

CFD simulation of particle accretion by Euler-Euler approach

Submitted by: Matthias RENNER

Registration Number: 1110587032

Academic Supervisor: Dr.rer.nat Wolfgang Hassler

Date of Submission: 29 January 2014

Declaration of Academic Honesty

I hereby affirm in lieu of an oath that the present bachelor's thesis
entitled

"CFD simulation of particle accretion by Euler-Euler approach"

has been written by myself without the use of any other resources than
those indicated, quoted and referenced.

Graz, 29 January 2014

Matthias Renner,

A handwritten signature in blue ink, appearing to read 'Renner', with a long, horizontal, wavy line extending to the right.

Preface

This work was written during the bachelor's degree program at the University of Applied Sciences Graz department Aviation. It deals with the numerical simulation of airflow around a profile to calculate catch efficiency and it describes the basics of aircraft icing. The information used was found in several sources, mostly provided by my supervisor Dr. Hassler and in the User's Manual of the software I worked with.

I think that this paper gives the reader a good overview of the numerical methods that were used and it may awaken interest to delve into the topic of aircraft icing and numerical simulations.

When this thesis was written, the department of Aviation lost its Head of Department, DI Bruno Wiesler, who was also a professor at the university in the lecture of thermodynamics and a dedicated researcher in the field of unmanned aerial vehicles for civil use. Therefore I want to devote this work to him.

Table of contents

Kurzfassung	5
Abstract	6
List of figures.....	7
List of used symbols	8
List of tables	9
Acronyms.....	10
1. Introduction.....	11
1.1. Liquid Water Content (LWC)	11
1.2. Collection efficiency	11
1.3. Icing conditions	12
1.3.1. Rime ice	14
1.3.2. Glaze ice.....	14
1.3.3. Beak ice.....	15
1.3.4. Icing hazards.....	15
1.4. Multiphase flow.....	17
2. Methods	18
2.1. Numerical simulations of a multiphase flow	18
2.1.1. The computational domain	18
2.1.2. The Lagrangian formulation.....	19
2.1.3. The Eulerian approach	20
2.2. Technology used	22
2.2.1. Hardware.....	22
2.2.2. Software.....	23
2.3. Mesh generation in ICEM CFD	23
2.4. Simulation setup in CFX-Pre 13.0.....	24
2.4.1. Solver control and analysis type	24
2.4.2. Calculation of the collection efficiency by Euler-Lagrange ...	25
2.4.3. Calculation of the collection efficiency by Euler-Euler	26
2.5. Running the setup in CFX-Solver Manager 13.0	27
3. Results	28

3.1.	Lagrangian formulation	28
3.1.1.	Solver run	28
3.1.2.	Water wall mass flow density and volume fraction.....	29
3.1.3.	Collection efficiency with Lagrange	32
3.2.	Eulerian approach	34
3.2.1.	Solver run	34
3.2.2.	Impingement limits and volume fraction	34
3.2.3.	Collection efficiency with Euler-Euler	36
Discussion	39
3.3.	Discussion of methods	39
3.3.1.	Lagrangian method	39
3.3.2.	Euler-Euler approach	39
3.4.	Discussion of results.....	40
4.	Summary and outlook	41
References	42

Kurzfassung

Diese Arbeit behandelt 2 Arten der numerischen Berechnung der „collection efficiency“ eines 2-dimensionalen Profils. The Ergebnisse dieser Simulationen werden zukünftig in Vereisungscodes eingesetzt werden, um die Form von Eisanlagerung vorherzusagen. Da Vereisung an Flügeln und Steuerflächen eines Luftfahrzeuges ein sicherheitskritisches Thema ist betreibt auch die FH Joanneum Forschung in diesem Bereich. Dazu wurde von einer Arbeitsgruppe des Studiengangs Luftfahrt ein eigener Vereisungscod mit dem Namen ICEAC2D entwickelt.

Die Berechnungen in dieser Arbeit werden einerseits mit der Methode von Lagrange durchgeführt, bei der die Wassertropfen als einzelne Partikel betrachtet werden und so ihre Flugbahn ermittelt wird. Die zweite Methode basiert auf dem Eulerverfahren, welches die Flüssigkeit als zerstäubtes Fluid annimmt. Beide Simulationen wurden mit dem Programm ANSYS CFX durchgeführt.

In dem Kapitel Results werden die berechneten Strömungsfelder die collection efficiencies, sowie die benötigte Rechenzeit dargestellt. Das Hauptziel dieser Arbeit ist der Vergleich der Lagrange und der Eulermethode um für verschiedene Fälle die beste Berechnungsart auswählen zu können.

Im letzten Teil dieses Dokuments werden die Vor- und Nachteile der beiden Methoden, basierend auf den zuvor berechneten Werten, evaluiert. Dieser Abschnitt enthält außerdem Verbesserungsvorschläge für zukünftige Projekte.

Abstract

This thesis treats two different ways of calculating the collection efficiency on a 2-dimensional aerofoil. The data gained will later be used in an icing code to predict ice shapes. Since icing on aircraft components is a critical topic in aviation the University of Applied Sciences Graz also is doing research on icing simulation and de-icing procedures. For this purpose an icing code called ICEAC2D was developed by a project group of the University's department for Aviation.

The calculations in this paper are on the one hand executed using the Lagrangian formulation where the airflow and the particle tracks are computed separately. On the other hand the Eulerian method is shown where the water droplets are assumed to be a dispersed fluid in the continuum air. For both methods the software ANSYS CFX 13.0 is used.

In the results section the computing time, the flow fields and the collection efficiencies are displayed. The main objective of this work is the comparison of the Lagrangian and the Eulerian approach.

Based on the results, the approaches are discussed to show their most convenient fields of application considering advantages and drawbacks of the simulation methods. The last part of this paper also provides possible points for improvement for future projects.

List of figures

Fig. 1.1: Typical shape of rime ice (Paraschivoiu et al., 2004, p. 28).	4
Fig. 1.2: Growth of glaze on an aerofoil (Paraschivoiu et al., 2004, p. 28).	5
Fig 1.3: Beak icing (Gent et al., 2000, p.2876).	5
Fig 2.1: Impingement limits on an aerofoil in the Lagrangian formulation (Silveira et al., 2003, p.3).	10
Fig 2.2: C-Domain with locally refined mesh.	14
Fig 3.1: Water volume fraction contours for NACA 0012 aerofoil.	29
Fig 3.2: Water wall mass flow density on the profile surface.	30
Fig 3.3: Water wall mass flow density of a NACA 0012 profile at 0° angle of attack and 50 000 particles.	31
Fig 3.4: Water wall mass flow density at 350 000 particles.	31
Fig 3.5: Water wall mass flow density at 500 000 particles.	32
Fig. 3.6: Collection efficiency for the upper surface.	33
Fig. 3.7: Collection efficiency for the lower surface.	33
Fig. 3.8: Water volume fraction in the Eulerian calculation	35
Fig. 3.9: Impingement limits demonstrated with the water mass flow	36
Fig 3.10: Velocity field of the water droplets.	36
Fig. 3.11: Volume fractions over the X-position of the nodes as exported from CFX Post.	37
Fig. 3.12: Collection efficiency as calculated for the upper surface.	37
Fig. 3.13: Volume fractions over the X-position of the nodes for the lower surface.	38
Fig. 3.14: Collection efficiency as calculated for the lower surface.	38

List of used symbols

β = local collection efficiency

E = global collection efficiency

S_F = total frontal area, m^2

\vec{V}_∞ = free stream velocity, m/s

LWC = liquid water content, g/m^3

\dot{m} = mass flow rate, g/s

α = volume fraction

\vec{n} = normal vector

List of tables

Table 2.1: Technical data of the computers at the department of Aviation.	23
Table 3.1: Comparison of the solver parameters at different particle numbers.	28
Table 3.2: Comparison of the water wall mass flow density at different particle numbers.	29
Table 3.3: Computing time and iterations until convergence for the Eulerian calculation.	34

Acronyms

CFD	computational fluid dynamics
HPC	high performance computing
MVD	mean volumetric diameter
RMS	root mean square
VDM	volume median diameter

1. Introduction

Since commercial flight operations regularly have to face the phenomenon of ice accretion on several components of an aircraft, numerous icing related accidents have happened. Not only because of this, efforts have been made to diminish or even prevent ice accumulation. To put into effect reasonable anti-icing or de-icing systems, the process and the types of icing have to be well understood. Numerical simulations can be auxiliary if they are executed in the right way. The topic of this thesis is the numerical simulation of the amount of water impinging on an aerofoil, termed collection efficiency, which is a crucial parameter in every icing code.

This section will introduce the reader to the field of ice formation and its impacts on aircraft performance and safety. It also contains reference to upcoming sections which will deal with the numerical methods and the parameters of a simulation.

The terms 'liquid water content' and 'collection efficiency' are two terms that will be used frequently throughout this thesis and that is why they will be explained in the following.

1.1. *Liquid Water Content (LWC)*

Taken from Paraschivoiu et al. (2004), the liquid water content is defined as the mass of water in a control volume or a cloud. The units of the LWC are normally given as grams of water per cubic meter of air (g/m^3). There are several parameters that will affect aircraft icing, but the liquid water content is considered to be the most important one (p. 37).

1.2. *Collection efficiency*

The collection efficiency is another important parameter in icing. Paraschivoiu et al. (2004) define the collection efficiency as the ratio of droplets impinging on a surface, to the droplets which would impinge if they would follow straight line trajectories (p. 40). Due to the deflection of the airflow for example at the leading edge, the water droplets are also deflected to a certain degree depending on their velocity and size. This degree of deflection influences the collection efficiency of an object. As found in Paraschivoiu et al. (2004) there is a local collection efficiency β

and a global collection efficiency E . For the calculation of ice shapes only β is of interest (p. 41). The calculation of β will be shown in section 2.1.2.

1.3. *Icing conditions*

Paraschivoiu et al. (2004) mention that there are mainly two conditions which have to be met to speak of icing conditions. On the one hand the ambient temperature has to be somewhere between 0°C and -40°C and on the other hand supercooled water droplets must be present (p. 27). A supercooled water droplet consists of pure water under 0°C. In this state the droplets are very unstable and change their phase from liquid to ice instantly when hitting a body, for example the leading edge of an aerofoil.

In means of temperature, Gent et al. (2000) demonstrate that not only is the ambient temperature important – which can easily be measured and therefore is available to the pilot – but in addition to that the temperature of the surface dictates whether and to which extent ice accretion may or may not take place. Thermal protection systems aim to raise the surface temperature above 0°C, either temporarily in de-icing or on a permanent basis as applied in anti-icing systems. For unprotected surfaces the total temperature is a more relevant parameter. For example the tips of a rotor blade will be roughly 20-25°C warmer than the root of the blade due to kinetic heating. A similar effect can be observed for military jets and any other aircraft travelling at an ‘ice free’ speed at Mach numbers > 0.6. Even though civil transport aircraft operate at such speeds, they need to be protected for their low speed procedures like take-off, climb and descend through clouds or flight in a holding pattern (p. 2879).

One final point to note is the change in the temperature of a surface during the initial period of icing, the direction of which varies depending on the total air temperature. If the total air temperature is below zero, then the convective and evaporative cooling mechanisms combine to dissipate the latent heat of the supercooled droplets and this maximizes the rate of accretion. The temperature of the accreting surface in this icing condition will tend to *rise* due to the release of latent heat and will be at the freezing point if the accretion is particularly wet. On the other hand, if the total temperature is positive due to kinetic heating (i.e. in the region of a rotor tip), the only ‘cooling’ mechanism is likely to be evaporation. Under these conditions, the surface temperature will *fall* to zero and hence ice may accrete.

(Gent et al., 2000, p. 2877)

Beside the temperature there are some other important parameters influencing the formation of ice.

Upon impinging on aircraft surfaces, the shape of the ice accretion which will be formed depends mainly on the liquid water content (LWC), the droplet diameter, the ambient temperature, the airspeed, the surface roughness and the shape of the accreting components which pick up the droplets. There are principally two different types of ice deposit which form on aircraft during flight through clouds, containing supercooled droplets, or precipitation.

(Paraschivoiu et al., 2004, p. 27)

For the relation between the concentration of water in the airflow and the mass of water which impacts on a surface, Gent et al. (2000) predict that the mass of water will increase linearly with the LWC and therefore the potential for ice accumulation rises. The collection efficiency has an equally important significance in calculating the actual mass of water present at a surface. If for example a large mass of water is portioned in droplets of a sufficiently small diameter, it may happen that only a few droplets impinge on the aerofoil while the rest is deflected by the airflow leaving the resulting accretion small. The maximum water impact is therefore given by an optimum combination of droplet diameter and LWC on the surface of a body of given size and shape. Since the diameter of the droplets will vary within a cloud, the spectrum is characterized by the volume median diameter (VMD) or sometimes referred to as median volumetric diameter (MVD). This is yonder diameter above and below which 50% of the mass or volume of water is contained. For the maximum diameter one can assume that it will be of the order of twice the VMD (p. 2881-2882). Gent et al. (2000) further note: "The range of diameters (mean effective droplet diameter, which equates to VMD) specified in JAR-29 extends from 15 to 40 μm (50 μm for intermittent icing)" (p. 2882). For the LWC the following requirements may be applied.

In the civil requirements (e.g. FAR 25, appendix C) or the UK military design requirements (see Ministry of Defence reports DEF STD 00-970 (1994) and 00-971 (1987)), the maximum continuous value of LWC that may be encountered is 0.8 g/m³ at 0°C. This is associated with stratoform clouds. At -30°C, the maximum LWC likely to be encountered in stratoform cloud reduces to 0.2 g/m³.

1.3.1. Rime ice

When operating under icing conditions, aircraft encounter in general three different types of ice shapes. Taken from Paraschivoiu et al. (2004) the first and least severe type of ice accretion is rime ice. Fig. 1.1 shows a typical shape of rime ice, which is formed at low temperatures, low airspeeds and a low LWC. This type of icing occurs when the supercooled water droplets hit the leading edge of the wing and freeze instantly, leaving no liquid water to run downstream. The visual appearance of rime ice is opaque and milky white due to the presence of air bubbles enclosed in the ice (p. 27). Due to the low LWC it is also of a rather dry nature. Penalties to the aircraft performance will be evaluated in section 1.3.4.

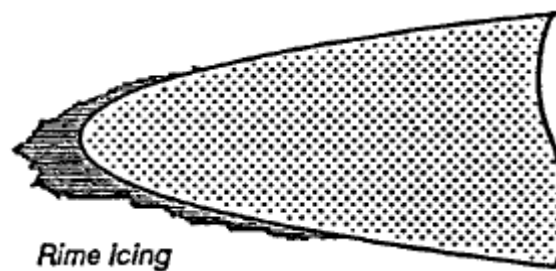


Fig. 1.1: Typical shape of rime ice (Paraschivoiu et al., 2004, p. 28).

1.3.2. Glaze ice

Paraschivoiu et al. (2004) also describe the second type occurring frequently, called glaze ice. Its appearance is clear and rather wet originating from the higher temperatures around 0°C at which this form of ice accretion can be encountered. The other parameters for the formation of glaze are high airspeed and higher LWC compared to rime ice (p. 28). A typical shape of this ice formation is shown in Fig. 1.2. The hazards accompanied by glaze ice are very severe and are also discussed in section 1.3.4.

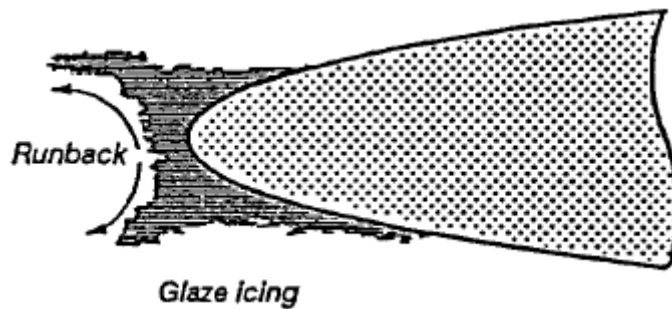


Fig. 1.2: Growth of glaze on an aerofoil (Paraschivoiu et al., 2004, p. 28).

1.3.3. Beak ice

Gent et al. (2000) describe the third form of ice accretion mainly found on the tip of a helicopter rotor in flight, where the effect of kinetic heating is diluted by cooling due to adiabatic expansion. The only region conducive to ice accretion is the low-pressure region on the upper surface of the profile close to the leading edge. The rigidity of beak ice is slushy and the visual nature is similar to rime ice (p. 2877). The shape of beak ice is demonstrated in Fig. 1.3.

Found in Gent et al. (2000) in general only one of the three types will be encountered on components like the fuselage, the stabilizer or the wing, but due to the varying speed along the span of a helicopter rotor, rime ice may be present near the root of the blade, glaze ice further outboard and beak ice in the regions of the rotor tips (p. 2877).



Fig 1.3: Beak icing (Gent et al., 2000, p.2876).

1.3.4. Icing hazards

Several incidents related to aircraft icing show that the ice shapes described in the previous section are affecting the performance of aircraft in a severe way.

Paraschivoiu et al. (2004) name the most serious penalties encountered as decreased maximum lift, increased drag, decreased stall angle, changes in the pressure distribution, early boundary layer transition, increased stall speed and reduced controllability (p. 57). Especially glaze ice is affecting the wings' properties. As Paraschivoiu et al. (2004) demonstrate, it is not only increasing the weight and changing the balance of the aircraft component, but also inflicts a dramatic change in the shape of the aerodynamic surface. This leads to a disturbed pressure distribution over the wing and a reduction of lift up to 30%, which makes this kind of icing the most severe one of the three (p. 28).

Gent et al. (2000) mention an example where a 35% loss in maximum lift, which is typical for the degradation imposed by ice accretion, would translate to an increase by 24% in stall speed, with obvious consequences to the safety of operations at low speed like landing or flight in a holding pattern (p. 2883). Gent et al. (2000) also name several other penalties. It is possible for ice accretion on propulsion system components like air intakes, engine nacelles, inlet ducts, propellers, fan blades, spinners or inlet guide vanes to reduce the propulsion efficiency and add to aircraft drag. During take-off iced aircraft will have an increased ground roll, decreased stall margins and climb rate. Ice on wings and tail surfaces can be particularly hazardous during take-off and can cause serious stability and control problems. During cruise iced aircraft suffer from a decreased maximum speed capability and efficiency, reduced ceiling height, higher fuel consumption and change in trim characteristics. For aeroplanes with one or more failed engines the combination of reduced lift, increased drag and reduced power can result in the loss of capability to maintain level flight. During approach or landing the combination of extended flaps and ice on the horizontal tailplane may cause the tailplane to stall, resulting in unwanted 'pitch-overs'. Last but not least ice on aircraft probes and sensors can lead to inaccurate data distorting the airspeed indication, leading to incorrect engine power settings and potentially the loss of the aircraft (p. 2885).

Considering all these hazards a numerical simulation of ice accretion and de-icing procedures can help to make the certification process of new aircraft safer and cheaper, saving hours of dangerous and cost intensive test flights. As mentioned before the calculation of the collection efficiency is a main step towards a successful forecast of the expected ice shapes, which makes it reasonable to encourage research in this region.

1.4. *Multiphase flow*

In the field of computational fluid dynamics (CFD), problems involving more than one fluid are common nature. The computation of the movement of water droplets in air can be seen as a prime example of a two phase multiphase flow. Air represents the continuous phase which means that it is one connected region over the whole domain. Water is the dispersed phase and is present in discrete regions which are not connected. Prosperetti and Tryggvason (2007) write that an exact definition of a multiphase flow is difficult and sometimes even depends on the experimenter's point of view or motivation. There are analytical methods, but they reach their limit rather early and are only suitable for small or large Reynolds numbers. Therefore numerical simulation and computation are the only tools to investigate flows at intermediate Reynolds numbers or systems of higher complexity (pp. 1-2). The following section treats different types of approaches and shows the calculation of the collection efficiency on a NACA 0012 profile.

2. Methods

There is a solution to every problem. In case of the simulation of water droplets in air there are two main types of trajectory calculations in use today. Gent et al. (2000) mention the first and most frequently adopted method to be the traditional Lagrangian formulation. The second and more recent type is the Eulerian approach (p. 2886) whose realisation in CFX 13.0 (ANSYS, Inc., 2010) was the main task of this work. At first, the two methods will be explained with some additional information in the sections below. Subsequently the process of calculating the collection efficiency by the Euler-Lagrange formulation and by the Euler-Euler approach using the software CFX 13.0 will be explained step by step.

2.1. *Numerical simulations of a multiphase flow*

At first, the term 'computational domain' will be clarified, because it is important to define the framework in which a simulation can be executed.

2.1.1. The computational domain

The definition of the computational domain was mentioned by Prosperetti and Tryggvason (2007) and relates to a profile or any other solid structure in a stream. The aim of a simulation is to compute the flow around an object in an unbounded domain but the software needs to have a specified area where the equations for each method can be evaluated. Therefore the computational domain is limited to a certain area surrounding the object of interest where far-field boundary conditions are fulfilled (p. 99). Prosperetti and Tryggvason (2007) quote a well-defined explanation of the computational domain for a solid sphere:

The placement of the outer boundary is dictated by a few competing requirements: the computational domain must be small enough so that the flow can be adequately resolved with as few grid points as possible, thus reducing the computational cost; the inflow and free-stream sections of the outer boundary must be sufficiently far from the sphere so that the far-field boundary condition can be imposed with high accuracy; and the outflow section of the outer boundary must also be sufficiently far downstream of the

sphere so that any unavoidable approximation of the outflow boundary conditions is prevented from affecting the flow upstream. Many simulations of flow over a sphere, especially those using spectral methods, have been done using an outer domain that is a sphere or an ellipsoid of radius about 10-15 times the particle diameter.

(Kurose and Komori, 1999; Johnson and Patel, 1999; Mittal, 1999; Bagchi and Balachandar, 2002c qtd. in Prosperetti and Tryggvason, 2007, p. 99).

2.1.2. The Lagrangian formulation

The principle behind the Euler-Lagrange approach is the treatment of the fluid phase as a continuum and the dispersed phase as a large number of particles, droplets or bubbles. For the fluid the time-averaged Navier-Stokes equations are solved, while the trajectories of the dispersed phase are calculated on a force balance. The dispersed phase may exchange momentum, mass, and energy with the fluid phase while the fundamental assumption is made, that the volume fraction of the dispersed phase is low. On the other side high mass loading ($\dot{m}_{particles} \geq \dot{m}_{fluid}$) is acceptable. For the calculation of the collection efficiency, Silveira et al. (2003) write that the droplets are considered to be rigid spheres which do not change their shape or phase, but are influenced by aerodynamic drag. Since the diameters of the droplets under typical icing conditions are very small, gravity and buoyancy can be neglected (p. 1).

A derivation of the force balance equations can be found in Silveira et al. (2000, p. 2). The derivation shows that a numerical simulation is required to solve the equations, since the drag force depends on the droplet velocity, which in turn depends on the local air velocity. Silveira et al. (2003) furthermore state that the initial position of the droplets is chosen so that the airflow in the near-field of the object is not affected. After disposal of the droplets, the trajectory equations are solved until the particles intercept with the profile or leave the computational domain. The impingement limits are calculated in an iterative process where the mean between one trajectory hitting the surface and an overlying one, not intersecting with the body, is computed. If this new trajectory misses the surface, it is chosen as the new upper limit and if it ends on the profile surface, it becomes the new hitting trajectory. The

process is repeated until the distance between the initial positions of the trajectories is smaller than a required value (p. 3). For better understanding the concept is shown in Fig 2.1.

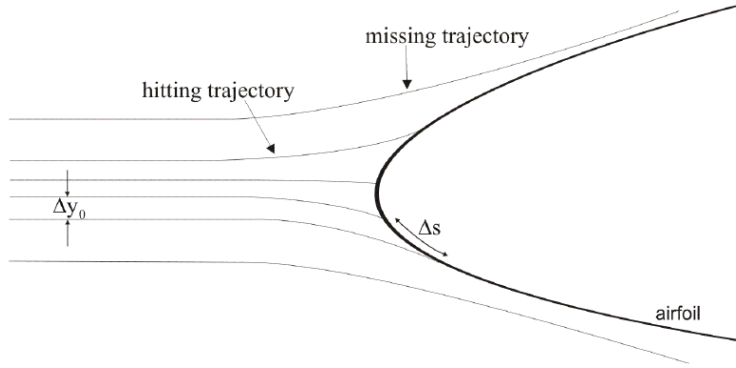


Fig 2.1: Impingement limits on an aerofoil in the Lagrangian formulation (Silveira et al., 2003, p.3).

Based on the impingement limits, Silveira et al. (2003, p. 3) defined the local collection efficiency β as

$$\beta = \frac{dy_0}{ds}. \quad (2.1)$$

To calculate the water mass flow rate that impinges on the wing, Paraschivoiu et al. (2004) multiply the maximum amount of water that would impinge, if the particles would follow straight lines, by the collection efficiency. The effective water wall mass flow rate is then given by

$$\dot{m}_e = \beta V_\infty S_F LWC \quad (2.2)$$

where S_F is the total frontal area.

This coefficient depends not only on the parameters given in the equation but also the angle of attack, the shape and the maximum thickness of the cross section of the wing and the MVD (p. 87).

2.1.3. The Eulerian approach

There are several definitions and explanations for this formulation. According to the Fluent 6.1 Documentation (Fluent Inc.), in the Eulerian or Euler-Euler approach, the two phases are mathematically treated as interpenetrating continua. The extension from a single phase to a multiphase flow brings up the need for additional

equations and mechanisms for the exchange of momentum, heat, and mass between the phases. In addition to that the concept of phasic volume fractions has to be introduced, since the volume of a phase cannot be occupied by the other phases. These volume fractions are continuous functions of space and time and always have to sum to unity for one control volume.

Silveira et al. (2003) present two ways of calculating a multiphase flow with the Eulerian approach. On the one hand there is the passive scalar transport approach, where air is the continuous phase and water is the dispersed phase. On the other hand the full multiphase formulation assumes both fluids to be continuous (pp. 3-5). The boundary conditions at the profile and the governing equations are the main differences between these two approaches.

The governing equations for the continuous phase, which is air in this case, are given by Cao et al. (2008) as the 2D steady and incompressible Reynolds-averaged Navier-Stokes equations including a continuity equation, momentum equations, turbulent kinetic energy equation and a turbulent dissipation rate equation. Some assumptions can be made to ease the solution:

1. The droplets are not affecting the airflow, because their volume fraction is sufficiently small.
2. Only drag, arising from the airflow, is an external force imposed on the droplets and the turbulent fluctuations of the airflow are not affecting the droplet movement.
3. The physical parameters of the droplets are constant as no heat transfer or evaporation is present during the droplet movement.
4. As mentioned before the droplets are spheres with an MVD.

According to the first assumption the equations for air and water can be solved independently. For the dispersed phase the equations are reduced to the continuity and momentum equations for each component of the water velocity, because the energy equation needs not to be solved and fluctuation terms and phase-change can be neglected (pp. 244-245).

For the calculation of interfacial transfer of momentum, heat and mass, the contact surface area between the two phases must be known. CFX 13.0 offers the user either the mixture model or the particle model depending on the morphology of the fluids. Both models are responsible for the computation of the interfacial area density, which is the interfacial area per unit volume between the phases.

2.1.3.1. The mixture model

This model is rather simple and treats both phases symmetrically. It is only available when the morphology of the fluid pair consists of two continuous fluids as applied in the full multiphase formulation of Silveira et al. (2000). For details, see “The Mixture Model in the CFX-Solver Modeling Guide” (ANSYS, Inc., 2010).

2.1.3.2. The particle model

The morphology of the fluids for this model to be available in CFX 13.0 has to be Continuous Fluid | Dispersed Fluid or Continuous Fluid | Dispersed Solid. For the simulation of airflow containing water droplets around a NACA 0012 profile this model was chosen.

The following sections describe the steps taken to run the calculation of the collection efficiency.

2.2. Technology used

Specific software and hardware capable of dealing with the high amount of data and computing power are required to execute a numerical flow simulation. The technical infrastructure used for the realisation of this work will be discussed in the following.

2.2.1. Hardware

The department of Aviation at the University of Applied Sciences Graz features a high performance computing (HPC) laboratory, where computing power intense CFD analysis can be run using a computer cluster. For the 2D flow simulation applied here, one computing machine delivered sufficient power. The technical data of one of these computers is listed in table 2.1.

Table 2.1: Technical data of the computers at the department of Aviation.

Processor (CPU)	Intel® Xeon® CPU X5690 @ 3.47 GHz
Core speed	1596 MHz
Cores	12
Random Access Memory (RAM)	26.6 GiB

2.2.2. Software

In addition to the operating system Linux 2.6.34.10-0.6-desktop x86_64 distribution openSUSE 11.3 (x86_64) installed on the computers in the HPC laboratory, ANSYS CFX 13.0 Pre, ANSYS CFX-Solver Manager 13.0, CFD-Post 13.0 and MATLAB® were used for calculations and visualisation.

2.3. Mesh generation in ICEM CFD

The NACA 0012 profile for the calculations was generated in ICEM CFD by Dr. Wolfgang Hassler in a typical “C-Domain” as shown in Fig. 2.1 with two inlets on the left, an outlet on the right as well as two adiabatic walls on the top and at the bottom of the domain. The boundary conditions at the profile surface depended on the calculation and will be described in the respective sections. The grid was refined in the near-field and in the wake of the profile to resolve the boundary layer and the flow behind the aerofoil.

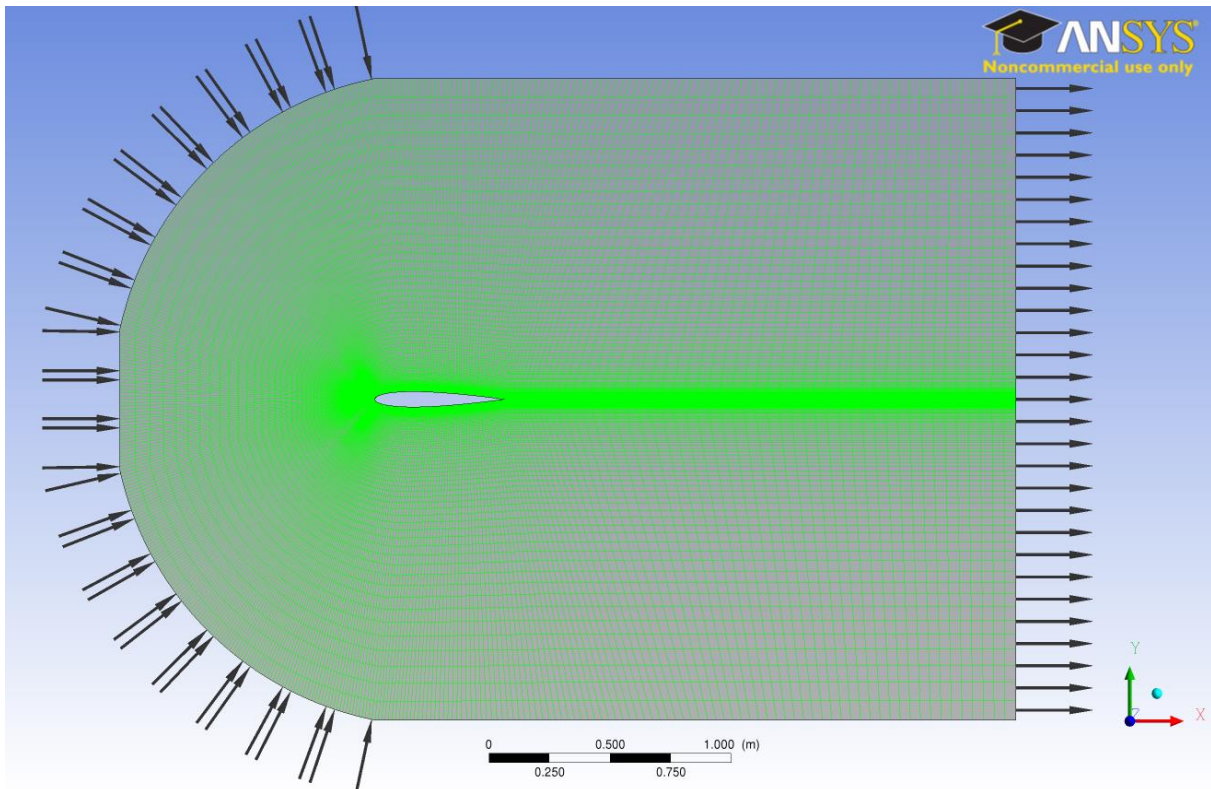


Fig 2.2: C-Domain with locally refined mesh.

2.4. Simulation setup in CFX-Pre 13.0

This section deals with the parameter settings in the CFX-Pre-processor. There are some properties that are consistent with Euler-Lagrange and Euler-Euler calculations and will therefore be described only once. The sections involved are section 2.4.1 and 2.5.

2.4.1. Solver control and analysis type

The simulation type chosen was steady state with no external solver coupling. The advection scheme and the turbulence numeric were both set to high resolution at a conservative length scale with auto timescale and time scale factor 1. To reach the convergence criteria of root mean square (RMS) residuals at $1e-06$, the maximum iterations were set to 500. The checkbox for Global Dynamic Model Control in the Advanced Options tab was set to check.

2.4.2. Calculation of the collection efficiency by Euler-Lagrange

The following sub-sections describe the fluids and particles for the calculation of the collection efficiency by particle tracking.

2.4.2.1. Fluid properties

Air was set as the continuous fluid with material properties out of the CFX 13.0 library for air at 25°C. The morphology for water was Particle Transport Fluid with a specified diameter of 40 micron. The domain model featured a reference pressure of 101325 Pa with a non-buoyant, stationary domain and no mesh deformation. The heat transfer was fluid dependent and the turbulence was set to Shear Stress Transport with an automatic wall function. The heat transfer model for air was based on total energy with a viscous work term while water had no heat transfer model. Since the equations for the flow field and the droplets were solved individually, the one-way coupling option for the particle coupling was sufficient and the drag force was modelled by a Schiller Naumann coefficient.

2.4.2.2. Boundary conditions

The boundary conditions for the outlet as well as for the top and the bottom wall were equal for the Euler and the Lagrange calculation and therefore will only be described here. The outlet boundary contained a subsonic flow regime and the option for mass and momentum was set to static pressure at 0 Pa relative pressure. The top and the bottom wall were both adiabatic free slip walls with an equation dependant wall interaction and perpendicular and parallel velocity restitution coefficients of 1.0.

To keep the required computing power as low as possible, the inlet consisted of two segments so that the particles were only released in an area where they had the chance to impinge on the profile. The main inlet, where only air was present, consisted of a subsonic flow regime with Cartesian velocity components. Since the calculation was executed at an angle of attack of 0°, only the U component was set to a value of 66.8 m/s. Turbulence was assumed to be low and the heat transfer was chosen to be at a static temperature of -26.1°C. The same conditions were applied to the particle inlet, except for the fluid values, where 350 000 water particles were uniformly injected at zero slip velocity and a mass flow rate of 0.000387963 kg/s.

This number was obtained by multiplying the *LWC* of 0.997 [g/m³] with V_{∞} (66.8 m/s) and the area of the particle inlet.

The surface of the profile was defined as a smooth no slip wall and a heat transfer at 0°C. The wall interaction was again equation dependent, but the velocity restitution coefficients were set to 0.

To start the calculation more effectively, the initialisation parameters for velocity, temperature and turbulence were set according to those at the main inlet.

2.4.2.3. Post-processing in CFX-Post 13.0

The post processing of the Lagrangian calculation differed from the Euler calculation while the solver run is defined equally and therefore described for both simulations in section 2.5.

To obtain the collection efficiency for further use, the value of the water wall mass flow density had to be extracted in the post-processor by using a polyline on the surface of the profile. For further detail see section 3.1.2.

2.4.3. Calculation of the collection efficiency by Euler-Euler

2.4.3.1. Fluid properties

In the basic settings air was defined as a continuous fluid with the material properties of air at 25°C. Water was the dispersed fluid with a mean diameter of 40 µm. The reference pressure for both fluids was 101325 Pa. The turbulence for air was modelled using the Shear Stress Transport with an automatic wall function. For air a Dispersed Phase Zero Equation was chosen. The fluid particle model for interphase transfer was selected. Drag was calculated by a Schiller Naumann coefficient.

2.4.3.2. Boundary conditions

Very important settings in this calculation are the boundary conditions for the profile surface. The surface of the aerofoil was chosen to be an outlet with a degassing condition, allowing air to leave the domain. At the main inlet the volume fraction for air was set to 1 while water was 0. As before in the Lagrangian formulation the subsonic flow regime had only one Cartesian velocity component, being U , with 66.8 m/s. The volume fraction of water at the water inlet was calculated

in an extra variable by dividing the mass of air per cubic meter, found in the material library, by the LWC. To increase effectiveness, the domain was initialised using the parameters of the main inlet.

2.4.3.3. Post-processing in CFX-Post 13.0

The calculation of the collection efficiency with the Eulerian approach was performed following an equation found in Silveira et al. (2003, p. 7).

$$\beta = - \frac{\alpha \vec{V} \cdot \vec{n}}{\alpha_{\infty} V_{\infty}} \quad (2.3)$$

The necessary variables were exported from a polyline on the profile surface to a text file. For further detail, see section 3.2.2.

2.5. Running the setup in CFX-Solver Manager 13.0

After the fluid values and the boundary conditions had been set, the CFX-Solver Manager 13.0 was opened to initiate the solver run. For high accuracy the run was executed with double precision and on 4 partitions. Since the solver required a high amount of computing power to solve the up to 500 000 particles of the Lagrangian formulation, the memory allocation factor was set to 3. To be able to compare the computing time of the two approaches this setup was also used for the Eulerian calculations.

3. Results

The results gained from the preceding calculations are summed up and visualized in this section, and selected data will be mentioned to compare the formulations in section 4. Section 3.1 discusses the outcome of the Lagrangian method and section 3.2 will show the results of the Eulerian method.

3.1. *Lagrangian formulation*

The parameters that will be shown in the following include the computing time for calculations with several particle numbers, the number of iterations in the solver manager and the distribution of the water wall mass flow density along the profile surface.

3.1.1. Solver run

Since the comparison of the two approaches concerning computing time is fairly difficult, because it is hard to determine how many representative particles were used in the Eulerian solution, several runs with the Lagrangian model were executed. As shown in Table 3.1 the computing time increases with the number of particles, while the iterations, needed to reach convergence, stay constant. This is because the flow field is computed independently from the particles and therefore does not require more iteration to be calculated. Only the trajectory calculation influences the computing time when leaving all other parameters constant.

Table 3.1: Comparison of the solver parameters at different particle numbers.

Number of particles	Total averaged CPU time (4CPUs)	Total wall clock time	Iterations until convergence
50 000	5min 35.04sec	5min 40sec	108
100 000	7min 16.51sec	7min 24sec	108
350 000	15min 44.16sec	16min 6sec	108
400 000	17min 30.55sec	17min 54sec	108
500 000	20min 49.39sec	21min 19sec	108

3.1.2. Water wall mass flow density and volume fraction

The water wall mass flow density represents the mass of water which impinges on the surface per area and time. It is therefore the maximum mass flow multiplied by the collection efficiency. In CFX 13.0 this variable can directly be found in the post processor and after smoothing, it can be applied in an icing code. One can observe, looking at Table 3.2, that with higher particle number the mass of particles impinging per second decreases.

Table 3.2: Comparison of the water wall mass flow density at different particle numbers.

Number of Particles	Max. Water wall mass flow density [kg/m ² /s]
50 000	0.135151
100 000	0.0982918
350 000	0.0842501
400 000	0.0813979
500 000	0.0810907

Fig 3.1 illustrates the distribution of the water volume fraction. The area where the volume fraction of the droplets is near 0 is called ‘shadow area’ and is located mainly behind the point of the highest thickness of the profile.

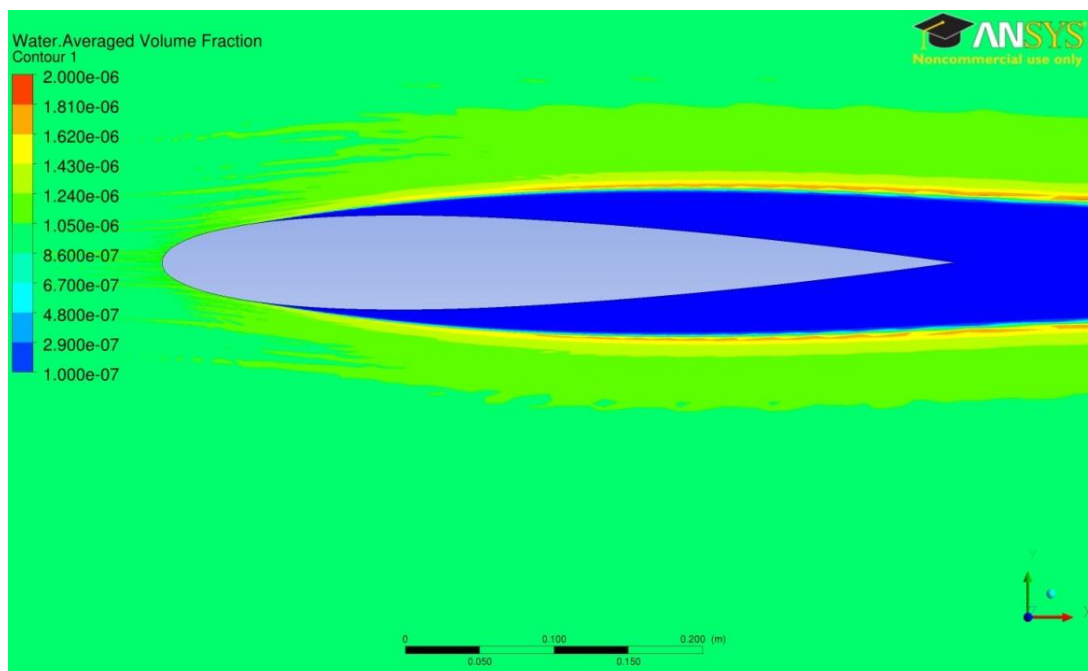


Fig 3.1: Water volume fraction contours for NACA 0012 aerofoil.

The most water can be found in the area of the stagnation point, because the particles are barely deflected by the flow field. Fig 3.2 shows the water wall mass

flow density on the surface of the profile and the impinging limits. As already shown in Table 3.2, the amount of water impinging on the surface is decreasing with increasing particle numbers. An explanation is the amount of real droplets that are represented by one particle package in the simulation. If for example only one particle was computed, the mass flow would be extremely high, because this one particle would represent all droplets. The figures 3.3-3.5 show the distribution of the water wall mass flow density along the profile chord length. The figures also show that the impingement is restricted to a small area of about 0.06 meters. An increased number of particles enhances the smoothness of the water density distribution which reduces the required smoothing iterations later in the icing code.

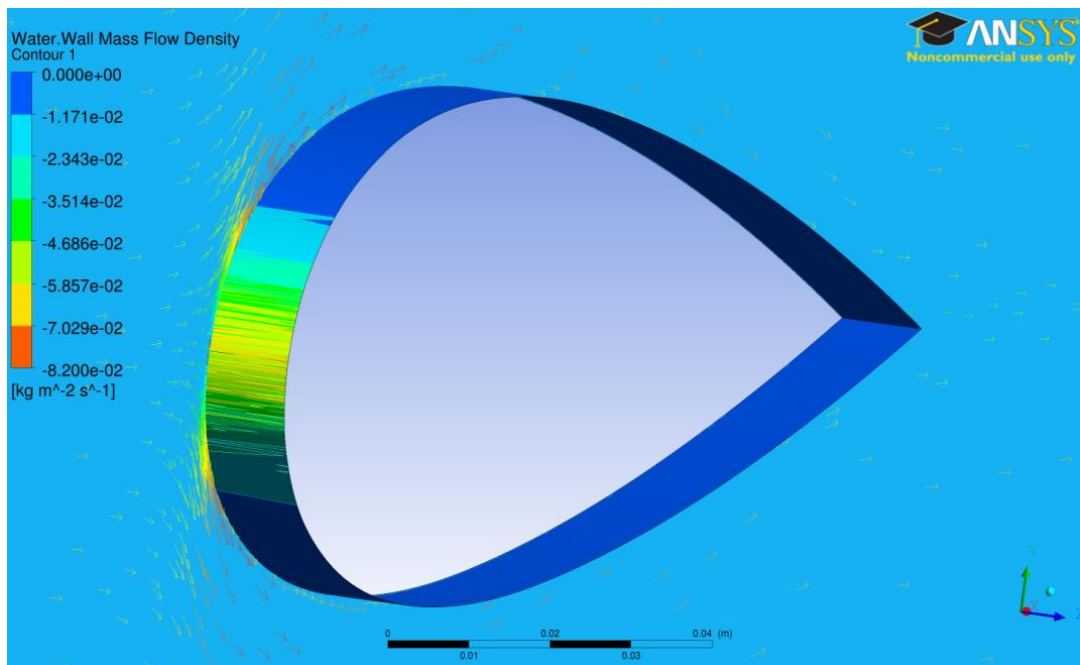


Fig 3.2: Water wall mass flow density on the profile surface.

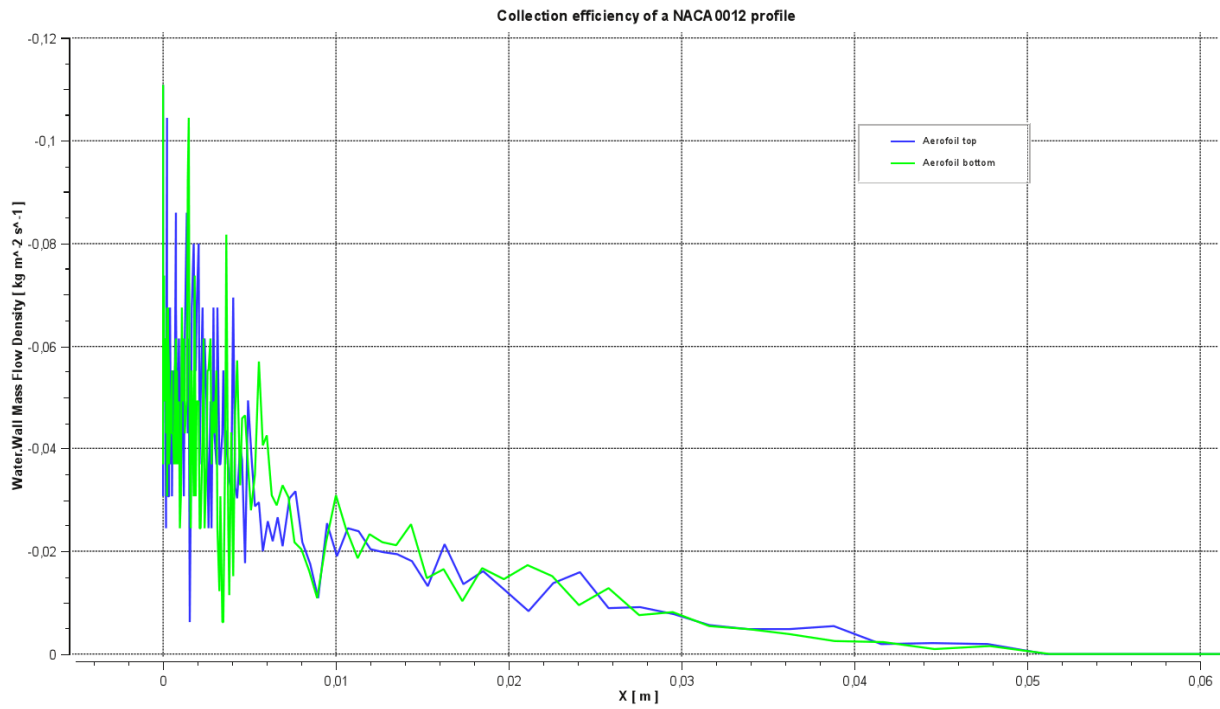


Fig 3.3: Water wall mass flow density of a NACA 0012 profile at 0° angle of attack and 50 000 particles.

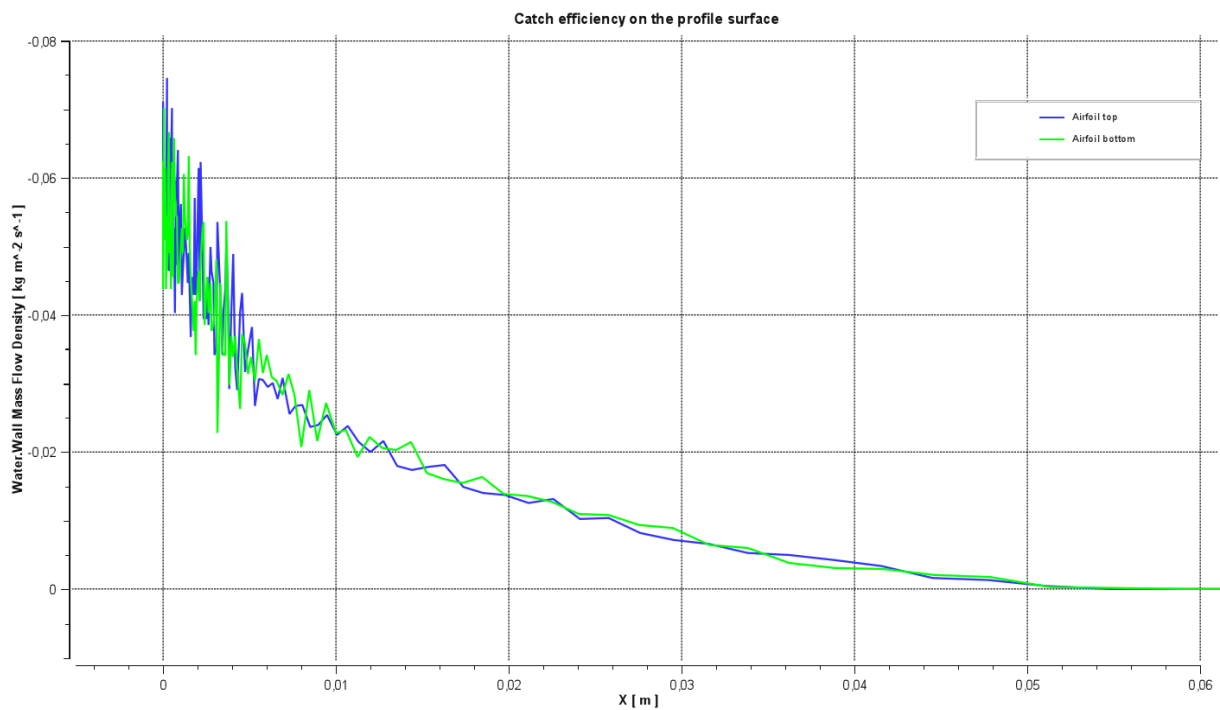


Fig 3.4: Water wall mass flow density at 350 000 particles.

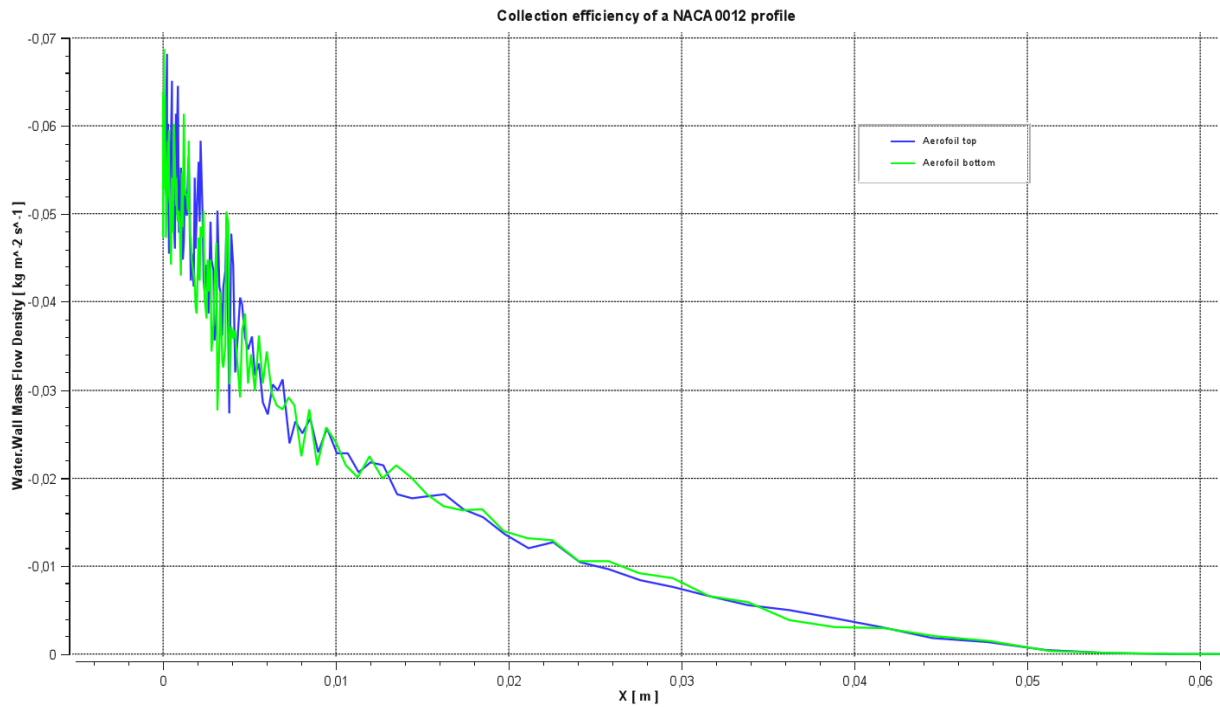


Fig 3.5: Water wall mass flow density at 500 000 particles.

3.1.3. Collection efficiency with Lagrange

As Table 3.2 shows, there is not much of a difference of the water wall mass flow density between 350,000 and 500,000 particles. The collection efficiency was therefore only calculated for the case with 350,000 particles. Figures 3.6 and 3.7 show the collection efficiencies for the upper and the lower surface of the NACA 0012 profile as they were calculated with a Matlab script. It can be seen that the trend is, as expected, similar to the water wall mass flow density.

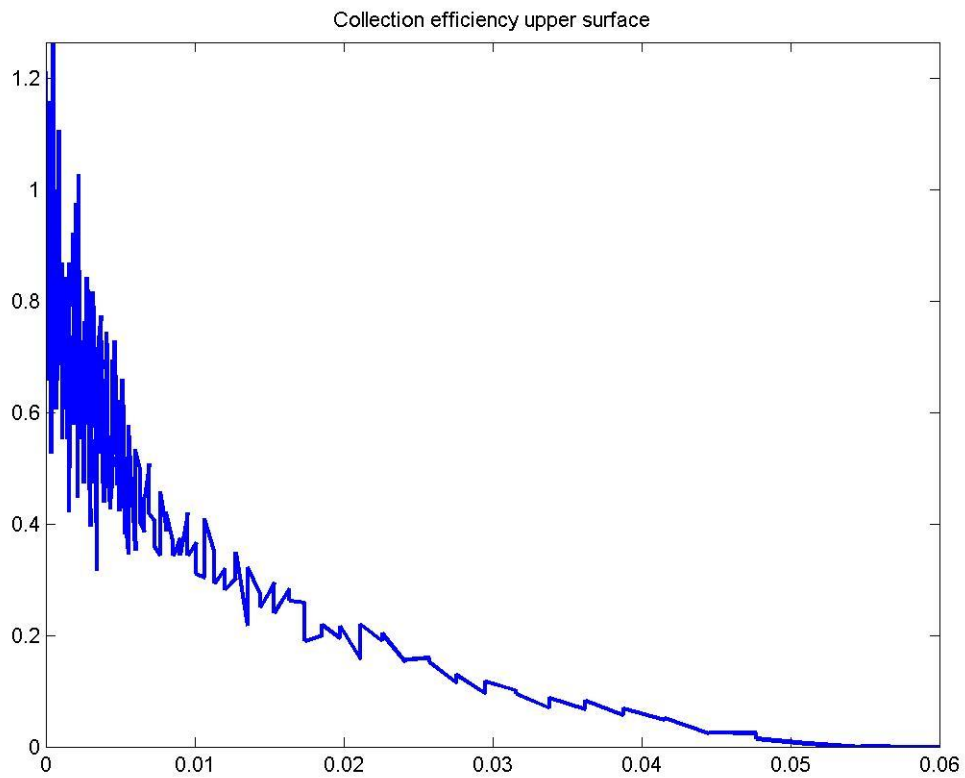


Fig. 3.6: Collection efficiency for the upper surface.

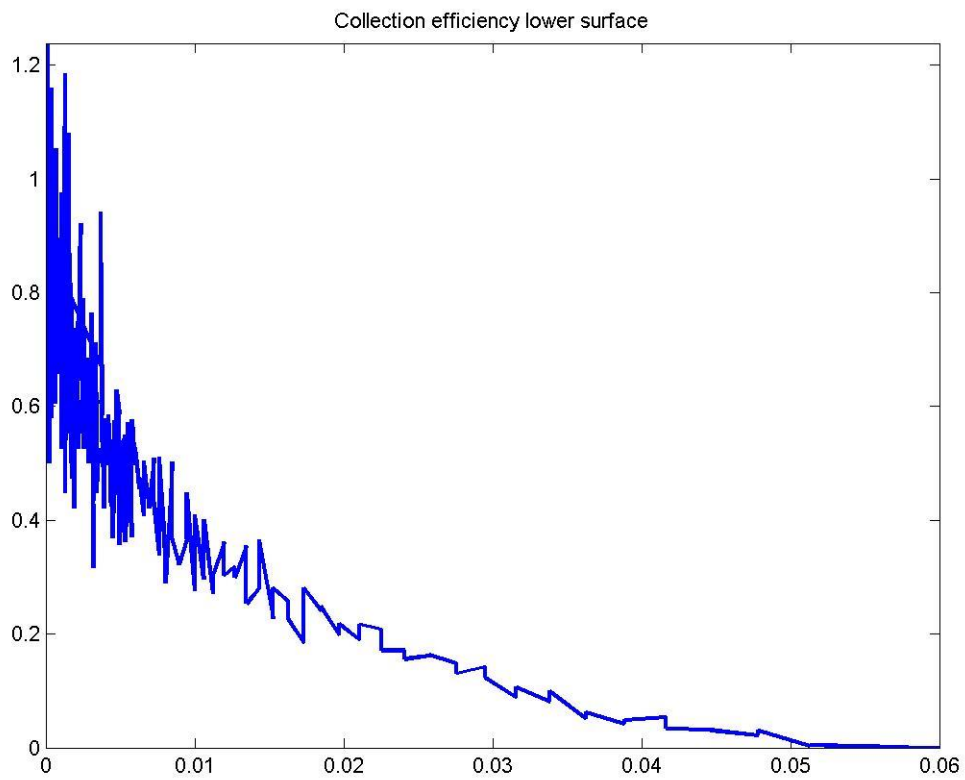


Fig. 3.7: Collection efficiency for the lower surface.

3.2. Eulerian approach

In this chapter the outcome of the Eulerian method is presented including the velocity field and the volume fraction distribution.

3.2.1. Solver run

Since there was no number of particles to define, only one run was executed with the Eulerian approach. The values for comparison with the Lagrangian formulation are shown in Table 3.3. The computing time required to run the calculation was in the area of 100 000 particles of the Lagrangian method and the iterations until convergence were slightly less.

3.2.2. Impingement limits and volume fraction

For the calculation of the collection efficiency the volume fraction, the impingement velocities and the normal vector on the profile surface must be known. In Fig. 3.8 one can see the distribution of the volume fraction and the same 'shadow area' where nearly only air is present.

Table 3.3: Computing time and iterations until convergence for the Eulerian calculation.

Total averaged CPU time (4CPUs)	Total wall clock time	Iterations until convergence
6min 40.76sec	6min 43sec	92

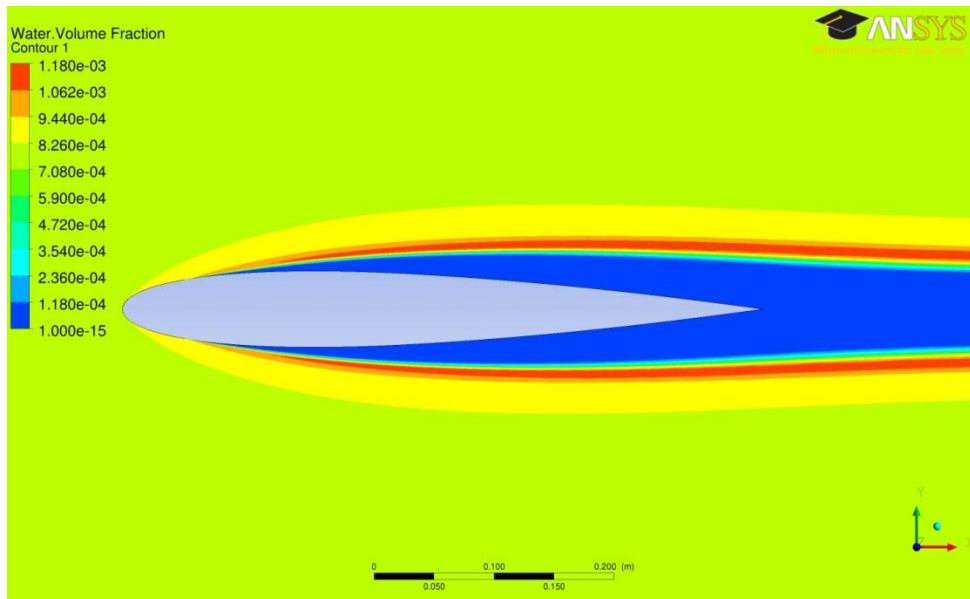


Fig. 3.8: Water volume fraction in the Eulerian calculation.

The area where the water droplets impinge on the surface is restricted to a small region at the front of the profile and is displayed in Fig. 3.9. The vectors in the proximity of the leading edge which seem to enter the aerofoil are the impingement velocities and are shown in Fig. 3.10

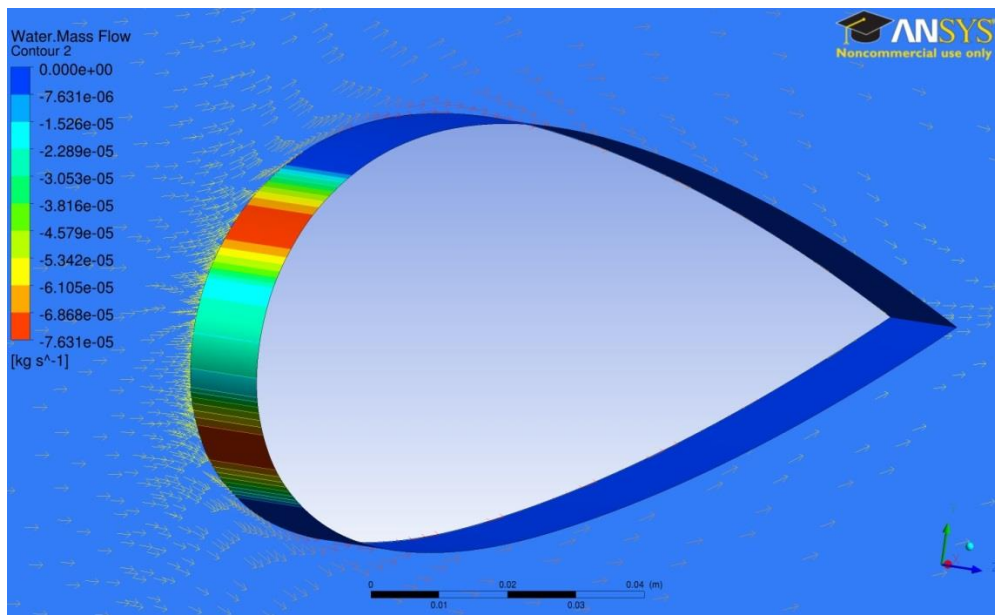


Fig. 3.9: Impingement limits demonstrated with the water mass flow.

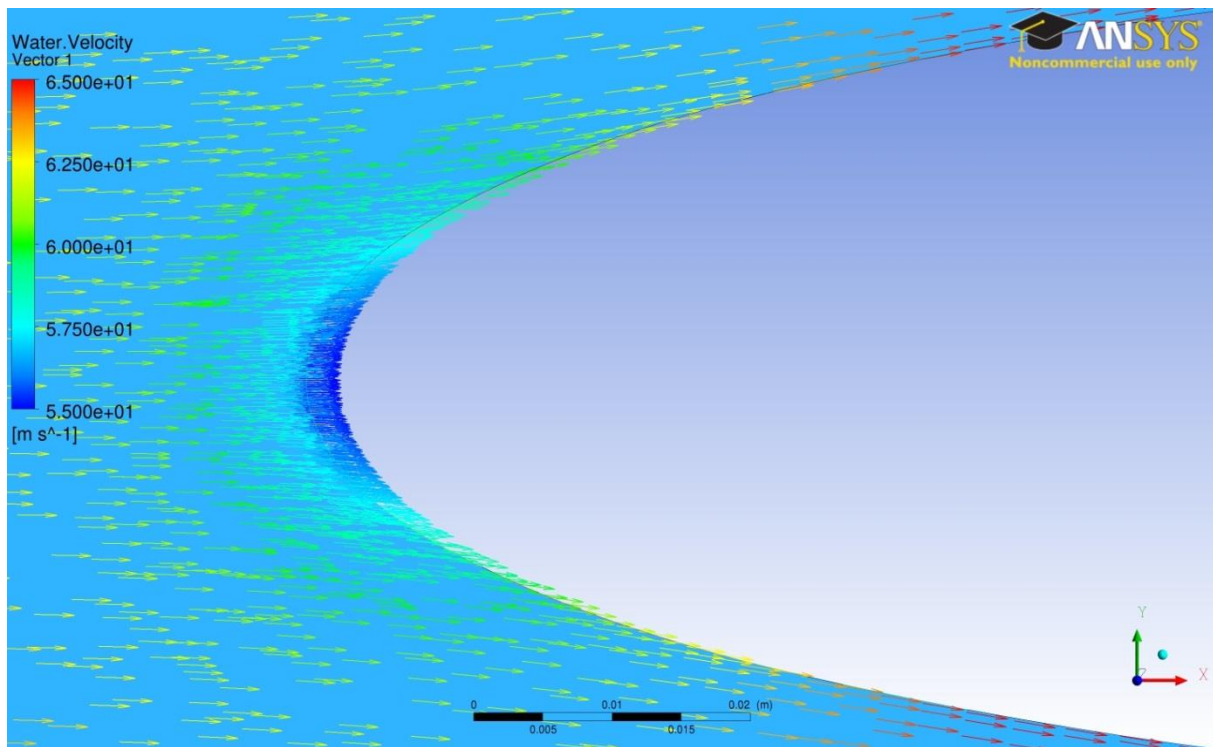


Fig. 3.10: Velocity field of the water droplets.

3.2.3. Collection efficiency with Euler-Euler

As described in section 2.4.3, the collection efficiency for the calculations with the Euler-Euler approach was determined in a Matlab file. The plots of the water volume fraction as it was originally exported from the CFX Post-processor as well as the plots of the calculated collection efficiency are shown in Figures 3.11 to 3.14.

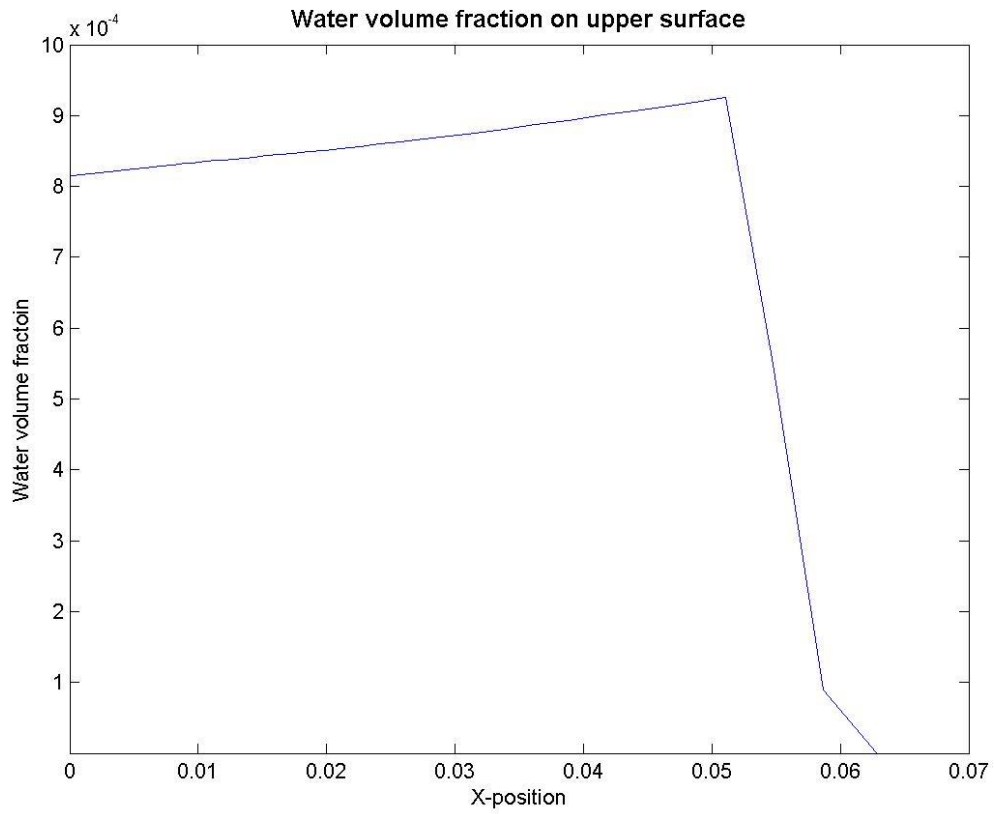


Fig 3.11: Volume fractions over the X-position of the nodes as exported from CFX Post.

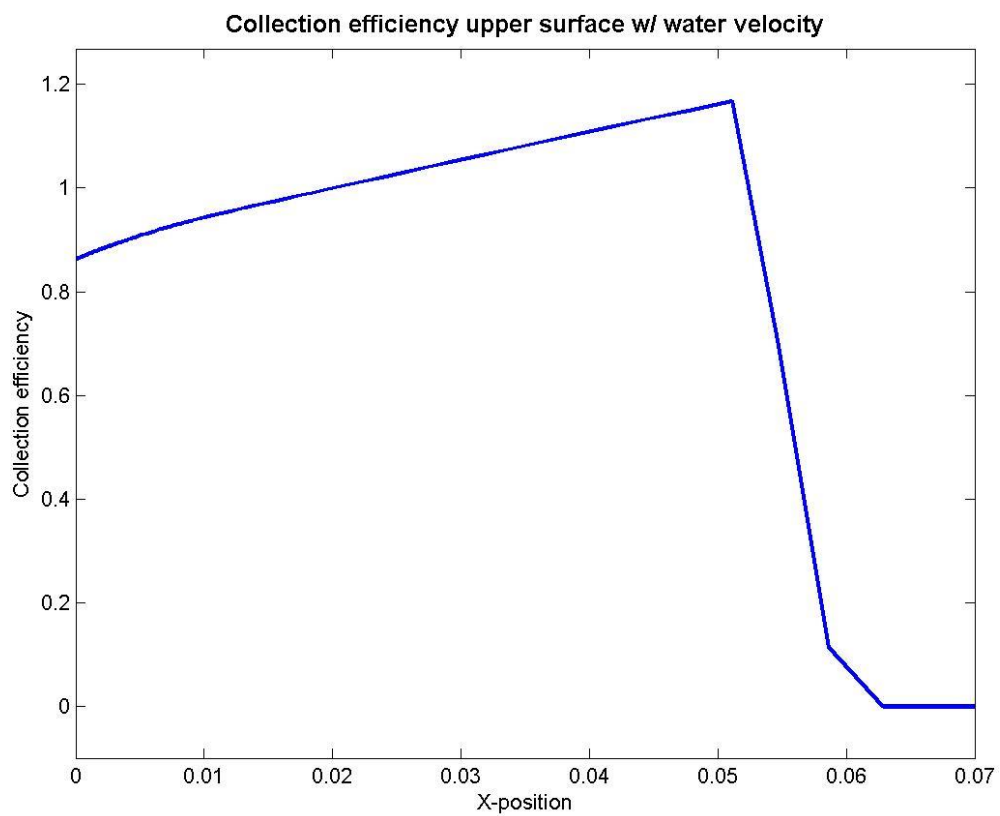


Fig. 3.12: Collection efficiency as calculated for the upper surface.

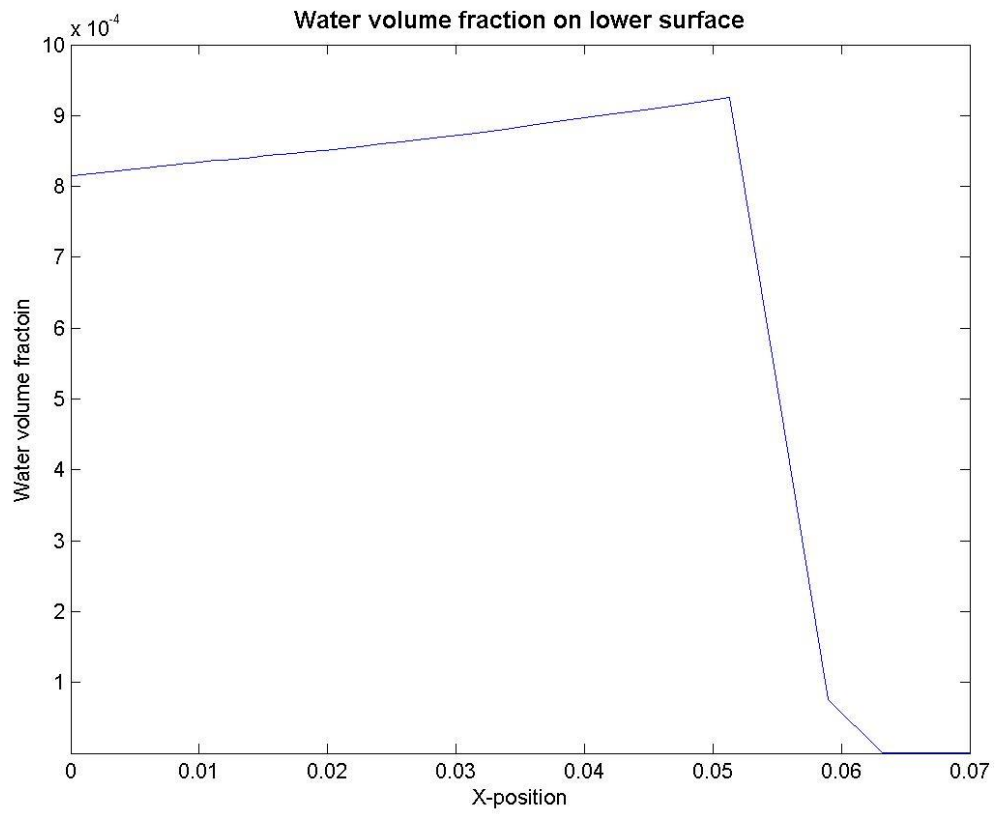


Fig 3.13: Volume fractions over the X-position of the nodes for the lower surface.

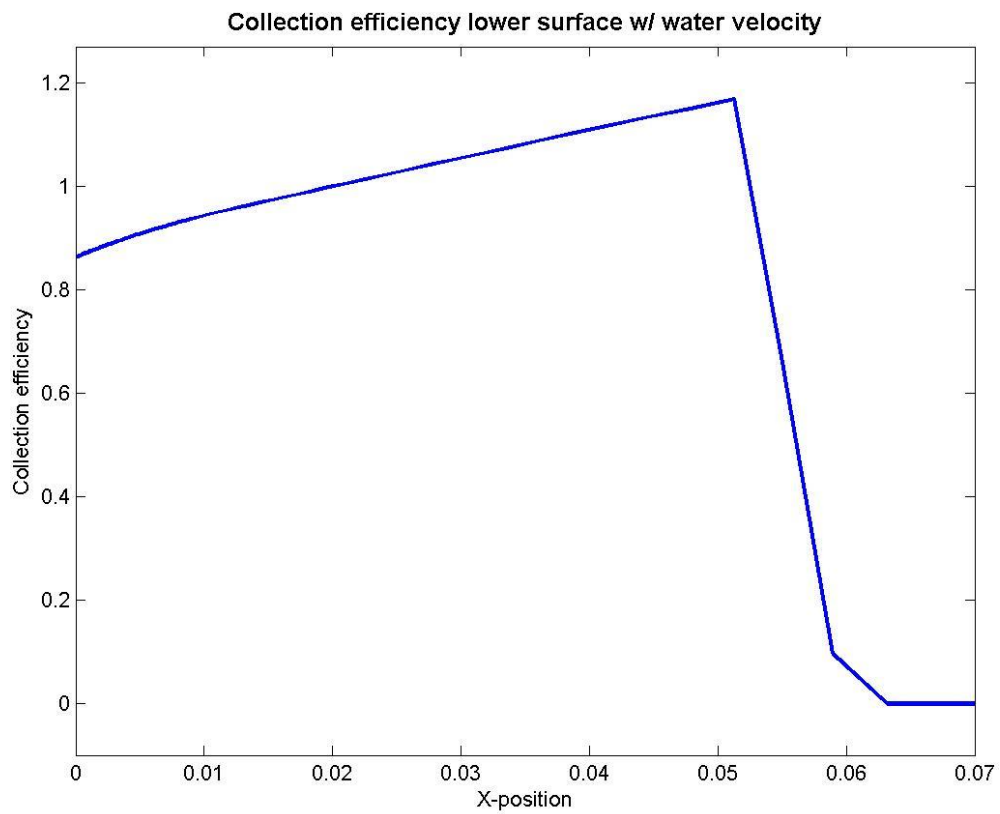


Fig 3.14: Collection efficiency as calculated for the lower surface.

Discussion

3.3. *Discussion of methods*

The two methods applied for the calculations obtained during this project were the Euler-Lagrange formulation and the Euler-Euler approach. Each of these practices has its advantages and drawbacks. The ambition of the following chapter is to show where one or the other method delivers the best results and whether one of them can be favoured over the other.

3.3.1. Lagrangian method

For the simulation of a simple two-dimensional flow field the Lagrangian method is the tool of choice, because it is computationally cost-effective for such situations. The downside of this method is its consumption of computing power, when calculating the trajectories of the hundreds of thousands of particles.

Silveira et al. (2003) found that the Lagrangian formulation may also encounter problems, as the interpolation of the velocities arises, since the values are only known in the grid nodes. The process of finding the cell, where the droplet is located, can be computationally expensive, when a large grid is used or when a three-dimensional problem needs to be solved (p.3). A big advantage in post-processing is the fact that the water wall mass flow density needs no further calculations and can be directly exported to an Excel-sheet. The sheet is then imported to an icing code, where the gradient of the mass flow is smoothed before the ice shapes can be simulated. If the icing code is not able to smooth the gradients shown in Fig3.6 and 3.7, they have to be prepared in extra software which would require additional time before icing shapes can be predicted.

3.3.2. Euler-Euler approach

The Euler-Euler approach has various advantages over the Lagrangian formulation which are, according to Cao et al. (2008) for example that the impingement limits and the collection efficiency can be obtained directly from the droplets flow field. This especially benefits the studies of ice accretion on multi-

element aerofoils and three-dimensional ice shapes (p. 244). Cao et al. (2008) furthermore find that in the two-dimensional icing problem, rime ice can be effectively simulated by an Eulerian method and the impingement limits are more convenient than the ones of the traditional Lagrangian method (p. 248). The calculation of the collection efficiency in this thesis shows that it cannot directly be exported from the droplets flow field, but must be calculated using equation (2.3). This is rather difficult, because there are several variables that have to be extracted from the post-processor. As shown in section 3.2, calculating the collection efficiency and plotting it in Matlab was not properly achieved in this study. The reason for that is that the necessary normal vector n appearing in formula 2.3 could not be extracted from the calculations. It needs to be found out how this can be done. It has to be furthermore stated, that the value called “water volume fraction” might be the water mass fraction. Calculations have shown that a volume fraction of water of $8.3e-4$ would mean that 41% of the mass inside a cubic meter in the free stream would be contributed by water. Concerning the computing time the Eulerian approach is much faster and can deliver good results in a shorter time. The boundary conditions are somewhat more complex to define, but the effort pays off.

3.4. Discussion of results

It has to be stated that a numerical simulation will only be as good as the method and the user behind it. Experiments have shown that today’s best icing codes reach an approximation of around 40% of the real ice shapes. Therefore wind tunnel tests and experimental data are still an important part of the certification process and will not be replaced by CFD analysis in the next couple of years. Furthermore, the simulations mentioned earlier are only valid for droplets with a small diameter. The effect of bigger droplets on the airflow was not taken into account and may change the results tremendously. What can be said is that the outcome of a numerical simulation can be used as a rough estimation of what will be experienced in a wind tunnel test and therefore accelerate the extraction of meaningful results.

4. Summary and outlook

The problem of icing is a constant factor in the improvement of existing systems and the certification of new aircraft. Even though only a small area on the front section of the wing is affected by ice accretion, the impact on the properties and functionality can be huge. Numerical simulations help to understand the process of icing and improve the development of protection systems implemented today.

In this paper the two main types of approaches that are applied today were presented. The Lagrangian formulation calculates the flow field of the air and afterwards computes the trajectories of the injected droplets to their impinging points on the aerofoil. The Eulerian approach treats the air and the water as two phases which exchange energy mass and heat and uses one grid to solve the values for both phases in one step. As an outcome of both methods the mass of water that impinges on an aerofoil at an angle of 0° was calculated to be used in an icing code.

Section 3 displayed the amount of water that in the end contributes to the formation of ice and the area of the wing, where this water is present. Due to the fact that the small droplets are deflected by the airflow around the aerofoil, the highest water density was encountered in the area around the stagnation point. In this region the droplets almost maintain a straight line trajectory.

To assign the described methods to their most convenient field of application, the advantages and drawbacks of each method were pointed out. It turned out that the Eulerian approach has many advantages over the Lagrangian formulation, which is widely used today.

It is expected that future projects will extend the described models to a three-dimensional wing section and form the foundation for a 3-D anti-icing simulation. Further research may be necessary to improve the accuracy of the applied procedures. New technologies developed in the future may one day be capable of simulating ice accretion with a high precision. This may lead to a situation where test flights are only needed to prove the calculations as it is today with strength calculations.

References

ANSYS, Inc., CFX 13.0, Southpointe 275 Technology Drive Canonsburg, PA, November 2010.

Cao, Y., Zhang, Q., Sheridan, J., “Numerical simulation of rime ice accretion on an aerofoil using an Eulerian method,” *The Aeronautical Journal*, May 2008, pp. 243-249.

Fluent Inc., “Fluent 6.1 Documentation,” [web site], URL: <http://jullio.pe.kr/fluent6.1/help/index.htm> [cited 11 November 2013].

Gent, R. W., Dart, N. P., Cansdale, J. T., “Aircraft icing”, *Phil. Trans. R. Soc. Lond.*, Farnborough, Hampshire, UK, 2000, pp. 2873-2909.

Paraschivoiu, I., Saeed, F., „Aircraft icing“, John Wiley & Sons, Inc., Toronto, Canada, 2004, pp. 25-28.

Prosperetti, A., Tryggvason, G., *Computational Methods for Multiphase Flow*, Cambridge University Press, New York, United States of America, 2007, Chaps. 1, 4.

Silveira, R. A., Maliska, C.R., Estivam, D. A., Mendes, R., “Evaluation of collection efficiency methods for icing analysis,” *Proceedings of COBEM 2003*, Vol. 1810, ABCM, Sao Paulo, Spain, November 2003, pp. 10.

The MaThWorks, Inc., MATLAB 7.12.0(R2011a), 2011

Appendix

```
%Matlab script for the calculation of the collection efficiency with data
%of the Euler-Lagrange method. Required documents: 350k_top.csv,
%350k_bottom.csv
clc
clear all
close all

%1 2 3 4 5 6 7 8 9 10 11
%X Y Z NX NZ V Vu Vv Vw WaterVol Wwmassflow

vinf = 66.8; %m/s %Anströmgeschwindigkeit
LWC = 0.997e-3; %kg/m^3
TOP=csvread('350k_top.csv',6,0); %Einlesen der Werte auf der Oberseite
BOTTOM=csvread('350k_bottom.csv',6,0);
mprhomax = vinf*LWC; %kg/(m^2*s) max. Massenstromdichte

X_t=TOP(:,1);
Y_t=TOP(:,2);
waterwallmassflowd_t=-TOP(:,12); %kg/(m^2*s)
%Sr_t = max(Y_t)-min(Y_t); %m^2 Referenzfläche Oberseite
%sum_t = sum(waterwallmassflowd_t); %Summe der mfdensities auf Oberseite
% mpunkt = beta*vinf*Sr*LWC --> beta = mpunkt/(vinf*Sr*LWC) Fomrel 2.2
%beta_t = waterwallmassflowd_t/(Sr_t*LWC*vinf); %beta nach Formel 2.2
Y_t_max = Y_t(108); %Maximaler Y-Wert bei dem noch Teilchen auftreten
%Per Hand herausgesucht
%Automatisierung zur Bestimmung des Wertes sehr
%wünschenswert

X_b=BOTTOM(:,1);
Y_b=BOTTOM(:,2);
waterwallmassflowd_b=-BOTTOM(:,12); %kg/(m^2*s)
%Sr_b = max(Y_b)-min(Y_b); %m^2 Referenzfläche Unterseite
%sum_b = sum(waterwallmassflowd_b); %mf densities Unterseite
%beta_b = waterwallmassflowd_b/(Sr_b*LWC*vinf); %Wie bei beta_t
Y_b_max = Y_b(108);

Sr_imp = Y_t_max-Y_b_max; %Sref impact (Auftrefffläche der Partikel)
%mf = (sum_t+sum_b)*Sr_imp; %kg/s Mass flow rate auf das Profil
Sr_p = max(Y_t)-min(Y_b); %Sref profil (Sref an max. Profildicke)
%mdotmax = Sr_p*vinf*LWC; %kg/s Maximaler Massenstrom
%waterwallmassflowd_max = mdotmax/Sr_p; %kg/(m^2*s) Maximale waterwmflowd

%mdot_calc = (sum_t+sum_b)*Sr_p; %kg/s
%Versuch, den Massenstrom aus der waterwmflowd
%rückzurechnen

gcoll_eff = Sr_imp/Sr_p; %Globale collection efficiency des Profils
lcoll_eff_t = waterwallmassflowd_t./mprhomax; %Lokale coll. eff Oberseite
lcoll_eff_b = waterwallmassflowd_b./mprhomax;

%
figure(1);
hold on;
plot(X_t,Y_t,'-r','LineWidth',2);
plot(X_b,Y_b,'-b','LineWidth',2);
title('Contour of the NACA 0012 Profile',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
```

```

grid on;
xlabel('Position/Chord Length Ratio x/c');
ylabel('Y-position');
leg=legend('Upper Surface' , 'Lower Surface');
set(leg, 'FontSize',10);
axis([0 max(X_t) min(Y_b) max(Y_t+0.1)]);
figure(12);
plot(X_t,lcoll_eff_t,'b','LineWidth',2);
title('Collection efficiency upper surface');
axis([0 0.06 min(lcoll_eff_t) max(lcoll_eff_t)]);
%figure(13);
% plot(X_t,waterwallmassflowd_t)
% title('Water.wall.mass.flow.density upper surface',...
%       'FontWeight','bold','fontSize',12,'fontName','Arial');
% axis([0 0.06 min(waterwallmassflowd_t) max(waterwallmassflowd_t)]);

figure(21);
plot(X_b,lcoll_eff_b,'b','LineWidth',2);
title('Collection efficiency lower surface');
axis([0 0.06 min(lcoll_eff_b) max(lcoll_eff_b)]);
%figure(22);
% plot(X_b,waterwallmassflowd_b)
% title('Water.wall.mass.flow.density upper surface',...
%       'FontWeight','bold','fontSize',12,'fontName','Arial');
% axis([0 0.06 min(waterwallmassflowd_b) max(waterwallmassflowd_b)]);

saveas(1,'Profile_350k.pdf','pdf');
saveas(12,'Coll_eff_upper_350k.pdf','pdf');
saveas(21,'Coll_eff_lower_350k.pdf','pdf');

saveas(12,'Coll_eff_upper_350k.jpg','jpg');
saveas(21,'Coll_eff_lower_350k.jpg','jpg');

```

```

%Matlab script for the calculation of the collection efficiency with data
%of the Euler-Euler method. Required documents:
EulerEuler_CalcData_top.csv,
%EulerEuler_CalcData_bottom.csv
clc
clear all
close all
digits(18)
%1 2 3 4 5 6 7 8 9
%X Y Z airvsuperf vair airvolfrac watervsuperf vwater watervolfrac

vinf = 66.8; %m/s free stream velocity
LWC = 0.997e-3; %kg/m^3
TOP=csvread('EulerEuler_CalcData_top.csv',6,0);
BOTTOM=csvread('EulerEuler_CalcData_bottom.csv',6,0);
mprhmax = vinf*LWC; %kg/(m^2*s) max. massflowdensity
alphaw_inf = 0.000775875; %free-stream water volume fraction??fragwürdig

%Calculation of the free-stream water volume fraction
rhoair = 1.1835; %kg/m^3 at 25°C and 101325 Pa
rhow = 997.04; %kg/m^3 at 25°C

%how much m^3 do 0.997e-3 kg watter need?
volwater = LWC/rhow; %m^3 and that is the volume fraction of water which
is...
                %defined as Vi/V. Here V is 1m^3 and Vi is the
                %volwater

%1m^3 in the free-stream therefore has:
%volair = 1-vollWC; %m^3 of air, Matlab calculates this as 1.0000....
volfair = 0.9999990004; %m^3 of air

%calculation of the mass fraction based on the volume fraction exported
%from ANSYS CFX, here waterb_volfrac. Example value: 8.305159e-4
example = 0.0008305159; %value taken from waterb_volfrac array 85

mfH2O = (example*rhow)/((1-example)*rhoair+(example*rhow)); %0.4119
%so the mass fraction of water would be 41% if the exported values are
%volume fractions. this seems very unrealistic, maybe export error or
%converting error. calculations will be executed with the data available

%values of upper surface
X_t=TOP(:,1); %X-values of nodes upper surface
Y_t=TOP(:,2); %y-values of nodes upper surface
vt_superf_air=TOP(:,4); %superficial air velocity upper surface
vt_air=TOP(:,5); %air velocity upper surface
airt_volfrac=TOP(:,6); %volume fraction of air on upper surface
vt_superf_water=TOP(:,7); %superficial water velocity on upper surface
vt_water=TOP(:,8); %water velocity on upper surface
watert_volfrac=TOP(:,9); %water volume fraction on upper surface

%collection efficiency according to formula 2.3, p.17:
betat = (watert_volfrac.*vt_water./(vinf*alphaw_inf)); %with water velocity
betat2 = (watert_volfrac.*vt_superf_water./(vinf*alphaw_inf));%with
                %superficial velocity

%values for lower surface following the structure of the upper surface
X_b=BOTTOM(:,1);
Y_b=BOTTOM(:,2);
vb_superf_air=BOTTOM(:,4);

```

```

vb_air=BOTTOM(:,5);
airb_volfrac=BOTTOM(:,6);
vb_superf_water=BOTTOM(:,7);
vb_water=BOTTOM(:,8);
waterb_volfrac=BOTTOM(:,9);

betab = (waterb_volfrac.*vb_water./(vinf*alphaw_inf));
betab2 = (waterb_volfrac.*vb_superf_water./(vinf*alphaw_inf));

%
figure(1);%Contourplot of profile
hold on;
plot(X_b,Y_b,'-r','LineWidth',2);
plot(X_t,Y_t,'-b','LineWidth',2);
title('Contour of the NACA 0012 Profile',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
grid on;
xlabel('X-position');
ylabel('Y-position');
leg=legend('Lower Surface','Upper Surface');
set(leg,'FontSize',10);
axis([0 max(X_b) min(Y_b) max(Y_b+0.1)]);

figure(11); %Exported volume fraction values on upper polyline
plot(X_t,watert_volfrac)
title('Water volume fraction on upper surface',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Water volume fractoin');
axis([0 0.07 min(watert_volfrac) 1e-3]);

figure(12); %Display of collection efficiency with water velocity on
surface
plot(X_t,betat,'b','LineWidth',2);
title('Collection efficiency upper surface w/ water velocity',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Collection efficiency');
axis([0 0.07 min(betat-0.1) max(betat+0.1)]);

figure(13);%Collection efficiency with superficial velocity
plot(X_t,betat2,'b','LineWidth',2);
title('Collection efficiency upper surface w/ superficial-water
vel',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Collection efficiency');
axis([0 0.07 min(betat2)-0.0001 max(betat2)+max(betat2)*0.1]);

%Plots for the lower surface
figure(21);
plot(X_b,waterb_volfrac)
title('Water volume fraction on lower surface',...
'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Water volume fractoin');
axis([0 0.07 min(waterb_volfrac) 1e-3]);

figure(22);%Collection efficiency on lower surface with watervelocity
plot(X_b,betab,'b','LineWidth',2);

```

```

title('Collection efficiency lower surface w/ water velocity',...
      'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Collection efficiency');
axis([0 0.07 min(betab-0.1) max(betab+0.1)]);

figure(23);%Collection efficiency with superficial velocity
plot(X_b,betab2,'b','LineWidth',2);
title('Collection efficiency lower surface w/ superficial-water
vel',...
      'FontWeight','bold','fontSize',12,'fontName','Arial');
xlabel('X-position');
ylabel('Collection efficiency');
axis([0 0.07 min(betab2)-0.0001 max(betab2)+max(betab2)*0.1]);

saveas(11,['exportet_volfrac_upper.pdf'],'pdf');
saveas(21,['exportet_volfrac_lower.pdf'],'pdf');
saveas(12,['coll_eff_upper.pdf'],'pdf');
saveas(22,['coll_eff_lower.pdf'],'pdf');

saveas(11,['exportet_volfrac_upper.jpg'],'jpg');
saveas(21,['exportet_volfrac_lower.jpg'],'jpg');
saveas(12,['coll_eff_upper.jpg'],'jpg');
saveas(22,['coll_eff_lower.jpg'],'jpg');

```