;
% get soundspeed at turning point altitude
[c0 dc0 w0 dW0]=Validprof_AERNOM(Zturn,T,W,phi);

% analytical solution to first angle
Routine to compute the turning point altitude

% The turning point is the altitude at which there is a maximum in ...
... effective sound speed = c_thermo + wind speed projected in the ...
... 2D plane which usually occurs at lower altitudes for a positive ...
...lapse rate which will be the standard case.

function [Zturn]=turn_point(T,W,phi)

dz=0.1;
z_down=0;
z_up=10;

z=z_down:dz:z_up;
ceff=zeros(1,length(z));

for k=1:length(z)
    [~,c,d,w]=Validprof_AERNOM(z(k),T,W,phi);
    ceff(1,k)=c+w;
end

ind_turn=ceff==max(ceff);
Zturn=z(ind_turn);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

2-dimensional sound pressure level decay routine

function[R Lp CutField rlimit zlimit]=AERNOM_2D(Lw,zs,zt,freq,sigma,phi, ...
    theta,T,W,h,P,N_pts source,V)

    % produce equivalent single source SPL
    Lw=10.*log10(10.^(Lw./10)./(N_pts source));

    % Initial settings
% Storage loss vectors
GeomLoss=zeros(1,1);
Labs_corr=zeros(length(freq),1);
EGA=zeros(length(freq),1);
R= zeros(1,1);
GroundLevel=zeros(length(freq),1);
Lp= zeros(1,1);

% EGA sub-routine
% launch 2 rays per receiver point 
% 1st step: launch direct ray
TRACK=1;
[R_direct Z_direct T_direct S_direct theta_end Slowness]=RK4_ray(zs,theta,...
    zr,TRACK,T,W,phi);

% apply moving source effects
[Eff_freq_ratio Amp_factor]=Mveff_enroute(Z_direct,zs,V,T,W,Slowness,theta...,
    ,phi);
% Apply Amp_factor on Lw;
Lw=Lw+Amp_factor;

% Compute separation angle to produce direct/reflected ray interaction
% (EGA portion of the loss)
sep_angle=sep_AERMOM(zs,zr,theta,R_direct,T,W,phi);
theta_reflected=theta+sep_angle;
% 2nd step: launch reflected ray
TRACK=2;
[R_reflec Z_reflec T_reflec S_reflec Graz_angle nn]=RK4_ray...
    (zs,theta_reflected,zr,TRACK,T,W,phi);

% Store useful information
dTau=T_reflec-T_direct;
Ratio_dist=S_reflec/S_direct;
EGA(:,1)=EGA_AERNOM(dTau,Ratio_dist,Graz_angle,freq,\sigma);

% Geometrical loss sub-routine (including refraction)
c0=Validprof_AERNOM(0,T,W,\phi);
[Geomloss]=refract_AERNOM(theta,R_direct,Z_direct,R_reflec,Z_reflec,...
sep_angle,c0,T_direct,S_direct,theta_end);
GeomLoss(1,1)=Geomloss;

% Absorption loss sub-routine
Labs_corr(:,1)=AbsorptionLoss_AERNOM(R_direct,Z_direct,freq,T,h,P,...
Eff_freq_ratio);
rlimit=R_direct;
zlimit=Z_direct;
R(1,1)=R_direct(end);

% Compute OSPL on the ground (logarithmic summation of freq band)

% Starting to generate the field on the ground - per frequency
for pp=1:length(freq)
GroundLevel(pp,:)=Lw(pp)+EGA(pp,:) +GeomLoss-Labs_corr(pp,:);
end

% Proceed to Log summation of different contributions from frequencies
GLstor=10.^(GroundLevel(:,1)./10);
% Sum all the values of this vector
Sum_Sstor=sum(GLstor);
Lp(1,1)=10*log10(Sum_Sstor);

% Prepare for shadow zone prediction
CutField=GroundLevel(:,1);
end
4th order Runge-Kutta integration ray-tracing code

function[R_final Z_final T_final S_final Reflec_angle Slowness]=RK4_ray...
    (z,Theta,zr,TRACK,T,W,phi)

% The TRACK option specifies the nature of rays that we need to track for...
...the output:
% --1: DIRECT ONLY
% --2: INDIRECT ONLY

% transform theta in rd, speeds up computation
theta=pi*Theta/180;

% Initializing
Graz_angle=zeros(1,1);

% put source at origin of the grid
ri=0;
zi=z;

% Compute initial conditions for ray-tracing algorithm
[c dc w dw]=Validprof_AERNOM(z,T,W,phi);
a=cos(theta)/(c+w*cos(theta));
b=sin(theta)/(c+w*cos(theta));
vec=[ri zi ai bi];

% starting the loop to get the ray curves over the entire grid
imax=10000; % ensure that imax is big enough

% storage matrix
r=zeros(imax+1,4); %stores r,z,a at all the incremental steps
r(1,:)=veci;
Slowness=zeros(1,imax+1);
Slowness(1,1)=cos(theta)/(c+w);

% looping in the integration process
test=0;
time=0;
S=0;
gr=0;

for i=1:imax;

% ----- separation between direct and reflected portions
% direct portion
if gr==0
    % by default the step size is set to 1 km
    % Towards lower altitudes, propagate rays using a coarse
    % size step decrease
    if veci(2)>1
        h=.5;
    end

    % For low altitudes, use the golden secant method
    if veci(2)<1
        Vert_distance=veci(2);
        h=Vert_distance/2;
    end

% reflected portion

elseif gr==1
% use the golden secant method to get back to the receiver location
Vert_distance=abs(veci(2)-zr);
h=max(Vert_distance/2,1e-5);
end

% Ray tracing algorithm {4th order Runge-Kutta integration method}

[c1 dc1 W1 dW1]=Validprof_AERNOM(veci(2),T,W,phi);
ceff1=c1/(1-W1*veci(3));
costheta=veci(3)*ceff1;
omega1=c1/(c1+W1*costheta);

% 1st step
vecistor1=h*[(veci(3)*c1^2/omega1+W1)/ceff1,c1^2*veci(4)/(omega1*ceff1)...,
0,(-omega1*dc1/c1-veci(3)*dW1)/ceff1];
vecistor2=veci+0.5*vecistor1;

% 2nd step
[c2 dc2 W2 dW2]=Validprof_AERNOM(vecistor2(2),T,W,phi);
ceff2=c2/(1-W2*vecistor2(3));
costheta2=vecistor2(3)*ceff2;
omega2=c2/(c2+W2*costheta2);
vecistor3=h*[(vecistor2(3)*c2^2/omega2+W2)/ceff2,...
c2^2*vecistor2(4)/(omega2*ceff2),0,...
(-omega2*dc2/c2-vecistor2(3)*dW2)/ceff2];
vecistor4=veci+0.5*vecistor3;

% 3rd step
[c4 dc4 W4 dW4]=Validprof_AERNOM(vecistor4(2),T,W,phi);
ceff4=c4/(1-W4*vecistor4(3));

costheta4=vecistor4(3)*ceff4;
omega4=c4/(c4+W4*costheta4);

vecistor5=h*[(vecistor4(3)*c4^2/omega4+W4)/ceff4,...
c4^2*vecistor4(4)/(omega4*ceff4),0,...
(-omega4*dc4/c4-vecistor4(3)*dW4)/ceff4];
vecistor6=veci+vecistor5;
time=time+h*1000/ceff4;

% 4th step
[c6 dc6 W6 dW6]=Validprof_AERNOM(vecistor6(2),T,W,phi);
ceff6=c6/(1-W6*vecistor6(3));

costheta6=vecistor6(3)*ceff6;
omega6=c6/(c6+W6*costheta6);

vecistor7=h*[(vecistor6(3)*c6^2/omega6+W6)/ceff6,...
c6^2*vecistor6(4)/(omega6*ceff6),0,...
(-omega6*dc6/c6-vecistor6(3)*dW6)/ceff6];

% Final integration step
vecip1=veci+1/6.*(vecistor1+2.*vecistor3+2.*vecistor5+vecistor7);

r(i+1,:)=vecip1;
veci=vecip1;
test=test+1;
Slowness(1,i+1)=costheta6/(c6+W6);
S=S+h;
Graz_angle=acos(costheta6);
% Start ground reflection modifications for next step
if vecip1(2)<1e-8;
    veci(4)=-vecip1(4);
    % store grazing angle for reflection coefficient calculation
    gr=gr+1;
end

% Conditions to stop the computation

if TRACK==1 % defining the condition of direct ray
    if veci(2)<zr
        break
    end
end
end

if TRACK==2 && gr==1 % defining the condition of reflected ray
if veci(2)>zr
    break
end
end
end

% Extract ray coordinates - from which it is possible to get the parameters
% from geometrical considerations
Ri=r(:,1);
Zi=r(:,2);
ind=find(Zi==0);

R_final=Ri(ind);
Z_final=Zi(ind);
% Once we have the ray trajectory, compute the geometrical parameters and...  
...gather the information

% we need to properly interpolate to the correct receiver locations
r_end=R_final(end-1:end);
z_end=Z_final(end-1:end);

r_final=spline(z_end,r_end,zr);

if TRACK==1  % direct ray

s_extra=sqrt(((r_final-R_final(end))^2+(zr-Z_final(end))^2));
T_final=time-s_extra*1000/ceff6;
S_final=S-s_extra;
Reflec_angle=Graz_angle;

elseif TRACK==2  % reflected ray

s_extra=sqrt(((r_final-R_final(end))^2+(zr-Z_final(end))^2));
T_final=time+s_extra*1000/ceff6;
S_final=S+s_extra;
Reflec_angle=Graz_angle;

end
Slowness=Slowness(Slowness~==0);
R_final=vertcat(R_final(1:end-1),r_final);
Z_final=vertcat(Z_final(1:end-1),zr);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Moving effects routine

    theta_launch,phi)

    % Refine ray trajectory
    Z_interp=Z(1):-0.01:0;
    Slowness_interp=spline(Z,Slowness,Z_interp);
    
    for k=1:length(Z_interp);
        Wind(k)=polyval(W,Z_interp(k));
    end
    
    W_S=Wind.*Slowness_interp;
    
    % Frequency shift near source altitude
    % Get sound speed, wind speed at source altitude
    [cs dc ws dw]=Validprof_AEROM(zs,T,W,phi);
    
    % apply formula from Ref: D. Roy "Doppler frequency shift in a Refractive
    % Atmosphere", J. Aircraft VOL. 24, NO. 9, Sept 1997
    Freq_shift=1/(1-V*cos(phi)*cosd(theta_launch)/(cs+ws));
    
    Eff_freq_ratio=Freq_shift.*(1-W_S);
    
    % Amplification factor - assuming radially oscillating sphere (approx)
    M=V/(cs+ws);
    Amp_factor=10*log10(1/(1-M*cos(phi)*cosd(theta_launch))^-3);
    
end
Routine to compute the separation launching angle

function[sep_angle]=sep_AERNOM(zs,zr,theta_ini,R_ini,T,W,phi)

range_receiver=R_ini(end);

% launch 2 rays with different separation angles
sep1=-5e-3;
sep2=-1e-3;
[R1 Z1 t1 s1 angle1]=RK4_ray(zs,theta_ini+sep1,zr,2,T,W,phi);
[R2 Z2 t2 s2 angle2]=RK4_ray(zs,theta_ini+sep2,zr,2,T,W,phi);

% interpolate for correct launching angle
deltaR1=range_receiver-R1(end);
deltaR2=range_receiver-R2(end);
x_coord=[deltaR1 deltaR2];
y_coord=[sep1 sep2];

sep_angle=spline(x_coord,y_coord,0);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Sub-routine to compute the Excess Ground Attenuation (EGA)

function[EGA]=EGA_AERNOM(DeltaT,Ratio_r,Reflec_angle,freq,sigma)

% Introduce 1/3 OB frequencies and their corresponding bandwidth
OBfreq=[50 63 80 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 ...
2000];
dfreq=(2^(1/6)-2^(-1/6)).*OBfreq; % formula for 1/3 OB freq band

% initialize vector
EGA=zeros(length(freq),1);

for pp=1:length(freq)
    f=freq(pp);
    Rp=Ray_ground(Reflec_angle,freq(pp),sigma);
    ind=find(freq(pp)==OBfreq(1,:));
    mu=2*pi*dfreq(1,ind)/(2*freq(pp));
    eta=2*pi*sqrt(1+(dfreq(1,ind)/2*freq(pp))^2);
    EGA(pp,1)=10*log10(1+(abs(Rp)/Ratio_r)^2+2*abs(Rp)/Ratio_r*...
        sin(mu*DeltaT/f)*cos(eta*DeltaT/f+angle(Rp))/(mu*DeltaT/f));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Function that provides the plane-wave coefficient

function [Rp]=Ray_ground(Graz_angle,f,sigma)

    % Delany and Bazley model
    Zc=1+0.0511*(sigma/f)^0.75+1i*0.0768*(sigma/f)^0.73;

    % Plane-wave reflection coefficient
    Rp=(sin(Graz_angle)-1/Zc)./(sin(Graz_angle)+1/Zc);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Sub-routine to compute the geometrical and refraction losses

function [Geomloss]=refract_AEROM(Theta,R1,Z1,R2,Z2,sep,c0,T_direct,...
    S_direct,theta_end)
z=Z1(1);

% compute spherical loss along straight ray
Rdirect=sqrt(R1(end)^2+z^2);
deltaL_geom=-10*log10(4*pi*(1e3*Rdirect)^2);

% correction for R2, bring value back to level at direct side of receiver
ind=find(Z2==min(Z2));
Z_temp=Z2(1:ind);
R_temp=R2(1:ind);
R_new=spline(Z_temp,R_temp,Z1(end));

% Follow Lamancusa’s method
gamma=abs(R(end)-R_new)^2;

% get equivalent straight line propagation (same wavefront)
d1=c0*T_direct/1000; % in km

% next step
gamma=2*d1*tand(sep/2);

% ray tube areas
Ar=gamma*d1;
Arp=gamma*S_direct;

% calculate receiving angle on ground
A_ray=Arp*cosd(-Theta-theta_end*180/pi);

% compute alpha_refraction and get total geometrical loss (including % refraction)
alpha_R=abs(A_ray/Ar);
deltaL_refrac=-10*log10(alpha_R);
Geomloss=deltaL_refrac+deltaL_geom;

dend

Sub-routine to compute the absorption losses

function[Labs_corr]=AbsorptionLoss_AERNOM(R_direct,Z_direct,freq,T,h,P,...
    Eff_freq_ratio)

% Refine ray trajectory
% from ray trajectory compute finer trajectory (the ray tracing scheme uses
% steps of 1 km most of the distance travelled - 10 m layers are optimal
% for absorption loss computation)
Z_interp=Z_direct(1):-0.01:0;
R_interp=spline(Z_direct,R_direct,Z_interp);
S_interp=sqrt((Z_interp(2:end)-Z_interp(1:end-1)).^2+...
    (R_interp(2:end)-R_interp(1:end-1)).^2);

% VOLPE method constants: pure-tone to 1/3 octave-band correction
A=0.867942;
B=0.111761;
C=0.95824;
D=0.008191;
E=1.6;
F=9.2;
G=0.765;

% Compute PT absorption loss
Labs_corr=zeros(length(freq),1);
for dd=1:length(freq)
    abs=0;
for pp=1:length(Z_interp)-1
    Z_mean=.5*(Z_interp(pp)+Z_interp(pp+1));
    Temp=polyval(T,Z_mean);
    hum=polyval(h,Z_mean);
    Press=polyval(P,Z_mean);
    abs_stor=1000*absorption_ANSI(Eff_freq_ratio(pp)*...
    freq(dd),Temp,hum,Press)*S_interp(pp);
    abs=abs+abs_stor;
end

% Compute 1/3 OB absorption loss (VOLPE method)
if abs<150
    Labs_corr(dd,1)=A*abs*(1+B*(C-D*abs))^E;
else
    Labs_corr(dd,1)=F+G*abs;
end
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ANSI S1.26 absorption algorithm

function [alpha_ANSI]=absorption_ANSI(f,T,h,P)

% reference P and T values - see relaxation terms
Pr=101325; % [Pa]
T20=293.15; % [K]
Tr=T/T20;
rhoR=P/Pr;

% 2nd step: given T,P and h at a certain altitude, compute alpha for ...
...specific f
% relaxation terms
fN=rhoR*Tr^(-0.5)*(9+280*h*exp(-4.17*(Tr^(-1/3)-1)));
f0=rhoR*(24+40400*h*(0.02+h)/(0.391+h));
b1=0.1068*exp(-3352/T)/(fN+f^2/fN);
b2=0.01275*exp(-2239.1/T)/(f0+f^2/f0);

% apply formula - this gives the absorption coefficient alpha in dB/m
alpha_ANSI=8.686*f^2*sqrt(Tr)*(1.84e-11*(rhoR)^(-1)+Tr^(-3)*(b1+b2));
end

Sub-routine to compute the shadow zone diffracted pressure field

function[SdZ_level]=shadow_press_enroute(sigma,Ri,Zi,rr,CutField,c0)

% The en route method is based on a secondary source sound with an
% associated power level the corresponds to the cutoff field
Lw=CutField;

% microphone height receivers and source height
zr=1.2; % in m

% freq vector [1/3 OBF]
f=[50 63 80 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000];

% determination of average radius of curvature for complex effective sound
% speed profiles
[Rav Rini]=RlimitAv(Ri,Zi);
Rlim=1000*Rav; % turn Rlim in m

Zs=1.2; % in m
% setting computation for matrix of receiver points
TL=zeros(length(rr),length(f));
GroundLevel=zeros(length(rr),length(f));
SdZ_level=zeros(1,length(rr));

for N=1:length(f)
  % Constants
  k0=2*pi.*f(N)/c0;

  % residue series
  l=(Rlim./(2.*k0.^2)).^(1/3);

  y0=Zs./l;  % in m!!
y=ZR./l;
[Rp Zc]=Ray_ground(90,f(N),sigma);
q=1i*k0*1/Zc;

  x0=-1;
  b=0;
  while length(b)<20
    % Call solver for the equation giving the roots bN.
    [x,fval]=fsolve(@x myfun(x,q),x0,optimset('Display','off'));
    b=horzcat(x,b);
    x0=x-2;  % Adequate recipe to search for zeros without affecting the ... 
              % accuracy.
  end
  b=b(b<0);  %only keep negative zeros.

% ---- residue series solution (Berry & Daigle)
kN=k0.*sqrt(1+b./(k0.^2.*l.^2).*exp(-1i*2*pi/3));
tauN=(kN.^2-k0.^2).*l.^2;
W1_s=2*sqrt(pi)*exp(1i*pi/6).*airy((tauN-y*ones(size(tauN)))).*...
  exp(2*1i*pi/3));

W1=(2*sqrt(pi)*exp(1i*pi/6).*airy((tauN).*exp(2*1i*pi/3))).^2;

for n=1:length(rr)
  W1_r=2*sqrt(pi)*exp(1i*pi/6).*airy((tauN-n.*ones(size(tauN)))).*...
  exp(2*1i*pi/3));
  Zres=1.*W1_s.*W1_r./(W1.*(tauN-q.^2));

% ---- Resulting pressure in shadow zone
P=0;
Pstor=sum(pi*exp(1i*pi/6)./l.^2.*besselh(0,1,kN.*rr(n)).*Zres);
P=P+Pstor;

p=P;
TL(n,N)=20*log10(abs(p));
end
end

% Generate frequency-dependant SPL in shadow zone
for k=1:length(f)
  GroundLevel(:,k)=Lw(k)+TL(:,k);
end

% for each receiver point within the shadow zone proceed to overall Lp[dBA]
for N=1:length(rr)
  GLstor=10.*(GroundLevel(N,:)/10);
  Sum_Sstor=sum(GLstor);
  SdZ_level(1,N)=10*log10(Sum_Sstor);
end
end
Function that generates an average limiting ray radii of curvature

function [Rav Rini]=RlimitAv(R,Z)

L=length(R);
N_point=floor(L/5);

% go over the whole vectors to get the radius arc length
Rcurv=zeros(1,N_point);
dtheta=zeros(1,N_point);

for k=1:N_point
    Rtemp=R(5*(k-1)+1:5*k);
    Ztemp=Z(5*(k-1)+1:5*k);
    [xc,yc,Rcurv_temp,a] = circfit(Rtemp,Ztemp);
    [arclen,seglen] = arclength(Rtemp,Ztemp);
    Stemp=arclen;
    dtheta(1,k)=Stemp/Rcurv_temp;
    Rcurv(1,k)=Rcurv_temp;
end

% apply averaging technique
Num=Rcurv.^2.*dtheta;
Den=Rcurv.*dtheta;

Rav=sum(Num)/sum(Den);

[xc,yc,Rini,a] = circfit(R,Z);
end
Function to solve nonlinear equation in residue series solution

function F = myfun(x,q)
F=airy(1,x)+q*exp(1i*pi/3)*airy(x);
end
Appendix B

Additional codes

This appendix provides the semi-analytical ray-tracing solution based on NORD 2000 and specifically constructed in the context of this thesis to propagate en route aircraft noise. This code has been referenced as the NORD 2000 “finer” method throughout the thesis. Only the 2-dimensional run in the illuminated zone portion of the code are shown here, as the shadow zone wedge diffraction solution was taken from the original NORD 2000 code. Then, the modified HPM (FFP only) code and the TSSR routine are provided.

B.1 The NORD 2000 “finer” code

Contents

- INPUTS
- Extract linear fits in effective sound speed configuration
- Compute the limiting angle
- Compute 2-dimensional sound pressure level decay in illuminated zone
- Sub-routine to compute successive linear fits in atmospheric layers
- Sub-routine to compute limiting angle
- Sub-routine to 2-dimensional SPL decay in illuminated zone
- finally add up all losses
- Sub-routine to compute the EGA in NORD 2000
- Function to compute parameters for EGA computation
• Function to compute EGA parameters along direct ray
• Function to compute EGA parameters along reflected ray
• Launch reflected ray - guessing theta1 & theta2
• Sub-routine to semi-analytically construct rays in layered atmosphere
• Sub-routine to compute ray arc parameters in atmospheric layers
• Sub-routine to compute the travelled time along semi-analytical rays
• Sub-routine to compute the absorption losses along semi-analytical rays
• Sub-routine to compute the refraction losses along semi-analytical rays

% 2-dimensional vertical run using the finer version of NORD 2000

% Developed from March to May 2011

INPUTS

• Lw: 1/3 octave-band sound power level spectrum [dBA]
• zs: source altitude [m]
• zr: receiver altitude [m]
• T: polynomial fit for Temperature [K]
• h: polynomial fit for humidity [% water vapor]
• P: polynomial fit for pressure [Pa]
• W: polynomial fit for wind speed [m/s]
• phi: angle between 2-dimensional vertical plan and wind direction
• h_layer: atmospheric width layer [m] sigma: ground flow resistivity [rayls]
• N_ptssource: number of point sources along the line source

OUTPUT

L: sound pressure level [dBA]

function[L]=N2K_finer(Lw,zs,zr,T,h,P,W,phi,h_layer,sigma,N_ptssource)
Extract linear fits in effective sound speed configuration

\[ [z \ g \ c0]=\text{Layered\_lin}(zs,T,W,\phi,h\_layer); \ % \text{100 m layer is OK} \]

**Compute the limiting angle**

\[ [\text{thetalim}]=\text{limit\_angle\_N2K}(zs,g,c0,h\_layer); \ % \text{100 m layer is OK} \]

**Compute 2-dimensional sound pressure level decay in illuminated zone**

\[
N\_\text{angles}=\text{abs}(89.9+\text{thetalim})/\text{deltatheta};
\]

\[
\text{theta}=\text{linspace}(-89.9,\text{floor(}\text{thetalim})-0.2,\text{N\_angles});
\]

\[
\text{Lp}=\text{zeros}(1,\text{length(}\text{theta}));
\]

\[
\text{range}=\text{zeros}(1,\text{length(}\text{theta}));
\]

\[
\text{for } p=1:\text{length(}\text{theta})
\]

\[
\text{theta}\_\text{launch}=\text{theta}(p);
\]

\[
\text{if } \text{theta}\_\text{launch}=\text{thetalim}
\]

\[
\text{islim}=1;
\]

\[
\text{else } \text{islim}=0;
\]

\[
\text{end}
\]

\[
[L \ \text{range\_point} \ L\text{cutoff}]=\text{dL\_enroute\_spectrum\_finlinear\_illuminated...}
\]

\[
(Lw,T,h,P,zs,zr,g,c0,\text{theta}\_\text{launch},\text{sigma},N\_\text{pts\_source},\text{islim});
\]

\[
\text{Lp}(p)=L;
\]

\[
\text{range}(p)=\text{range\_point};
\]

\[
\text{end}
\]

\[
\text{end}
\]

Sub-routine to compute successive linear fits in atmospheric layers

\[
\text{function}[z \ g \ cbot]=\text{Layered\_lin}(zs,T,W,\phi,h\_layer)
\]
% create z vector, 10 values per layer
z=0:h_layer/10:zs;

% Number of layers?
N_layer=floor(zs/h_layer);

% "new z"
z=0:h_layer/10:N_layer*h_layer;

% sort z in descendant altitudes
z=sort(z,'descend');

% create a matrix from the vector - omission of 1st point...
z_layers=reshape(z(2:end),10,N_layer);

% storage vectors
Temp=zeros(size(z_layers));
Windspeed=zeros(size(z_layers));

[a b]=size(z_layers);

% for all altitudes need to compute the effective sound speed profile
for k=1:a
    for p=1:b
        Temp(k,p)=polyval(T,z_layers(k,p)/1000);
        Windspeed(k,p)=polyval(W,z_layers(k,p));
    end
end

% Once we have Temperature for all altitudes go ahead and compute effective
% sound speed
ceff=20.*sqrt(Temp)+Windspeed.*cos(phi);

% Break down values valid for particular layers

for n=1:N_layer
    % Once have ceff as a function of altitude, proceed to linear fit
    Coeff_lin=polyfit(z_layers(:,n),ceff(:,n),1);

    % extract g and c0 from Coeff_lin
    g(1,n)=Coeff_lin(1);
    cbot(1,n)=Coeff_lin(2);
end
end

Sub-routine to compute limiting angle

function[thetalim]=limit_angle_N2K(zs,g,c0,h_layer)
% initializing
z=0:h_layer:zs;

% at ground, angle of 0 deg - 1st step
  cground=c0(end);
  c_up=c0(end)+h_layer*g(end);
  costheta=c_up/cground;

% loop through all atmospheric layers
for p=1:length(g)-1
    c_down=c0(end-p)+g(end-p)*(p-1)*h_layer;
    c_up=c0(end-p)+g(end-p)*p*h_layer;
    costheta_new=c_up/c_down*costheta;
    costheta=costheta_new;
end

thetalim=acosd(costheta);
if isreal(thetalim)==0
    thetalim=0;
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Sub-routine to 2-dimensional SPL decay in illuminated zone

function[Lp range_point Lcutoff]=dL_enroute_spectrum_finelinear_illuminated...
    (Lw,T,h,P,zs,zr,g,c0,theta_launch,omega,Cv2,Ct2,N_ptsound,isl)

% Code based on COMPRO Version 19ABC, January 4, 2010
% Birger Ploving, DELTA Noise & Vibration (bp@delta.dk)

% Constants (Volpe method: absorption losses for spectrum)
A=0.867942;
B=0.111761;
C=0.95824;
D=0.008191;
E=1.6;
F=9.2;
G=0.765;

"point source along line source" sound power level
Lw=10.*log10(10.^(Lw/10)./N_ptsound);

% 1/3 Octave-band frequencies
freq=[50 63 80 100 125 160 200 250 315 400 500 630 800 1000 1250 1600 2000];

% work out ground loss for geometry setup in all other cases
dLflat=zeros(1,length(freq));
dL_abs=zeros(1,length(freq));
for p=1:length(freq)
    [dL range_point]=flatTerrain_enroute_finer(zs,zr,theta_launch,sigma,g,c0,...
        freq(p),Cv2,Ct2);
dLflat(p)=dL;
    \% -- work out absorption loss for geometry setup
    dL_abs(p)=Abs_loss_curv_finer(g,c0,zs,theta_launch,T,h,P,freq(p));
end

\% Apply PT to OB correction for absorption:
dL_abs_corr=zeros(size(dL_abs));

\% Now take care of PT to OB correction term for absorption losses
for pp=1:length(dL_abs)
    if dL_abs(1,pp)<150
        dL_abs_corr(1,pp)=A.*dL_abs(1,pp).*(1+B.*(C-D.*dL_abs(1,pp))).^E;
        else dL_abs_corr(1,pp)=F+G*dL_abs(1,pp);
    end
end

\% work out refraction loss using Lamancusa's implementation
[GeomLoss]=N2K_refract_finer(zs,theta_launch,0.1,c0,g);

dL=dLflat+dL_abs_corr+GeomLoss;
\% take care of Lcutoff first
if islim==1
    Lcutoff=[];
else
    Lcutoff=dL;
end

\% Add losses to original spectrum to get A-weighted
Lp_freq=Lw+dL;

% Compute Overall A-weighted SPL from Lp spectrum Lp_freq
Lp_temp=10.^(Lp_freq./10);
Lp=10*log10(sum(Lp_temp));
end

/// SUBROUTINE TO COMPUTE THE EGA IN NORD 2000 ///

function [dL range_point]=flat_terrain_enroute_finer(zs,zr,theta_launch,...
  sigma,g,c0,freq,Cv2,Ct2)

  % transversal separation of rays (straight line case) - turbulence only
  rho=2*zs*zr/(zs+zr);

  % call for direct and reflected ray parameters
  [Tau1,Tau2,PsiG,R1,R2,range_point]=calc_flat_par_enroute_finer(zs,zr,...
    theta_launch,g,c0);
  dTau=Tau2-Tau1; % essential for interferences

  for ii=1:length(freq)
    f=freq(ii);
    % Ground impedance using Dela2ny and Bazley model
    Zg=1+9.08*(1000*f/sigma)^(-0.75)+1i*11.9*(1000*f/sigma)^(-0.73);
    k0=2*pi*f/c0(end);
    % calculate coherence coefficients
    Ff=f_calc(dTau,f);
    % Fdt=fdt_calc(dTau,dTauP,f(ii)); % neglect that factor for the moment
    Fc=fc_calc_enroute(Cv2,Ct2,c0,rho,range_point,f);
    % Fr=fr_calc(PsiG,sigmar,k0);
    F=Ff*Fc; % total coherence factor
% calculate Qc
if F>0.001 % coherent propagation included

else
    Qc=0;
end
Qc=calc_qc(Tau2,PsiG,Zg,2*pi*f);
dL(ii)=10*log10(abs(1+exp(1i*2*pi*f*dTau)*Fc*Qc*(R1/R2)^2);
end %ii-loop
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Function to compute parameters for EGA computation

function [Tau1,Tau2,PsiG,R1,R2,range_pt]=calc_flat_par_enroute_finer...
    (zs,zr,theta_launch,g,c0)
% direc ray
[Tau1,R1,range_pt]=dirray_finer(zs,zr,theta_launch,g,c0);

% reflected ray
[Tau2,RS,RR,PsiG,dum]=reflray_finer(zs,zr,theta_launch,g,c0,range_pt);
R2=RS+RR;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Function to compute EGA parameters along direct ray

function [tau,d,range_pt]=dirray_finer(zs,zr,theta_launch,g,c0)

% illuminated zone (theta<limiting angle)
[r z S theta_final]=ray_layered(g,c0,zs,theta_launch,100);
dr=abs(zr*tand(theta_final));
tau_direct=track_time_finer(g,c0,z,S);
tau_minus=sqrt(dr^2+zr^2)/c0(end);
tau=tau_direct-tau_minus;
d=sum(S)-sqrt(dr^2+zr^2);
range_pt=r(end)-dr;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Function to compute EGA parameters along reflected ray

function [tau,rs,rr,psig,r_reflec]=reflray_finer(zs,zr,theta_launch,...
    g_layers,cbot,range_receiver)

sep1=-5e-3;
theta_reflec1=theta_launch+sep1;
[r z S theta_final]=ray_layered(g_layers,cbot,zs,theta_reflec1,100);
% consider theta_final to launch reflected portion neglecting curvature
dz=zr;
   dr=abs(dz*tand(theta_final));
   range_reached1=dr+r(end);
% 2nd ray
sep2=-1e-3;
theta_reflec2=theta_launch+sep2;
[r z S theta_final]=ray_layered(g_layers,cbot,zs,theta_reflec2,100);
% consider theta_final to launch reflected portion neglecting curvature
dz=zr;
   dr=abs(dz*tand(theta_final));
   range_reached2=dr+r(end);

% interpolate for correct launching angle
deltaR1=range_receiver-range_reached1;
deltaR2=range_receiver-range_reached2;
x_coord=[deltaR1 deltaR2];
y_coord=[sep1 sep2];

theta_reflected=theta_launch+spline(x_coord,y_coord,0);

% Get correct reflected ray
[r z S theta_final]=ray_layered(g_layers,cbot,zs,theta_reflected,100);
r_reflec=r(end);
rs=sum(S);

% Initialization
r1=0;
z1=zs;
r=zeros(1,length(g_layers)+1);
z=zeros(1,length(g_layers)+1);

Subroutine to semi-analytically construct rays in layered atmosphere

function[r z S theta_final]=ray_layered(g_layers,cbot,zs,theta_launch,...
    h_layer)

% Initialization
r1=0;
z1=zs;
r=zeros(1,length(g_layers)+1);
z=zeros(1,length(g_layers)+1);
S=zeros(1,length(g_layers));
r(1,1)=0;
z(1,1)=zs;

% walk through each layer in descending order
for p=1:length(g_layers)
    z2=zs-p*h_layer;
    g=g_layers(p);
    c0=cbot(p);
    z_omega=-c0/g;
    [r_omega r2 R]=ray_param_layer(r1,z1,z2,g,c0,theta_launch);

% Re-initialize theta
rtemp=linspace(r1,r2,10000);
ztemp=z_omega+sign(g).*sqrt(R^2-(rtemp-r_omega).^2);

deltar=rtemp(end)-rtemp(end-1);
deltaz=ztemp(end)-ztemp(end-1);

theta_launch=atand(deltaz/deltar);

% store ray trajectory (r,z) and successive travelled distance - in each
% layer
r(1,p+1)=r2;
z(1,p+1)=z2;
S(1,p)=sqrt((r2-r1)^2+(z2-z1)^2);

% Re-initialize r1,z1
r1=r2;
z1=z2;

end
% add extra line to get theta_final, essential for EGA
ztest=[z(end-1) z(end)];
rtest=[r(end-1) r(end)];

dr=r(end)-spline(ztest,rtest,1);
theta_final=atand(dr);
end

Sub-routine to compute ray arc parameters in atmospheric layers

function [r_omega r2 R]=ray_param_layer(r1,z1,z2,g,c0,theta_launch)

% compute sound speed at top of layer
cs=c0+g*z1;

% compute radii of curvature from linear fit
R=cs/(g*cosd(theta_launch));

% ray center altitude
z_omega=-c0/g;

% solve for circle center range
r_omega=r1-sign(g)*sqrt(R^2-(z1-z_omega)^2);

% get back to second point range - equ (2)
r2=r_omega+sign(g)*sqrt(R^2-(z2-z_omega)^2);
end
Sub-routine to compute the travelled time along semi-analytical rays

function [time_track]=track_time_finercgecfezeg(c0,z,S)

% for the finely layered atmosphere formulation, work through all the
% layers
time_travelled=zeros(1,length(g));

for k=1:length(g)
    Ceff_up=c0(k)+g(k)*z(k+1);
    Ceff_down=c0(k)+g(k)*z(k);
end

% from this vector create one that represents the "average effective s.s."
Ceff_mean=mean([Ceff_up Ceff_down]);

% Simply apply basic c=d/t formula:
time_travelled(1)=S(k)/Ceff_mean;
end

% Now add all the travelled times in successive layers:
time_track=sum(time_travelled);
end

Sub-routine to compute the absorption losses along semi-analytical rays

function [deltaL_abs]=Abs_loss_curv_finercgecfezcsetheta_launche5ff(<all for y_track given atmospheric profile and geometry
zt_track y_track 3_track
ray_layeredcgecfezsetheta_launche5ff:<}
alpha_loss=zeros(1,length(Z_track));
for k=1:length(Z_track)-1
    z=Z_track(k)/1000; % attention: z in km here since we are using...
    ...Sutherland/Bass data
    Temp=polyval(T,z);
    humidity=polyval(h,z);
    Press=polyval(P,z);
    alpha_loss(k)=S_track(k)*absorption_ANSI(f,Temp,humidity,Press);
end
% assume the atmospheric layers are 10 m thick (constant for all altitudes)
deltaL_abs=-sum(alpha_loss);
end

Sub-routine to compute the refraction losses along semi-analytical rays

function [GeomLoss]=N2K_refract_finer(zs,theta_launch,sep_angle,c0,g)

% first launch the ray
[r z S]=ray_layered(g,c0,zs,theta_launch,100);

Rdirect=sqrt(r(end)^2+zs^2);
deltaL_geom=-10*log10(4*pi*Rdirect^2);

% % separate upwind and downwind refraction
% % eps=sign(g);

% need to launch 2 very close rays to get ray tube area

[R1 Z1 S1]=ray_layered(g,c0,zs,theta_launch,100);
[R2 Z2 S2]=ray_layered(g,c0,zs,theta_launch+sep_angle,100);
% Start Lamancusa’s method (at receiver point only)
gamma=sqrt((R1(end)-R2(end))^2+(Z1(end)-Z2(end))^2);

% Get equivalent travel distance for homogeneous media (straight line propagation), having travelled for the same time
time=track_time_finercgecfey5e35:<(d5ncfcend:,travel_time<

d1=c0(end)*travel_time;

% Need to get back from receiver separation to launching angle separation
gamma=2*d1*tan(pi*sep_angle/360);

% Compute ray tube areas
Ar=gamma*d1;
Arp=gamma*sum(S1);

% get receiving angle on ground
dZ=Z1(end)-Z1(end-1);
dR=R1(end)-R1(end-1);
theta_end=-atan(dZ/dR);
A_ray=Arp*cosd(-theta_launch-theta_end);

% get alpha_refraction (independent of weather data at ground!)
alpha_R=abs(A_ray/Ar);
deltaL_refrac=-10*log10(alpha_R);
GeomLoss=deltaL_refrac+deltaL_geom;
end
B.2 HPM (FFP only) for en route noise propagation

The inputs are taken from a .txt file which is represented in Figure (B.1). The output is simply the ground contour.

Contents

- HybridPropagationQuicker_enroute Function
- MultiFreqQuickerRev_enroute Function
- Absorption coefficient generation function
- density_enroute function
- Combine_PE_FFP_QuickerRev_enroute Function
• JRFPModelQuickerRev Function

HybridPropagationQuicker_enroute Function

Modified from original HPM written by J. Rosenbaum to compute en route ground contours using the FFP only.

function [] = HybridPropagationQuicker_enroute_final(fileString, caseNumber)
tic
    indexFileString = strfind(fileString, '\');
    if isempty(indexFileString)
        indexFileString = strfind(fileString, '/');
    end

    fileStringModel = fileString(1:indexFileString(length(indexFileString)-1));

    [fileStringModel 'Auxiliary Functions']
    addpath([fileStringModel 'Auxiliary Functions']);

    % Read in inputs from .txt file

    [propagationModel, gridGeometry, xmin, xmax, ymin, ymax, verticalPlot2D, Lw,...
        directionalSource, discrete_or_bands, freqvector, startandstop,...
        broadbandResults, range, height, deg, deltatheta, receiverHeight, deltartPlot,...
        terrain, impedances, inputZorEFR, ZorEFR, lengthfr, atmosphere, Tzero,...
        Temperature, Wind, windDirection, turbulence, realizN, A, KO, CTovTO, Cvovc0,...
        Humidity, Pressure]...
    = readText_enroute([fileString ' Hybrid Model Parameters_enroute.txt']);

    fullbands = [10 12.5 16 20 25 31.5 40 50 63 80 100 125 160 200 250 315 400 ...
                500 630 800 1000 1250 1600 2000 2500 3150 4000 5000 6300 8000 10000];

    startFreqIndex = find(fullbands == startandstop(1));
    stopFreqIndex = find(fullbands == startandstop(2));
freqvector= fullbands(startFreqIndex:stopFreqIndex);

deltatheta= deltatheta*(pi/180);
Windtheta=windDirection*(pi/180);

thetas=[Windtheta:deltatheta:pi+Windtheta];

% when flat terrain and even ground, the wind is the only parameter that...
% ...breaks the axisymmetry

% thetas=0:deltatheta:2*pi;
% Run the model through all the frequencies in freqvector
Level_total_freq=zeros((ymax-ymin)/deltarPlot+2,(xmax-xmin)/deltarPlot+1,...
    length(freqvector));

L_down_tot=zeros(1,ceil(range/deltarPlot),length(freqvector));
for freqnumber= 1:length(freqvector)

    freq= freqvector(freqnumber);
    levelat1m=Lw(freqnumber);

    [Lp_down]=MultiFreqQuickerRev_enroute(freq,propagationModel,gridGeometry,...
        xmin,xmax,ymin,ymax,verticalPlot2D,levelat1m,directionalSource,...
        range,height,deg,deltatheta,thetas,...
        receiverHeight,deltarPlot,terrain,impedances,inputZorEFR,ZorEFR,...
        lengthefr,atmosphere,Tzero,Temperature,Wind,windDirection,...
        turbulence,realizN,A,...
        KO,CTovT0,Covc0,Humidity,Pressure,...
        fileString,caseNumber);
end

% apply log averaging

temp=10.^((L_down_tot./10));

temp=10.^((Level_total_freq./10));

tot=10*log10(sum(temp,3));
% Save the broadband results with the x_als and y_als variables to
% allow for easier future plotting
save([fileString,broadbandName,'broadband.mat'], 'tot', 'xgrid', 'ygrid', ...
    'freqvector', 'L_down_FINAL');

% HORIZONTAL GROUND CONTOUR
figure(1)
selgcf,'color','white')
contourf(xgrid./1000,ygrid./1000,tot,'EdgeColor','none')
title('Transmission Loss [dB]')
xlabel('X [km]')
ylabel('Y [km]')
caxis([0 50])
set(gca,'fontsize',16)
saveas(gcf,[fileString levelPlotName '_figure.fig'])
cpu_final=toc;
end

MultifreqQuickerRev_enroute Function

function [L_down]= MultifreqQuickerRev_enroute_final(freq,propagationModel,...
    gridGeometry,xmin,xmax,ymin,ymax,verticalPlot2D,levelat1m,...
    directionalSource,range,height,deg,deltatheta,thetas,receiverHeight,...
    deltarPlot,terrain,impedances,...
    inputZorEFR,ZorEFR,lengthfr,atmosphere,Tzero,temperature,Wind,...
    windDirection,turbulence,realizN,A,K0,CToV0,CovV0,...
    Humidity, Pressure,fileString,caseNumber)
% from T0 get sound speed on ground
czero=sqrt(1.4*8314*Tzero/29);

omega= 2*pi*freq;
kzero= omega/czero;
lambda= 2*pi/kzero;

%deltar must be less than 1/10 of a wavelength
deltarlim = .1*lambda;
%determine deltar that will be divisor of deltarPlot
deltar= deltarPlot/ceil(deltarPlot/deltarlim);
clear deltarlim

%deltaz can be increased when using the FFP only
deltaz=200;
%corrected height so that height is integer multiple of deltaz and deltaz
%is divisor of receiver height
height= ceil((height)/deltaz)*deltaz;
clear deltazlim

z_FFP= [0 receiverHeight:deltaz:height];
zN_FFP= length(z_FFP);

% Define the flight path, reading in 'User Defined Flight Path.xls'
[flightTrack]= xlsread([fileString 'User Defined Flight Path new.xls']);
xflightPositions= flightTrack(:,1).';
yflightPositions= flightTrack(:,2).';
zflightPositions= flightTrack(:,3).';

% Compute and/or assign the ground impedance
    Z= ZorEFR;
% Compute the atmospheric absorption coefficient
[betaAtmAbs] = airatten_2006_Function_enroute(freq,z_FFP,Temperature,...
            Humidity,Pressure);

% Compute density as function of altitude (necessary for en route noise
% propagation) - added line on MAY 26 2011
[density] = density_enroute(Temperature,Pressure,z_FFP);

% Contouring (horizontal sound field) runs
% *Define 2D horizontal (Cartesian) grid for contours*

xminTerrain = xmin;
xmaxTerrain = xmax;
ymaxTerrain = ymax;

% Set the x and y vectors taken the fact that theta spans from 0 to pi
% and not up to 2*pi
xVals = xminTerrain:deltarPlot:xmaxTerrain;
xValsN = length(xVals);
yVals = 0:deltarPlot:ymaxTerrain;
yValsN = length(yVals);

% Construct the contouring grid
[Xgrid,Ygrid] = meshgrid(xVals,yVals);
% number of represented points along the flight path
sourceN = length(xflightPositions);
% number of runs needed to cover all azimuthal angles from source
runN = length(thetas);
% initialize matrix to contain contour
LevelAll = zeros(yValsN,xValsN,sourceN);

rangesAtThetas = range*ones(1,length(thetas));
rho = 0:deltar:deltar*ceil(rangesAtThetas(1)/deltar);
polarTerrain= cell(runN,1);
for pp=1:runN
    polarTerrain(pp)=zeros(1,length(rho));
end
impedancesAtThetas=ones(runN,1);
ZAtThetas=Z*ones(runN,1);
lengthefrAtThetas=NaN*ones(runN,2);

% wind projection in 2D planes
Wind_projected=zeros(length(Wind),length(thetas));
for k=1:length(thetas)
    % vector of sound speed profile parameters due to wind for each
    % azimuthal angle - projection of a polynomial fit is OK
    Wind_projected(:,k)=Wind.*cos(thetas(k)-(windDirection*pi/180));
end

heightwabs=NaN;
zN=zN_FFP;
z=z_FFP;

[receiver,strArraySSP]= Combine_PE_FFP_QuickerRev_enroute...
    (propagationModel,verticalPlot2D,directionalSource,terrain,...
    atmosphere,impedancesAtThetas,freq,czero,rangesAtThetas,height,...
    zflightPositions(1),deltarPlot,deg,ZAtThetas,lengthefrAtThetas,...
    Temperature,Wind_projected,receiverHeight,...
    runN,omega,kzero,deltar,deltaz,heightwabs,zN,z,zN_FFP,z_FFP,...
    betaAtmAbs,turbulence,realizN,A,K0,CTovT0,Cvovc0,levelat1m,caseNumber,...
    polarTerrain,sourceN,density);

% Save data in case computer crashes
save([fileString num2str(freq) ' Hz freq.mat’],’receiver’);
% Save donwind run for comparison report
L_down=receiver{1};
% *Run the model for all the represented points along the flight path*
for l=1:sourceN
% *Prepare grid and matrices necessary for combining sound fields for *
% different sources, and plotting*
% convert plotting grid to polar coordinate system
[thetaGrid,rGrid] = cart2pol(Xgrid-xflightPositions(l),...
    Ygrid-yflightPositions(l));
Level= zeros(yValsN,xValsN);

% *Interpolate from polar coordinate grid to Cartesian coordinate grid* 
for mm= 1:yValsN %represent the y axis.
    for n= 1:xValsN %represent the x axis

    if rem(rGrid(mm,n),deltarPlot)==0

        if rem(thetaGrid(mm,n),deltatheta)==0
            % if the cartesian grid point falls exactly on a polar
            % gridpoint, then the level at that cartesian gridpoint
            % equals the level at the corresponding polar gridpoint

            % May 23 new line, assign neg value if out of bound
            % value
            if round(rGrid(mm,n)/deltarPlot+1)>length(receiver... 
                {round(thetaGrid(mm,n)/deltatheta+1)})
                Level(mm,n)=-150;
            else
            Level(mm,n)= receiver{round...
                (thetaGrid(mm,n)/deltatheta+1)}(round...
                (rGrid(mm,n)/deltarPlot+1));
            end
        end
    end
%otherwise, if the cartesian gridpoint falls exactly on the r value, but between two theta values of the polar coordinate system:
%set the two polar coordinate system theta values that this theta is between
floorTheta= deltatheta*floor... (thetaGrid(mm,n)/deltatheta);
ceilTheta= floorTheta+deltatheta;
%find the coordinates A and B where A and B can be found in the Numerical Recipes book, p. 107
Ainterp= (ceilTheta-thetaGrid(mm,n))/deltatheta;
Binterp= 1-Ainterp;
%interpolate the level at the r value between two values of theta
...two values of theta

if round(rGrid(mm,n)/deltarPlot+1)>length...
    (receiver{round(floorTheta/deltatheta+1)})... || round(rGrid(mm,n)/deltarPlot+1)>...
    length(receiver{round(ceilTheta/deltatheta+1)})
    Level(mm,n)=-150;
else

    Level(mm,n)= 10*log10(Ainterp*10^... (receiver{round(floorTheta/deltatheta+1)})...
    (round(rGrid(mm,n)/deltarPlot+1))/10)+... Binterp*10^-(receiver{round(ceilTheta/deltatheta+1)})...
    (round(rGrid(mm,n)/deltarPlot+1))/10));
end
end

elseif rem(thetaGrid(mm,n),deltatheta)==0
if the cartesian gridpoint falls between two r values in the polar coordinate system but exactly on a theta value:

set the two polar coordinate system r values that this r ... is between

floorR = deltarPlot*floor(rGrid(mm,n)/deltarPlot);
ceilR = floorR+deltarPlot;

find the coordinates A and B where A and B can be found in the Numerical Recipes book, p. 107

Ainterp = (ceilR-rGrid(mm,n))/deltarPlot;
Binterp = 1-Ainterp;

interpolate the level at the theta value between two... values of r

if round(floorR/deltarPlot+1)>length(receiver... {round(thetaGrid(mm,n)/deltatheta+1)}) || ...
round(ceilR/deltarPlot+1)>length...
(receiver{round(thetaGrid(mm,n)/deltatheta+1)})

Level(mm,n)=-150;
else

Level(mm,n) = 10*log10(Ainterp*10^-... (receiver{round(thetaGrid(mm,n)/deltatheta+1})...
(round(floorR/deltarPlot+1))/10)+...
Binterp*10^-(receiver{round...
(thetaGrid(mm,n)/deltatheta+1})...
(round(ceilR/deltarPlot+1))/10));
end
else

if the cartesian gridpoint falls between both two r values and two theta values in the polar coordinate system:

set the two polar coordinate system theta values that this ...
theta is between
floorTheta= deltax>Theta<floor(thetaGrid(mm,n)/deltatheta);
ceilTheta= floorTheta+deltatheta;
%set the two polar coordinate system r values that this ...
r is between
floorR= deltax>Plot<floor(rGrid(mm,n)/deltax>Plot);
ceilR= floorR+deltax>Plot;
%find the coordinates A and B along an r where A and B ...
can be found in the
%Numerical Recipes book, p. 107
AinterpR= (ceilTheta-thetaGrid(mm,n))/deltatheta;
BinterpR= 1-AinterpR;
%interpolate the level at the lower r value between to ...
two values
%of theta
% May 23 new line, assign neg value if out of bound
% value
if round(floorR/deltax>Plot+1)>length(receiver...
{round(floorTheta/deltatheta+1)}) || ... 
round(floorR/deltax>Plot+1)>length(receiver...
{round(ceilTheta/deltatheta+1)}) || ... 
round(ceilR/deltax>Plot+1)>length...
(receiver{round(ceilTheta/deltatheta+1),1}) ... || round(ceilR/deltax>Plot+1)>length...
(receiver{round(floorTheta/deltatheta+1),1})
Level(mm,n)=-150;
else
LevelLR= 10*log10(AinterpR*10^(receiver...
{round(floorTheta/deltatheta+1),1})...
(1,round(floorR/deltax>Plot+1))/10)+...
BinterpR*10^(receiver{round(ceilTheta/deltatheta+1),1})
(1,round(floorR/deltarPlot+1))/10));

% interpolate the level at the upper r value between to two values of theta
LevelUR= 10*log10(BinterpR*10^(receiver...
{round(floorTheta/deltatheta+1),1})...
(1,round(ceilR/deltarPlot+1))/10)+...
BinterpR*10^(receiver{round(ceilTheta/deltatheta+1),1})
(1,round(ceilR/deltarPlot+1))/10));
% find the coordinates A and B between the r's at the correct theta value where A and B can be found in the ... Numerical Recipes book, p. 107
AinterpTheta= (ceilR-rGrid(mm,n))/deltarPlot;
BinterpTheta= 1-AinterpTheta;
% interpolate the level at the correct theta value between to two values of r
Level(mm,n)= 10*log10(AinterpTheta*10^(LevelLR/10)+...
BinterpTheta*10^(LevelUR/10));
end
end
end

LevelAll(:,3,1)= Level;
end


% 
% *Logarithmically add the sound levels for each source along the flight path*
levelTotal= 10*log10(sum(10.(LevelAll/10),3));

% first step: get rid of 0 values that come from interpolation boundaries
% then proceed to the symmetry construction
[a b]=size(levelTotal);
levelTotal=levelTotal(1:a,1:b);
[a b]=size(levelTotal);
levelTotal_contour=zeros(2*a,b);
levelTotal_contour(a+1:2*a,:)=levelTotal;

for p=1:a
    levelTotal_contour(p,:)=levelTotal(a-p+1,:);
end

% Construct vector of Levels taking the symmetry with respect with the wind
% construct TL_contour from TLstor

% proceed to the symmetry construction
[a b]=size(levelTotal);
levelTotal=levelTotal(1:a,1:b);
[a b]=size(levelTotal);
levelTotal_contour=zeros(2*a,b);
levelTotal_contour(a+1:2*a,:)=levelTotal;

for p=1:a
    levelTotal_contour(p,:)=levelTotal(a-p+1,:);
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Absorption coefficient generation function

function [beta]= airatten_2006_Function_enroute(f,Z, Temperature, ... 
    Humidity, Pressure)
% Altitude dependency of T and h, and p
% new line: Temp and humidity altitude dependency
t=polyval(Temperature,Z./1000);

h=polyval(Humidity,Z./1000);

% Pressure variation with altitude (minor seasonal variations)
ps = polyval(Pressure, Z./1000); % from Pa to bar
ps = 1;
to = 293.15;
ff = f./ps;

fro = (ps/pso).*(24 + 4.04e4.*h.*((h+0.02)/(h + 0.391)));
frn = (ps/pso).*sqrt(to/t).*((9 + 280.*h.*exp(-4.17.*...
     ((to/t).^0.333 - 1))));

aoverps = ff.^2./pso.*(1.84e-11.*sqrt(t/to) + (t/to).^2.5.*...
     (0.01278.*exp(-2239.1./t)./(fro + ff.^2./fro) + 0.1068.*exp(-3352./t)...
     /(frn + ff.^2./frn)));

alpha = aoverps.*ps;
a = 8.686.*alpha;
beta = a./(20*log10(exp(1)));
end

density_enroute function

function [density]=density_enroute(T,P,z_FPP)

% constants
M=29; % for air
R=8314;

% direct calculation
Temp=polyval(T,z_FFP./1000);
Pressure=1e5.*exp(polyval(P,z_FFP./1000));

density=Pressure.*M./(R.*Temp);
end

Combine_PE_FFP_QuickerRev_enroute Function

function [receiver,strArraySSP]= Combine_PE_FFP_QuickerRev_enroute_final...
(propagationModel,verticalPlot2D,directionalSource,terrain,atmosphere,...
impedancesAtThetas,freq,czero,rangesAtThetas,height,...
zflightPositions,deltarPlot,deg,ZAtThetas,lengthfrAtThetas,...
Temperature,Wind_projected,receiverHeight,...
runN,omega,kzero,deltar,deltaz,heightwabs,zN,z,zN_FFP,z_FFP,...
betaAtmAbs,turbulence,realizN,A,KO,CTovTO,Cvovc0,levelat1m,...
caseNumber,polarTerrain,sourceN,density)

levelat1m=10*log10(10^(levelat1m/10)/sourceN);
%initialize the cell array to contain results for the runs at different
%azimuthal angles
receiver= cell(runN,1);

% *Assign necessary parameters for each azimuthal angle*
for n= 1:runN
    runN-n;

    range= rangesAtThetas(n)-deltar;
    Wind= Wind_projected(:,n); % Wind speed vector at angle theta(n)
strArraySSP(n)= cellstr(' ');

[receiver{n,1}]= JRFFPModelQuickerRev_enroute(verticalPlot2D,...
  atmosphere,freq,czero,range,height,zflightPositions,...
  ZAtThetas(1),Temperature, Wind,...
  receiverHeight,n,omega,kzero,deltarPlot,deltar,deltaz,...
  zN_FFP,z_FFP,betaAtmAbs,levelat1m,density);

end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

JRFFPModelQuickerRev_enroute Function

function [receiverFFP,receiverFFPgrid]= JRFFPModelQuickerRev_enroute...
  (verticalPlot2D,atmosphere,freq,czero,range,height,sourceHeight,...
   Z,Temperature, Wind,...
   receiverHeight,runNumber,omega,kzero,deltarPlot,deltar,...
   deltax,zN,z,betaAtmAbs,levelat1m,density)

  % Define the atmosphere: calculate the sound speed profile and apply atmospheric

  [cz,grad]= AtmosphereRev_enroute(czero,z,Temperature,deltaz,Wind);
  k= omega./cz+j.*betaAtmAbs(1:zN);

  % define deltarFFP according to minimum of effective sound speed - taking 3
  % points per wavelength (2 points per wavelength being the minimum for FFT
  % IFFT calculation)
  deltarFFP=min(cz)/freq/3;
% The FFP requires much larger ranges to be calculated than will be used
range= 3*range; %defined by ES

% Define the range vector
r= 0:deltarFFP:range;
rN= length(r);

Fast-Field Program code follows closely to the description of the FFP method in

deltak= 2*pi/(deltarFFP*rN);
ks= .5*deltak:deltak:ceil(3*omega/(czero*deltak)) deltak+.5*deltak;

%Salamons p.159 says to use deltak for kt
kr= ks-j*deltak;
krN= length(kr);

KZI= (k(1)^2-kr.^2).^(.5);

Sdelta= sqrt(2*pi*kr); %Sdelta is 1 x krN
Rkz1= (KZI-k(1)/Z)./(KZI+k(1)/Z); %Rkz1 is 1 x krN

m= round(sourceHeight/deltaz+2);
kz0= (k(m)^2-kr.^2).^(.5);
% modify deltaz
Deltaz=[receiverHeight deltaz-receiverHeight deltaz*ones(1,length(z)-3)];

if verticalPlot2D==1
    Pall= zeros(zN,krN);
    Pall(1,:)= Rkz1+1;
end
% boundary conditions on ground
P = Rkz1+1;
dP = j*KZ1.*(Rkz1-1);

% going up from ground to source altitude
for n = 1:m-1

  deltaz=Deltaz(n);
  %n signifies the level number
  KZn = (k(n)^2-kr.^2).^(.5);
  Plzplus1 = cos(KZn*deltaz).*P+(KZn).^(-1).*sin(KZn*deltaz).*dP;

  dPlzplus1 = -KZn.*sin(KZn*deltaz).*P+cos(KZn*deltaz).*dP;
  P = Plzplus1.*exp(j*kz0*deltaz);
  if verticalPlot2D==1
    Pall(n+1,:) = P;
  end

  dP = density(n)/density(n+1)*dPlzplus1.*exp(j*kz0*deltaz);
  %essentially if they are equal, allowing for roundoff errors
  if abs(z(n+1)-receiverHeight)<.00001
    Preceiver = P;
  end

end

% values at source location from bot to top computation
Pml = Plzplus1;
%Pml is 1 x krN
dPml = dPlzplus1;
%dPml is 1 x krN

% start top to bot loop
P = ones(1,krN);
if verticalPlot2D==1
    Pall(zN,:) = P;
end

dP = j*(k(zN)^2-kr.^2).^(.5);

Puzplus1 = P.*exp(j*kz0*deltaz);
dPuzplus1 = dP.*exp(j*kz0*deltaz);

for n = zN-1:-1:m
    deltaz = Deltax(n);
    % n signifies the level number
    KZn = (k(n)^2-kr.^2).^(.5);
    P = cos(KZn*deltaz)*Puzplus1+(KZn).^(-1).*sin(KZn*deltaz).*dPuzplus1;
    if verticalPlot2D==1
        Pall(n,:) = P;
    end
    dP = -KZn.*sin(KZn*deltaz)*Puzplus1+cos(KZn*deltaz).*dPuzplus1;
    if n>0
        Puzplus1 = P.*exp(j*kz0*deltaz);
        dPuzplus1 = density(n)/density(n-1)*dP.*exp(j*kz0*deltaz);
    end
    % essentially if they are equal, allowing for roundoff errors
    if abs(z(n)-receiverHeight)<.00001
        Preceiver = P;
    end
end

% final values at source location from top to bot loop
Pmu = P; % Pmu is 1 x krN
dPmu = dP; % dPmu is 1 x krN

% scaling
Pm = -Sdelta ./((dPmu./Pmu)-(dPml./Pml));

if verticalPlot2D==1
    Pall(1:m-1,:) = Pall(1:m-1,:).*(((exp(j*abs(z(1:m-1)-
        sourceHeight).*kz0)).*(ones(m-1,1)*Pm))./(ones(m-1,1)*Pml)); ...
        ...Pm is 1 x krN
    Pall(m+1:zN,:) = Pall(m+1:zN,:).*(((exp(j*abs(z(m+1:zN)-
        sourceHeight).*kz0)).*(ones(zN-m,1)*Pm))./(ones(zN-m,1)*Pmu));
    Pall(m,:) = Pm;
end
receiverHeightN = receiverHeight/deltaz+1;

if receiverHeightN<m
    Preceiver = Preceiver.*(exp(j*abs(receiverHeight-sourceHeight).*kz0))... 
        .*Pm./Pml;
elseif receiverHeightN>m
    Preceiver = Preceiver.*(exp(j*abs(receiverHeight-sourceHeight).*kz0))... 
        .*Pm./Pmu;
elseif receiverHeight==m
    Preceiver = Pm;
end

clear dP Plzplus1 dPlzplus1 Puzplus1 dPuzplus1 Pm Pml dPml Pmu dPmu Rkz1 ...
    Sdelta ks;

% New line (remove Window and NaN values)
temp=Preceiver;
ind_NAN=isnan(temp);
temp(ind_NAN)=0;

% Transform back to the spatial domain to calculate the pressure
pc = (((1-i)/(pi*sqrt(2)))*(exp(deltak*r*(1-(i/2)))*(2*pi/deltarFFP).*ifft...
         (temp,rN)+exp(-deltak*r*(1-(i/2)))*(deltak).*fft(temp,rN)))./sqrt(r);
% Calculate the sound level in dB
receiverFFP= levelat1m+20*log10(abs(pc(2:floor(range/(3*deltarFFP)+1))).../
-abs(exp(j*kzero)))
receiverFFP= receiverFFP(rindex);
deltarPlot/deltarFFP;
floor(range/(3*deltarFFP)+1);
end

B.3 The TSSR code

The TSSR code provides a way to compute the excess attenuation (relative to a homogeneous atmosphere) in an acoustic shadow zone, in the presence of atmospheric turbulence.

Contents

- INPUTS
- OUTPUT
  - Sub-routine to compute the turbulent coherence factor

% Turbulent Scattered Shadow Ray (TSSR) method by Dr. Lam (2009)

INPUTS

- f: frequency [Hz]
- Rav: limiting ray radii of curvature [m]
- r: receiver range from secondary source [m]
- zr: receiver altitude [m]
- zs: secondary source altitude [m]
• $c_0$: ground sound speed [m/s]

• $\mu_2$: the square of the standard deviation of the fluctuating part of the index of refraction for a Gaussian turbulence model

• $N$: number of realizations (snapshots)

• $\text{diffmethod}$: Maekawa or Menounou

• $Z_g$: ground impedance

**OUTPUT**

$EA_{turb}$: Excess Attenuation in shadow zone relative to the propagation in a homogeneous atmosphere [dBA]

```
function [EA_turb] = TSSR(f, Rav, r, zr, zs, c0, mu2, N, diffmethod, Zg)

% Initial setup
% analytical solution giving range from source to start of shadow zone
rini = sqrt(2 * Rav * zs);

k = 2 * pi * f / c0;
lambda = c0 / f; % wavelength

% According to Dr Lam's method, diffpath computed from r1 and d0...
... (see Fig.1 of paper); d0 is the range within the shadow and...
... $r_1 = \sqrt{s_2^2 + hr^2}$, $s_2$ being the distance along the ray and $hr$...
... is the ray height

r2 = r - rini; % range within the shadow zone
```
% 1st step: implement turbulence
% reference values for the calculation
mu2ref=3e-5;

% Get ray height standard deviation from regression curve
% regression coefficients
a11=0.13;
a31=0.1;
a32=3.2e-5;
a33=-1.2e-9;
% apply different fit according to range in shadow zone
if r2<=1000
    dZstd=sqrt(mu2/mu2ref)*a11*r2;
else
    dZstd=sqrt(mu2/mu2ref)*(a31*r2+a32*r2^2+a33*r2^3);
end

% with the given ray height standard deviation, generate a random ray
% height change using a Gaussian distribution with mean 0
dZ=normrnd(0,dZstd,[1,N]);

% separate cases where z_ray < 1km and z_ray > 1km
rcrit=sqrt(Rav^2-(Rav-1000)^2);

if r2<=rcrit
    z_ray=Rav-sqrt(Rav^2-r2^2)+dZ;
s2=Rav*asin(r2/Rav);
else
    theta=90-asind((Rav-1000)/Rav);
deltar=r2-rcrit;
z_ray=1000+deltar*tand(theta)+dZ;
s2=Rav*asin(rcrit/Rav)+sqrt(deltar^2+(deltar*tand(theta))^2);
end
hr=z_ray-zr;

% get rid of all negative ray heights and assign a 0 value for diffraction
[a b]=find(hr>0);
N_zero=length(hr)-length(b);
hr(hr(b));

% Once we have all these quantities, compute the path difference & Fresnel
% Number N
r1=sqrt(s2.^2+hr.^2);

% choice of diffraction method
if strcmp(diffmethod,'Maek')
diffpath=r1-r2;
NFresnel=2*diffpath./lambda; % Fresnel Number

% Finally get the IL using Maekawa’s empirical fit
IL_diff=horzcat(10.*log10(3+20.*NFresnel),zeros(1,N_zero));
else
% Menounou’s solution
phi=3*pi/2+asin(hr./r1);
phi0=pi/2;

% Fresnel numbers: N1 and N2
N1=2*k*r2/pi.*cos((phi-phi0)/2).^2;
N2=2*k*r2/pi.*cos((phi+phi0)/2).^2;

% apply Menounou’s formula - provides diffraction losses (keeping notations
% from the paper)
ILs=20.*log10(sqrt(2*pi.*N1)./tanh(sqrt(2*pi.*N1)))-1;
ILb=20.*log10(abs(1+tanh(0.6*log10(N2./N1))));
ILsb=(6.*tanh(sqrt(N2))-2-ILb).*(1-tanh(sqrt(10.*N1)));
IL_diff=horzcat(ILs+ILb+ILsb+3,zeros(1,N_zero));
end

% 3rd step: ground interaction term
% Use mutual coherence in the presence of turbulence
[T]=turb_coh_fact(zs,zr,1.1,f,c0,mu2,r2);

% Computing direct and reflected path lengths
Rdirect=sqrt(r2^2+(zs-zr)^2);
Rreflected=sqrt(r2^2+(zs+zr)^2);

% Call for plane-wave reflection coefficient (the en route geometry calls
% for large enough distances within the shadow zone to consider plane-wave
% reflection coefficients)
Graz_angle=asind(Rdirect/Rreflected);
Rp=Ray_ground(Graz_angle,f,sigma);

% turbulence in the next step
A1=exp(-absorption_ANSI(f,293,70)*Rdirect/8686);
A2=exp(-absorption_ANSI(f,293,70)*Rreflected/8686);
p2=(A1/Rdirect)^2+(abs(Rp)*A2/Rreflected)^2+T*2*A1*A2*abs(Rp)*...cos(k*(Rreflected-Rdirect)+angle(Rp))/(Rreflected*Rdirect);

TL_turb=10*log10(p2);

% add up both losses to compute TL
TL=-IL_diff+TL_turb;

% 4th step: Proceed to energy averaging of these values to produce a...
...single TL value

TL_stor=10.*(TL./10);
% sum values and apply log to get TLav_turb - relate to Excess Attenuation
TL_turb_av=10*log10(sum(TL_stor)./N);

r_direct=sqrt(r^2+(zs-zr)^2);

% compute EA in the presence of turbulence - absorption taken into account
EA_turb=TL_turb_av+20*log10(r_direct)+absorption_ANSI(f,293,70)...
    *r_direct/1000;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Sub-routine to compute the turbulent coherence factor

% Method carried by Daigle
function [T]=turb_coh_fact(zs,zr,L0,f,c0,mu2,r)

% constants
k=2*pi*f/c0;

% height separation between 2 adjacent rays
h=1/(1/2*(1/zs+1/zr));

% rho
rho=sqrt(pi)/2*L0/h*erf(h/L0);

% second formulation: Daigle (referenced by Sutherland)
D=r/(k*L0^2);
B=((1+1/D^2)^0.5)-1;
M=D*(2*B)^0.5;
A=(1/(D^2*(B+1)*(8*B)^0.5))*(D*B/2)*log((1+M)/(1-M))+atan(D*B/(1-M))+...
    atan(D*B/(1+M));
sigma2=sqrt(pi)*mu2*k^2*r*L0*(1+A)/2;
\[ T = \exp(-\sigma_2^* (1 - \rho)) \]

end
Example of runs

This appendix provides examples of runs using the NOAA data processing code, AEROM and NORD 2000’s formulation in the illuminated zone.

### C.1 Running the NOAA atmospheric data code

The first step is to run the NOAA_process code shown in appendix A. In this particular example, the Pittsburgh 2010 winter seasonal atmospheric averaged data is considered. After having downloaded the .dat file on the NOAA ftp website corresponding to the Pittsburgh weather station, the NOAA_process code is run with inputs ('75520.dat','Pittsburgh.mat'). This run can take a few minutes. Then, the Weather_pfit code is run in order to compute the polynomial fits for pressure, temperature, humidity and wind speed as a function of altitude in km. Considering the 2010 atmospheric averaged data above Pittsburgh, the inputs to this code are: (2010,[1,2,3],2,'Pittsburgh.mat'). The corresponding outputs are shown in Table (C.1), where for each atmospheric parameter X:

\[ X(z) = p_0 + p_1 \times z + p_2 \times z^2 + p_3 \times z^3 + p_4 \times z^4 + p_5 \times z^5 + p_6 \times z^6 \]

for \( z \) in km.
Table C.1: Polynomial coefficients for the 2010 winter atmospheric averaged data above Pittsburgh, Pa.

<table>
<thead>
<tr>
<th></th>
<th>p5</th>
<th>p4</th>
<th>p3</th>
<th>p2</th>
<th>p1</th>
<th>p0</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>0</td>
<td>0</td>
<td>-2.9E+01</td>
<td>7.5E+02</td>
<td>-1.1E+04</td>
<td>9.5E+04</td>
</tr>
<tr>
<td>T</td>
<td>-1.1E-03</td>
<td>4.7E-02</td>
<td>-7.2E-01</td>
<td>4.8E+00</td>
<td>-1.8E+01</td>
<td>2.8E+02</td>
</tr>
<tr>
<td>h</td>
<td>0</td>
<td>0</td>
<td>-2.0E-04</td>
<td>6.3E-03</td>
<td>-6.8E-02</td>
<td>2.9E-01</td>
</tr>
<tr>
<td>W</td>
<td>0</td>
<td>0</td>
<td>5.9E-03</td>
<td>-1.7E-01</td>
<td>3.4E+00</td>
<td>8.2E+00</td>
</tr>
</tbody>
</table>

C.2 Running AERNOM

This section provides the inputs in order to generate the ground contour shown on the left of Figure (5.24) in section (5.7.8):

- The A-weighted source power level is shown in Table (4.1).
- The value for DIR is taken to be 0°.
- The Mach number is $M_c = 0.8$.
- $z_s = 10$ km.
- $z_r = 1$ m.
- $\delta R = 0.5$ km.
- $\delta R_{grid} = 1$ km.
- $X_{max} = 40$ km.
- $Y_{max} = 40$ km.
- $\delta \phi = \frac{\pi}{5}$ radians.
- $\delta \theta = 1°$.
- $N_{ptsources} = 100$.
- The polynomial fit for $T$ is taken from [30].
- $W = [0, 0, 0, 5, 0]$ for a linear wind speed profile.
- The polynomial fit for $h$ is taken from [30].
The polynomial fit for \( P \) is taken from [30].

Bass & Sutherland do not handle some of the atmospheric data the same way that the NOAA atmospheric data is taken into account. In particular, for the humidity vapor polynomial fit, the coefficients are used in the following expression:

\[
h = 100 \times 10^{4A_0 + p_1 \times z + A_2 \times z^2 + A_3 \times z^3 + A_4 \times z^4 + A_5 \times z^5 + A_6 \times z^6}
\]

whereas the coefficients for the pressure fit are used in the following expression:

\[
P = 10^5 \times \exp(A_0 + A_1 \times z + A_2 \times z^2 + A_3 \times z^3 + A_4 \times z^4 + A_5 \times z^5 + A_6 \times z^6)
\]

### C.3 Running NORD 2000

This section provides the inputs in order to generate the sound pressure level decay shown on Figure (5.9) in section (5.4.1):

- The A-weighted source power level is shown in Table (4.1).
- \( z_s = 10 \) km.
- \( z_r = 1 \) m.
- The polynomial fit for \( T \) is taken from [30].
- \( W = [0, 0, 0, 0.003, 0] \) for a linear wind speed profile.
- The polynomial fit for \( h \) is taken from [30].
- The polynomial fit for \( P \) is taken from [30].
- \( \phi = 0^\circ \) for a downwind refracting type propagation.
- \( h_{layer} = 5000 \) m.
- \( N_{sources} = 1 \).
Bibliography


[59] URL [www.aircraftnoisemodel.org](http://www.aircraftnoisemodel.org)
