

Lagrangian Particle Tracking on Large Unstructured Three-Dimensional Meshes

Markus Widhalm, * Arno Ronzheimer,† Jörg Meyer‡

Institute of Aerodynamics and Flow Technology, DLR, 38108 Braunschweig, Germany

A prerequisite for the prediction of ice accretion on an aircraft flying through clouds of supercooled liquid water is the accurate determination of the water impingement rate on various components of the aircraft. For this purpose, a droplet impingement module has been developed using the datastructure of the unstructured Navier-Stokes solver TAU. Since nowadays large computational grids are common practice, an efficient algorithm for determination of the droplet trajectories on such grids had to be implemented.

This paper describes the physics and details of the implemented numerical algorithm. It summarizes lessons learned during development. The paper concludes with the presentation of code validation results and examples of applications.

I. Introduction

The study of particle-laden turbulent flows has significant importance in a variety of scientific fields such as biology, chemical engineering, atmospherical sciences, oceanography, and aerospace engineering. For aircraft design and certification, the primary interest is transport of atmospherical precipitation (micron-sized supercooled water droplets in clouds and fog, rain droplets, snow flakes, ice crystals, and hail stones) in the flow field past an aircraft and their impingement on an airframe structure.

Two different approaches have been proposed to model particle motion in a fluid: the Lagrangian and the Eulerian approach. In the Lagrangian model, the motion of an individual particle is modelled. In contrast, particles are regarded as continuum in the Eulerian model and their average spacing is described by a particle density function. Both approaches have been used in a number of implemented particle trajectory codes.^{1, 3, 6, 12, 13, 16, 17, 20, 21, 26, 34, 36}

In this paper, the Lagrangian description has been employed to model particle motion. The Lagrangian model has the advantage that it can represent intersecting droplet trajectories observed in the vicinity of blunt bodies, see Figure 1. Moreover, an ambiguity is avoided which is exhibited by the Eulerian model in the case of two fluid streams merging over a sharp edge, see Ref. 30. However, for the Lagrangian model the particle densities are not available everywhere in the field. To partially alleviate this disadvantage, plane cuts of the field have been implemented, on which the particle density is computed.

In this paper, particles are assumed to be spherical and the only forces acting on them are aerodynamic drag, gravity and buoyancy forces. Other forces, like the Basset history force, the Saffman lift force or the effect of rarefaction on small droplets are neglected. The effect of particle movement on the surrounding fluid is neglected by the assumption of low particle concentration. Thus, the integration of particle trajectories is decoupled from the flow solution and can be performed as a post-processing step.

The motion of individual particles in the Lagrangian model is described by Newton's second law. Newton's second law is a second order ordinary differential equation, for the integration of which a number of standard methods are available, e.g. explicit Runge-Kutta schemes, Adams-Bashforth predictor/corrector methods, cf. Simeon³² or Shampine.²⁹ However, the particle drag is dependant on the relative velocity of particle and surrounding fluid. Thus, for every integration step, the velocity of the fluid at the particle position must be known.

*Research Scientist, Numerical Methods Branch, email:markus.widhalm@dlr.de

†Research Scientist, Transport Aircraft Branch, email:arno.ronzheimer@dlr.de

‡Research Scientist, Configuration Design Branch, email:joerg.meyer@dlr.de

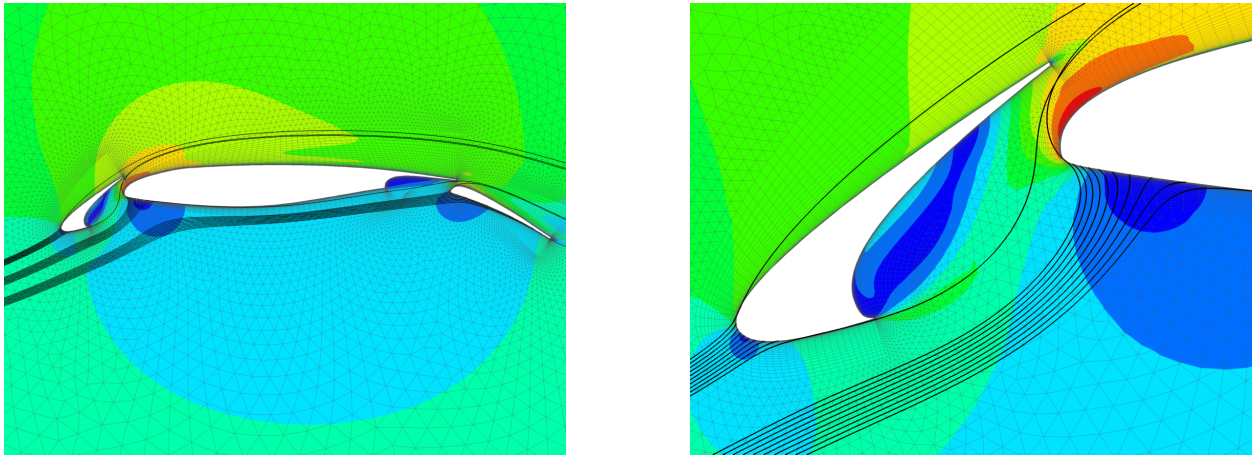


Figure 1. Particle trajectories around a three element airfoil (left) and close up view of intersecting trajectories between slat and wing (right).

To determine the fluid velocity at the particle position two procedures are performed in sequence: First the cell of the computational mesh is located in which the particle resides. Then the velocity in the cell is interpolated to obtain the fluid velocity at the particle position.

For the first step, a search algorithm is required, whereas the second step is performed by employing the linear basis functions of the basic grid volume elements. A number of particle tracking codes have been designed using these two ingredients, see e.g. Ref. 7, 13, 19, 27. With the advent of modern parallelized Navier-Stokes solvers like the DLR unstructured flow solver TAU,⁹ mesh sizes have grown to such an extent that the calculation of particle trajectories can only be performed in an efficient manner with a very careful implementation of the two above mentioned steps.

The present paper describes the implementation and validation of a very efficient particle trajectory code using the data-structure and libraries of the unstructured flow solver TAU. The development was performed along the following requirements:

- Calculation of catch efficiency for impinging particles.
- Automatic error control during integration of droplet/particle trajectories.
- Flux conservation for catch efficiency calculation.
- Particle concentration can be determined at any point in the flow field.
- Option for streamline visualization (i.e. mass-less particles).
- Compatibility with partitioned flow-field data.
- Numerical efficiency and robustness.
- User friendly control interface.

The paper is organized as follows: First the governing equations of particle motion in a fluid are introduced and algorithms for the numerical solution of the equations are discussed. Then we present validation results and lessons learned during implementation of the code. The paper closes with some real world problems to which the code has been applied.

II. Physics

II.A. Governing equations of particle motion

The governing equation for the motion of a particle in a fluid is Newton's second law,

$$m_p \ddot{\vec{x}}_p = \sum \vec{F}. \quad (1)$$

In the present paper, the index p denotes a particle property whereas index f is used for properties of the surrounding fluid.

The sum of all forces acting on the particle may consist of the aerodynamic drag and forces resulting from the density difference between particle and fluid (buoyancy and gravity),

$$\sum \vec{F} = \vec{F}_{\text{Drag}} + \vec{F}_{\text{Buoyancy}} + \vec{F}_{\text{Gravity}} + \dots$$

All other forces are neglected.

Introducing the relative velocity between particle and fluid $\vec{x}_{\text{rel}} = \vec{x}_p - \vec{x}_f$ and using a drag law, where the drag varies with the relative velocity squared, (1) may be written

$$m_p \ddot{\vec{x}}_p = -C_D A_p \frac{\rho_f}{2} |\vec{x}_{\text{rel}}| \vec{x}_{\text{rel}} - \rho_f V_p \vec{g} + m_p \vec{g},$$

where A_p , V_p , m_p and C_D are the cross section, the volume, the mass and the drag coefficient of the particle, respectively. ρ_f is the density of the fluid. \vec{g} is the gravity vector.

By rearranging the equation above, (1) may be expressed as

$$\ddot{\vec{x}}_p = -\frac{C_D \text{Re}_p}{24} \frac{18 \mu_f}{d_p^2 \rho_p} \vec{x}_{\text{rel}} + \frac{\rho_p - \rho_f}{\rho_p} \vec{g}. \quad (2)$$

with d_p , the particle diameter, and Re_p , the particle Reynolds number, defined by

$$\text{Re}_p = \frac{|\vec{x}_{\text{rel}}| d_p \rho_f}{\mu_f}. \quad (3)$$

If (2) is written in dimensionless form (dashed quantities are dimensionless),

$$\ddot{\vec{x}}_p' = -\frac{C_D \text{Re}_p}{24} \frac{1}{K} \vec{x}_{\text{rel}}' + \text{Fr}^2 \frac{\rho_p - \rho_f}{\rho_p} \frac{\vec{g}}{|\vec{g}|}, \quad (4)$$

then two similarity parameters of the particle-fluid interaction become apparent, the so-called inertia parameter K ,

$$K = \frac{d_p^2 \rho_p \dot{x}_{\text{ref}}}{18 \mu_f L_{\text{ref}}}, \quad (5)$$

and the Froude number, $\text{Fr} = \dot{x}_{\text{ref}} / \sqrt{|\vec{g}| L_{\text{ref}}}$.

The inertia parameter K is the ratio between a particle relaxation time and a characteristic time of the fluid flow. The Froude number Fr is the ratio between gravitational forces and fluid forces.

To allow numerical integration of (4), a relationship between Re_p and C_D needs to be defined. For spherical particles and $\text{Re}_p \ll 1$, Stokes' relation is applicable

$$C_D = \frac{24}{\text{Re}_p}. \quad (6)$$

Since much larger particle Reynolds numbers are encountered during calculation of trajectories, experimental results for the drag of a rigid sphere as given in Ref. 10 and 11 are employed. Langmuir and Blodgett¹⁰ proposed an analytical relationship approximating experimental drag data. However, the proposed relationship is computationally expensive. Therefore, a piece-wise approximation of the experimental data has been used in the present implementation, see Figure 2 and 3.

II.B. Definition and determination of local catch efficiency

The local catch efficiency, often designated β , is defined as the ratio of particle mass flux in the undisturbed oncoming flow to the mass flux of particles impinging on the surface. Since the particle flux can be thought of as being delimited by a streamtube, extending from undisturbed flow to the impingement locations on the surface, β corresponds to the cross section of this imaginary streamtube in the oncoming flow divided by the surface area on which the particles impinge.

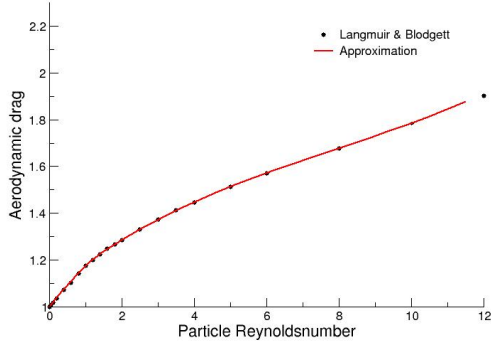


Figure 2. Experimental data and implemented approximation for sphere drag ($C_D Re_p/24$) for particle Reynolds numbers between 0 and 12.

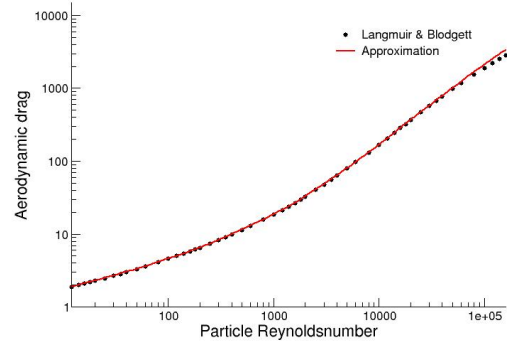


Figure 3. Experimental data and implemented approximation for sphere drag ($C_D Re_p/24$) for particle Reynolds numbers between 12 and 160000.

In the particle module of TAU, the streamtube is delimited by four particle trajectories to calculate the local catch efficiency. Part IV.A gives the rationale for the minimum number of four trajectories needed for this calculation. Thus, β is the ratio of the area A_0 spanned by the four particles in the release plane projected on to a plane perpendicular to the oncoming flow and the area A_m spanned by the corresponding particle impingement points, see Figure 4,

$$\beta = \frac{A_0 \cos \alpha}{A_m}. \quad (7)$$

The calculated β value is assigned to the centroids (+) of the quadrilaterals.

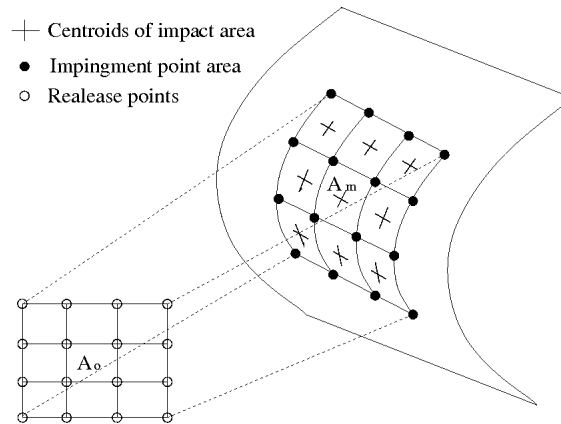


Figure 4. Determination of local catch efficiency β for 3D droplet impingement

III. Numerical Algorithm

The particle trajectory integration comprises of the following steps, compare Figure 5:

1. Release particles from a single point or from a predefined starting line or grid, preferably with equidistant spacing of release positions.
2. Locate the hosting element at release position.
3. Interpolate flow quantities to starting position.

4. Compute initial time-step size.
5. During each step of the particle integration the following steps need to be performed:
 - (a) Search the hosting element for the current particle position.
 - (b) Interpolate the flow quantities in the hosting element to the current particle position.
 - (c) Calculate the forces acting on the particle and integrate equation of motion to obtain new particle position.
6. Perform trajectory path integration until trajectory intersects with a boundary face of the computational domain.
7. Decrease time-step of integration rapidly until a minimum time-step is reached to locate the exact impingement position.

From the list above it is obvious that locating a particle position in the mesh, interpolating flow quantities to the particle position, and integrating the equations of particle motion are key components of any Lagrangian particle path integrator. Therefore, they are described in more detail below.

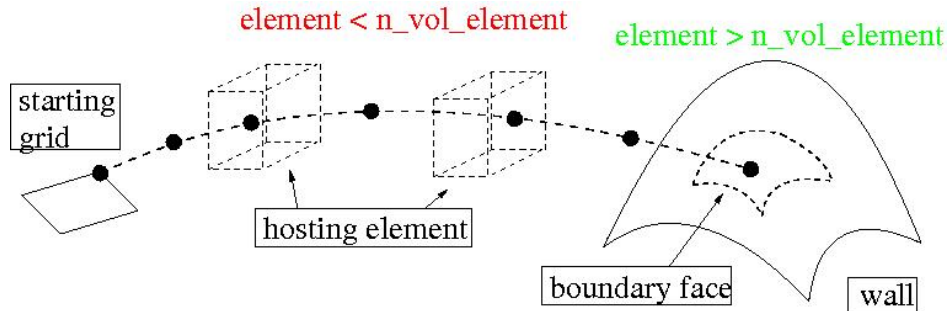


Figure 5. Integration of a particle path through several hosting elements until the boundary of the computational domain is reached.

III.A. Search and interpolation algorithm

For a given particle position, the search algorithm has the purpose of finding the grid volume in which the particle is currently located. Since searching the hosting volume element has to be performed several times during each integrator step, the computational efficiency of the search algorithm is crucial to a Lagrangian particle path integrator.

The search algorithm in the presented trajectory integrator is based on linear shape functions, as proposed by Löhner and Ambrosiano.¹³ The shape functions are implemented for all element types supported by TAU: triangles, quadrilaterals, tetrahedra, prisms, hexahedra and pyramids. The linear shape functions used for triangles and tetrahedra may be analytically inverted to find the element local coordinates corresponding to the particle position. For all other element types a Newton-Raphson iteration is employed to invert the shape functions.

Once local coordinates are determined, they may be used for three purposes: (1) The question inside/outside of an element can be answered by a simple comparison. (2) If the particle is inside the current element, the local coordinates can be used to interpolate the flow quantities to the particle position. (3) If the particle is located outside of the element, it is possible to determine the face, which is closest to the current particle position.

In particular the last feature may be combined with a list holding the element number of the element adjacent to each element face, see Figure 6. The adjacent element may be either a volume or a boundary element.

The adjacent element list is created in a preprocessing step which relies on the numbering of elements, faces, and points as defined in the TAU-code library. The neighbouring element list is implemented as a one-dimensional array, since the number of adjacent elements is known for every element type.

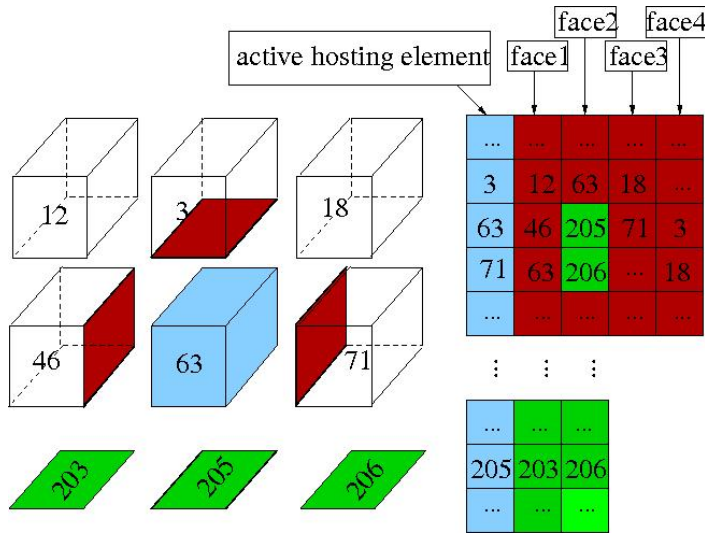


Figure 6. Example of the structure of a neighbouring element list.

In TAU, volume and boundary elements are numbered continuously with the boundary elements indices starting with an offset equal to the maximum number of volume elements, referred to in Figure 5 as `n_vol_element`. Thus, a trajectory leaving the computational domain can be easily detected and integration stops whenever the search algorithm returns an element number greater than the maximum number of volume elements, see Figure 5. This feature prevents the difficulty observed by Löhner¹⁴ with the path integration jumping over thin boundary structures.

III.B. Trajectory path integration scheme

Integration of the ordinary differential Equation (4) may be carried out either with multistep schemes (Adams-Bashforth type) or with multistage schemes (Runge-Kutta type) as described comprehensively by Darmofal and Haines.⁷

For the following reasons, we decided to perform the integration of Equation (4) by different embedded explicit Runge-Kutta schemes: Runge-Kutta schemes are not as sensitive as multistep schemes in the startup phase and step-size changes during the course of integration incur no overhead. Another advantage of embedded Runge-Kutta schemes is that error estimation is a free by-product of the integration step and can be conveniently used for time-step control.

The following four embedded Runge-Kutta methods are implemented in the code: A third order scheme (RK23) by Bogacki/Shampine,² two fourth order schemes (RK45) by Cash/Karp⁴ and by Dormand-Prince,⁸ and a sixth order integrator (RK68) by Dormand-Prince.⁸ In our experience, the (RK45) integrator by Dormand-Prince proved almost always to be superior to the other integrators.

Currently trajectory path integration has been implemented only for steady state flows.

IV. Lessons Learned during Code Implementation

IV.A. Striping with triangular grids for β evaluation

Equation (7) suggests that the calculation of β is straightforward as long as an arbitrary shaped start area A_0 and impingement area A_m are defined, delimited by an arbitrarily number of trajectories. However, it turns out that a minimum number of four trajectories is required to delimit the streamtube connecting the areas A_0 and A_m , which are then quadrilaterals. We performed experiments with four trajectories delimiting either a quadrilateral or two adjacent triangles that form a quadrilateral, but found striping in the β results for the triangles, see Figure 7. The observed behaviour can be attributed to the fact that triangular meshes are unable to properly represent area changes for shear distortions of the grid.

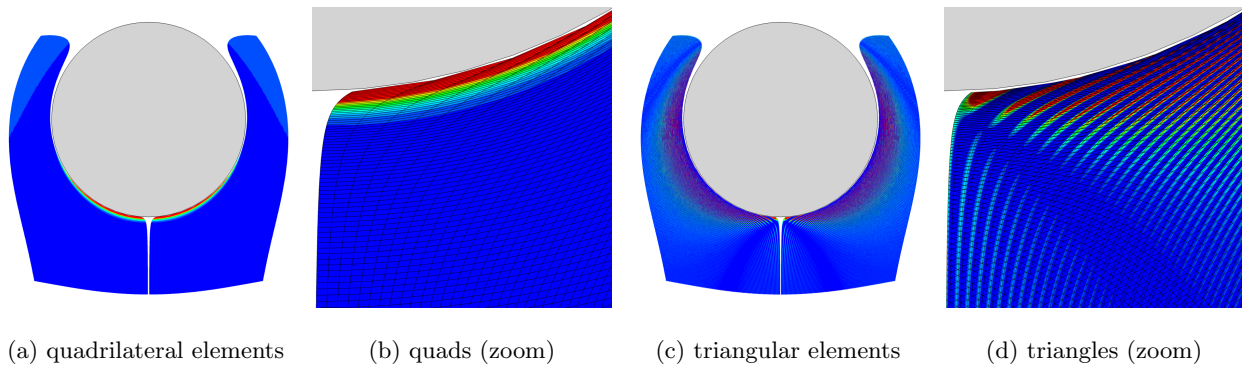


Figure 7. Particle concentration on a cut plane in the flow around a swept cylinder. β is calculated from identical trajectories using four trajectories delimiting quadrilateral areas in (a), (b) and triangular areas (c), (d). The mesh connects the centroids of the quadrilaterals in (b) and of triangles in (d).

IV.B. Back-tracking of Particle Path

An often complained disadvantage of particle trajectory methods is the trial and error procedure required to determine appropriate particle release positions. To alleviate this problem, we initially considered reversing the integration direction and to track particle trajectories back to their starting points. However, it soon became clear that even for exactly matched velocities, reverse integration never reached the starting point. Instead the particle after some distance departs from the corresponding forward integrated trajectory with very high speed normal to the main flow direction, see Figure 8. After reviewing literature, especially Ref. 29, it became clear that the observed behaviour is inherent in second order differential equations. Second order ordinary differential equation have a weak and a strong solution. During numerical integration of the differential equation, an algorithm will always converge towards the strong solution. For the forward integration direction the strong solution is to minimize the velocity differences. For the reverse integration direction the strong solution is to maximize the velocity differences between fluid and particle, which is the behaviour observed. Therefore, back-tracking a particle trajectory is in principle impossible.

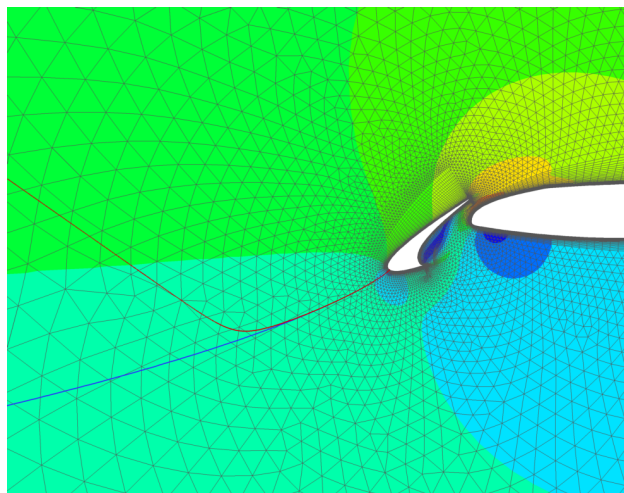


Figure 8. Forward integrated (blue) and backward integrated (red) particle trajectory in the flow around a three element airfoil.

However, there is a solution to the problem of selecting release points: If a mass-less particle is considered, then the equation of particle motion reduces to a first order differential equation. A first order ordinary differential equation has only a single solution if the solution exists at all. Therefore, the integration direction for the trajectory of a mass-less particle may be reversed without any problem. The trajectory of a mass-less particle corresponds to a streamline of the given flow field.

IV.C. Breaking dead-locks during search

The third difficulty encountered were occasional infinite loops during the search of particle location and a subsequent program dead-lock. A closer investigation of the problem revealed that the dead-locks always occurred if the particle position was near the faces of a volume element, where the location search algorithm performed an endless search through multiple adjacent elements. It turned out that the infinite search was caused by round-off errors in the basis function representation of the elements: the search algorithm determined that the given current point location was never in the currently considered element but in one of the adjacent elements.

Resolving the dead-lock was only possible by allocating a list of elements considered in the search up to now. Whenever during the search an element was encountered a second time, the element size was temporarily increased to allow the particle to be located inside of the element and values of the flow variables extrapolated to the particle position. Particle path integration will then continue normally.

V. Validation of the Implementation

The particle trajectory module for TAU has been validated by comparison with analytical impingement data reported in the literatur for a circular cylinder,¹⁵ a sphere³¹ and an experimental investigation on a NACA 23012 airfoil.²³ A few representative results of the cylinder test case will be shown below.

The local catch efficiency and impingement limits of the particles impinging on the cylinder depend on the size of the cylinder and the particle diameter as well as the flow conditions. In general, two similarity parameters are required to summarize this influence, e.g. the inertia parameter K , see Equation (5), and the particle Reynolds number, see Equation (3). However, in their investigation Langmuir and Blodgett¹⁰ used K and a similarity parameter designated ψ instead of the particle Reynolds number. The similarity parameter ψ is defined by

$$\psi = \frac{\text{Re}_0^2}{K}, \quad (8)$$

where Re_0 is a particle Reynolds number expressed with fluid freestream velocity.

From the experimental data presented in Ref. 15 and Ref. 10, we selected the total collection efficiency E and the maximum impingement angle q and compared them to numerical results. The impingement angle q is measured from the horizontal axis to the impingement location on the cylinder. Figure 9 and Figure 10 depict the comparison of the collection efficiency E and the maximum impingement angle q for two of the four considered ψ values, namely $\psi = 1000$ and $\psi = 10000$. The numerical results of the particle path integrator are in both cases very close to the experimental data of Ref. 15 and Ref. 10. For very high inertia parameters K , the impingement angle exhibits wiggles that are due to the fact that for trajectories impinging close to the 90° position, a very small numerical error leads to a large change of the impingement angle. Contrary to Ref. 10, no damping procedure has been applied while determining the maximum impingement angle.

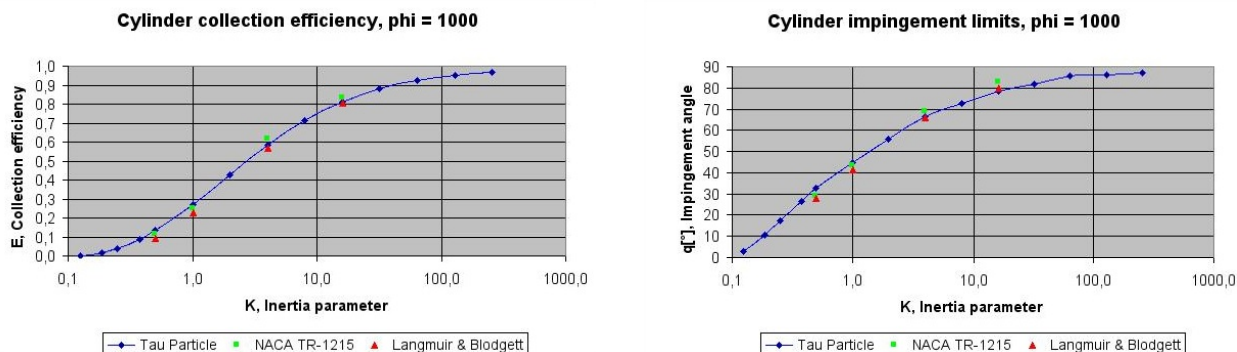


Figure 9. Total collection efficiency E (left) and maximum impingement angle q (right) for impingement on a cylinder, $\psi = 1000$.

All experimental data used in the comparison and a more detailed description of the test cases may be found in Ref. 15 and in Ref. 10.

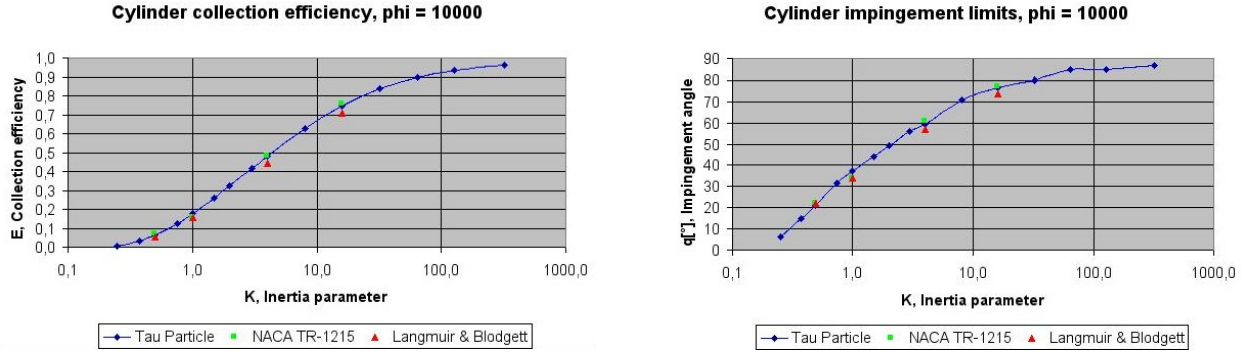


Figure 10. Total collection efficiency E (left) and maximum impingement angle q (right) for impingement on a cylinder, $\psi = 10000$.

VI. Applications

VI.A. Very High Bypass Ratio Nacelle - VHBR

The first test case, a generic VHBR nacelle, has been selected to demonstrate an impingement analysis for a droplet spectrum, automatic refinement of droplet release points, and the numerical efficiency of the TAU particle module on three-dimensional grids. The generic VHBR nacelle is a wind tunnel model tested at a Mach number of 0.2 and a Reynolds number of 2 million. The angle of attack was set to 8 degrees with the engine operating at full thrust. This test condition simulates take-off with water impingement. The computational grid contained about 1.8 million points and was generated with the software package Centaur.⁵ Details of the engine including impingement on fan blades and other engine components is not modelled.

For the impingement analysis, a so-called Langmuir D droplet size distribution was employed as defined in Ref. 10. The Langmuir D distribution is a more realistic representation of the droplet size distribution in icing clouds than a monodisperse droplet spectrum, i.e. droplets with only a single diameter. A median volumetric diameter (MVD) of $20 \mu\text{m}$ was selected for the droplet distribution. For an MVD of $20 \mu\text{m}$, Table 1 gives the droplet diameters and percentage of the total water content in the seven diameter classes of a Langmuir D distribution. These values have been used in the droplet impingement calculation, which produces results for any number of droplet diameter classes in a single calculation.

Diameter class #	1	2	3	4	5	6	7
Diameter ratio (d/d_0)	0.31	0.52	0.71	1.0	1.37	1.74	2.22
Diameter [μm]	6.2	10.4	14.2	20	27.4	34.8	44.4
Percentage of liquid water [%]	5	10	20	30	20	10	5

Table 1. Water droplet diameter distribution for a Langmuir D distribution with an MVD of $d_0 = 20 \mu\text{m}$.

For an impingement analysis of an engine nacelle, selecting droplet release locations is particularly challenging, since the frontal area of nacelle leading edge and hub is small compared to the frontal area of the whole engine. Nacelle leading edge and hub are the only components where droplets will impinge. The release area, which is usually selected to be delimited by a quadrilateral for simplicity, has to enclose all trajectories impinging on the nacelle. If trajectories are released with equidistant spacing, then a large number of trajectories will be swallowed by the engine inlet boundary, leading to an inefficient use of the started trajectories, see Figure 11(a). A solution to this problem is automatic local refinement of the start grid, from which droplet trajectories are released. This feature of the TAU particle module is demonstrated in Figure 11.

Figure 11(a) shows the β -distribution and impingement limits on the VHBR nacelle and the β -distribution on the start grid in front of the engine for a 150×164 starting grid without any refinement. It is obvious, that the impingement limits depicted by a red line on the nacelle and a black line on the hub have a large distance to the area for which the β -distribution has been calculated. A certain distance between the impingement

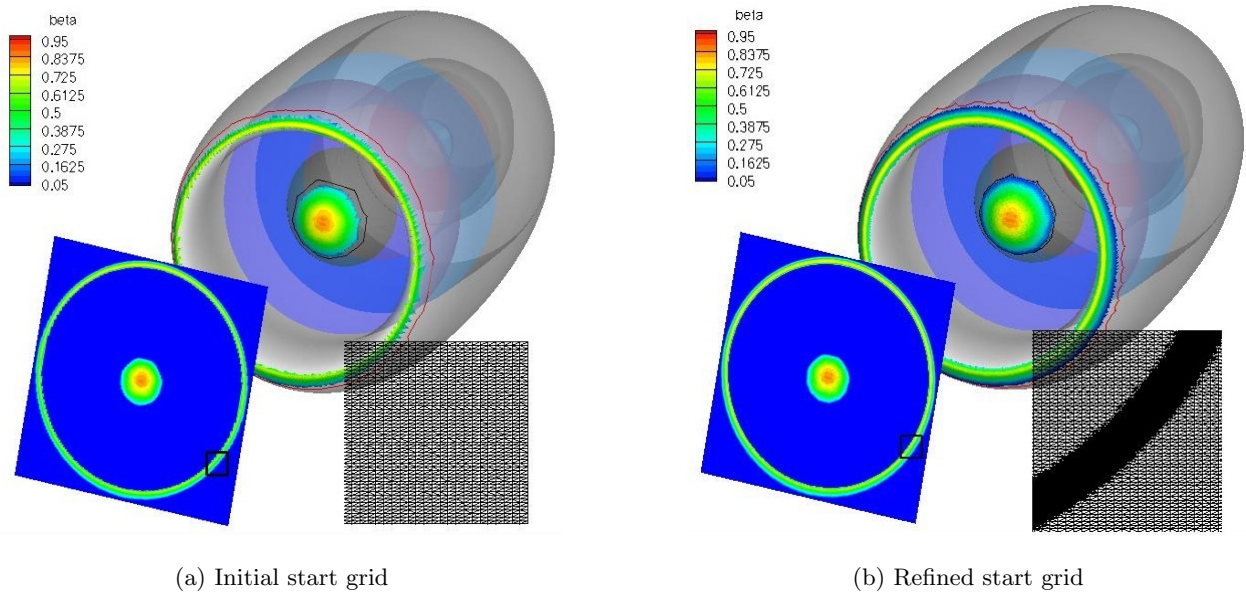


Figure 11. Initial (a) and refined (b) start grid with the computed β -distribution on hub and nacelle of an VHBR engine for a droplet diameter of $20 \mu\text{m}$. Impingement limits on hub and nacelle are depicted by black and red lines, respectively.

limits and the area, for which β has been computed, is normal for our implementation. The distance results from the fact that the impingement limit is simply the convex hull of all impinging trajectories, whereas β is only calculated for those quadrilaterals of the start grid, for which all four delimiting trajectories impinge. In addition, the edge of the area for which β has been computed is not very smooth. Obviously, the chosen start grid was too coarse and a refinement of the start grid is necessary.

The following strategy has been introduced in the TAU particle module in order to locally refine the start grid without significantly increasing the number of trajectories: Whenever any of the four trajectories delimiting a quadrilateral impinges, this quadrilateral is divided into four new quadrilaterals and five trajectories at the newly created corner points are launched for every refined quadrilateral. Thus, a quad-tree structure of quadrilaterals is built.

Figure 11(b) shows again catch efficiency and impingement limits for the same droplet diameter and initial start grid resolution as in Figure 11(a). However, this time the start grid has been refined up to seven times. The computational savings are enormous. Moreover, the impingement limits on nacelle and hub are much closer to the area where β has been calculated and the edge of the β -distribution is much smoother for this calculation.

Droplet diameter [μm]	6.2	10.4	14.2	20	27.4	34.8	44.4
# of particles released	50694	56178	55377	54386	54698	56771	57194
# of impinging particles	19309	26970	28186	29243	30568	32981	28762
Impingement Area Nacelle	0.0037	0.0095	0.0147	0.0214	0.028	0.0324	0.0370
Impingement Area Hub	0.001	0.0024	0.0034	0.0042	0.0045	0.0046	0.0047
WallClockTime (s)	327	307	271	250	210	202	187

Table 2. Number of starting and impinging droplets with impingement area and run time of for each droplet diameter class of a Langmuir D spectrum with an MVD of $20 \mu\text{m}$.

With this refinement strategy, all particle diameters in the Langmuir D spectrum with an MVD of $20 \mu\text{m}$ have been calculated. The starting grid resolution was taken again to be 150×164 trajectories and a maximum of 10 refinement levels have been used. Table 2 lists the impingement area for each droplet diameter. For reference, it should be mentioned that the surface area of the hub is 0.0049m^2 and 0.1454m^2 for the nacelle. As expected, the impingement area of the nacelle and hub increases continuously with droplet diameter due

to increasing droplet inertia. The hub impingement area levels out at a droplet diameter of $27.4 \mu\text{m}$, while the impingement area on the nacelle continuous to increase. The impingement area of the hub levels out because already droplets with a diameter of $27.4 \mu\text{m}$ impinge on almost the whole hub surface. Even larger droplets can not cover more than the whole surface.

The computations were performed on a single processor Linux PC. As seen from Table 2, the computational time decreases with increasing particle diameter. This behaviour is due to the fact that bigger droplets have larger inertia and are less influenced by the flow. For a droplet diameter of $20 \mu\text{m}$ around 220 trajectories per second were computed.

VI.B. Airbus A330 with fuselage mounted satellite antenna radome

Modern communication needed on long distance flights have recently increased the number of aircraft with onboard broadband satellite communication systems installed. Because of installed bandwidth, the antenna radomes have a comparatively large size. Due to aerodynamic and structural restrictions and the requirement that the antenna has to remain in line of sight of a satellite, only two installation locations on top of the fuselage are feasible: one in front of the fuselage wing intersection, see Figure 12, and the other behind this intersection.

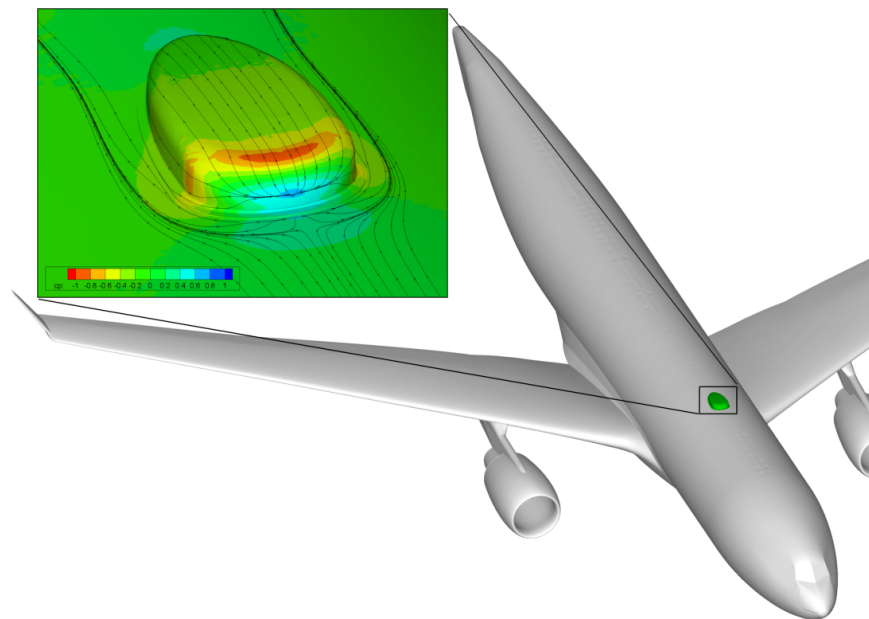


Figure 12. Position of satellite antenna radome on the forward fuselage of an A330 aircraft. Inset: c_p -distribution and skin friction lines (black) on satellite antenna radome.

Especially in the forward installation position, it is required to show for certification that the radome will not accrete ice in quantities that are sufficient to increase aircraft drag or can impact and damage other aircraft components if shed. In the past, the impingement analysis supporting certification has mostly been carried out with Euler or potential flow solvers. DLR has recently performed an investigation on a TW560 antenna system installed on an A330 with the Navier-Stokes flow solver TAU and the TAU particle module, Ref. 33. The flow calculation and impingement analysis of the radome were performed on a half model of the wing-body-nacelle configuration at a mesh size of about 10 million grid points. The flight and icing conditions corresponded to typical holding conditions.

The investigation revealed a very interesting flow topology and impingement pattern on the radome. Because of the blunt radome shape, the adverse pressure gradient in front of the radome leads to the separation of the fuselage boundary layer well in front the radome. Because of the early separation, the stagnation point on the forward facing side of the radome is not located near the fuselage attachment point of the radome, but at about two third of the radome's height. This is clearly visible from the blue color

of the c_p -distribution and the skin friction lines in the inset on Figure 12. Downstream of the separation a recirculation zone forms below the free shear layer. The recirculation zone in front of a blunt body is usually referred to as "horseshoe" vortex because of its characteristic form. A close investigation of the horseshoe vortex reveals that there are actually two vortices forming that are depicted in Figure 13. Figure 13 was produced with the TAU particle module operating in streamline mode (mass-less particles). It should be noted that none of the flow features described above could have been adequately described with a flow solver neglecting fluid viscosity.

In order to explore the impingement pattern on the satellite antenna radome in such a complex flow topology, impingement for two Langmuir D distributions with an MVD of 15 and 20 μm was computed. Surprisingly, impingement was only detected for droplet diameter classes with a diameter of 6.2 and 10.4 μm from the 20 μm Langmuir D distribution and with a diameter of 7.8 μm from the 15 μm Langmuir D distribution. Obviously, smaller droplets are too light to impinge. Larger droplets do not impinge because separation happens early enough to deflect the larger droplets away from the radome. Even for the diameter range from 6–10.5 μm , the size of the impingement zones is very small, see Figure 14. There is a small impingement area near the stagnation point and a second impingement zone on the front part of the radome fuselage attachment point. Apparently, the particles get deflected by the horseshoe vortex to impinge near the attachment point of the radome. Unfortunately, the two above mentioned impingement zones are connected by two large blue surfaces in Figure 14 due to a visualization problem.

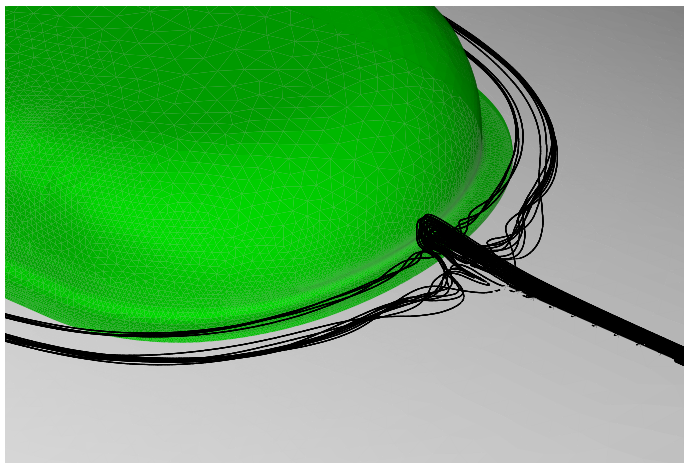


Figure 13. Streamlines of the flow past the satellite antenna radome forming two horseshoe vortices in front of the radome.

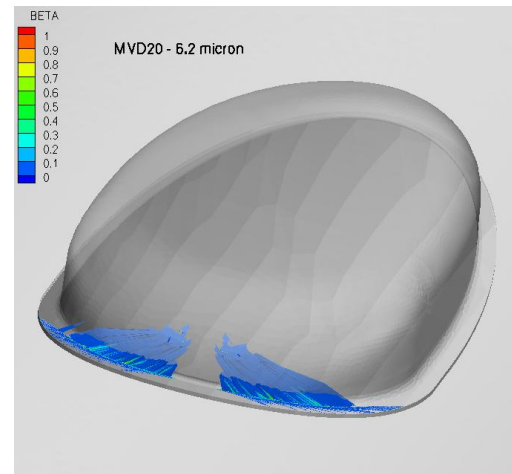


Figure 14. Impingement locations of droplets from the 6.2 μm diameter class of a Langmuir D distribution with an MVD of 20 μm .

VI.C. Dornier 228 flight icing experiment

A flight experiment with a Dornier 228 airplane was performed under a wide range of simulated icing conditions created by means of an icing tanker aircraft (Figure 15). The flight experiment HYLTEC¹⁸ (HYbrid Laminar TEChnology) has been designed to generate insight into the operating mechanisms and performance of two different wing ice protection systems available for transport aircraft. The data obtained during one of the flight tests have been compared with a numerical computation of ice accretion at the flight test conditions.

As a first step, a relevant CAD model of the Dornier 228 test aircraft had to be generated, see Figure 16. The original aircraft has a length of 15 m, a height of 2.4 m from sponson bottom to vertical stabilizer tip, and a wing span of 17 m. The wing area is 32 m². The two four-blade propellers have a diameter of 2.7 m and are both rotating counterclockwise at 1500 rpm. The exact dimensions of wing and laminar glove could be retrieved from design drawings of the experiment. For all other parts of the aircraft, such as fuselage, cockpit, dorsal, horizontal and vertical stabilizer, the CAD model is based on measuring the original DLR research aircraft.

Modelling of the propellers received special attention in the computer simulation, as it had been observed during the flight test from an externally mounted camera, see Figure 16, that water droplet trajectories were



Figure 15. DLR test aircraft Do228 in the water spray plume of the Dornier icing tanker aircraft.

strongly influenced by the right propeller. Moreover, about one fourth of the laminar glove span is located in the downwash of the right propeller. Therefore, an actuator disc model of the propellers based on blade element theory, see Ref. 28, was included in the computational model. The actuator disc creates a constant suction effect upstream and a constant swirl flow downstream of the disc. It is often a sufficient alternative for simulating a real rotating propeller which would require an unsteady simulation with prohibitively large computational times. The turbine inflow and outflow have been neglected in the computational model because of expected low influence. The computational mesh for the half model contained around 14 million points, of which approximately 9.5 million points were situated in the prismatic elements resolving the boundary layer.

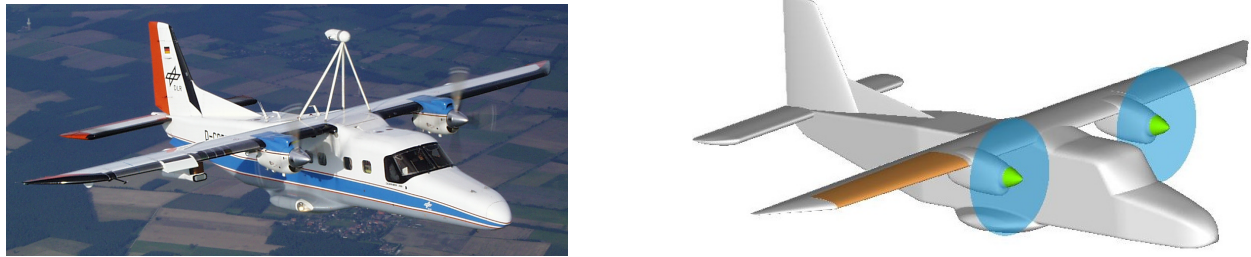


Figure 16. DLR test aircraft Do228 compared to the CAD model simplified for numerical analysis. The laminar glove is highlighted on the CAD model for better visualisation.

Table 3 lists the flight and icing conditions measured during the second flight test on 15th March 2002, see Ref. 18. During the flight test, the aircraft was in a clean configuration. It is well known, that Medium Volume Diameter (MVD) and Liquid Water Content (LWC) in an icing tanker plume depend on the distance between tanker aircraft and test aircraft, ambient temperature, and spray rig configuration (type and number of nozzles and pump performance). The distance between icing tanker and test aircraft during the flight tests has been measured by means of a differential GPS system and was in the range of 35–40 m. From the calibration charts of the icing tanker it could be concluded that the plume at the test aircraft position had an MVD of 35 to 40 μm plus some super large droplets (SLD). The complete spraying time lasted 13 minutes from turning on and turning off the spray rig.

The flight conditions in the simulation were set according to the conditions given in Table 3. Additionally, an angle of attack needed to be defined. However during the flight test, the angle of attack varied rapidly in a wide range to allow the test aircraft to stay at a certain fixed distance from the tanker aircraft and keeping the water spray plume centered on the wing. For the simulation, we decided to use only one fixed angle of attack set corresponding to a lift coefficient of 0.5, which agrees with level flight at this atmospheric conditions. All other necessary information for setting the propeller, such as blade angles, local lift and drag coefficients over the radius, was extracted from a similar propeller once developed for the Do228 at the DLR.

The numerical simulation was performed with the classic $k-\omega$ turbulence model, Ref. 35. Preconditioning

Flight and Icing conditions	
Altitude	13.000 ft
IAS	140 kts
Temperature	-10°C
LWC	$\approx 1.2 \text{ g/m}^3$
MVD	35 - 40 μm + SLD
Icing duration	13 min.
Ice thickness	$\approx 12 - 15 \text{ mm}$?

Table 3. Flight and icing conditions from the test flight for the numerical investigation.

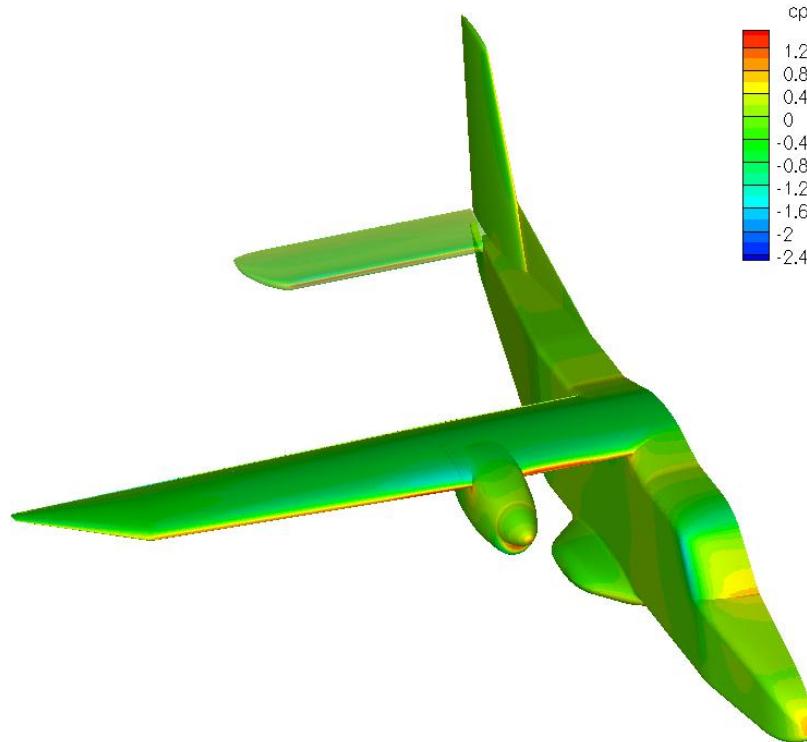


Figure 17. Computed surface pressure distribution on the Do 288.

due to the low Mach number proved to be unnecessary and a convergence of the density residual by 5 orders of magnitude to steady state was reached. Due to the low flight speed of 250 km/h no shocks appear in the flow. The pressure distribution on the surface of the Do228 model is shown in Figure 17. The influence of the swirl from the counterclockwise rotating propeller is well visible in Figure 17 generating an area of increased suction on the upper side of the wing outboard of the nacelle and a stronger pressure peak on the leading edge of the wing inboard of the nacelle.

For computation of trajectories in the propeller flow field, we had to introduce a special treatment of the actuator disc boundary condition in the calculation of particle trajectories. Contrary to all other boundaries of the aircraft, which are simply solid walls for the particles, the actuator disc had to be made transparent to particles. Thus, the particles would be able to experience the action of the higher velocity in the vicinity of the propeller and the swirl downstream of the propeller. The effect of the flow past wing and propeller on selected particle trajectories is depicted in Figure 18.

Figure 18(a) shows particles moving over the upper side of the wing. The most inboard row of particles moves through the propeller disc and encounters the upward-moving blade. Short behind the disc, the

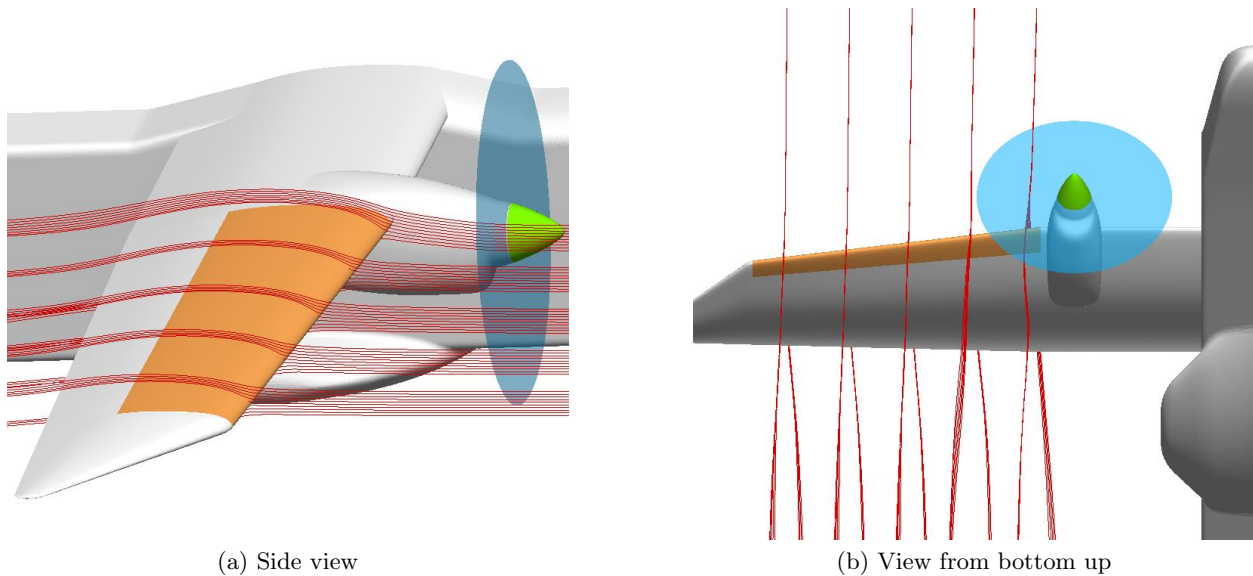


Figure 18. Particle trajectories influenced by propeller and aircraft wing. On the left, view from side. On the right, view from bottom up.

particles get an upward deflection caused by the swirl. In Figure 18(b), the spanwise displacement of particle trajectories by the interaction of the propeller downwash with the wing is depicted. The larger velocity in the propeller downwash increases local circulation on the wing. As can be seen in Figure 18(b) for the two most inboard rows of particles, this also increases the separation of the particles passing the wing on the upper and lower side.

A first impingement analysis was made for the wing only. The starting grid for particles was located 50 m in front of the airplane. A droplet diameter of $35 \mu\text{m}$ was chosen according to Table 3. The effect of gravity was neglected. The initial equidistant release mesh contained 30×800 trajectories. A maximum of 5 refinement levels were allowed. The computer time for this calculation was 20 minutes on a dual core Linux Opteron machine. In Figure 19, the catch efficiency distribution on the wing leading edge is shown. The catch efficiency β behaves as expected with a maximum near the stagnation line. The maximum β is around 0.62 on average but peak values of up to 1.7 are observed in the region behind the propeller. From the maximum value near the stagnation point, β decreases uniformly to zero.

A comparison of the impingement area with flight test results would require a flight test performed at very cold temperatures to avoid run-back combined with a very short spraying time. Unfortunately, such a flight test has not been performed. The closest approximation we could find in the video tape recordings of an outboard camera was a picture extracted after a few minutes of spraying when ice accretion just became discernable for the first time, see Figure 20. Due to the fact that the exact camera location of Figure 20 is known, we have tried to visualize Figure 19 from the same viewing position as the original camera. Comparison of both figures seem to indicate that the lower impingement limit of the flight tests was matched fairly well. However, the upper impingement limit seems to be not as far downstream in the calculation as compared to the flight test situation presented in Figure 20. This discrepancy may have been either caused by run-back icing, the angle-of-attack variations during the flight tests, or the influence of the supercooled large droplets present in the icing tanker plume. At present, it is not clear which of the three possibilities is the most relevant cause and this will be subject for further investigation in the future.

Additionally, we performed an impingement analysis for the complete airplane, see Figure 21. Droplets with a diameter of $35 \mu\text{m}$ were again released 50 m in front of the aircraft. For this calculation we used four different starting grids, one for the wing, a second for the cockpit, and the third and fourth grid for the vertical and horizontal stabilizer, respectively.

The results of the impingement analysis are depicted in Figure 21. They indicate a catch efficiency β that does not exceed 0.45 on the nose of the aircraft, cockpit windows, sponson, and vertical tail. On spinner, wing, and horizontal stabilizer however, the maximum catch efficiency exceeds 0.7. As expected, the maximum β observed on the leading edge of the horizontal stabilizer is larger than that on the wing's

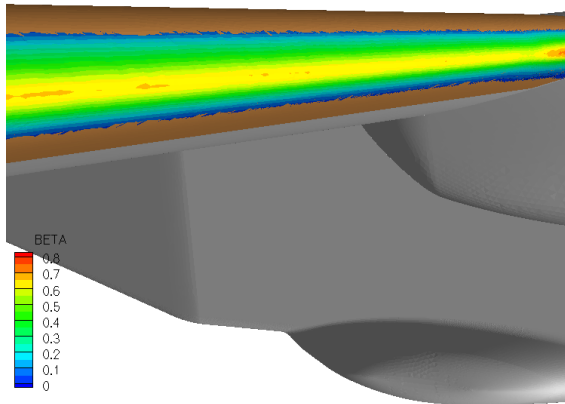


Figure 19. Computed catch efficiency on the Do228 wing visualized from a similar view point as in Figure 20.



Figure 20. Ice accretion on the Do 228 wing after a few minutes of spraying.

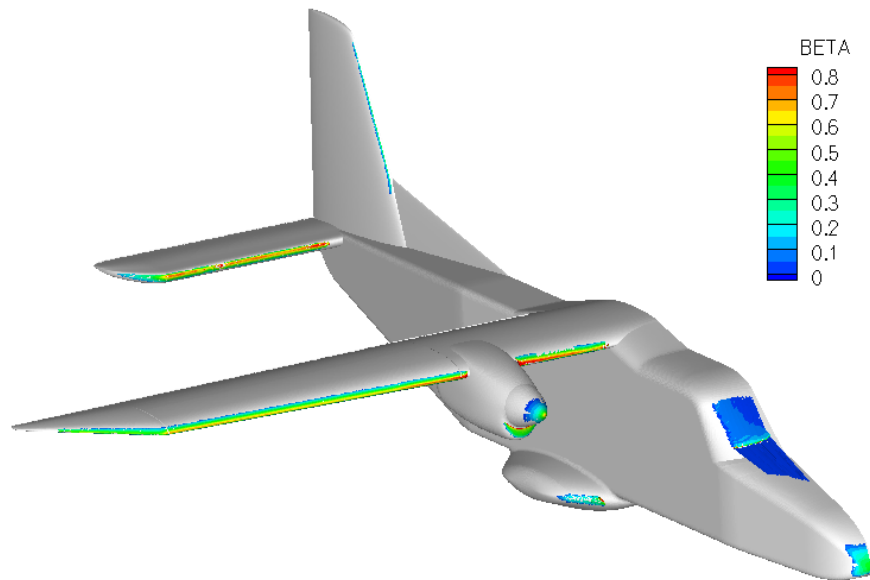


Figure 21. Catch efficiency β for the complete Do 228 aircraft.

leading edge since the chord-length and the leading edge radius of the horizontal stabilizer is much smaller than on the wing. A relatively high catch efficiency is also observed in the turbine inlet position of the nacelle. However, since the inlet was not modelled, it is not clear whether the catch efficiency on a real inlet lip would have had a similar magnitude or whether the largest amount of water would have been ingested by the engine.

In addition to computing the catch efficiency on the whole aircraft configuration, we have made a first attempt to calculate the resulting ice accretion on the wing and to compare it with an ice shape accreted during flight test. During flight test, the thickness of ice accretion was measured by a camera looking towards a disc mounted in streamwise direction on the wing leading edge, see Figure 22. On the disc, markings were applied with color bands of 10 mm thickness each in a different colour. In Figure 23, the ice accretion after 13 minutes of water spraying can be seen. The picture was taken short before the complete ice accretion began to shed from the upper side of the wing. The wing area behind the ice thickness disc is free of ice

because a hot air ice protection system was operated in this area. The ice consists mainly of glaze ice and on the upper side of granular rime ice. The ice thickness after 13 minutes of icing is about 30 to 35 mm as can be seen from the ice thickness disc in Figure 23.



Figure 22. Clean wing before water spraying with discs installed for measuring the ice thickness. View is from the wing tip mounted camera.



Figure 23. Ice shape on the laminar glove after 13 minutes of water spraying on the Do228 wing short before ice shedding.

Using the icing conditions given in Table 3 and the β distribution from Figure 19 we have made a first attempt to compute the expected ice shapes by applying Messinger's method. The resulting ice accretion is depicted in Figure 24 with the catch efficiency β shown below it.

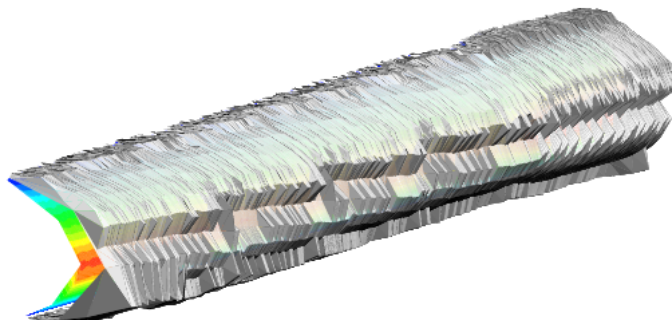


Figure 24. Computed ice accretion on the Do228 wing.

VII. Conclusions

This paper presented the development of an efficient particle trajectory integrator within the framework of TAU for computing particle catch efficiencies. The implementation of the particle trajectory module has been validated by comparison with analytical solutions and experimental data. The numerical results showed good agreement.

In the present paper, a few special features implemented into the trajectory integrator have been introduced, i.e. support for selecting appropriate particle starting locations, automatic refinement of the starting grids, and cut planes for determining particle concentrations at any predefined position in the flow field. All three features greatly simplify the application of the presented particle trajectory code in an operational environment, especially if impingement on very thin aircraft structures is considered.

The capability to visualize streamlines in a very large computational domain also helps to verify flow topologies and phenomena.

The particle trajectory module has been applied to three different real world problems: Impingement

on a generic very high bypass ratio nacelle, icing of a satellite radome antenna mounted on the top of an A330, and icing of a complete Dornier Do228 aircraft. Concerning the Do228 flight experiment, the numerical modeling was able to achieve the same relevant flow behavior as in the flight test. Subsequently, the presented comparison with the extracted flight test data and the numerical impingement and ice accretion results provide an insight about the prediction capabilities of the TAU particle tracer.

Summarizing, the Lagrangian particle integration has proven to be an efficient and promising tool and may be used as a stepping stone for a future tool suite to determine 3D ice accretions.

Acknowledgments

The author wishes to thank Christian Bartels of Airbus, Hamburg for his continuous support while creating this manuscript and Stefan Seydel of the DLR flight operations and flight facility department for his comments on setting up a proper Dornier 228 airplane model. Martin Hepperle and Axel Raichle developed propeller data which have been used for modeling the propeller of the Dornier 228 by an actuator disc.

References

- ¹Balászházy, I.: *Simulation of particle paths in bifurcating tubes*. J. Comp. Phys., 110, pp. 11–22, 1994.
- ²Bogacki, P. and Shampine, L.F.: *A 3(2) pair of Runge-Kutta formulas*. Appl. Math. Letters, 2, pp. 1–9, 1989.
- ³Chen, X.-Q. and Pereira, J.C.F.: *Efficient computation of particle dispersion in turbulent flows with a stochastic-probabilistic model*. Int. J. Heat Mass Transfer, Vol. 40, No. 8, pp. 1727–1741, 1997.
- ⁴Cash, J.R. and Karp, A.H.: *ACM Transactions on Mathematical Software* Vol. 16, pp. 201–222, 1990.
- ⁵*Centaur Soft*: <http://www.Centaursoft.com>
- ⁶Chorda, R., Blasco, J.A., and Fueyo, N.: *An efficient particle-locating algorithm for application in arbitrary 2D and 3D grids*. Int. J. Multiphase Flow, 28, pp. 1565–1580, 2002.
- ⁷Darmofal, D.L. and Haimes, H.: *An Analysis of 3D Particle Path Integration Algorithms*. J. Comp. Phys., 123, pp. 182–195, 1996
- ⁸Dormand, J.R. and Prince, P.J.: *A family of embedded Runge-Kutta formulae*. J. Comp. Appl. Math., Vol. 6, No. 1, pp. 19–26, 1980.
- ⁹Gerhold, T., Galle, M., Friedrich, O. and Evans, J.: *Calculation of Complex Three-Dimensional Configurations employing the DLR TAU-Code*. AIAA-97-0167 1997.
- ¹⁰Langmuir, I., Blodgett, K.B.: *A Mathematical Investigation of Water Droplet Trajectories*. AAF Technical Report 5418, 19. Feb. 1946.
- ¹¹Bowden, D.T., Gensemer A.E., and Skeen, C.A.: *Engineering Summary of Airframe Icing Technical Data*. Technical Report ADS-4, FAA, Washington, 1963.
- ¹²Kipfer, P., Reck, F., and Greiner, G.: *Local Exact Particle Tracking on Unstructured Grids*. Computer Graphics Forum, Vol. 22, Nr. 2, pp. 1–9, 2003.
- ¹³Löhner, R. and Ambrosiano, J.: *A Vectorized Particle Tracer for Unstructured Grids*. J. Comp. Phys., 91, pp. 22–31, 1990.
- ¹⁴Löhner, R.: *Robust, Vectorized Search Algorithms for Interpolation on Unstructured Grids* J. Comp. Phys., 118, pp. 380–387, 1995.
- ¹⁵Brun, R.J., Lewis, W., Perkins, P.J., and Serafini, J.S.: *Impingement of Cloud Droplets on a Cylinder and Procedure for Measuring Liquid-Water Content and Droplet Sizes in Supercooled Clouds by Rotating Multicylinder Method*. NACA tech. rep., NACA TR-1215, 1952.
- ¹⁶Naterer, G. F.: *Multiphase flow with impinging droplets and airstream interaction at a moving gas/solid interface*. J. Multiphase Flow, 28, pp. 451–477, 2002.
- ¹⁷Norment, H.G.: *Three-Dimensional Airflow and Hydrometeor Trajectory Calculation with Applications*. AIAA paper, AIAA 23rd Aerospace Sciences and Meeting, Reno, NV, January 14–17, 1985.
- ¹⁸Meyer, J.B.: *HYLTEC Do 228 - Hybrid Laminar Flow Technology Systems, Instrumentation and Flight Test*. DLR int. rep., DLR IB 124-2007/904, 2007.
- ¹⁹Li, G., Modest, F.: *An Effective Particle Tracing Scheme on Structured/Unstructured Grids in Hybrid Finite Volume/PDF Monte Carlo Methods*. J. Comp. Phys., 173, pp. 187–207, 2001.
- ²⁰Morency, F., Beaugendre, H., and Habashi, W.G.: *FENSAP-ICE: A Study of Effects of Ice Shapes on Droplet Impingement*. AIAA Paper, AIAA 2003-1223, 2003.
- ²¹Oliviera, P.J., Gosman, A.D., and Issa, R.I.: *A method for particle location and field interpolation on complex, three-dimensional computational meshes*. Adv. in Engineering Software, Vol. 28, pp. 607–614, 1997
- ²²Papadakis, M., Elangovan, R., Freund M., George, A., and Breer, M.D.: *Water Droplet Impingement on Airfoils and Aircraft Engine Inlets for Icing Analysis*. J. Aircraft, vol. 28, no. 3, pp. 165–174, 1991.
- ²³Papadakis, M., Rachman, A., See-Cheuk, W., and Hsiung-Wei, Y.: *Water Impingement Experiments on a NACA 23012 Airfoil with Simulated Glaze Ice Shapes* AIAA paper, AIAA 2004-0565, 2004.
- ²⁴Papadakis, M., Hung, K.E., and Yeong, H.W.: *Experimental Investigation of Water Impingement on Single and Multi-Element Airfoils*. AIAA paper, AIAA 2000-0100, 2000.

²⁵Papadakis, M., Hung, K.E., Vu, G.T., Yeong, H.W., Bidwell, C.S., Breer, M.D., Bencic, T.J.: *Experimental Investigation of Water Droplet Impingement on Airfoils, Finite Wings, and an S-Duct Engine Inlet*. NASA technical memorandum, NASA TM-2002-211700, 2002.

²⁶Patankar, N.A., Joseph, D.D.: *Lagrangian numerical simulation of particulate flows*. J. Multiphase Flow, 27, pp. 1685–1706, 2001.

²⁷Pokrajac, D., Ladic, P.: *An efficient algorithm for high accuracy particle tracking in finite elements*. Advances in Water Resources, 25, pp. 353–369, 2002.

²⁸Raichle, A., Melber-Wilending, S., Himisch, J.: *A New Actuator Disc Model for the TAU Code and Application to a Sailplane with a Folding Engine*. Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Springer, STAB 2006, Darmstadt.

²⁹Shampine, L.F.: *Computer Solution of Ordinary Differential equations. The initial value problem*. W.H. Freeman, San Francisco, 1975.

³⁰Tong, X.-L. and Luke, E.A.: Eulerian Simulations of Icing Collection Efficiency Using a Singularity Diffusion Model. AIAA paper, AIAA-2005-1246, 2005.

³¹Dorsch, R.B., Saper, P.G., and Kadow, C.F.: Impingement of Water Droplets on a Sphere. NACA technical note, NACA TN-3587, 1955.

³²Simeon, B.: *Ordinary partial Differential Equations* Lecture Series, Technical University of Munich.

³³Wild, J. and Widhalm, M.: *Calculation of Water Impingement on the TW560 Antenna installed on the A330-200 and A330-300*. DLR int. rep., DLR IB 124-2006/915, 2006.

³⁴Seldner, D., Westermann, T.: *Algorithms for Interpolation and Localization*. J. Comp. Phys., 79, pp. 1–11, 1988.

³⁵Wilcox, D.C.: *Turbulence Modeling for CFD*. 2nd ed. 1998, DWC Industries, La Cañada, California.

³⁶Zhou, Q.; Leschziner, M. A.: *An improved particle-locating algorithm for Eulerian-Lagrangian computations of two-phase flows in general coordinates*. Int. J. Multiphase Flow, 25, pp. 813–825, 1999.