ICE PARTICLE TRAJECTORY SIMULATION

A Thesis by

Ian Greg Suares

B.Tech., Indian Institute of Technology Madras, 2002

Submitted to the College of Engineering
And the faculty of the Graduate School of
Wichita State University in partial fulfillment of
the requirements for the degree of
Master of Science

December 2005
I have examined the final copy of this Thesis for form and content and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science in Aerospace Engineering.

______________________________

Dr. Michael Papadakis, Committee Chair

We have read this Thesis and recommend its acceptance:

______________________________

Dr. Kamran Rokhsaz, Committee Member

______________________________

Dr. Alexandre Boukhgueim, Committee Member
ACKNOWLEDGEMENTS

First and foremost I would like to thank my advisor, Dr. Papadakis, for his guidance and support throughout my Masters program. I am also grateful to him for giving me the opportunity to work on this research project which has helped me increase my knowledge and develop new skills. I would like to extend my gratitude to Dr. Kamran Rokhsaz and Dr. Alexandre Boukhgueim for serving on the committee.

I would like to thank the Aircraft Design & Manufacturing Research Center (ADMRC) for proving the grant to conduct this research. I am grateful to my colleagues Hsiung-Wei and Joel who worked with me in this project.

I am grateful to all my friends especially Joel, Ballu and Manoj for their support and companionship. I thank the Newman Center for providing a home away from home. Finally, I thank my parents and my brother for their constant love and support which enabled me to achieve this Master’s degree.
ABSTRACT

Large ice particles shed from an airframe can cause damage to downstream aerodynamic surfaces and aft mounted engines. A simulation tool was developed to compute the trajectories of shed ice particles and determine the probability of these particles passing through a particular downstream location. The flowfield into which the ice particles shed was determined using CFD. The aerodynamic forces and moments acting on the ice particles were obtained from published literature and experimental data. Three, four and six degree of freedom trajectory simulation models were developed to compute the trajectories of shed ice particles. Monte Carlo simulations were performed by varying the aerodynamic and geometric properties of the ice particles to obtain a probability map depicting the regions where the ice particles were most likely to strike.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Aircraft Icing</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Ice Shedding</td>
<td>5</td>
</tr>
<tr>
<td>1.3 Thesis Objective</td>
<td>9</td>
</tr>
<tr>
<td>1.4 Contributions</td>
<td>11</td>
</tr>
<tr>
<td>2. LITERATURE REVIEW</td>
<td>13</td>
</tr>
<tr>
<td>2.1 Ice Shedding Research</td>
<td>13</td>
</tr>
<tr>
<td>2.2 Trajectory Simulation of Random Shaped Bodies</td>
<td>16</td>
</tr>
<tr>
<td>2.3 Probabilistic Techniques to Model Uncertainties</td>
<td>18</td>
</tr>
<tr>
<td>3. ICE PARTICLE TRAJECTORY SIMULATION METHODOLOGY</td>
<td>21</td>
</tr>
<tr>
<td>3.1 Flowfields</td>
<td>22</td>
</tr>
<tr>
<td>3.1.1 Uniform Flowfield</td>
<td>22</td>
</tr>
<tr>
<td>3.1.2 Airfoil Flowfields</td>
<td>22</td>
</tr>
<tr>
<td>3.1.3 Wing Flowfields</td>
<td>25</td>
</tr>
<tr>
<td>3.1.4 CFD Flowfield Interpolation Methods</td>
<td>29</td>
</tr>
<tr>
<td>3.1.4.1 Triangular Element Interpolation</td>
<td>30</td>
</tr>
<tr>
<td>3.1.4.2 Quadrilateral Element Interpolation</td>
<td>33</td>
</tr>
<tr>
<td>3.1.4.3 Tetrahedral Element Interpolation</td>
<td>37</td>
</tr>
<tr>
<td>3.1.4.4 Pyramidal Element Interpolation</td>
<td>38</td>
</tr>
<tr>
<td>3.1.4.5 Wedge Element Interpolation</td>
<td>38</td>
</tr>
<tr>
<td>3.1.4.6 Hexahedral Element Interpolation</td>
<td>41</td>
</tr>
<tr>
<td>3.2 Trajectory Simulation Models</td>
<td>44</td>
</tr>
<tr>
<td>3.2.1 3-DOF Trajectory Simulation Model</td>
<td>44</td>
</tr>
<tr>
<td>3.2.2 4-DOF Trajectory Simulation Model</td>
<td>48</td>
</tr>
<tr>
<td>3.2.3 6-DOF Trajectory Simulation Model</td>
<td>48</td>
</tr>
<tr>
<td>3.3 Ice Particle Mass Properties</td>
<td>53</td>
</tr>
<tr>
<td>3.4 Ice Particle Aerodynamic Characteristics</td>
<td>57</td>
</tr>
<tr>
<td>3.4.1 Aerodynamic Data for 3-DOF and 4-DOF Trajectory Simulations</td>
<td>58</td>
</tr>
<tr>
<td>3.4.1.1 Square Plate</td>
<td>58</td>
</tr>
<tr>
<td>3.4.1.2 Rectangular Plate</td>
<td>61</td>
</tr>
<tr>
<td>3.4.1.3 Semi-circular Shell</td>
<td>63</td>
</tr>
<tr>
<td>3.4.2 Aerodynamic Data for 6-DOF Trajectory Simulations</td>
<td>64</td>
</tr>
<tr>
<td>3.4.2.1 Aerodynamic Characteristics of the Rectangular Plate</td>
<td>64</td>
</tr>
<tr>
<td>3.4.2.2 Interpolation of Experimental Data</td>
<td>65</td>
</tr>
<tr>
<td>4. MONTE CARLO METHOD</td>
<td>70</td>
</tr>
<tr>
<td>5. SIMULATION TOOLS</td>
<td>72</td>
</tr>
<tr>
<td>5.1 FORTRAN: WSU Ice Particle Trajectory Code</td>
<td>72</td>
</tr>
<tr>
<td>5.2 MSC.Easy5: Ice Particle Trajectory Simulation Models</td>
<td>72</td>
</tr>
</tbody>
</table>
LIST OF TABLES

<table>
<thead>
<tr>
<th>Tables</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1  Local parametric coordinates of all nodes of a quadrilateral element</td>
<td>33</td>
</tr>
<tr>
<td>Table 2  Mass and inertia matrix for square, rectangular and semi-circular ice shapes</td>
<td>56</td>
</tr>
<tr>
<td>Table 3  Parameters varied in Monte Carlo simulation of square plate trajectories</td>
<td>89</td>
</tr>
<tr>
<td>Table 4  Parameters varied in Monte Carlo simulation of rectangular plate trajectories</td>
<td>89</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figures</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Rime ice formation on a wing and typical rime ice cross section.</td>
</tr>
<tr>
<td>1.2</td>
<td>Glaze ice formation on a wing and typical cross section of glaze ice.</td>
</tr>
<tr>
<td>1.3</td>
<td>Mixed ice formation and typical mixed ice cross-section.</td>
</tr>
<tr>
<td>1.4</td>
<td>Pneumatic boots on the leading edge of a wing, and (inset) cross section of an inflated and deflated pneumatic boot</td>
</tr>
<tr>
<td>1.5</td>
<td>Working of an electro-expulsive system.</td>
</tr>
<tr>
<td>1.6</td>
<td>Ice shed from the operation of an electro-expulsive system in a wind tunnel.</td>
</tr>
<tr>
<td>1.7</td>
<td>Engine damage sustained from ice particle strike.</td>
</tr>
<tr>
<td>1.8</td>
<td>Shapes of simulated shed ice particles.</td>
</tr>
<tr>
<td>1.8a</td>
<td>Cuboidal block.</td>
</tr>
<tr>
<td>1.8b</td>
<td>Rectangular plates (high aspect ratio).</td>
</tr>
<tr>
<td>1.8c</td>
<td>Rectangular plates (low aspect ratio).</td>
</tr>
<tr>
<td>1.8d</td>
<td>Rectangular chip.</td>
</tr>
<tr>
<td>1.8e</td>
<td>Wedge.</td>
</tr>
<tr>
<td>1.8f</td>
<td>Triangular extrusion.</td>
</tr>
<tr>
<td>1.8g</td>
<td>Hemispherical shell.</td>
</tr>
<tr>
<td>1.8h</td>
<td>Semi-circular shell.</td>
</tr>
<tr>
<td>1.8i</td>
<td>Disk and fragmented disk representing radome ice.</td>
</tr>
<tr>
<td>1.8j</td>
<td>LEWICE glaze ice shape and fragmented LEWICE ice shapes.</td>
</tr>
<tr>
<td>1.9</td>
<td>Ice fragments tested in the wind tunnel.</td>
</tr>
<tr>
<td>1.9a</td>
<td>Rectangular plate.</td>
</tr>
<tr>
<td>1.9b</td>
<td>Semi-circular shell.</td>
</tr>
<tr>
<td>1.9c</td>
<td>Hemispherical shell.</td>
</tr>
</tbody>
</table>
1.10 Translational and angular movements permitted in different trajectory simulation models. ........................................................................................................12
1.10a 3-DOF model .........................................................................................................12
1.10b 4-DOF model .........................................................................................................12
1.10c 6-DOF model .........................................................................................................12
3.1 Overview of the ice particle trajectory simulation methodology...........................21
3.2 Computational grid. ...............................................................................................23
3.3 Close-up of grid with NACA 23012 airfoil. ..........................................................23
3.4 Close-up of iced airfoil grid...................................................................................24
3.5 Close-up of the 22.5-min glaze ice shape on NACA 23012 leading edge. ............24
3.6 Velocity flowfield and streamlines for a clean NACA 23012 airfoil; AOA of 0°. ...........................................................................................................................24
3.7 Velocity flowfield and streamlines for a clean NACA 23012 airfoil; AOA of 4°. ...........................................................................................................................24
3.8 Velocity flowfield and streamlines for a NACA 23012 airfoil with a 22.5 minute ice shape; AOA of 0°. ................................................................................25
3.9 Velocity flowfield and streamlines for a NACA 23012 airfoil with a 22.5 minute ice shape; AOA of 4°. ................................................................................25
3.10 Full view of the computational grid. ......................................................................26
3.11 Clean wing and streamlined body in test section...................................................26
3.12 Iced wing configuration; Looking from wing tip towards wing root. .................26
3.13 Close-up view of the computational grids. ............................................................27
3.13a Clean GLC-305 wing ..........................................................................................27
3.13b Iced GLC-305 wing ............................................................................................27
3.14 Velocity contours for the clean and iced wings at AOA of 0°...............................28
3.14a Clean wing ...........................................................................................................28
3.14b Iced wing ............................................................................................................28
3.15 Streamlines for the clean and iced wings at AOA of 0° ........................................ 28
3.15a Clean wing ............................................................................................................. 28
3.15b Iced wing ............................................................................................................... 28
3.16 Element shapes used to discretize 2D CFD domains ............................................. 29
3.17 Element shapes used to discretize 3D CFD domains ............................................. 29
3.18 Variation of parametric coordinate ‘p’ in a triangular element: p is zero on edge 12, one at node 1 and varies linearly in between .............................................. 31
3.19 Quadrilateral elements in Cartesian and parametric coordinate systems ............. 33
3.20 Variation of parametric coordinate ‘p’ normal to face 234 ................................... 37
3.21 Pyramidal element split into two tetrahedral elements .......................................... 38
3.22 Wedge in Cartesian and parametric coordinate systems ....................................... 39
3.22a Wedge in Cartesian space ..................................................................................... 39
3.22b Wedge in parametric space ................................................................................... 39
3.23 Hexahedron in Cartesian and parametric coordinate systems ............................. 42
3.23a Hexahedron in Cartesian space ........................................................................... 42
3.23b Hexahedron in parametric space .......................................................................... 42
3.24 Aerodynamic forces and moments acting on an ice particle after it sheds from an airplane surface ................................................................. 45
3.25 Forces and moments acting on an ice particle ...................................................... 49
3.26 A body rotating about an axis ............................................................................... 53
3.27 Dimensions of the square plate ice shape and axis system in which its inertia matrix was calculated ................................................................. 55
3.28 Dimensions of the rectangular plate ice shape and axis system in which its inertia matrix was calculated ................................................................. 55
3.29 Axis system in which the inertia matrix for the semi-circular shell ice shape was calculated ................................................................. 55
3.30 Dimensions of the semi-circular shell’s cross-section ........................................ 55
### Chapter 3

#### 3.31 Reference area and chord length of different ice particles. ........................................58

#### 3.32 Variation of lift and drag coefficients for a square plate. ........................................59

#### 3.33 Aerodynamic forces acting at the center of pressure of the square plate causing a moment at the center of gravity. .................................................................60

#### 3.34 Variation of pitching moment coefficient for a square plate. .................................60

#### 3.35 Variation of lift and drag coefficients for a rectangular plate. ...............................61

#### 3.36 Variation of pitching moment and side force coefficients for a rectangular plate. .................................................................62

#### 3.37 Comparison of lift and drag coefficients of the rectangular plate and the square plate. ..................................................................................................................62

#### 3.38 Comparison of pitching moment coefficient of the rectangular plate and the square plate. ..............................................................................................................63

#### 3.39 Variation of lift and drag coefficients for a semi-circular shell. .............................63

#### 3.40 Variation of pitching moment and side force coefficients for a semi-circular shell. ......................................................................................................................64

#### 3.41 Variation of force and moment coefficients with yaw and pitch angles for the rectangular plate at 0° roll angle. .................................................................66

  3.41a Normal force coefficient.........................................................................................66

  3.41b Axial force coefficient..........................................................................................66

  3.41c Side force coefficient..........................................................................................66

  3.41d Rolling moment coefficient ...............................................................................66

  3.41e Pitching moment coefficient .............................................................................66

  3.41f Yawing moment coefficient ..............................................................................66

#### 3.42 Variation of force and moment coefficients with yaw and pitch angles for the rectangular plate at 30° roll angle. .................................................................67

  3.42a Normal force coefficient.....................................................................................67

  3.42b Axial force coefficient.........................................................................................67

  3.42c Side force coefficient..........................................................................................67
3.42d Rolling moment coefficient .................................................................67
3.42e Pitching moment coefficient ...............................................................67
3.42f Yawing moment coefficient ...............................................................67
3.43 Angle of attack ($\alpha$) and angle of sideslip ($\beta$) of the relative velocity vector
(V) ..................................................................................................................68
3.44 Angle of attack ($\alpha$) and angle of sideslip ($\beta$) combinations for which
aerodynamic data are known. .........................................................................68
3.45 Triangular interpolation method used to obtain experimental aerodynamic
data ................................................................................................................69
4.1 Monte Carlo Simulations ........................................................................71
4.2 Footprint map of ice particle trajectories ...............................................71
5.1 Schematic of the WSU ice particle trajectory code ....................................73
5.2 Schematic of the 3-DOF ice trajectory simulation model in MSC.Easy5 ......75
5.3 Expansion of the Force and Moment Calculator sub-model ...................77
5.4 Schematic of the 6-DOF ice trajectory simulation model in MSC.Easy5 ......78
5.5 Matrix Algebra Tool window .................................................................79
6.1 Dimensions for 3-DOF ice particle trajectory simulations ......................81
6.2 Ice piece shed near the airfoil leading edge in 3-DOF simulations ..............81
6.3 3-DOF trajectories of different ice shapes in 2D uniform flowfields .........82
6.3a Without aerodynamic damping .............................................................82
6.3b With aerodynamic damping $C_{\text{damp}} = 0.0166 \text{ lb-ft-sec}$ .................82
6.4 3-DOF trajectories of square plate in uniform flowfields for cases with
aerodynamic damping ($C_{\text{damp}} = 0.0166 \text{ lb-ft-sec}$) and without aerodynamic
damping .........................................................................................................83
6.5 Comparison of parameters that affect square plate trajectory calculations for
simulations with aerodynamic damping ($C_{\text{damp}} = 0.0166 \text{ lb-ft-sec}$) and
without aerodynamic damping ($C_{\text{damp}} = 0 \text{ lb-ft-sec}$) .......................84
6.6 3-DOF trajectories of different ice shapes in 2D, $0^\circ$ AOA NACA 23012
airfoil flowfield and 2D uniform flowfield ...................................................86
6.6a Without aerodynamic damping.................................................................86
6.6b With aerodynamic damping $C_{damp} = 0.0166 \text{ lb-ft-sec}$........................86
6.7 3-DOF trajectories of square plate in 2D clean and iced airfoil flowfields........87
6.7a Without aerodynamic damping.................................................................87
6.7b With aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec}$.........................87
6.8 Probability map of 3-DOF ice particle trajectories shed in 2D uniform
   flowfield........................................................................................................90
6.8a Without aerodynamic damping.................................................................90
6.8b With aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec}$.........................90
6.9 Probability distribution of 3-DOF ice particle trajectories shed in 2D, $0^\circ$
   AOA clean airfoil flowfield........................................................................91
6.9a Without aerodynamic damping.................................................................91
6.9b With aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec}$.........................91
6.10 Probability map of 3-DOF square plate trajectory footprints in 2D airfoil
   flowfields; with aerodynamic damping, $C_{damp}=0.0166 \text{ lb-ft-sec}$................94
6.10a Clean airfoil flowfield................................................................................94
6.10b Iced airfoil flowfield..................................................................................94
6.11 Probability map of 3-DOF square plate trajectory footprints in 2D airfoil
   flowfields; without aerodynamic damping....................................................94
6.11a Clean airfoil flowfield................................................................................94
6.11b Iced airfoil flowfield..................................................................................94
6.12 Ice particle shed off the side of an aircraft fuselage......................................95
6.13 4-DOF trajectories of ice particles shed in 3D uniform flowfield without
   aerodynamic damping, $C_{damp} = 0.0 \text{ lb-ft-sec}$........................................97
6.13a $y$ vs. $x$.................................................................................................97
6.13b $z$ vs. $x$.................................................................................................97
6.14 4-DOF trajectories of ice particles shed in 3D uniform flowfield with
   aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec}$.................................97
6.14a y vs. x .................................................................................................................. 97
6.14b z vs. x .................................................................................................................. 97

6.15 Comparison of 4-DOF rectangular plate trajectories computed with experimental and empirical aerodynamic data; without aerodynamic damping .......................................................................................................................... 98
6.15a y vs. x .................................................................................................................. 98
6.15b z vs. x .................................................................................................................. 98

6.16 Comparison of 4-DOF rectangular plate trajectories computed with experimental and empirical aerodynamic data; with aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec.}$ ......................................................................................... 99
6.16a y vs. x .................................................................................................................. 99
6.16b z vs. x .................................................................................................................. 99

6.17 Notation of gravity, lift and drag forces acting on the ice piece ................... 100

6.18 Top view of initial position of the ice piece with respect to the wing mid-semispan location ................................................................................................................................. 100

6.19 4-DOF trajectories of ice shapes shed in clean GLC-305 swept wing flowfield without aerodynamic damping ............................................................. 101
6.19a y vs. x .................................................................................................................. 101
6.19b z vs. x .................................................................................................................. 101

6.20 4-DOF trajectories of ice shapes shed in clean GLC-305 swept wing flowfield with aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec.}$ ............................................................. 101
6.20a y vs. x .................................................................................................................. 101
6.20b z vs. x .................................................................................................................. 101

6.21 Comparison of parameters that affect square plate trajectory calculations in 3D clean and iced wing flowfields; without aerodynamic damping ................................................................. 104

6.22 Comparison of parameters that affect square plate trajectory calculations in 3D clean and iced wing flowfields; with aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec.}$ ................................................................. 105

6.23 4-DOF square plate trajectory footprint map at the engine inlet location. ........ 106
6.23a Without aerodynamic damping ........................................................................ 106
6.23b With aerodynamic damping, $C_{damp} = 0.0166$ ft-lb-sec

6.24 Probability distributions of the 4-DOF square plate trajectory footprints at the engine inlet location.

6.24a Without aerodynamic damping

6.24b With aerodynamic damping, $C_{damp} = 0.0166$ ft-lb-sec

6.25 4-DOF rectangular plate trajectory footprint map at the engine inlet location.

6.25a Without aerodynamic damping

6.25b With aerodynamic damping, $C_{damp} = 0.0166$ ft-lb-sec

6.26 Probability distributions of 4-DOF rectangular plate trajectory footprints at the engine inlet plane for the case without aerodynamic damping.

6.27 Probability distributions of 4-DOF rectangular plate trajectory footprints at the engine inlet plane for the case with aerodynamic damping, $C_{damp} = 0.0166$ ft-lb-sec

6.28 Variation of y-displacement, z-displacement, yaw angle ($\psi$), pitch angle ($\theta$) and roll angle ($\phi$) with x-displacement of 6-DOF rectangular plate trajectory in 3D uniform flowfield.

6.29 Variation of y-displacement, z-displacement, yaw angle ($\psi$), pitch angle ($\theta$) and roll angle ($\phi$) with x-displacement of 6-DOF rectangular plate trajectory in 3D, clean and iced, $0^\circ$ AOA GLC-305 wing flowfield.
# LIST OF SYMBOLS

## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D</td>
<td>Two-dimensional</td>
</tr>
<tr>
<td>3D</td>
<td>Three-dimensional</td>
</tr>
<tr>
<td>AOA</td>
<td>Angle of attack</td>
</tr>
<tr>
<td>CFD</td>
<td>Computational Fluid Dynamics</td>
</tr>
<tr>
<td>DOF</td>
<td>Degrees of Freedom</td>
</tr>
<tr>
<td>GBU</td>
<td>Guided Bomb Unit</td>
</tr>
<tr>
<td>JDAM</td>
<td>Joint Direct Attack Munition</td>
</tr>
<tr>
<td>LWC</td>
<td>Liquid Water Content</td>
</tr>
<tr>
<td>MAC</td>
<td>Mean Aerodynamic Chord</td>
</tr>
<tr>
<td>MSL</td>
<td>Mean Sea Level</td>
</tr>
<tr>
<td>MVD</td>
<td>Mean Volumetric Diameter</td>
</tr>
<tr>
<td>MPPL</td>
<td>Most Probable Point Locus</td>
</tr>
<tr>
<td>NACA</td>
<td>National Advisory Committee for Aeronautics</td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautics and Space Administration</td>
</tr>
<tr>
<td>POST</td>
<td>Program to Optimize Simulated Trajectories</td>
</tr>
<tr>
<td>RSM</td>
<td>Response surface method</td>
</tr>
<tr>
<td>SLAM-ER</td>
<td>Standoff Land Attack Missile – Extended Range</td>
</tr>
<tr>
<td>WSU</td>
<td>Wichita State University</td>
</tr>
</tbody>
</table>

## Units

<table>
<thead>
<tr>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>°C</td>
<td>Degree Celsius, unit of temperature</td>
</tr>
</tbody>
</table>
ft  Feet, unit of length
ft/s  Feet per second, unit of speed
knots  Knots, unit of speed
lb  Pounds, unit of force
lbm  Pound-mass, unit of mass
mph  Miles per hour, unit of speed
m/s  Meters per second, unit of speed
slugs  Slugs, unit of mass
sec  Seconds, unit of time

Mathematical symbols

c  Chord length of the ice particle
$C_D$  Coefficient of drag
$C_{D0}$  Coefficient of drag at AOA of $0^\circ$
$C_{damp}$  Aerodynamic rotational damping coefficient
$C_F$  Coefficient of force
$C_L$  Coefficient of lift
$C_M$  Coefficient of moment
$C_N$  Coefficient of normal force
$C_{SF}$  Coefficient of side force
d  Diameter of the semi-circular shell ice shape
D  Drag on the ice particle
F  Force on the ice particle
g Acceleration due to gravity
H Moment of momentum of the ice particle
I Moment of inertia of the ice particle about a rotation axis
[I] Inertia matrix of the ice particle about its center of mass
J Jacobian matrix of transformation
L Lift on the ice particle
m_{ice} Mass of the ice particle
M Moment on the ice particle
N Normal force on the ice particle
N_i Shape functions
p,q,r,s Parametric coordinate space
P,Q,R Angular velocities about body axes
\dot{P}, \dot{Q}, \dot{R} Angular accelerations about body axes
q_1,q_2,q_3,q_4 Quaternion
S Reference area of the ice particle
SF Side force on the ice particle
t Time
t_s Time step
U_i Flow properties at nodes of a CFD element
v_{ice} Volume of the ice particle
V Relative velocity of the ice particle with respect to the flow
V_{ice} Velocity of the ice particle in the inertial axis system
W Flow velocity
x,y,z  Axes in Cartesian coordinate space

\ddot{x}, \ddot{y}, \ddot{z}  Acceleration of the ice particle along the x-, y- and z- axes

\ddot{x}_{adn}, \ddot{y}_{adn}  Acceleration of the ice particle due to aerodynamic forces

\alpha  Angle of attack

\alpha_j  Coefficients in functions used to interpolate velocities in CFD grids

\beta  Angle of sideslip

\delta  Orientation angle of the ice particle with respect to the x-axis

\gamma  Inclination of the relative velocity vector with respect to the x-axis

\lambda  Integration drift correction gain

\rho  Density

\omega  Angular velocity

\psi, \theta, \Phi  Yaw, Pitch and Roll Euler angle

\xi, \eta, \zeta  Axes of the parametric coordinate system
CHAPTER 1

INTRODUCTION

Ice is known to accumulate on aircraft surfaces in certain cold weather conditions while in flight or on the ground. Ice accretions, especially on aerodynamic surfaces, have an adverse affect on the aircraft’s performance and handling qualities. Over 15% of weather related accidents have been caused by ice accretions on airframes (Ref. 1). Ice accretions and their effects on airplane performance has been the topic of extensive research. Computer codes capable of simulating the shapes of ice accretions have been developed which have helped aircraft and ice protection system manufacturers protect airplanes from the harmful effects of icing.

The shedding of accreted ice also poses a danger to the airplane. Aircraft components can be damaged if they are struck by large ice particles. Airplane manufacturers usually conduct flight tests to make sure that shed ice will not damage critical aircraft components. Research in the field of ice shedding is limited and efforts to model the events that take place after ice sheds are relatively recent. As a result, simulation tools that have been developed to assess the ice shedding risk to an airplane are not advanced enough to be used by aircraft makers in place of flight tests. In this thesis, an effort has been made to improve the technique used to simulate trajectories of shed ice particles.

1.1 Aircraft Icing

Aircraft icing occurs when water droplets in the atmosphere freeze after coming into contact with aircraft surfaces whose temperatures are below 0°C. The probability of
Ice accretion is high when temperatures are below freezing and liquid moisture is present in the atmosphere. Aircrafts can accumulate ice in weather conditions like freezing rain or snow, or while flying through moisture containing clouds in subfreezing temperatures.

**Fig. 1.1** Rime ice formation on a wing and typical rime ice cross section (Ref. 2).

Ice accretions are broadly classified as rime, glaze or mixed according to their formation and appearance. Rime ice has a milky, granular appearance and a rough surface as shown in Fig. 1.1, and is formed when water droplets freeze instantly upon

**Fig. 1.2** Glaze ice formation on a wing and typical cross section of glaze ice (Refs. 1, 2).
impact with the aircraft surface. Rime ice is formed at ambient temperatures of -20°C to 0°C. Glaze ice or clear ice is formed at temperatures between -10°C to 0°C when water droplets flow over the aircraft surface after impact and then freeze to form a glossy layer of solid ice (Fig. 1.2). Mixed ice is a combination of rime and glaze ice (Fig. 1.3).

![Mixed ice formation and typical mixed ice cross-section (Ref. 2).](image)

**Fig. 1.3** Mixed ice formation and typical mixed ice cross-section (Ref. 2).

Research on the physics of ice accretion has resulted in the development of computer codes like LEWICE which are capable of predicting ice accretions (Ref. 3). Several wind tunnel experiments have been conducted to determine the effects of rime, glaze and mixed ice accretions on wings and tails. In general, ice accretions on wings increases drag, reduces lift and lowers the stall angle (Refs. 4-8). Wind tunnel tests in Refs. 4 and 5 show that glaze ice has the most detrimental aerodynamic effect. Reference 7 shows that glaze ice can increase drag of an airfoil by 313% and reduce its lift by 43%. Ice can affect roll control if it accretes ahead of the ailerons. Ice on a horizontal tail in Ref. 8 reduced the lift coefficient by 45% and the stall angle by 6° when compared to a non-iced tail. The reduction in tail lift coefficient and stall angle has an adverse effect on the pitch handling qualities. In addition to the loss of performance, ice can cause
operational problems such as loss of visibility due to ice on windscreens, attenuation of radio and radar signals from ice on radomes and antennas, and erroneous readings on pressure instruments such as altimeter and airspeed indicator due to ice accretion on the pitot-static ports.

To minimize the adverse effects of icing, most aircraft nowadays are equipped with ice protection systems at critical locations such as wings, horizontal tails and propellers. Ice protection systems are categorized into two main groups: anti-icing systems which prevent ice from accumulating on aircraft surfaces and de-icing systems which remove accreted ice. Pneumatic boots (Fig. 1.4) and electro-expulsive systems (Fig. 1.5) remove ice mechanically by inflating and deflating tubes in a boot and by delivering a hammer like blow from an electric discharge respectively. Thermal ice protection systems use electricity or hot air to heat critical aircraft surfaces.

Fig. 1.4  Pneumatic boots on the leading edge of a wing, and (inset) cross section of an inflated and deflated pneumatic boot (Ref. 9).

The functioning of ice protection systems can cause smaller ice accretions to form. De-icing systems are periodically activated resulting in ‘inter-cycle ice’ formations
in between periodic activations and ‘residual ice’ formations immediately after deicer activations. Functioning of thermal anti-icing systems can cause ice accretions downstream of the ice protection system known as ‘runback ice’ when water flows downstream over heated airplane surfaces and freezes on surfaces not protected by anti-icing systems. From wind tunnel experiments presented in Refs. 7 and 10, inter-cycle and runback ice was seen to reduce lift and increase drag.

**Fig. 1.5 Working of an electro-expulsive system.**

Large amounts of information are available on the formation of ice, its accretion and its aerodynamic effects. However, research on the hazards of shedding of accreted ice has been limited. An aviation mishap caused due to ice shedding occurred on December 27th, 1991, when a McDonnell Douglas MD-81 transport jet crashed in Sweden a few minutes after takeoff (Ref. 11). Clear ice that formed on the wings, because of cold fuel, came off during takeoff and was ingested by the engine resulting in engine blade damage and ultimately engine failure.

### 1.2 Ice Shedding

Any ice that accumulates on protected and unprotected aircraft surfaces eventually sheds. Ice shedding can take place due to the functioning of ice protection
systems (Fig. 1.6) or from the action of aerodynamic forces and warm temperatures on ice accretions. Shed ice particles mostly originate from the following ice accretions:

- Ice on surfaces not protected by ice protection systems.
- Ice formed due to the failure of the ice protection system.
- Ice accumulation because of delay in activating the ice protection system.
- Residual and inter-cycle ice accretions. (Inter-cycle ice accretions shed when pneumatic boots and electro-expulsive systems operate)
- Runback ice from the functioning of thermal anti-icing systems.
- Ice accretions occurring beyond the limits of the ice protection systems.
- Accumulations on ground due to adverse weather conditions.
- Ice formed on wings due to cold fuel (cold soaking effect).

All the above ice accretions do not have major aerodynamic degradation effects but can be a danger to the aircraft if they shed. The main hazard from ice shedding is to the engine. Shed ice particle can enter the engine causing damage to the engine blades affecting its operability, or in extreme circumstances result in engine failure. Small
turbine engines powering executive jets and regional jets are more sensitive to compressor blade damage and adverse engine operation during ice ingestion than large turbine engines used in large aircrafts. Figure 1.7 shows the picture of a set of fan blades damaged due to ice ingestion. Shed ice could also strike and damage other parts of the aircraft or block the movement of control surfaces.

Maximum ice shedding occurs after an icing encounter when the aircraft is flown into regions where the ambient temperature is above freezing. The danger to the airplane is mainly from large shed ice particles. Large fragments are formed in holding conditions, when the aircraft is forced to spend a considerably large amount of time at low altitudes in icing conditions. When the aircraft transitions to descent and landing phase the accreted ice begins to melt and shed.

Ice can shed from the leading edge of wings and empennages, windshields, airplane nose, antennae, propellers and rotors. The sizes and shapes of shed ice particles are random. A few generic shapes of shed ice are shown in Fig. 1.8. The cuboidal block shown in Fig. 1.8a is symbolic of a large ice accretion capable of damaging the engine of an executive jet. Figures 1.8b, 1.8c and 1.8d depict shapes representative of ice shed from wing surfaces. Shapes depicted in Figs. 1.8e and 1.8f describe runback ice shapes shed off a wing. The hemispherical shell and disks in Figs. 1.8g and 1.8i represent ice shapes shed from the radome or nose of the airplane while the semi-circular and LEWICE ice shapes shown in Figs. 1.8h and 1.8j respectively, are illustrative of ice shed off the leading edge of the wing. The LEWICE ice shapes are obtained by running the ice accretion code LEWICE with different icing conditions.
1.8a Cuboidal block

1.8b Rectangular plates (high aspect ratio)

1.8c Rectangular plates (low aspect ratio)

1.8d Rectangular chip

1.8e Wedge

1.8f Triangular extrusion

1.8g Hemispherical shell

1.8h Semi-circular shell

1.8i Disk and fragmented disk representing radome ice

1.8j LEWICE glaze ice shape and fragmented LEWICE ice shapes.

Fig. 1.8 Shapes of simulated shed ice particles.
Ice shedding is a random and complex event. The randomness of the shapes and sizes of shed ice particles, the tumbling of ice particles and the aerodynamic forces acting on the ice particles makes the event difficult to simulate. There has been no analytical model developed to predict the trajectory of shed ice particles. Computational simulation of ice particle trajectories conducted by aircraft manufacturers assume that the particle trajectories are driven by aerodynamic drag forces (Ref. 13). For large ice fragments, lift and aerodynamic moments influence the trajectory along with the drag. The lack of an experimental database containing the aerodynamic forces and moments experienced by an ice particle at different orientations to the flow is an added disadvantage. Consequently, aircraft manufacturers conduct flight tests to make sure that any shed ice will not damage critical aircraft components. A PC-based trajectory simulation tool can be a cost effective alternative to expensive flight tests conducted during aircraft certification. It could also be useful in the aircraft design phase to ensure that critical components are not placed in areas which have a high probability of being hit by shed ice.

1.3 Thesis Objective

The main goal of this thesis was to develop a simulation methodology to predict the trajectory of shed ice particles and to determine the areas that have a high chance of being struck by shed ice. The specific objectives of the research effort carried out to accomplish this goal were:

- Develop an analytical model to simulate the trajectory of shed ice particles in two-dimensional and three-dimensional flowfields.
• Formulate a probabilistic approach to model the random nature of ice shedding and obtain a probability map identifying the ‘hot spots’ or areas where ice particles are most likely to strike.

• Code the trajectory model and probabilistic method in FORTRAN and MSC.Easy5 simulation tools so that aircraft manufacturers can utilize it.

The research effort described in this thesis was the computational part of a project aimed at developing a method to simulate ice particle trajectories. The experimental portion of the project consisted of wind tunnel testing to determine the aerodynamic forces and moments that act on ice particles at different orientations to the flowfield. Wind tunnel tests on a flat plate, semi-circular shell and hemispherical shell, depicted in Fig. 1.9, were performed by Jacob (Ref. 14). The aerodynamic database developed from the wind tunnel tests were used in this work to compute the trajectories of ice fragments.

Fig. 1.9 Ice fragments tested in the wind tunnel.
1.4 Contributions

In this thesis, three, four and six degree of freedom (DOF) trajectory models were developed. The 3-DOF simulation model allows the particle to move along two mutually perpendicular axes and rotate about the third perpendicular axis (Fig. 1.10a) while the 4-DOF model allows translational movement about three mutually perpendicular axes and rotation about one of the axis, as depicted in Fig. 1.10b. The 6-DOF model in Fig. 1.10c allows ice particle translation and rotation about all three mutually perpendicular axes. The flowfield into which the ice particle sheds was determined using the CFD software FLUENT. The random nature of ice shedding was modeled using the Monte Carlo method. The author’s contributions in different areas of this thesis are summarized below:

- **Literature review**: The author conducted a review of previous research efforts related to the trajectory analysis of shed ice particles and random-shaped bodies, and the use of probabilistic methods in different engineering problems.

- **Flowfield computations using CFD**: The author used CFD to compute the two-dimensional (2D) flowfields presented in this thesis. Three-dimensional (3D) flowfields presented in this thesis were provided by the WSU icing laboratory.

- **CFD flowfield interpolation**: CFD data contains flow information (e.g. flow velocity) at discrete locations in the flowfield domain and interpolation is required to determine flow information at any desired location in the domain. The author wrote computer codes to interpolate 3D CFD flowfields. The codes for interpolating 2D CFD flowfields was provided by the WSU icing laboratory.
Coding of trajectory simulation model in FORTRAN: A FORTRAN code to simulate 3-DOF and 4-DOF trajectories was available at WSU and the author continued the development of this code. The author developed the 6-DOF trajectory simulation FORTRAN code.

Developing trajectory simulation models in MSC.Easy5: The trajectory simulation models were also coded in MSC.Easy5, which is a simulation tool used in the aircraft industry. The author developed the MSC.Easy5 trajectory simulation models presented in this thesis.

Monte Carlo simulations: The author programmed the Monte Carlo method in FORTRAN and MSC.Easy5 to obtain a probability map showing the areas that are likely to be hit by shed ice particles.

Wind tunnel testing: The author was also extensively involved in experiments conducted at the WSU 7-ft by 10-ft wind tunnel to determine the aerodynamic forces and moments that act on the ice fragments shown in Fig. 1.9.
CHAPTER 2

LITERATURE REVIEW

A literature survey was carried out to review previous research efforts on the trajectory simulation of shed ice particles and other random shaped bodies. Probabilistic techniques used in different engineering problems were also evaluated so that an appropriate technique could be used to model randomness in the ice shedding event.

2.1 Ice Shedding Research

Research efforts in the field of ice shedding are recent. Few research papers are available on the simulation of ice particle trajectories. Brief summaries of the works reviewed are provided in the following.

Chandrasekharan and Hinson (Ref. 13) used a modified water droplet trajectory code called ICE to track the trajectories of ice particles around a fuselage. The original ICE code computes the trajectories of icing droplets and analyzes their impingement on airplane surfaces. The modified code allowed the user to specify the size, weight and drag characteristics of solid ice particles. The flowfield around the aircraft was computed using VSAERO. The trajectories of three different ice shapes were computed. A disk of 4" diameter and 1" thickness was assumed to be the shape of ice shed from a radome. Two slabs of size 3"×1"×1" and 1.5"×1.5"×0.3" were used to simulate shedding from the aircraft windshield. The trajectories of the ice pieces were assumed to be dependant only on drag. It was assumed that a tumbling ice shape would not produce significant lift and side force. The drag of the disk was approximated to that of a sphere and the drag of the slab was approximated to the drag of a flat plate normal to the flow. The simulation was
used to assess if shedding of these ice shapes would pose a threat to the aft mounted engines of a Learjet 40 business jet. The Learjet 45 business jet had a 2 ft longer fuselage than Learjet 40, and had undergone extensive flight tests with no incidence of engine damage from ice particles. The numerical simulations showed that Learjet 40 had ice-shedding characteristics similar to Learjet 45, eliminating the need for expensive flight tests.

Kohlman and Winn (Ref. 15) developed a 4-DOF trajectory simulation model to compute the trajectories of ice particles shed from an airframe. The ice particles were assumed to be square plates of uniform thickness shed from the fuselage into a uniform flowfield. Lift and drag were assumed to be the main aerodynamic forces that acted on them. Empirical equations that modeled available experimental lift and drag coefficients of a flat plate were used to compute the lift and drag forces. The rotation of the ice piece was limited to a single axis and opposition to the rotation was represented by a rotational damping coefficient. The rotation or oscillation of the shed ice fragment during the trajectory was found to depend on the initial angle of attack and the rotational damping coefficient. The thickness of the ice piece affected its weight and therefore its vertical displacement while the area of the square ice shape affected the lateral displacement (displacement perpendicular to the fuselage in the direction of the wing). By varying the size, thickness, initial angle of attack and rotational damping coefficient, the locus of trajectories was obtained.

Another method, similar to Kohlman and Winn’s model, was formulated by Santos et al. (Ref. 16). It was a 3-DOF simulation model capable of computing ice particle trajectories in a 2D domain. The shed ice fragments were assumed to be squares
with constant thickness shed into a non-uniform flowfield from a point slightly above the leading edge of the wing close to the fuselage. The trajectory was calculated till the ice particle reached a station located approximately two chords downstream of the leading edge of the wing. The non-uniform flowfield around the wing was computed using the FLUENT code. Simulations were performed for a range of initial angles, initial angular velocities and rotational damping. A distribution showing the probability of ice impact was obtained at the station where trajectory calculations were stopped.

Wright et al. (Ref. 17) described a method to determine the trajectories of ice particles by calculating the drag force on the particles and then finding the velocities of the particles from the net force. Integrating the velocities over time would result in the trajectory of the ice particles. Experimental drag coefficients of different ice shapes were used to calculate the drag force on the ice particles.

The techniques presented in Refs. 13, 15, 16 and 17 to simulate the trajectories of shed ice have not been experimentally validated. References 15 and 16 provide simple models to simulate trajectories of square ice particles with the ice particle rotation assumed to be about only one axis. Shed ice particles, however, could experience more complex rotations about all three axes. In some of the previous efforts reviewed, it was assumed that only lift force, drag force and pitching moment act on the ice particle. The side force, rolling moment and yawing moment of the ice particle were not considered during trajectory computations. The forces applied to the shed ice particle were based on empirical data adapted from experimental studies conducted with bluff bodies.
2.2 Trajectory Simulation of Random Shaped Bodies

Few analytical models have been developed to simulate the trajectories of shed ice particles, so research efforts in modeling the trajectories of random shaped bodies were investigated.

The shedding of ice from airplanes is similar to the shedding of heat shield particles from spacecrafts in a way that the sizes of shed particles are random. References 18 and 19 present methods for evaluating the trajectories of heat shield particles shed from spacecrafts. Davies and Park (Ref. 18) formulated differential equations that describe the behavior of the shed particles and developed numerical algorithms for solving the equations. The analytical method took into account evaporation of the heat shield particle and conservation of mass, momentum and energy during trajectory computations. Trajectories were simulated for a range of particle diameters, initial velocities, temperatures and initial locations on the Galileo probe surface. Naughton et al. (Ref. 19) simulated trajectories of heat shield particles shed from an X-38 crew return vehicle and evaluated the risk to the vehicle. The simulation model did not consider the evaporation of heat shield particles while computing their trajectories. Simulations were conducted for a combination of particle diameters, densities and initial release velocities. The risk to the vehicle was assessed considering the momentum of the particle during impact.

In Refs. 18 and 19, the trajectory models were based on heat shield particles being spheres with diameters of the order of millimeters and the controlling force for the trajectories being the aerodynamic drag. The trajectories of the particles were found to
depend on the initial conditions. The flowfield around the spacecrafts was evaluated using CFD codes.

Availability of aerodynamic characteristics (forces and moments) for random shapes is very limited and thus, most simulation methods model only the drag force acting on a particle. If the aerodynamic characteristics of the object are known, a 6-DOF trajectory simulation model (Ref. 20) containing non-linear rigid body equations of motion can be used to compute the trajectory of the body. The aerodynamic data of random shapes can be obtained either from wind tunnel test or using CFD codes. Many research efforts to model trajectories of missiles have used CFD codes to compute not only the flowfield around missiles but also the aerodynamic forces and moments acting on the missiles.

Murman et al. (Ref. 21) used a coupled CFD/6-DOF simulation model to predict the trajectories of a GBU-31 Joint Direct Attack Munition (JDAM) store released from an F/A-18C fighter aircraft. The aerodynamic forces and moments that acted on the store were determined using a CFD code and the 6-DOF trajectory simulation model calculated the store’s trajectory. The location and orientation of the store at a particular time step, say \( t_s = t \), was acquired from the 6-DOF trajectory model and entered into the CFD code. The aerodynamic force and moments computed with the CFD code were then entered into the 6-DOF trajectory simulation model which calculated the store’s location and orientation at the next time step, \( t_s = t + \Delta t \). This process was repeated and the path of the store was computed. The trajectory was compared to flight test data obtained from telemetry and photography. For the initial portions of the trajectory, the predicted
trajectory and the flight data showed good agreement but for the later portions of the trajectory differences were observed.

Ray (Ref. 22) used CFD to form a database of aerodynamic forces and moments that act on an SLAM-ER (Standoff Land Attack Missile – Extended Range) missile at different orientations. This database was used as an input to a 6-DOF trajectory code to compute the trajectory of the missile released from an S-3B Viking aircraft. The aerodynamic database computed using CFD generally tended to agree with wind tunnel aerodynamic data. The simulated trajectories were compared to those obtained from flight tests and the predicted trajectories correlated well with the flight tests.

2.3 Probabilistic Techniques to Model Uncertainties

Probabilistic methods consider statistical distributions in the input variables and provide probability distributions for the output variables. There are uncertainties associated with ice shedding especially with the shapes and sizes of shed ice particles. Different probabilistic techniques used to solve engineering problems were investigated so that an appropriate technique could be selected in modeling the effects of uncertainties on the ice particle trajectory.

Monte Carlo method was used by Meiboom (Ref. 23) and Desai et al. (Ref. 24) to model uncertainties when simulating 6-DOF trajectories of the Ariane-5 booster recovery system and the Stardust Return Capsule respectively. In Ref. 23, the aerodynamic characteristics of the rocket along with information of the rocket ascent trajectory, atmospheric properties and booster mass and inertia were used as input for the 6-DOF trajectory simulation model. Uncertainties in the aerodynamic properties and deviations in the atmospheric and booster properties were incorporated in the model using Monte
Carlo methods. Before a trajectory was simulated, a set of random numbers were generated which determined deviations for the input parameters. A total of 150 trajectory computations were performed and probability distributions for parameters such as booster relative velocity, roll angle etc. were obtained.

In Ref. 24, trajectory analysis was performed using 3-DOF and 6-DOF versions of a code called POST (Program to Optimize Simulated Trajectories). The path traveled by the Stardust capsule from its separation with the spacecraft in space to its landing on earth was computed. Monte Carlo method was used to model off-nominal conditions that may arise due to uncertainties in capsule mass properties, separation attitude and atmospheric properties. The objective was to see if the capsule would stay within the design limits in off-nominal conditions. A spread of probable landing locations was also determined. Forty one different uncertainties were identified and 3000 trajectory simulations were carried out.

Kim et al. in Ref. 25 used the Response Surface Method to design the nose fairing of a space launcher optimized for minimum drag. The shape of the fairing was a linear combination of four shape functions and the coefficients of the shape functions consisted of the design variables (variables to be optimized). Simple body shapes with minimum drag like the von Karman ogive, power law body with n=0.69 etc. were selected as the shape functions. Since the flowfield around the nose was computed using a Navier Stokes code, the response surface method was used to reduce the number of optimization runs. The response surface method involves determining a polynomial approximation to the response, in this case drag, as a function of design variables from data determined through numerical analysis.
Reference 26 provides several example applications of different probabilistic methods in mechanical and aerospace engineering along with the advantages and disadvantages of the method. The Monte Carlo method is most widely used to study the effect of uncertainties but it is time-consuming as it requires executing a large number of simulations. Probabilistic techniques such as Most Probable Point Locus (MPPL) and Response Surface Method (RSM) require less number of simulations and are used in cases where simulation codes take a long time to run.
CHAPTER 3

ICE PARTICLE TRAJECTORY SIMULATION METHODOLOGY

Figure 3.1 shows the overview of the ice particle trajectory simulation model. The flowfield into which the ice particle sheds, the ice particle geometry and the ice particle aerodynamic characteristics form inputs to the trajectory simulation model. The equations of motion contained in the trajectory simulation model determine the translational and angular accelerations of the ice particle from the aerodynamic and gravitational forces that act on it. The ice particle’s trajectory is obtained by integrating the translational and angular accelerations. Three, four and six degree of freedom trajectory simulation models capable of computing the trajectories of shed ice particles in 2D and 3D flowfields were developed.

Fig. 3.1 Overview of the ice particle trajectory simulation methodology.
3.1 Flowfields

The trajectory of an ice particle is influenced by the flowfield into which it sheds as the aerodynamic forces that act on it depend on the relative velocity between the particle and the airflow. The flowfield about airfoils and wings were computed and used to simulate the trajectories of ice particles shed from wings. Trajectory computations were also carried out assuming that the particles were shed into a uniform flowfield.

3.1.1 Uniform Flowfield

Uniform flowfields have a constant velocity at every location in the flowfield. The uniform flowfield considered for the trajectory simulations had a steady velocity of 220 knots (approximately 113 m/s or 371 ft/s) to represent the flow near an aircraft in holding condition.

3.1.2 Airfoil Flowfields

Two-dimensional flowfields about clean and iced airfoils were determined using the CFD software - FLUENT. The clean airfoil was a NACA 23012 airfoil with a chord of 7.3 ft. The iced airfoil was a 7.3-ft chord NACA 23012 airfoil with a 22.5-minute glaze ice shape on its leading edge. The glaze ice shape was representative of an ice protection system failure case and was computed with the NASA Glenn LEWICE 2.2 ice accretion code (Ref. 27) under the following icing conditions:

- Angle of attack (AOA) = 2.5°
- MVD = 20 µm
- LWC = 0.5 g/m³
- Freestream Speed = 175 mph
- Static pressure = 13.75 psi
- Total temperature = 22.78°F
- Time steps = 1.25 minutes
- Accretion time = 22.5 minutes

Note that the ice shape was initially computed for a 3-ft chord airfoil and was scaled to a chord of 7.3 ft. The full computational grid used to calculate the flowfields is shown in Fig. 3.2, the grid close-up for the clean airfoil is shown in Fig. 3.3 and the close-up of the iced airfoil grid is illustrated in Fig. 3.4. The grid topology around the glaze ice shape at the leading edge of the NACA 23012 airfoil is depicted in Fig. 3.5.
Fig. 3.4 Close-up of iced airfoil grid.  
Fig. 3.5 Close-up of the 22.5-min glaze ice shape on NACA 23012 leading edge.

Flowfields were computed for holding conditions: freestream velocity of 220 knots, pressure altitude of 15000 ft and temperature of -9°C. Figures 3.6 and 3.7 show the velocity fields about a clean airfoil for angles of attack (AOA) of 0° and 4° respectively. Figures 3.8 and 3.9 show the flowfield about iced airfoils at 0° and 4° AOA respectively.
3.1.3 Wing Flowfields

Three dimensional flowfields about clean and iced wings at 0° AOA were determined using FLUENT. Due to availability of experimental data for comparison (Ref. 28), a CFD model of a swept wing positioned in the WSU 7-ft by 10-ft wind tunnel was developed, as shown in Fig. 3.10. The swept finite wing had a GLC-305 airfoil section, aligned in the streamwise direction. The section remained constant from root to tip. The wing had 28° leading edge sweep, 15.6° trailing edge sweep, 60-in long semispan, 7.3 ft² area (half wing), aspect ratio of 6.8 (left and right wing), taper ratio of 0.4 and linear geometric twist of 0° at the root and -4° (washout) at the tip. The wing root and tip chords were 25.2 inches and 10.08 inches respectively. In addition, the wing mean aerodynamic chord (MAC) was 18.72 inches and was located 25.74 inches from the wing root. In the wind tunnel tests, a streamlined body was used to raise the wing model above the tunnel floor boundary layer as illustrated in Fig. 3.11. The streamlined body included
in the CFD analyses was 55.6-in long and 4.4-in high. Computations were performed with the clean wing (Fig. 3.11) and the wing with a three-dimensional simulated 22.5-min glaze ice shape (Fig. 3.12) obtained using the LEWICE code and the method described in Ref. 28. Figures 3.13a and 3.13b show the grid topology in the vicinity of the clean and iced wing models respectively.

Fig. 3.10  Full view of the computational grid.

Fig. 3.11  Clean wing and streamlined body in test section.  
Fig. 3.12  Iced wing configuration; Looking from wing tip towards wing root.
FLUENT was used to compute the flowfields for the clean and iced GLC-305 swept wing at an angle of attack of 0°. The flowfield computations were performed for a freestream velocity of 146 mph (approximately 65.2 m/s) since the experiments in Ref. 28 were conducted at this speed. Velocity contours and streamlines of the computed flowfield for the different wing configurations are illustrated in Figs. 3.14 and 3.15. For the clean and iced wings at an AOA of 0°, the velocity contours at the wing root and wing tip are depicted in Figs. 3.14a and 3.14b respectively. Streamlines for the clean and iced wing at an angle of attack of 0° are presented in Figs. 3.15a and 3.15b respectively.

**Fig. 3.13 Close-up view of the computational grids.**
Fig. 3.14  Velocity contours for the clean and iced wings at AOA of 0°.

Fig. 3.15  Streamlines for the clean and iced wings at AOA of 0°.
3.1.4 CFD Flowfield Interpolation Methods

When determining the flowfield through CFD analysis, the computational domain was discretized using different types of elements. Triangular and tetrahedral elements, depicted in Fig. 3.16, were used to discretize 2D domains while tetrahedrons, pyramids, wedges and hexahedrons (Fig. 3.17) were used to discretize 3D domains. FLUENT provided flowfield properties (velocities, temperature, pressure etc.) at the nodes of these elements and interpolation routines were developed to determine flow properties within each element. Interpolation functions were used to express the flow properties within the element in terms of the flow properties at the nodes.

Fig. 3.16 Element shapes used to discretize 2D CFD domains.

Fig. 3.17 Element shapes used to discretize 3D CFD domains.
In the ice particle trajectory code, the interpolation routines commonly applied in finite element methods (Ref. 29, 30) were used to determined the velocity of the flowfield at the ice particle location so that aerodynamic forces and moments acting on the ice particle could be calculated. The interpolation methods used for the different types of grid elements are described in Sections 3.1.4.1 – 3.1.4.6. Generally, the procedure for interpolating the flowfield velocity at a point P (representing the location of the ice particle) inside a CFD element can be divided into two main parts:

a. Mapping the element and the point P from the global Cartesian coordinate space to a local parametric coordinate space using appropriate transformation equations. The parametric coordinates of any point inside the element vary from 0 to 1 or from -1 to 1 regardless of its location in Cartesian space.

b. Using the parametric coordinates of the point P in an interpolation function to determine the flowfield velocity at that point. The interpolation polynomials were isoparametric, i.e., they were of the same order as the polynomial used for mapping the elements from Cartesian to parametric space.

3.1.4.1 Triangular Element Interpolation

The Cartesian coordinates of any point in a triangular element can be written as a linear function of the Cartesian coordinates of its nodes as in equation 3.1.

\[
\begin{align*}
  x &= x_1N_1 + x_2N_2 + x_3N_3 \\
  y &= y_1N_1 + y_2N_2 + y_3N_3
\end{align*}
\]

(3.1)

where \((x_i,y_i)\) are the Cartesian coordinates of the nodes of the triangular element

\((x,y)\) are the Cartesian coordinates of a point within the triangular element

\(N_1, N_2, N_3\) are the shape functions of the triangular element
N₁, N₂ and N₃ provide a measure of the proximity of the point with the corresponding node. The shape functions have a value of one at respective nodes and zero at the other nodes, for example N₁ has a value of one at node 1 and zero at nodes 2 and 3. For a triangular element, the shape function values of a point (N₁, N₂, N₃) are equal to its parametric coordinates (p,q,r). The parametric coordinates measure the distance along the normal to the side of the triangle (as illustrated in Fig. 3.18) such that:

- p = 0 on side 23, p = 1 at node 1
- q = 0 on side 31, q = 1 at node 2
- r = 0 on side 12, r = 1 at node 3

Thus if a point (x,y) is in a triangular element, its coordinates in local parametric space will be in the range of 0 to 1 regardless of its coordinates in global Cartesian space.

![Parametric Coordinate 'p' Variation](image)

Fig. 3.18 Variation of parametric coordinate ‘p’ in a triangular element: p is zero on edge 12, one at node 1 and varies linearly in between.

Equation 3.1 is also the transformation equation from parametric space to Cartesian space for a triangular element. Any point in the triangular element can be written as a linear function of its parametric coordinates as in equation 3.2.

\[
\begin{align*}
\left\{ \begin{array}{c}
x = x₁p + x₂q + x₃r \\
y = y₁p + y₂q + y₃r
\end{array} \right.
\end{align*}
\] (3.2)
Another property of the parametric coordinates is that they sum to unity as in equation 3.3.

\[ p + q + r = 1 \]  \hspace{1cm} (3.3)

From equation 3.3 it can be seen that only two of the parametric coordinates are independent just as in the 2D Cartesian coordinate system. Combining equations 3.2 and 3.3, the complete mapping between parametric space and Cartesian space can be expressed in matrix form as shown in equation 3.4.

\[
\begin{bmatrix}
  x \\
  y \\
  1 \\
\end{bmatrix} = M \cdot \begin{bmatrix}
  N_1 \\
  N_2 \\
  N_3 \\
\end{bmatrix} = M \cdot \begin{bmatrix}
  p \\
  q \\
  r \\
\end{bmatrix} = \begin{bmatrix}
  x_1 & x_2 & x_3 \\
  y_1 & y_2 & y_3 \\
  1 & 1 & 1 \\
\end{bmatrix} \cdot \begin{bmatrix}
  p \\
  q \\
  r \\
\end{bmatrix}
\]  \hspace{1cm} (3.4)

For a point P \((x_P, y_P)\) in the triangular element, the matrix M in equation 3.4 was determined from the Cartesian coordinates of the nodes of the element containing the point. The parametric coordinates \(p_P, q_P\), and \(r_P\) were determined by numerically solving equation 3.4. The x- and y-velocities at the point P were then interpolated from the node velocities using equation 3.5. As the element was isoparametric, the function used to interpolate the velocity was similar to the function used to transform the element from Cartesian to parametric space.

\[
\vec{V}(x_p, y_p) = p_p \cdot \vec{V}_1 + q_p \cdot \vec{V}_2 + r_p \cdot \vec{V}_3
\]  \hspace{1cm} (3.5)

where \(\vec{V}_1, \vec{V}_2, \vec{V}_3\) are the flowfield velocity vectors at the nodes of the triangular element.

Equation 3.5 expresses the velocity at a point within the triangular element as a weighted average of the velocity values at the nodes of the element. The parametric coordinates of the point act as the weights in the average.
3.1.4.2 Quadrilateral Element Interpolation

The mapping from Cartesian space to parametric space for a quadrilateral element is shown in Fig. 3.19. This mapping is different from the mapping of the triangular element detailed in Section 3.1.4.1. In global Cartesian (x-y) space, the quadrilateral is irregularly shaped while in local parametric (ξ-η) space it is square shaped with sides ranging from -1 to 1 in the ξ- and η-directions. Table 1 provides the coordinates of the nodes in parametric space.

![Fig. 3.19 Quadrilateral elements in Cartesian and parametric coordinate systems.](image)

**Table 1 Local parametric coordinates of all nodes of a quadrilateral element**

<table>
<thead>
<tr>
<th>Node</th>
<th>ξ</th>
<th>η</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

The variation of any flowfield property (U) within the quadrilateral element can be written as a non-linear function of its parametric coordinates (ξ and η) as in equation 3.6.

\[
U = \alpha^U_1 + \alpha^U_2 \xi + \alpha^U_3 \eta + \alpha^U_4 \xi \eta \quad (3.6)
\]
The coefficients $\alpha_i^U, \alpha_2^U, \alpha_3^U$ and $\alpha_4^U$ can be determined from the values of the flow property at the nodes $(U_i)$ using equation 3.7. Substituting the parametric coordinates of the four nodes from Table 3.1 and the values of the flow property at the nodes $(U_i)$ into equation 3.6, the matrix relation in equation 3.8 is obtained. By inverting the $C$ matrix of equation 3.7, the values of $\alpha_j^U$ is found in terms of $U_i$ in equation 3.7.

$$\begin{bmatrix} \alpha_1^U \\ \alpha_2^U \\ \alpha_3^U \\ \alpha_4^U \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \leftrightarrow \{\alpha_j^U\} = [C]^{-1} \cdot \{U_i\}$$

(3.7)

$$\{U_i\} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} \alpha_1^U \\ \alpha_2^U \\ \alpha_3^U \\ \alpha_4^U \end{bmatrix} = [C] \cdot \{\alpha_j^U\}$$

(3.8)

Using the result $\{\alpha_j^U\} = [C]^{-1} \cdot \{U_i\}$ from equation 3.7, the polynomial expansion of equation 3.6 can be expressed as equation 3.9. The polynomials $N_i = N_i(\xi, \eta)$ are known as shape functions and they represent the proximity of the point to the respective quadrilateral nodes. The shape functions for a quadrilateral element are given by equation 3.10. The shape functions $N_i$ have a value of one at node $i$ and zero at other nodes.

$$U = \begin{bmatrix} 1 & \xi & \eta & \xi \eta \end{bmatrix} \cdot \{\alpha_j^U\} = \begin{bmatrix} 1 & \xi & \eta & \xi \eta \end{bmatrix} \cdot [C]^{-1} \cdot \{U_i\} = \left[ N_1 \quad N_2 \quad N_3 \quad N_4 \right] \cdot \{U_i\}$$

(3.9)

$$N_1 = \frac{1}{4} (1-\xi)(1-\eta) \quad N_3 = \frac{1}{4} (1+\xi)(1+\eta)$$

$$N_2 = \frac{1}{4} (1+\xi)(1-\eta) \quad N_4 = \frac{1}{4} (1-\xi)(1+\eta)$$

(3.10)
Equation 3.9 shows that the interpolation function of equation 3.6 is a weighted average of the flow property values at the nodes of the quadrilateral with the shape functions acting as the weights in the average equation.

If the interpolated velocity value for a point needs to be determined, equation 3.6 shows that its local parametric coordinates need to be found. To determine the parametric coordinates a transformation between the two coordinate systems needs to be developed. Since the quadrilateral element is isoparametric, the mapping functions are of the same type as the interpolation function of equation 3.6. Equation 3.11 shows the transformation functions from parametric to Cartesian space.

\[
\begin{align*}
\xi & = \alpha^\xi_1 + \alpha^\xi_2 \xi \eta + \alpha^\xi_3 \eta^2 + \alpha^\xi_4 \xi \eta^2 \\
y & = \alpha^y_1 + \alpha^y_2 \xi + \alpha^y_3 \eta + \alpha^y_4 \xi \eta
\end{align*}
\]  
(3.11)

The coefficients \(\alpha^\xi_j\) and \(\alpha^y_j\) in equation 3.11 were calculated from the Cartesian coordinates of the nodes \((x_i,y_i)\) in the same manner as in equation 3.8 where \(U_j\) was calculated from \(U_i\).

For a point \(P (x_P,y_P)\) in a quadrilateral in Cartesian space, its parametric coordinates \((\xi_P,\eta_P)\) were calculated iteratively because the mapping functions of equation 3.11 were the non-linear and inverse functions for the parametric coordinates \((\xi,\eta)\) in terms of the physical coordinates \((x,y)\) were not available. The Newton-Raphson method was used to compute the parametric coordinates of the point \(P\). An initial guess value for the parametric coordinates \((\xi_0,\eta_0)\) was made and the corresponding physical coordinates \((\hat{x}_0,\hat{y}_0)\) were calculated from equation 3.11. From the error between the actual physical coordinates \((x_P,y_P)\) and the physical coordinates \((\hat{x}_0,\hat{y}_0)\) calculated from the initial
guess, the corrections (Δξ, Δη) to the initial guess for the parametric coordinates were evaluated numerically by solving equation 3.12. For the next iteration of equation 3.12, the corrections to the initial guess value (Δξ, Δη) were used to compute the new values of the parametric coordinates (ξ₁, η₁) using equation 3.13. This process was repeated until the components of the error vector {error} in equation 3.12 were less than a set tolerance. The subscript ‘k’ denotes the kth-iteration while a ‘k’ of 0 corresponds to the initial guess value. Note that the matrix J in equation 3.12 is known as the Jacobian matrix and was computed by differentiating equation 3.11 with respect to ξ and η, and substituting the parametric coordinates (ξₖ, ηₖ) in the differentiated equation.

\[
\left\{ \begin{align*}
    x_p - \hat{x}_k = \frac{\partial x}{\partial \xi} \Delta \xi + \frac{\partial x}{\partial \eta} \Delta \eta \\
    y_p - \hat{y}_k = \frac{\partial y}{\partial \xi} \Delta \xi + \frac{\partial y}{\partial \eta} \Delta \eta 
\end{align*} \right. \Rightarrow \{ \text{error} \} = J \left[ \begin{align*}
    \Delta \xi \\
    \Delta \eta 
\end{align*} \right] 
\]

(3.12)

\[
\left\{ \begin{align*}
    \xi_{k+1} &= \xi_k + \Delta \xi \\
    \eta_{k+1} &= \eta_k + \Delta \eta 
\end{align*} \right.
\]

(3.13)

The parametric coordinates of the point P were then used in the interpolation functions of equation 3.14 to determine the x- and y-velocity components at the point P. The coefficients αₓ and αᵧ were determined from the velocity values at the element nodes in the same manner as αᵤ was calculated from Uᵢ in equation 3.8.

\[
\left\{ \begin{align*}
    V_x(x_p, y_p) &= \alpha_1 \xi + \alpha_2 \eta + \alpha_3 \xi \eta + \alpha_4 \xi^2 \eta + \alpha_5 \xi \eta^2 \\
    V_y(x_p, y_p) &= \alpha_6 \xi + \alpha_7 \eta + \alpha_8 \xi \eta + \alpha_9 \xi^2 \eta + \alpha_{10} \xi \eta^2 
\end{align*} \right.
\]

(3.14)
3.1.4.3 Tetrahedral Element Interpolation

The interpolation procedure used for a tetrahedral element was obtained from Ref. 31. The procedure was a 3D extension of the technique used with the 2D triangular element in Section 3.1.4.1. Equation 3.15 shows the transformation from parametric space to Cartesian space for a tetrahedral element. The parametric coordinates \((p,q,r,s)\) of a point in the tetrahedron are a measure of its distance along the face normals of the tetrahedron as shown in Fig. 3.20, and vary from 0 to 1. For a tetrahedron, the parametric coordinates of the point are also equal to the shape functions values \((N_1, N_2, N_3, N_4)\).

\[
\begin{bmatrix}
  x \\
  y \\
  z \\
  1
\end{bmatrix}
= \begin{bmatrix}
  x_1 & x_2 & x_3 & x_4 \\
  y_1 & y_2 & y_3 & y_4 \\
  z_1 & z_2 & z_3 & z_4 \\
  1 & 1 & 1 & 1
\end{bmatrix} \begin{bmatrix}
  p \\
  q \\
  r \\
  s
\end{bmatrix}
= M \begin{bmatrix}
  p \\
  q \\
  r \\
  s
\end{bmatrix}
= M \begin{bmatrix}
  N_1 \\
  N_2 \\
  N_3 \\
  N_4
\end{bmatrix}
\]

(3.15)

where \((x_i,y_i,z_i)\) are the Cartesian coordinates of the nodes of the tetrahedron

\((x,y,z)\) are the Cartesian coordinates of a point in the tetrahedron

\((p,q,r,s)\) are the parametric coordinates of the point in the tetrahedron

\(N_1, N_2, N_3, N_4\) are the shape functions of the tetrahedral element

Fig. 3.20 Variation of parametric coordinate ‘p’ normal to face 234.
For a point P \((x_p, y_p, z_p)\) in the tetrahedral element, the parametric coordinates \((p_p, q_p, r_p, s_p)\) were determined by numerically solving equation 3.15. The x-, y- and z-velocities at point P were then interpolated from the node velocities using equation 3.16.

\[
\vec{V}(x_p, y_p) = p_p \cdot \vec{V}_1 + q_p \cdot \vec{V}_2 + r_p \cdot \vec{V}_3 + s_p \cdot \vec{V}_4
\]

where \(\vec{V}_1, \vec{V}_2, \vec{V}_3, \vec{V}_4\) are the flowfield velocity vectors at the nodes of the tetrahedron

### 3.1.4.4 Pyramidal Element Interpolation

To interpolate velocity values at a point in a pyramidal element, the element was first split into two tetrahedral elements as shown in Fig 3.21. The tetrahedral element containing the point was then identified and the tetrahedral interpolation method in Section 3.1.4.3 was used to provide flow properties at the point.

![Fig. 3.21 Pyramidal element split into two tetrahedral elements.](image)

### 3.1.4.5 Wedge Element Interpolation

The interpolation method used for a wedge element was similar to that used for the 2D quadrilateral element in Section 3.1.4.2. The wedge element and the point P within it were mapped from Cartesian space to a parametric domain as shown in Fig. 3.22. The wedge in parametric space extended from -1 to 1 in the \(\zeta\)-direction and from 0
to 1 in the $\xi$- and $\eta$-directions. The mapping function used for the transformation is given in equation 3.17 (Ref. 32).

![Diagram of a wedge in Cartesian and parametric coordinate systems]

**Fig. 3.22** Wedge in Cartesian and parametric coordinate systems.

The coefficients $x_j^{\alpha}$, $y_j^{\alpha}$, $z_j^{\alpha}$ in equation 3.17 were calculated from the Cartesian coordinates of the nodes ($x_i,y_i,z_i$). Substituting the parametric coordinates of the nodes ($\xi_i,\eta_i,\zeta_i$) and the x-coordinates ($x_i$) of the nodes into the first relation (x-coordinate) in equation 3.17, the relation given in equation 3.18 was obtained. Inverting the matrix $C$ in equation 3.18 provided the $\{x_j^{\alpha}\}$ values as a function of the x-coordinates of the nodes as shown in equation 3.19. Similarly, the coefficients $\{y_j^{\alpha}\}$ and $\{z_j^{\alpha}\}$ were determined in terms of the y and z coordinates at the element nodes.
The Newton-Raphson iteration method using equation 3.20 was applied to arrive iteratively at the parametric coordinates of the point P from the initial guess values as in section 3.1.4.2. The Jacobian matrix J used in equation 3.20 was obtained by differentiating equation 3.17 with respect to $\xi$, $\eta$ and $\zeta$ and then substituting the parametric coordinates $(\xi_k, \eta_k, \zeta_k)$ in them.

\[
\begin{bmatrix}
\alpha_1^x \\
\alpha_2^x \\
\alpha_3^x \\
\alpha_4^x \\
\alpha_5^x \\
\alpha_6^x
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 1 & 0 & 0 \\
-1 & 1 & 0 & -1 & 1 & 0 \\
-1 & 0 & 1 & -1 & 0 & 1 \\
-1 & 0 & 0 & 1 & 0 & 0 \\
1 & -1 & 0 & -1 & 1 & 0 \\
1 & 0 & -1 & -1 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix}
\] (3.18)

The relation used for mapping from parametric to Cartesian space was also used to interpolate the velocity values at a point in the wedge. Equation 3.17 can be written for the velocity components $V_x$, $V_y$ and $V_z$ as shown in equation 3.21. The coefficients
\[ \{ \alpha^x_j \}, \{ \alpha^y_j \} \text{ and } \{ \alpha^z_j \} \] were determined in a manner similar to that used for computing the coefficients \( \{ \alpha^i_j \} \) with equation 3.19.

\[
\begin{align*}
V_x &= \alpha^x_1 + \alpha^x_2 \xi + \alpha^x_3 \eta + \alpha^x_4 \xi \zeta + \alpha^x_5 \eta \zeta + \alpha^x_6 \eta \xi \\
V_y &= \alpha^y_1 + \alpha^y_2 \xi + \alpha^y_3 \eta + \alpha^y_4 \xi \zeta + \alpha^y_5 \eta \zeta + \alpha^y_6 \eta \xi \\
V_z &= \alpha^z_1 + \alpha^z_2 \xi + \alpha^z_3 \eta + \alpha^z_4 \xi \zeta + \alpha^z_5 \eta \zeta + \alpha^z_6 \eta \xi
\end{align*}
\] (3.21)

Note that, the interpolation functions in equation 3.21 can be written in terms of shape functions \( (N_i) \) and the flow properties at the nodes \( (U_i) \) as shown in equation 3.22 using the result \( \{ \alpha^U_j \} = [C]^{-1} \cdot \{ U_i \} \) from equation 3.19. The shape functions are like the averaging weights for the interpolation function. Shape functions for a wedge are given in equation 3.23.

\[
U = \begin{bmatrix} 1 & \xi & \eta & \xi \zeta & \eta \zeta \end{bmatrix} \cdot \{ \alpha^U_j \} = \begin{bmatrix} 1 & \xi & \eta & \xi \zeta & \eta \zeta \end{bmatrix} \cdot [C]^{-1} \cdot \{ U_i \} = \sum_{i=1}^{6} N_i U_i
\]

\[
\begin{align*}
N_1 &= \frac{1}{2} (1 - \xi - \eta)(1 - \zeta) & N_4 &= \frac{1}{2} (1 - \xi - \eta)(1 + \zeta) \\
N_2 &= \frac{1}{2} \xi (1 - \zeta) & N_5 &= \frac{1}{2} \xi (1 + \zeta) \\
N_3 &= \frac{1}{2} \eta (1 - \zeta) & N_6 &= \frac{1}{2} \eta (1 + \zeta)
\end{align*}
\] (3.23)

3.1.4.6 Hexahedral Element Interpolation

The interpolation procedure used for a hexahedral element was obtained from Ref. 33. The technique was a 3D extension of the interpolation method used with 2D quadrilateral elements in Section 3.1.4.2. The mapped hexahedron in parametric
coordinate space extended from -1 to 1 along the $\xi$-, $\eta$-, and $\zeta$-axis as shown in Fig. 3.23.

Equation 3.24 gives the mapping relations from parametric space to Cartesian space.

$$
\begin{align*}
  x &= \alpha_1^x + \alpha_2^x \xi + \alpha_3^x \eta + \alpha_4^x \zeta + \alpha_5^x \xi \eta + \alpha_6^x \xi \zeta + \alpha_7^x \eta \zeta + \alpha_8^x \xi \eta \zeta \\
  y &= \alpha_1^y + \alpha_2^y \xi + \alpha_3^y \eta + \alpha_4^y \zeta + \alpha_5^y \xi \eta + \alpha_6^y \xi \zeta + \alpha_7^y \eta \zeta + \alpha_8^y \xi \eta \zeta \\
  z &= \alpha_1^z + \alpha_2^z \xi + \alpha_3^z \eta + \alpha_4^z \zeta + \alpha_5^z \xi \eta + \alpha_6^z \xi \zeta + \alpha_7^z \eta \zeta + \alpha_8^z \xi \eta \zeta
\end{align*}
$$

Equation 3.24

Substituting the parametric coordinates of the nodes ($\xi_i, \eta_i, \zeta_i$) and the x-coordinates ($x_i$) of the nodes into the first relation (x-coordinate) in equation 3.24, the relation given in equation 3.25a was obtained. Inverting the matrix $C$ in equation 3.25a provided the $\alpha_j^x$ values as a function of the x-coordinates of the nodes as shown in equation 3.25b. Similarly, the coefficients $\alpha_j^y$ and $\alpha_j^z$ were determined in terms of the y and z coordinates at the element nodes.

The Newton-Raphson method (see Sections 3.1.4.2 and 3.1.4.5) was applied to arrive iteratively at the parametric coordinates of the point $P$ from the initial guess value. The $x$, $y$ and $z$ velocities were then obtained from the parametric coordinates using
equation 3.26. The coefficients \( \{ x_j^\alpha \}, \{ y_j^\alpha \} \) and \( \{ z_j^\alpha \} \) were determined in a manner similar to that used for computing the coefficients \( \{ x_j^\alpha \} \) with equation 3.25b.

\[
\{ x_j \} = \begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7 \\
x_8 \\
\end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
\end{bmatrix} \begin{bmatrix} x_1^\alpha \\ x_2^\alpha \\ x_3^\alpha \\ x_4^\alpha \\ x_5^\alpha \\ x_6^\alpha \\ x_7^\alpha \\ x_8^\alpha \\
\end{bmatrix} = [C] \cdot \{ x_j^\alpha \} \quad (3.25a)
\]

\[
\begin{bmatrix} x_1^\alpha \\ x_2^\alpha \\ x_3^\alpha \\ x_4^\alpha \\ x_5^\alpha \\ x_6^\alpha \\ x_7^\alpha \\ x_8^\alpha \\
\end{bmatrix} = \frac{1}{8} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 \\
\end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\
\end{bmatrix} \Leftrightarrow \{ x_j^\alpha \} = [C]^{-1} \cdot \{ x_j \} \quad (3.25b)
\]

\[
\begin{align*}
V_x &= \alpha_1^{v_x} + \alpha_2^{v_x} \xi + \alpha_3^{v_x} \eta + \alpha_4^{v_x} \xi \eta + \alpha_5^{v_x} \xi \xi + \alpha_6^{v_x} \xi \xi \eta + \alpha_7^{v_x} \eta \xi + \alpha_8^{v_x} \xi \eta \xi \\
V_y &= \alpha_1^{v_y} + \alpha_2^{v_y} \xi + \alpha_3^{v_y} \eta + \alpha_4^{v_y} \xi \eta + \alpha_5^{v_y} \xi \xi + \alpha_6^{v_y} \xi \xi \eta + \alpha_7^{v_y} \eta \xi + \alpha_8^{v_y} \xi \eta \xi \\
V_z &= \alpha_1^{v_z} + \alpha_2^{v_z} \xi + \alpha_3^{v_z} \eta + \alpha_4^{v_z} \xi \eta + \alpha_5^{v_z} \xi \xi + \alpha_6^{v_z} \xi \xi \eta + \alpha_7^{v_z} \eta \xi + \alpha_8^{v_z} \xi \eta \xi \\
\end{align*}
\quad (3.26)
\]

Once again, interpolation functions can be written in terms of shape functions \( (N_i) \) and the flow properties at the nodes \( (U_i) \) as shown in equation 3.27. Shape functions for a hexahedron are given in equation 3.28.

\[
U = N_1 U_1 + N_2 U_2 + N_3 U_3 + N_4 U_4 + N_5 U_5 + N_6 U_6 + N_7 U_7 + N_8 U_8 \quad (3.27)
\]
\[ N_1 = \frac{1}{8} (1 - \xi)(1 + \eta)(1 - \zeta) \]
\[ N_2 = \frac{1}{8} (1 - \xi)(1 + \eta)(1 + \zeta) \]
\[ N_3 = \frac{1}{8} (1 + \xi)(1 + \eta)(1 + \zeta) \]
\[ N_4 = \frac{1}{8} (1 + \xi)(1 + \eta)(1 - \zeta) \]
\[ N_5 = \frac{1}{8} (1 - \xi)(1 - \eta)(1 - \zeta) \]
\[ N_6 = \frac{1}{8} (1 - \xi)(1 - \eta)(1 + \zeta) \]
\[ N_7 = \frac{1}{8} (1 + \xi)(1 - \eta)(1 + \zeta) \]
\[ N_8 = \frac{1}{8} (1 + \xi)(1 - \eta)(1 - \zeta) \]

3.2 Trajectory Simulation Models

Three, four and six degree of freedom trajectory simulation models were developed to simulate ice particle trajectories. The 3-DOF model can compute trajectories in 2D space when a simple ice shedding risk analysis is necessary. The 4-DOF simulation model could simulate the 3D trajectories of ice particles that rotate predominantly about one axis after shedding. The 6-DOF model can be utilized to compute 3D trajectories of ice particles that have a tendency to rotate about different axes. Appropriate equations of motion were used in each model to determine the angular and translational accelerations from the aerodynamic forces and moments acting on the ice particle at different instances of time. The accelerations were then integrated to obtain the path of the ice particle. The aerodynamic forces and moments for the ice particles presented in this thesis are described later in Section 3.4. The equations of motion used in the three trajectory simulation models are described in Sections 3.2.1-3.2.3.

3.2.1 3–DOF Trajectory Simulation Model

The equations of motion for the 3-DOF trajectory simulation model allowed the ice particle to move in a 2D plane and rotate about an axis perpendicular to the plane. The
ice particle was tracked in an axis system fixed to the airplane so that the location of the ice particle with respect to the airplane could be computed. Ice shedding is a fast event, thus the airplane is assumed only to translate during the event and hence the airplane axis system is considered to be an inertial axis system which is translating but not rotating (see Appendix A).

After an ice particle sheds from an aircraft surface, aerodynamic forces and moments act on it. For an axis system located at the centroid of the ice particle and parallel to the inertial (or airplane) axis system, the geometrical relationship of the angles and aerodynamic forces is shown in Fig. 3.24. The lift force ($L$) on the ice particle acts perpendicular to the relative velocity vector ($V$) and the drag force ($D$) acts in the direction opposite to the relative velocity vector. The angle of attack ($\alpha$) is the angle between the relative velocity vector and the ice particle. The orientation of the ice particle with respect to the x-axis is given by the orientation angle ($\delta$).

Fig. 3.24  Aerodynamic forces and moments acting on an ice particle after it sheds from an airplane surface.

where

- $L$ = Lift force on the ice piece
• **D** = Drag force on the ice piece
• **M** = Moment on the ice piece
• **V_{ice}** = Velocity of the ice piece in the inertial reference frame
• **W** = Wind velocity vector in the inertial reference frame
• **V** = Velocity of the ice piece relative to air
• **γ** = Inclination of the relative velocity vector V with the x-axis
• **α** = Angle of attack
• **δ** = Orientation of the ice piece with the x-axis

The translational accelerations \( (\ddot{x}_{adn}, \ddot{y}_{adn}) \) resulting from the aerodynamic forces as determined from Fig. 3.24 are given in equations 3.29 and 3.30. Adding the acceleration due to gravity gives the translational acceleration of the ice particle in the inertial axis system (equation 3.31).

\[
\ddot{x}_{adn} = -\frac{1}{m_{ice}} \left( L \sin γ + D \cos γ \right) \\
\ddot{y}_{adn} = \frac{1}{m_{ice}} \left( L \cos γ - D \sin γ \right)
\]  

(3.29)  

(3.30)

where \( m_{ice} \) is the mass of the ice particle

\[
\begin{bmatrix}
\dot{x} \\
\dot{y}
\end{bmatrix} = \begin{bmatrix}
\ddot{x}_{adn} \\
\ddot{y}_{adn}
\end{bmatrix} + \begin{bmatrix}
g_x \\
g_y
\end{bmatrix}
\]  

(3.31)

The angular acceleration of the ice particle was determined using equation 3.32 from the aerodynamic moment (M), the aerodynamic rotational damping coefficient \( (C_{damp}) \) and the angular velocity \( (\dot{δ}) \).
\[ \ddot{\delta} = \frac{1}{I}(M - C_{\text{damp}} \dot{\delta}) \]  \hspace{1cm} (3.32)

where \( I \) is the mass moment of inertia of the ice particle.

The damping coefficient, \( C_{\text{damp}} \), quantifies the ice particle’s opposition to rotation. The value of \( C_{\text{damp}} \) for the semi-circular ice shape was assumed to be zero while the \( C_{\text{damp}} \) value for the square and rectangular ice shapes ranged from 0 to 0.025 ft\cdotlb\cdotsec and were obtained from Ref. 15. The damping coefficient values are estimates and there are no experimental data available for damping coefficients of ice shapes.

The lift, drag and moment values for an ice particle are dependent on the relative velocity (\( V \)) and the angle of attack (\( \alpha \)). The relative velocity of the ice particle with respect to the airstream is the difference between the velocity of the ice particle (\( V_{\text{ice}} \)) and the flowfield velocity (\( W \)) as shown in equation 3.33. The inclination (\( \gamma \)) of the relative velocity vector with the x-axis was obtained using equation 3.34 from the x and y components of the relative velocity vector (\( V_x \) and \( V_y \)). The inclination angle (\( \gamma \)) of the relative velocity and the orientation of the ice particle (\( \delta \)) were used to determine the angle of attack (\( \alpha \)) using equation 3.35.

\[ V = V_{\text{ice}} - W \] \hspace{1cm} (3.33)

\[ \gamma = \tan^{-1}\left(\frac{V_y}{V_x}\right) \] \hspace{1cm} (3.34)

\[ \alpha = \delta - \gamma \] \hspace{1cm} (3.35)

Equations 3.31 and 3.32 constitute the 3-DOF equations of motion from which the translational and angular accelerations could be determined from the forces and moments.
acting on the ice particle. The accelerations were integrated using the 4th order Runge-Kutta scheme to give the trajectory of the ice particle.

### 3.2.2 4–DOF Trajectory Simulation Model

The 4-DOF trajectory simulation model was similar to the model described in Ref. 15. The model simulated ice particle trajectories in 3D when the rotation of the ice particle was restricted to only one axis. The equations of motion for the 4-DOF model (equations 3.35 to 3.39) were the same as that for the 3-DOF model described in Section 3.2.1 with an additional relation (equation 3.37) for the acceleration in the z-direction due to aerodynamic side force (SF).

\[
\begin{align*}
\ddot{x}_{adn} &= -\frac{1}{m_{\text{ice}}} (L \sin \gamma + D \cos \gamma) \\
\ddot{y}_{adn} &= \frac{1}{m_{\text{ice}}} (L \cos \gamma - D \sin \gamma) \\
\ddot{z}_{adn} &= \frac{SF}{m_{\text{ice}}} \\
\{\ddot{x}\} &= \{\ddot{x}_{adn}\} + \{\ddot{g}\} \\
\{\ddot{y}\} &= \{\ddot{y}_{adn}\} + \{\ddot{g}\} \\
\{\ddot{z}\} &= \{\ddot{z}_{adn}\} + \{\ddot{g}\} \\
\ddot{\delta} &= \frac{1}{I} (M - C_{\text{damp}} \dot{\delta})
\end{align*}
\]

### 3.2.3 6–DOF Trajectory Simulation Model

The 6-DOF trajectory simulation model used three body forces (normal, axial and side forces) and three moments (pitching, yawing and rolling moments) to compute the translational and angular acceleration of the ice particle. The 6-DOF equations of motion
are broken down into translational motion due to body forces and rotational motion due
to moments. Figure 3.25 illustrates an ice particle acted upon by a force and a moment.
The force (\( \mathbf{F} \)) is the resultant vector of the normal, axial and side forces acting on the
body. The moments \( M^b_x, M^b_y, M^b_z \) are the components of the moment vector \( \mathbf{M} \) and
represent the rolling, pitching and yawing moments respectively. Figure 3.25 also depicts
the body and the inertial axis systems used in the equations of motion (see Appendix A).
Superscripts ‘i’ and ‘b’ designate the inertial reference frame and body reference frame
respectively. The inertial system is a non-rotating axis system while the body axis system
is a rotating axis system that is aligned with the ice particle as shown in Fig. 3.25.

![Fig. 3.25 Forces and moments acting on an ice particle.](image)

where \( \mathbf{F} \) is the resultant aerodynamic force vector,

\[ M^b_x, M^b_y, M^b_z \] are the moments acting about the body axes,

\[ x^i, y^i, z^i \] are the axes of the inertial reference frame, and

\[ x^b, y^b, z^b \] are the axes of the body reference frame

The position of the ice particle was computed with respect to the inertial axis
system, and the orientation of the body with respect to the inertial axis system was
determined using quaternions (see Appendix B). The equations of motion were used to
compute the position and orientation of the body at the time step, \( t+dt \), from the forces and moments acting on it at time, \( t \).

The translational acceleration of the body in the inertial reference frame was computed using Newton second law of motion given in equation 3.40. The force vector \( \mathbf{F}_{\text{Body}} \) is the vector \( \mathbf{F} \) illustrated in Fig. 3.25 transformed from body reference frame to inertial reference frame. The velocity and position were determined by integrating the accelerations.

\[
\dot{x}^i = \frac{\mathbf{F}_{\text{Body}}^i}{m_{\text{ice}}} + g^i
\]  

(3.40)

where \( \dot{x}^i \) = Acceleration vector in inertial reference frame

\( g^i \) = Gravity vector in inertial reference frame

\( \mathbf{F}_{\text{Body}}^i \) = Body force vector in inertial reference frame

\( m_{\text{ice}} \) = Mass of the ice particle

The rotational motion was governed by Euler’s equations of motion. For a body rotating with angular velocity \( \mathbf{\omega}^b \), equation 3.41 provides the relationship between the moments \( \mathbf{M}^b \) and moment of momentum \( \mathbf{H}^b \) in the body reference frame (Ref. 34). The body moment of momentum \( \mathbf{H}^b \) is a product of the body’s inertia matrix ‘[I]’ and its angular velocity vector \( \mathbf{\omega}^b \) as shown in equation 3.42 (Ref. 34).

\[
\mathbf{M}^b = \frac{d}{dt} \left( \mathbf{H}^b \right) + \mathbf{\omega}^b \times \mathbf{H}^b 
\]

(3.41)

\[
\mathbf{H}^b = [I] \mathbf{\omega}^b
\]

(3.42)

where \( \mathbf{M}^b \) = Moment acting on the body in body reference frame

\( \mathbf{H}^b \) = Moment of momentum in body reference frame
\( \omega^b \) = The angular velocity of the body about its axes

\([I]\) = The inertia matrix (described in equations 3.48-3.54)

Substituting the expression for the body moment of momentum \((H^b)\) from equation 3.42 into equation 3.41, the relation in equation 3.43 for the body moments \((M^b)\) is obtained.

\[
\begin{align*}
M^b &= [I] \frac{d}{dt} (\omega^b) + (\omega^b \times [I] \omega^b) \\
\frac{d}{dt} (\omega^b) &= [I]^{-1} \left[ M^b - (\omega^b \times [I] \omega^b) \right] 
\end{align*}
\] (3.43)

where \( \frac{d}{dt} (\omega^b) = \begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} \) = angular acceleration in body reference frame

\[ M^b = \begin{bmatrix} M_x^b \\ M_y^b \\ M_z^b \end{bmatrix} \] = moment of momentum in body reference frame

\( \omega^b = \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \) = angular velocity in body reference frame

In addition, equation 3.43 can be written in matrix form as shown in equation 3.44. \( M_x^b, M_y^b \) and \( M_z^b \) represent the moments acting about the body axes (Fig. 3.25). Note that \( P, Q \) and \( R \) denote the angular velocities about the \( x^b, y^b \) and \( z^b \) body axes respectively.

\[
\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}^{-1} \begin{bmatrix} M_x^b \\ M_y^b \\ M_z^b \end{bmatrix} - \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \] (3.44)
Equation 3.44 is solved to obtain the angular accelerations in the body reference frame. The angular accelerations are then integrated to determine the angular velocities. Next, using equation 3.45 (Ref. 35), the quaternion rates are computed from the angular velocity vector \( \omega^b \). Note that the quaternion \( q \) in equation 3.45a represents the orientation of the body at time \( t \).

\[
q = [q_1, q_2, q_3, q_4]
\]

where \( q_1, q_2, q_3, q_4 \) = Quaternion element depicting the orientation of the body

\[
\begin{align*}
\dot{q}_1 &= -\frac{1}{2}(q_2P + q_3Q + q_4R) + \lambda q_1 \\
\dot{q}_2 &= \frac{1}{2}(q_1P + q_3R - q_4Q) + \lambda q_2 \\
\dot{q}_3 &= \frac{1}{2}(q_1Q + q_4P - q_2R) + \lambda q_3 \\
\dot{q}_4 &= \frac{1}{2}(q_1R + q_2Q - q_3P) + \lambda q_4
\end{align*}
\]

(3.45a)

where \( \lambda \) is known as the integration drift correction gain given by

\[
\lambda = 1 - \left( q_1^2 + q_2^2 + q_3^2 + q_4^2 \right).
\]

The quaternion rates are numerically integrated to obtain the quaternion that describes the new orientation of the body at time \( t + dt \). The Euler angles \( (\psi, \theta \) and \( \phi \) are computed next from the quaternion elements using equation 3.45b so as to make it easier to visualize the orientation of the particle in the inertial axis system (see Appendix B).

\[
\begin{align*}
\psi &= \cos^{-1}\left( \frac{q_1^2 + q_2^2 - q_3^2 - q_4^2}{\cos \theta} \right) \cdot \text{sign}[2(q_3q_4 + q_1q_2)] \\
\theta &= \sin^{-1}\left[ -2(q_4q_3 - q_1q_2) \right] \\
\phi &= \cos^{-1}\left( \frac{q_2^2 - q_3^2 - q_4^2 + q_1^2}{\cos \theta} \right) \cdot \text{sign}[2(q_1q_4 + q_3q_2)]
\end{align*}
\]

(3.45b)
3.3 Ice Particle Mass Properties

The mass and moment of inertia of the ice particle are needed by the trajectory simulation model when computing the translational and angular accelerations of the ice particle from the aerodynamic forces and moments. The mass of the ice particle \( m_{\text{ice}} \) is the product of its volume \( v_{\text{ice}} \) and density \( \rho_{\text{ice}} \) as shown in equation 3.46.

\[
m_{\text{ice}} = \rho_{\text{ice}} \cdot v_{\text{ice}}
\]

where \( \rho_{\text{ice}} = 1.7793 \) slugs/ft\(^3\) or 57.2464 lbm/ft\(^3\)

The moment of inertia is a measure of the resistance an object offers to angular moments just as mass is a measure of the resistance to linear forces. For a body rotating about an axis, as illustrated in Fig. 3.26, the moment of inertia \( I \) for the body about that axis is given by equation 3.47.

\[
I = \int r^2 \cdot dm = \int \rho_{\text{body}} \cdot r^2 \cdot dv
\]

where \( dm \) = differential mass of the body
\( r \) = distance of ‘dm’ from the axis of rotation (see Fig. 3.26)
\( dv \) = volume of mass dm

Fig. 3.26 A body rotating about an axis.
\( \rho_{\text{body}} = \text{density of the body} \)

An inertia matrix \([I]\) is required when the axis of rotation of a body keeps changing. The components of the inertia matrix are depicted in equation 3.48 and their values depend on the body geometry, the choice of origin and the choice of coordinate axes. The diagonal elements of the matrix (\(I_{xx}, I_{yy}\) and \(I_{zz}\)) are known as ‘moments of inertia’ and can be calculated using equations 3.49 to 3.51 and the off-diagonal elements (\(I_{xy}, I_{yx}, I_{yz}, I_{zy}, I_{zx}\) and \(I_{xz}\)) are known as ‘products of inertia’ calculated using equations 3.52 to 3.54. For an axis of rotation ‘l’ oriented along unit vector \(\hat{u}\) and passing through the origin of the chosen coordinate axis system, the moment of inertia about ‘l’ can be computed using the inertia matrix with equation 3.55.

\[
[I] = \begin{bmatrix}
I_{xx} & I_{xy} & I_{xz} \\
I_{yx} & I_{yy} & I_{yz} \\
I_{zx} & I_{zy} & I_{zz}
\end{bmatrix}
\]  

(3.48)

\[
I_{xx} = \iiint \rho_{\text{ice}} (y^2 + z^2) \cdot dx dy dz
\]  

(3.49)

\[
I_{yy} = \iiint \rho_{\text{ice}} (x^2 + z^2) \cdot dx dy dz
\]  

(3.50)

\[
I_{zz} = \iiint \rho_{\text{ice}} (x^2 + y^2) \cdot dx dy dz
\]  

(3.51)

\[
I_{xy} = I_{yx} = \iiint \rho_{\text{ice}} (-xy) \cdot dx dy dz
\]  

(3.52)

\[
I_{yz} = I_{zy} = \iiint \rho_{\text{ice}} (-yz) \cdot dx dy dz
\]  

(3.53)

\[
I_{zx} = I_{xz} = \iiint \rho_{\text{ice}} (-zx) \cdot dx dy dz
\]  

(3.54)

\[
I_{l} = \hat{u} \cdot [I] \cdot \hat{u}
\]  

(3.55)

The dimensions of the square and rectangular plates are shown in Figs. 3.27 and 3.28 respectively, along with the principal axis system in which the inertia matrix was
calculated. Figure 3.29 shows the principal axis system in which the inertia matrix of the semi-circular shell was calculated and Fig 3.30 portrays the dimensions of the semi-circular shell in x-z plane of the axis system. For all the ice shapes, the origin of the principal axis system was located at their center of mass.

![Fig. 3.27 Dimensions of the square plate ice shape and axis system in which its inertia matrix was calculated.](image1)

![Fig. 3.28 Dimensions of the rectangular plate ice shape and axis system in which its inertia matrix was calculated.](image2)

![Fig. 3.29 Axis system in which the inertia matrix for the semi-circular shell ice shape was calculated.](image3)

![Fig. 3.30 Dimensions of the semi-circular shell's cross-section.](image4)
The mass and inertia matrix of the square plate, rectangular plate and the semi-circular shell were computed, and are given in Table 2. The trajectory simulation models utilized the mass and moment of inertia values when computing the translational and angular accelerations of the ice particle. The 3-DOF and 4-DOF trajectory model utilized the $I_{xx}$, $I_{yy}$ or $I_{zz}$ values of the inertia matrix depending on the axis the ice particle was assumed to rotate (x, y, or z axis) about. When computing 6-DOF trajectories, the complete inertia matrix was used since the ice particle was free to rotate about any axis passing through its center of mass.

### Table 2 Mass and inertia matrix for square, rectangular and semi-circular ice shapes

<table>
<thead>
<tr>
<th></th>
<th>Mass, $m_{\text{ice}}$ (lbm)</th>
<th>Inertia Matrix, $I$ (slugs-ft(^2))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Square Plate</strong></td>
<td>1.26</td>
<td>$\begin{pmatrix} 5.971 \times 10^{-4} &amp; 0 &amp; 0 \ 0 &amp; 5.971 \times 10^{-4} &amp; 0 \ 0 &amp; 0 &amp; 1.194 \times 10^{-3} \end{pmatrix}$</td>
</tr>
<tr>
<td><strong>Rectangular Plate</strong></td>
<td>0.74</td>
<td>$\begin{pmatrix} 1.928 \times 10^{-3} &amp; 0 &amp; 0 \ 0 &amp; 4.819 \times 10^{-4} &amp; 0 \ 0 &amp; 0 &amp; 2.409 \times 10^{-3} \end{pmatrix}$</td>
</tr>
<tr>
<td><strong>Semi-circular Shell</strong></td>
<td>0.8</td>
<td>$\begin{pmatrix} 2.492 \times 10^{-3} &amp; 0 &amp; 0 \ 0 &amp; 1.258 \times 10^{-4} &amp; 0 \ 0 &amp; 0 &amp; 2.411 \times 10^{-3} \end{pmatrix}$</td>
</tr>
</tbody>
</table>
3.4 Ice Particle Aerodynamic Characteristics

Aerodynamic characteristics of an ice particle include the forces and moments it experiences when placed in a stream of air (or freestream). The aerodynamic characteristics are dependent on the ice particle’s orientation relative to the freestream, the velocity of the freestream and the ice particle’s shape and size.

The aerodynamic characteristics of the rectangular plate and the semi-circular shell were obtained from wind tunnel tests (Ref. 14) while the aerodynamic characteristics of the square plate were obtained from Ref. 15. In Refs. 14 and 15, the aerodynamic forces and moments on different ice particles are given in non-dimensional coefficient form which are independent of ice particle size and freestream velocity. Using equations 3.56 and 3.57, the aerodynamic forces and moments experienced by the ice particle were computed from the force and moment coefficients (\(C_F\) and \(C_M\)).

\[
F = \left(\frac{1}{2} \rho_{\infty} V^2 S\right) C_F \quad (3.56)
\]

\[
M = \left(\frac{1}{2} \rho_{\infty} V^2 Sc\right) C_M \quad (3.57)
\]

where

- \(F\) is an aerodynamic force
- \(M\) is an aerodynamic moment
- \(\rho_{\infty}\) is the density of the freestream
- \(V\) is the relative velocity of the ice particle with respect to the freestream
- \(S\) is the reference area of the ice particle
- \(c\) is the chord length of the ice particle

The reference area (\(S\)) and the chord length (\(c\)) of different ice particles are given in Fig. 3.31. For the square plate and the rectangular plate, the reference area is the planar...
area and for the semi-circular shell it is the projected area. The chord length of the semi-
circular shell was the radius \((d/2)\) and the chord length for the square plate and the
rectangular plate was the breadth as depicted in Fig. 3.31.

![Reference area and chord length of different ice particles.](image)

**Fig. 3.31 Reference area and chord length of different ice particles.**

### 3.4.1 Aerodynamic Data for 3-DOF and 4-DOF Trajectory Simulations

The 3-DOF ice particle trajectory simulation model requires the lift, drag and
pitching moment acting on the ice particle and the 4-DOF trajectory simulation model
needs an additional side force value. The aerodynamic data used in the simulation of
different ice particle trajectories are presented in Sections 3.4.1.1 to 3.4.1.3.

#### 3.4.1.1 Square Plate

Empirical aerodynamic data for a low aspect ratio flat plate were obtained from
Ref. 15. The lift and drag force coefficients \((C_L, C_D)\) for \(0^\circ\) to \(90^\circ\) angles of attack \((\alpha)\)
is given by empirical equations 3.58 and 3.59. The variation of lift and drag coefficients
for \(\alpha\) from \(0^\circ\) to \(360^\circ\) is illustrated in Fig. 3.32.

\[
\begin{cases}
C_L = 1.31 \tan \alpha & \text{for } 0^\circ \leq \alpha \leq 35^\circ \\
C_L = 1.17 \cos \alpha & \text{for } 35^\circ < \alpha \leq 90^\circ
\end{cases}
\]  

(3.58)
\[
\begin{align*}
C_D &= 1.25 \cdot \left(0.01 + 0.531C_L^2\right) \quad \text{for } 0^\circ \leq \alpha \leq 35^\circ \\
C_D &= 1.25 \cdot |1.17 \sin \alpha| \quad \text{for } 35^\circ < \alpha \leq 90^\circ
\end{align*}
\]  
(3.59)

**Fig. 3.32 Variation of lift and drag coefficients for a square plate.**

The trajectory simulation model presented in Ref. 15 assumes that the pitching moment at the center of gravity of the ice particle was due to the normal force acting at the center of pressure of the ice particle (Fig. 3.33). The normal force was the result of the aerodynamic lift and drag and is given by equation 3.60. The center of pressure was assumed to be at the quarter chord location when the angle of attack was less than 35°. For this case, the distance between the center of pressure and the center of gravity of the ice particle was set equal to the quarter chord length. At angles of attack greater than 35°, the center of pressure was set closer to the center of gravity and the distance between these points was reduced to 1/20th of the chord length. The empirical curve for the pitching moment coefficient is given by equation 3.61 and was obtained using these assumptions. The variation of pitching moment with angle of attack from 0° to 360° is plotted in Fig. 3.34.
Fig. 3.33 Aerodynamic forces acting at the center of pressure of the square plate causing a moment at the center of gravity.

\[
C_N = C_D \cdot \sin(\alpha) + C_L \cdot \cos(\alpha) \quad (3.60)
\]

\[
\begin{align*}
C_m &= \left( \frac{C_N(\alpha)}{4} \right) & \text{for } 0^\circ \leq \alpha \leq 35^\circ \\
C_m &= \left( \frac{C_N(\alpha)}{20} \right) & \text{for } 35^\circ < \alpha \leq 90^\circ
\end{align*} \quad (3.61)
\]

Fig. 3.34 Variation of pitching moment coefficient for a square plate.

The coefficient of side force \(C_{SF}\) on the square plate was assumed to be equal to the coefficient of drag at zero degree angle of attack \((C_{D0} = 0.0125)\) as shown in equation...
3.62. The side force on the ice particle was computed using equation 3.63 using the side component (z-component) of the relative velocity ($V_z$).

$$C_{SF} = C_{D0}$$  \hspace{1cm} (3.62)

$$SF = \left( \frac{1}{2} \rho V_z^2 \right) S C_{D0}$$ \hspace{1cm} (3.63)

3.4.1.2 Rectangular Plate

The aerodynamic characteristics for the rectangular plate was obtained from tests conducted at the WSU 7-ft by 10-ft wind tunnel (Ref. 14). The variation of lift, drag, side force and pitching moment coefficients with angle of attack for a 12" × 6" rectangular plate with the 12" side of the plate facing the wind are illustrated in Figs. 3.35 and 3.36.

Fig. 3.35 Variation of lift and drag coefficients for a rectangular plate.
Fig. 3.36 Variation of pitching moment and side force coefficients for a rectangular plate.

Figures 3.37 compares the lift and drag coefficients and 3.38 illustrates the difference in pitching moment between the empirical aerodynamic data of the square plate and the experimentally obtained aerodynamic data of the rectangular plate for angle of attacks varying from $0^\circ$ to $90^\circ$.

Fig. 3.37 Comparison of lift and drag coefficients of the rectangular plate and the square plate.
3.4.1.3 Semi-circular Shell

The aerodynamic characteristics for the semi-circular shell were also obtained from wind tunnel tests (Ref. 14) and the variation of lift, drag, side force and pitching moment coefficients with angle of attack are illustrated in Figs. 3.39 and 3.40.
3.4.2 Aerodynamic Data for 6-DOF Trajectory Simulations

The 6-DOF trajectory simulation model requires the normal, axial and side forces and the pitching, yawing and rolling moments on the ice particle. The experimental aerodynamic data of the rectangular plate and the manner in which the experimental data was interpolated are given in Sections 3.4.2.1 and 3.4.2.2 respectively.

3.4.2.1 Aerodynamic Characteristics of the Rectangular Plate

The aerodynamic data for the 12" × 6" rectangular plate, with the 12" side facing the wind, was obtained from wind tunnel tests conducted at the WSU 7-ft by 10-ft wind tunnel. The aerodynamic force and moment data experienced by the rectangular plate at different orientations to the flow is illustrated are Figs. 3.41 and 3.42. Figure 3.41 depicts the normal, axial and side force coefficients and rolling, pitching and yawing moment coefficients on the rectangular plate oriented at 0° roll for different pitch and yaw angle
orientations. Similarly Fig. 3.42 depicts the variation of forces and moment coefficients on the ice particle at different pitch and yaw angle orientations for the rectangular plate at a roll angle of 30°. A detailed description of the wind tunnel tests and the complete aerodynamic database for the rectangular plate is given in Ref. 14.

3.4.2.2 Interpolation of Experimental Data

The aerodynamic forces and moments on the ice particle depend on the orientation of the relative velocity vector with the body axes. Two angles express this orientation: the angle of attack (\( \alpha \)) and the angle of sideslip (\( \beta \)) as shown in Fig. 3.43. The values for the orientation angles are determined from the components of the relative velocity vector in body axes using equations 3.64 and 3.65.

\[
\alpha = \tan^{-1} \frac{V^b_x}{V^b_z} \tag{3.64}
\]

\[
\beta = \sin^{-1} \frac{V^b_y}{|V|} \tag{3.65}
\]

The \( \alpha \) and \( \beta \) angles for which aerodynamic data were reduced from the wind tunnel test data are shown in Fig. 3.44. Because the rectangular plate is symmetric about its three body axes the aerodynamic data for a relative velocity vector pointing in any direction can be determined from \( \alpha \) and \( \beta \) angles ranging from 0° to 90°.
Fig. 3.41 Variation of force and moment coefficients with yaw and pitch angles for the rectangular plate at 0° roll angle.
Fig. 3.42 Variation of force and moment coefficients with yaw and pitch angles for the rectangular plate at 30° roll angle.
Fig. 3.43  Angle of attack (\(\alpha\)) and angle of sideslip (\(\beta\)) of the relative velocity vector (\(V\)).

where  \(V\) = Relative velocity vector of the ice particle with respect to the freestream

\(x^b, y^b, z^b\) = Axes of the body reference frame

\(V^b_x, V^b_y, V^b_z\) = Components of the relative velocity vector in the body axes

Fig. 3.44  Angle of attack (\(\alpha\)) and angle of sideslip (\(\beta\)) combinations for which aerodynamic data are known.

The aerodynamic data for any combination of \(\alpha_p\) and \(\beta_p\) was determined by finding out a triangle enclosing the point \((\alpha_p, \beta_p)\) in Fig. 3.44 and using the triangular
interpolation method described in Section 3.1.4.1 to interpolate the aerodynamic data (see Fig. 3.45).

![Diagram of triangular interpolation method](image)

**Fig. 3.45 Triangular interpolation method used to obtain experimental aerodynamic data.**

where \((\alpha_P, \beta_P)\) = Angle of attack and angle of sideslip for which aerodynamic data needs to be determined

\((\alpha_1, \beta_1), \ (\alpha_2, \beta_2), \ (\alpha_3, \beta_3)\) = Angle of attack and angle of sideslip for which aerodynamic data are known.
CHAPTER 4

MONTE CARLO METHOD

The ice particle trajectory is influenced by various parameters that are stochastic (random) in nature. Examples include size and shape of the ice particle, shedding location, initial orientation of the ice piece etc. The Monte Carlo method was selected to model uncertainties related to ice shedding. This method has been used by many investigators to simulate phenomena that are random in nature (Refs. 23, 24).

Figure 4.1 illustrates the manner in which Monte Carlo simulations were applied to the ice particle trajectory analysis. The uncertain input parameters in the trajectory simulation model were identified, for example, the size and initial orientation of the ice particle. The limits for the uncertain parameters were set, and a random number generator was used to produce random variations in the uncertain input parameters within the set limits. A large number of input vectors containing values of the uncertain parameters were generated using a random number generator, and a trajectory analysis was performed using each input vector. The points where the trajectories passed a given downstream station such as the engine inlet plane were recorded as depicted in Fig. 4.2 to give the ‘footprint map’ of the ice particle. From this footprint map a probability map was computed for the ice particle passing through a particular location at the engine inlet plane. The footprint map of the ice particle trajectory could be obtained on any desired plane. However, the footprint map of the trajectory at the engine inlet plane was computed because shed ice particles striking the engine would pose a hazard to the airplane.
The random number generator used to vary the uncertain input parameters had a uniform probability distribution, that is any value of the parameter between its limits had an equal probability of being selected. A large number of simulations were performed for random variations in input parameters to evaluate the influences of various combinations of uncertain parameters on the trajectory of the ice particle. The main drawback of the Monte Carlo method is that it is time consuming since a large number of trajectories (e.g. 60,000) need to be computed to get an accurate probability distribution.
CHAPTER 5

SIMULATION TOOLS

The ice particle trajectory simulation methodology was programmed on two platforms - FORTRAN and MSC.Easy5. MSC.Easy5 is a commercial simulation package utilized in the aerospace industry.

5.1 FORTRAN: WSU Ice Particle Trajectory Code

A FORTRAN program called the WSU Ice Particle Trajectory Code was developed using Compaq Visual Fortran Professional Edition 6.1.0. The code consists of a trajectory simulation module and a flowfield interpolation module as shown in Fig. 5.1. The first module is capable of simulating 3-, 4- and 6-DOF ice particle trajectories, determining the aerodynamic forces and moments on the ice particle and performing Monte Carlo simulations. This module was connected to a flowfield interpolation module which read flowfields computed with FLUENT, interpolated the flowfield velocity at the ice particle location and provided the velocity to the trajectory simulation program. The interpolation module was used when trajectory simulations needed to be performed in non-uniform CFD flowfields.

5.2 MSC.Easy5: Ice Particle Trajectory Simulation Models

MSC.Easy5 is a graphical user interface based software employed to design, model and simulate dynamic systems characterized by differential, difference and algebraic equations (Ref. 36). This software package has the capability of performing a
variety of linear and non-linear analyses. Models are built from primitive functional blocks, like summers, dividers, integrators, etc. and application specific blocks from specialized libraries such as aerospace, thermal, hydraulic, pneumatic and others. The aerospace vehicle library of MSC.Easy5 contains built-in modules (codes) or “components” that perform various functions required to build aerospace vehicle simulation models. The components are classified into categories of body dynamics, aerodynamics, environment, instruments and controls based on their functions. These components were primarily developed to build aerospace vehicle simulation models efficiently and to compute their flight paths. MSC.Easy5 has the ability to incorporate user developed codes in the model along with the built-in components.

**Fig. 5.1 Schematic of the WSU ice particle trajectory code.**

Although MSC.Easy5 was not designed for ice shedding simulations, it contains all the required mathematical models for trajectory analysis. Thus, an effort was made to develop a methodology that would allow the integration of MSC.Easy5 with FLUENT flowfield data and ice fragment aerodynamic coefficients obtained from wind tunnel tests to compute ice fragment trajectories. The main objective of this effort was to provide the
aircraft industry with an option of using commercially available software to conduct ice shedding analysis.

The ice particle trajectory simulation models were developed in MSC.Easy5 Version 7.2.1. The MSC.Easy5 models developed were capable of simulating ice particle trajectories released in uniform and non-uniform flowfields.

5.2.1 3- and 4-DOF Trajectory Simulation Model

The schematic of the MSC.Easy5 model developed for 3- and 4-DOF ice particle trajectory analysis presented in Fig. 5.2. Each block in the schematic signifies a module that performs a specific task. MSC.Easy5 refers to these modules as components. The ‘Modified Rigid Body Equations of Motion’ module, also called the RB module, contains non-linear, 6-DOF rigid body equations of motion. It computes the trajectory from inputs of the six body forces and moments along the x-, y- and z- body axes. For computing the 3-DOF trajectories, the forces along the x- and z- body axes with moments about the y-axis were non-zero while the other forces and moments were set to zero. In the 4-DOF simulations, the forces along the three body axes and the moment about the y-body axis were considered.

The RB module is a modified version of a built-in module in MSC.Easy’s aerospace library. The module was modified because singularities arose during some trajectory simulations when the built-in module was used. The problem was traced to the calculation of Euler angle orientation from quaternions. Only quaternions having unit magnitude describe orientations. In some cases, the quaternions did not have unit magnitude because of numerical integration errors giving rise to singularities when
computing Euler angles. In the modified RB module, the quaternions were normalized before the Euler angles were computed.

**Fig. 5.2 Schematic of the 3-DOF ice trajectory simulation model in MSC.Easy5.**

The “Ice Piece Geometry” module is a user written code developed to compute the ice particle mass and moments of inertia and supply this information to the “Force and Moment Calculator” sub-model and the RB module. The RB module computed the translational and angular accelerations acting on the ice particle in body and inertial reference frames and integrated the accelerations to obtain translational and angular velocities in both reference frames. The orientation of the body reference frame with respect to the inertial reference frame was represented by Euler angles and quaternions within this module. The forces and moments acting on the ice particle in the body reference frame formed the input to the RB module. The translational velocities, computed in this module, in the inertial reference frame were sent to the ‘Position’
module where they were integrated to obtain the location of the ice particle in the inertial coordinate system. The position of the ice particle was used by the ‘Flowfield Reader’ module to determine the flowfield velocity at the position of the ice particle. The ‘Flowfield Reader’ module was developed to read CFD flowfield data and interpolate the flowfield velocity. The axis direction in the inertial coordinate system and the CFD grid system were different, so conversions between the two coordinate systems were performed. The ‘Coordinate System Position Converter’ module converted the location of the ice particle from the inertial axis system to the CFD grid system so that the ‘Flowfield Reader’ module could interpolate the flowfield velocity at the ice particle location. The interpolated velocity in the CFD grid system was mapped by the ‘Coordinate System Velocity Converter’ to the inertial coordinate system. The translational velocity of the ice particle computed by the RB module and the flowfield velocity at the ice particle location computed by the ‘Flowfield Reader’ module were sent to the ‘Relative Velocity Calculator’ module to calculate the relative velocity of the ice particle with respect to the flow. From the relative velocity vector, the angle of attack of the ice particle was computed by the ‘Angle of Attack Calculator’ module. The ‘Relative Velocity Calculator’ and the ‘Angle of Attack Calculator’ are user-written FORTRAN codes. The angle of attack was used by the ‘Force and Moment Calculator’ sub-model to determine the forces and moments that acted on the ice particle. The sub-model contained table look-up functions and a user-written ‘Forces and Moments’ module as shown in Fig. 5.3. Tables containing coefficients of normal force, axial force and pitching moment of the ice particle were entered into ‘Table look-up’ modules which were capable of interpolating tabular data. In the 4-DOF model, an extra table lookup function to
determine side force on the ice particle was present. The ‘Force and Moment’ module was a user-written code that computed the actual force and moments from coefficient values. The RB module used the body forces and moments to compute the trajectory of the ice particle. The ‘Termination Code’ module terminated the trajectory calculation when the ice piece reached a user specified downstream location. The ‘Units’ module allows unit system (SI or U.S – custom) specification and unit conversions. This module is required in MSC.Easy5 when building models using its aerospace library.

![Fig. 5.3 Expansion of the Force and Moment Calculator sub-model.](image)

### 5.2.2 6-DOF Trajectory Simulation Model

The schematic of the MSC.Easy5 model developed to simulate 6-DOF ice particle trajectories is shown in Fig. 5.4. It is similar to the 4-DOF model described in Section 5.2.2, however, all six aerodynamic forces and moments are non-zero. In the 6-DOF trajectory simulation model, the aerodynamic forces and moments depend on the angle of attack ($\alpha$) and angle of sideslip ($\beta$). The ‘Alpha and Beta’ module determined the $\alpha$ and $\beta$ values from the relative velocity of the ice particle. The ‘Aerodynamic Data
Interpolation’ is a user-written module that interpolated the aerodynamic force and moment coefficients. The ‘Force and Moment Calculator’ module computed the forces and moments from the aerodynamic coefficients.

Fig. 5.4 Schematic of the 6-DOF ice trajectory simulation model in MSC.Easy5.

5.2.3 Monte Carlo Simulations

To perform the Monte Carlo simulations, an executable file was built that could run on MSC.Easy5’s Matrix Algebra Tool also known as MAT. MAT is an interactive tool in MSC.Easy5 (Fig. 5.5), which is mostly used for performing numerical calculations and process simulation results (Ref. 36). The Matrix Algebra Tool has a random number generator and the capability to run MSC.Easy5 executable models multiple times in a loop, making it possible to calculate the large number of ice particle trajectories required
for Monte Carlo simulations. A function script “mc”, provided in Appendix C, was written to perform the Monte Carlo simulations.

![Matrix Algebra Tool window](image)

*Fig. 5.5 Matrix Algebra Tool window.*
CHAPTER 6

RESULTS AND DISCUSSIONS

Three, four and six degree of freedom trajectories of ice particles were simulated using the WSU ice particle trajectory code and MSC.Easy5. Monte Carlo simulations were also performed to determine the regions where the ice particle was most likely to strike. The results presented in Sections 6.1 to 6.3 were obtained from both the WSU ice particle code and the MSC.Easy5 models unless specified otherwise.

6.1 3-DOF Ice Particle Trajectories

Three degree of freedom trajectories of ice particles and Monte Carlo simulations were performed for ice particles shed in 2D uniform and non-uniform flowfields. The ice particle was assumed to shed from the leading edge of a wing as shown in Fig. 6.1. The trajectory of the ice particle was simulated until it reached the engine inlet plane which was located approximately 15 ft downstream of the leading edge of the wing. This distance was used to approximate the distance between the engine inlet and the wing leading edge of a generic regional/business jet configuration with aft-mounted engines. The engine was assumed to be 2.5 ft in diameter and the center of the engine was 3.5 ft above the airfoil leading edge as shown in Fig. 6.1. Prior to shedding, the ice shape was located 0.3 ft ahead and 0.4 ft above the leading edge of the airfoil as shown in Fig. 6.2. The 3-DOF trajectory simulations were performed assuming a pressure altitude of 15,000 ft MSL and airspeed of 220 knots, which is a typical flight condition for a business or transport jet in a holding pattern.
6.1.1 Ice Particle Trajectories in 2D Uniform Flowfield

The trajectories of square plate, rectangular plate and semi-circular ice shapes shed in a uniform flowfield were determined. The effect of aerodynamic damping on the trajectory of the square plate was also investigated.

![Diagram](image)

**Fig. 6.1** Dimensions for 3-DOF ice particle trajectory simulations.

![Diagram](image)

**Fig. 6.2** Ice piece shed near the airfoil leading edge in 3-DOF simulations.

6.1.1.1 Trajectories of Square Plate, Rectangular Plate and Semi-Circular Shell

The trajectories of the square plate, rectangular plate and semi-circular shell are shown in Fig. 6.3. Prior to shedding, the ice particles were oriented at an angle of 145.63° with the x-axis, that is δ in Fig. 6.2 had a value of 145.63°. Figures 6.3a and 6.3b show
the trajectory of the ice particle for cases without and with rotational aerodynamic damping respectively. A rotational damping value ($C_{damp}$) of 0.0166 lb-ft-sec about the rotation axis was used to compute the trajectories presented in Fig. 6.3b.

![Fig. 6.3](image)

**Fig. 6.3 3-DOF trajectories of different ice shapes in 2D uniform flowfields.**

Figure 6.3a shows that the trajectories simulated without aerodynamic damping strike the airfoil. The trajectory calculations were not terminated after the ice particles struck the airfoil in Fig 6.3a. In Fig 6.3b, the square plate had a vertical displacement, that is y-direction displacement of approximately 2.0 ft at the engine inlet location while the rectangular plate was displaced by about -0.9 ft at the engine inlet location. Trajectories for the semi-circular shell with aerodynamic damping were not performed because estimates of aerodynamic damping coefficient for this geometry were not available.

Figure 6.3 shows that the trajectories of the ice particles are dependant on rotational aerodynamic damping values. The results presented in Section 6.1.1.2
demonstrates the effect of aerodynamic damping on key parameters affecting the trajectory.

6.1.1.2 Effect of Aerodynamic Damping on Trajectories of Square Plate

Trajectories of the square plate simulated with and without rotational aerodynamic damping are compared in Fig. 6.4. Figure 6.5 shows the variation of ice particle parameters as a function of location downstream of the release point. The parameters depicted in Fig. 6.5 include y-direction displacement, Cl and Cd of the ice particle, ice particle orientation angle (δ) with the x-axis and angle of attack (α) with the relative velocity vector, magnitude of the relative velocity vector and orientation angle rate, accelerations in the x and y directions of the reference frame. These data are provided to elucidate the trajectory behavior of the ice fragment. The graph relating AOA to x distance in Fig. 6.5 shows that the ice particle AOA varied between -90° and 90°. The change in the AOA sign in this graph indicates that the plate experienced small amplitude oscillations (rotations) about its pitch axis.

Fig. 6.4 3-DOF trajectories of square plate in uniform flowfields for cases with aerodynamic damping (Cdamp = 0.0166 lb-ft-sec) and without aerodynamic damping.
Fig. 6.5 Comparison of parameters that affect square plate trajectory calculations for simulations with aerodynamic damping ($C_{\text{damp}} = 0.0166$ lb-ft-sec) and without aerodynamic damping ($C_{\text{damp}} = 0$ lb-ft-sec).

Comparing the variation in ice particle orientation ($\delta$) and angle of attack ($\alpha$) for square plate trajectories with and without aerodynamic damping in Fig 6.5, it is seen that the presence of aerodynamic damping reduces the rotation of the ice particle. For the
square plate trajectory without aerodynamic damping the ice particle oscillated more freely and at certain times during its trajectory the lift force acted toward the airfoil (negative y-acceleration) resulting in the square plate striking the airfoil. When rotational aerodynamic damping was present, the ice particle after some small amplitude oscillations oriented itself at an angle of attack of $90^\circ$ after which it was mostly driven by drag. Once the square plate oriented itself at an angle of attack of $90^\circ$ its pitching moment was nearly zero as shown in Fig. 3.34 and small perturbations in the forces experienced by the plate resulted in small amplitude oscillations about its pitch axis.

6.1.2 Ice Particle Trajectories in 2D Airfoil Flowfield

Trajectories of the 12" × 6" rectangular plate and 5.13" × 5.13" square plate were simulated in 2D airfoil flowfields and compared to the trajectories computed using a uniform flowfield.

6.1.2.1 Trajectories of the Square and Rectangular Plates in Airfoil Flowfield

The trajectories of the square and rectangular plates shed in a 0° AOA NACA 23012 airfoil flowfield is illustrated in Fig. 6.6. The trajectories of the ice particle computed without and with aerodynamic damping are presented in Figs. 6.6a and 6.6b respectively. Trajectories of the ice particles shed in a uniform flowfield are also provided in these figures for comparison.

The vertical displacement (y-displacement) of the ice particles at the engine inlet plane was greater in the airfoil flowfield than in the uniform flowfield. The difference in trajectories caused by the airfoil and uniform flowfields is due to the difference in velocities which resulted in different forces and moments experienced by the ice particle.
In Fig. 6.6a the trajectories of the ice particles simulated in the airfoil flowfield did not strike the airfoil unlike the trajectories obtained with the uniform flowfield. In Fig. 6.6b, the vertical displacement of the rectangular plate at the engine inlet plane is positive when shed in an airfoil flowfield and negative when shed in a uniform flowfield. The charts in Fig. 6.6 show that the flowfield into which the ice particle sheds influences its trajectory as expected. The effect of different flowfields on the trajectory of square plate is described in Section 6.1.2.2.

![Graph showing 3-DOF trajectories of different ice shapes in 2D, 0° AOA NACA 23012 airfoil flowfield and 2D uniform flowfield.](image)

**Fig. 6.6** 3-DOF trajectories of different ice shapes in 2D, 0° AOA NACA 23012 airfoil flowfield and 2D uniform flowfield.

### 6.1.2.2 Effect of Flowfield on the Trajectory of a Square Plate

Trajectories for a square plate shed from clean and iced NACA 23012 airfoils at 0° and 4° AOA are depicted in Figs. 6.7a and 6.7b for the cases without and with aerodynamic damping respectively. The results presented in these figures indicate the following:
Effect of airfoil angle of attack: The square plate’s vertical displacement increased as the airfoil angle of attack was increased from 0° to 4° for the clean and iced airfoil cases with and without aerodynamic damping.

Effect of ice contamination: For an AOA of 4° airfoil flowfields, the differences in the trajectories of the ice fragment released in the clean and iced airfoil flowfields were small for the cases with and without aerodynamic damping. For the case without aerodynamic damping and for 0° AOA, the trajectory of the ice fragment released in the clean airfoil flowfield underwent more vertical displacement than that of the ice fragment released in the flowfield of the iced airfoil. With the addition of aerodynamic damping the trajectories for an AOA of 0° were similar for both the clean and iced airfoil cases.
• **Effect of ice fragment aerodynamic damping in pitch**: The addition of aerodynamic damping to the rotation of the ice shape about its pitch axis resulted in higher vertical displacements downstream of the release location for all cases tested.

### 6.1.3 Monte Carlo Simulations

Monte Carlo simulations of 12" × 6" rectangular plate and 5.13" × 5.13" square plate trajectories in a uniform flowfield were conducted to determine probability maps of particle trajectory paths. Monte Carlo simulations were conducted with square and rectangular plates using a NACA 23012 airfoil at 0° angle of attack. Monte Carlo simulations were also conducted with a square ice particle (plate) released in clean and iced airfoil flowfields. These flowfields were computed for angles of attack of 0° and 4°.

#### 6.1.3.1 2D Uniform Flowfield

Monte Carlo simulations of rectangular and square plate trajectories shed in uniform flowfield were performed by computing 60,000 ice fragment trajectories for randomly varying input parameters. Table 3 lists the parameters that were varied for the Monte Carlo analysis performed with square plate trajectories. The coefficients of lift and drag were varied between -10% and +10% of their nominal values. The nominal values of $C_L$ and $C_D$ were computed from equations 3.58 and 3.59. The length of the plate was varied from 0.2 ft to 0.5 ft while its thickness was bounded between 0.08 ft and 0.16 ft. The initial orientation of the square plate ranged from 0° to 90°. Table 4 lists the parameters that were varied when conducting Monte Carlo simulations for the 12" × 6" rectangular plate. The plate length was varied from 0.7 ft to 0.13 ft while its breadth was...
bound between 0.4 ft and 0.6 ft. Its thickness ranged from 0.02 ft to 0.03 ft and its initial orientation varied from 0° to 90°. The $C_L$ and $C_D$ values for the plate were not varied since they were obtained from wind tunnel tests. For Monte Carlo simulations conducted with rotational aerodynamic damping of 0.0166, the $C_{damp}$ value was varied according to equation 6.1 (Ref. 15) with the length of the ice particle ($L$).

$$C_{damp} = \frac{0.0166}{L_o^2} L^2$$

(6.1)

where $L_o$ is the original length of the ice particle.

$L_o = 0.428$ ft for the square plate and 1 ft for the rectangular plate.

**Table 3 Parameters varied in Monte Carlo simulation of square plate trajectories**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ice particle CL</td>
<td>-10% to +10% of nominal value given by equation 3.58</td>
</tr>
<tr>
<td>Ice particle CD</td>
<td>-10% to +10% of nominal value given by equation 3.59</td>
</tr>
<tr>
<td>Length of ice particle</td>
<td>0.2 ft to 0.5 ft</td>
</tr>
<tr>
<td>Thickness of ice particle</td>
<td>0.08 ft to 0.16 ft</td>
</tr>
<tr>
<td>Initial orientation of ice particle ($\delta$)</td>
<td>0° to 90°</td>
</tr>
</tbody>
</table>

**Table 4 Parameters varied in Monte Carlo simulation of rectangular plate trajectories**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of the ice particle</td>
<td>0.7 ft to 1.3 ft</td>
</tr>
<tr>
<td>Breadth of the ice particle</td>
<td>0.4 ft to 0.6 ft</td>
</tr>
<tr>
<td>Thickness of the ice particle</td>
<td>0.02 ft to 0.03 ft</td>
</tr>
<tr>
<td>Initial orientation of the ice particle ($\delta$)</td>
<td>0° to 90°</td>
</tr>
</tbody>
</table>

Figures 6.8a and 6.8b show the probability map of the ice particle trajectories at the engine inlet plane for cases without and with aerodynamic damping respectively. These probability distributions can be used to determine the likelihood of an ice particle crossing the engine inlet boundary. The results from these figures indicate the following:
• **Effect of aerodynamic damping:** The y-coordinate at the engine inlet plane where the ice particle was most likely to strike was more positive for cases where aerodynamic damping was present than for those where aerodynamic damping was absent.

• **Effect of ice particle shape:** The rectangular plate trajectories had a higher probability of having greater y-displacements at the engine inlet plane than square plate trajectories.

![Graphs showing probability maps for ice particle trajectories with and without aerodynamic damping.](image)

**Fig. 6.8** Probability map of 3-DOF ice particle trajectories shed in 2D uniform flowfield.

### 6.1.3.2 2D Airfoil Flowfield

Monte Carlo simulations were conducted for rectangular and square plates shed near a clean NACA 23012 airfoil at 0° AOA. Monte Carlo simulations were also conducted for the square plate shed in the flowfields of clean and iced NACA 23012 airfoil at angles of attack of 0° and 4°.
6.1.3.2.1 Monte Carlo Simulations of Rectangular and Square Plates

Figures 6.9a and 6.9b depict the probability map of the square and rectangular plate trajectories at the engine inlet plane for cases without and with aerodynamic damping respectively. The results in Fig. 6.9 indicate that the rectangular plate was more likely to have a higher y-displacement at the engine inlet plane than the square plate for cases with and without aerodynamic damping. In Fig. 6.9a for the case without aerodynamic damping, the rectangular plate had a 32% chance of entering the engine inlet. In Fig 6.9b for the case with aerodynamic damping, the square plate had a 56% probability of entering the engine inlet.

Fig. 6.9 Probability distribution of 3-DOF ice particle trajectories shed in 2D, 0° AOA clean airfoil flowfield.
6.1.3.2.2 Monte Carlo Simulations of Square Plate in Clean and Iced Airfoil Flowfields

Monte Carlo simulations were performed with a square plate released in clean and iced airfoil flowfields. Eight Monte Carlo simulations were conducted as follows:

- Clean NACA 23012 airfoil – square plate trajectories computed with and without aerodynamic damping; 0° and 4° AOA (4 cases)
- NACA 23012 airfoil with 22.5-min ice shape – square plate with and without aerodynamic damping; 0° and 4° AOA (4 cases)

Probability distributions for the ice particle locations at the x-station corresponding to the engine inlet plane obtained from the eight Monte Carlo simulations are shown in Figs. 6.10 and 6.11. The blue histograms depict the probability distributions obtained with the clean and iced airfoil flowfields at 4° AOA while the red histograms represent corresponding probability distributions calculated for the 0° AOA case. The results presented in these figures indicate the following:

- **Effect of angle of attack:** The probability for higher vertical displacements was greater at an angle of attack of 4° than at an AOA of 0° for both the clean and iced airfoil cases. This was true for trajectory computations performed with and without aerodynamic damping.
- **Effect of ice contamination:** For cases with and without aerodynamic damping, the probability distributions of the fragment’s vertical displacement for the iced airfoil at 0° and 4° AOA had a wider vertical spread than the corresponding probability distributions obtained with the clean airfoil. The probability charts in Figs. 6.10 and 6.11 also show that in all cases the displacement along the y-axis at
the engine inlet plane where square plate was most likely to strike was greater (more positive) for the iced airfoil compared to that of the clean airfoil.

- **Effect of aerodynamic damping in trajectory calculations:** The ice particles were more likely to have a higher y-displacement at the engine inlet plane when aerodynamic damping was present than when aerodynamic damping was absent.

The initial angle of attack and weight of the square plate were found to be the key parameters affecting its y-displacement at the engine inlet station when simulations were performed using $4^\circ$ AOA clean and iced flowfields for cases with aerodynamic damping. For the blue histogram in Fig. 6.10a, it is seen that ice particles weighing less than 0.5 lb and having initial angles of attack less than $15^\circ$ resulted in trajectories with y-displacements greater than 6.5 ft at the engine inlet station. Also, trajectories having y-displacements less than 2.7 ft resulted from ice particles with initial angles of attack greater than $75^\circ$. For the blue histogram in Fig. 6.10b, trajectories having y-displacements greater than 7 ft resulted from ice particles having initial angles of attack lower than $10^\circ$ while trajectories having y-displacements lower than 3.3 ft resulted from ice particles having initial angles of attack greater than $80^\circ$. In Fig. 6.11, when aerodynamic damping was absent, no definite trend or correlation was observed between the y-displacement of the ice particle at the engine inlet plane and ice particle parameters.

### 6.2 4-DOF Ice Particle Trajectories

Four degree of freedom trajectories of the square and rectangular plates in uniform and non-uniform flowfields were determined and are discussed in Sections 6.2.1
and 6.2.2. Four degree of freedom Monte Carlo simulations were also computed, and are presented in Section 6.2.3.

Fig. 6.10  Probability map of 3-DOF square plate trajectory footprints in 2D airfoil flowfields; with aerodynamic damping, $C_{\text{damp}} = 0.0166 \text{ lb-ft-sec}$.

Fig. 6.11  Probability map of 3-DOF square plate trajectory footprints in 2D airfoil flowfields; without aerodynamic damping.
6.2.1 Ice Particle Trajectories in 3D Uniform Flowfield

In the computation of 4-DOF trajectories in uniform flowfield, the particles were shed from the fuselage of an airplane, as illustrated in Fig. 6.12. The path of the ice fragment was computed until it reached the engine inlet plane which was located approximately 45 ft downstream of the shedding location. For all trajectory simulations, the initial orientation ($\delta$) of the ice particle was assumed to be 0.6 radians (or 34.37°). The airplane depicted in Fig. 6.12 is for illustration purposes. The trajectory simulations were performed assuming a uniform flowfield without the aircraft.

![Diagram of ice particle trajectory](image)

**Fig. 6.12 Ice particle shed off the side of an aircraft fuselage.**

6.2.1.1 Trajectories of Square and Rectangular Plates

The simulated trajectories of the square and rectangular plates are illustrated in Figs 6.13 and 6.14 for cases without and with aerodynamic damping respectively. The trajectories of the rectangular plate in Figs. 6.13b and 6.14b had lower $z$-displacement than the square plate trajectories at the engine inlet plane. This was because the drag on the rectangular plate was greater than the drag on the square plate, thus it reached the engine inlet plane faster and was under the influence of gravity for a shorter period compared to the square plate.
For the trajectories simulated with aerodynamic damping in Fig 6.14a, the rectangular plate trajectory had negative y-displacement at the engine inlet plane while the square plate trajectory had a positive y-displacement. The presence of aerodynamic damping opposed the rotation of the ice shapes and resulted in plate angles of attack where the pitching moment was zero. The rectangular plate oriented itself at a -60° AOA where pitching moment was zero. At this orientation, the lift force on the plate was in the negative y-direction and resulted in negative y-displacements at the engine inlet plane. The square plate eventually oriented itself at an AOA of 90° at which point it was mostly driven by drag and as a result it had a positive y-displacement at the engine inlet plane.

6.2.1.2 Effect of Aerodynamic Data on Trajectories of the Rectangular Plate

Aerodynamic data of the square plate was given by empirical equations 3.58 to 3.61, while that of the rectangular plate were determined experimentally from wind tunnel tests. Since the square plate and the rectangular plate are similar in shape, the empirical aerodynamic data of the square plate was used to simulate rectangular plate trajectories to determine the impact of aerodynamic data on the path. Figures 6.15 and 6.16 show the difference between rectangular plate trajectories obtained using empirical aerodynamic data and experimental aerodynamic data, for cases without and with aerodynamic damping respectively.

For the case without aerodynamic damping shown in Fig. 6.15a, the rectangular plate trajectory obtained using the experimental aerodynamic data had a positive lateral displacement (y-displacement) at the engine inlet plane while that computed with the empirical aerodynamic coefficients had a negative lateral displacement.
Fig. 6.13 4-DOF trajectories of ice particles shed in 3D uniform flowfield without aerodynamic damping, $C_{damp} = 0.0 \text{ lb-ft-sec}$.

Fig. 6.14 4-DOF trajectories of ice particles shed in 3D uniform flowfield with aerodynamic damping, $C_{damp} = 0.0166 \text{ lb-ft-sec}$.
For the case with aerodynamic damping shown in Fig. 6.16a, the rectangular plate trajectory computed using experimental aerodynamic data had a negative y-displacement at x=45 ft corresponding to the engine inlet plane while the rectangular plate trajectory obtained using analytical aerodynamic data had a positive y-displacement. Simulation using analytical data showed the ice particle orienting itself perpendicular to the freestream, due to the aerodynamic damping opposing rotation, after which it was mostly driven by drag. The trajectory simulated from experimental data showed the ice piece orienting itself at around -60° angle of attack where the pitching moment was zero. At this orientation, the lift acted toward the airplane thereby causing it to move closer to the aircraft which led to negative y-displacements at the engine inlet plane.

The difference in trajectories simulated using experimental and empirical aerodynamic data in Figs 6.15 and 6.16 shows the need for a database containing experimentally determined aerodynamic forces and moment for different ice particles.

Fig. 6.15 Comparison of 4-DOF rectangular plate trajectories computed with experimental and empirical aerodynamic data; without aerodynamic damping.
6.2.2 Ice Particle Trajectories in 3D Wing Flowfield

4-DOF trajectories were determined for different ice particles shed from a clean wing modes installed in a wind tunnel and position at 0° AOA. In addition, 4-DOF trajectories were computed for a square plate shed in clean and iced wing flowfields with the wing at 0° AOA. The ice particles were shed from the 50% semispan location of the half wing as shown in Fig. 6.17. In addition, Fig. 6.17 shows the forces at the centroid of the ice fragment, which include the fragment weight and the lift and drag forces on the fragment. In the simulation, the rotation of the ice fragment was assumed to be about the vertical axis only. The initial orientation angle of the ice fragment was 0.6 radians (approximately 34.38°) with respect to the negative x-axis and the fragment centroid was placed approximately 0.22 ft ahead and 0.2 ft above the leading edge of the mid-semispan location prior to shedding, as illustrated in Fig. 6.18. The computations of the ice particle
trajectories were terminated at a distance of 9 ft downstream of the shedding location (i.e. the x-coordinate of the trajectory ranged from 1.0 ft to 10.0 ft).

6.2.2.1 Trajectories of Square and Rectangular Plates

The trajectories of the square plate and the rectangular plate were simulated in the flowfield of a clean GLC-305 wing. Figures 6.19 and 6.20 depict the trajectories of the ice particles for cases without and with aerodynamic damping respectively. The trajectories of ice particles with aerodynamic damping moved further away from the wing than those without aerodynamic damping, as shown in Figs. 6.19a and 6.20a. The absence of aerodynamic damping caused the ice particle to oscillate during parts of the trajectory path with the lift force acting towards the wing. The rectangular plate had higher drag force than the square plate which caused it to cover the 10 ft distance from the shed location in a shorter time. The vertical distance covered by the square plate was less than that travelled by the rectangular plate.
Fig. 6.19 4-DOF trajectories of ice shapes shed in clean GLC-305 swept wing flowfield without aerodynamic damping.

Fig. 6.20 4-DOF trajectories of ice shapes shed in clean GLC-305 swept wing flowfield with aerodynamic damping, $C_{damp} = 0.0166$ lb-ft-sec.
6.2.2.2 Effect of Clean and Iced Wing Flowfield on Trajectories of the Square Plate

Trajectories of the square plate were simulated in clean and iced wing flowfields positioned at 0° AOA. Figures 6.21 and 6.22 show the behavior of different ice fragment parameters as a function of downstream distance from the point of shedding (x = 1.0 ft). Parameters include ice particle lateral (y-direction) and vertical (z-direction) displacements, lift coefficient (Cl), drag coefficient (Cd), ice particle orientation angle (δ) with respect to the negative x-axis of the inertial reference frame, angle of attack (α) with respect to the relative velocity vector, magnitude of relative velocity, orientation angle rate, and accelerations in x and y directions of the inertial reference frame. The results presented in these figures indicate the following:

- **Effect of iced wing flowfield:** Square plate trajectories simulated in the iced wing flowfield had greater lateral displacements (more negative y-displacements) downstream of the shedding location than corresponding trajectories obtained with the clean wing flowfield. This was due to the greater relative velocity of the ice fragment in the initial stages of the trajectory in the iced wing flowfield. In addition, the square plate trajectories in the iced wing flowfield were observed to have smaller vertical (z-direction) displacements than trajectories simulated in the clean wing flowfield. In the iced wing flowfield, the square plate had a higher relative velocity immediately after shedding which caused it to traverse the 9-ft distance in a shorter time and as a result the vertical displacement was also less.

- **Effect of ice fragment aerodynamic damping:** Square plate trajectories simulated in the clean and iced wing flowfields showed that the addition of aerodynamic damping resulted in trajectories moving away from the wing surface.
Aerodynamic damping opposed the rotation of the ice particle and caused it to orient itself perpendicular to the freestream after which it was mostly driven by drag. Conversely, in the absence of aerodynamic damping, the square plate was able to rotate freely and experienced negative angles of attack at certain portions of its trajectories during which the $y$-component of acceleration acted towards the wing, thereby, reducing the lateral displacement of the trajectories. In addition, square plate trajectories simulated with aerodynamic damping had smaller vertical displacements ($z$-displacements) than those simulated without aerodynamic damping.

6.2.3 Monte Carlo Simulations in Uniform Flowfield

Monte Carlo trajectory simulations were performed for cases with and without aerodynamic damping for the square and rectangular plates. The ice particles were assumed to shed from the fuselage of the airplane as shown in Fig. 6.12. Probability maps were computed at the engine inlet plane which depicted the regions the ice particles were most likely to strike.

6.2.3.1 Square Plate

Table 3 in Section 6.1.3.1 lists the input parameters that were varied for the 4-DOF Monte Carlo simulations of square plate trajectories. The length of the square plate was varied from 0.2 ft to 0.5 ft while its thickness was bound between 0.08 ft and 0.16 ft. The initial orientation of the square plate ranged from $0^\circ$ to $90^\circ$. The aerodynamic lift and drag coefficients were varied between -10% and 10% of their nominal values.
Fig. 6.21 Comparison of parameters that affect square plate trajectory calculations in 3D clean and iced wing flowfields; without aerodynamic damping.
Fig. 6.22 Comparison of parameters that affect square plate trajectory calculations in 3D clean and iced wing flowfields; with aerodynamic damping, $C_{damp} = 0.0166$ lb-ft-sec.
For the case without aerodynamic damping, shown in Fig. 6.23a, the lateral (y-direction) displacement of the square plate varied from -3.2 ft to 4.5 ft and the vertical displacement (z-direction) varied from 2 ft to 10.4 ft. The maximum concentration of the ice particle footprints was in a band between 2.4 ft and 4.2 ft in the vertical direction having no lateral displacement (Fig. 6.24a). When aerodynamic damping was present, the square plate vertical displacement ranged from 2 ft to 7.4 ft and the lateral displacement ranged from -0.2 ft to 8 ft, as shown in Fig. 6.23b. Figure 6.24b indicates that the ice particle had the maximum probability of having a vertical displacement between 2.4 to 4.0 ft and no lateral displacement at the engine inlet plane.

![Fig. 6.23 4-DOF square plate trajectory footprint map at the engine inlet location.](image)

### 6.2.3.2 Rectangular Plate

Table 4 in Section 6.1.3.1 lists the input parameters that were varied for the 4-DOF Monte Carlo simulations performed with the rectangular plate. The rectangular plate length was varied from 0.7 ft to 0.13 ft while its breadth was bound between 0.4 ft
and 0.6 ft. The thickness of the ice particle ranged from 0.02 ft to 0.03 ft. The initial orientation of the ice particle varied from 0° to 90°.

For the case without aerodynamic damping in Fig. 6.25, the y-displacement of the rectangular plate trajectories varied from -4 ft to 15 ft while the z-displacement varied from 1 ft to 2.4 ft. The trajectories had a maximum probability of passing through a region situated 14 ft to 15 ft in the y-directions and extending 1 ft to 1.2 ft in the z-direction (Fig 6.26).

When aerodynamic damping was present, Fig. 6.25b shows that the trajectories of the rectangular plate had y-displacements varied from -8.8 ft to 15.5 ft while the z-displacements varied between 1 ft and 2 ft. At the engine inlet plane, the rectangular plate was most likely to have y-displacements between 14 ft and 15.2 ft with z-displacements ranging from 1 ft to 1.4 ft, as depicted in Fig. 6.27.

Comparing the footprint map of the square and the rectangular plates at the engine inlet plane in Figs. 6.23 and 6.25, it is seen that the rectangular plate had a lower z-
displacement than the square plate. This was because higher drag forces acted on the rectangular plate than on the square plate making it reach the engine inlet plane in a shorter time. The shorter travel time resulted in smaller vertical displacements.

![Fig. 6.25 4-DOF rectangular plate trajectory footprint map at the engine inlet location.]

The trajectory of the rectangular plate had a high probability of having a y-displacements of around 15 ft at the engine inlet location (Figs. 6.26 and 6.27) while the square plate’s trajectory was most likely to have a y-displacement of 0 ft at the engine inlet plane (Fig. 6.25). The ice particles usually oriented themselves at an AOA at which the pitching moment was zero. The pitching moment of the rectangular plate was zero at 60° AOA while the square plate had zero pitching moment at 90° AOA. For the rectangular plate at 60° AOA the lift force acted on it in the positive y-direction hence its trajectory had a positive y-displacement at the engine inlet plane. The flat plate at 90° AOA was mostly driven by drag and its trajectories did not have a large y-displacement at the engine inlet plane.
Fig. 6.26 Probability distributions of 4-DOF rectangular plate trajectory footprints at the engine inlet plane for the case without aerodynamic damping.

Fig. 6.27 Probability distributions of 4-DOF rectangular plate trajectory footprints at the engine inlet plane for the case with aerodynamic damping, $C_{\text{damp}} = 0.0166$ ft-lb-sec.
6.3 6-DOF Ice Particle Trajectories

Six degree of freedom trajectories of the rectangular plate shed in uniform and non-uniform flowfields were simulated using the WSU trajectory code and MSC.Easy5. The trajectories in the uniform flowfield were simulated assuming the ice particle was shed from the fuselage of an airplane as depicted in Fig. 6.12. The trajectories were simulated until the ice particle reached the engine inlet plane which was 45 ft downstream of the shedding location. The trajectories simulated in the wing flowfield were obtained assuming the ice particle sheds from 50% semispan of the wing as depicted in Fig. 6.15 and the trajectories were simulated until the ice particle was 9 ft downstream of the shedding location ($x = 1$ ft).

6.3.1 Rectangular Plate Trajectories in 3D Uniform Flowfield

The trajectories of the rectangular plate released in a uniform flowfield are depicted in Fig. 6.28. These have also been compared to the 4-DOF trajectories of the rectangular plate in Fig. 6.28. In the 6-DOF simulations, the ice particle rotation was not restricted to one axis as in the 4-DOF simulations resulting in differing trajectories for the two simulations.

6.3.2 Rectangular Plate Trajectories in Clean and Iced Wing Flowfield

Trajectories for a rectangular plate shed from a clean and iced GLC-305 wing set at 0° AOA were computed with the WSU code and are presented in Fig. 6.29. The trajectories of the ice particle released in the clean and iced wing flowfields varied considerably. In the clean wing flowfield, the initial angle of attack of the ice particle was 37° and the ice particle rotated, sometimes resulting in the lift force acting towards the
wing. Conversely, the rectangular plate in the iced wing flowfield had an initial AOA of 50° and it did not rotate much since the pitching moment was to zero at 60° AOA. At this angle of attack, the lift force acted away from the wing and as a result the rectangular plate in the iced wing flowfield had more negative y-displacements compared to its trajectory in the clean wing flowfield.

![Graph showing variations in y-displacement, z-displacement, yaw angle (ψ), pitch angle (θ), and roll angle (φ) with x-displacement of 6-DOF rectangular plate trajectory in 3D uniform flowfield.](image)

**Fig. 6.28** Variation of y-displacement, z-displacement, yaw angle (ψ), pitch angle (θ) and roll angle (φ) with x-displacement of 6-DOF rectangular plate trajectory in 3D uniform flowfield.
Fig. 6.29 Variation of y-displacement, z-displacement, yaw angle (ψ), pitch angle (θ) and roll angle (φ) with x-displacement of 6-DOF rectangular plate trajectory in 3D, clean and iced, 0° AOA GLC-305 wing flowfield.
CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

A methodology to simulate 3-, 4- and 6-DOF trajectories of ice particles shed in uniform and non-uniform flowfields was developed. The 6-DOF model presented in this thesis was the first attempt to predict ice particle trajectories taking into account the effect of all six aerodynamic forces and moments. This trajectory computation effort was done in combination with wind tunnel tests (Ref. 14) which determined the aerodynamic characteristics of different ice particles. Computer codes were written in FORTRAN and MSC.Easy5 to determine ice particle paths.

Monte Carlo simulations of ice particle trajectories were performed to model the random nature of ice shedding, and ‘hot spots’ or locations on the engine inlet plane where the ice particle was most likely to strike were determined.

Interpolation routines were also developed to interpolate velocities from CFD data, which were used when calculating ice particle trajectories in non-uniform flowfields.

The 3-DOF model was used to compute trajectories and perform Monte Carlo simulations in uniform and non-uniform 2D flowfields. Monte Carlo simulations of 4-DOF ice particle trajectories were performed in 3D uniform flowfields. Using the 4-DOF and 6-DOF trajectory simulation models, ice particle paths were simulated in uniform and non-uniform 3D flowfields. Below is a summary of the conclusions drawn from the trajectory and Monte Carlo simulations:

- Factors that affect ice particle trajectories: The flowfield into which the ice particle was shed, the rotational aerodynamic damping coefficient and the initial
orientation of the ice particle were found to be the factors that had a significant influence on the trajectories.

- **Effect of damping on ice particle trajectories**: Aerodynamic damping reduced the tendency of the particle to rotate about its pitch axis and resulted in trajectories that were different from trajectories computed with no aerodynamic damping.

- **Effect of aerodynamic data on ice particle trajectories**: Four degree of freedom rectangular plate trajectories simulated using empirical aerodynamic data were different from the trajectories simulated using experimental aerodynamic data. This highlighted the need for experimental aerodynamic coefficients for a range of ice fragments.

- **Comparison of 4- and 6-DOF trajectories**: Four and six degree of freedom trajectories of the rectangular plate simulated in uniform flowfield were found to differ from each other. The 6-DOF trajectory of the 12" × 6" rectangular plate showed that the ice particle experienced rotation about all three axes.

- **Effect of flowfield on 6-DOF trajectories**: Six degree of freedom trajectory of the 12" × 6" rectangular plate simulated in a 3D clean wing flowfield was very different from a 6-DOF trajectory computed in a 3D iced wing flowfield. This showed that the flowfield into which the ice particle sheds had a significant effect on the ice trajectory path.

- **Monte Carlo simulations with 3-DOF trajectory code**: Monte Carlo simulations with square and rectangular plates shed in uniform and non-uniform flowfields were conducted. At a station 15 ft downstream of the shedding location, probability maps were determined depicting regions that the ice particle was most
likely to pass through. The region having high probability of ice impact was found to depend on flowfield, shape of the ice particle, and magnitude of aerodynamic damping.

- **Monte Carlo simulations with 4-DOF trajectory code:** Monte Carlo simulations were conducted with square and rectangular plates shed in a 3D uniform flowfield. The probability maps computed at a plane located 45 ft downstream of the shedding point showed that the region most likely to be struck by the ice fragment was considerably different for the square and rectangular plates used in the simulations.

**Recommendations for further work**

Extensive work is needed to elucidate the factors affecting the trajectories of shed ice particles. Recommendations to improve the trajectory simulation methodology presented in this thesis are proposed below:

- The ice particle trajectory simulation methodology needs to be validated through experiments. The values of rotational aerodynamic damping also need to be determined experimentally.

- Static aerodynamic forces and moments were used to compute the trajectories of ice particles. Future work should consider the effect of ice particle rotation on the aerodynamic forces and moments.

- The flowfield velocity on the ice particle is assumed to be equal to the flowfield velocity at the ice particle’s centroid. In reality different locations on the ice particle experience different flowfield velocities. In future, trajectory computations can include the effects of flowfield velocities at different locations.
on the ice particle since it influences the aerodynamic forces experienced by the ice particle.

- For Monte Carlo simulations, experimentally obtained probability distribution of the initial conditions can be used to improve the simulation results. In this thesis, a uniform probability distribution was used to vary the values of the initial parameters between their limits. A Gaussian or some other representative probability distribution can be used when performing Monte Carlo simulations.

- The trajectory code can be improved to conduct Monte Carlo simulations in 3D non-uniform flowfields. This has not been performed in this thesis because of the large amount of time it would take to simulate 60,000 trajectories. A single 4- or 6-DOF trajectory computation in non-uniform flowfields takes about 3.5 minutes on a 2.2GHz, 1GB RAM Pentium 4 processor. This is because the program spends a lot of time searching for the CFD element in which the centroid of the ice particle is located. A more efficient method needs to be developed to search through CFD data.
LIST OF REFERENCES


APPENDICES
APPENDIX A

INERTIAL AND BODY AXIS SYSTEMS

The inertial axis system or inertial reference frame is a fixed and non-rotating axis system in which position and orientation of a moving and rotating object can be tracked. The body axis system is an axis system that is fixed to the moving object and rotates along with the object. When ice sheds its location with respect to the airplane needs to be known, hence, in the ice particle trajectory simulation model, the inertial axis system \((x^i,y^i,z^i)\) is fixed to the airplane and the body axis system \((x^b,y^b,z^b)\) is fixed to the shed ice particle as shown in Fig. A1. The inertial axis system is centered on the left wing of the airplane in Fig. A1 but can be located at any location on the airplane convenient for trajectory simulations. The center of the body axis system was taken to be at the centroid of the ice particle.

Fig. A1  Inertial \((x^i,y^i,z^i)\) and body \((x^b,y^b,z^b)\) axis systems for the ice shedding event.

The orientation of the ice particle with the inertial axis system in the trajectory simulation model was described using different methods: In the 4-DOF trajectory simulation model, only one angle \((\delta)\) was used to keep track of the ice particle’s
orientation because the ice particle rotated about only one axis, and in the 6-DOF trajectory simulation model, Euler angles and quaternions were used to describe the orientation of the ice particle because the ice particle rotated about all three axes.
Euler angles and quaternions are methods used to describe the orientation of one axis system with another axis system. In the ice particle trajectory code, they are used to describe the orientation of the body with respect to the inertial or airplane reference frame.

**B.1 Euler Angles**

The most popular method to describe orientations because of simplicity and ease of visualization is the Euler angle method. The Euler angles specify three successive rotations to bring the inertial coordinate system in alignment to the body coordinate system. There are twelve possible ways to define the rotations and the order of rotation determines the final orientation. In aerospace applications the inertial axis system is first rotated about the z-axis by a yaw angle (ψ), then about the new y-axis by a pitch angle (θ) and finally about the new x-axis by a roll angle (φ) to bring it to align with the body axis system as shown in Fig. B1. The yaw angle, pitch and roll angles range from ±180°, ±90° and ±180° degrees respectively.

![Fig. B1 Euler angle rotations.](image-url)
A vector $\mathbf{V}^b$ in the body axis system can be transformed to inertial axis system $\mathbf{V}^i$ using the transformation matrix $[M]$ in equation B1 and vice versa using equation B2. The transformation matrices $[M]$ and $[M]^{-1}$ are the transpose of each other and their components are determined from the Euler angle orientation of the body axis system with the inertial axis system.

$$\{\mathbf{V}^i\} = [M]\{\mathbf{V}^b\} \tag{B1}$$

where $\{\mathbf{V}^i\}$ is the vector in the inertial axis system

$\{\mathbf{V}^b\}$ is the vector in the body axis system

$[M]$ is the transformation matrix

$$[M] = \begin{bmatrix}
\cos(\theta) \cdot \cos(\psi) & \sin(\phi) \cdot \sin(\theta) \cdot \cos(\psi) - \cos(\phi) \cdot \sin(\psi) & \cos(\phi) \cdot \sin(\theta) \cdot \cos(\psi) + \sin(\phi) \cdot \sin(\psi) \\
\cos(\theta) \cdot \sin(\psi) & \sin(\phi) \cdot \sin(\theta) \cdot \sin(\psi) + \cos(\phi) \cdot \cos(\psi) & \cos(\phi) \cdot \sin(\theta) \cdot \sin(\psi) - \sin(\phi) \cdot \cos(\psi) \\
-\sin(\theta) & \sin(\phi) \cdot \cos(\theta) & \cos(\phi) \cdot \cos(\theta)
\end{bmatrix}$$

$$\{\mathbf{V}^b\} = [M]^{-1}\{\mathbf{V}^i\} \tag{B2}$$


If a body has an angular velocity $(P, Q, R)$ about its body axes, then the rate of change of Euler angles is given by equations B3 to B5 (Ref. 22).

$$\dot{\phi} = P + Q \cdot \sin(\phi) \cdot \tan(\theta) + R \cdot \cos(\phi) \cdot \tan(\theta) \tag{B3}$$

$$\dot{\theta} = Q \cdot \cos(\phi) - R \cdot \sin(\phi) \tag{B4}$$

$$\dot{\psi} = Q \cdot \sin(\phi) \cdot \sec(\theta) + R \cdot \cos(\phi) \cdot \sec(\theta) \tag{B5}$$

It is seen from equations B3 and B5 that the roll angle rate ($\dot{\phi}$) and yaw angle rate ($\dot{\psi}$) are undefined when $\theta = \pm 90^\circ$. This singularity is called a ‘Gimbal lock’ and is a disadvantage of using Euler angles to describe orientations when simulating trajectories.
of rotating bodies. Since an ice particle tumbles after shedding, quaternions were used to keep track of the ice particle’s orientation in the trajectory simulation model. The Euler angles were computed from the quaternions after the trajectory simulation because it is easier to visualize orientation of a body using Euler angles.

B.2 Quaternions

Algebraically a quaternion (q) consists of four components as shown in equation B6. The quaternion can be viewed as a scalar plus a three component vector. The magnitude of a quaternion is given by equation B7. Quaternions having unit magnitude, i.e. |q| =1, have the ability to represent orientations of a body with respect to an inertial axis system.

\[
q = q_1 + q_2 \hat{i} + q_3 \hat{j} + q_4 \hat{k}
\]  \hspace{1cm} (B6)

\[
|q| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}
\]  \hspace{1cm} (B7)

Euler’s rotation theorem states that orientation of any rigid body can be described by a rotation D about an axis u as shown in Fig. B2. If A, B and C are the angles the axis u makes with the axes of the inertial reference frame in Fig. B2 then the components of the quaternion that describe this orientation are given by equation B8 to B11.

![Fig. B2](image)

**Fig. B2** Orientation brought about by a rotation ‘D’ about an axis ‘u’.
The transformation matrix $[M]$ which is used to transform vectors in equation B1 from the body axis system to the inertial axis system can be determined from the quaternion representing the orientation using equation B12.

$$
[M] = \begin{bmatrix}
q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2 \cdot q_3 - q_1 \cdot q_4) & 2(q_1 \cdot q_3 + q_2 \cdot q_4) \\
2(q_2 \cdot q_3 + q_1 \cdot q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3 \cdot q_4 - q_1 \cdot q_2) \\
2(q_2 \cdot q_4 - q_1 \cdot q_3) & 2(q_3 \cdot q_4 + q_1 \cdot q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2
\end{bmatrix}
$$

(B12)

For a tumbling body, the quaternion rates $(\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4)$ can be computed from the angular rates $P$, $Q$ and $R$ (about the body axes) using equations B13 to B16 (Ref. 35) without singularities occurring during calculation. Integrating the quaternion rates would result in the quaternion representing the new orientation but having a magnitude greater than unity. To preserve the unit magnitude of the quaternion, the integration drift correction gain $(\lambda)$ is used in equations B13 to B16.

$$
\dot{q}_1 = -\frac{1}{2}(q_2P + q_3Q + q_4R) + \lambda q_1
$$

(B13)

$$
\dot{q}_2 = \frac{1}{2}(q_1P + q_3R - q_4Q) + \lambda q_2
$$

(B14)
\[ \dot{q}_3 = \frac{1}{2}(q_1 Q + q_4 P - q_2 R) + \lambda q_3 \]  
\[ \dot{q}_4 = \frac{1}{2}(q_1 R + q_2 Q - q_3 P) + \lambda q_4 \]  

where \( \lambda \) is the integration drift correction gain given by \( \lambda = 1 - \left( q_1^2 + q_2^2 + q_3^2 + q_4^2 \right) \).

If the Euler angles of the orientation of the body axes in Fig. B2 is known, then the quaternion describing the orientation can be determined using equations B17-B20.

\[ q_1 = \cos\left(\frac{\phi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \]  
\[ q_2 = \cos\left(\frac{\phi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) \]  
\[ q_3 = -\cos\left(\frac{\phi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \]  
\[ q_4 = \sin\left(\frac{\phi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \]  

The Euler angles that describe an orientation can be computed from the quaternion that represents the orientation using equation B21-B23 (Ref. 35). Equations B21 to B23 can be derived by equating the transformation matrix \([M]\) obtained from quaternions in equation B12 to that obtained using Euler angles in equation B1.

\[ \psi = \cos^{-1}\left(\frac{q_1^2 + q_2^2 - q_3^2 - q_4^2}{\cos \theta}\right) \cdot \text{sign}[2(q_2 q_3 + q_4 q_1)] \]  
\[ \theta = \sin^{-1}\left[-2(q_2 q_4 - q_3 q_1)\right] \]  
\[ \phi = \cos^{-1}\left(\frac{q_1^2 - q_2^2 - q_3^2 + q_4^2}{\cos \theta}\right) \cdot \text{sign}[2(q_3 q_4 + q_1 q_2)] \]
APPENDIX C

MATRIX ALGEBRA TOOL SCRIPT FILE

The function script file ‘mc.ezemf’ is shown in the Table C1. The line numbers have been added to help explain the commands used in the function script and are not present in the actual file. The file is used to perform Monte Carlo simulation in MSC.Easy5. The Table C2 explains the commands used in function script file.

Table C1  MAT function script file

<table>
<thead>
<tr>
<th>Line#</th>
<th>Function Script File : mc.ezemf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>function z=mc(i)</code></td>
</tr>
<tr>
<td>2</td>
<td># Function to perform monte carlo simulations</td>
</tr>
<tr>
<td>3</td>
<td><code>pi=acos(-1);</code></td>
</tr>
<tr>
<td>4</td>
<td># Opening output file to write results</td>
</tr>
<tr>
<td>5</td>
<td><code>out=fopen('output-nodamp.csv','w');</code></td>
</tr>
<tr>
<td>6</td>
<td>#Headings for the columns in the output file</td>
</tr>
<tr>
<td>7</td>
<td><code>fprintf(out,'Iteration,Side,Thickness,Initial AOA,lt,dg,X,Y,Z \n');</code></td>
</tr>
<tr>
<td>8</td>
<td>#setting the time increments and max time of the simulation</td>
</tr>
<tr>
<td>9</td>
<td><code>setup_sim='TINC=0.0001, TMAX=5., INT MODE=8, PRINT CONTROL=0, PRINT2=0';</code></td>
</tr>
<tr>
<td>10</td>
<td>#Performing the Monte Carlo simulation</td>
</tr>
<tr>
<td>11</td>
<td>for k = 1:i</td>
</tr>
<tr>
<td>12</td>
<td>#Setting the random length, breadth, thickness and initial AOA of the ice particle</td>
</tr>
<tr>
<td>13</td>
<td><code>side = 0.2 + 0.3*rand(1);</code></td>
</tr>
<tr>
<td>14</td>
<td><code>thick = 0.08*(1 + rand(1));</code></td>
</tr>
<tr>
<td>15</td>
<td><code>breadth = side;</code></td>
</tr>
<tr>
<td>16</td>
<td><code>angle = pi/2*rand(1);</code></td>
</tr>
<tr>
<td>17</td>
<td><code>len = sprintf('PARAMETER VALUES, L_FOMA = %7.4f',side);</code></td>
</tr>
<tr>
<td>18</td>
<td><code>brd = sprintf('PARAMETER VALUES, B_FOMA = %7.4f',side);</code></td>
</tr>
<tr>
<td>19</td>
<td><code>thk = sprintf('PARAMETER VALUES, T_FOMA = %7.4f',thick);</code></td>
</tr>
<tr>
<td>20</td>
<td><code>ang = sprintf('PARAMETER VALUES, EUI_RB(1) = %8.5f',angle);</code></td>
</tr>
<tr>
<td>21</td>
<td>#Input values of length, breadth, thickness and initial AOA</td>
</tr>
<tr>
<td>22</td>
<td><code>r=MonteCarlo(len);</code></td>
</tr>
<tr>
<td>23</td>
<td><code>r=MonteCarlo(brd);</code></td>
</tr>
<tr>
<td>24</td>
<td><code>r=MonteCarlo(thk);</code></td>
</tr>
<tr>
<td>25</td>
<td><code>r=MonteCarlo(ang);</code></td>
</tr>
<tr>
<td>26</td>
<td>#Setting up the percentage change in lift and drag</td>
</tr>
<tr>
<td>27</td>
<td><code>lt = 0.1 - 0.2*rand(1);</code></td>
</tr>
<tr>
<td>28</td>
<td><code>lft = sprintf('PARAMETER VALUES, T1_FM = %8.5f',lt);</code></td>
</tr>
<tr>
<td>29</td>
<td><code>d = 0.1 - 0.2*rand(1);</code></td>
</tr>
<tr>
<td>30</td>
<td><code>drg = sprintf('PARAMETER VALUES, T2_FM = %8.5f',d);</code></td>
</tr>
<tr>
<td>31</td>
<td><code>r=MonteCarlo(lft);</code></td>
</tr>
<tr>
<td>32</td>
<td>#Setting the aerodynamic damping coefficient</td>
</tr>
<tr>
<td>33</td>
<td><code>cdamp = 0.0166;</code></td>
</tr>
<tr>
<td>34</td>
<td><code>damping = sprintf('PARAMETER VALUES, CDAMP = %6.4f',cdamp);</code></td>
</tr>
<tr>
<td>35</td>
<td><code>r=MonteCarlo(damping);</code></td>
</tr>
<tr>
<td>36</td>
<td>#Setting the x-location to end simulations</td>
</tr>
<tr>
<td>37</td>
<td><code>r=MonteCarlo('PARAMETER VALUE, XSTOP=45.0');</code></td>
</tr>
<tr>
<td>38</td>
<td>#Input values of simulation parameters</td>
</tr>
<tr>
<td>39</td>
<td><code>r=MonteCarlo(setup_sim);</code></td>
</tr>
</tbody>
</table>
```plaintext
# Input values of simulation parameters
r=MonteCarlo(setup_sim);

# Run the simulation
r=MonteCarlo('SIMULATE');

# Printing the iteration number
k

# Setting the values of X,Y,Z locations of the ice particle to variables
a=MonteCarlo('GET VALUE = X_pos2 ');
b=MonteCarlo('GET VALUE = X_pos ');
c=MonteCarlo('GET VALUE = Z_pos ');

# Printing the X and Z position to the output file
fprintf(out,'%5d,%7.4f,%7.4f,%8.5f,%8.5f,%8.5f,%6.4f,%6.4f,%6.4f
',k,side,thick,degangle,lt,dg,a,b,c);

# Closing opened files
r=MonteCarlo('CLOSE');
end

# Closing the output file
fclose(out)
```

### Table C2: Explanation of the MAT function script file

<table>
<thead>
<tr>
<th>Line #</th>
<th>Script Language</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Function z=mc(i)</td>
<td>Defines a function named 'mc' with an input 'i' which signifies the number of Monte Carlo simulations.</td>
</tr>
<tr>
<td>2</td>
<td>pi=acos(-1);</td>
<td>Calculating the value of ( \pi ).</td>
</tr>
<tr>
<td>3</td>
<td>out = fopen('output-nodamp.csv','w')</td>
<td>Opens the file 'output-nodamp.csv' in write mode 'w' and sets 'out' as the file handle.</td>
</tr>
</tbody>
</table>
| 4      | fprintf(out,'Iteration,Side,Thickness,Initial AOA,lt,thick,degangle

<table>
<thead>
<tr>
<th>Line #</th>
<th>Script Language</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>setup_sim='TINC=0.0001, TMAX=5.,INT MODE=8,PRINT CONTROL=0, PRINT2=0';</td>
<td>Defines simulation commands. Sets the time increment TINC to 0.0001 sec, the simulation time TMAX to 5 sec and the integration method (INT MODE = 8) to fixed step Runge-Kutta. It doesn’t print any variables to an Easy5 output file (PRINT CONTROL = 0 and PRINT2 = 0).</td>
</tr>
<tr>
<td>6</td>
<td>for k = 1:i</td>
<td>Starting the loop for 'i' simulations</td>
</tr>
<tr>
<td>7-11</td>
<td>side = 0.2 + 0.3<em>rand(1); thick = 0.08</em>(1 + rand(1)); breadth = side; angle = pi/2<em>rand(1); degangle = angle</em>180/pi;</td>
<td>Randomly varying the ice particle length, thickness, and initial angle of attack using the random number generator function RAND.</td>
</tr>
<tr>
<td>12-15</td>
<td>len = sprintf('PARAMETER VALUES, L_FOMA = %7.4f',side); brd = sprintf('PARAMETER VALUES, B_FOMA = %7.4f',side); thk = sprintf('PARAMETER VALUES, T_FOMA = %7.4f',thick); ang = sprintf('PARAMETER VALUES, EUI_RB(1) = %8.5f',angle);</td>
<td>The function ‘sprintf’ assigns the entire string in the inverted commas to the variable except the format expression, %7.4f. It puts the value of the second argument in place of format expression. These commands set the parameter values to be used in the Easy5 simulation, e.g. L_FOMA is the parameter that defines the length of the ice particle in the Easy5 simulation and it is given the value of the variable ‘side’.</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>Line #</th>
<th>Script Language</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-19</td>
<td>r=MonteCarlo(len); r=MonteCarlo(brd); r=MonteCarlo(thk); r=MonteCarlo(ang);</td>
<td>Loads the values of ice particle length, breadth, thickness and initial angle of attack into the simulation.</td>
</tr>
<tr>
<td>20-25</td>
<td>lt = 0.1 - 0.2<em>rand(1); lft = sprintf('PARAMETER VALUES, T1_FM = %8.5f',lt); r=MonteCarlo(lft); dg = 0.1 - 0.2</em>rand(1); drg = sprintf('PARAMETER VALUES, T2_FM = %8.5f',dg); r=MonteCarlo(drg);</td>
<td>Randomly varies the percentage change in lift and drag and loads these values to the Easy5 simulation.</td>
</tr>
<tr>
<td>26-28</td>
<td>cdamp = 0.0166; damping = sprintf('PARAMETER VALUES, CDAMP = %6.4f',cdamp); r=MonteCarlo(damping);</td>
<td>Sets the value of aerodynamic damping and loads it into the Easy5 simulation.</td>
</tr>
<tr>
<td>29</td>
<td>r=MonteCarlo('PARAMETER VALUE, XSTOP=45.0');</td>
<td>Sets the value of XSTOP into the Easy5 simulation. XSTOP is the parameter which states where to stop the trajectory simulation in the 'Termination code module'.</td>
</tr>
<tr>
<td>30</td>
<td>r=MonteCarlo(setup_sim);</td>
<td>Loads the simulation setup parameters.</td>
</tr>
<tr>
<td>31</td>
<td>r=MonteCarlo('SIMULATE');</td>
<td>Runs the simulation.</td>
</tr>
<tr>
<td>32</td>
<td>k</td>
<td>Prints the simulation number on the MAT output pane.</td>
</tr>
<tr>
<td>33-35</td>
<td>a=MonteCarlo('GET VALUE = X_pos2'); b=MonteCarlo('GET VALUE = Y_pos '); c=MonteCarlo('GET VALUE = Z_pos ');</td>
<td>Obtain the values of X, Y and Z positions of the ice particle after running the simulation.</td>
</tr>
</tbody>
</table>
| 36     | fprintf(out,'%5d,%7.4f,%7.4f,%8.5f,%8.5f,%8.5f,%8.5f,%6.4f,%6.4f,%6.4f,%6.4f
',k,side,thick,degangle,lt,thick,a,b,c); | Prints the output to the output file. |
| 37     | r=MonteCarlo('CLOSE'); | Closes all the Easy5 files associated with the simulation. |
| 38     | end | FOR loop ends |
| 39     | fclose(out) | Closes the output file. |