Politecnico di Milano

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Corso di Laurea Magistrale in Ingegneria Aeronautica

A novel simulation framework for in-flight ice accretion and electro-thermal anti-icing including state-of-the-art liquid film models

Relatore
Prof. Alberto Guardone

Correlatori
Ing. Tommaso Bellosta
Ing. Bárbara Arizmendi

Candidati
Alberto Della Noce
Matr.874908

Mariachiara Gallia
Matr.905181

Sommario

L'accurata modellazione dei fenomeni di formazione di ghiaccio e dei dispositivi antighiaccio riveste fondamentale importanza, sia nella progettazione dell'aeromobile e sia nel garantirne la sicurezza in volo. Nel corso degli anni numerosi studi sono stati condotti per sviluppare programmi che permettessero di predire la formazione del ghiaccio sempre più accuratamente. In questo contesto, al Politecnico di Milano venne sviluppato un programma per la simulazione dei fenomeni di accrescimento di ghiaccio: PoliMIce. In questa tesi sono stati studiati gli effetti che uno strato di acqua di spessore non trascurabile ha nel processo di accrescimento di ghiaccio attraverso due modelli differenti: un modello stazionario e uno instazionario basato sulla teoria della lubrificazione. Inoltre è stato studiato e implementato un modello 2D di dispositivo antighiaccio elettro-termico. A causa delle limitazioni date dalla precedente versione di PoliMIce, è stato necessario sviluppare una nuova versione con una nuova struttura dati.

Il nuovo codice e i nuovi modelli per il *film* liquido superficiale sono stati testati effettuando simulazioni di accrescimento 2D in differenti condizioni ambientali, mostrando risultati in buon accordo con i dati ottenuti sperimentalmente. Infine, il modello di dispositivo antighiaccio è stato testato effettuando simulazioni 2D in tre differenti condizioni: *fully evaporative*, *partially evaporative* e *running-wet*. Anche in questo caso i risultati ottenuti sono in buon accordo con i riferimenti, confermando la bontà dell'approccio seguito.

Parole chiave: Ghiaccio, modelli di accrescimento, dispositivi antighiaccio, PoliMIce, modelli per il *film* liquido, SU2, C++

Abstract

The accurate modeling of icing phenomenon and ice protection systems is crucial for both the aircraft design problem and for safety requirements. Several studies have been conducted throughout the years to develop software that is capable of correctly predict the ice accretion phenomena. In this context, a framework for ice accretion simulations have been developed at Politecnico di Milano: PoliMIce. In this work the effect of non-negligible water film thickness during ice accretion is studied through the investigation of two different liquid film models: a steady model and an unsteady model based on the lubrication theory. Moreover, a 2D model for electro-thermal ice protection system has been studied and implemented. To achieve these objectives, a completely new code structure has been developed to overcome limitations of the previous version of PoliMIce.

The new software and the liquid film models have been tested with 2D simulations for different icing conditions, showing good agreement with reference data. Lastly, the IPS model capability has been tested on 2D simulations in three anti-ice condition: fully evaporative, evaporative and running-wet. The results obtained compare fairly well with available reference data, confirming the correctness of the present approach.

Keywords: Icing, ice accretion models, Ice Protecion Systems, PoliMIce, liquid film models, SU2, C++

iv

Ringraziamenti

Non è semplice ringraziare tutti quelli che hanno seguito lo sviluppo di questo lavoro. Le poche righe di ringraziamenti possono risultare banali e scontate ma non saranno mai abbastanza esaustive della mia profonda gratitudine verso coloro che hanno passato questo periodo insieme a me, contribuendo passo dopo passo al raggiungimento del traguardo.

Il primo ringraziamento è rivolto al Prof. *Alberto Guardone*, il quale ha permesso con le sue idee la realizzazione di questa tesi. E' stato un percorso stimolante, non privo di difficoltà, ma che mi ha permesso di imparare moltissimi strumenti e di sviluppare un modo di pensare e affrontare i problemi che mi saranno sicuramente di enorme aiuto nel futuro.

Ringrazio *Mariachiara*, con la quale inaspettatamente abbiamo lavorato fianco a fianco in questo ultimo mese. Il tuo contributo è stato fondamentale per il raggiungimento di questo obiettivo e sono certo che le tue capacità ti permetteranno di spingerti verso altri traguardi, più importanti di questo.

Un semplice grazie non sarà mai abbastanza per esprimere la mia riconoscenza verso *Tommaso*, in quanto soprattutto da amico ha contribuito alla realizzazione di questa tesi. Grazie ai tuoi consigli e alle tue idee ho imparato moltissimo e con il tuo aiuto sono riuscito a superare le difficoltà incontrate. Un grande in bocca al lupo per il futuro, che possa riservarti il meglio sotto ogni punto di vista.

Ringrazio inoltre *Bárbara*, la quale ha fornito un importante contributo allo sviluppo di questo lavoro.

Al di fuori dell'università ci sono moltissime persone che vanno ringraziate. In particolare devo ringraziare *Simone* e *Cristiano*, con i quali ho vissuto un anno bellissimo da coinquilini. Tutti i momenti passati insieme rimarranno sempre un prezioso ricordo che conserverò con estremo piacere. A entrambi

vanno i miei migliori auguri per il futuro, nonostante sia certo che le vostre capacità vi faranno raggiungere traguardi sempre più ambiziosi.

Grazie anche a chi mi ha accompagnato durante tutti questi anni universitari, dal primo all'ultimo giorno e sicuramente anche in futuro. *Paride*, *Antonio*, *Stefano*, *Gabriele*, *Marco* e tutti coloro con cui mi sono divertito in questi anni, condividendo alcuni momenti difficili e moltissimi momenti di allegria.

Grazie a *Lara*, perché la sua positività ha contribuito a rendere migliore questa esperienza universitaria, trovando sempre svago e felicità in ogni situazione.

Grazie a *Luca* e *Alessandro*, con i quali ho vissuto quattro anni bellissimi, divertenti e festosi che sempre ricorderò.

Il grazie più grande va alla mia famiglia. Ai miei genitori e ai miei zii, che più di ogni altro hanno creduto in me, forse addirittura più di me stesso. Grazie al vostro supporto ho raggiunto questo obiettivo, appoggiando le mie decisioni e non facendomi mancare mai nulla. Sicuramente gran parte di questo lavoro è merito vostro, perché solo con i vostri sforzi e con la vostra fiducia posso oggi scrivere queste righe sulla mia tesi di laurea.

Vorrei infine ringraziare chi queste righe non potrà leggerle. Non ci sarebbe spazio e tempo per spiegarne i motivi, quindi solo un immenso grazie anche ai miei nonni.

Alberto

Ringraziamenti

Giunta alla fine di questo percorso è arrivato il momento di ringraziare tutte le persone che mi hanno accompagnato in questi anni.

Innanzitutto vorrei ringraziare il Prof. *Alberto Guardone* per essere stato sempre disponibile, per la sua grande cortesia e per avermi dato la possibilità di svolgere questa tesi in un ambiente sereno e stimolante. Un grazie particolare anche a *Bárbara, Tommaso, Myles* per il loro prezioso aiuto nel risolvere i problemi e le difficoltà incontrate durante questo percorso. Grazie ad *Alberto* per aver condiviso con me questo progetto lavorando fianco a fianco in questo ultimo periodo.

Un ringraziamento speciale a tutti i miei amici e compagni di viaggio che mi hanno accompagnato in questo lungo percorso universitario iniziato cinque anni fa, grazie per aver reso più leggere le interminabili ore di lezione, per aver allietato le infinite attese prima degli esami e aver creato insieme a me mille ricordi in queste aule, senza di voi il Politecnico non sarebbe stato lo stesso. Grazie a *Fede, SamFio, Marco, AleFiz, Pietro, Luca, Massi, Gio, Chiara, Anto, SamGelmi, Paolino, Nic*. Un grazie particolare ad *Andrea* e *Alessio* con cui ho condiviso questo percorso di tesi fin dai primi giorni; sempre disponibili, pronti ad aiutarmi e a prestarmi i loro nodi nel momento del bisogno; abbiamo risolto i mille problemi che abbiamo incontrato nei nostri lavori e abbiamo condiviso giornate a lavorare insieme in aula tesisti.

Un grazie speciale a *Shara* senza la quale non sarei mai riuscita ad arrivare dove sono ora, con te ho condiviso tutte le gioie e i momenti difficili in questi cinque anni. Anche a distanza siamo sempre riuscite ad affrontare insieme tutto, fianco a fianco: gli esami, le levatacce per andare a lezione, gli scioperi di Trenord, le lezioni a giugno in Durando, le vacanze insieme, i lavori di gruppo, la parmigiana di tua mamma, la mitica sessione estiva del secondo

anno... anche quando i percorsi scelti per la magistrale ci hanno portato su strade diverse abbiamo continuato a sostenerci l'una con l'altra. Grazie soprattutto per questi ultimi mesi, per avermi supportata e ascoltata ogni volta in cui avevo bisogno di un aiuto o di una parola di conforto.

Grazie a tutti i miei amici, a quelli del paesello *Vitto, Venni, Nic, Marti, Deni, Marco* per essermi accanto letteralmente da una vita e per esserci stati anche in questo mio momento importante. Agli amici del liceo *Sofi, Eliana, Dani, Marco, Dusi, Mattia* per avermi accompagnato e allietato durante tutto questo percorso, nonostante i nostri mille impegni. Alle mie compagne di squadra, alle "vecchie" mitiche *Gabbianelle* e alle nuove ragazze, la mia valvola di sfogo dopo le lunghe giornate a Milano e durante le sessioni.

Il grazie più grande però va alla mia famiglia, a cui dedico questa tesi. Agli zii, ai cugini, ai nonni, mi avete sempre supportata in questo percorso appoggiando ogni mia scelta e gioendo di ogni mia piccola vittoria. In particolare grazie ai miei genitori senza i quali non avrei avuto la possibilità di raggiungere questo importante traguardo, mi avete sempre incoraggiata a non mollare e a dare il meglio di me. Grazie a mio fratello *Michele* che anche nella lontananza c'è sempre stato. Grazie ai miei nonni, tra una passeggiata, un caffè o una semplice chiacchierata mi avete insegnato che c'è molto altro oltre all'università e allo studio. Infine un grazie speciale alla mia *zia Maurizia*, hai permesso che tutto questo si realizzasse, mi hai sempre aiutato e supportato dandomi la possibilità di vivere esperienze stupende; ci sei sempre stata per me sia nei momenti più importanti che nella quotidianità.

Ultimo ma certamente non meno importante un ringraziamento ad *Enri*, sei stato un compagno di viaggio fondamentale, pazientemente mi hai sempre supportato e sopportato, specialmente in questi ultimi mesi di lavoro. In questi anni siamo cresciuti insieme e non riesco a immaginare come avrei fatto ad arrivare fin qui senza di te.

Mariachiara

Contents

List of Figures

List of Tables

Introduction

The protection of aircraft from the adverse effects of ice accretion has been a crucial design problem since the very early years of flight. During its lifespan, an aircraft experiences a wide variety of environmental conditions, which often change abruptly during normal operation and manufacturers must guarantee safety against all possible atmospheric hazards such as severe rain, snow and ice [26] [29]. If an aircraft flies through wet air such as rain or cloud droplets and if the temperature at the impact point is below freezing temperature, water particles can freeze and form ice on the most exposed zones such as wing leading edges, engine intakes, Pitot tubes, propellers and vents.

Ice formations have significant and crucial effects on performances as they modify the aerodynamic shape of wings as well as rotorcraft blades altering their ability to create lift. Furthermore, the increasing drag caused by ice needs to be compensated by an additional power, the nose is lifted up to maintain altitude, the angle of attack increases and this enhances the wetted surface allowing the aircraft to accumulate additional ice. Wind tunnel tests have shown that snow or ice accumulations no thicker than a piece of coarse sandpaper can reduce lift by 30% and increase drag by 40%. [59] Larger accretions can increase drag by 80% or even more. In extreme cirumstances the aircraft can be so iced that it can stall at much lower angles of attack than normal and it can roll or pitch uncontrollably as ice may affect the aircraft stability changing the weight distribution. Ice can also cause engine failure either by blocking or damaging compressor blades, icing the carburator or by
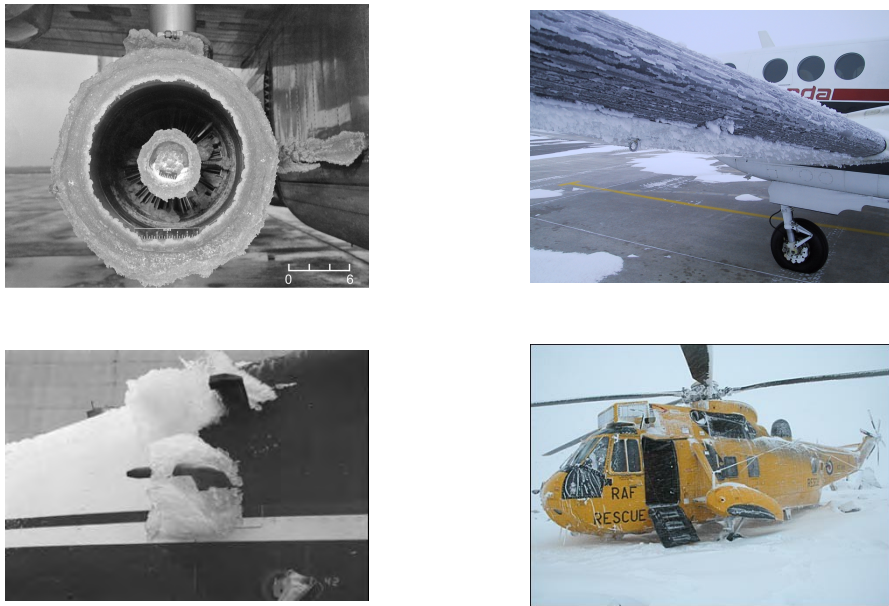
Figure 1.1 Example of ice accretion on an engine, wing, Pitot probe and helicopter. Images taken from [43], [63], [60] and [30]

ice crystal injection [35].

Several accidents are caused by ice formation every year [18] [32]. As shown in figure 1.2, out of a total of 3230 accidents caused by adverse weather conditions during the 1990s, 12% were caused by ice and 27% involved fatalities. For these reasons, since 1940s and 1950s, significant experimental and flight test programmes have been performed to investigate the physics of ice accretion in order to develop systems to protect the aircraft from ice damages. The early works by Hardy [31] and Messinger [36] represent a milestone in numerical analysis of aircraft icing. Messinger developed the first, and still in use, model of ice accretion over a surface. With the advent of the computer in the 1970s, significant progress on theoretical studies and simulations have been achieved. In the following years many research centers such as NASA *Lewis Research Center* in the United States, the *Defence Evaluation and Research Agency* (DERA) in the United Kingdom, the *Office National d'Etudes et de Recherches Aérospatiales* (ONERA) in France and *Centro Italiano Ricerche Aerospaziali* (CIRA) in Italy contributed to the progress on the study of ice physics by conducting many experimental campaigns and by developing codes which can
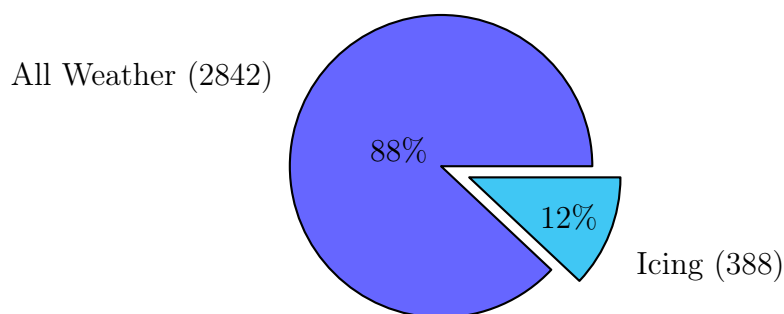
Figure 1.2 Accidents caused by ice respect to other weather conditions (1990-2000) [59]

predict the ice formation. Some codes that have been developed during the years are LEWICE [49] [65], CANICE [11], TRAJICE [25], FENSAP-ICE [38], ICEREMO [44] and POLIMICE [28]. They are nowadays a fundamental tool in aircraft design. As well as to predict ice shapes, they are used to investigate performances degradation of lifting surfaces and to aid the design of anti or de-icing systems.

Icing is a very complex phenomenon that includes multiple physics and different phases of water. Therefore the simulation of in-flight ice accretion requires the computation of the flow around the aircraft. The flow is composed by air and water droplets, thus a multiphase simulation must be performed. Once aerodynamic data and the impact rate of droplets on the surface are known, ice accretion is computed by solving the mass and energy balance for each element of the surface mesh. The mathematical models governing ice accretion will be explained in detail in chapter 2.

1.1 Fundamentals of ice accretion

Aircraft icing occurs when an aircraft flies through a cloud of supercooled water droplets. A cloud consist in small droplets and ice crystals suspended in air. In this condition, the impinging droplets will try to release their latent heat and freeze to form ice. Water droplets in the supercooled state are in a meta-stable equilibrium because they have a temperature below the freezing point but they are still liquid. When they impact on a cold surface the equilibrium is perturbed and they quickly freeze. Due to their relatively

3

small size, cloud droplets may frequently exist in the supercooled state down to $-20\,°C$ and, less frequently, down to $-30\,°C$ and $-35\,°C$ [26].

1.1.1 Ice types

The most common icing situation for aircrafts is the liquid icing, presented in detail in this section. However, ice formation can also occour with freezing rain, ice pellets or wet snow, which will not be considered in this work.
The type of ice is mainly dictated by environmental parameters, as reported in detail in section 1.1.2. Therefore, depending on the external weather conditions, three types of ice can form on an exposed surface: *rime* ice, *glaze* ice and *mixed* ice.

Rime ice

Rime ice forms when droplets are very small and the liquid portion freezes right upon impact before having the time to spread over the surface. The rapid solidification of water causes some bubbles of air to get trapped inside the ice. Therefore the density is usually quite low, around $880\,kg/m^3$ [39]. Figure 1.3 shows a rime formation over a wind tunnel model and its peculiar aspect is clearly seen. It tends to form when ambient conditions are a combinations of low ambient temperature, low speed and a low value of *Liquid Water Content*. Rime ice is less dense than glaze ice but its rough surface leads to a significant degradation of aerodynamic performances. However, due to its brittleness, it can be easily removed by mechanical de-icing systems.

Glaze ice

Glaze ice forms when, after the initial impact, some of the droplets freeze while the remaining portion flow over the surface forming a runback water film. The film can freeze downstream at a different location, producing localized thickening of the ice which may lead to the formation of horns, typical of the glaze ice type. In glaze conditions, the density of accreted ice is larger than rime ice. It forms at temperatures close to the freezing point, when water does not freeze instantaneously, high velocities or high *Liquid Water Content*. As showed in figure 1.4, its appearance is translucent due to the very small amount of trapped air bubbles respect to the rime type and it is
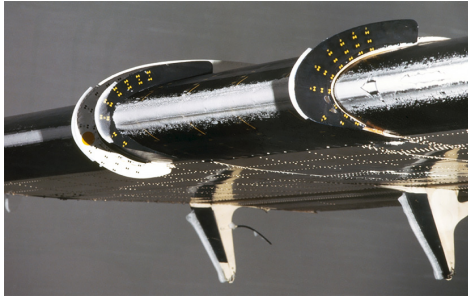
Figure 1.3 Example of rime ice [45]
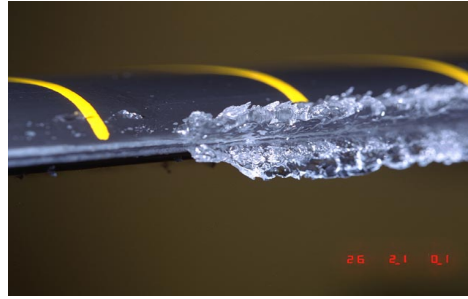
Figure 1.4 Example of glaze ice [27]

very compact, heavy and tenacious. Therefore, is very difficult to be removed by a de-icing device.

Mixed ice

Mixed ice is an intermediate condition between rime and glaze and it forms when droplets vary in size or when they are mixed together with snow. Its characteristics are halfway between glaze and rime ice. It can form when a sudden change in flight or cloud conditions takes place.

1.1.2 Icing relevant parameters

The rate and amount of ice accretion on a surface depends on the *air temperature*, the *geometry*, the *surface roughness*, the *airspeed*, the *liquid water content* (LWC), *the size of the droplets* in the cloud and the *collection efficiency* [26].

Air and surface temperature

The most critical parameter which governs ice accretion is the temperature as it controls heat exchanges between the aircraft and the surrounding air: the lower it is, the faster ice accretion is as convective heat fluxes are larger. The surface temperature is also involved in the evaluation of heat fluxes and it is important as it controls the capability of the surface to accumulate ice and it dictates whether accretion is possible and the rate at which accretion proceeds. Thermal protection systems aim to raise the temperature of the surface above the freezing point in order to prevent the formation of ice.

Surface roughness

The surface roughness has a lot of influence on the final ice shape. It is a very sensitive parameter in the computation of the convective heat transfer coefficient between the surface and air or water. Roughness influences the boundary layer transition point and thus, besides the heat transfer coefficient, plays an important role also in the determination of the skin friction coefficient. Furthermore, surface roughness affects the interaction of the droplets with the surface at the impact point. As a matter of fact, when a droplet impacts it can rebound, splash or stick on the surface and this behaviour is influenced by roughness as well as other parameters [6].

Airspeed

Airspeed influences the amount of droplets which are collected by the surface. High velocities let the surface collect more water and hence the possibility of ice accretion increases. The higher the velocity, the greater is the air volume intercepted in a given time interval and therefore the larger is the mass of impinging water. Moreover, high velocities at mild temperatures cause the impinged water to be driven downstream of the impact point. Lastly, high velocities lead to an higher aerodynamic heating and a subsequent increase in the local temperature causing a reduction of the ice thickness.

Liquid Water Content (LWC)

The *Liquid Water Content* (LWC) is a parameter expressing the grams of water per cubic meter of air. The cloud LWC affects both types of ice and their accretion rate. For large values of LWC, the latent heat which has to be removed to freeze completely the droplets is also large and that leads to the formation of glaze ice. As air temperature decreases, the probability of encountering a large amount of droplets decreases [26], hence the combination of low temperature and low LWC tends to lead to rime ice formation. In general, the mass of water at the surface of a body will increase linearly with the LWC, as explained in detail in chapter 2. Hence, the greater the value of the LWC, the greater is the potential for a large accumulation of ice [26]. The observed value of a cloud LWC depends on the cloud type and their typical values are reported in table 1.1.

Environment	Cloud Type	LWC (g/m^3)
Continental	Stratus	0.28
	Cumulus (clean)	0.26
	Cumulus (polluted)	0.3
	Cumuloninbus (growing)	1 - 3
	Cumuloninbus (dissipating)	1.0 - 1.5
	Fog	0.06
Maritime	Stratus	0.30
	Stratocumulus	0.44
Continental or Maritime	Cirrus ($-25\,°C$)	0.03
	Cirrus ($-50\,°C$)	0.002

Table 1.1   Typical observed cloud LWC [34]

Mean Volume Diameter (MVD)

The *Mean Volume Diameter* (MVD) is the representative value of the size of a droplet. Within a cloud, droplets have different sizes and their diameter can be described with a probability density function. The MVD is the value above and below which half the volume of water is contained. Half the water volume will be in drops larger than the MVD and half the volume in smaller drops [20] [26]. The size of droplets affects significantly the impingement limits and the capability of the surface to collect water. Droplets with large diameter, and therefore large inertia, are less affected by aerodynamic forces and tend to follow a straight trajectory without following the streamlines. On the other hand, droplets with small diameter tend to follow the air streamlines. The MVD is strongly dependent on atmospheric conditions, especially on temperature. As the ambient temperature decreases, droplets tend to minimize the heat exchange with the surrounding air and thus there is the tendency for the diameter to reduce. As a consequence, small droplets are typical in low temperature clouds while large droplets are more common in warmer clouds [26]. Typical values of the MVD are between 15 µm and 40 µm. However, icing encounters with droplets of larger size have been claimed to be the cause of several icing-related accidents. These are called *Supercooled Large Droplets* (SLD) and their MVD spaces between 40 µm and 400 µm. The
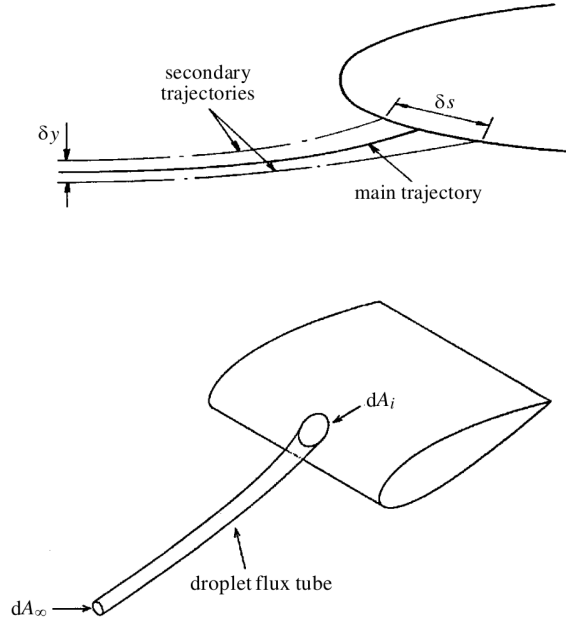
Figure 1.5 Definition of collection efficiency over a 2D and a 3D geometry [26]

condition in which SLD form is known as freezing rain and it is typically associated to temperatures between $-15\,°C$ and $5\,°C$.

Collection efficiency (β)

The *collection efficiency*, usually indicated with β, is one of the most fundamental parameters in ice accretion. It represents a measure of the water quantity collected by the body and it is an index of the wetness of the surface. According to figure 1.5, it is defined as the ratio between the far-field area and the surface area both enclosed by the same droplets trajectories:

$$\beta = \frac{dA_\infty}{dA_i} \tag{1.1}$$

Typical values of β space between 0 on a clean surface and 0.7-0.8 close to stagnation points, where the surface collects more water. As shown in chapter 2, the collection efficiency is linearly proportional to the accretion rate and therefore an high value of β implies an high ice accretion rate. The collection efficiency is strongly influenced by several parameters like the airfoil chord,

the airspeed, the angle od attack and the MVD. Studies on the influence of these parameters on β can be found in [26].

1.2 Ice Protection Systems

In order to guarantee safe flight in icing conditions, commercial and some military aircraft are equipped with *Ice Protection Systems* (IPS). Ice protection systems are deployed into aircraft to delay or to remove any ice accumulation on its surface and its components and therefore to maintain the in-flight performance and safety. The two main operation modes of IPS are the *de-icing* systems and the *anti-icing* systems. The former cyclically operates to remove the ice layer formed after some exposition period. When the system is not operating, the ice builds up on the surface; when it is actuated, the system removes the ice formation. On the other hand, the anti-ice system prevents any ice accretion on surfaces and continuously operates while the aircraft flies under icing conditions. Different technologies can be combined within the same aircraft to accomplish the most convenient and efficient protection. In this section, the most used and developed technologies will be presented.

De-icing boot

This technology has been widely proven as it has been developed in the early 1930s. It consists on a set of flexible rubber-like boots positioned on ice-prone areas which expand and contract cyclically according to the demanded protection, as shown in figure 1.6. Boots operate as de-icing systems as they remove an already formed layer of ice from surfaces where they are located. Following the crack, aerodynamic forces shed the ice which is carried downstream by the flow. Depending on the type of aircraft, boots are inflated with bleed air taken from engines or an additional pneumatic system is included for that purpose. To improve the performance of these systems, fluid freeze point depressant and chemical solutions that reduce the ice adhesion can be sprayed over the rubber boots.
One of the drawbacks of the technology is the degradation of the rubber boots and consequently of the performance caused by interaction with extreme environmental conditions, such as very low temperatures, UV radiation and atmospheric moisture. Therefore boots need frequent maintainance and
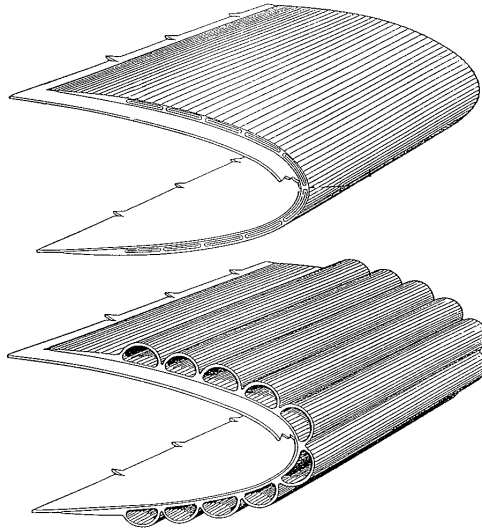
Figure 1.6 Goodrich pneumatic boots [59]

repairs in order to ensure an adequate safety level. Moreover, studies observed that the cyclic process does not remove the entire ice formation [12].

Pneumatic-Thermal protection

This technology is widely used and it has been deployed in numerous commercial, defence and business aircraft. Pneumatic-thermal protections use hot and pressurized air bled from one or more engine compressor stages. The air is supplied to the air cycle machines and exhausted through the *piccolo tube* of the IPS to provide thermal ice protection to surfaces at risk of ice accretion [17]. A schematic view of the system is shown in figure 1.7. The pressurised air moves across the perforations which are directed towards the protected area to heat the outer surface. The pneumatic-thermal protection can be very inefficient because the temperature of the air bled from the engine must be down-regulated for safety reasons before being used by the IPS [17]. Even though the leading edge is protected by the IPS, since the temperature of the surface can be below the freezing point, the water can flow downstream and freeze aft in unprotected areas. This particular situation is called runback ice and it might generate ice formations which cannot be removed by an IPS. To mitigate the risk of runback water, thermal IPS are designed to work in fully
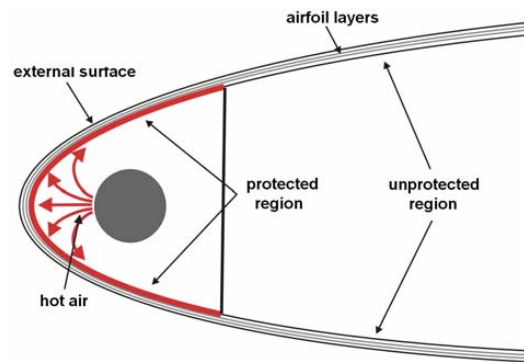
Figure 1.7   Cross section of a piccolo tube [21]

or partially evaporative conditions which require a large energy consumption. This technology can operate both in de-icing or anti-icing modes, depending if it is activated before or after the formation of ice.

Thermo-Electric protection

The thermo-electric protection is one of the most used nowadays and it is the preferred method for rotorcrafts because it presents the best fit to the design, due to the relatively small blades thickness. Furthermore, this protection is installed in many aircraft components which are susceptible to ice formation like Pitot probes. As the pneumatic protection, the thermo-electric heats surfaces at risk of ice accretion but the heat is generated by an electric current going through a resistive component which can be an internal coil wire, a conductive film or an heating rod. The resistance can be bonded on the inner surface of the metallic leading edge or embedded directly into the matrix of a composite material [48]. Also this technology can be used in de-icing or anti-icing mode, depending if it is activated before or after the formation of ice and can be also partially or fully evaporative, depending on the power supplied to the heated elements.

Chemical protection

This technology is mainly used on ground operations before the flight. Foams and chemical substances like glycol are sprayed on surfaces where there could be ice accretion in order to lower the freezing point of water or to reduce the adhesion force between surface and ice which is then shed by aerodynamic

forces. In-flight chemical protections have been deployed also on aircrafts, like in the case of the Hawker 800XP [19]. A micro-perforated panel is mounted into the leading edge and the chemical solution is pumped by means of an electrical pump from an external tank through the holes. This technology is very efficient from the energy point of view but it has many drawbacks which explain why it is not widely used in aviation. The main disadvantage is represented by the dependency on the quantity of fluid stored in the tank and, furthermore, the installation of the tank leads to additional weight that must be carried.

Other technologies

As well as mature IPS, in recent years many technologies are under development which can provide more reliable and convenient protection. The most important is the use of super-hydrophobic materials which can be applied over surfaces where ice could grow in order to reduce the adhesion force of the droplet [5]. This passive technology is very efficient as it does not need energy but it needs a continuous maintainence since the surface has to be always clean from dust and other impurities. Furthermore, ultrasound waves can be used to create stresses on ice causing its shedding [14]. Finally, nanomaterials such as conductive polymer nanocomposites can be used thanks to their tailorable and attractive heating properties. The IPS uses Joule heating of aligned carbon nanotube (CNT) arrays to create highly efficient de-icing and anti-icing of surfaces [15].

1.3   Thesis goal and outline

The work presented in this thesis focuses on the development of a unique, structured and integrated framework for the numerical modeling of ice accretion and anti-icing. This work fits in a project already under way at Politecnico di Milano which aims at the development of an highly modular framework for the simulation of in-flight ice accretion, named PoliMIce [28]. Some limitations emerged from the available version of the PoliMIce code, in particular its structure did not allow an easy implentation of new features, like the modeling of an IPS. Furthermore, its original version had been coded to be compatible only with the open source software OpenFOAM. The fol-

lowing addition of an alternative aerodynamic software, SU2 [23], required an additional interface, which was external to the code. This increased the complexity of the simulation loop. As a consequence of these issues, in this thesis was developed a revamped version of PoliMIce which is compatible with both CFD solvers. The main innovation introduced into the new code is the characterization of four different zones where all the related problems are defined: the body, the multiphase region, the dispersed phase and the flow as can be seen in figure 1.8, reporting a CFD simulation performed by means of SU2 on a NACA0012 at zero angle of attack. The particle tracking has been performed using PoliDrop, developed at Politecnico di Milano [8]. In figure 1.8, the body region is the clean airfoil highlighted in grey. The flow is represented on the background, reporting the variation of the Mach number in different colours. The particles advancing towards the airfoil represent the dispersed phase and the blue region on the leading edge is the multiphase region, that is where ice accretion effectively takes place. The zones are separeted by boundaries which allow the communication among the regions. Furthermore, any boundary can contain an arbitrary number of sub-boundaries allowing to deal with a problem with multiple boundary conditions on the same geometric boundary, like an IPS only on the leading edge of the wing. The ice accretion tools, already present in the previous version, were completely rewritten according to the new structure. Furthermore, the water liquid film flowing above the ice surface in glaze conditions was treated so far as isothermal and with constant thickness. In this thesis, different state-of-the-art models for the water film were studied in detail and implemented in the new version of PoliMIce, in order to estimate its properties and its impact on an ice accretion simulation. Moreover, a numerical model of an electro-thermal IPS was implemented. The reference design used for the IPS can be found in reference [53] and [54].

The C++ language was used to implement the data stuctures and the Unified Modeling Language (UML) was used to visualize the new design of the code and arrange all the classes, methods and attributes according to the wanted structure. The present work is organized as follows.

In chapter 1 the overview of the ice accretion problem in aviation was introduced, a brief description on the physics of ice was given, its most relevant parameters were discussed and the methods which are nowadays available to
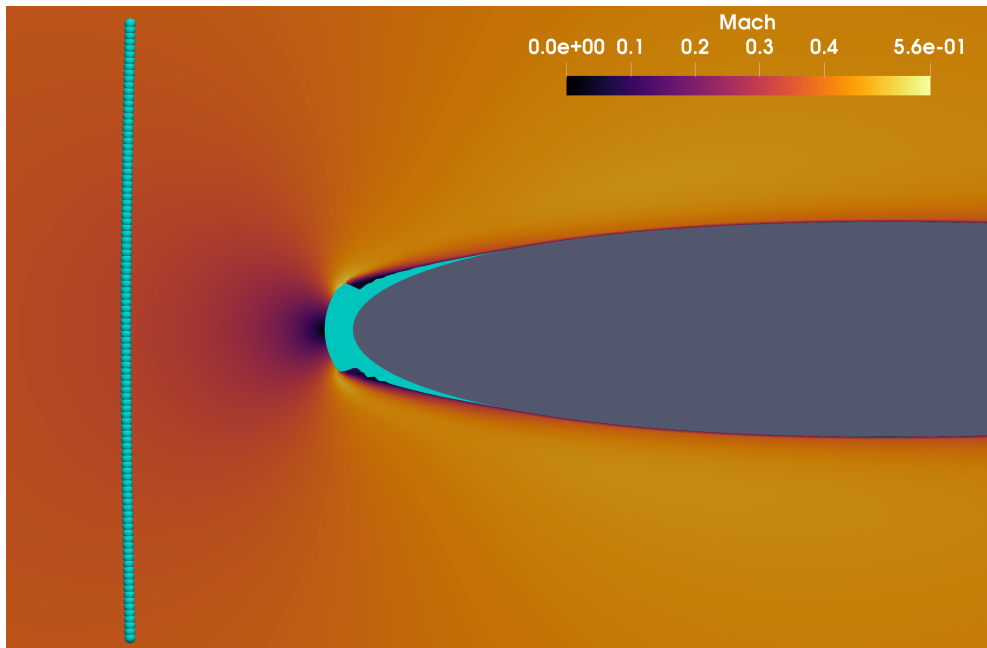
Figure 1.8 CFD simulation around an iced NACA0012 airfoil. The four zones of the problem are clearly visible

protect aircraft components from ice formations were presented.

Chapter 2 contains the ice accretion models, such as the Messinger and the models based on the solution of the 1D Stefan problem. Both the quasi-steady and the unstady models are presented and all of them are implemented in PoliMIce [28].

Chapter 3 presents the liquid film models studied in this work along with their numerical implementation in the PoliMIce code.

Chapter 4 reports the design of the electro-thermal anti-ice system implemented in PoliMIce. Both the geometrical and thermodynamic aspects are presented, including the model of the thermal and viscous boundary layer.

In Chapter 5 the new version of PoliMIce is presented. The code structure is described including all classes and their dependencies.

Chapter 6 contains the results of the test cases used to validate the code. Both the ice accretion and the IPS are tested with different liquid film models and compared to results obtained by experimental campaigns and other software.

Ice accretion models

The study of the behaviour of a substance during its phase change was initially led by B.P. Clapeyron and G. Lamé in 1831 but the first mathematical description of the phenomenon was proposed by J.Stefan around 1890 during his studies in ice formation on the polar sea [58]. This research does not apply only to sea icing but it is valid for every multiphase substance in the freezing process.

The first relevant model of ice accretion used in aeronautics was proposed by Messinger in 1953 and its model is based on the energy balance taking into account all thermal fluxes between ice, water and the surrounding air [36]. The assumptions of isothermal ice layer and insulated airfoil lead to important limitations of the model which result on an underestimation of the ice accretion with respect to experiments.

More recently, the model proposed by Messinger has been revisited by Myers considering the conduction through the ice layer and removing the hypothesis of insulated surface [39]. Myers model is based on an approximate solution of the Stefan problem and not on an a simple energy balance at the surface which leads to a formulation which better accounts for the two different mechanisms associated with rime and glaze ice formation.

In this chapter, the Messinger model is introduced first as it represents the basis of the modern ice accretion models. It will be recalled in chapter 4, as the presented model for an electro-thermal IPS is based on Messinger's approach. Then, the most recent models based on a solution of the Stefan problem are presented. In addition to the aforementioned Myers model, a modified

version of the latter improved by Gori [28] and an unsteady accretion model developed by Parma [29] are presented. All these models are implemented in PoliMIce.

2.1 Messinger model

The first model developed for ice accretion was the one by Messinger in 1953 [36]. It is based on an energy balance over an unheated surface in several icing conditions, taking into account the different heat fluxes involved in the process. Figure 2.1 shows all the thermal fluxes involved in the problem and a brief explanation of the main contributions in models is listed below:

- $\dot{Q}_c$ is the heat transferred by convection per unit time between air and ice or between air and water. It is defined as:

$$\dot{Q}_c = h_{c_*}(T_* - T_{air})A \tag{2.1}$$

  where h_{c_*} is the heat transfer coefficient, $*$ can be either ice i or water H_2O and A is the area through which the heat transfer occours.

- $\dot{Q}_a$ is the thermal power due to the aerodynamic heating and it is caused by friction between air and the exposed surface:

$$\dot{Q}_a = \frac{1}{2}\frac{h_c r A V_\infty^2}{C_{p_{air}}} \tag{2.2}$$

  h_c is the heat transfer coefficient between air and the surface, r is the adiabatic recovery factor which takes into account the effects of air compressibility and $C_{p_{air}}$ is the air specific heat at constant pressure.

- $\dot{Q}_l$ is the latent heat per unit time released or absorbed during freezing or fusion:

$$\dot{Q}_l = \rho_i L_F \frac{\partial B}{\partial t}A \tag{2.3}$$

  where $\frac{\partial B}{\partial t}$ is the variation of ice thickness, ρ_i the density of ice and L_F the latent heat of fusion of ice.

- $\dot{Q}_e$ or $\dot{Q}_s$ is the heat transferred per unit time by evaporation of water or sublimation of ice. It is defined as:

$$\dot{Q}_* = \chi_*[e(T) - e(T_{air})]A \tag{2.4}$$
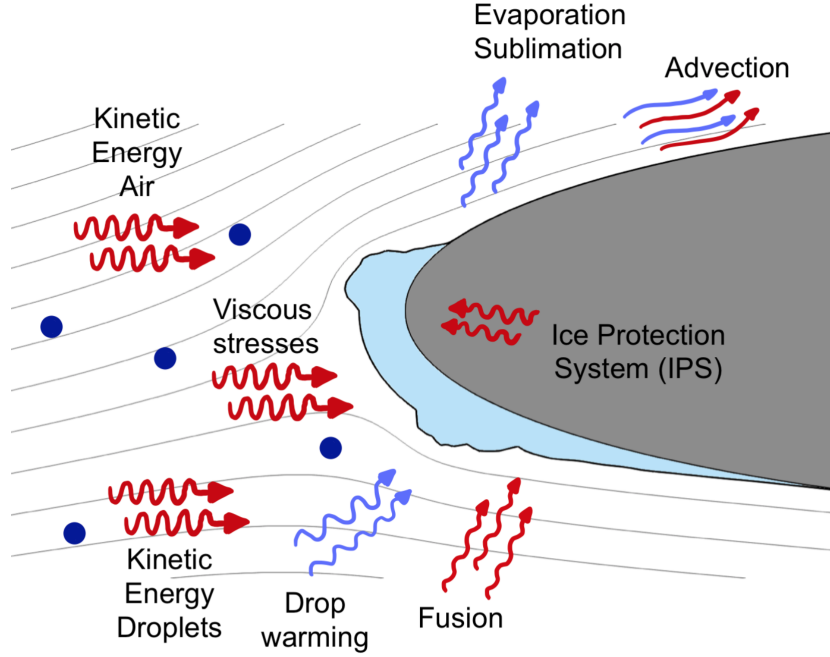
Figure 2.1   Heat fluxes involved in the ice accretion process

where χ_* is the evaporation (e) or sublimation (s) coefficient and $e(T)$ is called evaporation or sublimation function and returns the vapour pressure at a certain temperature. T is the temperature of water in case of evaporation and the temperature of ice for sublimation.

- $\dot{Q}_d$ is the latent heat released per unit time by impinging droplets:

$$\dot{Q}_d = A\,\beta\,LWC\,V_\infty\,C_{p_{air}}\,(T - T_d) \tag{2.5}$$

where T_d is the temperature of the droplet and T the temperature of the surface.

- $\dot{Q}_k$ is the thermal power related to the kinetic energy of impinging droplets and it is defined as:

$$\dot{Q}_k = \frac{1}{2}\,\beta\,LWC\,A\,V_\infty^3 \tag{2.6}$$

where it is assumed that droplets impact the surface with velocity V_∞.

The energy balance to be used depends on the equilibrium surface temperature. Messinger suggested to consider three possible conditions with different

17

freezing fraction F, which is the fraction of water that freezes with respect to the total amount that enters the control volume:

- $T_{wall} < 273.15\,\text{K}$: the temperature is below freezing and therefore all droplets freeze upon impact, $F = 1$. Equation 2.7 shows the corresponding energy balance:

$$\dot{Q}_c + \dot{Q}_s + \dot{Q}_d = \dot{Q}_l + \dot{Q}_a + \dot{Q}_k \qquad (2.7)$$

- $T_{wall} = 273.15\,\text{K}$: the droplets can either freeze or remain liquid, $0 < F < 1$. The corresponding energy balance is reported below:

$$\dot{Q}_c + \dot{Q}_e + \dot{Q}_d = \dot{Q}_l + \dot{Q}_a + \dot{Q}_k \qquad (2.8)$$

- $T_{wall} > 273.15\,\text{K}$: the temperature is above freezing and droplets remain liquid, $F = 0$. Equation 2.9 reports the corresponding energy balance:

$$\dot{Q}_c + \dot{Q}_e + \dot{Q}_d = \dot{Q}_a + \dot{Q}_k \qquad (2.9)$$

In the Messinger model, only the impinging mass is consideried to enter the control volume, therefore the freezing fraction is defined as:

$$F = \frac{\dot{m}_{ice}}{\dot{m}_{imp}} \qquad (2.10)$$

To determine the mass of ice, the starting point is the heat balance where both ice and water are present. From the equation 2.8, a value of the freezing fraction is obtained. If the computed freezing fraction is negative, then F is set to zero and the computation is performed again using equation 2.7. In contrast, if the freezing fraction is greater than one, the value is set to one and equation 2.9 is used.

The main approximation introduced in this model is that the temperature in the ice layer is always equal to the equilibrium temperature. This means that conduction through the ice layer, which is one of the most important contribution in the energy balance, is completely neglected.

Moreover, the freezing fraction is defined as $F = 0$, $F = 1$ or a value between 0 and 1 that remains constant whenever there is both liquid and solid phase. Therefore the freezing fraction moves from a unit to a lower value instantaneously, while in reality F decreases monotonically from $F = 1$ when only ice is present to $F = 0$ when no water freezes [39].

Lastly, this model is a one-dimensional model that does not consider the heat contribution related to the liquid film flowing over the iced surface.

2.2 The Stefan problem

The Stefan problem is a set of four Partial Differential Equations (PDEs) describing the evolution of a single-component two-phase substance during a phase change [29]. Its solution gives the temperature distribution within the two-phase layers and the position of the two boundaries, one between the layers and the other between the substance and air. From the mathematical point of view, this problem belongs to the family of the *moving boundary value problems* as the position of the interface between phases is unknown and depends on time and on the solution itself. Hence, the Stefan problem is a kind of *free boundary problem* which it has to be solved for an unknown solution and an unknown boundary.
Considering the reference system in figure 2.2, the Stefan problem can be formulated as follows:

$$\begin{cases} \dfrac{\partial T}{\partial t} = \dfrac{K_i}{\rho_i C_{p_i}} \dfrac{\partial^2 T}{\partial z^2} \\[2ex] \dfrac{\partial \theta}{\partial t} = \dfrac{K_w}{\rho_w C_{p_w}} \dfrac{\partial^2 \theta}{\partial z^2} \\[2ex] \dot{m}_{fr} + \dot{m}_h = \dot{m}_{in} - \dot{m}_{out} \\[2ex] \Delta \dot{Q} = \dot{Q}_{top} + \dot{Q}_{bottom} \end{cases} \tag{2.11}$$

The first two equations describe the heat exchange within the ice and the water layer. The third equation is the mass conservation which states that the mass entering and leaving the system balances the mass of water which freezes, increasing the ice thickness, and the mass of water which remains liquid, increasing the thickness of the water film. The last equation is called Stefan condition and it is an energy balance related to the heat fluxes involving the phase-change and thus it is evaluated at the interface. This equation states that the latent heat necessary to the phase-change balances the heat exchanged at the ice-water interface.
The Stefan problem does not have a solution in closed form but, introducing some assumptions, the system of PDEs can be transformed in a system of ODEs which admits an analytical solution. This allows to understand the physics of the problem and gives the chance to implement accurate but simple models in ice accretion codes.
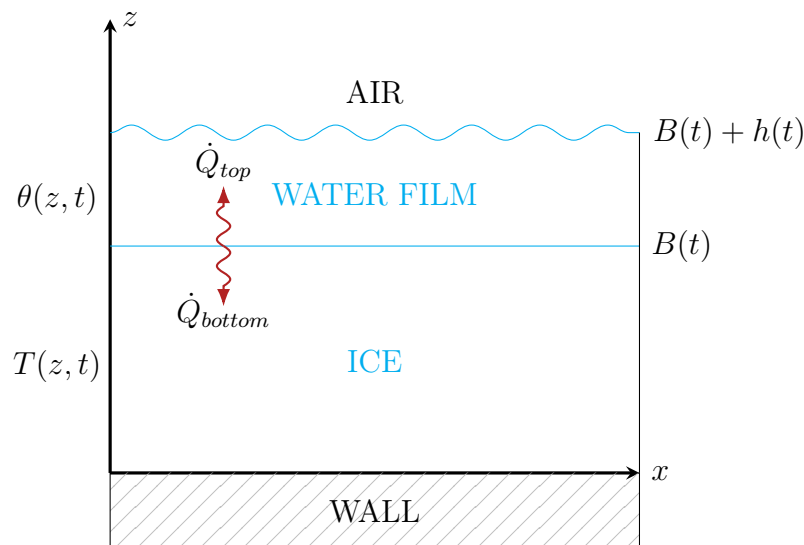
Figure 2.2   Reference system for an ice-water mixture over a surface

2.2.1   Myers model

The first ice accretion model based on the Stefan problem presented in this thesis is the model developed by T. Myers in 2001 [39]. In its work, Myers proposed a simplified solution of the Stefan problem reported in 2.11 introducing the effect of ice thermal conduction and the presence of liquid water. The model is based on several simplifications which allow to reduce the system of PDEs to a decoupled system of ODEs which admits a closed form solution. The main assumptions proposed by Myers are:

- The properties of ice do not vary with temperature, thus the value of its density is fixed. However, it can take two different values whether rime or glaze ice forms.

- The temperature of the surface is fixed and known. This is justified by the great dimension of the body with respect to the ice formation and its high thermal conductivity.

- Droplets are in thermal equilibrium with the surrounding air, which means that their temperatures are equal.

- The phase-change occours at a fixed temperature.

- The water vapour pressure varies linearly with the temperature according to the formula:

$$e(T) \approx -6.803 \times 10^3 + e_0 T \qquad (2.12)$$

where $e_0 = 27.03$ is a constant value.

With the aforementioned simplifications, the Stefan problem reported in 2.11 becomes:

$$
\begin{cases}
\dfrac{\partial T}{\partial t} = \dfrac{K_i}{\rho_i C_{p_i}} \dfrac{\partial^2 T}{\partial z^2} \\[2mm]
\dfrac{\partial \theta}{\partial t} = \dfrac{K_w}{\rho_w C_{p_w}} \dfrac{\partial^2 \theta}{\partial z^2} \\[2mm]
\rho_i \dfrac{\partial B}{\partial t} + \rho_w \dfrac{\partial h}{\partial t} = \beta\, LWC\, V_\infty \\[2mm]
\rho_i L_F \dfrac{\partial B}{\partial t} = K_i \dfrac{\partial T}{\partial z}\bigg|_{B(t)^-} - K_w \dfrac{\partial \theta}{\partial z}\bigg|_{B(t)^+}
\end{cases}
\qquad (2.13)
$$

In the Myers model, the only contribution to the accretion rate is due to the impinging droplets and this leads to the third equation. The energy balance is instead obtained substituting the heat Fourier's law on both sides of the interface.

Diverse boundary conditions can be imposed to the system leading to two different accretion laws, one for the rime and one for the glaze ice. Furthermore, Myers introduced a parameter called *rime limit thickness* which discerns whether rime or glaze ice occurs.

Rime ice

In the case of rime ice, droplets immediatly freeze at the impact with the surface, thus no water film forms. The reference system which has to be considered is shown in figure 2.3. The rime ice thickness is immediately obtained integrating the mass balance which leads to an ice thickness growing linearly with respect to time:

$$B(t) = \int_0^t \frac{\beta\, LWC\, V_\infty}{\rho_{ri}} d\tilde{t} = \frac{\beta\, LWC\, V_\infty}{\rho_{ri}} t \qquad (2.14)$$

According to Myers' hypothesis, the surface is maintained at a constant, fixed and known temperature while at the ice-air interface a Neumann boundary
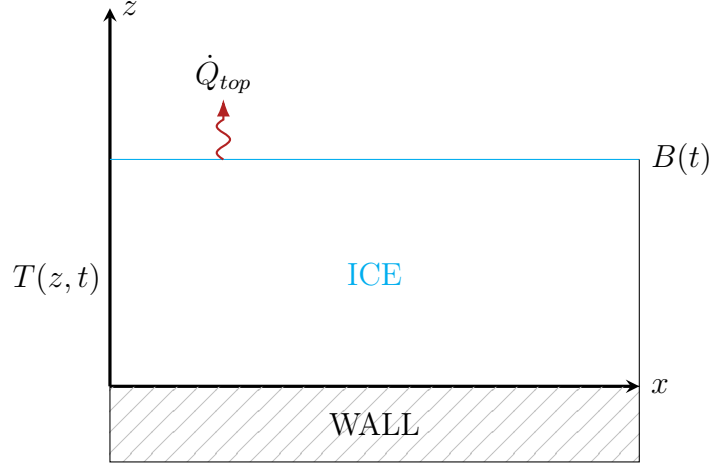
Figure 2.3   Reference system for a rime ice formation over a surface

condition is imposed stating that the heat flux on the ice surface is determined by convection, heat from incoming droplets, sublimation, aerodynamic heating, kinetic energy from incoming droplets and latent heat of freezing:

$$
\begin{cases}
T(0,t) = T_{wall} \\[2mm]
-K_i \dfrac{\partial T}{\partial z}\bigg|_B = \dfrac{\dot{Q}_c + \dot{Q}_d + \dot{Q}_s - \dot{Q}_a - \dot{Q}_k - \dot{Q}_l}{A} = \dfrac{\dot{Q}_{top}}{A}
\end{cases}
\tag{2.15}
$$

The heat diffusion within the ice layer is solved considering a quasi-steady approximation which allows to neglect the time derivative of the temperature, obtaining an ODE. The meaning of this simplification is that the ice growth rate is fairly slower than the heat conduction through the ice layer. However, this hypothesis can be applied only in certain conditions and, according to Myers, only when the ice thickness is lower than approximately 2.4 cm [39]. Under this condition, the heat diffusion equation becomes:

$$
\frac{\partial^2 T}{\partial z^2} \approx 0
\tag{2.16}
$$

The temperature profile can be obtained integrating twice the equation 2.16 and applying the boundary conditions reported in 2.15. This leads to a linear variation of the temperature within the ice layer:

$$
T(z) = T_{wall} + \frac{\dot{Q}_a + \dot{Q}_k + \dot{Q}_l - \dot{Q}_c - \dot{Q}_d - \dot{Q}_s}{K_i A} z = T_{wall} - \frac{\dot{Q}_{top}}{K_i A} z
\tag{2.17}
$$

Furthermore, Myers proposes a modification of the equation 2.17 which includes the effects of ice thickness when it is not negligible:

$$T(z) = T_{wall} + \frac{\dot{Q}_a + \dot{Q}_k + \dot{Q}_l - \dot{Q}_c - \dot{Q}_d - \dot{Q}_s}{A\left[K_i + \frac{B}{T_{wall} - T_{air}}\left(\dot{Q}_c + \dot{Q}_d + \dot{Q}_s\right)\right]}z \qquad (2.18)$$

Glaze ice

In the case of glaze ice, droplets do not freeze instantaneously at the impact point but they remain liquid forming a water film which flows over the clean and iced surface. In this situation, the reference system is the one reported in figure 2.2. Boundary conditions must be splitted in two parts because of the simultaneous presence of water and ice. The phase-change occours at the ice-water interface hence a Dirichelet condition setting the freezing temperature is there imposed. Instead, at the water-air interface, a Neumann boundary condition sets heat fluxes on the surface:

$$ICE : \begin{cases} T(0,t) = T_{wall} \\ T(B,t) = T_{freezing} \end{cases} \qquad (2.19)$$

$$WATER : \begin{cases} T(B,t) = T_{freezing} \\ -K_w \left.\frac{\partial \theta}{\partial z}\right|_{B+h} = \frac{\dot{Q}_c + \dot{Q}_e + \dot{Q}_d - \dot{Q}_a - \dot{Q}_k}{A} \end{cases} \qquad (2.20)$$

Using a similar argument to that considered in the rime ice case, Myers introduced a quasi-steady approximation to solve the heat equations in both layers. For the approximation to remain valid, the liquid film must be very thin, approximately thinner than $3\,\mathrm{mm}$ [39]. However, in aeronautical applications, the liquid film thickness is about $0.1\,\mathrm{mm}$ and thus this hypothesis is usually satisfied. With this simplification, the heat equations become:

$$\frac{\partial^2 T}{\partial z^2} \approx 0 \qquad (2.21)$$

$$\frac{\partial^2 \theta}{\partial z^2} \approx 0 \qquad (2.22)$$

Applying the boundary conditions 2.19 for ice and 2.20 for water and integrating twice, a linear variation of the temperature in both layers is obtained:

$$T(z) = T_{wall} + \frac{T_{freezing} - T_{wall}}{B}z \qquad (2.23)$$

$$\theta(z) = T_{freezing} + \frac{\dot{Q}_a + \dot{Q}_k - \dot{Q}_c - \dot{Q}_e - \dot{Q}_d}{AK_w}(z - B) \qquad (2.24)$$

As in the case of rime ice, Myers proposed a modification of the equation 2.24 including the water film thickness when it is not negligible:

$$\theta(z) = T_{freezing} + \frac{\dot{Q}_a + \dot{Q}_k - \dot{Q}_c - \dot{Q}_e - \dot{Q}_d}{A\left[K_w + \dfrac{h}{T_{freezing} - T_{wall}}(\dot{Q}_c + \dot{Q}_e + \dot{Q}_d)\right]}(z - B) \quad (2.25)$$

Even though the latter is formally linear with z, in the case of small water film thickness the temperature of the film can be considered constant:

$$\theta(z) \approx T_{freezing} \qquad (2.26)$$

Although the water film height is usually small and negligible, the equation 2.25 will be used in the computation when a good estimation of the liquid film height is available, as in the unsteady liquid film model presented in section 3.2 or in the anti-ice model presented in chapter 4. In all the other cases, the water layer will be considered isothermal and with constant height. Due to the simultaneous presence of ice and water, the glaze ice accretion rate is obtained using the Stefan condition:

$$
\begin{aligned}
\frac{\partial B}{\partial t} &= \frac{1}{\rho_{gi}L_F}\left(K_i\frac{\partial T}{\partial z}\bigg|_{B(t)^-} - K_w\frac{\partial \theta}{\partial z}\bigg|_{B(t)^+}\right) \\
&= \frac{1}{\rho_{gi}L_F}\left(K_i\frac{T_{freezing} - T_{wall}}{B} + \frac{\dot{Q}_c + \dot{Q}_e + \dot{Q}_d - \dot{Q}_a - \dot{Q}_k}{A}\right)
\end{aligned}
\qquad (2.27)
$$

In the case of not negligible water film thickness, the equation 2.27 is modified taking into account the effect of the height:

$$\frac{\partial B}{\partial t} = \frac{1}{\rho_{gi}L_f}\left(K_i\frac{T_{freezing}-T_{wall}}{B} - K_w\frac{\dot{Q}_a+\dot{Q}_k-\dot{Q}_c-\dot{Q}_e-\dot{Q}_d}{A\left[K_w+\frac{h}{T_{freezing}-T_{wall}}(\dot{Q}_c+\dot{Q}_e+\dot{Q}_d)\right]}\right)$$
$$(2.28)$$

The equation 2.27 implicitly depends on B itself, thus it can be solved numerically discretizing in time the accretion problem.

Once the ice thickness is known, the third equation of the Stefan problem

can be integrated with respect to time in order to obtain an estimation of the water film thickness:

$$
\begin{aligned}
h(t) &= \int_{t_g}^{t} \left(\frac{\beta \, LWC \, V_\infty}{\rho_w} - \frac{\rho_{gi}}{\rho_w} \frac{\partial B}{\partial t} \right) d\tilde{t} \\
&= \left(\frac{\beta \, LWC \, V_\infty}{\rho_w} \right) (t - t_g) \left(\frac{\rho_{gi}}{\rho_w} \right) (B - B_g)
\end{aligned}
\tag{2.29}
$$

where t_g indicates the time when the glaze ice first appears.

Rime limit thickness

For a smooth transition from rime to glaze, the ice and water thicknesses and growth rates must be continuous. Myers introduced the *rime limit thickness*, which is the height at which glaze ice first appears and allows to discern whether rime or glaze models must be used. For ice thickness lower than the limit, ice follows the rime accretion law while for higher values the phenomenon is described by the glaze model. To determine when this occour, the ice growth rate from the mass balance is substituted into the Stefan condition giving:

$$
L_F \, \beta \, LWC \, V_\infty = K_i \frac{\partial T}{\partial z} \bigg|_{B(t)^-} - K_w \frac{\partial \theta}{\partial z} \bigg|_{B(t)^+}
\tag{2.30}
$$

The heat fluxes are set considering the corresponding solution of the heat equation evaluated at the interface when glaze ice first appears. Named B_g the rime ice thickness, it yields:

$$
L_F \, \beta \, LWC \, V_\infty = K_i \frac{T_{freezing} - T_{wall}}{Bg} + \frac{\dot{Q}_c + \dot{Q}_e + \dot{Q}_d - \dot{Q}_a - \dot{Q}_k}{A}
\tag{2.31}
$$

Solving the last equation for B_g, the rime limit thickness results to be:

$$
B_g = \frac{A K_i (T_{freezing} - T_{wall})}{A \, L_F \, \beta \, LWC \, V_\infty + \dot{Q}_a + \dot{Q}_k - \dot{Q}_c - \dot{Q}_e - \dot{Q}_d}
\tag{2.32}
$$

The equation shows how the rime limit thickness depends on the environmental conditions, the collection efficiency and the freestream velocity. An important feature of this result is that it allows positive, negative and even infinite values for B_g. Therefore it is possibile to discern three cases:

- $B_g < 0$: glaze ice never appears. Rime ice conditions

- $B_g > 0$ and $B < B_g$: rime ice conditions

- $B_g > 0$ and $B > B_g$: glaze ice conditions

A negative value for B_g could occur either because the numerator is lower than zero, that is, the surface is too warm for any ice to grow. Alternatively, the denominator can be negative and this states that there will never be enough energy in the system to produce water and the accretion is pure rime [39].

2.2.2 Modified Myers model

In 2013 Garabelli and Gori, in their master thesis defended at Politecnico di Milano, proposed an improved Myers model based on the same hypothesis but with some modifications regarding both the rime and the glaze ice [24]. In the former case the mass conservation is modified considering a mass flux through neighbouring elements which allows to take into account the possibility that water can flow from a glaze mesh element to a neighbour rime element. In the latter, the temperature profile within the glaze ice is modified observing that the thermal conductivity of the wall is much higher than in the ice. These modifications lead to a new definition of the rime limit thickness which better agrees with experimental observations. Indeed, the Myers model generally provides an overestimation of the experimental ice shapes.

Rime ice

In the original Myers model, a surface rime element could collect water only due to the impinging droplets as reported in the mass balance in the Stefan problem 2.13. In reality it can collect water also from neighbouring glaze elements which release water into it as shown in figure 2.4.
Recalling what reported in the general form of the Stefan problem 2.11, the mass conservation can be written as:

$$\dot{m}_{fr} + \dot{m}_h = \dot{m}_{in} - \dot{m}_{out} \tag{2.33}$$
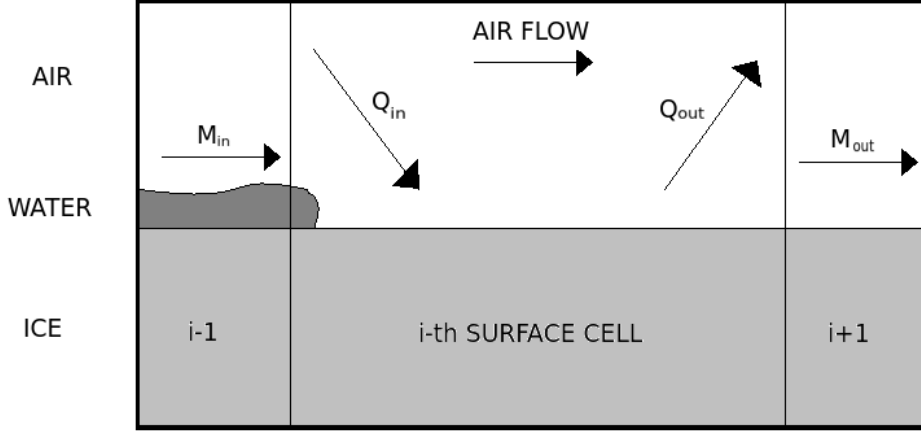
Figure 2.4 Mass balance on a surface element of a 2D accretion problem [42]

The term $\dot{m}_h$ is not present in rime conditions as there is no water flowing over the ice layer. The term $\dot{m}_{in}$ is modified taking into account also the incoming water from neighbouring elements while $\dot{m}_{out}$ involves only the mass flux leaving the surface due to the sublimation of ice as from a rime element there is no outgoing water flux. With these considerations, the mass balance 2.33 becomes:

$$A\rho_{ri}\frac{\partial B}{\partial t} = \dot{m}_{in} + A\,\rho_{ri}\,\beta\,LWC\,V_\infty - \frac{\dot{Q}_s}{L_S} \tag{2.34}$$

From the equation 2.34, the rime accretion rate can be computed dividing by $A\rho_{ri}$:

$$\frac{\partial B}{\partial t} = \frac{1}{\rho_{ri}}\left(\frac{\dot{m}_{in}}{A} + \beta\,LWC\,V_\infty - \frac{\dot{Q}_s}{AL_S}\right) \tag{2.35}$$

The rime ice thickness is obtained integrating in time the equation 2.35 which leads again to a thickness growing linearly in time:

$$
\begin{aligned}
B(t) &= \int_0^t \frac{1}{\rho_{ri}}\left(\frac{\dot{m}_{in}}{A} + \beta\,LWC\,V_\infty - \frac{\dot{Q}_s}{AL_S}\right)d\tilde{t} \\
&= \frac{1}{\rho_{ri}}\left(\frac{\dot{m}_{in}}{A} + \beta\,LWC\,V_\infty - \frac{\dot{Q}_s}{AL_S}\right)t
\end{aligned}
\tag{2.36}
$$

Glaze ice

The result of the Myers model in the case of glaze ice was a linear variation of the temperature within the ice layer, as shown by the equation 2.23. The consequence of the hypothesis of high thermal conductivity of the wall is to have an infinite heat flux at the surface but this is not represented by a linear variation of the temperature within the ice as the heat flux is computed evaluating the derivative of the temperature profile according to the Fourier's law:

$$\left.\frac{\partial T}{\partial z}\right|_0 = \left.\frac{\partial \left(T_{wall} + \dfrac{T_{freezing} - T_{wall}}{B}z\right)}{\partial z}\right|_0 = \frac{T_{freezing} - T_{wall}}{B} \qquad (2.37)$$

Garabelli and Gori, in their master thesis [24], proposed a parabolic variation of the temperature within the ice layer in order to obtain an infinite heat flux at wall. Hence, the temperature varies as:

$$T(z) = a\sqrt{z} + b \qquad (2.38)$$

Applying the boundary conditions 2.19, the temperature profile within the glaze ice is:

$$T(z) = T_{wall} + \frac{(T_{freezing} - T_{wall})}{\sqrt{B}}\sqrt{z} \qquad (2.39)$$

Such as in the Myers model, the glaze ice accretion rate is obtained using the Stefan condition:

$$\begin{aligned}\frac{\partial B}{\partial t} &= \frac{1}{\rho_{gi}L_F}\left(K_i\left.\frac{\partial T}{\partial z}\right|_{B(t)^-} - K_w\left.\frac{\partial \theta}{\partial z}\right|_{B(t)^+}\right)\\ &= \frac{1}{\rho_{gi}L_F}\left(K_i\frac{T_{freezing} - T_{wall}}{2B} + \frac{\dot{Q}_c + \dot{Q}_e + \dot{Q}_d - \dot{Q}_a - \dot{Q}_k}{A}\right)\end{aligned} \qquad (2.40)$$

The same equation can be modified including the effects of water film, when its height is not negligible:

$$\frac{\partial B}{\partial t} = \frac{1}{\rho_{gi}L_f}\left(K_i\frac{T_{freezing}-T_{wall}}{2B} - K_w\frac{\dot{Q}_a+\dot{Q}_k-\dot{Q}_c-\dot{Q}_e-\dot{Q}_d}{A\left[K_w+\frac{h}{T_{freezing}-T_{wall}}(\dot{Q}_c+\dot{Q}_e+\dot{Q}_d)\right]}\right)$$
$$(2.41)$$

It is important to notice that the first term in brackets is the half of the same term of the glaze accretion rate reported in the 2.27, computed by means of the Myers model. This leads to a slower accretion rate which better agrees with experimental results as the Myers model usually provides an overestimation of ice shapes.

However, the assumption of a parabolic temperature profile does not respect the heat diffusion equation in the quasi-steady form. Indeed, the result obtained integrating twice the equation 2.21 is a temperature profile linearly depending on z. Nevertheless the improved Myers model provides better results in predicting ice shapes [24].

Lastly, a new estimate for the liquid film height is computed considering the changes made in new model. Integrating the mass conservation in time and considering all the the mass fluxes, an equation similar to the 2.29 is obtained:

$$h(t) = \left(\frac{\dot{m}_{in} + \dot{m}_{imp} - \dot{m}_{out} - \dot{m}_e}{\rho_w} \right) (t - t_g) \left(\frac{\rho_{gi}}{\rho_w} \right) (B - B_g) \qquad (2.42)$$

The value obtained is just a rough estimation of the liquid film height since, for the hypothesis considered in these models, h is always very small and its effect is supposed to be negligible during the accretion process. Moreover, in the work of Garabelli and Gori [24], in the evaluation of $\dot{m}_{in}$ and $\dot{m}_{out}$ the liquid film height has been considered constant over time and this conflicts with the time dependent expression 2.42. In this thesis, new models for the estimation of the liquid film height have been studied, relaxing the hypothesis of negligible thickness and considering the effects of its motion in the ice accretion process.

Rime limit thickness

The rime limit thickness is computed as in the Myers model, substituting the ice growth rate into the Stefan condition:

$$L_F \left(\frac{\dot{m}_{in}}{A} + \beta\, LWC\, V_\infty - \frac{\dot{Q}_s}{AL_S} \right) = K_i \frac{\partial T}{\partial z}\bigg|_{B(t)^-} - K_w \frac{\partial \theta}{\partial z}\bigg|_{B(t)^+} \qquad (2.43)$$

The heat fluxes are evaluated at the interface when glaze ice first appear and that leads to:

$$L_F \left(\frac{\dot{m}_{in}}{A} + \beta\, LWC\, V_\infty - \frac{\dot{Q}_s}{AL_S} \right) = K_i \frac{T_{freezing} - T_{wall}}{2B_g} + \frac{\dot{Q}_{top}}{A} \qquad (2.44)$$

29

where $\dot{Q}_{top} = \dot{Q}_c + \dot{Q}_e + \dot{Q}_d - \dot{Q}_a - \dot{Q}_k$. Solving the equation 2.44 for B_g, the rime limit thickness results to be:

$$B_g = \frac{AK_i(T_{freezing} - T_{wall})}{2\left[L_F\left(\dot{m}_{in} + A\,\beta\,LWC\,V_\infty - \dfrac{\dot{Q}_s}{L_S}\right)\right] + \dot{Q}_a + \dot{Q}_k - \dot{Q}_c - \dot{Q}_e - \dot{Q}_d}$$

(2.45)

Also the rime limit thickness is smaller compared to the limit obtained in the Myers model. In this way the glaze ice condition is reached earlier and the final ice thickness at the end of the process is less than the one predicted by Myers.

2.2.3 Unsteady accretion model

The ice accretion models previously considered are based on a quasi-steady approximation of the heat diffusion problem. The hypothesis which leads to this simplification is that the ice growth rate is slower than the heat conduction through the ice layer. Although this approximation allows to simplify the problem reducing a system of PDEs to a system of ODEs, in many conditions it cannot be applied such as in very fast ice accretion.
In 2015 Parma, in his master thesis defended at Politecnico di Milano, proposed an exact unsteady accretion model in which the quasi-steady approximation is relaxed [29]. The parabolic PDE governing the heat diffusion admits an exact similarity solution and this requires that boundary conditions satisfy certain properties. In particular a constant value of temperature or an heat flux proportional to $\frac{1}{\sqrt{t}}$ can be imposed [9]. For this reason, the unsteady model developed by Parma holds only in the glaze regime. In rime conditions, the Myers model and its modified version continue to be valid.

Glaze ice

As already discussed, the heat conduction through the water layer can be neglected due to its small thickness. Therefore, the only heat diffusion problem considered is within the ice layer:

$$\frac{\partial T}{\partial t} = \alpha_i \frac{\partial^2 T}{\partial z^2}$$

(2.46)

where $\alpha_i = \frac{K_i}{\rho_i C_{p_i}}$. The problem does not have a reference scale in space and in time, thus the temperature is expected to vary with respect to a variable which is a combination of the independent variables. This means that a variation in space of the solution is also related to a variation in time. The solution of this particular problem can be found using the similarity approach, which means that the initial problem is changed to a problem of only one variable made by the combination of other variables and the solution has the same shape everywhere but with a different scale factor [42]. The similiarity variable can be defined as follows:

$$\xi(z,t) = \frac{z}{\sqrt{t}} \tag{2.47}$$

The equation 2.46 can be expressed as a function of ξ, which is a function of a combination of z and t. The variable Θ, which represents the temperature depending on the similarity variable, is introduced:

$$T(z,t) \longrightarrow \Theta(\xi(z,t)) \tag{2.48}$$

The derivatives are rewritten with respect to ξ:

$$\frac{\partial T}{\partial t} = \frac{\partial T}{\partial \xi}\frac{\partial \xi}{\partial t} = \Theta'(\xi)\left(-\frac{1}{2}\frac{z}{\sqrt{t^3}}\right) = \Theta'(\xi)\left(-\frac{1}{2}\frac{z}{t\sqrt{t}}\right) = -\frac{\xi}{2t}\Theta'(\xi) \tag{2.49}$$

$$\frac{\partial^2 T}{\partial z^2} = \frac{\partial^2 T}{\partial \xi^2}\left(\frac{\partial \xi}{\partial z}\right)^2 + \frac{\partial T}{\partial \xi}\frac{\partial^2 \xi}{\partial z^2} = \frac{1}{t}\Theta''(\xi) \tag{2.50}$$

Substituting the derivatives 2.49 and 2.50 into the heat equation 2.46, the PDE reduces to a second order ODE where Θ is a function of the similarity variable:

$$\Theta''(\xi) + \frac{\xi}{2\alpha_i}\Theta'(\xi) = 0 \tag{2.51}$$

Calling $\Theta'(\xi) = F(\xi)$, the second order ODE reduces to a system of first order ODEs, the solution of which is:

$$F(\xi) = A\exp\left(-\int_0^\xi \frac{s}{2\alpha_i}ds\right) = A\exp\left(-\frac{\xi^2}{4\alpha_i}\right) \tag{2.52}$$

Integrating the equation 2.52 with respect to ξ, the solution of the ODE 2.51 is obtained:

$$\Theta(\xi) = A\int_0^\xi \exp\left(-\frac{s^2}{4\alpha_i}\right)ds + D \tag{2.53}$$
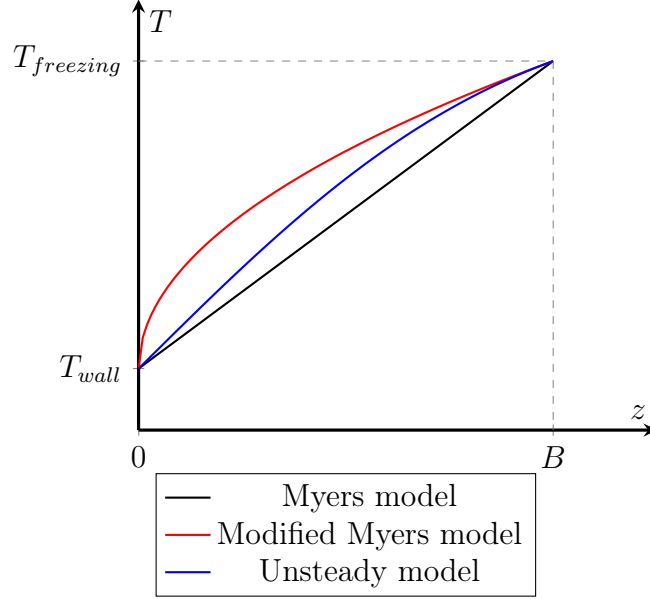
Figure 2.5   Temperature profiles within the glaze ice layer

The solution depends on the Gaussian integral, therefore it can be written as a function of the error function as follows:

$$\Theta(\xi) = A \operatorname{erf}\left(\frac{\xi}{2\sqrt{\alpha_i}}\right) + D \qquad (2.54)$$

The integration constants A and D are determined using the boundary conditions 2.19 for the glaze ice problem. Introducing the parameter $\lambda = \frac{B}{2\sqrt{\alpha_i t}}$ and replacing $\Theta(\xi(z,t))$ with $T(z,t)$, the similarity solution of the 2.46 is finally obtained:

$$T(z,t) = T_{wall} + (T_{freezing} - T_{wall})\frac{\operatorname{erf}\left(\dfrac{z}{2\sqrt{\alpha_i t}}\right)}{\operatorname{erf}(\lambda)} \qquad (2.55)$$

Figure 2.5 shows the temperature profiles within the glaze ice layer computed with the methods discussed so far. The unsteady profile is plotted at a fixed time. The ice accretion rate is computed by means of the Stefan condition, which is here recalled:

$$\rho_i L_F \frac{\partial B}{\partial t} = K_i \frac{\partial T}{\partial z}\bigg|_{B(t)^-} - K_w \frac{\partial \theta}{\partial z}\bigg|_{B(t)^+} \qquad (2.56)$$

32

Using the notation previously introduced, the ice thickness can be expressed as a function of λ. Thus, the ice accretion rate becomes:

$$\frac{\partial B}{\partial t} = \frac{d}{dt}\left(2\sqrt{\alpha_i t}\lambda(t)\right) = 2\sqrt{\alpha_i t}\frac{d\lambda(t)}{dt} + \lambda(t)\sqrt{\frac{\alpha_i}{t}} \tag{2.57}$$

The heat fluxes are set considering the boundary conditions at the water-ice interface. The derivative of the exact temperature profile evaluated at the interface is:

$$
\begin{aligned}
\left.\frac{\partial T}{\partial z}\right|_{B(t)^-} &= \frac{T_{freezing} - T_{wall}}{\operatorname{erf}(\lambda)} \frac{\partial}{\partial z}\left(\frac{2}{\sqrt{\pi}}\int_0^{\frac{z}{2\sqrt{\alpha_i t}}} e^{-s^2}ds\right)\Bigg|_{B(t)^-} \\
&= \frac{T_{freezing} - T_{wall}}{\operatorname{erf}(\lambda)} \frac{2}{\sqrt{\pi}}\exp\left[-\left(\frac{B}{2\sqrt{\alpha_i t}}\right)^2\right]\frac{1}{2\sqrt{\alpha_i t}} \\
&= \frac{T_{freezing} - T_{wall}}{\operatorname{erf}(\lambda)}\frac{\exp(-\lambda^2)}{\sqrt{\pi \alpha_i t}}
\end{aligned}
\tag{2.58}
$$

The heat flux on the water side of the interface is given by an energy balance, as reported in the boundary condition 2.20. The values of the heat flux at the interface 2.58 and 2.20, together with the expression of the ice accretion rate 2.57, are replaced into the Stefan condition giving a non-linear ODE in the unknown $\lambda(t)$:

$$\frac{d\lambda(t)}{dt} + \frac{1}{2}\frac{\lambda(t)}{t} - \frac{K_i(T_{freezing} - T_{wall})}{2\sqrt{\pi}\rho_i L_F \alpha_i}\frac{\exp(-\lambda(t)^2)}{t\operatorname{erf}(\lambda(t))} - \frac{\dot{Q}_{top}}{2A\rho_i L_F \sqrt{\alpha_i t}} = 0 \tag{2.59}$$

The equation 2.59 can be solved numerically discretizing the problem in time and providing an initial solution for $\lambda(t)$. Once $\lambda(t)$ is known, it is possibile to compute the ice thickness.

Rime limit thickness

In order to discern whether rime or glaze ice accrete, the rime limit thickness is again computed imposing the water height to zero in the mass balance and substituting the ice accretion rate into the Stefan condition. Parma obtains a non-linear equation in $\lambda(t)$ which has to be solved iteratively [42]. As this approach is quite slow from the computational point of view, the rime limit thickness proposed by Myers can be used as it is faster and more robust.

Moreover the quasi-steady Myers model is the limit reached by the unsteady model for times tending to infinity [29]:

$$B_g = \frac{AK_i(T_{freezing} - T_{wall})}{\left[L_F\left(\dot{m}_{in} + A\,\beta\,LWC\,V_\infty - \dfrac{\dot{Q}_s}{L_S}\right)\right] + \dot{Q}_a + \dot{Q}_k - \dot{Q}_c - \dot{Q}_e - \dot{Q}_d}$$

(2.60)

This limit has been corrected considering the actual mass balance, as shown in figure 2.4.

Liquid film models for ice accretion

3.1 Steady liquid film models

The steady liquid film model is based on the Stefan problem for phase change [58]. The mass balance of the models presented in chapter 2 has been modified to account for the mass of liquid film flowing in and out of a control volume and considering the height of the film constant over time, that is $\frac{\partial h}{\partial t} = 0$. The system of equations that must be solved is:

$$
\begin{cases}
\dfrac{\partial T}{\partial t} = \dfrac{K_i}{\rho_i c_i} \dfrac{\partial^2 T}{\partial z^2} \\[2mm]
\dfrac{\partial \theta}{\partial t} = \dfrac{K_w}{\rho_w c_w} \dfrac{\partial^2 \theta}{\partial z^2} \\[2mm]
\rho_i A \dfrac{\partial B}{\partial t} = \beta\, LWC\, V_\infty\, A + \dot{m}_{in} - \dot{m}_{out} - \dot{m}_* \\[2mm]
\rho_i L_f \dfrac{\partial B}{\partial t} = K_i \dfrac{\partial T}{\partial z}\bigg|_{B(t)^-} - K_w \dfrac{\partial \theta}{\partial z}\bigg|_{B(t)^+}
\end{cases}
\tag{3.1}
$$

where $\dot{m}_*$ is the mass lost by sublimation ($\dot{m}_s$ for the rime ice case) or evaporation ($\dot{m}_e$ for the glaze ice case). The first, second and fourth equations are solved using the ice accretion models presented in chapter 2. Once these equations are solved, the accretion rate can be computed and $\frac{\partial B}{\partial t}$ is known. The third equation can be rewritten to highlight the unknowns as follows:

$$
\dot{m}_{in} - \dot{m}_{out} = \dot{m}_{ice} - \dot{m}_{imp} + \dot{m}_*
\tag{3.2}
$$

There are two unknowns in this equation, the mass of water entering and exiting the control volume. To close the system is therefore necessary to add one more condition: the mass of water flowing into a control volume must be equal to the mass flowing out from neighbouring control volumes:

$$\dot{m}_{in} = \sum_{i=1}^{n} \dot{m}_{i,out} \qquad (3.3)$$

where n is the number of neighbouring elements. Therefore, the system of equations that must be solved for each control volume is:

$$\begin{cases} \dot{m}_{in} - \dot{m}_{out} = \dot{m}_{ice} - \dot{m}_{imp} + \dot{m}_e \\ \dot{m}_{in} - \sum_{i=1}^{n} \dot{m}_{i,out} = 0 \end{cases} \qquad (3.4)$$

Considering all the control volumes, the size of the system 3.4 is $(2N) \times (2N)$, where N is the number of elements. The main problem in solving this system concerns the connection between control volumes. This means that in order to determine the mass flowing into an element, it is first necessary to determine the outgoing mass from the neighbours.

In the previous version of PoliMIce, this problem has been overcome by considering the mass flowing out from neighbouring control volumes at the previous time step of the ice accretion problem time discretization. The time step was fixed and set to one second. This means that $\dot{m}_{out}$, considered as the mass flux entering in the neighbour control volume, is computed one second before the time at which the calculation are performed. This is not very relevant in slight glaze ice conditions, but becomes more and more relevant as the mass of water increases.

In this work, two possible approaches to improve this approximation have been analyzed. The first approach is a simple iterative approach using fixed point iterations to solve the problem at each time step. This method will be presented in section 3.1.1. The second approach consists in an implicit formulation of the problem to solve an implicit linear system to get the $\dot{m}_{out}$ and $\dot{m}_{in}$ for each control volume with no need of inner iterations. This will be presented in section 3.1.2.

3.1.1 Iterative formulation of the steady liquid film model

For each control volume, the ice accretion problem and liquid film equations must be solved. The first approach that will be analyzed is the iterative approach that solves the system using a simple fixed point iteration that is summarized in the pseudo algorithm 1. The loop reported in the algorithm is performed at each time step of the ice accretion calculation. The iterations are stopped when the error computed as in equation 3.5 is below a tolerance set by the user:

$$\text{Err} = \left\| \frac{\dot{m}_{in}^k - \dot{m}_{in}^{k-1}}{\dot{m}_{in}^{k-1}} \right\|_{L^\infty} \tag{3.5}$$

For each element, the difference between two subsequent iterations of the inflow mass is computed and then normalized by the previous iteration $\dot{m}_{in}$. Then, the greatest of these values, is taken as the error.

Before starting the iterations, the rime limit thickness is evaluated for each element of the iced boundary using the equation 2.32, 2.45 or 2.60, depending on the selected model. To evaluate B_g, all the quantities at previous time step are used. If glaze ice conditions occur, the outflow mass from the control volume is computed using the mass balance 3.2, here recalled for iteration k:

$$\dot{m}_{out}^k = \dot{m}_{in}^{k-1} - \dot{m}_{ice}^k + \dot{m}_{imp}^k - \dot{m}_e^k \tag{3.6}$$

The equation 3.6 represents the total amount of mass flowing out from a control volume. One of the most critical aspect is to determine in which direction this mass will flow. Since the height of the water film is infinitesimal, it can be assumed that the main driving force is the shear stress caused by air [61]. Therefore the direction of the film flow is the same of the shear stress. The next step is to determine whether an element's boundary is an inflow or outflow boundary. To do so, the normal to that boundary is determined. In a 2D problem, thus in a 1D surface mesh, the tangent versor to the element in both directions is computed as the difference between the position of the two nodes of the element:

$$\hat{\boldsymbol{n}}_1 = \frac{\boldsymbol{x}_1 - \boldsymbol{x}_0}{|\boldsymbol{x}_1 - \boldsymbol{x}_0|} \qquad \hat{\boldsymbol{n}}_0 = \frac{\boldsymbol{x}_0 - \boldsymbol{x}_1}{|\boldsymbol{x}_0 - \boldsymbol{x}_1|} = -\hat{\boldsymbol{n}}_1 \tag{3.7}$$

An example of a 1D element and its tangent directions is shown in figure 3.1. In a 3D problem, thus in a 2D surface mesh, the normals to an element
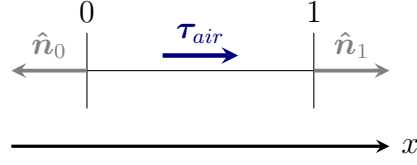
Figure 3.1   Example of a 1D surface element for a 2D ice accretion computation
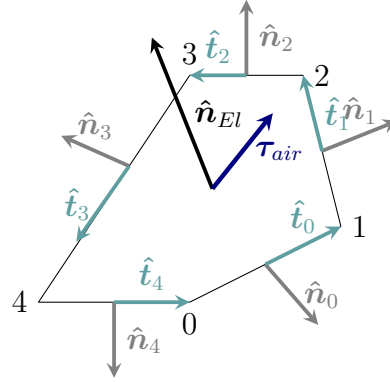
Figure 3.2   Example of a 2D surface element for a 3D ice accretion computation

boundaries are computed as the cross product of the normal to the element surface $\hat{\boldsymbol{n}}_{El}$ and the versor tangent to each boundary $\hat{\boldsymbol{t}}_i$:

$$\hat{\boldsymbol{t}}_i = \frac{\boldsymbol{x}_i - \boldsymbol{x}_{i-1}}{|\boldsymbol{x}_i - \boldsymbol{x}_{i-1}|} \qquad \longrightarrow \qquad \hat{\boldsymbol{n}}_i = \frac{\hat{\boldsymbol{t}}_i \times \hat{\boldsymbol{n}}_{El}}{\left|\hat{\boldsymbol{t}}_i \times \hat{\boldsymbol{n}}_{El}\right|} \qquad (3.8)$$

A visualization of a surface element for a 3D ice accretion calculation is shown in figure 3.2. Once the directions are known, for each boundary the scalar product $\boldsymbol{\tau} \cdot \hat{\boldsymbol{n}}_i$ is computed and the boundaries are labelled as inflow or outflow according to the following convention:

- $\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_i < 0$: inflow boundary

- $\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_i > 0$: outflow boundary

In figure 3.3 an example of this approach is shown for a 3D and 2D ice accretion calculation. The inflow boundaries are highlighted in green while the outflow boundaries are highlighted in blue. From figure 3.3 it is clear that for a 2D calculation there is only one inflow boundary and one outflow boundary for each element. Therefore the whole mass flowing out from a
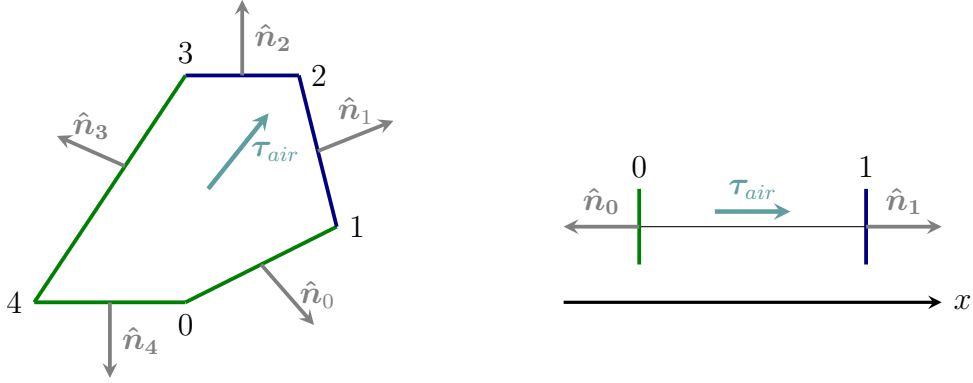
Figure 3.3 Example of inflow and outflow boundaries
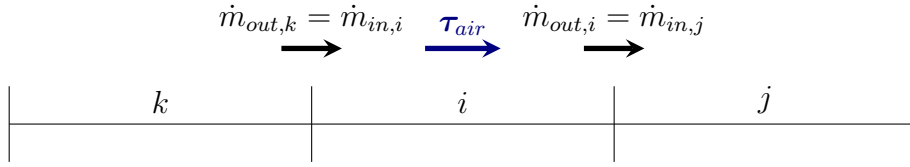
Figure 3.4 Example of water mass flow in 2D ice accretion problem

control volume will enter the next one as its shown in figure 3.4. In a 3D calculation, the problem becomes more complicated: for each surface element there are multiple inflow and outflow boundaries. It must be determined how the outflow mass is divided when flowing to neighbouring elements. The mass flowing out of a control volume is proportional to the length of the boundary and the projection of the shear stress on the normal to that boundary. Considering a linear density of the mass flow on the boundary, the outgoing mass flux through the boundary i of an element can be calculated as:

$$\dot{m}_{out,i} = \frac{(\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_i) L_i}{\sum_{j=1}^{N_b} (\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_j) L_j} \dot{m}_{out,El} \qquad (3.9)$$

where N_b is the number of outflow boundaries and L_i is the length of the ith boundary. From the equation 3.9 it can be shown that the sum of the outflow mass of each boundary is equal to the total mass flowing out of a control volume. Once that $\dot{m}_{out}$ from each boundary of all the elements is known, the mass of water flowing out of an element i from the boundary k can be set to be equal to the mass flowing into the neighbour element through that boundary as it is shown in figure 3.5. The total mass of water flowing into the
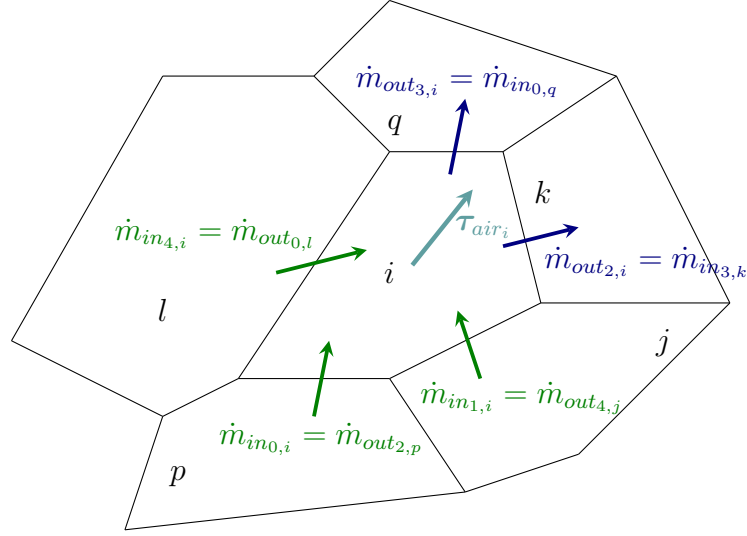
Figure 3.5   Example of water mass flow in a 3D calculation

control volume is then computed as the sum of the $\dot{m}_{out}$ from the element's neighbours, as reported in the equation 3.3.

The computed value of the inflow mass is used to start a new ice calculation at the same time step. The equations remain exactly the same, the only difference is that the used inflow mass is the one that was calculated at the previous iteration. The loop stops when the error 3.5 reaches a value below a tolerance set by the user. Usually a good trade off between accuracy and computational speed is Tol $\approx 10^{-8}$. Once convergence is reached the ice solution is stored and the time is incremented to the next time step.

3.1.2   Implicit formulation of the steady liquid film model

The second approach considered is to directly solve the linear system 3.4 in an implicit way.

First of all, as in the iterative procedure, the rime limit thickness is evaluated for each element of the iced boundary using the equation 2.32, 2.45 or 2.60, depending on the model. To evaluate B_g, quantities at previous time step are used. If glaze ice conditions occur, the system of equations 3.4 must be solved for each element. At first, each boundary of an element is labelled as inflow or outflow as already explained in section 3.1.1 for the iterative procedure. Then, for each element, the system 3.4 is written considering as unknown

Algorithm 1 Pseudo algorithm used to determine ice thickness and mass of water within an ice accretion time step

1: **procedure** DETERMINEICETHICKNESSANDWATERMASS(t^n)
2:     **loop** on each element of the iced boundary
3:         $B^0 \leftarrow$ PreviousTimeStepIceThickness(elementID)
4:         $\dot{m}_{in}^0 \leftarrow$ PreviousTimeStepIceInflowMass(elementID)
5:         B_g(elementID) $\leftarrow$ RimeLimit(elementID)
6:     **end loop**
7:     **while** Err $>$ Tol & $k <$ MaxIter **do**
8:         **loop** on each element of the iced boundary
9:             **if** B^0(elementID) $> B_g$(elementID) & B_g(elementID) > 0 **then**
10:                 B^k(elementID) $\leftarrow$ GlazeIce(elementID, time)
11:                 $\dot{m}_{out}^k = \dot{m}_{in}^{k-1} - \dot{m}_{ice}^k + \dot{m}_{imp}^k - \dot{m}_e^k$
12:             **else**
13:                 B^k(elementID) $\leftarrow$ RimeIce(elementID, time, $\dot{m}_{in}^{k-1}$)
14:                 $\dot{m}_{out}^k = 0$
15:             **end if**
16:         **end loop**
17:         **loop** on each element of the iced boundary
18:             $\dot{m}_{in}^k = \sum_{i=1}^n \dot{m}_{out,i}^k$
19:         **end loop**
20:         Err $= \left\| \frac{\dot{m}_{in}^k - \dot{m}_{in}^{k-1}}{\dot{m}_{in}^{k-1}} \right\|_{L^\infty}$
21:         $k = k + 1$
22:     **end while**
23: **end procedure**

$\dot{m}_{in}$ and $\dot{m}_{out}$. In a 2D ice accretion problem, the construction of the system is reported in the equations 3.10:

$$\begin{cases} \dot{m}_{in,i} - \dot{m}_{out,i} = \dot{m}_{ice,i} - \dot{m}_{imp,i} + \dot{m}_{e,i} \\ \dot{m}_{in,i} - \dot{m}_{out,j} = 0 \end{cases} \tag{3.10}$$

where the index i refers to the considered element and the index j to the neighbour element ahead of the element i, considering the direction of the shear stress $\boldsymbol{\tau}_{air}$ as shown in figure 3.4.

For a 3D problem it must be considered that the mass flowing out of a control volume must be divided within its boundaries following the rule presented in the equation 3.9. For each element the following system must be written:

$$\begin{cases} \dot{m}_{in,i} - \dot{m}_{out,i} = \dot{m}_{ice,i} - \dot{m}_{imp,i} + \dot{m}_{e,i} \\ \dot{m}_{in,i} - \sum_{j=1}^{N_{in,i}} \frac{(\boldsymbol{\tau}_{air,j} \cdot \hat{\boldsymbol{n}}_{l,j})L_{l,j}}{\sum_{k=1}^{N_{out,j}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_k)L_{k,j}} \dot{m}_{out,j} = 0 \end{cases} \tag{3.11}$$

where $N_{in,i}$ refers to the number of inflow boundaries of element i and $N_{out,j}$ to the number of outflow boundaries of element j which share the boundary l with element i. For example, for the element i of figure 3.5, the second equation of the system leads to the following equation:

$$\begin{aligned} \dot{m}_{in,i} &- \frac{(\boldsymbol{\tau}_{air,j} \cdot \hat{\boldsymbol{n}}_{4,j})L_{4,j}}{\sum_{k=1}^{N_{out,j}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_{k,j})L_{k,j}} \dot{m}_{out,j} \\ &- \frac{(\boldsymbol{\tau}_{air,p} \cdot \hat{\boldsymbol{n}}_{2,p})L_{2,p}}{\sum_{k=1}^{N_{out,p}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_{k,p})L_{k,p}} \dot{m}_{out,p} \\ &- \frac{(\boldsymbol{\tau}_{air,l} \cdot \hat{\boldsymbol{n}}_{0,l})L_{0,l}}{\sum_{k=1}^{N_{out,l}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_{k,l})L_{k,l}} \dot{m}_{out,l} = 0 \end{aligned} \tag{3.12}$$

Particular care must be taken when treating two particular types of elements' boundaries:

- Boundaries of domain boundary elements that are flagged as inflow.

- Boundaries that are flagged as inflow by both neighbouring elements.

In these two cases, the inflow mass corresponding to that boundary will be set to zero. For example, in the case of figure 3.5, if boundary 0 is of one of

the kind stated above, the second equation will be written as:

$$\dot{m}_{in,i} - \frac{(\boldsymbol{\tau}_{air,j} \cdot \hat{\boldsymbol{n}}_{4,j})L_{4,j}}{\sum_{k=1}^{N_{out,j}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_{k,j})L_{k,j}}\dot{m}_{out,j}$$
$$- \frac{(\boldsymbol{\tau}_{air,p} \cdot \hat{\boldsymbol{n}}_{2,p})L_{2,p}}{\sum_{k=1}^{N_{out,p}}(\boldsymbol{\tau}_{air,k} \cdot \hat{\boldsymbol{n}}_{k,p})L_{k,p}}\dot{m}_{out,p} = 0 \tag{3.13}$$

When the system is completely filled, it can be written as a linear system of the form $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{x}$ is the vector of the unknowns that are the total mass flowing in and out of each element. For the way the problem is formulated, for each row of the matrix $\mathbf{A}$ at the most $1 + N_b$ elements of the row are different from zero. Therefore $\mathbf{A}$ is as a sparse matrix and can make use of efficient data structures and solution methods. In this work, the system has been solved using a multifrontal LU factorization implemented in the library UMFPACK. Iterative sparse solvers should instead be used for large 3D problems.

Once $\dot{m}_{in}$ and $\dot{m}_{out}$ are computed, they are stored in the time step solution. The elements located at the transition rime-glaze are left at the end of the process because, from the equation 2.36, it can be seen that is necessary to know the value of the inflow mass into the cell to calculate the ice thickness.

3.2 Unsteady liquid film model

The main problem of the steady liquid film models is that the mass of water entering a control volume either freezes, evaporates or flows out, as can be seen in the mass conservation 3.2. This means that the influence of the liquid film height on the ice accretion cannot be taken into account properly. In the following section, a state-of-the-art unsteady liquid film model has been considered [40]. In this way, a good estimation of the liquid film height and its influence on the ice accretion can be evaluated. In this work, the liquid film is considered isothermal with $T_{H_2O} \approx T_{freezing}$. This hypothesis can be relaxed by considering a coupling with a proper energy balance to the determine the water temperature, like the one presented in section 4.3.4.

3.2.1 Lubrication theory

The unsteady liquid film model is based on a simplification of the Navier-Stokes equations, that is the lubrication theory. In this way the system is reduced to a single scalar equation which is only dependent on the height of the liquid film. The main part of the derivation of the final governing equation will be now presented. For a more detailed discussion, the reader may refer to [40].

The Navier-Stokes equation for an incompressible flow with homogeneous density, such as the water film flow, are now reported for clarity:

$$\begin{cases} \nabla \cdot \boldsymbol{u} = 0 \\ \rho_{H_2O} \left(\dfrac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla) \boldsymbol{u} \right) = -\nabla p + \mu_{H_2O} \nabla^2 \boldsymbol{u} + \rho_{H_2O} \boldsymbol{g} \end{cases} \tag{3.14}$$

where $\boldsymbol{u} = (u, v, w)$ is the fluid velocity, p is the fluid pressure and $\boldsymbol{g}$ is the gravity vector.

Some hypothesis are introduced in order to simplify these equations:

- The water film is thin and the flow regime is laminar. Being H the characteristic depth and L the characteristic length in the direction of the flow, $\frac{H}{L} \ll 1$. Moreover, for the flow to be laminar, $\frac{uL}{\nu} \frac{H}{L} \ll 1$, meaning that the Reynolds number must be approximately one or lower.

- The water properties, such as density ρ_{H_2O} and viscosity μ_{H_2O}, are considered to be constant. Under these hypothesis, the equations 3.14 can be reduced to [41]:

$$\begin{cases} \mu_{H_2O} \dfrac{\partial^2 u}{\partial z^2} = \dfrac{\partial p}{\partial x} - \rho_{H_2O} \boldsymbol{g} \cdot \hat{\boldsymbol{x}} \\ \mu_{H_2O} \dfrac{\partial^2 v}{\partial z^2} = \dfrac{\partial p}{\partial y} - \rho_{H_2O} \boldsymbol{g} \cdot \hat{\boldsymbol{y}} \\ 0 = \dfrac{\partial p}{\partial z} \\ \nabla \cdot \boldsymbol{u} = 0 \end{cases} \tag{3.15}$$

where x and y are the coordinates in the surface plane while z is the coordinate normal to the surface, as shown in figure 3.6.
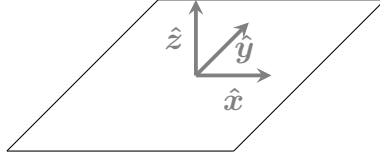
Figure 3.6 Coordinate system for the lubrication theory equations

The equations 3.15 must be solved with proper boundary conditions. First of all, a no-slip condition at the ice-water interface must be imposed:

$$u(x, y, B, t) = v(x, y, B, t) = 0 \tag{3.16}$$

Then, the continuity of shear and normal stress at the air-water interface must be considered:

$$\begin{cases} \mu_{H_2O} \dfrac{\partial u}{\partial z}\bigg|_{B+h} = \tau_{air,x} \\[2ex] \mu_{H_2O} \dfrac{\partial v}{\partial z}\bigg|_{B+h} = \tau_{air,y} \\[2ex] p - p_{air} = -\sigma \nabla^2 (B + h) \end{cases} \tag{3.17}$$

The last boundary condition concerns the continuity of mass flux at the air-water interface and at the ice-water interface:

$$\begin{cases} \rho_{H_2O} \, \hat{\boldsymbol{n}}_{H_2O} \cdot (\boldsymbol{u}_{air} - \boldsymbol{u}_h)\big|_{B+h} = LWC \, \hat{\boldsymbol{n}}_{H_2O} \cdot (\boldsymbol{u}_{H_2O} - \boldsymbol{u}_h)\big|_{B+h} \\[2ex] \rho_{H_2O} \, \hat{\boldsymbol{n}}_{ice} \cdot (\boldsymbol{u}_{H_2O} - \boldsymbol{u}_B)\big|_{B} = \rho_i \, \hat{\boldsymbol{n}}_{ice} \cdot (\boldsymbol{u}_{ice} - \boldsymbol{u}_B)\big|_{B} \end{cases} \tag{3.18}$$

where all the contributions are listed below:

- $\hat{\boldsymbol{n}}_{ice} = (-\frac{\partial B}{\partial x}, -\frac{\partial B}{\partial y}, 1)$ is the normal to the ice surface.

- $\boldsymbol{u}_B = (0, 0, \frac{\partial B}{\partial t})$ is the velocity of the ice-water interface.

- $\boldsymbol{u}_{ice} = (0, 0, 0)$ is the ice velocity.

- $\boldsymbol{u}_{H_2O}$ is the liquid film velocity.

- $\hat{\boldsymbol{n}}_{H_2O} = (-\frac{\partial(B+h)}{\partial x}, -\frac{\partial(B+h)}{\partial y}, 1)$ is the normal to the water surface.

- $\boldsymbol{u}_h = (0, 0, \frac{\partial(B+h)}{\partial t})$ is the velocity of the water-air interface.

- $\boldsymbol{u}_{air}$ is the airflow velocity.

The second boundary condition of the system 3.18 can be expanded as follows:

$$\rho_{H_2O}\left[-\frac{\partial B}{\partial x}u - \frac{\partial B}{\partial y}v + \left(w - \frac{\partial B}{\partial t}\right)\right]\bigg|_B = \rho_i\left(-\frac{\partial B}{\partial t}\right)\bigg|_B \qquad (3.19)$$

Applying the no-slip condition 3.16, the equation 3.19 reduces to:

$$w\bigg|_B = \left(1 - \frac{\rho_i}{\rho_{H_2O}}\right)\frac{\partial B}{\partial t} \qquad (3.20)$$

Following the same approach, the film velocity in the $\hat{\boldsymbol{z}}$ direction at the air-water interface can be written as:

$$\begin{aligned} w\bigg|_{B+h} = &\left(1 - \frac{LWC}{\rho_{H_2O}}\right)\left(\frac{\partial B}{\partial t} + \frac{\partial h}{\partial t}\right) + u\bigg|_{B+h}\left(\frac{\partial B}{\partial x} + \frac{\partial h}{\partial x}\right) \\ &+ v\bigg|_{B+h}\left(\frac{\partial B}{\partial y} + \frac{\partial h}{\partial y}\right) - \frac{\dot{m}_{imp} - \dot{m}_e}{A\rho_{H_2O}} \end{aligned} \qquad (3.21)$$

where the term $\left(1 - \frac{LWC}{\rho_{H_2O}}\right)$ can be approximated to one since $\frac{LWC}{\rho_{H_2O}} \ll 1$. The integration procedure will be shown for the momentum balance in the $\hat{\boldsymbol{x}}$ direction only. In a similar manner it can be performed for the momentum balance in the $\hat{\boldsymbol{y}}$ direction. Therefore in the latter case only the final result will be presented. The first equation of the system 3.15 can be integrated along the $\hat{\boldsymbol{z}}$ direction obtaining:

$$\begin{aligned} \int_0^z \mu_{H_2O}\frac{\partial^2 u}{\partial z^2} &= \int_0^z \frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}} \\ \mu_{H_2O}\frac{\partial u}{\partial z} &= \left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)z + C \\ \mu_{H_2O}u &= \left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)\frac{z^2}{2} + Cz + D \end{aligned} \qquad (3.22)$$

Applying the boundary conditions 3.17 and 3.16, the integration constants C and D can be determined as follows:

$$\begin{aligned} C &= \tau_{air,x} - \left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)(B+h) \\ D &= -\tau_{air,x}B - \left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)\left(\frac{B^2}{2} - (B+h)B\right) \end{aligned} \qquad (3.23)$$

In this way, the velocity profile in the water layer in the $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ direction is obtained substituting the values of the integration constants in the expression of the velocity:

$$\mu_{H_2O}u = \frac{1}{2}\left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)\left[z^2 - B^2 - 2\left(z - B\right)\left(B + h\right)\right] + \tau_{air,x}\left(z - B\right)$$

$$\mu_{H_2O}v = \frac{1}{2}\left(\frac{\partial p}{\partial y} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{y}}\right)\left[z^2 - B^2 - 2\left(z - B\right)\left(B + h\right)\right] + \tau_{air,y}\left(z - B\right)$$

$$(3.24)$$

The incompressibility constraint can be integrated along the film height, from B to $B + h$ obtaining:

$$w\Big|_{B+h} - w\Big|_{B} = -\int_{B}^{B+h}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)dz \qquad (3.25)$$

The right-hand side of the equation 3.25 can be manipulated by means of the Leibniz theorem:

$$\int_{B}^{B+h}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)dz = \frac{\partial}{\partial x}\left(\int_{B}^{B+h}udz\right) - u\Big|_{B+h}\frac{\partial\left(B + h\right)}{\partial x} + u\Big|_{B}\frac{dB}{dx}$$
$$+ \frac{\partial}{\partial y}\left(\int_{B}^{B+h}vdz\right) - v\Big|_{B+h}\frac{\partial\left(B + h\right)}{\partial y} + v\Big|_{B}\frac{dB}{dy}$$

$$(3.26)$$

The average velocity in the liquid film can be written using the definition of bulk velocity. Therefore, the averaged components of u and v are:

$$\bar{u} = \frac{1}{h}\int_{B}^{B+h}udz$$
$$\bar{v} = \frac{1}{h}\int_{B}^{B+h}vdz$$

$$(3.27)$$

Substituting the definitions of bulk velocity in the expression 3.26, the following relation is obtained:

$$\int_{B}^{B+h}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)dz = \frac{\partial}{\partial x}\left(h\bar{u}(h)\right) - u\Big|_{B+h}\frac{\partial\left(B + h\right)}{\partial x} + u\Big|_{B}\frac{dB}{dx}$$
$$+ \frac{\partial}{\partial y}\left(h\bar{v}(h)\right) - v\Big|_{B+h}\frac{\partial\left(B + h\right)}{\partial y} + v\Big|_{B}\frac{dB}{dy}$$

$$(3.28)$$

The terms $u\Big|_{B}$ and $v\Big|_{B}$ are set to zero for the non-slip boundary condition.

The expression 3.28 can be therefore simplified to:

$$-\int_{B}^{B+h} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) dz = -\frac{\partial}{\partial x}\left(h\bar{u}(h)\right) + u\Big|_{B+h}\frac{\partial (B+h)}{\partial x}$$
$$-\frac{\partial}{\partial y}\left(h\bar{v}(h)\right) + v\Big|_{B+h}\frac{\partial (B+h)}{\partial y} \tag{3.29}$$

Considering now the equation 3.25, $w\big|_{B+h}$ and $w\big|_{B}$ are replaced with the expressions of the mass flux continuity at the interfaces 3.21 and 3.20:

$$w\Big|_{B+h} - w\Big|_{B} = \left(\frac{\partial B}{\partial t} + \frac{\partial h}{\partial t}\right) + u\Big|_{B+h}\frac{\partial (B+h)}{\partial x} + v\Big|_{B+h}\frac{\partial (B+h)}{\partial y}$$
$$-\frac{\dot{m}_{imp} - \dot{m}_e}{A\rho_{H_2O}} - \left(1 - \frac{\rho_i}{\rho_{H_2O}}\right)\frac{\partial B}{\partial t} \tag{3.30}$$

The last two expressions are now replaced in 3.25 and the equation is simplified to obtain:

$$\frac{\partial h}{\partial t} - \frac{\dot{m}_{imp} - \dot{m}_e}{A\rho_{H_2O}} + \frac{\rho_i}{\rho_{H_2O}}\frac{\partial B}{\partial t} = -\frac{\partial}{\partial x}\left(h\bar{u}(h)\right) - \frac{\partial}{\partial y}\left(h\bar{v}(h)\right) \tag{3.31}$$

Replacing $\dot{m}_{ice} = \rho_i A \frac{\partial B}{\partial t}$, a transport equation for h is obtained:

$$\frac{\partial h}{\partial t} + \frac{\partial (h\bar{u}(h))}{\partial x} + \frac{\partial (h\bar{v}(h))}{\partial y} = \frac{\dot{m}_{imp} - \dot{m}_e - \dot{m}_{ice}}{A\rho_{H_2O}} \tag{3.32}$$

Substituting the equations 3.24 in the definition of bulk velocity, the values of the averaged components $\bar{u}$ and $\bar{v}$ are:

$$\bar{u}(h) = \frac{h}{2\mu_{H_2O}}\tau_{air,x} - \frac{h^2}{3\mu_{H_2O}}\left(\frac{\partial p}{\partial x} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{x}}\right)$$
$$\bar{v}(h) = \frac{h}{2\mu_{H_2O}}\tau_{air,y} - \frac{h^2}{3\mu_{H_2O}}\left(\frac{\partial p}{\partial y} - \rho_{H_2O}\boldsymbol{g}\cdot\hat{\boldsymbol{y}}\right) \tag{3.33}$$

Considering that the main driving force of the liquid film flow is the shear stress [10] [40] and that for aeronautical applications $h \approx 10^{-4}$ [39], equation 3.33 can be further simplified to:

$$\bar{u} = \frac{h}{2\mu_{H_2O}}\tau_{air,x}$$
$$\bar{v} = \frac{h}{2\mu_{H_2O}}\tau_{air,y} \tag{3.34}$$

Note that since the liquid film height is small and its velocity is much lower than the freestream velocity, it can be assumed that the shear stress that has been imposed on the air-water boundary is the one that would be obtained on a dry surface.

Equation 3.32 is a scalar PDE that depends only on the liquid film height h. This equation has a strong physical meaning: at a point (x, y) on the surface, the variation in space and time of liquid film height depends on the source term that is the impinged, evaporated and freezed water.

3.2.2 Discretization of the unsteady model

In this section, the discretization of the problem to evaluate the water film height and the ice thickness on a surface is presented. First, the transport equation derived in section 3.2 is discretized in space and time using a finite volume approach. The domain has to be divided into control volumes V_i and a 1D surface mesh in a 2D problem and a 2D surface mesh in a 3D problem are obtained, as shown in figures 3.1 and 3.2. Then, the equation 3.32 is integrated on each control volume of the domain:

$$\int_{V_i} \left[\frac{\partial h}{\partial t} + \nabla \cdot (h\bar{\boldsymbol{u}}(h)) \right] dV_i = \int_{V_i} \left(\frac{\dot{m}_{imp} - \dot{m}_e - \dot{m}_{ice}}{A\rho_{H_2O}} \right) dV_i \qquad (3.35)$$

Since the control volumes V_i are not time dependent, the Green-Gauss theorem can be applied trasforming a volume integral into a surface integral:

$$\int_{V_i} \left[\nabla \cdot (h\bar{\boldsymbol{u}}(h)) \right] dV_i = \int_{\partial V_i} \left[(h\bar{\boldsymbol{u}}(h)) \cdot \hat{\boldsymbol{n}} \right] d\partial V_i \qquad (3.36)$$

The boundary of the control volume ∂V_i is composed by the sum of the segments composing the elements. The integral 3.36 can be therefore rewritten as:

$$\int_{\partial V_i} \left[(h\bar{\boldsymbol{u}}(h)) \cdot \hat{\boldsymbol{n}} \right] d\partial V_i = \sum_{j=1}^{N_b} \int_{L_{j,i}} \left[(h\bar{\boldsymbol{u}}(h)) \cdot \hat{\boldsymbol{n}}_j \right] dL_{j,i} = \sum_{j=1}^{N_b} \int_{L_{j,i}} F(h) dL_{j,i}$$
$$(3.37)$$

where $L_{j,i}$ is the length of boundary j of the control volume i. In each control volume, the average water film height and a source term can be defined as

follows:

$$\bar{h}_i = \frac{1}{V_i} \int_{V_i} h \, dVi$$

$$\bar{S}_i = \frac{1}{V_i} \int_{V_i} \left(\frac{\dot{m}_{imp} - \dot{m}_e - \dot{m}_{ice}}{A\rho_{H_2O}} \right) dV_i \tag{3.38}$$

The integral 3.37 must be evaluated at the boundaries of the control volume but only the average value of h is available for each mesh element. It is therefore considered that for each control volume the value of h is constant and equal to the average defined in 3.38. To evaluate the fluxes at the interface, an upwind first order scheme has been considered and the flux of the element i at the interface with the element k can be written as:

$$Q(h_i, h_k) = \frac{1}{2} [F(h_i) + F(h_k)] + \frac{1}{2} \frac{F'(h_i)}{|F'(h_i)|} [F(h_i) - F(h_k)] \tag{3.39}$$

where in the present model $F(h_i) = \frac{\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_i}{2\mu_{H_2O}} \bar{h}_i^2$ and $F'(h_i) = \frac{\boldsymbol{\tau}_{air} \cdot \hat{\boldsymbol{n}}_i}{\mu_{H_2O}} \bar{h}_i$. The discretization in space of the transport equation can be therefore written as:

$$\frac{\partial \bar{h}_i}{\partial t} + \frac{1}{V_i} \sum_{j=1}^{N_b} [\text{sign}\,(F'(h_i))\, L_{j,i} Q(h_i, h_j)] = \bar{S}_i \tag{3.40}$$

The 3.40 is a semi-discrete equation as it is discrete in space while it is not in time. For the time discretization, a backward Euler scheme has been implemented since the forward Euler scheme would have required a too small time step to satisfy the CFL condition ($\Delta t \approx 10^{-9}$):

$$\bar{h}_i^{n+1} = \bar{h}_i^n - \frac{\Delta t}{V_i} \sum_{j=1}^{N_b} \left[\text{sign}\,\left(F'(h_i)^{n+1}\right) L_{j,i} Q(h_i, h_j)^{n+1} \right] + \Delta t \bar{S}_i^{n+1} \tag{3.41}$$

The discrete equation 3.41 can be solved together with the equation to determine the ice thickness, depending on the type of ice and on the model used (see chapter 2). The general system that must be solved for each control volume at each time step is below reported:

$$\begin{cases} B_i^{n+1} = B_i^n + \left(\dfrac{\partial B}{\partial t} \right)^{n+1} \Delta t \\[2em] \bar{h}_i^{n+1} = \bar{h}_i^n - \dfrac{\Delta t}{V_i} \sum_{j=1}^{N_b} \left[\text{sign}\,\left(F'(h_i)^{n+1}\right) L_{j,i} Q(h_i, h_j)^{n+1} \right] + \Delta t \bar{S}_i^{n+1}(h_i, B_i) \end{cases}$$

$$\tag{3.42}$$

A fixed point scheme has been used to solve the system 3.42 and it is summarized in the pseudo algorithm 2. The fixed point loop is performed at each time step of the ice accretion calculation. The iterations are stopped once the error 3.43 reaches a value below a tolerance set by the user. For each element, the difference on the liquid film water height between to subsequent iterations normalized by the height at the previous iteration is evaluated. Then, the greatest of these values is taken as the error. The idea is that convergence is reached once two subsequent values of the film height are very close.

$$\text{Err} = \left\| \frac{\bar{h}^k - \bar{h}^{k-1}}{\bar{h}^{k-1}} \right\|_{L^\infty} \tag{3.43}$$

The ice accretion loop is performed as follows. At first, the variables are initialized using the previous time step values. Then, as it was done for the iterative steady model, the rime limit thickness B_g is computed for each element and the accretion law is chosen based on the ice thickness at the previous time step. The ice accretion rate $\frac{\partial B}{\partial t}$ is calculated using the ice thickness B^{k-1} and the water film height h^{k-1} of the previous iteration. Then, in case of glaze ice accretion, the source term for the liquid film transport equation is evaluated as:

$$\bar{S}_i = \frac{\beta \, LWC \, V_\infty - \dfrac{\dot{Q}_e}{AL_e} - \rho_{gi} \dfrac{\partial B}{\partial t}}{\rho_{H_2O}} \tag{3.44}$$

The average liquid film height $\bar{h}_i^k$ is then computed as:

$$\bar{h}_i^{k,n} = \bar{h}_i^{k-1,n} - \frac{\Delta t}{V_i} \sum_{j=1}^{N_b} \left[\text{sign}\left(F'(h_i)^{k-1,n} \right) L_{j,i} Q(h_i, h_j)^{k-1,n} \right] + \Delta t \bar{S}_i^{k-1,n}(h_i, B_i) \tag{3.45}$$

If a rime ice control volume is next to a glaze ice element, the inflow water mass in the rime element must be evaluated to determine the correct ice thickness. This is done using the definition of mass flow rate $\dot{m} = \rho AV$, where ρ is the density of water, $A = \bar{h}L$ is the cross-sectional area obtained by multiplying the height of the film times the width of the boundary and V is the velocity of the film:

$$\dot{m} = \rho_{H_2O} \, \bar{h} \, L \, \bar{\boldsymbol{u}}(h) \cdot \hat{\boldsymbol{n}} \tag{3.46}$$

The height and the velocity of the film are the ones evaluated at the neighbour glaze cell. The error is then computed and, in order to have a more robust scheme, the value of h for the next iteration is initialized as a weighted average between $\bar{h}_i^k$ and $\bar{h}_i^{k-1}$

$$\bar{h}_i^{k+1} = a\bar{h}_i^k + (1-a)\bar{h}_i^{k-1} \tag{3.47}$$

where a is a weight parameter that assumes a value between 0 and 1. Its value has been set by trial and error and depends on the amount of water that flows on the ice accretion process. The used value is $a \approx [0.05, 0.4]$. At higher temperature, a lower value of a should be used.

Iterations are stopped once the error 3.43 reaches a tolerance that is usually Tol $\approx 10^{-8}$, which is a good trade off between computational speed and accuracy. Once convergence is reached, the ice solution is stored and time is incremented $t^{n+1} = t^n + \Delta t$ where Δt is a value set by the user.

Algorithm 2 Pseudo algorithm used to determine ice thickness and mass of water inside an ice accretion time step with the unsteady liquid film model

1: **procedure** DETERMINEICETHICKNESSANDWATERMASS(time)
2:     **loop** on each element of the iced boundary
3:         $B^0 \leftarrow$ PreviousTimeStepIceThickness(elementID)
4:         $\bar{h}^0 \leftarrow$ PreviousTimeStepIceLiquidFilmHeight(elementID)
5:         B_g (elementID) $\leftarrow$ RimeLimit(elementID)
6:     **end loop**
7:     **while** Err > Tol & $k <$ MaxIter **do**
8:         **loop** on each element of the iced boundary
9:             **if** B^0(elementID) $> B_g$(elementID) & B_g(elementID)> 0 **then**
10:                 B^k(elementID) $\leftarrow$ GlazeIce(elementID, time)
11:                 $\bar{h}_i^{k,n} = \bar{h}_i^{k-1,n} - \frac{\Delta t}{V_i} \sum_{j=1}^{N_b} \left[\text{sign}\left(F'(h_i)^{k-1,n}\right) L_{j,i} Q(h_i, h_j)^{k-1,n} \right] + \Delta t \bar{S}_i^{k-1,n}(h_i, B_i)$
12:             **else**
13:                 B^k(elementID) $\leftarrow$ RimeIce(elementID, time, $\dot{m}_{in}^{k-1}$)
14:                 $\bar{h}^k = 0$
15:             **end if**
16:         **end loop**
17:         Err $= \left\| \frac{\bar{h}^k - \bar{h}^{k-1}}{\bar{h}^{k-1}} \right\|_{L^\infty}$
18:         $k = k + 1$
19:     **end while**
20: **end procedure**

Chapter 4

A 2D electro-thermal Ice Protection System model

Anti-ice numerical simulation tools are very important in the aircraft development process and they can be used from the conceptual design to the certification phase [52]. A few works have been done on anti-icing system simulation. Some of these works start with the mathematical modeling of a de-icing system and use it to simulate an anti-icing system. These studies focus on transient heat transfer calculations inside a multi-layered wall. Some of them also try to solve the phase change and shedding problems [64]. This phenomena do not occur when the IPS works exclusively in anti-icing conditions. In this regime, previous studies mainly focus on the calculation of the heat transfer coefficient of air and water and the analysis of the runback water film. This is necessary to determine the amount of impinging water that evaporates and the distribution on the surface of the remaining portion. Moreover, it is useful to study the surface temperature to consider its effects on the degradation of the materials.

Al-Khalil made a research effort to develop the mathematical model of an IPS considering rivulets effects on the thermal balance [3]. Rivulets form when the water film flowing on the surface reaches a critical thickness and then breaks up. Al-Khalil implemented the numerical code ANTICE, which uses the boundary layer theory and the particle tracking from LEWICE, to predict and evaluate performances of an anti-icing system. An intensive experimental research program was led at NASA Glenn Research Center to validate

the code and results in different environmental conditions are available at reference [2]. These data have been used to validate many numerical codes such as LEWICE [2], CANICE [37] and the models by Silva et al. [53] and Bu et al. [13]. However, the boundary layer theory implemented in LEWICE was dedicated to calculate the heat transfer over rough and nearly isothermal surfaces, which is the case of an ice shape. As the airfoil surface in anti-ice condition is smooth and the temperature can be very different from point to point, Al-Khalil et al. studied the effect of using experimental heat transfer coefficient into the numeric simulations.

Another important work concerning the numerical simulation of an IPS was led by Morency et al. who implemented a model of an hot air anti-icing system in the ice accretion simulation code CANICE and validated their results with experimental data from Al-Khalil [37]. The authors published the results of three version of the code: CANICE A, which uses an experimental overall value for the heat transfer coefficient, CANICE B, which uses an estimation of the heat transfer coefficient, an estimation of the transition position and the turbulent boundary layer model developed by Ambrok and CANICE FD which uses finite differences to compute mass, momentum and heat balance around the airfoil. Furthermore, the procedure takes into account the laminar and turbulent boundary layer as well as transition.

In 2007, Silva et al. developed a model of a thermal IPS in anti-ice conditions by numerically solving the conservation equations for mass and energy into control volumes and compared their results with other codes and experimental data [53] [54]. The authors implemented a numerical code to evaluate integral equations of momentum and thermal boundary layer considering a non-isothermal airfoil surface with evaporation in laminar, transition, and turbulent flow regimes. This model was chosen to be implemented in PoliMIce because the results obtained were in accordance to the experimental measurements and the authors presented a detailed explanation of the implementation and development of the model.

In this chapter, the thermal IPS model proposed by Silva et al. is presented. In particular, the geometry of the heaters, the boundary layer theory and the numerical solver are explained.

4.1   An anti-ice simulation overview

Similar to an ice accretion computation, the development of a 2D simulation with a thermal IPS in anti-ice conditions requires three modules:

- **Aerodynamic solver**. It is the part in charge of the solution of the flow field.

- **Particle tracking**. It calulates droplets trajectories and the collection efficiency.

- **Thermodynamic solver**. It is the core of the software. It contains the solver for the conservation laws, the models of the boundary layer over smooth and non-isothermal surfaces and the solver which allows to compute the properties of the runback water film.

The implementation of a new thermodynamic solver together with a boundary layer evaluation routine was required because the ice-protected airfoil has a heated, permeable, nonisothermal, relatively smooth, and variable wetness surface interfacing with the boundary layer flow [54]. These features do not allow to obtain the solution of the flow field by means of a Navier-Stokes simulation as the liquid film changes the properties of the air boundary layer from point to point and this makes the imposition of proper boundary conditions very difficult. Thus, an Euler simulation is performed to solve the flow field and different models are implemented in the thermodynamic solver to compute the properties of the thermal boundary layer in the aforementioned conditions.

The purpose of the CFD simulation is only to compute the local values of pressure, velocity and temperature and to compute the droplets trajectories by means of a Lagrangian particle tracking. In this way, the heat transfer coefficient and the skin friction coefficient are computed at the beginning of the simulation in the thermodynamic solver. The obtained values are used in the mass and momentum conservation laws to compute the properties of the runback water film and other relevant parameters of the simulation, like the temperature of the surface.

The equations, which are non linear, are numerically solved in every finite volume on the surface until their residuals are below a given tolerance. At each iteration, the values of heat transfer coefficient, skin friction coefficient,
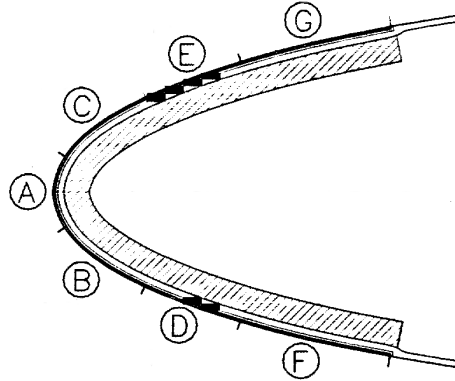
Figure 4.1 Airfoil leading edge and heated zones. Figure adapted from Figure 1 in reference [2]

No.	MATERIAL	$K\left[\frac{\text{W}}{\text{m K}}\right]$	$\rho\left[\frac{\text{kg}}{\text{m}^3}\right]$	$C_p\left[\frac{\text{J}}{\text{kg K}}\right]$
1	Heating Element (Alloy 90)	41.02	8906.26	385.19
2	Erosion Shield (SS 301 HH)	16.27	8025.25	502.42
3	Elastomer (Cox 4300)	0.256	1384	1256.04 ± 125.6
4	Fiberglass/Epoxy Composite	0.294	1794.07	1570.05
5	Silicone Foam Insulation	0.121	648.25	1130.44 ± 125.6

Table 4.1 Leading edge material properties [2]

surface temperature and water film properties are updated allowing to obtain the solution of the dynamic and thermal boundary layer despite the many peculiar features of the problem.

4.2 Geometric description

A thermo-electric protection heats surfaces at risk of ice accretion exploiting the Joule effect due to an electric current going through a resistive component. The structure of the heating system can be represented by a number of heating zones. The user can specify a variable heat flux at the surface, different on each zone, to obtain a customized heating power as shown in figure 4.1.

Every single heater is composed by several layers and figure 4.2 illustrates a
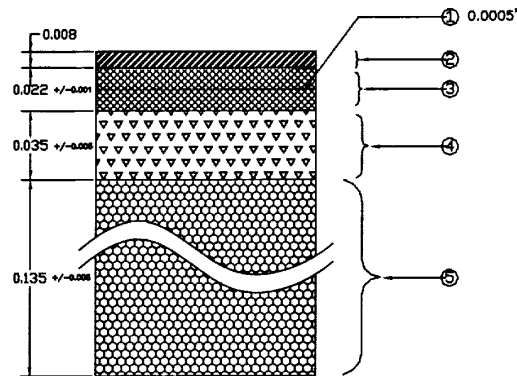
Figure 4.2   Leading edge material composition. Thicknesses in inches [2]

possible material composition of the leading edge heaters and their properties
are given in table 4.1. In particular, the metal heating element is embedded
between two layers of insulating material and it is designed for a wide range
of power loadings, from $0.15 \frac{\text{kW}}{\text{m}^2}$ to $50.0 \frac{\text{kW}}{\text{m}^2}$.

This setup was used by Al-Khalil at NASA Glenn Research Center for the
validation of ANTICE. This geometry is taken as reference in this work and
it is used for the validation of PoliMIce, as shown in chapter 6.

The new version of PoliMIce can handle multiple disjoint boundaries with
different boundary conditions and allows a convenient treatment of the prob-
lem. Therefore the airfoil is splitted in multiple boundaries according to the
geometry of the selected test case. The CFD simulation requires an Euler
boundary condition on each boundary whereas the heating power is set in
the PoliMIce configuration file.

4.3   Mathematical description

From now on, the reference system adopted in the formulation is shown
in figure 4.3, where s indicates the curvilinear abscissa. The zoom on the
airfoil surface shows that four different sub-problems interact with each other
and this domain division has inspired the new data structure of PoliMIce,
as shown in chapter 5. Thus, thanks to the new version of the software, the
mathematical modeling of each domain has been implemented in different
classes allowing an easy and convenient connection between the coupled
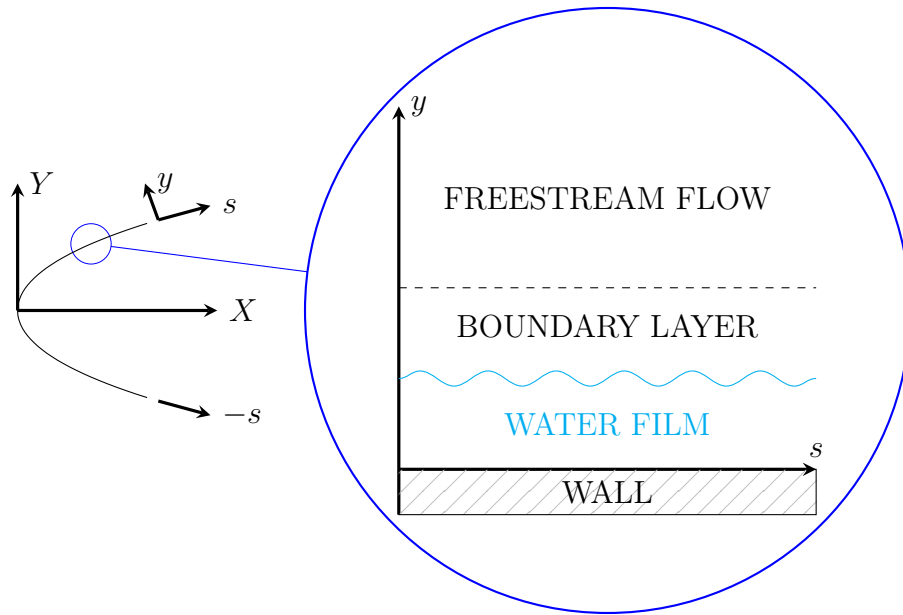
Figure 4.3 Airofil reference system with a detail of the different zones of the physical problem

problems.

The core of the current anti-ice model consists in the resolution of the energy and mass conservation laws into control volumes on the airfoil surface. The energy and mass terms considered for anti-icing are very similar to the one presented in chapter 2 for ice accretion, with the addition of the IPS heat fluxes. The equations will be described in detail in section 4.3.4. For anti-icing, the convective and the evaporative heat fluxes represent the main contribution on the total heat loss, thus a correct and accurate determination of the air heat transfer coefficient is one of the key points in a simulation of an airfoil with an IPS in anti-icing conditions. In this work, this is achieved through the implementation of a model for the boundary layer using equations in the integral form.

The models necessary in the thermodynamic solver are presented following the order in which they are called in the framework, as it can be seen from figure 4.10. The first models presented in this chapter are the ones of the momentum and thermal boundary layers as their correct determination establishes the starting point of the simulation. Then, a model for an accurate evaluation of the evaporation heat flux is reported due to its importance. A model for

a non-isothermal water film is then presented in section 4.3.3. Lastly, the conservation laws, the thermodynamic model of the problem and its resolution are finally presented in section 4.3.4.

4.3.1 Boundary layer model

The evaluation of the convective heat transfer coefficient and shear stress, needed for the motion of the water liquid film, is performed by solving the momentum and thermal boundary layer equations in the integral form for both laminar and turbulent flows. Transition is treated using a linear combination of the formulas obtained for the laminar and turbulent cases weighted by a probability function which represents the probability of the turbulent boundary layer to appear on a certain stage. These methods have been used by Silva et al. [53] and Bu et al. [13] and replicated in this work, due to their validity and suitability.

As the desired properties depend mostly on the regime, it is important to properly locate the transition onset and the transition end in order to identify whether the boundary layer is laminar, turbulent or in transition even though transition is a very difficult to predict phenomenon. The model proposed by Abu-Ghannam and Shaw [1], used by both Bu et al. [13] and Silva et al. [53], is based on the evaluation of the boundary layer momentum thickness at each stage and compared to a threshold which identifies the transition onset.

The boundary layer momentum equation can be expressed in nondimensional form introducing the skin friction coefficient C_f, the displacement thickness δ_1 and momentum thickness δ_2:

$$\frac{C_f}{2} = \frac{d\delta_2}{ds} + \delta_2 \left[\left(2 + \frac{\delta_1}{\delta_2} \right) \frac{1}{u_e} \frac{du_e}{ds} \right] \tag{4.1}$$

where u_e is the local velocity at the external edge of the boundary layer.

Kays and Crawford [33], based on some approximations, integrated the equation 4.1 and obtained the momentum thickness in both laminar and turbulent regime:

$$\delta_{2_{lam}} = \frac{0.664 \, \nu_{air}^{0.5}}{u_e^{2.84}} \left(\int_{s_{stag}}^{s} u_e^{4.68} ds \right)^{0.5} \tag{4.2}$$

$$\delta_{2_{turb}} = \frac{0.036 \, \nu_{air}^{0.5}}{u_e^{3.29}} \left(\int_{s_{tr}}^{s} u_e^{3.86} ds \right)^{0.8} + \delta_{2_{trans}} \tag{4.3}$$

where the equation 4.2 holds from the stagnation point to the transition onset and the 4.3 from the latter onwards. In this work the position of stagnation point is located by finding the point of minimum velocity.

According to Abu-Ghannam and Shaw [1], the beginning and the end of transition are located computing the Reynolds number based on the boundary layer momentum thickness and a threshold, defined as follows:

$$\mathrm{Re}_{\delta_2} = \frac{u_e\,\delta_2}{\nu_{air}} \tag{4.4}$$

$$\lambda = \frac{{\delta_2}^2}{\nu_{air}}\frac{du_e}{ds} \tag{4.5}$$

$$\begin{cases} F(\lambda) = 6.91 + 12.75\lambda + 63.64\lambda^2 & \lambda < 0 \\ F(\lambda) = 6.91 + 2.48\lambda - 12.27\lambda^2 & \lambda > 0 \end{cases} \tag{4.6}$$

$$\mathrm{Re}_{\delta_{2\mathrm{trans}}} = 163 + \exp\left(F(\lambda) - \frac{F(\lambda)}{6.91}\,\mathrm{Tu} \right) \tag{4.7}$$

Tu indicates the turbulence intensity, defined as the ratio between the root-mean-square of the velocity fluctuations and the mean velocity:

$$\mathrm{Tu} = \frac{u'_{\mathrm{RMS}}}{\bar{U}} \tag{4.8}$$

Under the hypothesis of homogeneous isotropic turbulence, $u'_{\mathrm{RMS}} = \sqrt{\frac{2}{3}k}$ where k is the turbulent kinetic energy.

Abu-Ghannam and Shaw, in their formulations, introduced the nondimensional parameter λ, defined in the equation 4.5, used to compute the Reynolds number at transition.

The location of the transition onset is found looping on each node of the surface mesh and computing the laminar momentum thickness with the equation 4.2. Then, based on this length, the Reynolds number is computed as reported in the equation 4.4. By means of the momentum thickness, the parameter λ is computed and therefore the threshold $\mathrm{Re}_{\delta_{2\mathrm{trans}}}$. Finally, a check on the local Reynolds numbers just obtained allows to determine whether the current node is the transition onset or not and this happens when Re_{δ_2} is greater than the threshold $\mathrm{Re}_{\delta_{2\mathrm{trans}}}$.

The determination of the end of transition is likewise computed with the same procedure but with different equations. From the transition onset the momentum thickness is calculated with the turbulent expression 4.3. The Reynolds

number based on the curvilinear abscissa at the transition onset allows to obtain the parameter $\mathrm{Re_L}$ as shown in the equation 4.9. The parameter λ is again computed allowing to get the threshold Reynolds number for the end of transition 4.10. Also in this case, a check on the values of the momentum thickness based on the Reynolds number and the threshold indicates whether the boundary layer on the current node of the mesh is at transition or fully turbulent. The end of transition is located on the node where Re_{δ_2} is greater than $\mathrm{Re}_{\delta_{2_\mathrm{E}}}$.

$$\mathrm{Re_L} = 16.8 \left(\mathrm{Re}_{\mathrm{S_{trans}}}\right)^{0.8} \tag{4.9}$$

$$\mathrm{Re}_{\delta_{2_\mathrm{E}}} = 540 + 183.5 \left(\mathrm{Re_L}\ 10^{-5} - 1.5\right)(1 - 1.4\lambda) \tag{4.10}$$

The presented method for locating the boundary layer transition can be summarized in the pseudo algorithm 3.

Once the flow regime has been found and the transition located, it is possibile to compute the desired flow properties, like the shear stress and the heat transfer coefficient. The shear stress is computed by means of the procedure developed by Cebeci and Bradshaw [16]. In the laminar regime, the skin friction coefficient can be expressed as a function on the Reynolds number based on δ_2 and a function of the nondimensional parameter λ reported in the equation 4.5:

$$C_{f_\mathrm{lam}} = \frac{2\ l(\lambda)}{\mathrm{Re}_{\delta_{2_\mathrm{lam}}}} \tag{4.11}$$

where:

$$l(\lambda) = \begin{cases} 0.225 + 1.61\lambda - 3.75\lambda^2 + 5.24\lambda^3 & 0 < \lambda < 0.1 \\ 0.225 + 1.472\lambda - (0.0147\lambda)/(\lambda + 0.107) & -0.1 < \lambda < 0 \end{cases} \tag{4.12}$$

In the turbulent regime, the skin friction coefficient depends on the Reynolds number based on the turbulent momentum thickness according to the following formula:

$$\frac{C_{f_\mathrm{turb}}}{2} = 0.0125\ \mathrm{Re}_{\delta_{2_\mathrm{turb}}}^{-0.25} \tag{4.13}$$

The computation of the convective heat transfer coefficient is based on the work led by Smith and Spalding [55] and Ambrok [4]. In 1958, Smith and Spalding found an approximate expression of the laminar heat transfer coefficient in

63

Algorithm 3 Pseudo algorithm used to locate transition and set the momentum thickness

---

1: **procedure** LOCATETRANSITION(airProperties, flowSolution)
2:     currentBoundaryID $\leftarrow$ stagnationBoundaryID
3:     currentPointID $\leftarrow$ stagnationPointID
4:     **loop** on each side of the airfoil
5:         **loop** until transitionOnsetPointID is found
6:             $\delta_2 \leftarrow$ getLaminarMomentumThickness(currentPointID)
7:             $\lambda \leftarrow$ getLambda(ν, u_e, δ_2)
8:             $F(\lambda) \leftarrow$ getF(λ)
9:             $\text{Re}_{\delta_{2_\text{trans}}} \leftarrow$ getReynoldsThresholdOnset($F(\lambda)$, Tu)
10:            $\text{Re}_{\delta_2} \leftarrow$ getReynoldsNumber(ν, u_e, δ_2)
11:            **if** $\text{Re}_{\delta_2} < \text{Re}_{\delta_{2_\text{trans}}}$ **then**
12:                $\delta_2(\text{currentPointID}) \leftarrow \delta_2$
13:            **else**
14:                $\delta_2(\text{currentPointID}) \leftarrow \delta_2$
15:                transitionOnsetBoundaryID $\leftarrow$ currentBoundaryID
16:                transitionOnsetPointID $\leftarrow$ currentPointID
17:            **end if**
18:         **end loop**
19:         currentBoundaryID $\leftarrow$ transitionOnsetBoundaryID
20:         currentPointID $\leftarrow$ transitionOnsetPointID
21:         **loop** until transitionEndPointID is found
22:             $\delta_2 \leftarrow$ getTurbulentMomentumThickness(currentPointID)
23:             $\lambda \leftarrow$ getLambda(ν, u_e, δ_2)
24:             $\text{Re}_{s_\text{trans}} \leftarrow$ getReynoldsNumber(ν, u_e, s_trans)
25:             $\text{Re}_\text{L} \leftarrow$ getReL
26:             $\text{Re}_{\delta_{2_\text{E}}} \leftarrow$ getReynoldsThresholdEnd($\lambda, \text{Re}_\text{L}$)
27:            **if** $\text{Re}_{\delta_2} < \text{Re}_{\delta_{2_\text{E}}}$ **then**
28:                $\delta_2(\text{currentPointID}) \leftarrow \delta_2$
29:            **else**
30:                $\delta_2(\text{currentPointID}) \leftarrow \delta_2$
31:                transitionEndBoundaryID $\leftarrow$ currentBoundaryID
32:                transitionEndPointID $\leftarrow$ currentPointID
33:            **end if**
34:         **end loop**

35:	currentBoundaryID ← transitionEndBoundaryID
36:	currentPointID ← transitionEndPointID
37:	**loop** until trailing edge
38:	δ_2 ← getTurbulentMomentumThickness(currentPointID)
39:	δ_2(currentPointID) ← δ_2
40:	**end loop**
41:	**end loop**
42:	**end procedure**

a two-dimensional or axisymmetric flow. In particular, the heat transfer coefficient at the airfoil stagnation point can be found through the calculation of the local Nusselt number as follows:

$$
\mathrm{Nu_{stag}} = \left(0.246 \ \mathrm{Re}_\infty \ \left. \frac{d(u_e/V_\infty)}{d(s/c)} \right|_{s=stag} \right)^{0.5}
\tag{4.14}
$$

For the laminar flow, Ambrok developed a semi-empirical equation to evaluate the heat transfer coefficient over a non-isothermal surface with a moderate pressure gradient:

$$
\mathrm{Nu_{lam}} = 0.3 \ \mathrm{Re}_s \ \Delta T \left(\int_{s_{stag}}^{s} \frac{u_e \ \Delta T^2}{\nu_{air}} ds \right)^{-0.5}
\tag{4.15}
$$

where $\mathrm{Re_s}$ is the Reynolds number based on the curvilinear abscissa and ΔT is the difference between the surface temperature and the local air temperature. In a numerical simulation of the performances of an IPS in anti-icing conditions, the temperature of the surface is not known and it is an ouput of the simulation. Therefore the influence of the surface temperature on the calculation of the air convective heat transfer coefficient suggests that the problem is implicit and has to be solved through iterations until the solution update, defined as the difference of temperature between two iterations, is below a given tolerance. That means that an initial value for the surface temperature must be assigned.

The local air convective heat transfer coefficient in the turbulent regime is again given by Ambrok through the computation of the local Stanton number:

$$
\mathrm{St_{turb}} = 0.0125 \ \mathrm{Re}_{\Delta_{2_{turb}}}^{-0.25} \ \mathrm{Pr}^{-0.5}
\tag{4.16}
$$

The computation of St_{turb} requires to know $Re_{\Delta_{2_{turb}}}$, that is the Reynolds number based on the enthalpy thickness, and it is given by an approximation introduced by Ambrok:

$$Re_{\Delta_{2_{turb}}} = \frac{1}{\Delta T} \left[0.0156 \, Pr^{-0.5} \, \mu^{-1} \int_{s_{tr}}^{s} \rho_e u_e \, \Delta T^{1.25} ds + \left(Re_{\Delta_{2_{tr}}} \, \Delta T_{tr} \right)^{1.25} \right]^{0.8}$$
(4.17)

where $Re_{\Delta_{2_{tr}}}$ and ΔT_{tr} are the Reynolds number based on the enthalpy thickness and the enthalpy thickness at the transition onset. The former is given through the laminar expression, evaluated at the transition onset:

$$Re_{\Delta_{2_{tr}}} = \frac{0.83}{\Delta T_{tr}} \left(\int_{s_{stag}}^{s_{tr}} \frac{u_e \, \Delta T^2}{\nu_{air}} ds \right)^{0.5}$$
(4.18)

All the equations reported so far hold either in the laminar or in the turbulent regime. The method based on the boundary layer momentum thickness allowed to locate the transition and represent it as a region with a defined length where the flow goes from fully laminar to fully turbulent. Reynolds et al. [46] defined the transition region statistically using a mean position and a standard deviation length. Experiments led by Schubauer and Klebanoff at NACA in 1955 [50], showed that the laminar boundary layer on a flat plate became turbulent quite suddenly at various locations on the plate. If the plate is divided into sections of finite thickness, as shown in figure 4.4, transition occours abruptly at some point and the location of the transition onset differs from section to section. If the sections' thickness is made infinitesimaly small, the locus of transition points becomes a continuous curve. The location of transition in any section is described by a probability distribution representing the probability that transition occours at a certain point. As can be seen in figure 4.4, the probability distribution is approximately normal, therefore a gaussian cumulative distribution function (CDF) is used and it goes from zero, in fully laminar regime, to unity in fully turbulent. The transition region is defined through the Reynolds numbers calculated at the transition onset and the transition end. Therefore, given a local Reynolds number based on the curvilinear abscissa, it is possibile to compute the probability of the turbulent boundary layer to apper at position s by means of the gaussian CDF:

$$\gamma(s) = \frac{1}{2} \left(1 + \text{erf} \frac{Re_s - \mu}{\sigma\sqrt{2}} \right)$$
(4.19)
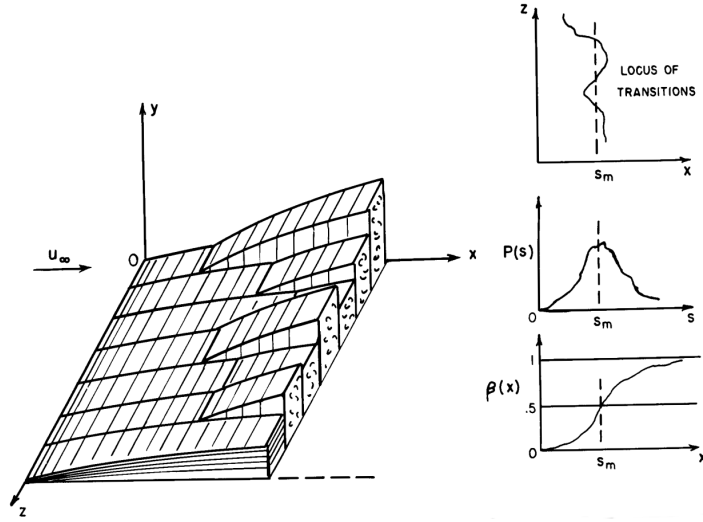
Figure 4.4   Transition model according to Reynolds et al. [46]

where the mean μ is the average between the Reynolds numbers at the onset and at the end. Reynolds et al. [46] defined the laminar region as the locus of points where the turbulent boundary layer has less than 3% of probability to appear, so for $s < s_m - 2\sigma$ as shown in figure 4.5. Thus, evaluating the equation 4.19 at the transition onset, is it possible to obtain the value of the standard deviation:

$$0.03 = \frac{1}{2}\left(1 + \operatorname{erf}\frac{\operatorname{Re}_{s_{\text{trans}}} - \mu}{\sigma\sqrt{2}}\right) \longrightarrow \sigma = \frac{\operatorname{Re}_{s_{\text{trans}}} - \mu}{\operatorname{erf}^{-1}(-0.94)\,\sqrt{2}} \qquad (4.20)$$

In the transition model, both the heat transfer and the skin friction coefficient are evaluated in the transition region by a linear combination of their values in the laminar and in the turbulenr regime. Equation 4.21 reports the expression for the heat transfer coefficient. The skin friction coefficient is obtained by replacing h by C_f in the following:

$$h(s) = \begin{cases} h_{\text{lam}}(\operatorname{Re}_s) & s \leq s_m - 2\sigma \\ [1 - \gamma(s)]\ h_{\text{lam}}(\operatorname{Re}_s) + \gamma(s)\ h_{\text{turb}}(\operatorname{Re}_s) & s_m - 2\sigma < s < s_m + 2\sigma \\ h_{\text{turb}}(\operatorname{Re}_s) & s \geq s_m + 2\sigma \end{cases}$$
$$(4.21)$$
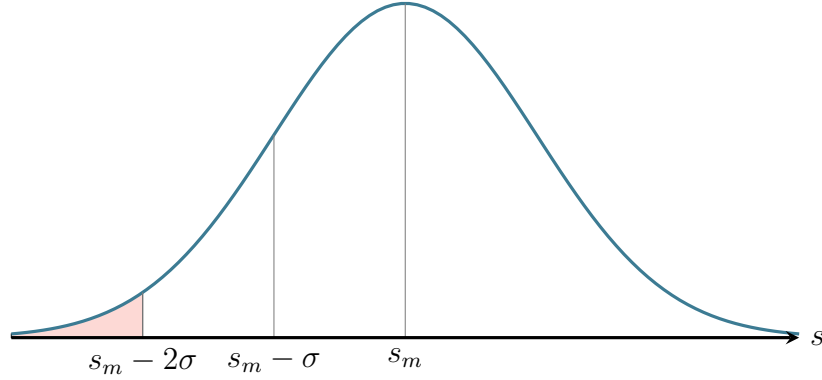
Figure 4.5 Normal distribution. The highlighted area corresponds to a probability of occurence smaller than 3%

4.3.2 Evaporation model

The mathematical model of an IPS in anti-icing conditions must accurately describe the evaporation process as it represents one of the main contribution among all the involved heat fluxes. A reliable description of the evaporation process is one of the key points in the study of an IPS performance.

The mass of water lost due to evaporation is calculated observing that the water film transfers heat and mass to the airflow. Therefore, there exists a region of water-air mixture through which the mass transfer takes place. Figure 4.6 shows that two fictitious surfaces, S and G, can be pointed out in the problem. The G surface separates the region in which the air properties differ appreciably from that in the bulk of the flow while S is the surface across which the mass transfer occurs. In particular, the G-state consists of air at temperature T_G and vapour content m_{w_G} while state S consists of air at temperature T_S higher than T_G and vapour content m_{w_S}. It is assumed that at the interface between the liquid and the gas there is a thin layer of saturated air at the temeperature of the film. In this section, the subscript w is used in place of H_2O for ease of reading. The differences in the thermodynamic states of the fluid at the described surfaces suggest that a mass transfer takes place in the direction of the arrows. According to Spalding [57], the mass transfer rate across a surface is determined by an Ohm's Law-type relation:

$$\dot{m}_e'' = g_m \, B_m \tag{4.22}$$

68
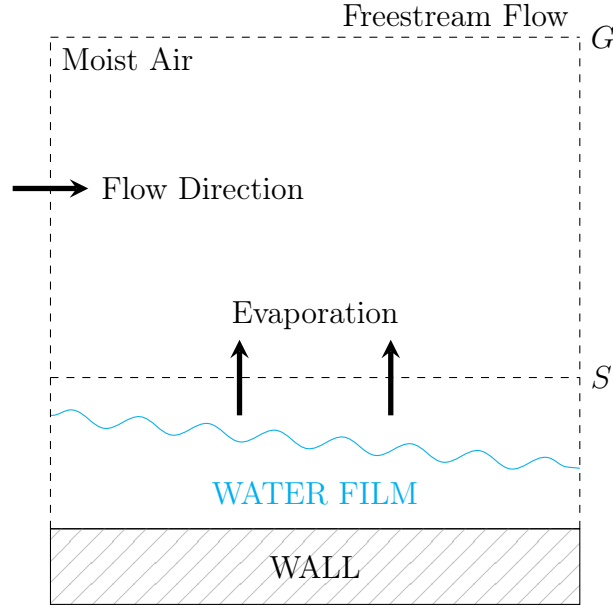
Figure 4.6 Control volume at the water-air interface for the mass transfer problem

where g_m represents a conductance which is deducible by means of the heat transfer theory and B_m a driving force which involves the properties of the fluid states. The apex $('')$ stands for flux, thus $\dot{m}_e''$ is a mass flow rate per unit surface. Spalding derives an expression of the driving force applying the mass conservation law to water in the control volume reported in figure 4.6. The result shows that B_m is related to the water concentrations by the equation:

$$B_m = \frac{m_{w_G} - m_{w_S}}{m_{w_S} - 1} \tag{4.23}$$

An adequate description of the thermodynamic state of this system can be obtained by assuming perfect-gas behavior and using only the Gibbs-Dalton laws for gas mixtures and a table of vapor pressure of water at some temperature. Kays and Crawford [33] combined these relations and obtained the following relation for the mass concentration of water vapour in the mixture:

$$m_w = \frac{p_w}{1.61p - 0.61p_w} \tag{4.24}$$

where p_w represents the partial pressure of the water vapour while p is the total pressure of the mixture.

Spalding obtained also a relation used to compute the conductance g_m, which

is reported below:

$$g_m = \text{St } \rho u_e \text{ Le}^{2/3} \frac{\ln(1 + B_m)}{B_m} \tag{4.25}$$

where Le is the Lewis number and St the Stanton number. The latter is directly proportional to the air heat transfer coefficient and this explains why an accurate prediction of h_{air} is also necessary for a precise calculation of the evaporative heat flux. The reader is referred to the book of Spalding [57] for a detailed description on the hypothesis and steps leading to equation 4.25.

4.3.3 Liquid film model

A detailed model for the water film forming on a protected airfoil is a crucial part in the development of a numerical model for an IPS. In chapter 2 the water film has been introduced within the description of the glaze ice accretion models but its properties have not been discussed in detail as it has been treated as isothermal and with constant thickness. In chapter 3, in the unsteady model, the second hypothesis has been relaxed but the film has been considered again as isothermal in the streamwise direction because it has been treated the case of a thin liquid film flowing over an iced surface. These hypothesis do not hold in the present case as the heating power provided by the heating elements might lead to the formation of a film with a not negligible thickness and with a temperature which can be very different from point to point. The model discussed in this section can be found in the work of Silva et al. [53] and it is based on the exact solution of the Navier-Stokes equations for an incompressible flow with homogeneous density, reported below for completeness:

$$\begin{cases} \nabla \cdot \boldsymbol{u} = 0 \\ \dfrac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\, \boldsymbol{u} - \nu \nabla^2 \boldsymbol{u} = -\dfrac{1}{\rho} \nabla p \end{cases} \tag{4.26}$$

Figure 4.7 shows a detail of the water film with the the reference system and the symbols adopted in the following discussion. For a thin water film, the velocity vector has a component only in the s direction and thus the incompressibilty constraint indicates that the velocity is constant along s:

$$\boldsymbol{u} = u(y)\boldsymbol{s} \tag{4.27}$$
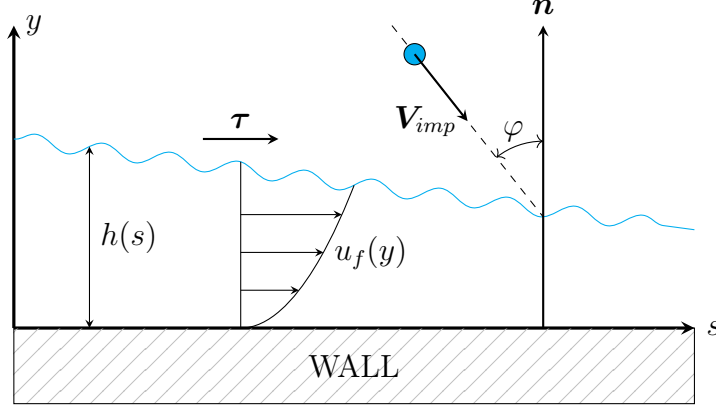
Figure 4.7   Water film detail

As the considered problem is steady, the momentum balance leads to a simple balance between the pressure gradient and the viscous forces:

$$\frac{1}{\rho_{H_2O}} \frac{dp_e}{ds} = \nu_{H_2O} \frac{d^2 u_f}{dy^2} \tag{4.28}$$

where from now on the subscript f stands for film and p_e the pressure at the air-water interface as the pressure in the film in the normal direction is constant from the y component of the momentum balance equation. The water film is driven by the shear force. For slow water speed compared to airflow, the boundary layer solution is approximately the same as the one over a stationary wall [37]. Thus the shear force is the sum of the wall friction τ at the water-air interface and the momentum per unit area from incoming water droplets. The equation 4.28 is solved applying a no-slip condition at the wall and the condition on shear stresses at the water-air interface, indicated with h:

$$\begin{cases} u_f(s, 0) = 0 \\ \mu_{H_2O} \dfrac{du_f}{dy}\bigg|_{y=h} = \tau + \dot{m}''_{imp} V_{imp} \sin(\varphi) \end{cases} \tag{4.29}$$

Integrating twice the equation 4.28 and applying the boundary conditions 4.29, a parabolic velocity distribution at every s is obtained:

$$u_f(s, y) = \frac{1}{2\,\mu_{H_2O}} \frac{dp_e}{ds} y^2 + \frac{1}{\mu_{H_2O}} \left(\tau + \dot{m}''_{imp} V_{imp} \sin(\varphi) - h(s) \frac{dp_e}{ds} \right) y \tag{4.30}$$

71
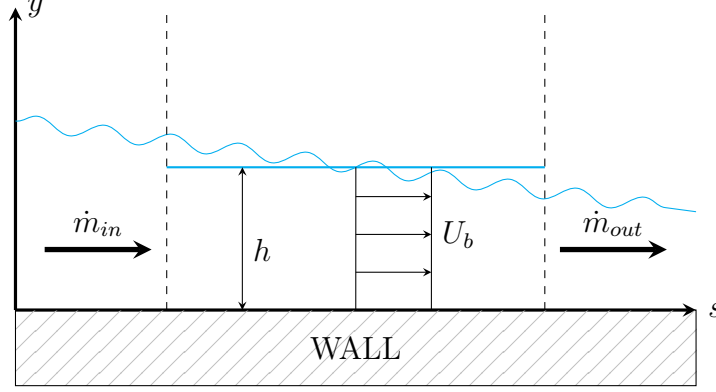
Figure 4.8 Treatment of the film thickness in a finite volume

The thickness of the film is calculated considering a mean velocity profile and a mean mass flux of water. In the control volume in figure 4.8, a mean mass flux is defined averaging the incoming and outgoing flux of water. The mean velocity is obtained by means of the definition of bulk velocity:

$$
\begin{aligned}
U_b &= \frac{1}{h} \int_0^h u_f(s,y) dy \\
&= \frac{1}{6 \, \mu_{H_2O}} \frac{dp_e}{ds} h^2 + \frac{1}{2 \, \mu_{H_2O}} \left(\tau + \dot{m}''_{imp} V_{imp} \sin(\varphi) - h \frac{dp_e}{ds} \right) h
\end{aligned}
\tag{4.31}
$$

Therefore, the mean thickness of the film is determined by the mass conservation across the control volume considering a mean mass flux flowing with the bulk velocity:

$$
\frac{\dot{m}_{in} + \dot{m}_{out}}{2} = \rho_{H_2O} \, h \, U_b \longrightarrow h = \frac{\dot{m}_{in} + \dot{m}_{out}}{2 \, \rho_{H_2O} \, U_b}
\tag{4.32}
$$

Substituting the equation 4.32 in the definition of bulk velocity 4.31, a cubic algebraic equation in the unknown h is obtained and allows to compute the mean film thickness within a control volume:

$$
-\frac{1}{3 \, \mu_{H_2O}} \frac{dp_e}{ds} h^3 + \frac{1}{2 \, \mu_{H_2O}} \left[\tau + \dot{m}''_{imp} V_{imp} \sin(\varphi) \right] h^2 - \frac{\dot{m}_{in} + \dot{m}_{out}}{2 \, \rho_{H_2O}} = 0
\tag{4.33}
$$

Once the thickness is know, it is possible to compute the mean velocity through the equation 4.31.
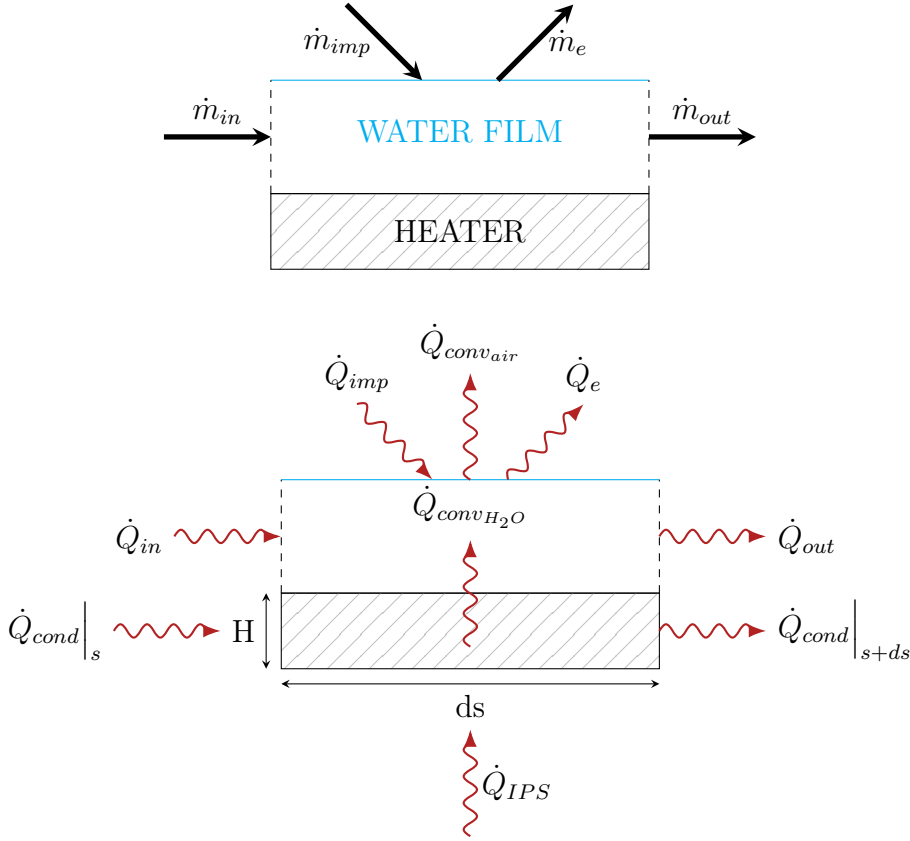
Figure 4.9 Contributions to mass and energy conservations in a finite volume

4.3.4 Thermodynamic model

Once the models for computing heat transfer coefficients, evaporating mass and water film properties have been explained, is it possible to build the thermodynamic problem which represents the core of the electro-thermal anti-ice model. The airfoil surface is discretized into finite volumes and within a volume the mass conservation of water must hold. Figure 4.9 shows all the contributions involved in the mass transfer therefore the mass conservation yields:

$$\dot{m}_{in} + \dot{m}_{imp} = \dot{m}_{out} + \dot{m}_e + \dot{m}_{ice} \tag{4.34}$$

The equation 4.34 includes also the possibility of formation of runback icing. The nature of the problem allows to consider the energy conservation law in both the water film and the surface of the airfoil. The equations, in the unknowns T_{wall} and T_{H_2O}, are linked through the water convective heat flux.

The energy conservation applied on the airfoil surface takes into account only the conduction in the direction parallel to the surface due to the small thickness of the heating elements. The thermal conductivity can be estimated by means of an electrical analogy considering the thermal resistance of the materials forming the heating element in parallel, from the heater to the external surface on the top. Moreover it is possible that within a control volume, the heat might be transferred from the surface by convection only to water, air, or to both of them, depending on the area of the surface element covered by water. Therefore the wetness factor F is defined to represent the wetted area fraction in the finite volume:

- $F = 1$ if the surface of the element is fully wet.

- $0 < F < 1$ if the element is partially wet.

- $F = 0$ if the surface of the element is fully dry.

Taking into account the possibility of a variable thermal conductivity or a variable thickness of the heating elements in the streamwise direction, the energy conservation on the airfoil surface yields:

$$\frac{d}{ds}\left(k_{wall}\ H\ \frac{dT_{wall}}{ds}\right) - F\ h_{H_2O}\ (T_{wall} - T_{H_2O}) + \dot{q}''_{IPS}$$
$$- (1 - F)\ [h_{air}\ (T_{wall} - T_{rec})] = 0 \tag{4.35}$$

where T_{rec} is the recovery temperature and takes into account the effects of aerodynamic heating in the boundary layer. It is defined as:

$$T_{rec} = (1 - r)\ T_e + r\ T_0 \tag{4.36}$$

where T_e is the local temperature outside the boundary layer, T_0 the stagnation temperature and r the recovery factor. The latter depends on the boundary layer regime and it is assumed to be $\sqrt{\text{Pr}}$ for laminar flows and $\sqrt[3]{\text{Pr}}$ for turbulent flows. The Prandtl number Pr is almost constant for an ideal gas for a wide range of temperatures and its value is roughly 0.71.

By applying the energy conservation law in the water film layer and taking into account all the heat fluxes shown in figure 4.9, is it possible to obtain

the following equation:

$$
\begin{aligned}
& F\,A\,h_{air}\;(T_{rec} - T_{H_2O}) + F\,A\,h_{H_2O}\;(T_{wall} - T_{H_2O}) \\
& + \dot{m}_{in}\,c_{p_{H_2O}}\;(T_{in} - T_{ref}) - \dot{m}_{out}\,c_{p_{H_2O}}\;(T_{out} - T_{ref}) \\
& + \dot{m}_{imp}\,\left[c_{p_{H_2O}}\;(T_{imp} - T_{ref}) + \frac{V_{imp}^2}{2}\right] - \dot{m}_e\,\left[i_{l-v} + c_{p_{H_2O}}\;(T_{H_2O} - T_{ref})\right] \\
& - \dot{m}_{ice}\,\left[i_{l-s} + c_{p_{H_2O}}\;(T_{H_2O} - T_{ref})\right] = 0
\end{aligned}
\tag{4.37}
$$

where T_{ref}, 273.15 K, is the reference temperature for computing the sensible enthalpy. The temperatures T_{in} and T_{out} are the temperatures of the incoming and outgoing water respectively. The temperature of the water film in the finite volume is considered to be the average of the two, thus:

$$
T_{H_2O} = \frac{T_{in} + T_{out}}{2}
\tag{4.38}
$$

The convective heat transfer coefficient of water represents the link between the water film model and the thermodynamics of the problem. It is calculated through the Chilton and Colburn analogy which directly relates the heat transfer, mass transfer and friction coefficients and permits the prediction of an unknown transfer coefficient when one of the others is known. Its definition is reported below:

$$
\mathrm{St} = \frac{C_f}{2}\,\mathrm{Pr}^{-2/3}
\tag{4.39}
$$

Substituting the definition of Stanton number, the heat transfer coefficient of water is obtained as follows:

$$
h_{H_2O} = \frac{1}{2}\,\rho_{H_2O}\,u_f(s, h(s))\,c_{p_{H_2O}}\,C_f\,\mathrm{Pr}_{H_2O}^{-2/3}
\tag{4.40}
$$

4.4 Architecture of the model implementation

As anticipated in section 4.1, the implementation of an anti-ice numerical simulation requires the generation of a domain mesh suitable for a CFD analysis, necessary to solve the Euler equations around the airfoil. The flow field is necessary to compute the droplets trajectories and evaluate the impingement properties, like the collection efficiency. The results from the CFD simulation and the collection efficiency are then used as inputs by the thermodynamic solver developed in the previous section. First, the implementation of the

thermal and momentum boundary layer model provides the shear stress and the heat transfer coefficient around the airfoil. The thermodynamic properties of air needed within the models have to be evaluated at the recovery temperature, which is initially unknown as the flow regime has to be defined through the momentum thickness procedure. Thus, an initial value for the recovery temperature has to be given. Since also the surface temperature is unknown, an initial guess to T_{wall} has to be provided too. In PoliMIce, the guesses are set to be $T_{wall} = 303.15\,\mathrm{K}$ and $T_{rec} = 273.15\,\mathrm{K}$. Then, the mass and energy conservation laws applied to the airfoil surface and to the water film are solved in each finite volume giving as output the temperature of the wall, the temperature of the water film and the runback mass flux. From these properties, all the heat fluxes and the water film features are known. The equations 4.35 and 4.37 are non-linear as h_{air} depends on T_{wall} and h_{H_2O} depends on T_{H_2O}. The last dependence is due to the fact that h_{H_2O} is linked to the water film model through the equation 4.40 which depends on the water film thickness given by the equation 4.33. Finally, the average mass flux in a finite volume needed for the computation of the film thickness depends on T_{H_2O} through the evaporative term in the mass conservation law. Consequently, the problem is numerically solved by means of a bisection method applied to the water energy conservation 4.37: T_{wall} is known by the initial guess so the only unknown is T_{H_2O} which is also implicitly contained in h_{H_2O}, $\dot{m}_{out}$ and $\dot{m}_e$. The equation 4.37 is solved iteratively until the absolute value of the residual is below a given tolerance. In PoliMIce the tolerance is set to be equal to 10^{-6}. Once the temperature of water is known, the new temperature of the wall can be computed through the energy conservation 4.35. The aforementioned calculation is repeated into every finite volume, from the stagnation point to the trailing edge, in order to obtain the temperature distribution on the airfoil surface and all the related quantities.

Once the first iteration is performed, the recovery temperature T_{rec} is updated and the loop is repeated giving a new temperature distribution, a new value for the heat fluxes and for the water film properties. The simulation stops when the solution update, defined as the temperature difference on the wall between two iterations on a certain element, is smaller than a tolerance defined by the user. Therefore at each iteration the value of the biggest temperature difference is saved and updated till convergence. The flowchart in figure 4.10

summarizes the numerical implementation of the model. In order to avoid any misleading interpretation, the iterations of the bisection method are indicated with it_b while the iterations of the main loop with it. The function $F(T_{H_2O})$ indicates the water energy conservation law.
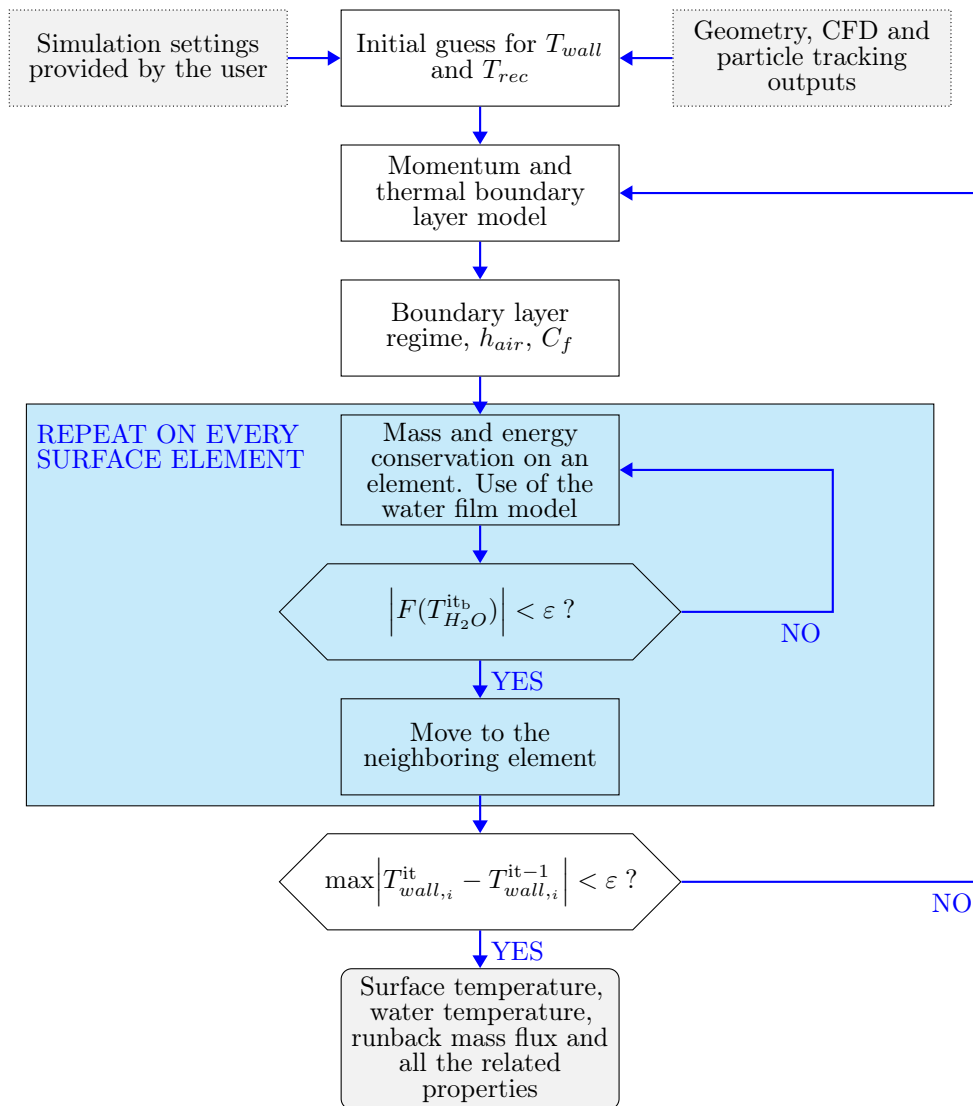
Figure 4.10 Flowchart of the numerical implementation of the model

<div align="right">

Chapter **5**

</div>

PoliMIce

PoliMIce is an highly modular software for the simulation of in-flight ice accretion and for the simulation of the performances of an electro-thermal ice protection system in anti-ice conditions. All the models reported so far have been implemented in the software. As already mentioned in chapter 1, a previous version has been coded at Politecnico di Milano by Garabelli and Gori in their master thesis [24]. Several limitations emerged from that version, therefore in this work a completely new software has been developed. The C++ language has been used to implement the code. The choice to use this programming language was dictated by the main feature of an *object-oriented programming* that is the notion of object. An object is a very general entity, in particular it is an instance of a class and the latter is a code template for creating objects. Objects of different classes can interact with each other, and that makes the implementation of a data structure very convenient. In this chapter, the reasons which led to a development of a new version of PoliMIce are explained and the code is then presented, providing an explanation of the implementation and of the data structures.

5.1 A new software for icing and anti-ice simulations

PoliMIce has been written using the mesh and geometry libraries PoliMesh and PoliGeom, developed at Politecnico di Milano by Bellosta in his master thesis [7]. The library for the geometry introduces the notion of spatial

entities like point, vectors and planes while PoliMesh contains the tools used to handle the mesh and the solution of the aerodynamic computation.

The aforementioned libraries can handle a mesh and a solution in both the OpenFOAM and SU2 formats, thus PoliMIce can be used with both CFD solvers, specifying the choice in the `PoliMIce.cfg` configuration file.

Besides the use of new libraries, the main novelty in the code is its data structure which has allowed the implementation of the new electro-thermal anti-ice system and all the models employed in the ice accretion simulation. As already anticipated in the introduction of this work, four different zones can be highlighted in an ice accretion or in an anti-ice simulation:

- **Flow region**. It is the region of the aerodynamic flow field.

- **Dispersed phase region**. It is the region of the water droplets carried by the fluid.

- **Multi-phase region**. It is the region enclosing the ice formation and the water film.

- **Body region**. It is the region of the solid surface.

Four main classes have been implemented, one for each region. Each class contains both attributes and methods, which are the data fields and functions in the C++ language. These classes construct an object for each region and, creating an instance, it is possible to allow an interaction with other classes. In this way the data structure results to be very convenient as various models for solving different kinds of problems can be attached to this framework. In this version of PoliMIce, the models reported in chapters 2, 3 and 4 are implemented but in the future other problem solvers can be added, such as a model for an ice shedding simulation.

Another important novelty is the treatment of multiple boundaries as it is possible to handle a surface splitted in different parts. The use of different geometric boundaries is very useful in an anti-ice simulation as it is possible to assign a certain value of the heat flux provided by the heating element on each boundary, as shown in figure 4.1. The number of boundaries with their tags has to be specified in the configuration file, thus the geometry has to be defined with multiple boundaries before running PoliMIce, for example during the mesh generation.

Figure 5.1 shows the PoliMIce framework. As it can be seen, PoliMIce must be coupled with other solvers to obtain flow field and particle tracking results. Ice accretion is clearly an unsteady problem: as ice grows on the surface, the aerodynamic is changed and consequently the water droplets trajectories and the ice shape are modified. Two time scales can be considered in this problem: the first is related to the flow field modification due to ice growing while the second is related to ice accretion. Since the latter time scale is much larger than the former, a quasi-steady approach is adopted: the accretion time is divided into intervals and the flow field and droplets trajectories are kept constant in each sub-interval. In this way, a series of steady aerodynamic computations are performed instead of an unsteady one, thus saving computational time. This approach is called multi-step and allows to take into account the feedback of ice accretion on the flow field.

A typical ice accretion multi-step loop is shown in figure 5.1 and it starts with the CFD simulation. The choice in this case is between two open source softwares: SU2 or OpenFOAM. Then, using the solution of the flow field computation, a Lagrangian particle tracking solver is run to compute the collection efficiency necessary for icing computation. It has been chosen to use a Lagrangian particle tracking with respect to an Eulerian one because in this way it is easier to describe the physics of the droplets moving in the flow field. Using these results, the ice accretion simulation is performed using PoliMIce. Lastly, the new geometry of the airfoil is computed considering the new ice thickness and the mesh is updated according to the new shape. This can be done either by creating a new mesh over the new surface or by deforming the existing one. The loop is performed till the total exposure time is reached.

It is clear that the geometry and the results of the CFD and particle tracking simulations must be available by previous calculations. Then, through a configuration file, it is possible to set up the PoliMIce simulation, specifing the needed parameters. A C++ class `Config` was written in order to parse the configuration file and store the settings as class members.

In the following sections the details of the implementation of each class are reported, presenting an explanation on all their dependencies.
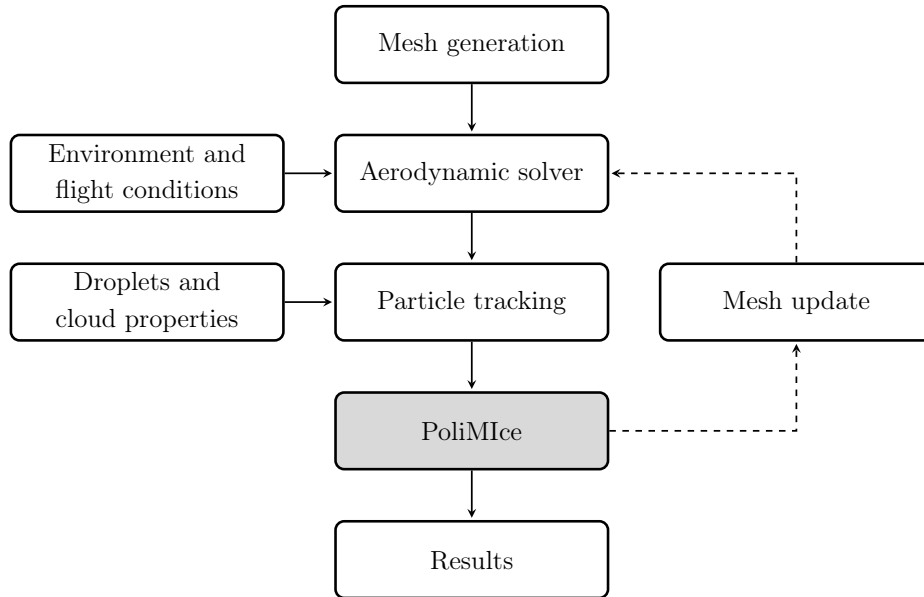
Figure 5.1   Flowchart of the PoliMIce framework. The loop is executed only in a multi-step ice accretion simulation

5.2   Data structures

As anticipated in the previous section, PoliMIce has been coded on top of two libraries which manage geometric entities, mesh and flow field solution. The dependencies of PoliMIce on the aforementioned libraries are shown in figure 5.2.

PoliGeom has been developed in the framework of the development of a robust Lagrangian particle tracking, named PoliDrop [8], to store and define geometrical functions. Its two main classes are `Point`, which defines a point in space and `Vector`, which defines the difference between two points. Operators are overloaded to allow operations between two `Vector` objects, such as sum and subtraction. Points and vectors are used to represent all the geometric entities defined in the library. Their topology is defined by means classes `Vertex`, `Curve`, `Surface` and `Volume`.

PoliMesh has been developed to import and manage different format of meshes. Thus SU2, VTK and OpenFOAM formats are supported according to the rules defining the mesh topology and its connectivity. PoliMesh can
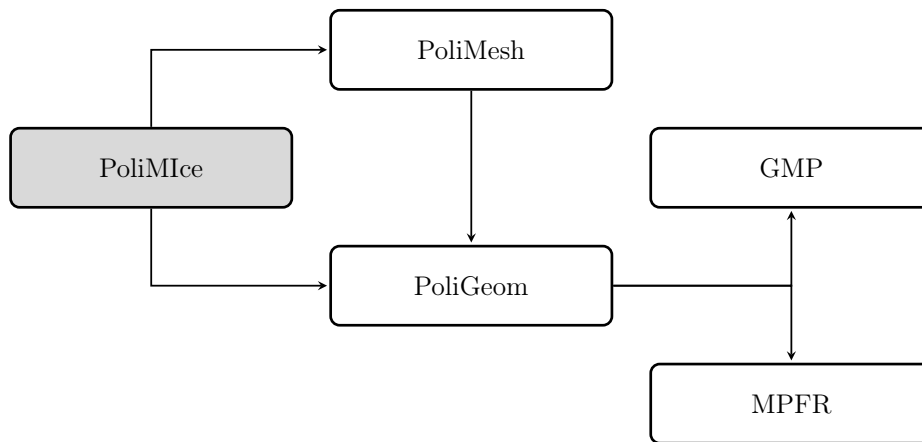
Figure 5.2 Library dependencies of PoliMIce.

handle both 2D and 3D unstructured meshes, defining elements, faces, edges and nodes. As in a CFD computation the discretization of complex geometries is performed by means of unstructured grids, the connectivity between diverse entities must be stored.

Together with the mesh, PoliMesh manages the solution of the CFD computation. The methods are implemented in different classes, in particular the classes `DiscreteSolution` and `DiscreteDomain` contain the methods to handle the solution and the computational discrete domain respectively. The class `DiscreteProblem` is the container of the whole discretized problem, containing both the discrete domain and its solution. The reader is referred to the work by Bellosta for a more detailed description of the libraries [7].

In order to treat the four regions of the simulation, four main classes have been implemented: `Flow`, `Body`, `DispersedPhase` and `MultiPhaseRegion`. Each one of these contains all the attributes and methods needed to describe its specific problem.

One of the main features of the C++ language is the so called inheritance. When some objects have the same attributes or methods but one needs further details, it is possible to create a new class, called *child class* from an existing class, called *parent class*. In this way the child class inherits all the content from the parent class and allows the implementation of further methods and attributes. Thanks to the inheritance, hierarchy is inherently defined in data structures and code duplication is avoided. In PoliMIce, inheritance has been

exploited to define the properties of water and the ice accretion models. Class `Water` is the parent class and four child classes inherit its content, one for each of the three state of matter: `SolidRime`, `SolidGlaze`, `Liquid` and `Vapour`. The different properties of the rime and glaze ice suggested to implement two separated classes for the solid state.

The ice accretion problem has been treated in a similar manner. The class `IceAccretion` is the parent class having three child classes, one for each accretion model implemented: `Myers`, `ModifiedMyers` and `Unsteady`.

Inheritance has been also used to build the structure of the solution container. As each region defines a particular problem, the result of the calculations given as outuput from each part of the simulation are stored in four classes, one for each region: `FlowSolution`, `BodySolution`, `DispersedPhaseSolution` and `MultiPhaseSolution`. All the aforementioned classes inherit the solution structure defined in the parent class `Solution`.

The solution of a discretized problem must be always coupled with the mesh on which it is defined. Therefore, on each boundary it has to be possible to access the local mesh with the related solution. The class `ProblemWrapper` builds an object containing the local mesh and a vector of pointers to the corresponding solution for each region.

The modeling of the anti-ice system reported in chapter 4 is contained in the class `AntiIce` and minor classes have been implemented to store all the numerical methods used in the code (class `Numerics`), the models to compute the water vapour pressure (class `VapourPressure`) and to store the methods used to read and print the input and output files (class `InputOutputManager`). As anticipated in the previous section, the tokenization and parsing of the `PoliMIce.cfg` configuration file is dealt by the class `Config`, which then stores the selected settings as class members.

A complete map of all the dependencies of PoliMIce classes is too complex to be shown in a diagram. Therefore, in figure 5.3 it is shown only a map of the class hierarchy described above and in the following section the most important classes are presented in detail.
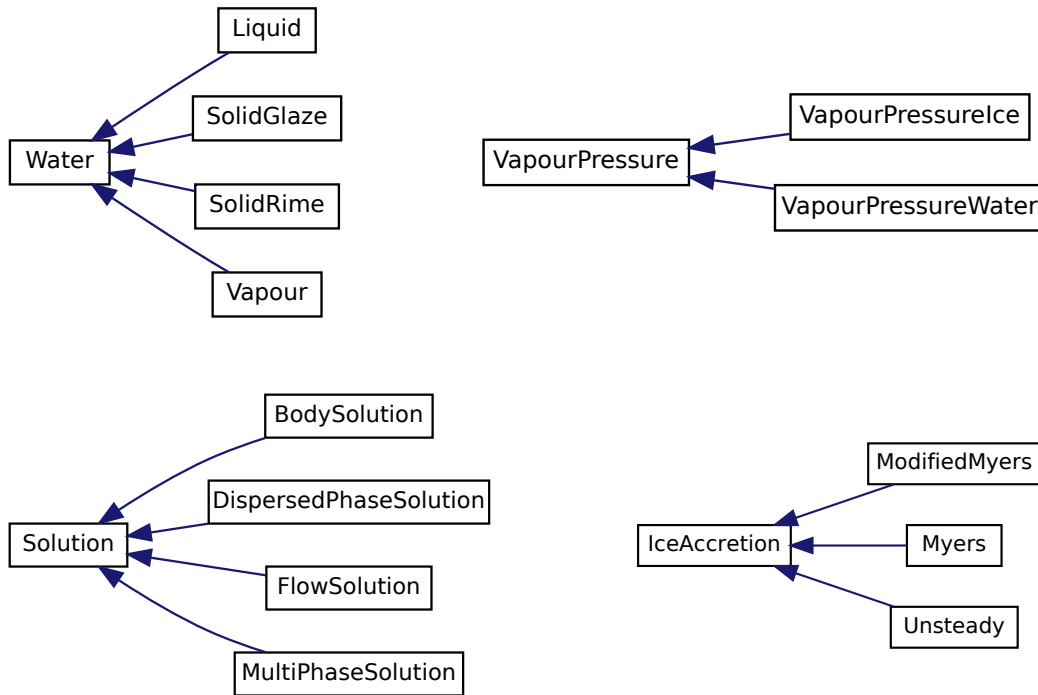
Figure 5.3 Class hierarcy of PoliMIce

5.3 Classes

The classes of PoliMIce are presented following the logical flow through which are called in the execution of the program. Only the most relevant classes are presented in detail, the others act just as containers of methods and do not need a particular explanation.

5.3.1 Config

`Config` is the first object created by the program, as it is needed to read the configuration file and store the settings selected by the user. The parsing of the configuration file is carried out by the method `Config::readConfigFile-`(`...`) which stores the options as class attributes. Inside this method, several other methods are called to perform the tokenization: when reading the file, the code searches for each keyword reading each line and dividing it into tokens. When the correct sequence of tokens is found, the method gives as

output the value corresponding to each keyword. If no value is provided and if the input field is not mandatory, the code automatically assigns the default value to the corresponding field. Otherwise it sends an error message, warning the user to provide a value. The fields requiring an option to select within a certain list are easily managed by means of the `map` template of the C++ STL library. They are associative containers, formed by a combination of a key value and a mapped value. The key values are generally used to sort and uniquely identify the elements, while the mapped values store the content associated to this key. In PoliMIce an integer number corresponds to each string option so that, when a certain option is selected, the code uniquely recognize it by associating an integer number. In this way, the method `Config::readConfigFile(...)` stores in the class every setting of the configuration file. All the attributes are set as private as this avoids accidental modifications of the low-level part of code. The data fields are thus accessed by means of methods which just return the value of the wanted private attribute. For example, if in another class it is necessary to know the selected CFD solver, the method `Config::getCFDSolver()` will return the option set by the user. All the methods needed to access the `Config` class attributes are named as `Config::getDesiredOption()`. These methods are very simple as they just return a value without performing any computation. Moreover, as they are also called many times within the code, they are qualifed as `inline`. In this way, the simple body of the function is directy substitued in the function call by the compiler, increasing speed and performances.

5.3.2 Solution

The class `Solution` defines the structure of the solution variables container. As shown in figure 5.3, three child classes inherit its properties, one for each computational region. For each problem, many variables have to be stored for each entity of the mesh, that are nodes or elements. Therefore it has been implemented a general structure that could handle different solutions with different number of variables of different types, which can be scalar and vector. The implemented structure is a *Structure of Arrays* (SoA). Instead of defining an array of length equal to the number of entities for each variable, in PoliMIce the solution is stored dynamically allocating an array of `double` type, with a size equal to the number of entities times the number of variables
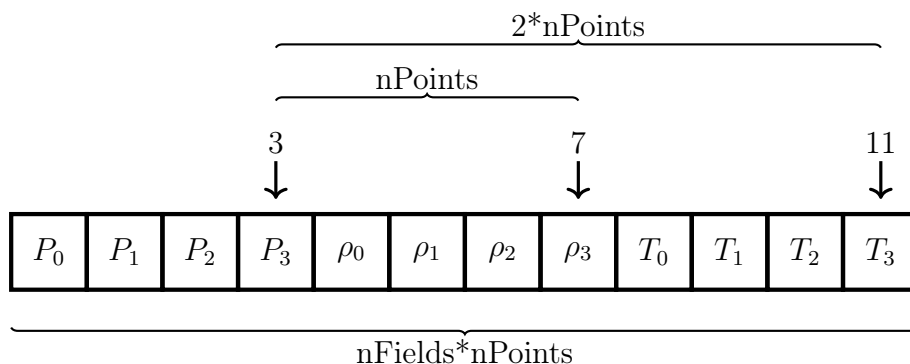
Figure 5.4 Storage of the FlowSolution variables

$(nEnt) \times (nVar)$. Even though in both cases the allocated memory is the same, initializing multiple arrays takes more instructions in terms of machine code respect to only one. Furthermore, the access to different objects can be logically slower. In the parent class `Solution`, a pointer to the solution array (`double* data`) and a few attributes, such as the number of mesh entities, number of mesh dimensions and number of variables are set as protected as they can be shared among all the three child classes only. Each child class contains a constructor, which allocates the memory for storing the specific type of solution through the operator `new` and a certain number of methods, equal to the number of solution variables, which point to a certain location of the `data` array and return the value therein saved. Variables are stored with the following logic: every variable is tagged with an index which assignes an order. A shift is then applied to each variable in order to fill the entire array. The shift is equal to the tag number times the number of entities $(nTag) \times (nEnt)$. In this way, the array is splitted into equal parts, each of them containing the values of the variable for every mesh entity. This procedure is shown in the example reported it figure 5.4, where three variables on a four entites mesh are stored in the solution structure. As the variable pressure is tagged with the index zero, the pressure at each entity of the mesh is stored at the beginning of the array `data`. The variable density is instead tagged with the index one and, since there are four mesh entities, the shift is equal to four and thus the density at each entity is stored after the pressure with a constant shift. With this procedure the whole array is filled with the whole solution of the problem.

5.3.3 Flow

`Flow` is one of the main classes of PoliMIce as it manages the flow problem and it is the first region built during the execution of the program. The importance of the class lies in the fact that, starting from the domain mesh imported by the CFD simulation, it builds as many standalone boundary meshes as many are the boundaries of the represented object, associating each mesh with its own solution. This procedure allows to discard the whole domain mesh, as it is not used in an icing simulation, and use only the mesh of the specified boundaries. This task is performed by the method `Flow::initFromDiscreteProblem(...)`: a pointer to the object `DiscreteProblem` is given as input, which loads both the mesh and the solution of the whole discretized problem, as described in the previous section. A vector of `ProblemWrapper` objects is then initialized: its length is equal to the number of physical boundaries. For every boundary the standalone mesh is then defined: a method implemented in the class `Mesh` of PoliMesh remaps the connectivity assigning a local ID to each node. In this way it is possible to define a new mesh on each boundary, with its own topology and connectivity. A pointer to `FlowSolution` is then created, allocating the memory for storing the solution. Not all the solution variables of the CFD simulation are stored in the object `FlowSolution`, but only the ones needed in a PoliMIce simulation. The array of solution is hence filled looping through all the nodes of the whole domain mesh: using the connectivity from global ID to local ID just defined, it is possible to assign to each node of the new boundary mesh the value of the solution variables on the corresponding node of the whole domain mesh. When this procedure is completed, every boundary of the flow problem will have its own local mesh with its own flow solution.

Several attibutes of the class are declareted as public and they represent the flow variables needed for the simulation:

- ID of boundary containing the stagnation point on the clean geometry.

- ID of stagnation point on the clean geometry.

- X-component of the freestream velocity.

- Y-component of the freestream velocity.

- Z-component of the freestream velocity.

- Modulus of the freestream velocity.

- Freestream temperature.

- Total temperature.

- Freestream pressure.

- Total pressure.

- Recovery factor.

As private, three matrices are declared. Two of them contain the boundary ID and the point ID of both the transition onset and of the transition end location on both sides of an airfoil. The other, named as `disjointBoundaryConnectivity`, contains the connectivity between disjoint boundaries. This connectivity allows to know how multiple geometric boundaries are connected and it is used for jumping from one boundary to its neighbour.

The class also contains all the methods used to compute the boundary layer properties reported in chapter 4, such as the transition location, the momentum thickness and shear stresses.

5.3.4 DispersedPhase

Class `DispersedPhase` contains the attributes related to the cloud properties, such as the LWC and the air relative humidity. The method `DispersedPhase::setPTSolution(...)` is in charge of uploading the solution of the particle tracking simulation and storing it in the object `DispersedPhaseSolution`. This method works similarly to `Flow::initFromDiscreteProblem(...)` previously described: a vector of `ProblemWrapper` objects is initialized to store a mesh and the solution of the particle tracking on each boundary. Then a pointer to the class `DispersedPhaseSolution` is created and the collection efficiency on each element is stored.

`DispersedPhase` contains also the method `DispersedPhase::betaSmoothing(...)` used for smoothing the collection efficiency if too much oscillating. The user can set the number of iterations for the smoothing in the configuration file and the method calculates the average of β on an element with its neighbours,

weighted on their area. If the number of iterations is high (around 10) the collection efficiency will be smooth but less accurate as their values are spread out on a greater area.

5.3.5 Body

Class `Body` contains and defines the attributes of the body problem. The method `Body::setBodyMeshAndSolution(...)` initializes the body problem. As in `DispersedPhase` and `Flow`, a vector of `ProblemWrapper` objects is initialized to store the mesh and solution of the clean geometry on each boundary. If the PoliMIce computation is performed on a clean profile, boundaries of the flow and body problem are coincident. In this case, the method calls `Body::setSolutionOnBodyMesh(...)` which sets the body problem solution on an already existing body mesh created by the `Flow` class. The method creates a pointer to the class `BodySolution` to store the heat fluxes of the IPS specified in the configuration file, the heat flux at the wall and the wall temperature. If meshes on the boundaries are instead different, such as in a multi-step ice accretion, the method calls `Body::initFromBodyMesh(...)`: the boundary meshes and the corresponding solution are initialized from the domain mesh around the clean geometry like it was done in the `Flow` class. The solution is then stored again in `BodySolution`.

5.3.6 MultiPhaseRegion

Class `MultiPhaseRegion` wraps together the three problems, body, flow and dispersed phase in a single object and defines the region where ice accretion computations are performed. The multiphase region is defined as the region included between the boundary of the flow problem and the boundary of the body problem. The region is therefore defined by the multiphase domain and two boundaries. These zones are stored in a vector of vectors of `ProblemWrapper` objects called `regions`. The first region contains the vector of `ProblemWrapper` containing the boundaries of the body while the second the ones of the flow problem. Through the external boundary of the multiphase region, which coincides with the flow, it is possible to access the `DispersedPhaseSolution`, the `MultiphaseSolution` and the `FlowSolution`.

The method `MultiPhaseRegion::setMultiPhaseRegion(...)` sets the multiphase region starting from the flow, the dispersed phase and the body problem.

The class `MultiPhaseRegion` contains methods needed for an anti-icing or icing computation, such as the computation of the heat transfer coefficient or the normals to the surface elements, needed to update the geometry once the new ice thickness is computed.

The class `IceAccretion` can access the flow, particle tracking and body solution through the `MultiPhaseRegion`. In this way ,all the quantities needed for ice accretion are accessible through only one object.

5.3.7   IceAccretion

The class `IceAccretion` is the core of the accretion solver. As shown in figure 5.3, this class has three child classes, one for each ice accretion model presented in chapter 2: `IceAccretionMyers`, `IceAccretionModMyers` and `IceAccretionUnsteady`. `IceAccretion` is an abstract class that, in addition to concrete methods, contains several pure virtual methods that are overridden in each child class. Abstraction is very useful as allows the creation of a common structure that is inherited by each derived class. In this way, the abstract class does not contain the implementation of the so-called virtual methods but it is used only to create the structure. The implementation is provided in each child class.

Some methods necessary for the ice accretion routine, such as the `geometryUpdating(...)`, are the same for all the models. Those are concrete methods that are implemented in the parent class `IceAccretion`. On the other hand, other methods such as the computation of the ice accretion rate, the rime limit or the temperature in the ice layer, depend on the chosen model. Therefore they are declared as virtual in `iceAccretion` and implemented in each child class.

The core of the accretion computation is the pure virtual method `IceAccretion::iceAccretion(...)`. It is implemented in each child class and defines the ice accretion loop. Inside this method, several other functions are called to perform the computations according to the selected model. The computation is performed through a loop that runs over all the boundaries and elements of the surfaces where ice may grow. For each element, the rime limit is first

computed with the method `rimeLimit(...)`. Then, depending on this value, the ice accretion rate and the temperature on the ice surface are computed through the methods `rimeAccretionRate(...)` and `rimeTemperature(...)` if the ice is of type rime and through the methods `glazeAccretionRate(...)` and `glazeTemperature(...)` if the ice is of type glaze. In the latter case, two more quantities must be computed which are the water film thickness and the water mass flux. These computations depend on the film model specified in the configuration file and are computed by means of some methods contained in the `WaterFilm` class. Lastly, the new solution is saved in the multiphase region and the new geometry of the iced surface is computed by the method `IceAccretion::geometryUpdating(...)`.

Figure 5.5 shows an extract of the source code of the method `Unsteady::-iceAccretion(...)` which performs the accretion loop when the unsteady accretion model is selected. The other models are not presented since they follow the same structure.

```cpp
double Unsteady::iceAccretion (...) {
    // Variables initialization
    ...
    // Loop through every MP region boundary
    for (int boundaryID=0; boundaryID<this->...nBoundaries; ++boundaryID) {
        // Initialize pointer for the new multiphase solution
        MultiPhaseSolution* newSolPtr;
        // Loop through every element of each boundary
        for (int elementID=0; elementID<this->...nElements(); ++elementID) {
            // Compute rime limit
            this->Bg = rimeLimit (...);
            // Select accretion law
            if (this->B >= this->Bg && this->Bg > 0) {
                // GLAZE ICE
                rate = glazeAccretionRate (...);
                // Compute partial and total ice thickness
                this->dB = this->dB0 + deltaB;
                this->B  = this->B0 + deltaB;
                // Compute film thickness
                this->h = this->film->waterFilmThickness (...);
                // Compute surface temperature and heat flux at wall
                this->tempSurf = this->glazeTemperature (...);
                this->qW       = heatFluxAtWall (...);
            } else {
                // RIME ICE
                rate = rimeAccretionRate (...);
                // Compute partial and total ice thickness
                this->dB = this->dB0 + deltaB;
                this->B  = this->B0 + deltaB;
                // Compute surface temperature and heat flux at wall
                this->tempSurf = rimeTemperature (...);
                this->qW       = heatFluxAtWall (...);
                // Compute similarity parameter in case next step is glaze
                this->lambda = this->B/(2 * sqrt(this->alfaIce*time));
            }
            // Fill the new solution array with the new variables
            newSolPtr ->....
        }
        // Loop through every element
        for (int elementID=0; elementID<this->...nElements(); ++elementID) {
        // Compute mass flux through elements
        this->film->massFlux (...);
        }
        // Compute error if steady iterative film model is selected
        Err = computeError (...);
        // Setting the new solution in the multiphase region
        this->mpRegion->regions [1][boundaryID].setMPSolution (newSolPtr);
        // Updating geometry
        geometryUpdating (...);
    }
    return Err;
}
```

Figure 5.5 Extract of the Unsteady::iceAccretion(...) method

93

Numerical simulations

In this chapter, numerical results for both ice accretion and electro-thermal Ice Protection System are presented in order to verify PoliMIce and the new models that have been implemented. In section 6.1 are shown the results obtained for ice accretion in various conditions, considering different film models. Results are compared with both experimental campaigns held at NASA Glenn Research Center [49] [66] [51] and numerical results obtained with LEWICE [49] [51] [66] and ONERA [66]. In section 6.2, the results of IPS simulations in anti-ice conditions are presented and compared with experimental data acquired at NASA Glenn Research Center [2] and numerical results from ANTICE and by Bu et al. and Silva et al. [13] [54].

6.1 Ice accretion test cases

Different test cases have been replicated, in order to asses the capability of the models and of the code to capture icing in diverse atmospheric conditions. The first test case in section 6.1.1 represents a mixed ice condition at 0° angle of attack as a representative test for symmetric ice accretion. Then in section 6.1.2 is presented a test case at lower temperature where ice grows as rime ice to asses the capability of the software to capture ice shapes when a very small amount or no liquid film forms on the ice surface. The third test case presented in section 6.1.3 is representetive of glaze ice condition. It has been chosen to asses the different capabilities of the three liquid film models implemented in PoliMIce to capture the liquid film motion. In section 6.1.4

a simulation of ice accretion over a long exposure time is presented. This is representetive of the case of IPS failure and a consequent large accretion of ice. For all the test cases the unsteady ice accretion model presented in chapter 2 has been used and three different liquid film models have been considered: the first one is a steady model that was described in detail by Garabelli and Gori [24], the second is the steady liquid film model presented in section 3.1 in both the iterative and implicit implementation and the last is the unsteady model presented in section 3.2.

As it was presented in chapter 5, the PoliMIce framework is composed of different software. For all the test cases presented in this chapter, the following software have been used:

- **Mesh generation**: uhMesh [22].

- **Aerodynamic solver**: SU2 6.1.0 [23].

- **Particle tracking**: PoliDrop [8].

- **Mesh update**: SU2 6.1.0 [23].

The software uhMesh is an unstructured-hybrid mesh generator developed at Politecnico di Milano [22] which allows the generation of 2D grids. The parameters of the structured mesh on the airfoil surface depend on the test case while the far-field is the same for all test cases: a circular surface of radius $R = 8$ m. An example of computational grid is shown in figure 6.1, with a detail on the airfoil leading edge.

For what concerns the boundary layer mesh, for all test cases it was necessary to set the height of the first layer of quadrilateral elements to keep the $y+$ value around one in order to resolve the viscous sub-layer without the need of wall functions that are not available in SU2. For all test cases the independence of the solution on the mesh was checked by considering different space discretizations with an increasing refinement. An example is shown in figure 6.2 where a coarse, medium and fine mesh were used to study the grid convergence of the test case presented in section 6.1.1.

The computation of the aerodynamic field was performed by solving the RANS equations for all test cases by using the software SU2, which exploits a node centered Finite Volume (FV) discretization. The convective fluxes were solved with the Roe numerical method [47] using a second order upwind
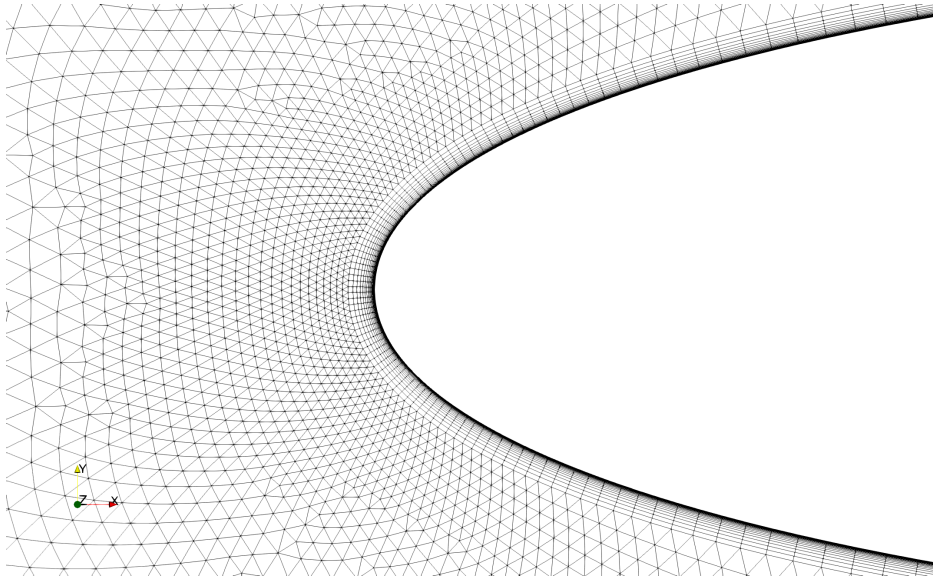
Figure 6.1 Computational grid for the ice accretion test case reported in section 6.1.1 on the airfoil leading edge

scheme with the Venkatakrishnan slope limiter [62]. Flow variables gradients are computed through a weighted least squares method at the nodes. The chosen turbulence model is the one equation model by Spalart-Allmaras [56], using an upwind scheme again with second order reconstruction with Venkatakrishnan slope limiter.

An adiabatic wall boundary condition was applied on the airfoil in each test case. At the far-field, temperature, pressure and Mach number were imposed. The specific value of these variables depends on the considered test case and they will be described in the corresponding section.

Lastly, the values of the collection efficiency were obtained using the software PoliDrop [8]. The time integration scheme used is the forward Euler for all the simulations with a time step of $dt = 10^{-5}$ s. A cloud adaptation option strategy was used. In this way, the software adapts the number of particles n_p and their distribution in space through iterations to obtain a proper collection efficiency. The final number of particles reached by the solver depends on the test case.
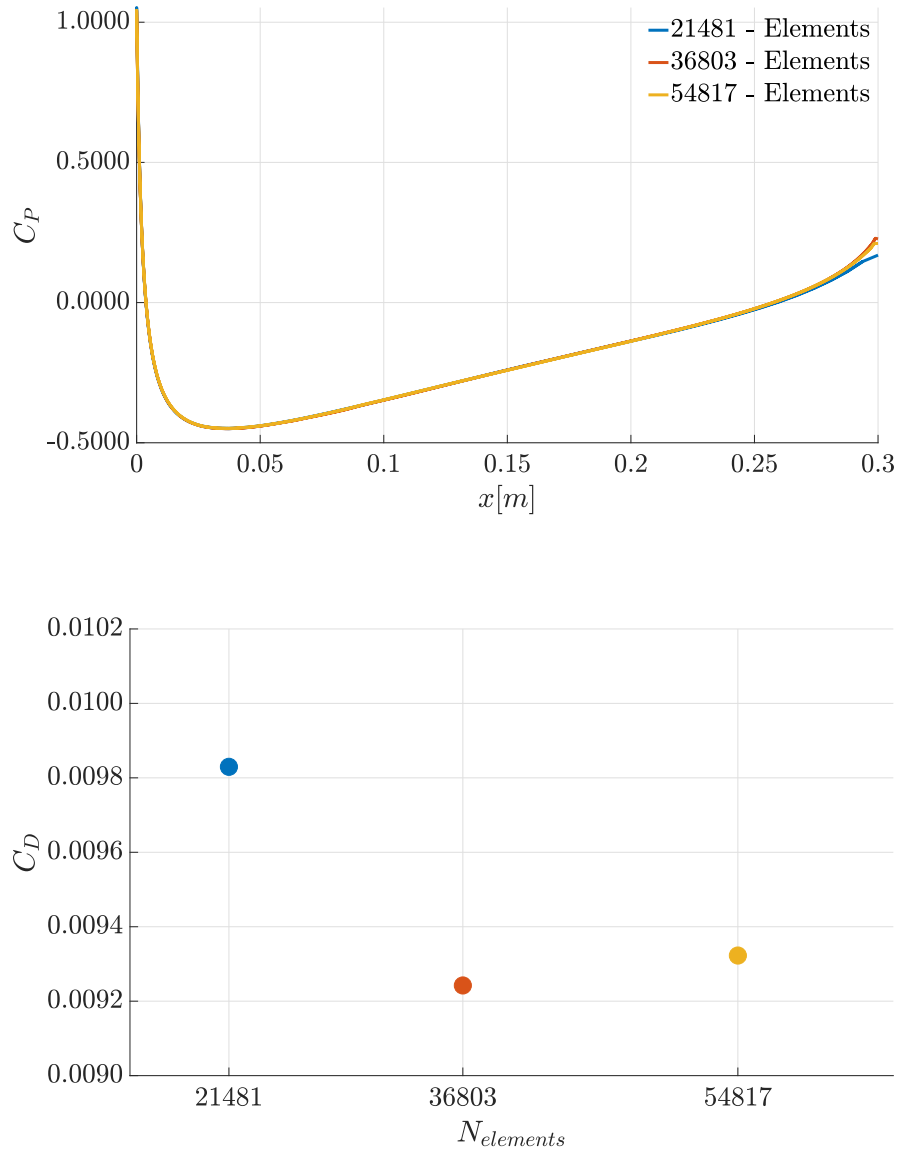
Figure 6.2 Grid convergence for the test case presented in section 6.1.1 using three increasingly finer mesh

Mixed ice	
Airfoil	NACA0012
Chord	$0.3\,\mathrm{m}$
$\boldsymbol{\alpha}$	$0°$
$\mathbf{V}_\infty$	$129\,\mathrm{m/s}$
$\mathbf{T}_\infty$	$260.55\,\mathrm{K}$
$\mathbf{P}_\infty$	$90\,700\,\mathrm{Pa}$
MVD	$20\,\mu\mathrm{m}$
LWC	$0.5\,\mathrm{g/m^3}$
t	$120\,\mathrm{s}$
$\boldsymbol{\Delta}$t	$5\,\mathrm{s}$

Table 6.1 Parameters for the mixed ice accretion test case

6.1.1 Mixed ice accretion

The first test case used for the validation of PoliMIce represents a condition of mixed ice accretion: the ice starts to form as rime but, close to the stagnation point where the effects of air compressibility are more evident, the temperature reaches higher values which lead to the formation of glaze ice. Therefore at high velocities and low temperatures both ice types coexist.

The flight and atmospheric conditions are reported in table 6.1. The total accretion time is $t = 120\,\mathrm{s}$ while flow and particle tracking computations have been performed every $5\,\mathrm{s}$. For the unsteady film model simulation, the inner time step have been set to $dt = 0.01\,\mathrm{s}$. The mesh used for this test case has 36803 elements, with 363 boundary elements on the airfoil. It has been chosen after the convergence study, observing that a finer mesh would have not improved the results.

In this test case, the number of parcels used by PoliDrop is around $n_p \approx 150000$, the actual number is slightly different at each time step because of the cloud adaptation. The collection efficiency obtained on the clean profile is shown in figure 6.3 where negative values of the curvilinear abscissa refer to the pressure side of the airfoil while positive value to the suction side. The collection efficiency is symmetric as expected for a symmetric airfoil at $0°$ angle of attack. The results obtained in PoliMIce using different liquid
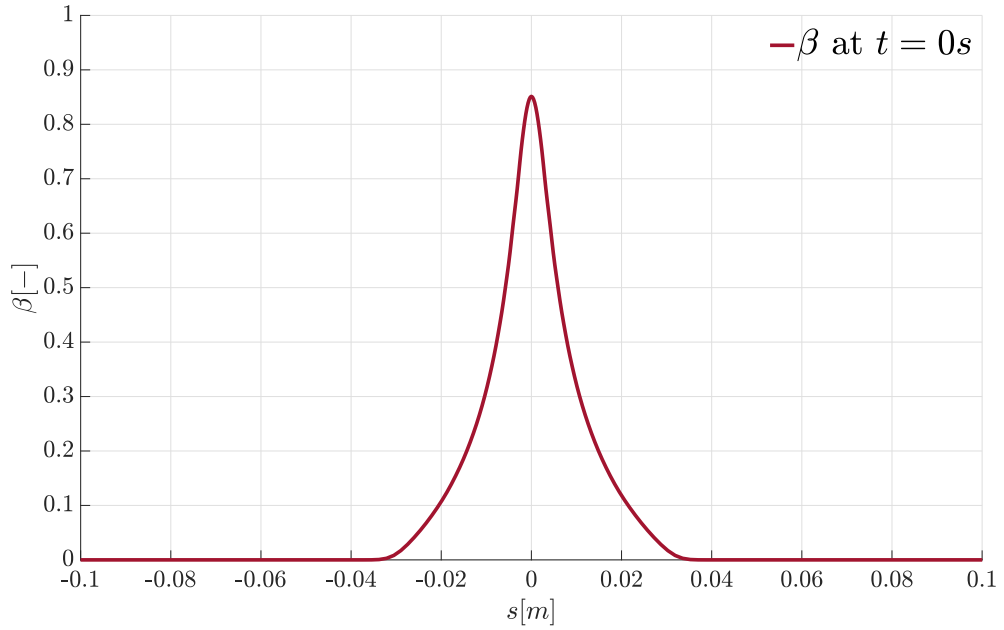
Figure 6.3 Collection efficiency on the clean profile for the mixed ice accretion test case. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

film models are shown in figure 6.4, along with the experimental shape from reference [49]. For a simulation at $0°$ angle of attack a symmetric ice shape is expected. This is clearly the case of the result obtained with the unsteady film model and the steady model in the implicit formulation. On the other hand, for the steady model in the iterative formulation and the model that was previously implemented in PoliMIce this does not happen. As it can be seen in figure 6.4, the asymmetry is more relevant in the iterative formulation of the steady film model with respect to the model previously implemented since the iterative procedure is based on the same model and numerical methods as the one that was already implemented. This causes the numerical errors that were found in PoliMIce as stated in [24] to propagate during the fixed point iterations. On the other hand, with the implicit formulation the shape is very similar to the previous implementation, but in this case the symmetry of the ice shape is achieved thanks to the different numerical method used to solve the mass flux.

It is important to notice that the experimental shape is not perfectly symmetric since there is a gravity component that acts both on the droplet trajectory and on the film movement which is not taken into account in this work. The ice

(a)

(b)

(c)
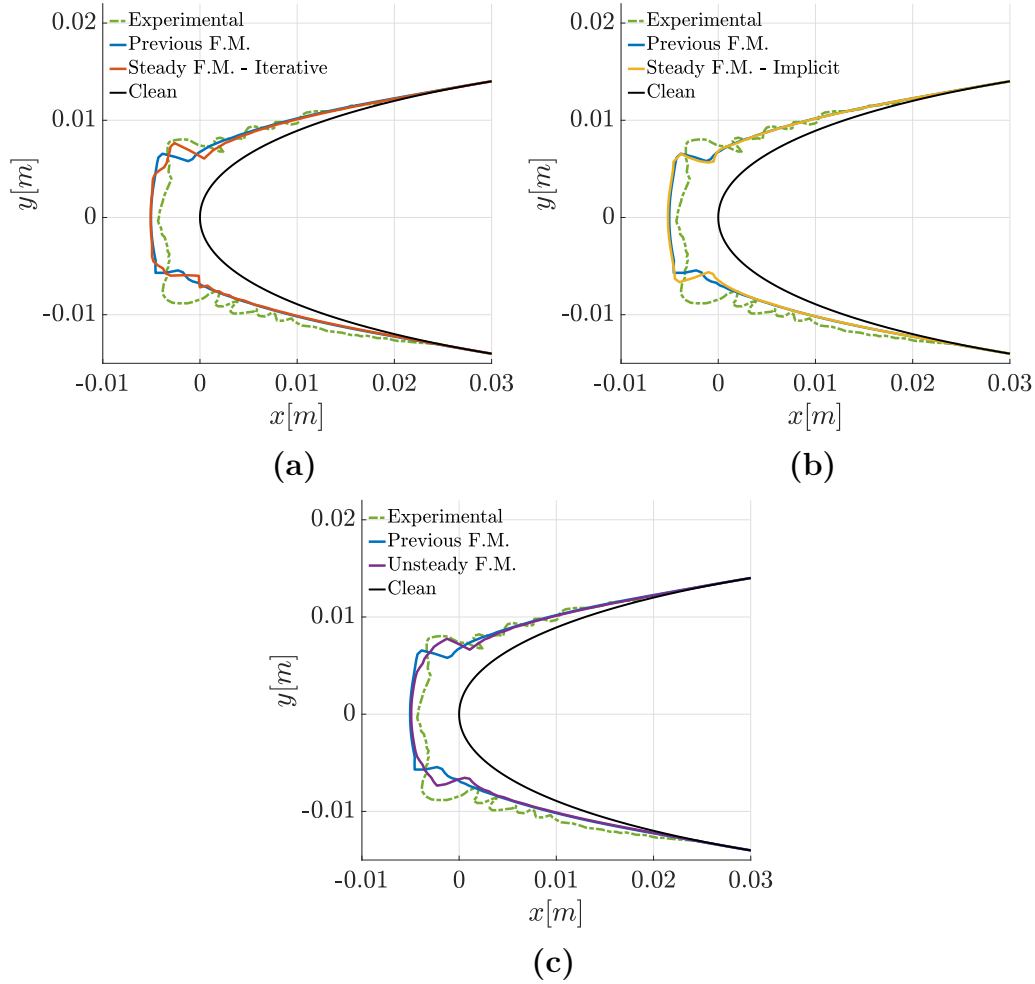
Figure 6.4 Mixed ice accretion. Comparison among experimental data [49] and PoliMIce with different liquid film models: (a) steady with the iterative implementation (section 3.1.1), (b) steady with the implicit implementation (section 3.1.2), (c) unsteady (section 3.2) and the film model previously implemented in PoliMIce [24]

accretion limits are instead accurately predicted whatever film model is used. Contrarily, the ice thickness close to the stagnation point is over predicted in all cases and it could be due to multiple reason. The main problem could be that conduction through water is taken into account on the normal direction only. This is due to the fact that in all the presented models, the temperature of water has been considered constant and equal to the freezing temperature. This means that heat fluxes caused by water moving on the surface are not considered.

The differences in the ice shape between the iterative and implicit formulation of the steady film model depends mainly on the fact that on the iterative formulation all the quantities, like surface temperature, ice thickness and others are updated at every iteration, while in the implicit formulation these quantities refer to the previous time step. This is the main backdraw of the implicit formulation: once the mass flux are computed, their influence in other quantities is not taken into account till the next computation time step. Future works could focus on considering this aspect. The main difference between the models regards the capability of capturing the so called *horns*. They are typical of glaze ice formations. As it can be noticed, the unsteady film model is able to capture the horns position on both side, although the shape is not accurately predicted. Moreover, the steady film model with the iterative implementation captures the shape of the top horn properly while the low one is cut because of the propagation of numerical errors.

An important simplification adopted in PoliMIce is that the mesh is deformed considering that ice always grows in the normal direction with respect to the clean profile. This is not what happens in reality. This simplification was adopted because, on relatively complex geometries, it avoids the mesh tangling problem. The aforementioned simplification causes the ice shapes to follow the airfoil shape and therefore to be quite smooth and it makes capturing ice irregularities more difficult.

Table 6.2 shows a comparison of the iced surface computed for the different ice shapes and it is an indication of the total volume of ice that formed on the airfoil. The surface has been computed numerically as the area enclosed between the clean airfoil and the ice shape. As it can be seen, the software LEWICE clearly underestimate the total surface of ice with an error of 17% while PoliMIce still underestimate the ice volume but the computed errors are

	Iced surface [m^2]	Err %
Experimental	1.4249×10^{-4}	-
LEWICE	1.1813×10^{-4}	17.10%
Previous F.M.	1.2195×10^{-4}	14.42%
Steady F.M. - Iterative	1.2237×10^{-4}	14.12%
Steady F.M. - Implicit	1.2286×10^{-4}	13.78%
Unsteady F.M.	1.2357×10^{-4}	13.28%

Table 6.2   Iced surface comparison for the mixed ice accretion test case

lower with respect to LEWICE. It can be noticed that the unsteady liquid film model is the one that gives the lowest error among the PoliMIce results. In figure 6.5 it is shown a comparison among the results obtained with PoliMIce with the unsteady liquid film model, the ice shape computed with LEWICE and the experimental one [49]. The unsteady film model has been chosen for this comparison because it gave the best results in terms of ice shape and surface of ice among PoliMIce results.

Lastly, figure 6.6 presents the results of the liquid film at three time steps: $t_1 = 40\,\mathrm{s}$, $t_2 = 80\,\mathrm{s}$ and $t_3 = 120\,\mathrm{s}$. In the left column of the figure the mass flow rate of water is compared among the three different models. At $t_1 = 40\,\mathrm{s}$, where the mass of water on the surface is quite low, the steady-iterative and unsteady film model give very similar results. The film model previously implemented and the implicit formulation give an underestimation with respect to the other two models. As time evolves and more water remains liquid on the surface, the models become increasingly different one from the other: the previous film model and the steady film model in the iterative implementation give an asymmetric result already at $t_2 = 80\,\mathrm{s}$ and then propagates over the next time steps. On the other hand, the unsteady film model is slightly asymmetric only at the last time step. The steady film model with the implicit implementation remains quite symmetric for all the time steps. The right column of the figure represents the water film height at time $t_1 = 40\,\mathrm{s}$, $t_2 = 80\,\mathrm{s}$ and $t_3 = 120\,\mathrm{s}$. In this case, only the unsteady film model is presented since for the other three cases the liquid film height is fixed and set equal to $h = 0.0001\,\mathrm{m}$. As it can be seen from figure 6.6, keeping the height constant with the value suggested by Myers [39] is an overestimation
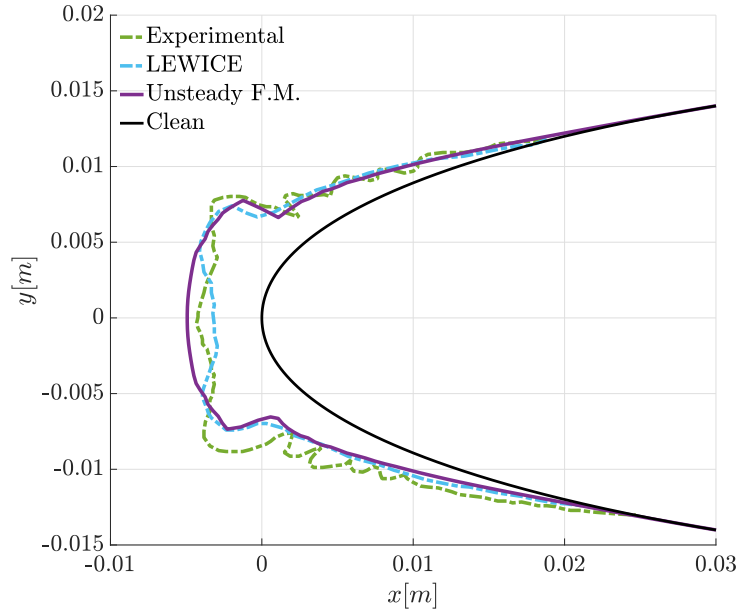
Figure 6.5 Mixed ice accretion. Comparison between experimental data and numerical results with LEWICE [49] and PoliMIce with the unsteady liquid film model (section 3.2)

for mixed ice condition because even at the last time step, where the film height is the highest, the mean value is $h \approx 2.5 \times 10^{-6}$ m. The peak observed at the stagnation point is due to the fact that there the shear stresses assume the lowest value. Therefore there is no mechanism to drive away the liquid film from that point.

6.1.2 Rime ice accretion

The second test case refers to a condition of completely rime ice accretion. The test case was conducted in the Icing Research Tunnel (IRT) at the NASA Lewis Research Center [51]. Table 6.3 presents the flight and atmospheric conditions.

The total exposure time is $t = 420$ s while CFD and particle tracking are run every 10 s. The mesh used for this test case has 71388 elements with 960 boundary elements on the airfoil. The number of parcels used by PoliDrop is around $n_p \approx 300000$ and it is slightly different at each time step due to the cloud adaptation. The collection efficiency obtained on the clean airfoil
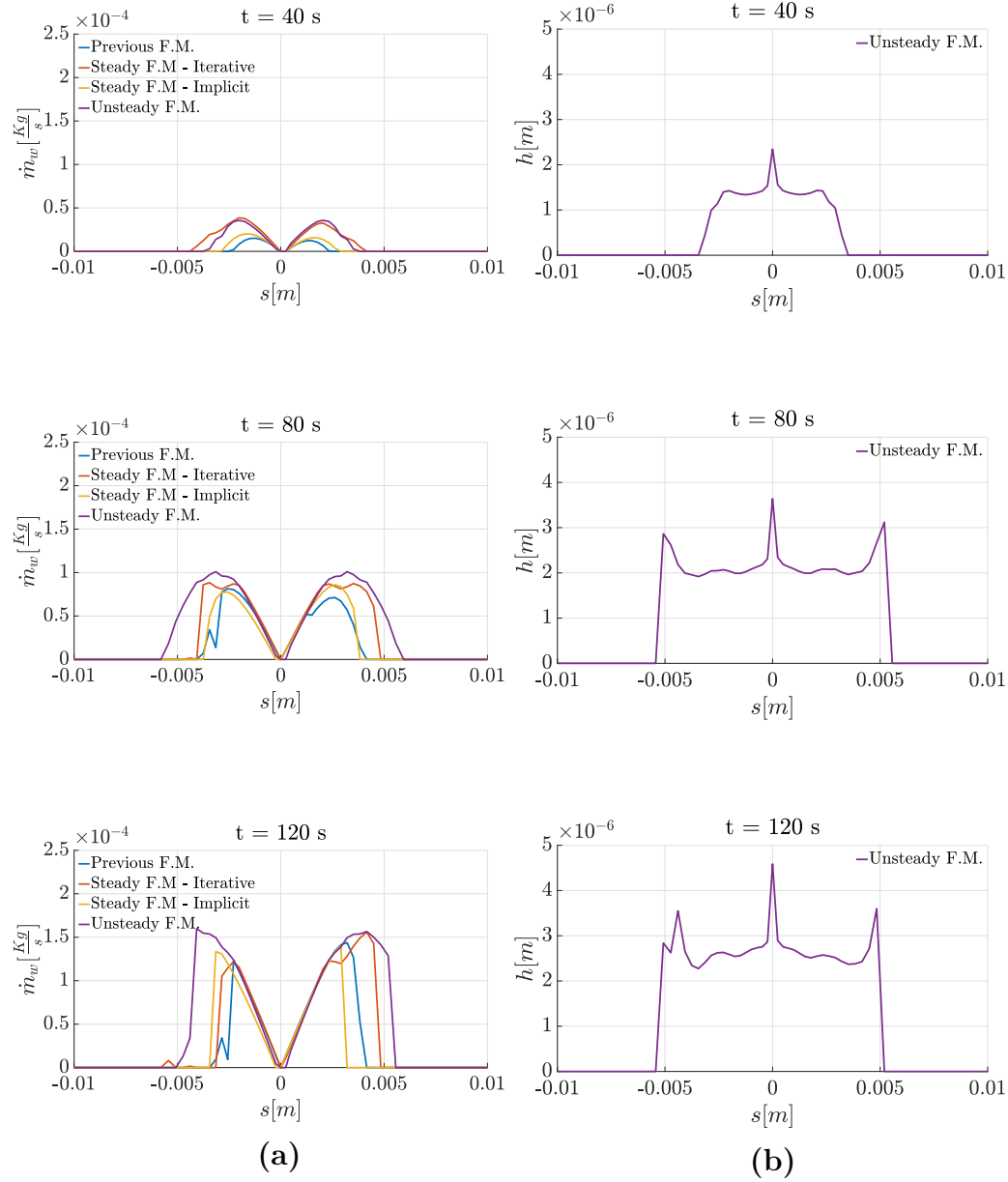
Figure 6.6 Mixed ice accretion. Comparison between: (a) water mass flux obtained with different liquid film models at $t_1 = 40\,\mathrm{s}$, $t_2 = 80\,\mathrm{s}$ and $t_3 = 120\,\mathrm{s}$ and (b) liquid film height obtained with the unsteady liquid film model at $t_1 = 40\,\mathrm{s}$, $t_2 = 80\,\mathrm{s}$ and $t_3 = 120\,\mathrm{s}$. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

Rime ice	
Airfoil	NACA0012
Chord	$0.5334\,\mathrm{m}$
$\boldsymbol{\alpha}$	$4°$
$\mathbf{V}_\infty$	$102.8\,\mathrm{m/s}$
$\mathbf{T}_\infty$	$241.40\,\mathrm{K}$
$\mathbf{P}_\infty$	$101\,300\,\mathrm{Pa}$
MVD	$20\,\mathrm{\mu m}$
LWC	$0.55\,\mathrm{g/m^3}$
t	$420\,\mathrm{s}$
$\boldsymbol{\Delta}\mathbf{t}$	$10\,\mathrm{s}$

Table 6.3 Parameters for the rime ice accretion test case

	Iced surface $[\mathrm{m^2}]$	Err %
Experimental	4.9980×10^{-4}	-
LEWICE	5.4096×10^{-4}	8.24%
PoliMIce	5.2531×10^{-4}	5.10%

Table 6.4 Iced surface comparison for the rime ice accretion test case

is shown in figure 6.7. In figure 6.8 is presented the ice shape obtained with PoliMIce and it is compared with experimental data and numerical results obtained with LEWICE. In this case only one ice shape is shown and not one for each liquid film model, as in the previous test case. This is due to the fact that the ice accretion is completely rime and therefore there is no liquid film to be modeled and all the simulations gave the same result. As can be seen from figure 6.8, the limit of ice accretion are predicted accurately while the ice thickness close to the stagnation point is once again overpredicted while the overall ice shape is quite well replicated.

Lastly, in table 6.4, the iced surface is presented to give an estimation of the total volume of ice. Again, the surface has been computed numerically as the area enclosed by the airfoil and the ice shape. In this case both software give an overestimation of the total ice surface, with PoliMIce giving a lower error of 5.10% while LEWICE gives an higher value of 8.24% with respect to the
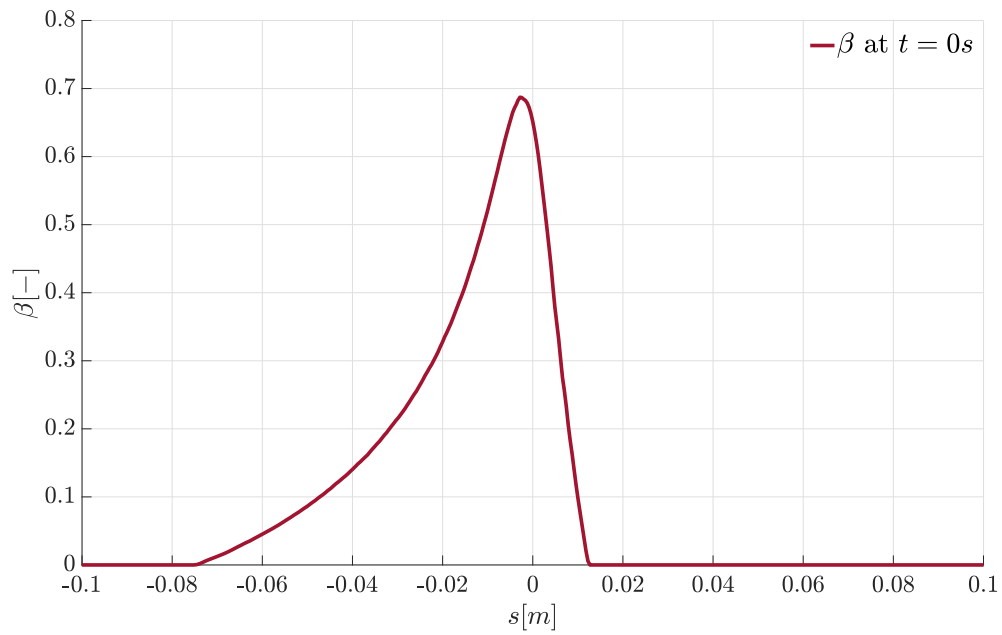
Figure 6.7 Collection efficiency on the clean profile for the rime ice accretion test case. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side
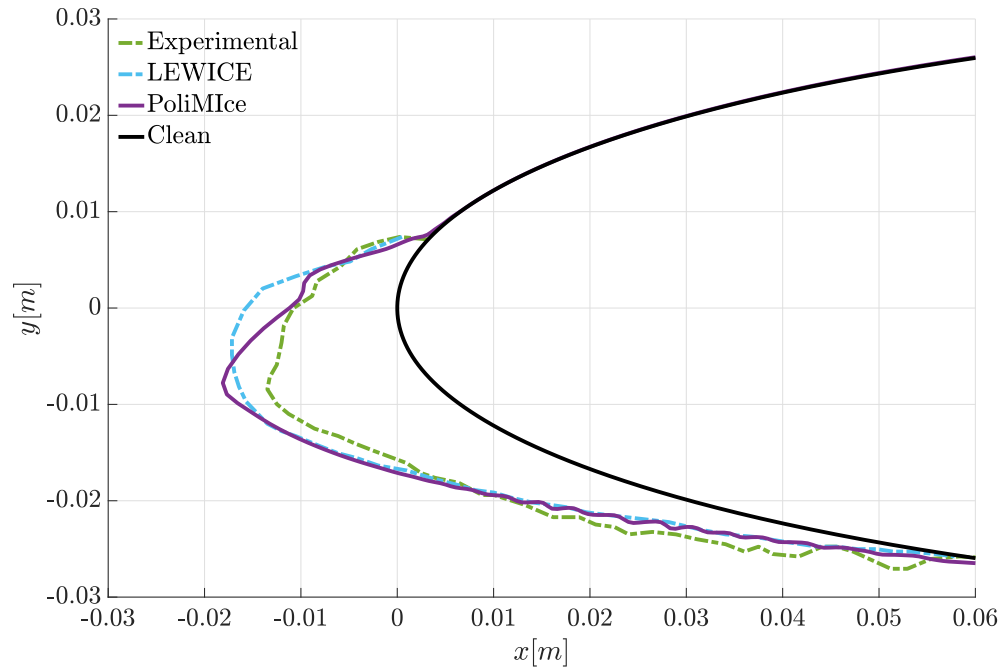
Figure 6.8 Rime ice accretion. Comparison between experimental data and numerical results with LEWICE [51] and PoliMIce

107

Glaze ice	
Airfoil	NACA0012
Chord	$0.5334\,\text{m}$
$\boldsymbol{\alpha}$	$4°$
$\mathbf{V_\infty}$	$58.1\,\text{m/s}$
$\mathbf{T_\infty}$	$269.1\,\text{K}$
$\mathbf{P_\infty}$	$95\,600\,\text{Pa}$
MVD	$20\,\mu\text{m}$
LWC	$1.3\,\text{g/m}^3$
t	$480\,\text{s}$
$\boldsymbol{\Delta}\mathbf{t}$	$10\,\text{s}$

Table 6.5 Parameters for the glaze ice accretion test case

experimental ice shape.

6.1.3 Glaze ice accretion

The third test case refers to a condition of glaze ice accretion, in order to asses the capability of the different liquid film models. The test case was conducted in the Icing Research Tunnel (IRT) at NASA Lewis Research Center [66]. Table 6.5 reports the flight and atmospheric conditions.

The total exposure time is $t = 480\,\text{s}$ while CFD and particle tracking solutions are update every $10\,\text{s}$. For the unsteady film model simulation, the inner time step has been set to $dt = 0.001\,\text{s}$. In this test case the used mesh has 37818 elements with 411 boundary elements on the airfoil.

The number of parcels reached by the solver with the cloud adaptation is around $n_p \approx 200000$ and the actual number results to be slightly different at each time step. The collection efficiency obtained on the clean profile is shown in figure 6.9. In figure 6.10 is presented the ice shape obtained with PoliMIce with different liquid film models and it is compared with experimental data from reference [66]. The ice accretion limits are very well predicted with the unsteady liquid film model and the iterative formulation of the steady model. In the case of the model previously implemented in PoliMIce and the implicit formulation, the limits are clearly underestimated. This is due to the
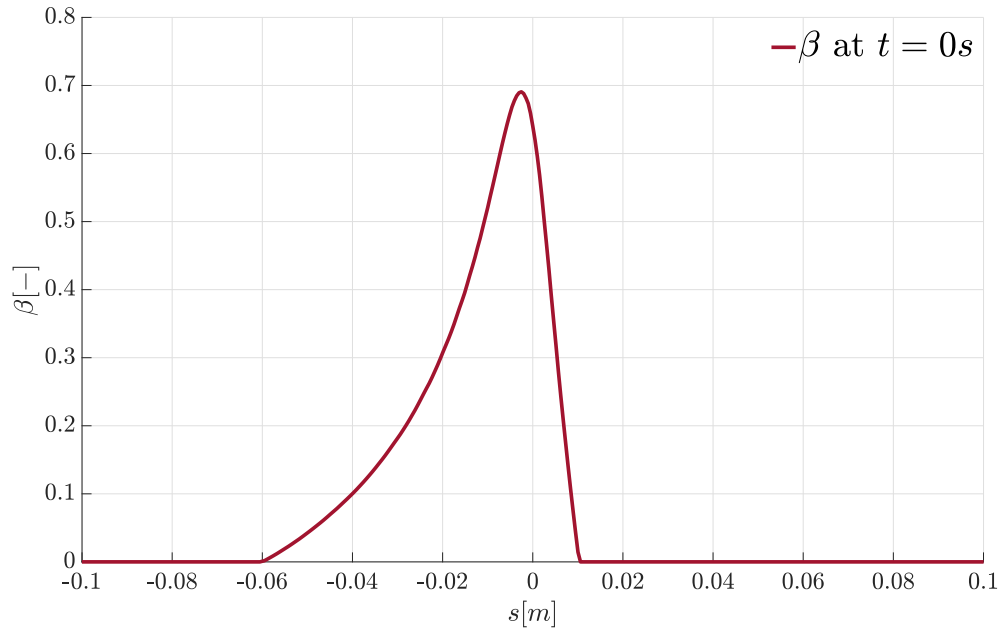
Figure 6.9 Collection efficiency on the clean profile for the glaze ice accretion test case. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

fact that the quantities evaluated at the previous time step cause the upper and lower horn to start forming at the beginning of the simulation. For this reason, once horns start forming separation occurs behind them and since no model for the detachment of the water film from the surface is considered, no water impinges in that region.

For the steady iterative model and the unsteady model, the iterations to evaluate the quantities are updated at every step and then the liquid film is able to move after the collection efficiency limits leading to ice formation closer to the experimental ice accretion limits.

As it can be seen from figure 6.10, the ice thickness close to the stagnation region is overestimated in all cases. A reduction can be observed in the case of the unsteady liquid film model. This is due to the fact that it is the only model that considers a non-constant liquid film height and its effect in the ice accretion rate computation. This means that in cases where there is more water that remains liquid on the surface, its influence on the ice accretion is not negligible anymore. A further improvement in the model could be obtained by considering the heat fluxes exchanged in the direction tangent to the surface due to the liquid film motion.

(a)

(b)

(c)
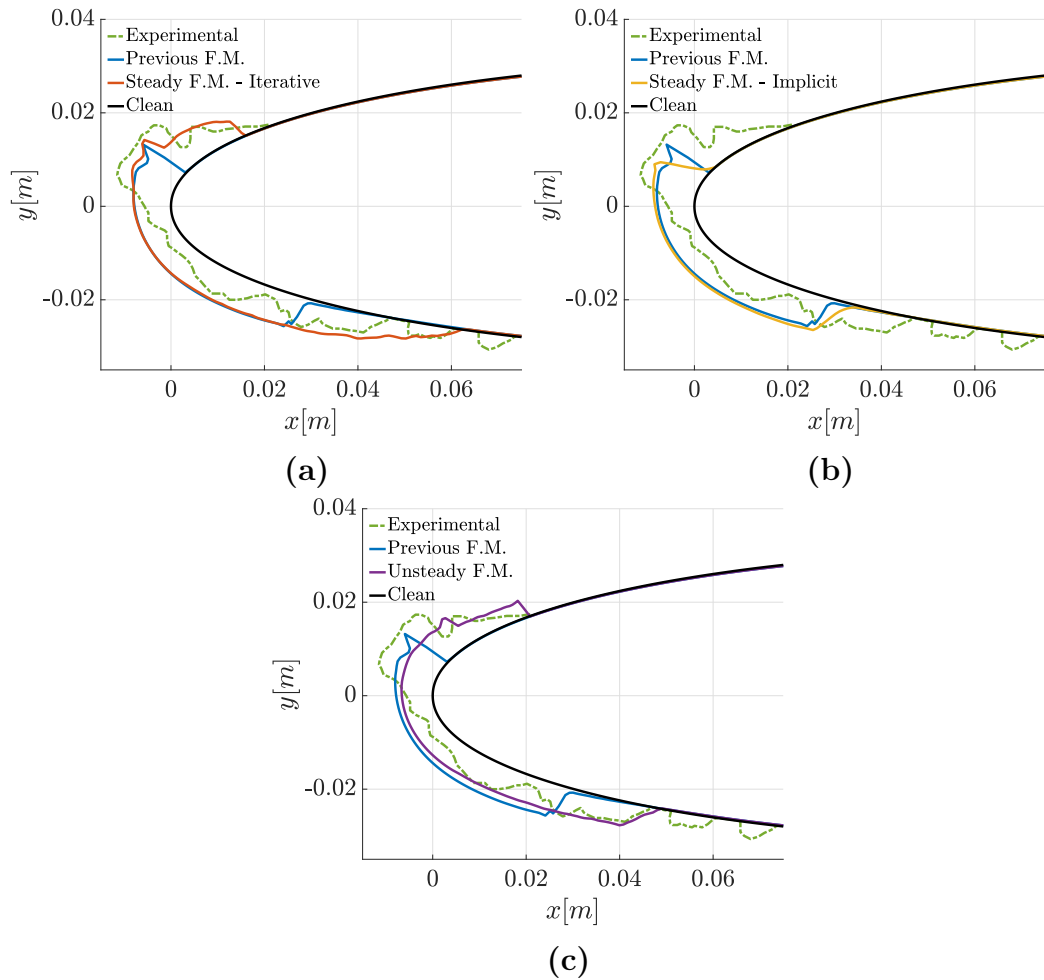
Figure 6.10 Glaze ice accretion. Comparison among experimental data [66] and PoliMIce with different liquid film models: (a) steady with the iterative implementation (section 3.1.1), (b) steady with the implicit implementation (section 3.1.2), (c) unsteady (section 3.2) and the film model previously implemented in PoliMIce [24]

Lastly, a consideration regarding the ability of the models to capture the shape of the upper horn must be done. In this case, both the previously implemented and the steady film models are more or less able to capture the formation of the upper horn. This does not happen when the unsteady liquid film model is used, where a small bump forms where the horn should form but it does not grow as in the other cases. The reason that causes this behavior can be ascribed to the numerical method used to discretize the model equation: a first order upwind scheme results to be quite diffusive close to discontinuities, which is what would cause the horns to form. Further studies should consider the possibility to use a more accurate numerical scheme for the unsteady liquid film model.

It must be considered that also in this case the ice shape resembles the shape of the airfoil due to the fact that the direction of ice accretion is the normal direction computed on the clean airfoil.

As an indication of the total volume of ice that formed on the airfoil, in table 6.6 is reported a comparison of the iced surfaces computed for the different ice shapes. As it can be seen, the software LEWICE and ONERA give an overestimation of the total surface of ice with an error with respect to the experimental surface of 21.75% and 29.50% respectively. For what concerns the results obtained with PoliMIce, as it could be expected the previously implemented film model and the steady one with the implicit formulation give a clear underestimation of the volume of ice and this is due to the horn forming early during the accretion process. On the other hand, the steady film model with the iterative formulation overestimates the total surface of the ice with an error of 27.58%. Once again the use of the unsteady film model gives the more accurate result. In figure 6.11, the results obtained with PoliMIce are compared with experimental data and numerical results obtained with LEWICE and ONERA. The model chosen for this comparison is the unsteady liquid film model because it gave the best results in terms of ice surface and the ice shape was quite well predicted. It can be seen that the ice accretion limits are predicted very well in PoliMIce while they are over predicted in the other cases. As it was stated before, the ice thickness close to the stagnation point is overestimated by PoliMIce with respect to the other codes.

Figure 6.12 shows the results of the liquid film models at three different

	Iced surface [m^2]	Err %
Experimental	5.2408×10^{-4}	-
LEWICE	6.3807×10^{-4}	21.75%
ONERA	6.7872×10^{-4}	29.50%
Previous F.M.	4.0621×10^{-4}	22.49%
Steady F.M. - Iterative	6.6865×10^{-4}	27.58%
Steady F.M. - Implicit	4.4299×10^{-4}	15.47%
Unsteady F.M.	5.1487×10^{-4}	1.76%

Table 6.6 Iced surface comparison for the glaze ice accretion test case
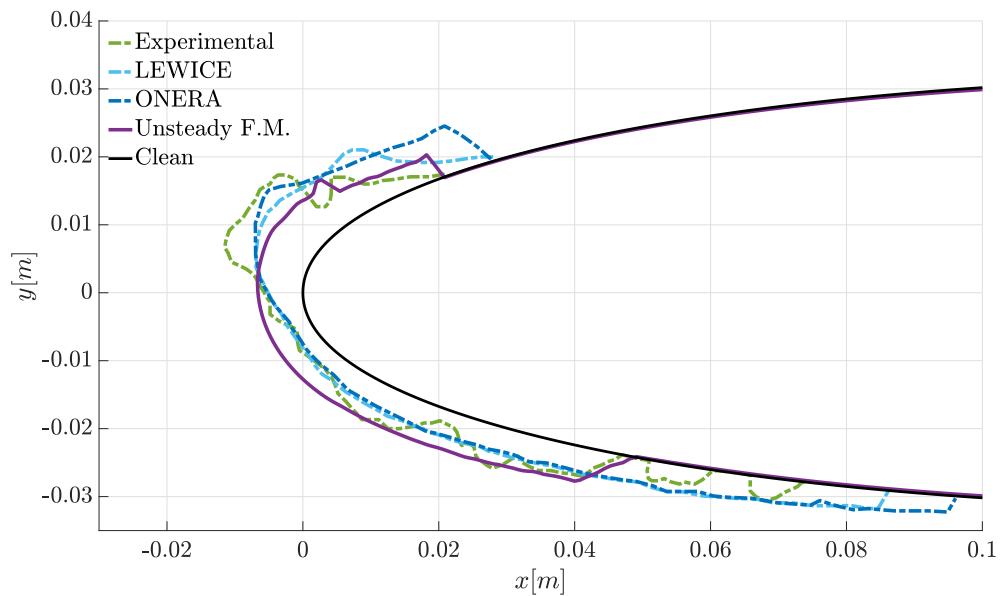
Figure 6.11 Glaze ice accretion. Comparison among experimental data and numerical results with LEWICE, ONERA [66] and PoliMIce with the unsteady liquid film model (section 3.2)

time steps: $t_1 = 160\,\mathrm{s}$, $t_2 = 320\,\mathrm{s}$ and $t_3 = 480\,\mathrm{s}$. In the left column the mass flow rate of water is compared between the three different models. At $t_1 = 160\,\mathrm{s}$ the steady-iterative and the unsteady film model give very similar results, while the previous film model and the implicit formulation give an underestimation in terms of water limit. This difference is due to the early formation of the horns explained earlier. As time evolves and more water remains liquid on the surface, the models become more and more different one from the other also because of the differences in ice shapes.

In the right column of the figure is reported the water film height. Only the unsteady film model is presented, since for the other three cases the liquid film height is fixed and set equal to $h = 0.0001\,\mathrm{m}$. As already observed in mixed ice conditions, the Myers hypothesis on the water film height results in an overestimation even for glaze ice conditions: the mean value for the three time step considered is $h \approx 1.5 \times 10^{-5}\,\mathrm{m}$.

As it can be seen from these plots, the liquid film height remain approximately constant in time and this explains why the steady model in the iterative formulation and the unsteady model gave similar results in this test case.

It can be seen from figure 6.12 and figure 6.9 that, using both the unsteady liquid film model and the steady film model in the iterative formulation, ice is able to form also past the impinging limit. This is typical of glaze ice accretion, where the water that impinges on the surface does not freeze right away but flows aft and freezes later.

6.1.4   Long exposure ice accretion

The last test case for ice accretion was conducted in the Icing Research Tunnel (IRT) at the NASA Lewis Research Center [67]. This is representative of a condition producing a large ice accretion. Moreover the test case concerns a different airfoil with respect to the NACA0012 used till now, in order to verify the capability of the software to simulate ice accretion on different geometries. Table 6.7 presents the flight and atmospheric conditions.

The total time exposure is $t = 1350\,\mathrm{s}$ while the CFD and particle tracking computation are run every 60 s, except the last step which lasts 30 s. For the unsteady film model simulation, the inner time step was set to $dt = 0.001\,\mathrm{s}$. The mesh used for this test case has 34050 elements with 422 elements on the airfoil boundary. The number of parcels reached by the particle tracking
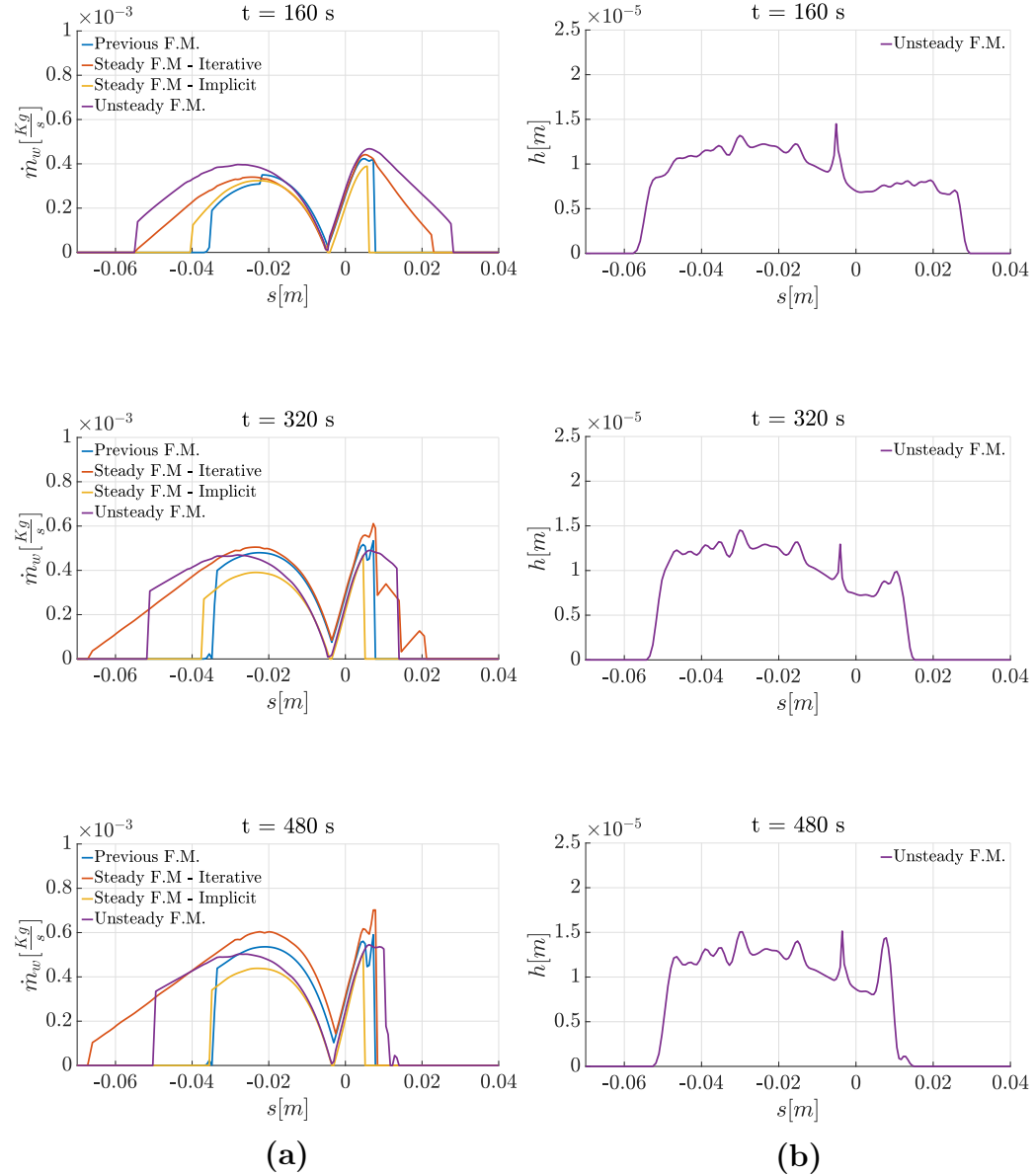
Figure 6.12 Glaze ice accretion. Comparison between: (a) water mass flux obtained with different liquid film models at $t_1 = 160\,\text{s}$, $t_2 = 320\,\text{s}$ and $t_3 = 480\,\text{s}$ and (b) liquid film height obtained with the unsteady liquid film model at $t_1 = 160\,\text{s}$, $t_2 = 320\,\text{s}$ and $t_3 = 480\,\text{s}$. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

Long exposure	
Airfoil	GLC305
Chord	0.9144 m
$\boldsymbol{\alpha}$	6°
$\mathbf{V}_\infty$	90 m/s
$\mathbf{T}_\infty$	263.15 K
$\mathbf{P}_\infty$	101 325 Pa
MVD	20 µm
LWC	0.43 g/m^3
t	1350 s
$\boldsymbol{\Delta}$**t**	60 s

Table 6.7 Parameters for long exposure ice accretion test case

with the cloud adaptation is around $n_p \approx 150000$. The collection efficiency on the clean airfoil is shown in figure 6.13.

Figure 6.14 presents different ice shapes obtained with PoliMIce using different liquid film models and their comparison with experimental data from reference [67]. As it can be clearly seen from this figure, the steady film and the previously implemented model give a very different ice shape compared to experimental data. The steady film model captures a similar shape that remains narrower with respect to the experimental one. On the other hand the previously implemented model predicts a shape which is completely different from the expected one. This can be explained by how the water mass fluxes are evaluated in the iterative and implicit implementation of the steady film model with respect to the previously implemented model. As a matter of fact, these are all based on the same mathematical model but for what concerns the previously implemented model the mass entering the control volume is the one computed at the previous time step. As shown in figure 6.15, at a certain time the value of the mass flux is clearly overestimated with respect to the new steady film model. The overestimated fluxes are then used in the next time step leading to an increasing of the error as time proceeds. Lastly, the ice shape predicted using the unsteady film model shows good agreement with experimental results. It can be seen that, while in the other cases the ice thickness at the stagnation point is overestimated, in this case
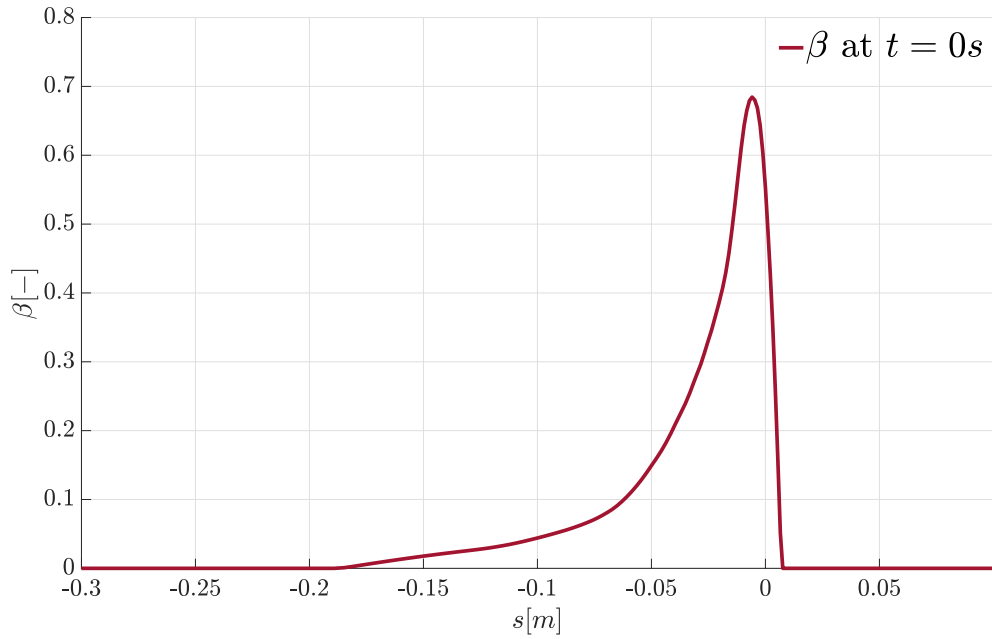
Figure 6.13 Collection efficiency on the clean profile for the long exposure ice accretion test case. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

considering the heat conduction in the water layer leads to a better prediction of the ice thickness.

Figure 6.16 presents the ice shape comparison among liquid film models at three time steps $t_1 = 480\,\mathrm{s}$, $t_2 = 780\,\mathrm{s}$ and $t_3 = 1080\,\mathrm{s}$. The four ice shapes at $t_1 = 480\,\mathrm{s}$ show slight differences, as it happened in previous test cases. This was expected since the exposure time is comparable to the other cases. As time progress, the ice shapes become increasingly different due to the change of the aerodynamics which becomes very different as time evolves causing the ice shape to differ even more. This is clearly seen in the case of the previously implemented model, where the error in the mass flux computation propagates with time causing the ice shape and aerodynamics to be considerably different. Table 6.8 presents the iced surfaces computed for different ice shapes. In this case both LEWICE and PoliMIce clearly underestimate the experimental data. As it can be seen, the iced surface predicted by both the steady and unsteady film model is quite similar, although the shape is better predicted using the unsteady model. In all cases the error with respect to the experimental data is quite high: 37.17% and 35.92% for the steady model in the iterative

(a)

(b)

(c)
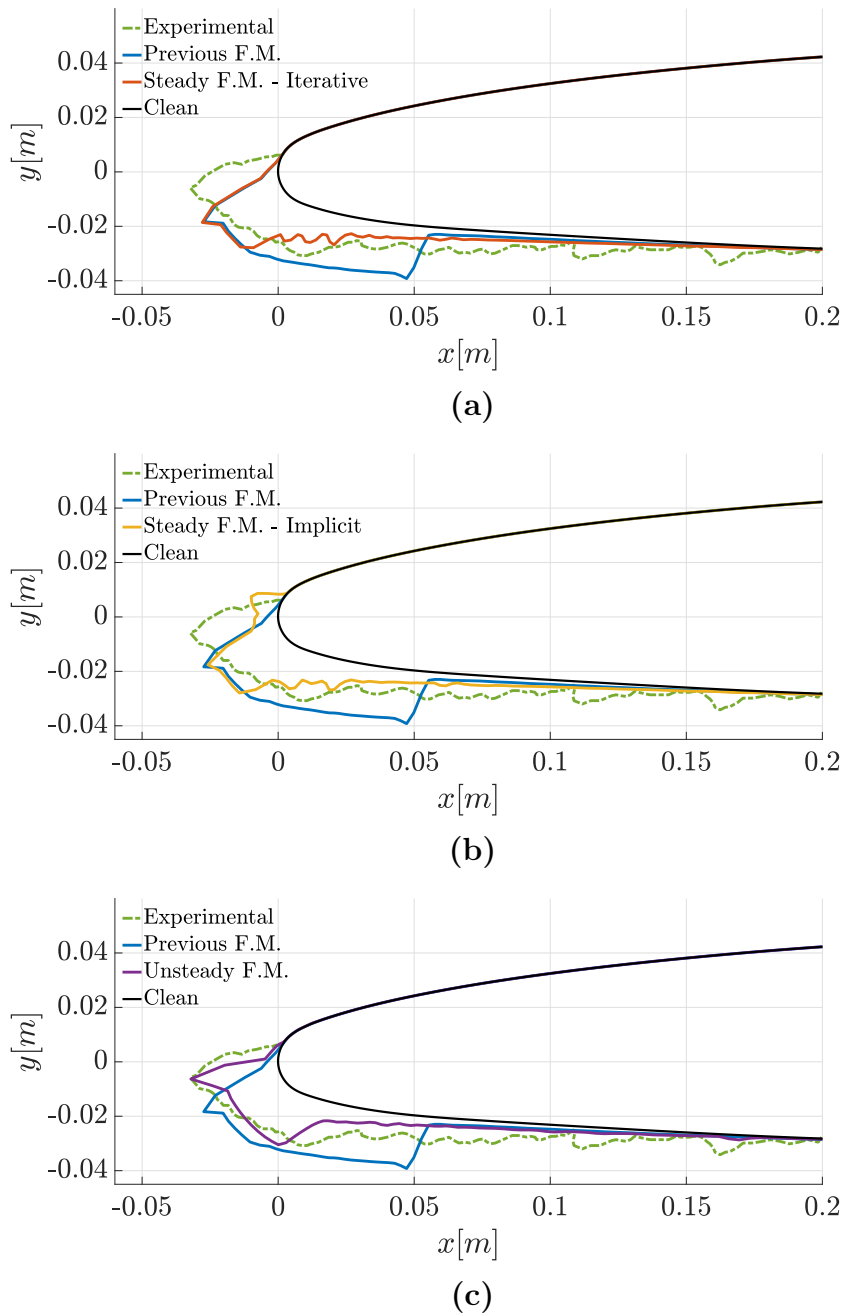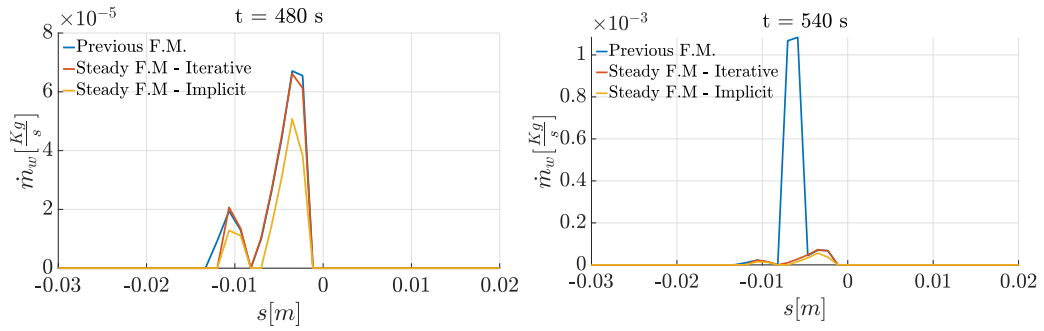
Figure 6.14 Long exposure ice accretion. Comparison among experimental data [67] and PoliMIce with different liquid film models: (a) steady with the iterative implementation (section 3.1.1), (b) steady with the implicit implementation (section 3.1.2), (c) unsteady (section 3.2) and the film model previously implemented in PoliMIce [24]

117

Figure 6.15 Water mass flux at two time step $t_1 = 480\,\text{s}$ and $t_2 = 540\,\text{s}$ obtained with steady liquid film models: iterative implementation (section 3.1.1), implicit implementation (section 3.1.2) and the film model previously implemented in PoliMIce [24]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

	Iced surface [m^2]	Err %
Experimental	1.9992×10^{-2}	-
LEWICE	1.2277×10^{-2}	38.59%
Previous F.M.	1.7792×10^{-2}	11.00%
Steady F.M. - Iterative	1.2560×10^{-2}	37.17%
Steady F.M. - Implicit	1.2810×10^{-2}	35.92%
Unsteady F.M.	1.2227×10^{-2}	38.84%

Table 6.8 Iced surface comparison for the long exposure ice accretion test case

and implicit formulation respectevely and 38.84% for the unsteady film model. It can be observed that the iced surface predicted using the previously implemented model is the closest to experimental data with an error of 11.00%. This would have been a good result although it is clear that this is not the case since the shape obtained is caused by the propagation of an error of the water mass flux computation.

Figure 6.17 shows the comparison of PoliMIce results with experimental data and numerical results from LEWICE [67]. Once again, the ice shape obtained using the unsteady film model has been chosen for this comparison because it gave the closest results to experimental data. The shape computed with
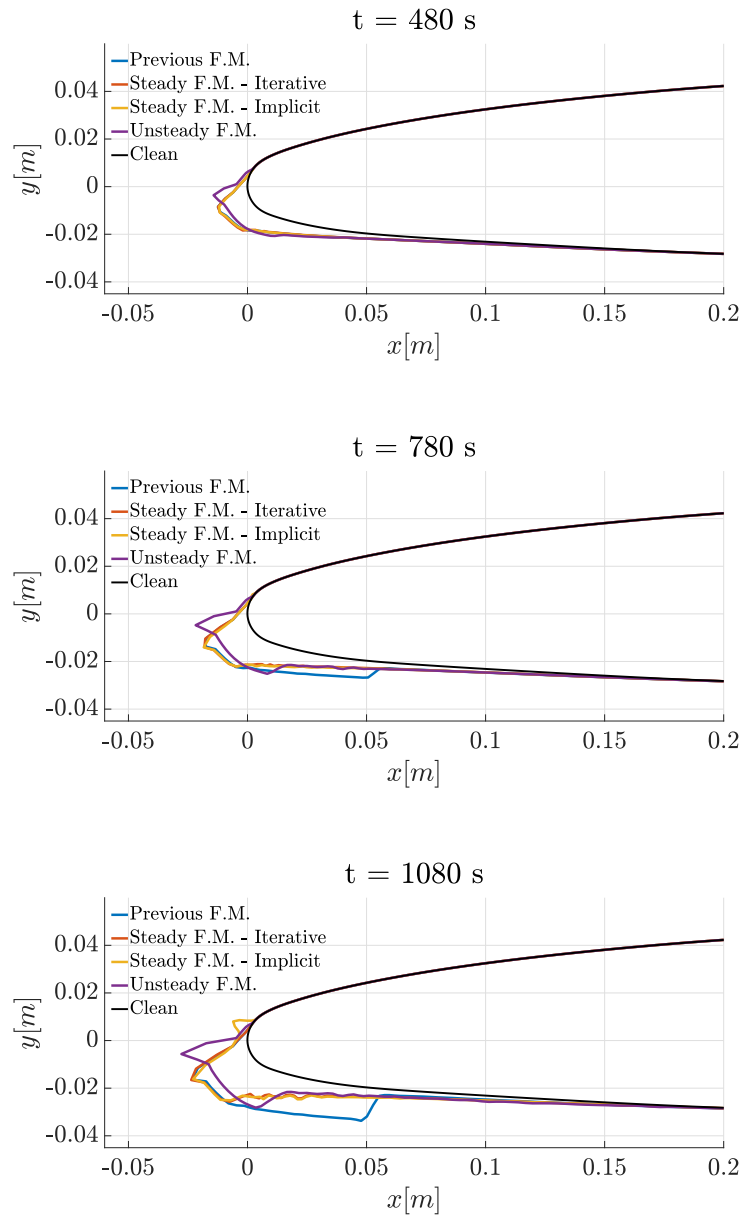
Figure 6.16 Long exposure ice accretion. Comparison among PoliMIce results at three time steps $t_1 = 480\,\text{s}$, $t_2 = 780\,\text{s}$ and $t_3 = 1080\,\text{s}$ with different liquid film models: steady with the iterative implementation (section 3.1.1), steady with the implicit implementation (section 3.1.2), unsteady (section 3.2) and the film model previously implemented in PoliMIce [24]
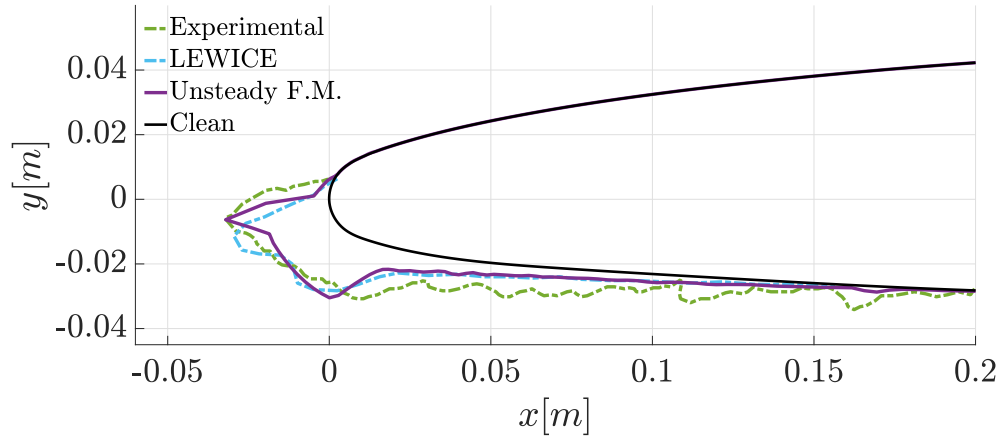
Figure 6.17 Long exposure ice accretion. Comparison between experimental data and numerical results with LEWICE [67] and PoliMIce with unsteady liquid film model (section 3.2)

PoliMIce shows good agreement with both the experimental and LEWICE computation. The ice thickness is underestimated by both software in the lower part of the airfoil.

Figure 6.18 reports the results regarding the liquid film obtained with different models at three time steps: $t_1 = 480\,\mathrm{s}$, $t_2 = 780\,\mathrm{s}$ and $t_3 = 1080\,\mathrm{s}$. In the left column is shown a comparison of the mass flow rate between the steady and unsteady model. In this case the previously implemented film model has been excluded from the comparison because the value of the mass fluxes were about an order of magnitude larger then the other models due to the error previously explained. As it can be seen in figure 6.18, the steady film model underestimate the value of the mass fluxes and this causes the ice shape to remain narrower. At time step $t_1 = 480\,\mathrm{s}$ the unsteady film model is able to capture the film motion even after the impinging limits. As time evolves the small horns start to form and the liquid film limits are reduced. In the right column of the figure is reported the liquid film height evaluated with the unsteady film model. Once again, Myers hypothesis of a constant liquid film thickness of $h = 0.0001\,\mathrm{m}$ is an overestimation. The average film thickness at the last time step is $h \approx 1.5 \times 10^{-5}\,\mathrm{m}$.

(a)

(b)
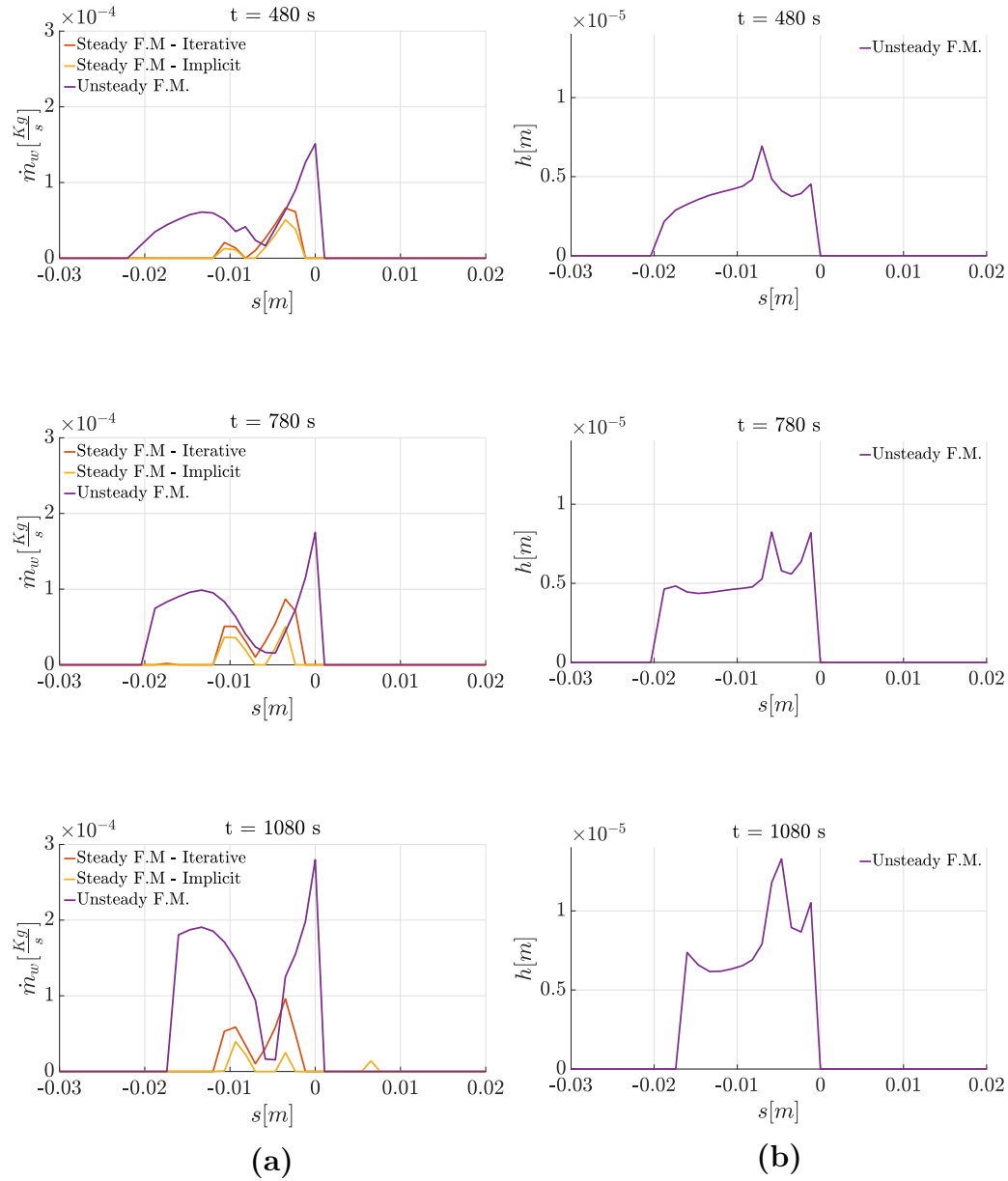
Figure 6.18 Long exposure ice accretion. Comparison between: (a) water mass flux obtained with different liquid film model at $t_1 = 480$ s, $t_2 = 780$ s and $t_3 = 1080$ s and (b) liquid film height obtained with the unsteady liquid film model at $t_1 = 480$ s, $t_2 = 780$ s and $t_3 = 1080$ s. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side
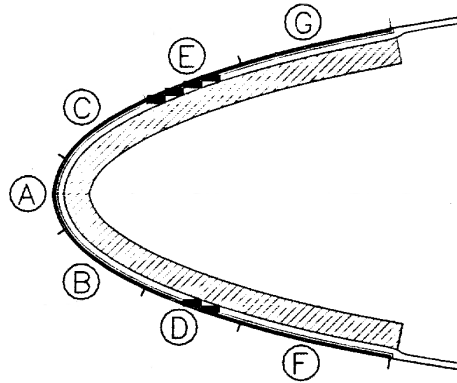
Figure 6.19 Airfoil leading edge and heated zones [2]

6.2 Electro-thermal IPS test cases

The electro-thermal Ice Protection System model implemented in chapter 4 has been tested in three different conditions. Many experiments have been conducted at NASA Glenn Research Center by Al-Khalil to validate the code ANTICE in different environmental conditions [2]. As shown in figure 6.19, for this experimental campaign on the leading edge of the airfoil were set seven heaters controlled separately to provide different heat fluxes, depending on the run. As it was described in chapter 4, each heating element was composed by different layers. The material composition is show in figure 4.2 while the material characteristics are presented in table 4.1. Table 6.9 reports the streamwise distance from the leading edge of the airfoil of each heating element, normalized by the chord. Negative values refer to the pressure side of the airfoil while positive values refer to the suction side.

The experimental data obtained by Al-Khalil are an important database of results for the validation of an IPS numerical simulation. Therefore these test cases have been used also to validate the models developed by Silva et al. and Bu et al. [54] [13]. Three test cases were selected to validate PoliMIce and the results were compared with both experimental data and numerical results obtained by means of ANTICE and by means of the codes of Silva and Bu. For each test case will be analyzed the runback water mass flux $\dot{m}_w$, the surface temperature on the airfoil T_{wall}, the convective heat transfer coefficient h_{air} and the overall heat transfer coefficient h_{eff} which is introduced to consider

Heater	Start ($\frac{s}{c}$)	End ($\frac{s}{c}$)
F	-0.1024	-0.0607
D	-0.0607	-0.0329
B	-0.0329	-0.0051
A	-0.0051	0.0157
C	0.0157	0.0435
E	0.0435	0.0713
G	0.0713	0.1129

Table 6.9 Heater locations on the airfoil

the total heat lost on the surface:

$$h_{eff} = \frac{\dot{Q}_{lost}}{1 \, \Delta S \, \Delta T} \tag{6.1}$$

where ΔS is the element size and $\dot{q}_{lost}$ the total heat by the surface, defined as follows:

$$
\begin{aligned}
\dot{Q}_{lost} = \left(\dot{q}^{''}_{conv} - \dot{q}^{''}_{imp} + \dot{q}^{''}_{e} + \dot{q}^{''}_{ice} \right) 1 \, \Delta S \\
- \dot{m}_{in} c_{p_{H_2O}} \left(T_{in} - T_{ref} \right) + \dot{m}_{out} c_{p_{H_2O}} \left(T_{out} - T_{ref} \right)
\end{aligned}
\tag{6.2}
$$

where the heat flux contributions are the same explained in chapter 4 here reported for clarity:

$$
\begin{aligned}
\dot{q}^{''}_{conv} &= (1 - F) \; h_{air} \; (T_{rec} - T_{wall}) + F \; h_{H_2O} \; (T_{wall} - T_{H_2O}) \\
\dot{q}^{''}_{imp} &= \dot{m}^{''}_{imp} \left[c_{p_{H_2O}} \; (T_{imp} - T_{ref}) + \frac{V^2_{imp}}{2} \right] \\
\dot{q}^{''}_{e} &= \dot{m}^{''}_{e} \left[i_{l-v} + c_{p_{H_2O}} \; (T_{H_2O} - T_{ref}) \right] \\
\dot{q}^{''}_{ice} &= \dot{m}^{''}_{ice} \left[i_{l-s} + c_{p_{H_2O}} \; (T_{H_2O} - T_{ref}) \right]
\end{aligned}
\tag{6.3}
$$

The software used in the PoliMIce framework for testing the IPS model are the same that have been used for ice accretion, here reported for clarity:

- **Mesh generation**: uhMesh [22]

- **Aerodynamic solver**: SU2 6.1.0 [23]

- **Particle tracking**: PoliDrop [7]
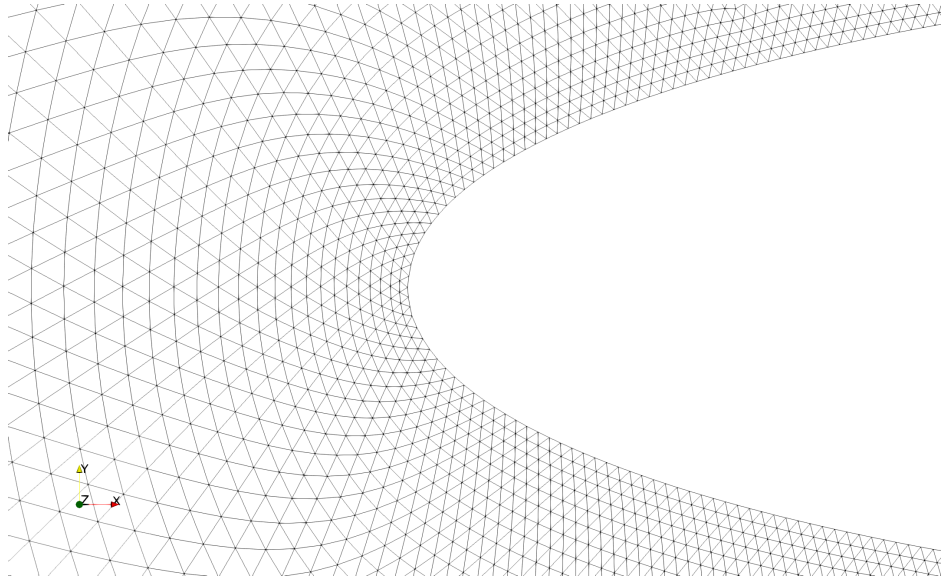
Figure 6.20   Computational grid for the IPS test cases on the airfoil leading edge

As all test cases refer to anti-ice conditions, no ice accretion occurs and therefore a mesh update is not needed. By means of uhMesh, an unstructured mesh was generated and used for all test cases. The structured boundary layer mesh is not necessary for IPS simulations since an Euler CFD simulation is performed. The airfoil boundary is divided into multiple zones because, for the anti-ice simulation, different heat fluxes of the heaters must be imposed on different boundaries. The far-field is again a circular surface of radius $R = 8\,\text{m}$. The mesh used for the IPS test cases is shown in figure 6.20, with a detail on the airfoil leading edge.

The aerodynamic computation was performed using the software SU2 to solve the Euler equations. The convective fluxes were solved with the Roe numerical method [47] with a second order upwind scheme with no slope limiter. The convergence criteria was set to a residual reduction of 10 orders of magnitude.

A slip boundary condition was set on all airfoil boundaries. At the far-field, temperature, pressure and Mach number were imposed. The specific value of these variables depend on the the test case and they will be described in the corresponding section.

The collection efficiency was evaluated using PoliDrop [8]. The Forward Euler time integration scheme was used for all test cases with a time step of

Test Case 22A	
Airfoil	NACA0012
Chord	$0.914\,\mathrm{m}$
$\boldsymbol{\alpha}$	$0°$
$\mathbf{V}_\infty$	$44.7\,\mathrm{m/s}$
$\mathbf{T}_\infty$	$265.45\,\mathrm{K}$
$\mathbf{P}_\infty$	$90\,000\,\mathrm{Pa}$
MVD	$20\,\mathrm{\mu m}$
LWC	$0.78\,\mathrm{g/m^3}$

Test Case 22A	
Heater	$\dot{q}''_{IPS}$ $(\mathrm{kW/m^2})$
F	9.92
D	10.23
B	32.55
A	46.50
C	18.60
E	6.98
G	10.23

**Table 6.10** Simulation parameters and heat fluxes provided by heaters for test case 22A

$dt = 10^{-5}\,\mathrm{s}$. In this case the cloud adaptation strategy was not used and the number of particles in the cloud was set as fixed to $n_p = 100000$.

6.2.1 Test case 22A

The first test case refers to run 22A from reference [2]. It represents an evaporative anti-ice condition considering a relatively mild ambient temperature. The icing conditions are presented in table 6.10 along with the heat fluxes set on the heaters. The turbulence level Tu, necessary for the calculation of the boundary layer transition, has been set to 3.1%, as assumed by Bu et al. [13].

The heat fluxes set during the experimental campaign have been chosen considering the amount of water that impinges on the airfoil. Due to the geometry of the test case, an higher value of the heat flux is needed close to the stagnation point since it is the location where the largest amount of water impinges. The heat fluxes are then decreased as moving further away from the leading edge.

Figure 6.21 shows the pressure coefficient and the collection efficiency obtained with SU2 and PoliDrop respectively. The pressure coefficient is compared with the one obtained with ONERA 2D [54], showing good agreement. Moreover, the collection efficiency evaluated by PoliDrop is compared with results of ANTICE [2], ONERA 2D [54] and FLUENT [13]. Once again, a good

(a)

(b)
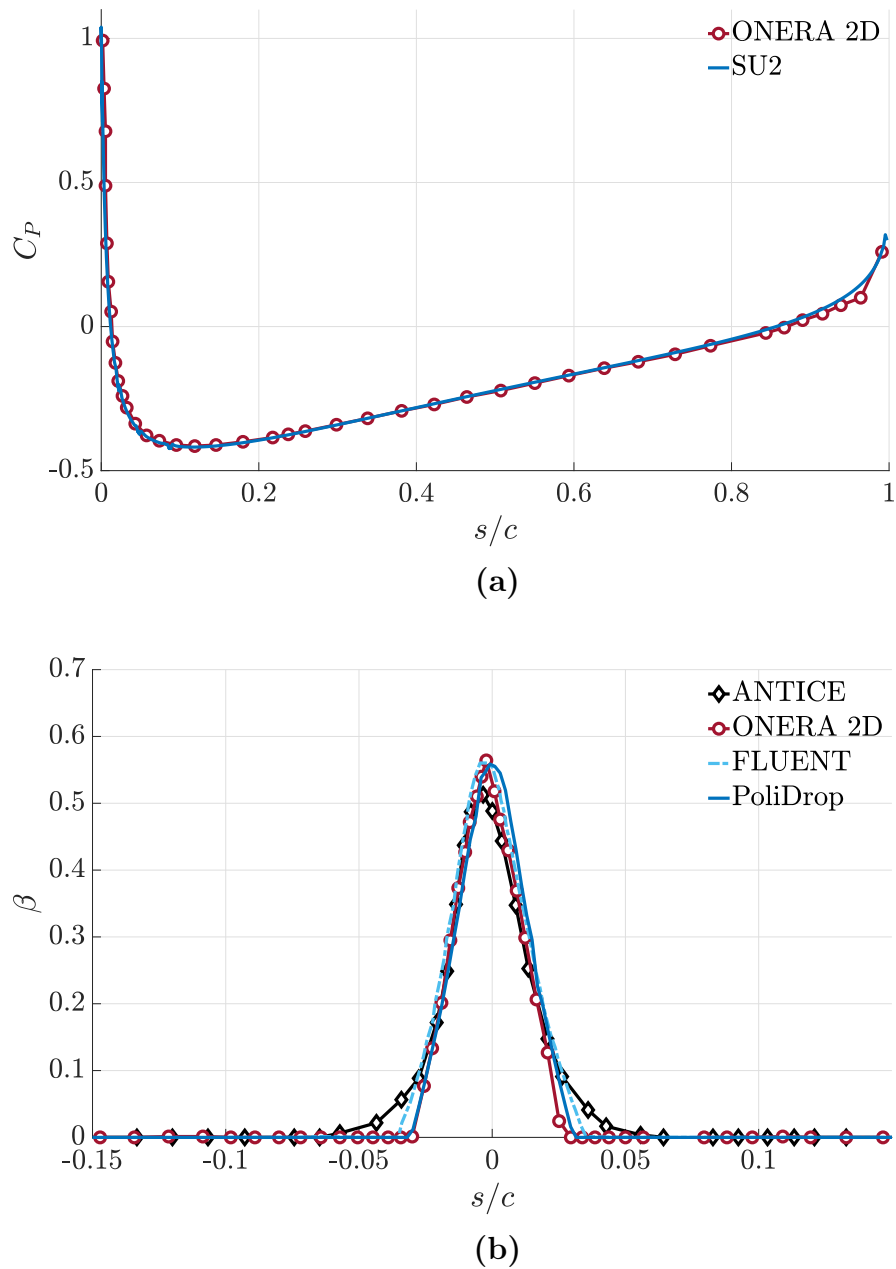
Figure 6.21 Test case 22A. (a) Pressure coefficient obtained with SU2 and compared with numerical results obtained with ONERA 2D [54]. (b) Collection efficiency evaluated with PoliDrop [7] and compared with numerical results obtained with the software ANTICE [2], ONERA 2D [54] and FLUENT [13]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

agreement with reference results is obtained. A slight difference in the position of the peak of the collection efficiency is noticed and it could be due to the not inclusion of gravity in the particle tracking computation.
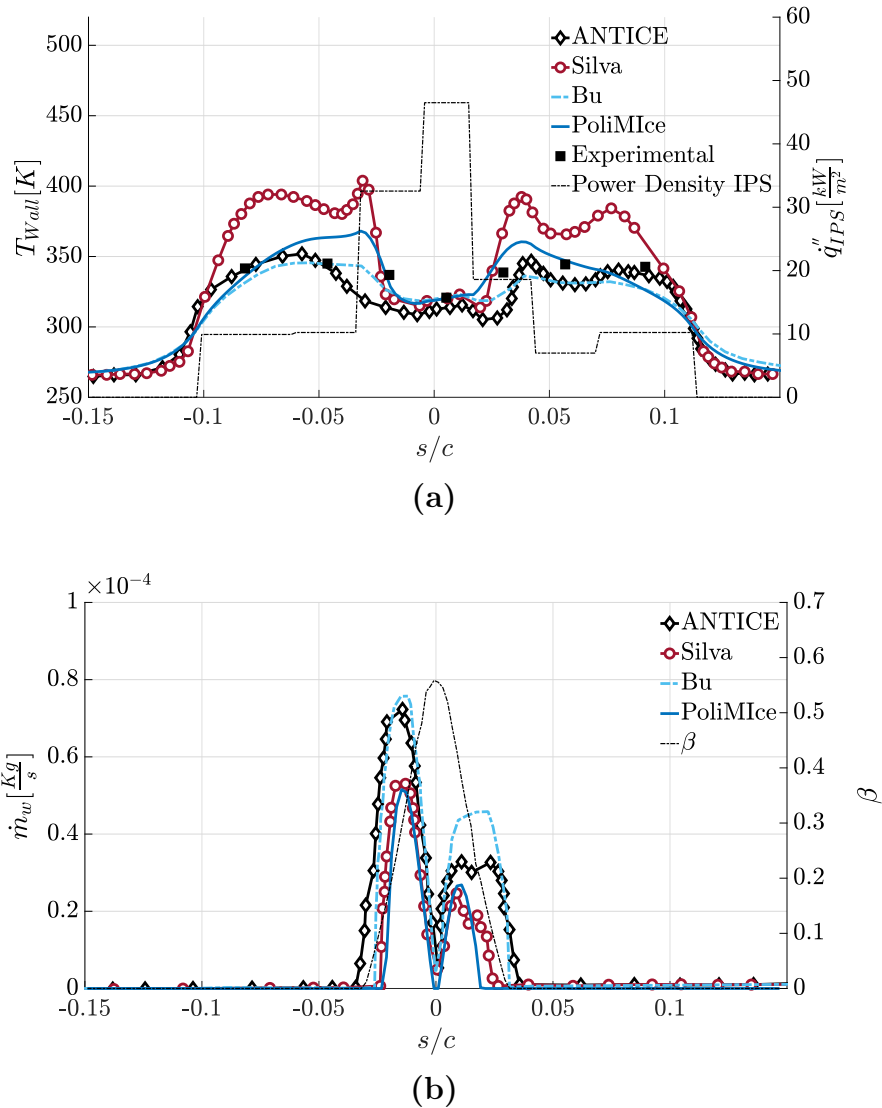
In figure 6.22, the surface temperature computed through PoliMIce is shown along with experimental data from experiments by Al-Khalil [2] and with numerical results from Silva et al., Bu et al. and ANTICE [54] [13] [2]. Figure 6.22 shows also the results concerning runback water. The results obtained with PoliMIce show good agreement with reference results.

The surface temperature is lower close to the stagnation point and increases rapidly after the runback water limits. This behaviour is due to the fact that in the region close to the leading edge, most of the heat is used to evaporate water. After the runback water limits, temperature increases because the surface is fully dry and the heat released by the heaters causes the temperature to increase. The runback water limits are very close to the collection efficiency ones and this implies that the ice protection system is working in fully-evaporative conditon: the water film flowing on the airfoil evaporates before reaching the end of the protected region.

Figure 6.23 shows the prediction of the convective heat transfer coefficient obtained with PoliMIce and it is compared with numerical results obtained by Silva et al. and Bu et al. [54] [13]. Figure 6.23 also presents the overall heat transfer coefficient resulted using PoliMIce, compared with experimental results from Al-Khalil [2]. The convective heat transfer coefficient shows good agreement with reference results: the peak position and value is very well captured. The main difference is noticed in the region right after the runback water limits this is due to the difference on the evaluation of the aforementioned limits. The trend of the heat transfer coefficient follows very well the behaviour predicted by Bu, especially in the transition and turbulent regime.

From these results, it can be confirmed that this test case corresponds to a fully-evaporative anti-ice condition. This is a very safe condition as the runback water does not reach the end of the protected region. On the other hand, it is very energy consuming since the heating power required to evaporate all the water is quite high. A reduction in the power given to heaters could lead to more runback water, still without reaching the protected region limits.

(a)

(b)

Figure 6.22 Test Case 22A. (a) Surface temperature and (b) runback water: comparison among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

(a)

(b)
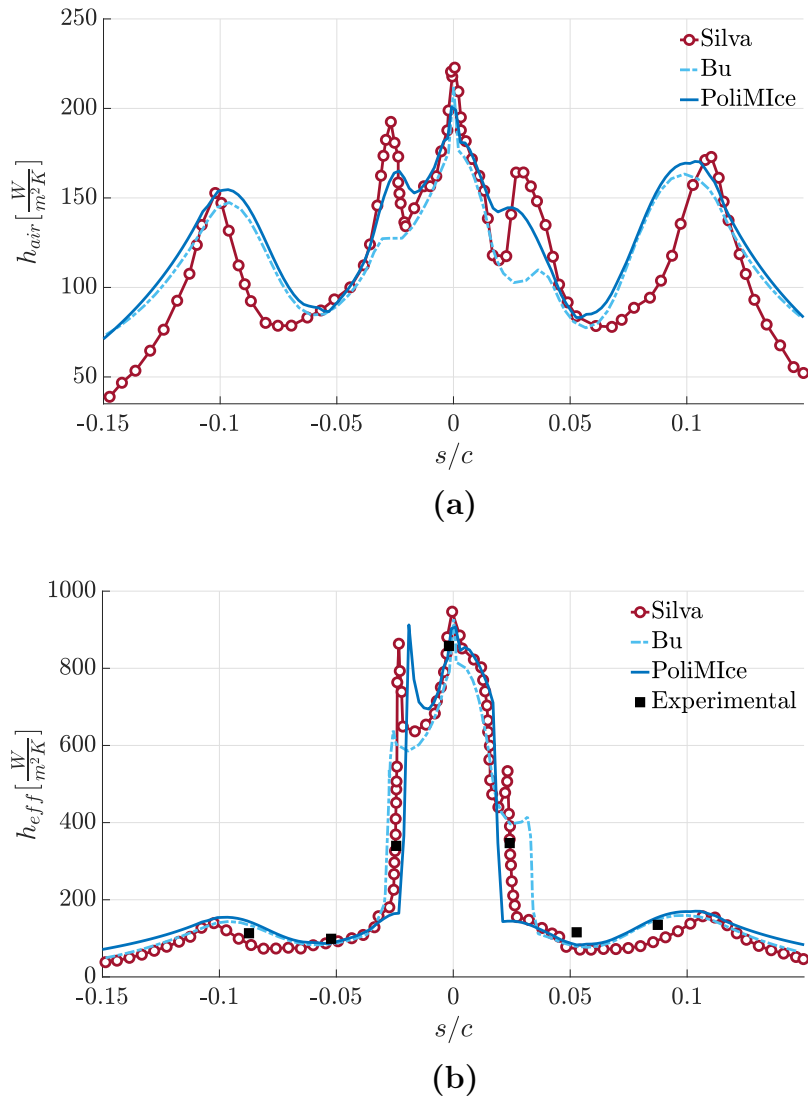
Figure 6.23 Test case 22 A. (a) Convective heat transfer coefficient and (b) overall heat transfer coefficient: comparision among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

Test Case 67A	
Airfoil	NACA0012
Chord	0.914 m
$\boldsymbol{\alpha}$	$0°$
$\mathbf{V}_\infty$	89.4 m/s
$\mathbf{T}_\infty$	251.35 K
$\mathbf{P}_\infty$	90 000 Pa
MVD	20 μm
LWC	0.55 g/m^3

Test Case 67A	
Heater	$\dot{q}''_{IPS}$ (kW/m^2)
F	20.15
D	21.70
B	32.55
A	43.40
C	26.35
E	18.60
G	18.60

Table 6.11   Simulation parameters and heat fluxes provided by heaters for test case 67A

6.2.2   Test case 67A

The second test case refers to run 67A from reference [2]. Also in this case, the parameters of the simulation represent an anti-ice system in evaporative condition, at lower temperature and higher free stream velocity with respect to case 22A. The turbulence level Tu has been set to 1.9%, as indicated by Bu et al. [13]. In table 6.11 are reported the environmental conditions and the heat fluxes provided by the heaters.

In figure 6.24 is presented the comparison of the pressure coefficient obtained by means of SU2 and the numerical result obtained with ONERA 2D [54]. The same CFD and particle tracking are valid for both the test cases 67A and 67B as they have the same environmental conditions. The only difference is in the heat fluxes provided by the IPS, which represent fully evaporative and running wet regimes. The two curves are perfectly superimposed, therefore the CFD calculations obtained for this test case show good agreement with the reference results.

The same figure reports also the collection efficiency obtained with PoliDrop [8] and compared with results obtained with the software ONERA 2D [54], FLUENT [13] and ANTICE [2]. The peak value and position is very well predicted compared to other results. The upper impingement limit shows very good agreement with the results obtained with ONERA 2D while the lower impingement limit is slightly underestimated. This could be caused by

(a)

(b)
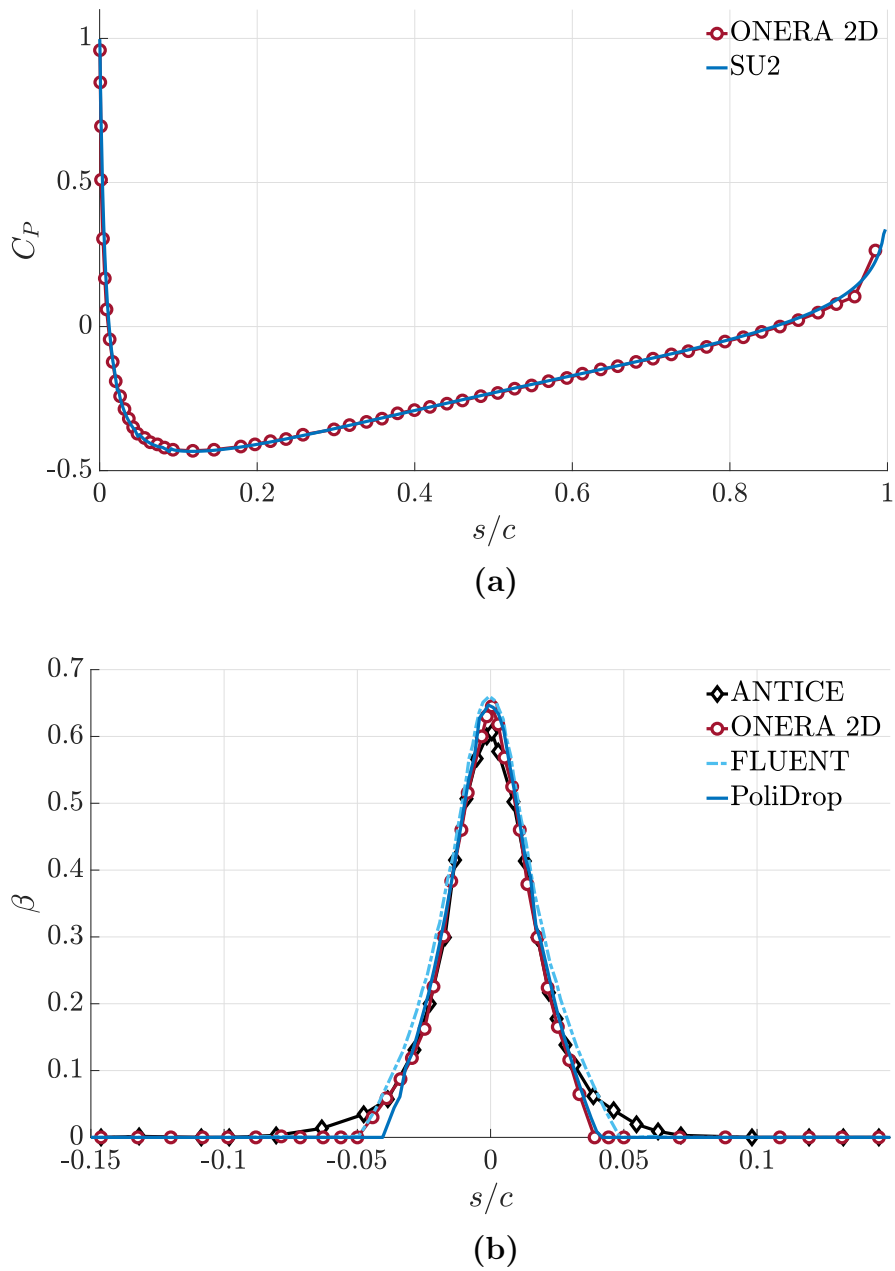
Figure 6.24 Test cases 67A and 67B. (a) Pressure coefficient obtained with SU2 and compared with numerical results obtained with ONERA 2D [54]. (b) Collection efficiency obtained with PoliDrop [7] and compared with numerical results obtained with the software ANTICE [2], ONERA 2D [54] and FLUENT [13]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side.

the non-inclusion of gravity effect in the simulation of the droplet trajectory with PoliDrop.

In figure 6.25 the surface temperature and the runback water computed by means of PoliMIce are compared with both experimental data from Al-Khalil [2] and numerical results from the works of Silva et al. [54] and Bu et al. [13]. For both these quantities, results obtained with PoliMIce show good agreement with reference results, particularly with numerical results by Bu et al. [13]. In figure 6.25 it can be seen that, as it could be expected, the surface temperature is higher where the heaters are placed while there is a drop below the freezing temperature in non-protected regions. This could lead to runback icing if the water flows past the protection limit, which does not happen in this test case. Another important aspect is that, as it happened for test case 22A, the surface temperature at the stagnation point is lower than on neighbour points, even though the heat flux is larger. This is due to the fact that, close to the stagnation point, there is also the peak in the collection efficiency. Therefore, more water is present on the surface and most of the heat coming from the heater is used to evaporate it. Contrarily, at the limits of the protected region, the temperature increases because only runback water is present, the amount of water that remains on the surface is lower or there is no liquid film. In this region the surface temperature is under estimated and it could be due to the fact that the small mass of water could lead to the formation of rivulets, which are not modeled in this work. Indeed, rivulets would yield a wetness fraction lower then one leading to an increase in the surface temperature.

The runback water limits are larger than the impingement limits because a large amount of water does not evaporate and flows aft. In this test case the limits remain inside the protected region, thus the heat fluxes are sufficient to evaporate the right amount of water without the risk of runback ice formation. Figure 6.26 shows the results of the convective heat transfer coefficient and the overall heat transfer coefficient obtained with Polimice. Both are compared with numerical results and experimental data. For what concerns the convective heat transfer coefficient, the results of PoliMIce are in good agreement with reference results. As it was expected, the peak is at the stagnation point and then decreases in the laminar region. The heat transfer coefficient then increases during transition and decreases again where the boundary layer

(a)

(b)
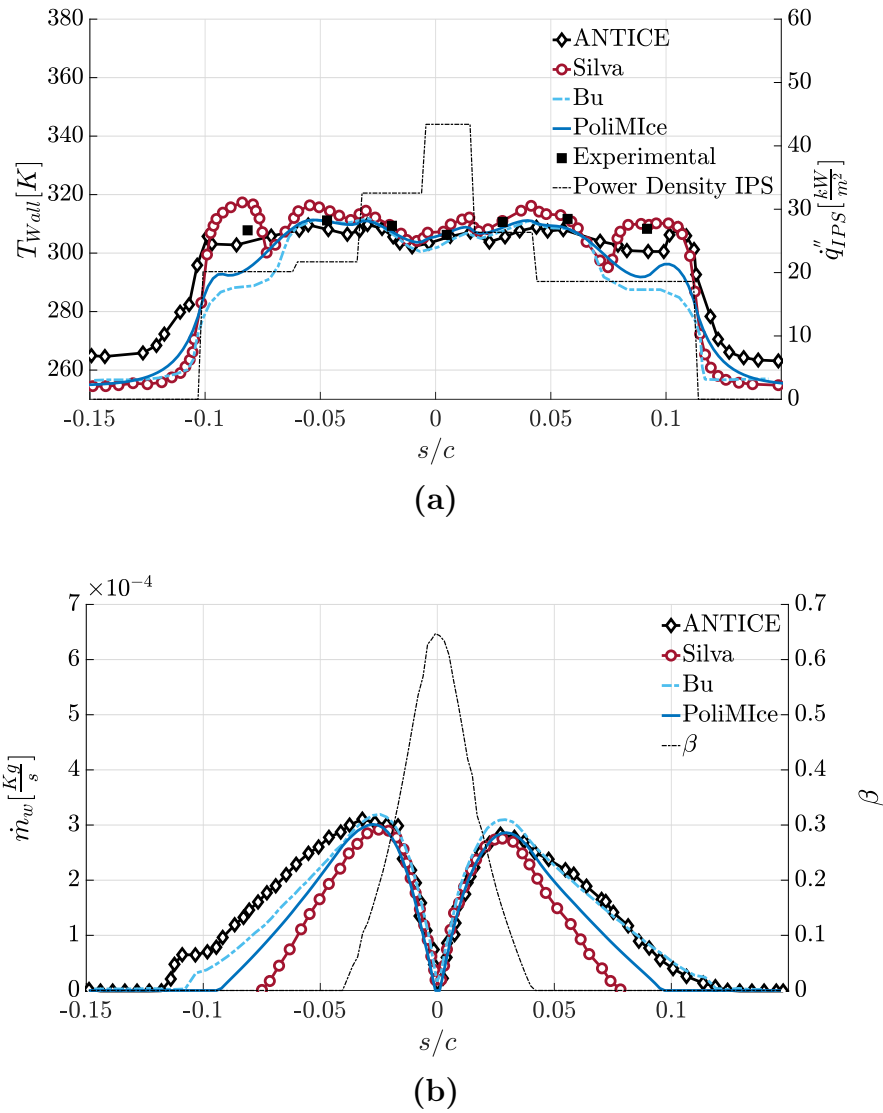
Figure 6.25 Test case 67A. (a) Surface temperature and (b) runback water: comparison among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side
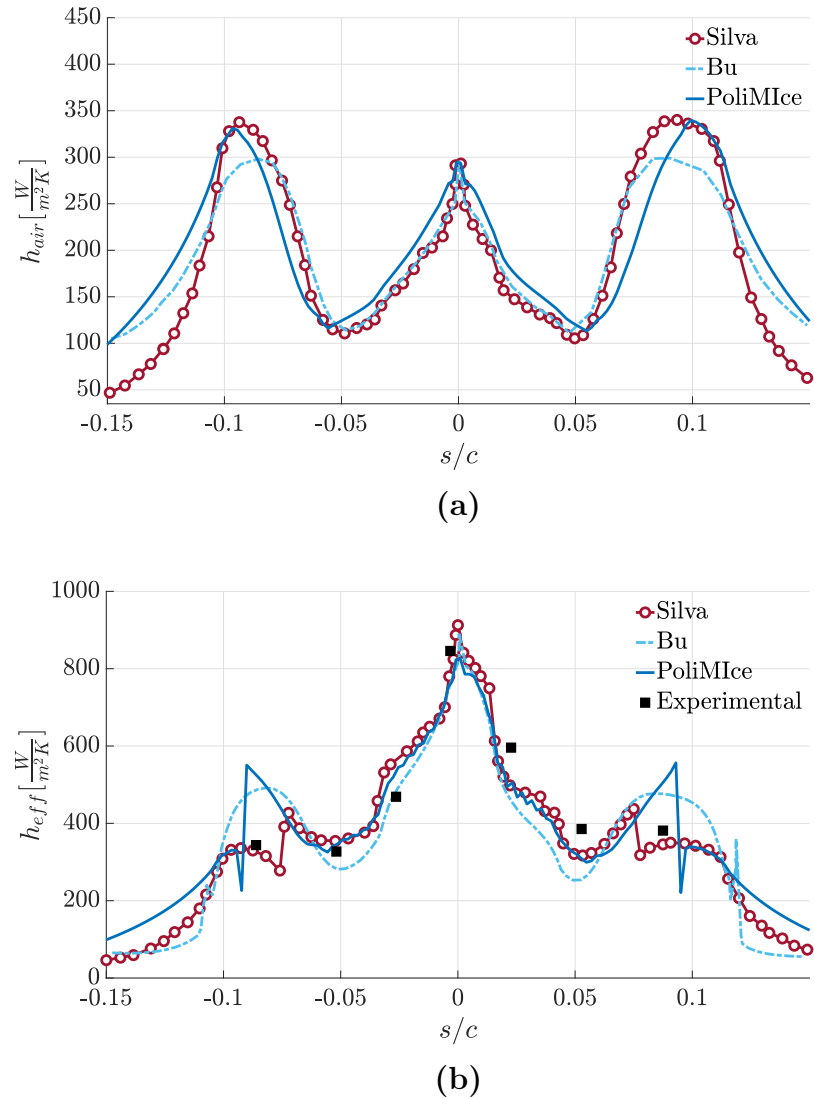
Figure 6.26 Test case 67A. (a) Convective heat transfer coefficient and (b) overall heat transfer coefficient: comparison among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side.

Test Case 67B	
Airfoil	NACA0012
Chord	0.914 m
$\boldsymbol{\alpha}$	0°
$\mathbf{V}_\infty$	89.4 m/s
$\mathbf{T}_\infty$	251.35 K
$\mathbf{P}_\infty$	90 000 Pa
MVD	20 μm
LWC	0.55 g/m³

Test Case 67B	
Heater	$\dot{q}''_{IPS}$ (kW/m²)
F	8.37
D	11.94
B	10.85
A	15.19
C	9.92
E	12.87
G	8.68

Table 6.12 Simulation parameters and heat fluxes provided by heaters for test case 67B

becomes turbulent.

The effective heat transfer coefficient shows good agreement with reference results in the protected region. It can be noticed a discontinuity corresponding to the runback water film end. The jump is caused by the way the overall heat transfer coefficient is computed. In PoliMIce, where no water is present, the overall heat transfer coefficient is just set equal to h_{air} causing the discontinuity in the value. Moreover, in the region where water is present, a slight oscillation can be noticed and it is due to the calculation of the water temperature as the average of the inflow and outflow temperature in an element.

6.2.3   Test case 67B

The last test case refers to run 67B from reference [2]. The environmental conditions are the same as case 67A, reported in table 6.12 for clarity. On the other hand, the heat fluxes of the heaters have lower values respect to case 67A. The turbulence level Tu has been set to 3.0%, as assumed by Bu et al. [13].

The surface temperature and the runback water predicted using PoliMIce are presented in figure 6.27.

The surface temperature is in good agreement with reference results. As it was expected, the temperature at the stagnation point is lower than at
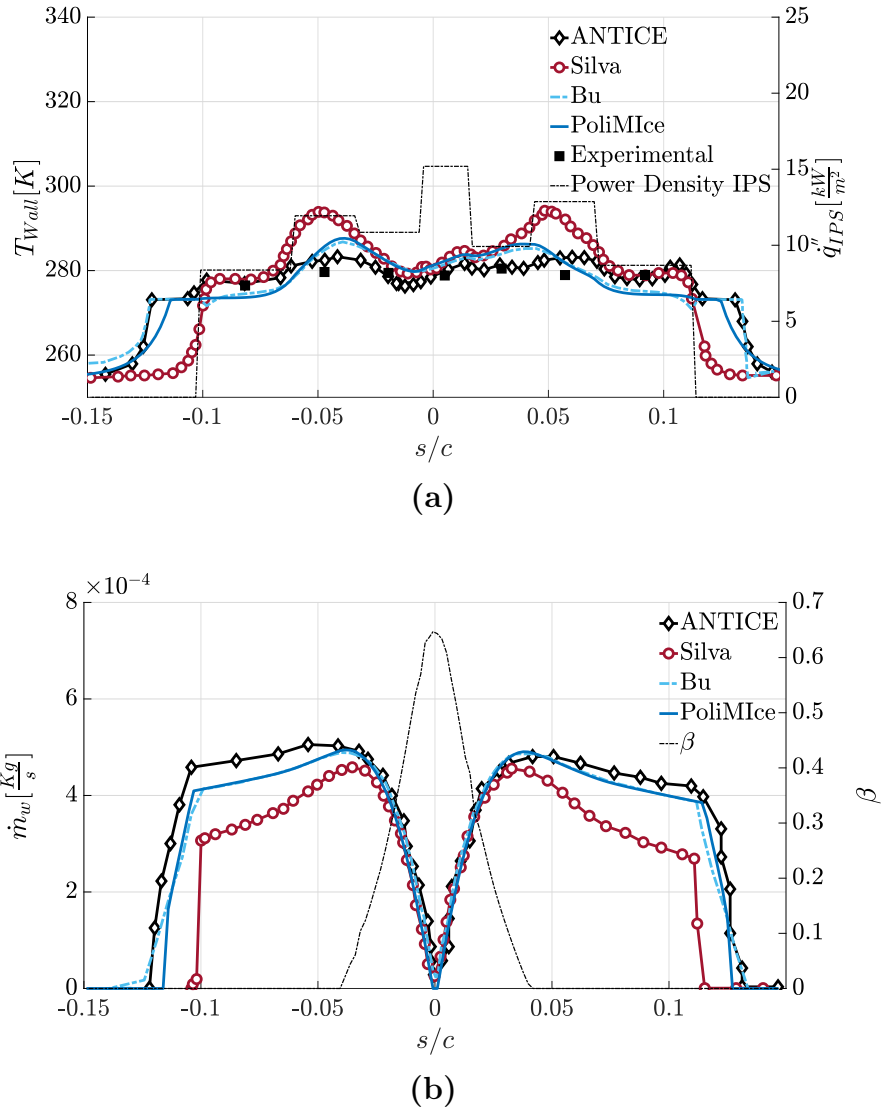
Figure 6.27 Test case 67B. (a) Surface temperature and (b) runback water: comparison among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

neighbouring points due to the higher amount of water in that region. The temperature remains lower with respect to test case 67A due to the lower heating power provided by the heaters and to the higher amount of water that remains liquid on the surface, as it can be seen in figure 6.27. Due to the lower power of the heaters, the runback water reaches the boundaries of the protected region leading to the subsequent formation of runback ice. This is clear from the surface temperature trend: there is a region right after the end of the last heater where the temperature remains constant and equal to the freezing temperature. This behaviour is caused by the release of latent heat in that zone.

The convective heat transfer coefficient and the overall heat transfer coefficient obtained with PoliMIce are shown in figure 6.28. Both are compared with numerical and experimental data. The results obtained with PoliMIce are once again in good agreement with reference results for both the convective heat transfer coefficient and the overall heat transfer coefficient. As it happened in test case 67A, there is a jump in the heat transfer coefficient at the end of the protected region for the same reason that was explained before.

From these results it can be seen that the power supplied by the heater is not sufficient to protect from ice formation in the considered environmental and flight conditions. The heat provided is not enough to evaporate all the mass of water in the protected region, leading to the formation of runback ice which cannot be removed leading to aerodynamic performance degradation and to serious accidents.
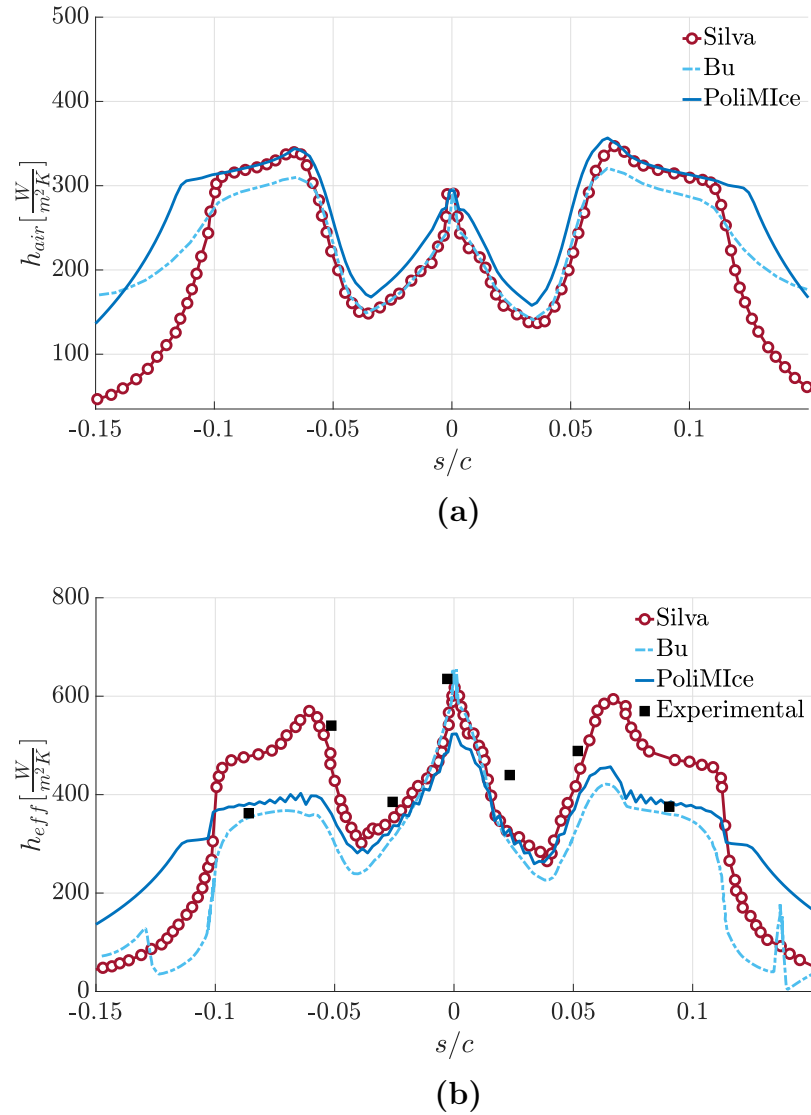
Figure 6.28 Test case 67B. (a) Convective heat transfer coefficient and (b) overall heat transfer coefficient: comparison among experimental results from Al-Khalil [2] and numerical results from ANTICE [2], Bu et al. [13] and Silva et al. [54]. $s < 0$ refers to the pressure side of the airfoil, $s > 0$ to the suction side

Conclusions and future developments

In the present thesis, state-of-the-art liquid film models and a model for a 2D thermo-electric anti-ice system have been studied. The modeling of ice protection systems introduces new kind of boundary conditions to the ice accretion simulations. Thus a complete recoding of the PoliMIce framework has been made to allow more flexibility. Moreover, two different liquid film models have been studied and implemented in order to estimate the influence of liquid film on ice accretion. First, a steady model has been considered with two different numerical approaches: an iterative and an implicit one. The second is a state-of-the-art unsteady liquid film model based on the lubrication theory. All models have been studied and coupled with the already known ice accretion models.

The new version of PoliMIce and the liquid film models have been tested on four different test cases representative of in-flight icing conditions. The computed ice shapes showed good agreement with experimental results with all the implemented models and in all test cases. Clear improvement has been noticed using the unsteady liquid film model especially in glaze and long exposure ice accretion, where the influence of the water film becomes non-negligible.

Further improvement could be achieved by relaxing the hypothesis of constant liquid film temperature to consider heat fluxes in the direction parallel to the surface. Regarding the unsteady film model, an investigation of the effect

of gravity and surface tension could be conducted along with an analysis of different numerical methods for the discretization: an higher order scheme could be implemented for solving the water film thickness transport equation in order to better capture discontinuites like horns. Moreover, more complex liquid film phenomena could be added to the models, like the formation of rivulets or the detachment of water from the ice surface. Lastly, the model could be extended to include the effects of inertia forces on liquid film movement in rotating frames, like an helicopter rotor.

As it was clear, in all test cases irregular geometries were really hard to capture. Therefore, further studies could analyze new methods to evaluate the node displacement and consequent mesh deformation to better capture the direction of ice accretion.

A state-of-the-art IPS mathematical model for anti-ice simulations was studied and implemented considering the influence of the film flow on the airflow boundary layer.

This model has been tested in three different anti-ice condition: fully evaporative, partially evaporative and running-wet. In all cases the results regarding surface temperature, water mass fluxes and heat transfer coefficients showed good agreement with both experimental data and numerical results.

Further improvements to the model could include the modeling of rivulets and film instabilities phenomena that would change the heat transfer occurring in the water layer. Moreover, this IPS model could be coupled with ice accretion models presented in chapter 2 to be able to simulate more complex anti-ice and de-ice conditions, like runback ice or de-ice using an electro-thermal IPS. Lastly, the extension of this model to a 3D environment could be studied in order to expand the capability of the IPS simulations.

Bibliography

[1] B. J. Abu-Ghannam and R. Shaw. "Natural Transition of Boundary Layers—The Effects of Turbulence, Pressure Gradient, and Flow History". In: *Journal of Mechanical Engineering Science* 22.5 (Oct. 1980), pp. 213–228 (cit. on pp. 61, 62).

[2] Kamel Al-Khalil et al. "Validation of NASA thermal ice protection computer codes. III - The validation of ANTICE". In: *35th Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, Jan. 1997 (cit. on pp. 56, 58, 59, 95, 122, 125–136, 138).

[3] K.M. Al-Khalil. "Numerical Simulation of an Aircraft Anti-icing System Incorporating a Rivulet Model for the Runback Water". PhD thesis. University of Toledo, 1991 (cit. on p. 55).

[4] G.S. Ambrok. "Approximate solution of equations for the thermal boundary layer with variations in boundary layer structure". In: *Soviet Physics* 2 (Jan. 1957), pp. 1979–1986 (cit. on p. 63).

[5] C. Antonini et al. "Understanding the effect of superhydrophobic coatings on energy reduction in anti-icing systems". In: *Cold Regions Science and Technology* 67.1 (2011), pp. 58–67 (cit. on p. 12).

[6] Chengxin Bai and A. D. Gosman. "Development of Methodology for Spray Impingement Simulation". In: *SAE Technical Paper Series*. SAE International, Feb. 1995 (cit. on p. 6).

[7] Tommaso Bellosta. "A Lagrangian 3D particle trakcing solver for in-flight ice accretion". MSc thesis. Politecnico di Milano, 2019 (cit. on pp. 79, 83, 123, 126, 131).

[8]   Tommaso Bellosta, Gianluca Parma, and Alberto Guardone. "A robust 3D particle tracking solver for in-flight ice accretion using arbitrary precision arithmetic". In: *VIII International Conference on Computational Methods for Coupled Problems in Science and Engineering*. 2019 (cit. on pp. 13, 82, 96, 97, 124, 130).

[9]   A.C. Boucíguez, R.F. Lozano, and M.A. Lara. "About the Exact Solution in Two-Phase Stefan Problem". In: *Engenharia Térmica (Thermal Engineering)* 6 (2007), pp. 70 –75 (cit. on p. 30).

[10] Y. Bourgault, H. Beaugendre, and W.G. Habashi. "Development of a shallow-water icing model in FENSAP-ICE". In: *Journal of Aircraft* 37.4 (2000), pp. 640–646 (cit. on p. 48).

[11] N.T. Brahimi, P. Tran, and I. Paraschivoiu. "Numerical simulation and thermodynamic analysis of ice accretion on aircraft wings". In: *Center de Development Technologique de Ecole Polytechnique de Montreal, Final report CDT Project C* 159 (1994), p. 1994 (cit. on p. 3).

[12] Andy P. Broeren, Michael B. Bragg, and Harold E. Addy. "Effect of Intercycle Ice Accretions on Airfoil Performance". In: *Journal of Aircraft* 41.1 (Jan. 2004), pp. 165–174 (cit. on p. 10).

[13] Xueqin Bu et al. "Numerical simulation of an airfoil electrothermal anti-icing system". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 227.10 (Oct. 2012), pp. 1608–1622 (cit. on pp. 56, 61, 95, 122, 125–136, 138).

[14] Marc Budinger et al. "Ultrasonic Ice Protection Systems: Analytical and Numerical Models for Architecture Tradeoff". In: *Journal of Aircraft* 53.3 (May 2016), pp. 680–690 (cit. on p. 12).

[15] Samuel T. Buschhorn et al. "Electrothermal Icing protection of Aerosurfaces Using Conductive Polymer Nanocomposites". In: *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. American Institute of Aeronautics and Astronautics, Apr. 2013 (cit. on p. 12).

[16] Tuncer Cebeci and Peter Bradshaw. *Physical and Computational Aspects of Convective Heat Transfer*. Springer Berlin Heidelberg, 1984 (cit. on p. 63).

[17]   Imon Chakraborty et al. "An Integrated Approach to Vehicle and Subsystem Sizing and Analysis for Novel Subsystem Architectures". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 230.3 (July 2015), pp. 496–514 (cit. on p. 10).

[18]   Jeffrey Cole and Wayne Sand. "Statistical study of aircraft icing accidents". In: *29th Aerospace Sciences Meeting*. 1991, p. 558 (cit. on p. 2).

[19]   Raytheon Aircraft Company. *Aircraft Specification and Description*. Tech. rep. Dec. 2004 (cit. on p. 12).

[20]   *Compliance of Transport Category Airplanes with Certification Requirements for Flight in Icing Conditions*. Federal Aviation Administration, Oct. 2014 (cit. on p. 7).

[21]   Carlos Donatti et al. "Ice Accretion Simulation in Presence of a Hot Air Anti-Icing System". In: *19th International Congress of Mechanical Engineering*. Nov. 2007 (cit. on p. 11).

[22]   D. Dussin et al. "Hybrid grid generation for two-dimensional high-Reynolds flows". In: *Computers & fluids* 38.10 (2009), pp. 1863–1875 (cit. on pp. 96, 123).

[23]   Thomas D. Economon et al. "SU2: An open-source suite for multiphysics simulation and design". In: *Aiaa Journal* 54.3 (2016), pp. 828–846 (cit. on pp. 13, 96, 123).

[24]   Mattia Garabelli and Giulio Gori. "PoliMIce: un ambiente di simulazione per la previsione dell'accrescimento di ghiaccio su velivoli". MSc thesis. Politecnico di Milano, 2013 (cit. on pp. 26, 28, 29, 79, 96, 100, 101, 110, 117–119).

[25]   R.W. Gent. *TRAJICE 2, A Combined Water Droplet and Ice Accretion Prediction Program for Aerofoil*. Tech. rep. Farnborough, Hampshire: Royal Aerospace Establishment(RAE), Nov. 1990 (cit. on p. 3).

[26]   R.W. Gent, N.P. Dart, and J.T. Cansdale. "Aircraft Icing". In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 358.1776 (Nov. 2000). Ed. by G. Poots, pp. 2873–2911 (cit. on pp. 1, 4–9).

[27]   *Glaze Ice on a Powered Force Model (PFM) Rotor*. NASA Glenn Research Center. URL: https://www1.grc.nasa.gov/aeronautics/icing/#gallery (cit. on p. 5).

[28]   G. Gori et al. "PoliMIce: A simulation framework for three-dimensional ice accretion". In: *Applied Mathematics and Computation* 267 (2015), pp. 96 –107 (cit. on pp. 3, 12, 14, 16).

[29]   Giulio Gori et al. "Local Solution to the Unsteady Stefan Problem for In-Flight Ice Accretion Modeling". In: *Journal of Aircraft* 55.1 (Jan. 2018), pp. 251–262 (cit. on pp. 1, 16, 19, 30, 34).

[30]   Yiqiang Han. "Theoretical and Experimental Study of Scaling Methods for Rotor Blade Ice Accretion Testing". PhD thesis. Aug. 2011 (cit. on p. 2).

[31]   J. K. Hardy. "Protection of Aircraft Against Ice". In: *The Journal of the Royal Aeronautical Society* 51.435 (Mar. 1947), pp. 271–305 (cit. on p. 2).

[32]   Sharon Monica Jones et al. *Subsonic Aircraft Safety Icing Study*. Tech. rep. NASA Langley Research Center, 2008 (cit. on p. 2).

[33]   William M. Kays. *Convective Heat and Mass Transfer*. McGraw-Hill Science/Engineering/Math, 1993 (cit. on pp. 61, 69).

[34]   E. Linacre and B. Geerts. *Cloud Liquid Water Content, Drop Sizes, and Number of Droplets*. 1999. URL: http://www-das.uwyo.edu/~geerts/cwx/notes/chap08/moist_cloud.html (cit. on p. 7).

[35]   Decang Lou and David W. Hammond. "Heat and mass transfer for ice particle ingestion inside aero-engine". In: *Journal of turbomachinery* 133.3 (2011) (cit. on p. 2).

[36]   B.L. Messinger and Lockheed Aircraft Corporation. *Equilibrium Temperature of an Unheated Icing Surface as a Function of Air Speed*. publisher not identified, 1953 (cit. on pp. 2, 15, 16).

[37]   Francois Morency, Fatih Tezok, and Ion Paraschivoiu. "Anti-Icing System Simulation Using CANICE". In: *Journal of Aircraft* 36.6 (Nov. 1999), pp. 999–1006 (cit. on pp. 56, 71).

[38] Francois Morency et al. "FENSAP-ICE-A comprehensive 3D simulation system for in-flight icing". In: *15th AIAA Computational Fluid Dynamics Conference*. 2001, p. 2566 (cit. on p. 3).

[39] Tim G. Myers. "Extension to the Messinger Model for Aircraft Icing". In: *AIAA Journal* 39.2 (Feb. 2001), pp. 211–218 (cit. on pp. 4, 15, 18, 20, 22, 23, 26, 48, 103).

[40] Tim G. Myers and Jean P.F. Charpin. "A mathematical model for atmospheric ice accretion and water flow on a cold surface". In: *International Journal of Heat and Mass Transfer* 47.25 (2004), pp. 5483–5500 (cit. on pp. 43, 44, 48).

[41] Alexander Oron, Stephen H. Davis, and S. George Bankoff. "Long-scale evolution of thin liquid films". In: *Reviews of modern physics* 69.3 (1997), p. 931 (cit. on p. 44).

[42] Gianluca Parma. "A model for in-flight ice accretion based on the exact solution of the unsteady Stefan problem". MSc thesis. Politecnico di Milano, 2015 (cit. on pp. 27, 31, 33).

[43] Mark G. Potapczuk. "Aircraft Icing Research at NASA Glenn Research Center". In: *Journal of Aerospace Engineering* 26.2 (Apr. 2013), pp. 260–276 (cit. on p. 2).

[44] A.J. Press and N.P. Dart. *ICECREMO user guide–version 3*. Tech. rep. ICECREMO/SPEC/DERA/ND990506/3, 2000 (cit. on p. 3).

[45] Thomas Ratvasky and Kurt Blankenship. *Rime Ice*. NASA Glenn Research Center. URL: https://flightsafety.org/asw-article/discoveries-on-ice/ (cit. on p. 5).

[46] William C. Reynolds, William Morrow Kays, and Stephen Jay Kline. *Heat Transfer in the Turbulent Incompressible Boundary Layer. IV; Effect of Location of Transition and Prediction of Heat Transfer in a Known Transition Region*. Tech. rep. Washington DC: NASA Headquarters, 1958 (cit. on pp. 66, 67).

[47] Philip L. Roe. "Approximate Riemann solvers, parameter vectors, and difference schemes". In: *Journal of computational physics* 43.2 (1981), pp. 357–372 (cit. on pp. 96, 124).

[48] Francesco De Rosa. "Electrically Heated Composite Leading Edges for Aircraft Anti-icing Applications". PhD thesis. University of Naples "Federico II", 2010 (cit. on p. 11).

[49] Gary A. Ruff and Brian M. Berkowitz. *Users manual for the NASA Lewis ice accretion prediction code (LEWICE)*. Tech. rep. Cleveland, OH: NASA Lewis Research Center, 1990 (cit. on pp. 3, 95, 100, 101, 103, 104).

[50] Galen B. Schubauer and Philip S. Klebanoff. *Contributions on the mechanics of boundary-layer transition*. Tech. rep. NACA, 1955 (cit. on p. 66).

[51] Jaiwon Shin and Thomas H. Bond. *Experimental and computational ice shapes and resulting drag increase for a NACA 0012 airfoil*. Tech. rep. NASA Lewis Research Center, 1992 (cit. on pp. 95, 104, 107).

[52] Guilherme Silva, Otávio Silvares, and Euryale Zerbini. "Airfoil Anti-ice System Modeling and Simulation". In: *41st Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, Jan. 2003 (cit. on p. 55).

[53] Guilherme A. L. Silva, Otavio M. Silvares, and Euryale J. G. J. Zerbini. "Numerical Simulation of Airfoil Thermal Anti-ice Operation, Part 1: Mathematical Modelling". In: *Journal of Aircraft* 44.2 (Mar. 2007), pp. 627–633 (cit. on pp. 13, 56, 61, 70).

[54] Guilherme A. L. Silva, Otavio M. Silvares, and Euryale J. G. J. Zerbini. "Numerical Simulation of Airfoil Thermal Anti-ice Operation, Part 2: Implementation and Results". In: *Journal of Aircraft* 44.2 (Mar. 2007), pp. 634–641 (cit. on pp. 13, 56, 57, 95, 122, 125–134, 136, 138).

[55] A. G. Smith and D. B. Spalding. "Heat Transfer in a Laminar Boundary Layer with Constant Fluid Properties and Constant Wall Temperature". In: *The Journal of the Royal Aeronautical Society* 62.565 (Jan. 1958), pp. 60–64 (cit. on p. 63).

[56] Philippe Spalart and Steven Allmaras. "A one-equation turbulence model for aerodynamic flows". In: *30th aerospace sciences meeting and exhibit*. 1992, p. 439 (cit. on p. 97).

[57] D. B. Spalding. *Convective Mass Transfer*. McGraw-Hill Book Company, 1963 (cit. on pp. 68, 70).

[58] J. Stefan. "Über die Theorie der Eisbildung, insbesondere über die Eisbildung im Polarmeere". In: *Annalen der Physik* 278.2 (1891), pp. 269–286 (cit. on pp. 15, 35).

[59] John Steuernagle, Kathleen Roy, and David. *Aircraft Icing*. Tech. rep. 2008 (cit. on pp. 1, 3, 10).

[60] Arnold Tafferner et al. "Improved thunderstorm weather information for pilots through ground and satellite based observing systems". In: *90th AMS Annual Meeting*. Sept. 2010 (cit. on p. 2).

[61] Scott K. Thomas, Robert P. Cassoni, and Charles D. MacArthur. "Aircraft anti-icing and de-icing techniques and modeling". In: *Journal of aircraft* 33.5 (1996), pp. 841–854 (cit. on p. 37).

[62] Venkat Venkatakrishnan. "Convergence to steady state solutions of the Euler equations on unstructured grids with limiters". In: *Journal of computational physics* 118.1 (1995), pp. 120–130 (cit. on p. 97).

[63] *Wing Icing*. Pilot Getaways. URL: `http://pilotgetaways.com/mag/nd09/icing` (cit. on p. 2).

[64] W. Wright, K. Dewitt, and Jr. T. Keith. "Numerical Simulation of Icing, Deicing, and Shedding". In: *29th Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, Jan. 1991 (cit. on p. 55).

[65] William Wright. *User's manual for LEWICE version 3.2*. Tech. rep. Cleveland, OH, United States: NASA Glenn Research Center, 2008 (cit. on p. 3).

[66] William B. Wright, R.W. Gent, and Didier Guffond. *DRA/NASA/ONERA Collaboration on Icing Research. Part 2; Prediction of Airfoil Ice Accretion*. Tech. rep. NASA Glenn Research Center, 1997 (cit. on pp. 95, 108, 110, 112).

[67] William B. Wright and Adam Rutkowski. *Validation Results for LEWICE 2.0. [Supplement]*. Tech. rep. NASA Lewis Research Center, 1999 (cit. on pp. 113, 115, 117, 118, 120).