



Technical University of Crete
School of Production Engineering and Management

Parametric Geometrical Modeling of a Diffuser Augmented Wind Turbine

By

Kyriakos Charalampous

A dissertation submitted in partial fulfilment of the
requirements for the degree of

Master of Science

Supervisor

Dr. Ioannis K. Nikolos, Associate Professor

Chania, June 2017

Copyright © Kyriakos Charalampous, 2017.
All rights reserved.



Technical University of Crete
School of Production Engineering and Management

Parametric Geometrical Modeling of a Diffuser Augmented Wind Turbine

By

Kyriakos Charalampous

Approved by:

Dr. Ioannis K. Nikolos
Associate Professor
Technical University of Crete
School of Production Engineering
& Management

Dr. Anargiros Delis
Associate Professor
Technical University of Crete
School of Production Engineering
& Management

Dr. Fotios Kanellos
Assistant Professor
Technical University of Crete
School of Production Engineering
& Management

Date Defended: 17-07-2017

“Intentionally Left Blank”

Acknowledgements

I would like to thank Associate Professor Ioannis K. Nikolos for the fruitful discussions and guidance throughout this dissertation. I would also like to thank my colleagues Angelos Klothakis, Giorgos Strofylas, Stavros Leloudas and Dimitrios Inglezakis for the companionship and cooperation all these years.

I would like to thank also Omnilink GreenTech for providing me with the opportunity to participate in their research projects in cooperation with the Technical University of Crete. The experience gained during this research work is invaluable. More specifically, part of this work was made in the context of two research projects with the previously-mentioned company, which is greatly acknowledged.

Finally, I would like to thank my family, for their endless love and support.

Kyriakos Charalampous

“Intentionally Left Blank”

Abstract

The development and application of renewable energy has become an important issue in recent years due to the serious effects of global warming and the inevitable reduction of the fossil fuel deposits. Driven by an increasingly energy hungry world and a push towards sustainable and environmentally friendly energy sources, the wind energy is currently one of the most rapidly growing industries. Current designs, such as conventional horizontal axis wind turbines (HAWTs) require expansive designs geometrically rather than aerodynamically augmented. Therefore, it is necessary to develop innovative wind capturing devices that can produce energy in the locations where large HAWTs are impractical to be installed.

The Diffuser Augmented Wind Turbine (DAWT) is one of the innovative wind energy conversion systems that have been investigated over the last decades, to decrease the cost of wind energy harvesting and provide a sustainable solution for small scale energy production in site-specific cases. DAWTs enhance the power output of the rotor by increasing the speed of the approaching wind, through the employment of a static shroud. Therefore, a simple and quick design tool is necessary for designers to develop efficient wind energy conversion systems. This work presents a methodology for the parametric design of a Diffuser Augmented Wind Turbine, developed in Grasshopper®, a graphical algorithm editor tightly integrated into Rhinoceros 3D CAD application. The geometric algorithm gives the user the ability to design the main parts composing a complete DAWT system, including the turbine blades, the centrebody, the diffuser geometry, as well as the internal flap, the tower and the central column structures.

The developed algorithm eventually produces compound solid DAWTs, which can be further imported to mesh generation and analysis programs through standard geometry exchange protocols, for cooperation with Computational Fluid Dynamics (CFD) and Computational Structural Dynamics (CSD) solvers. To this end, a simplified DAWT design, produced by the proposed design methodology, is used to produce hybrid unstructured mesh, to be used for CFD analysis of the flow through the DAWT (with a rotating rotor). This mesh generation is used to demonstrate the feasibility of the design process.

“Intentionally Left Blank”

Contents

Acknowledgements	I
Abstract.....	III
Contents	V
List of Figures.....	VII
List of Tables.....	XI
Introduction.....	1
1. Historical Background and Operation Principles.....	2
1.1 Wind Turbines	2
1.2 Diffuser Augmented Wind Turbines	3
1.2.1 Basic DAWT Physics	3
1.2.2 Principal DAWT Investigations	6
1.2.2.1 The beginning of DAWT geometry evolution	6
1.2.2.2 Final generation of DAWT designs	9
1.2.2.3 Further DAWT investigations	12
2. Methodology	18
2.1 Introduction to Grasshopper®	18
2.2 Introduction to the algorithm.....	21
2.2.1 Blades	21
2.2.1.1 External surface.....	21
2.2.1.2 The Internal Structure.....	23
2.2.2 DAWT Design.....	28
2.2.2.1 The Diffuser	28
2.2.2.2 The Internal Flap	32
2.2.2.3 The Hub.....	35
2.2.2.4 Central Column and Tower	37
2.2.2.5 DAWT Assembly.....	38
3. Mesh Generation	45
3.1 Surface mesh.....	45
3.1.1 Volume I.....	45
3.1.1.1 Blades mesh.....	45
3.1.1.2 Nose Cone and Blade Connectors Mesh	51
3.1.1.3 Rotating Cylinder Mesh	53

3.1.2 Volume II.....	54
3.1.2.1 Diffuser Mesh.....	54
3.1.2.2 Internal Flap Mesh	56
3.1.2.3 Central Column and Hub Mesh.....	57
3.1.2.4 Tower, Flow Domain (Box) and Cylinder Mesh	58
3.2 Layers Construction.....	60
3.2.1 Volume I.....	60
3.2.1.1 Blades layers	60
3.2.1.2 Nose Cone Prismatic Layers Construction.....	63
3.2.2 Volume II.....	64
3.2.2.1 Diffuser Layers.....	64
3.2.2.2 Internal Flap, Central Column and Hub Layers	66
3.2.2.3 Tower and Flow Domain (Box) Layers	67
3.3 Volume Mesh	69
3.3.1 Volume I.....	69
3.3.2 Volume II.....	72
4. Conclusions and Recommendations	76
Bibliography	77
Appendix.....	80
Non-Uniform Rational B-Splines (NURBS).....	80
The Knot Vector.....	81
Curve/Surface Definition	82
Computational Algorithm	83
The Weights	84

List of Figures

Figure 1.1: Wind Turbines Evolution.	3
Figure 1.2: Wind energy extraction [Hol81].....	4
Figure 1.3: Diffuser augmented wind energy extraction, combined with wake and external flow mixing [Hol81].....	5
Figure 1.4: Layout of the first phase shroud design of Kogan [Car14].	7
Figure 1.5: Effect of boundary layer control slots, in preventing separation within a DAWT [Gil83].....	8
Figure 1.6: Igra's boundary layer control models [Igr77].	10
Figure 1.7: Igra's Model 3 rear view [Igr77].....	11
Figure 1.8: Schematic diagram of the modifications made to the Vortec 7 geometry [Phi03]. ...	13
Figure 1.9: The full-scale modified Vortec 7 [Phi03].....	13
Figure 1.10: Illustration of the scoop design proposed by Wang et al [Wan08].	14
Figure 1.11: Flow around a brimmed diffuser [Ohy10].	15
Figure 1.12: 500W wind-lens turbine [Ohy10].....	15
Figure 1.13: DAWT concept by Flodesign [Flod].....	17
Figure 1.14: DONQI Urban Windmill [Dor11].....	17
Figure 2.1: An overview of the developed graphical algorithm within the Grasshopper® environment.	18
Figure 2.2: Grasshopper® components and wires [Cha14].....	19
Figure 2.3 : Illustration of Grasshopper® data tree structure [Tree2].	20
Figure 2.4: The external surface of a small-scale wind turbine blade.	22
Figure 2.5: A separate blade region [Cha14].	23
Figure 2.6: The division of a blade region in 7 surface patches.	24
Figure 2.7: Resulting solid blade (Volume Difference).....	25

Figure 2.8: Example of a blade with two shear webs [Cha15].	26
Figure 2.9: Illustration of the final solid blade geometry [Cha15].	27
Figure 2.10: Demonstration of the “Scale” operation.	29
Figure 2.11: Demonstration of the “Translate” operation.	29
Figure 2.12: Illustration of the curves generated by the 360° rotation of the original diffuser profile curve.	30
Figure 2.13: Illustration of the external diffuser surface.	30
Figure 2.14: Various diffuser geometries produced by the algorithm.	31
Figure 2.15: Application of the “Scale” operation in the case of a flap profile.	32
Figure 2.16: The division of the internal flap profile curve based on the points located at the leading and trailing edges.	33
Figure 2.17: The resulted flap surface.	34
Figure 2.18: The hub profile curve.	35
Figure 2.19: The resulted hub surface.	36
Figure 2.20: Illustration of the blades’ connection to the hub.	36
Figure 2.21: The tower surface.	37
Figure 2.22: The surface of the central column structure.	38
Figure 2.23: The Solid union component.	39
Figure 2.24: The central column joined with the internal flap (1 st pair).	39
Figure 2.25: The diffuser joined with the tower (2 nd pair).	40
Figure 2.26: Union of the first two geometry pairs.	41
Figure 2.27: The complete DAWT system.	42
Figure 2.28: Illustration of the final DAWT system [Cha17].	43
Figure 2.29: Illustration of the final DAWT system.	44
Figure 3.1: Illustration of Volume I (Rotating Frame).	46

Figure 3.2: Illustration of Volume II (Inertial Frame).	46
Figure 3.3: The division of the blade into the root (blue), the middle (white) and the tip (red) regions.	47
Figure 3.4: The generated surface meshes for each blade region.	48
Figure 3.5: Close-up views of the leading (up) and trailing (down) edges at the root region of the blade.	49
Figure 3.6: Close-up views of the leading (up) and trailing (down) edges at the tip of the blade.	50
Figure 3.7: The surface mesh of a blade connector.	51
Figure 3.8: The surface mesh of the nose cone.	52
Figure 3.9: The surface mesh of the rotating cylinder, close to the blade tip region.	53
Figure 3.10: The diffuser's surface division into regions of different mesh parameters.	54
Figure 3.11: The surface mesh at the leading edge (up) and the main body (down) of the diffuser.	55
Figure 3.12: Close-up view of the internal flap's leading edge.	56
Figure 3.13: The surface mesh of the hub and central column.	57
Figure 3.14: The surface mesh of the tower and the flow domain's bottom surface (ground).	58
Figure 3.15: The surface mesh of the components included in Volume II.	59
Figure 3.16: Illustration of the generated prismatic layers near the blade's leading edge.	61
Figure 3.17: Illustration of the generated prismatic layers at the blade tip region.	62
Figure 3.18: Illustration of the generated prismatic layers around the nose cone.	63
Figure 3.19: The generated prismatic layers at the diffuser's leading edge (up) and middle surface (down).	65
Figure 3.20: The generated prismatic layers around the internal flap (up) and the central column (down).	66
Figure 3.21: Illustration of the generated prismatic layers around the hub.	67
Figure 3.22: The generated prismatic layers of the tower (up) and the flow domain's bottom surface (down).	68

Figure 3.23: Illustration of the volume mesh around the nose cone (up) and lengthwise the blade (down)..... 70

Figure 3.24: Close-up views of the volume mesh around the blade root (up) and near the blade tip (down)..... 71

Figure 3.25: Illustration of the volume mesh around the DAWT..... 72

Figure 3.26: Illustration of the volume mesh around the diffuser. 73

Figure 3.27: Illustration of the volume mesh near the tower’s root (up) and along the tower (down)..... 74

List of Tables

Table 3.1: The parameters for the surface mesh generation of each blade region.....	47
Table 3.2: Leading and trailing edge perimeter lengths.	48
Table 3.3: The parameters for the surface mesh generation of the blade connector.	51
Table 3.4: The parameters for the surface mesh generation of the nose cone.....	52
Table 3.5: The parameters for the surface mesh generation of the rotating cylinder.	53
Table 3.6: The parameters for the surface mesh generation of the diffuser.	54
Table 3.7: The parameters for the surface mesh generation at the internal flap.	56
Table 3.8: The parameters for the surface mesh generation of the hub and central column.	57
Table 3.9: The parameters for the surface mesh generation of the tower and flow domain.....	58
Table 3.10: The parameters for the blades layers construction.	60
Table 3.11: The parameters for the nose cone layer construction.	63
Table 3.12: The parameters for the diffuser prismatic layers construction.	64
Table 3.13: The parameters for layers construction of the tower and the flow domain's bottom surface.	67
Table 3.14: The parameters for the volume mesh construction of Volume I.	69
Table 3.15: The parameters for the volume mesh construction of Volume II.	72
Table 3.16: Final Mesh Statistics.....	75

“Intentionally Left Blank”

Introduction

In recent years, the ever increasing world energy consumption, in combination with the global awareness about the adverse environmental impact of the intense fossil fuels exploitation, have pushed towards the revision of traditional energy practices and the systematic investigation, development and application of alternative energy production technologies. Under this framework, the wind power sector is nowadays one of the fastest growing industries and the most substantial provider of renewable and sustainable energy [Hjo15]. However, the unpredictable nature of the wind and its relatively dilute energy content, are significant barriers to the even broader adoption and application of wind turbines technology, since these factors incline to upsurge the cost of the derived energy [Gil79].

Thus, in an effort to overcome this deficiency and put wind power in a competitive basis with conventional fuels in terms of energy production cost, the development of efficient wind concentrators that substantially increase the energy density of the wind, and therefore enhance the power output of wind turbines, has long been pursued [Phi03, Sho16]. The diffuser augmented wind turbine (DAWT) can be viewed as such an advanced wind energy conversion system, which is based on the attachment of a static diffuser around the turbine, to control the expansion of the wake and create a region of high subatmospheric pressure at the diffuser exit; a phenomenon that eventually results in the augmentation of the mass flow rate passing through the turbine. Hence, such a configuration has the potential to reach in a power performance coefficient beyond the Betz limit, and thus to extract additional power from the wind, compared to a conventional bare wind turbine with the same rotor diameter [Lil56, Gil78, Igr81, Han00, Phi03].

All available computational methodologies are currently applied for the design, analysis and optimization of wind turbines, such as FEM (Finite Element Method), CFD (Computational Fluid Dynamics), automated mesh generation and mesh deformation methodologies. As understood from above, the complexity of a wind turbine design has led to the creation of specialized software, through which the designer has the ability to produce the final product, taking into account various design features. Although a variety of methodologies have been reported in the open literature for the design and geometrical definition of wind turbine systems, there is still much room for further development, as the use of renewable energy technologies is constantly increasing.

In this study, a methodology for the parametric geometrical definition of a complete diffuser augmented wind turbine system, including the internal structure of the blades, was developed. Parametric design provides the user with the ability to obtain the final geometry in a simple way, while allows for the automatic generation of a completely different geometric design, by only altering the values of the established design parameters. Finally, as long as the definition of the blade's internal structure is concerned, the blade is divided into a finite number of non-overlapping regions, to assign different laminate thicknesses and materials [Gur14, Roo04]. Eventually, a number of sample meshes are presented, to demonstrate the capabilities of the particular algorithm.

The rest of this dissertation is organized as follows: In Chapter 1, a brief literature review on the diffuser augmented wind turbine concept is undertaken, along with a presentation of the basic governing principles. In Chapter 2, the proposed methodology for the parametric design of DAWTs is outlined, while in Chapter 3, the generation of an unstructured 3D computational

mesh around a DAWT design, produced by the particular algorithm, is demonstrated. Finally in Chapter 4, the conclusions emerged from this work, along with future work are presented.

1. Historical Background and Operation Principles

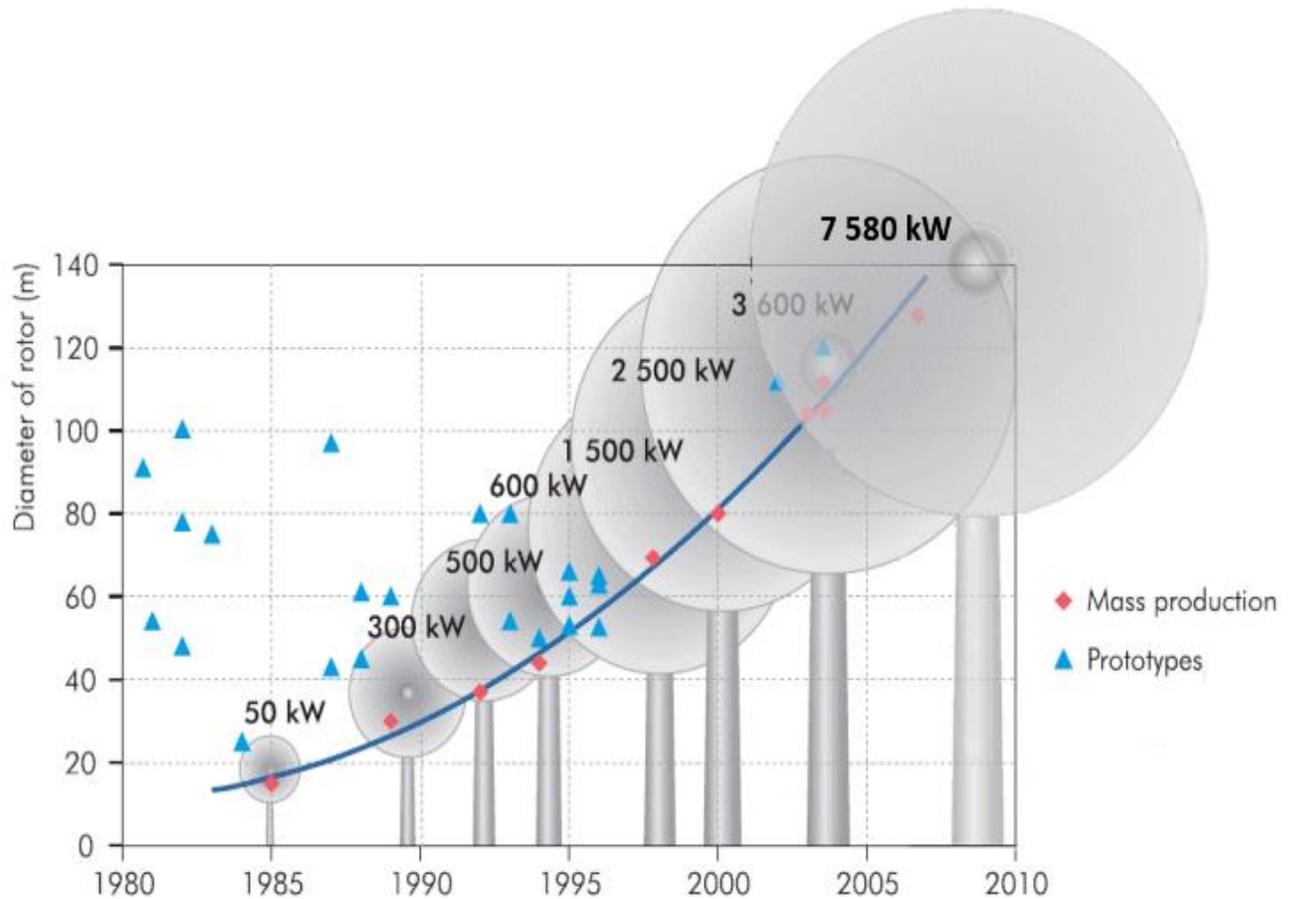
1.1 Wind Turbines

Since the beginning of history, man began to exploit wind energy. As early as 5000 BC, wind energy was used for the transportation of people and their goods by boats and rafts along the Nile River, while a few centuries later, the Persians used it for pumping water and grinding grains. As civilizations began to realize the power provided by the wind, the windmills spread from Persia to the surrounding areas of the Middle East, where they were widely used for food production.

Eventually, around 1000 AD, the wind technology also spread in northern European countries, such as the Netherlands, which adapted windmills in order to drain lakes and tributaries along the Rhine River. However, in contrast to the middle-eastern windmills, the European windmills were horizontal axis wind turbines and were used for almost any mechanical task (pumping water, sanding wood, etc.). An interesting feature of the early windmills was that they were built on posts, such that the whole windmill could be turned according to the wind direction. Also, these European horizontal axis windmills were lift-based machines and thus, they seemed to be more efficient than the middle-eastern vertical axis windmills, which were based on the drag force [Car14, Cha14].

The history of modern wind turbines begins in the early 1980s, as a result of the oil crisis of the 1970s. The security of energy supply and sustainability created a strong interest in the use of renewable energy, motivating the rapid development and production of wind turbines. However, the manufacturing and installation costs for a large wind turbine is very high. This is counterpoised by the fact that in the last decades wind turbines performance has increased remarkably, whilst the corresponding cost of energy has been decreased. At the same time, the technological background and the necessary computational design tools have been evolved greatly, so as to meet the current requirements; the evolutionary period of wind turbines is far from over. While higher velocity on-shore sites are getting fewer and fewer, a new road of off-shore wind sites has opened up. Figure 1.1 demonstrates the evolution of the size and power extraction capabilities of today's wind turbines [Car14].

One of the most important stages during the design of a wind turbine is the blading definition. The blades are designed to deliver maximum power with minimum cost. The blades selection is made according to the aerodynamic load, the construction cost and the efficiency of each turbine. Another key factor in the study of a wind turbine system is the choice of the construction materials of the blades, which defines the ideal material thickness according to the acceptable aerodynamic loads, without any significant reduction of the aerodynamic performance [Cha14]. Consequently, the designer aims to achieve a fine compromise between structural integrity and aerodynamic performance, at the lowest possible cost.



Source: International Energy Agency (IEA)

Figure 1.1: Wind Turbines Evolution.

1.2 Diffuser Augmented Wind Turbines

1.2.1 Basic DAWT Physics

The diffuser augmented wind turbine is an innovative wind energy conversion concept, which is based on the effect created by placing a diffuser around a bare wind turbine, to increase its power output due to the acceleration of the wind velocity that approaches the turbine. To accomplish this, DAWTs take advantage of two basic power augmentation principles. The first principle is to increase the mass flow rate through the turbine, whereas according to the second one, further mass flow is achieved by the mixing of the wake and the external flow behind the rotor, in a lower pressure than the atmospheric [Hoo09, Le16].

During the operation of a bare wind turbine at the Betz limit, the velocity at the rotor plane is reduced to $2/3$, compared to the freestream velocity. The reduction in flow velocity drives to a pressure rise in front of the rotor and this causes a fraction of the mass flow to be pushed sideways around the rotor. This loss in mass flow reduces the rotor's effective area to $2/3$ of the swept rotor surface. The first power augmentation principle as depicted in Figure 1.2, is based on

the fact that the aforementioned area could be increased by exerting a perpendicular force to the oncoming wind. According to the third law of Newton the oncoming air stream will try to exert a counteracting force, so that force equilibrium is reached. The radial force is generated by placing an annular lifting device around the rotor with its suction side pointing towards the hub [Hoo09].

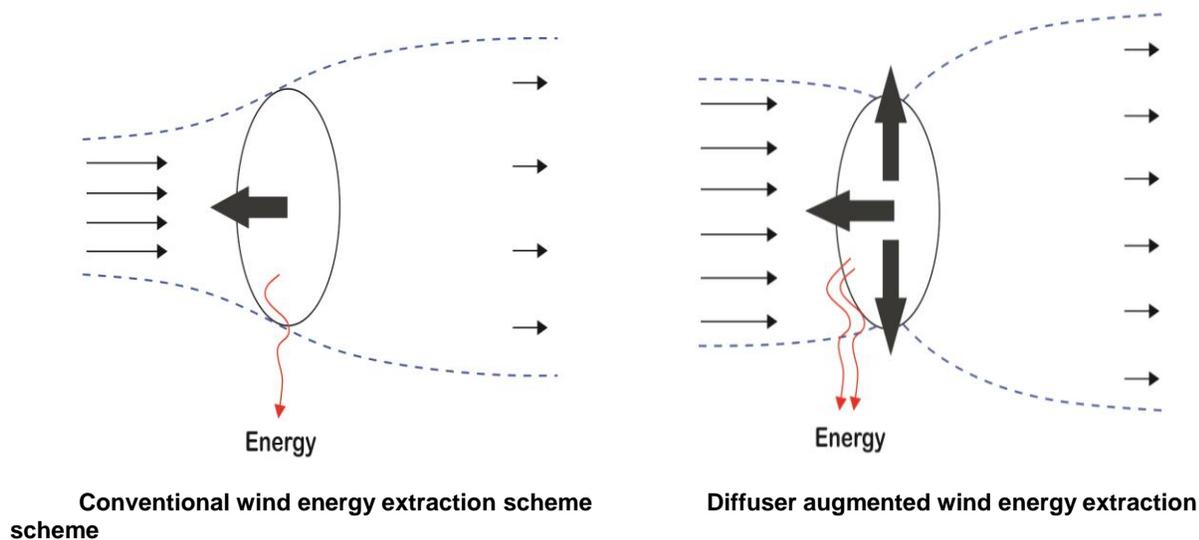


Figure 1.2: Wind energy extraction [Hol81].

Similarly, DAWTs increase the effective area of the rotor by adding a radial force to counteract the pressure force pushing the airflow around the rotor. Consequently, by adding a duct/diffuser, a force imbalance is produced and the only way for the system to achieve an equilibrium state is to increase the mass flow through the duct/diffuser, thus increasing the effective area of the turbine. The augmentation of the mass flow causes an increase in flow velocity through the rotor and since the power output is proportional to the cube of the velocity, the power extracted from the wind is increased [Lel16].

Another theory explaining the effect of the increased mass flow by placing an annular lifting surface around the rotor is the bound vorticity theory. An annular airfoil, with its suction side pointed towards the center, drives the air flow on the inside to accelerate. The suction is related to the lift of the airfoil and this is again, according to the Kutta-Joukowski theorem, related to the bound vorticity. The annular airfoil generates an inward radial lift force, which is accompanied by a ring vortex. According to the Biot-Savart law, the ring vortex will induce a higher velocity on the suction side and, thus, the enhancement of mass flow through the rotor plane. In other words, the bound vorticity increases the rotor swept area, and consequently the “swallowing” capacity of the complete system [Hoo09, Lel16].

The second power augmentation principle is based on mixing effects. When a rotor extracts energy from an air flow, it will cause, like every other obstacle in a flow, the generation of a wake. The wake behind the rotor has a pressure and a velocity deficit, compared to the undisturbed free stream flow. A low pressure behind the rotor is favorable but a low wake

velocity is not. Thereafter, the power augmentation of a DAWT is a direct consequence of the sub-atmospheric pressure around the rotor and the exit plane of the shroud. Therefore, ideally one wants to maintain a situation behind the rotor where the pressure is low while maintaining the mass flow in high levels. The high mass flow can be achieved by the wake expansion, which is already the case with a normal operating wind turbine, or by an increase in wake velocity.

According to the mixing principle, by mixing the wake flow with the undisturbed free stream flow, the momentum deficit behind the rotor is restored. The undisturbed free stream flow will in turn provide the extra momentum for the “exhausted” rotor wake flow to recover from the velocity deficit. Another effect of the mixing leads the wake to have an additional expansion which provides the rotor wake flow with more volume. By increasing the wake volume for the same mass flow through the diffuser, a lower exit pressure behind the rotor is achieved and thus an increased suction. Another advantage of the DAWT geometry is that the vortices created near the blade tips are reduced due to their close proximity to the diffuser wall. As a result, the mixing potential behind the exit plane of a DAWT is expected to be higher in contrast to a normal wind turbine [Hoo09, Lel16].

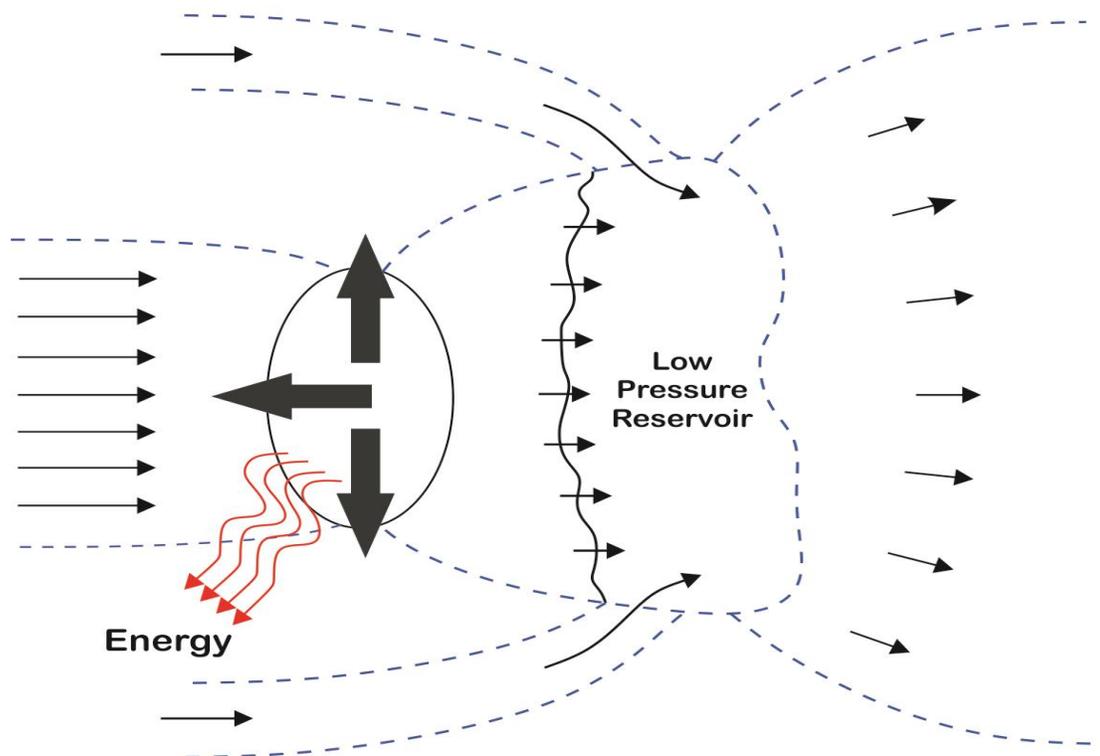


Figure 1.3: Diffuser augmented wind energy extraction, combined with wake and external flow mixing [Hol81].

1.2.2 Principal DAWT Investigations

1.2.2.1 The beginning of DAWT geometry evolution

The birth of the DAWT concept, though in a primitive form, is attributed to the pioneer of wind turbines technology, Albert Betz, who was one of the first to formulate a preliminary DAWT theory, in which it was assumed that the pressure at the diffuser exit plane was equal to the ambient pressure. However, besides the establishment of an early DAWT theory, Betz performed a study on the economic viability of the DAWT concept, compared to a bare wind turbine, concluding that the back then designs and applications of ducted wind turbines were unprofitable, mainly due to the duct's enormous manufacturing and installation cost. Based on these conclusions, the idea of ducted wind turbines was abandoned as a competitive and viable source of energy. However, at the beginning of 1950s, the global interest in DAWTs was triggered by the experimental results, published by the Japanese researcher Sanuki [San50], indicating a gain in power output up to 88% relative to the Betz limit. Three years later, the specific publication was followed by Iwasaki [Iwa53], who achieved a power increase of 30% by placing a cylindrical duct around a rotor.

In 1956, the British researchers Lilley and Rainbird [Lil56] published their theoretical studies on ducted windmills. In this work, they suggested that the increase in axial velocity and a reduction of blade tip losses would be the major contributors to the increase in extracted power from a duct, resulting in a potential augmentation in power output of at least 65% [Lil56]. These two parameters were in turn a function of internal frictional losses, diffuser exit area ratio and the external shape of the duct. They also suggested that the addition of an aerodynamic surface at the diffuser exit, like a flap, could further raise the power augmentation due to a decreased sub-atmospheric exit pressure. Finally, Lilley and Rainbird [Lil56] established some preliminary guidelines for the efficient diffuser design as well, indicating that the duct's geometry should provide an exit-area-ratio of 3.5 with an inlet area ratio of no more than 1.5 in order to maximize the flow uniformity across the rotor plane and reduce the unsteadiness of gusts.

By the end of the 1950s and into the 1960s, both theoretical and experimental investigations on the DAWT concept were undertaken by an Israeli research team, headed by professor Kogan, concluding that the augmentation potentiality was dependent on the rotor loading, the diffuser pressure recovery and the diffuser exit pressure. One of the first duct designs that Kogan explored is depicted in Figure 1.4. During this research, power augmentations around 3.5 was achieved. However, the oversized shroud, and thus the rising cost, led to a commercially uncompetitive and unsustainable product. Even worse, due to the nature of the air flow through the diffuser, the diffuser could not be shortened while maintaining the same exit-area-ratio and the resulting low exit pressure, which was responsible for such high augmentation values [Kog63a, Kog63b].

Furthermore, for apex angles beyond 90° , serious flow separation inside the diffuser was observed. However, the problem of flow separation with increasing diffuser apex angles could be treated with a ring shaped flap at the diffuser exit plane, resulting in the reduction of the pressure and therefore support the flow to overcome the positive pressure gradients within the diffuser. [Hoo09]. The optimal local disk loading coefficient (which is the pressure drop across the screen divided by the local dynamic pressure of the air) was shown to range between 0.18 and 0.22, while this work was the first to highlight the great importance of the sub-atmospheric exit pressures concerning the viability of the DAWT concept. Although high augmentation factors were achieved, as discussed by Igra and reported by Phillips [Igr81, Phi03], the high blockage

effect and the close proximity of the diffuser exit to the wind tunnel wall, made the results questionable.

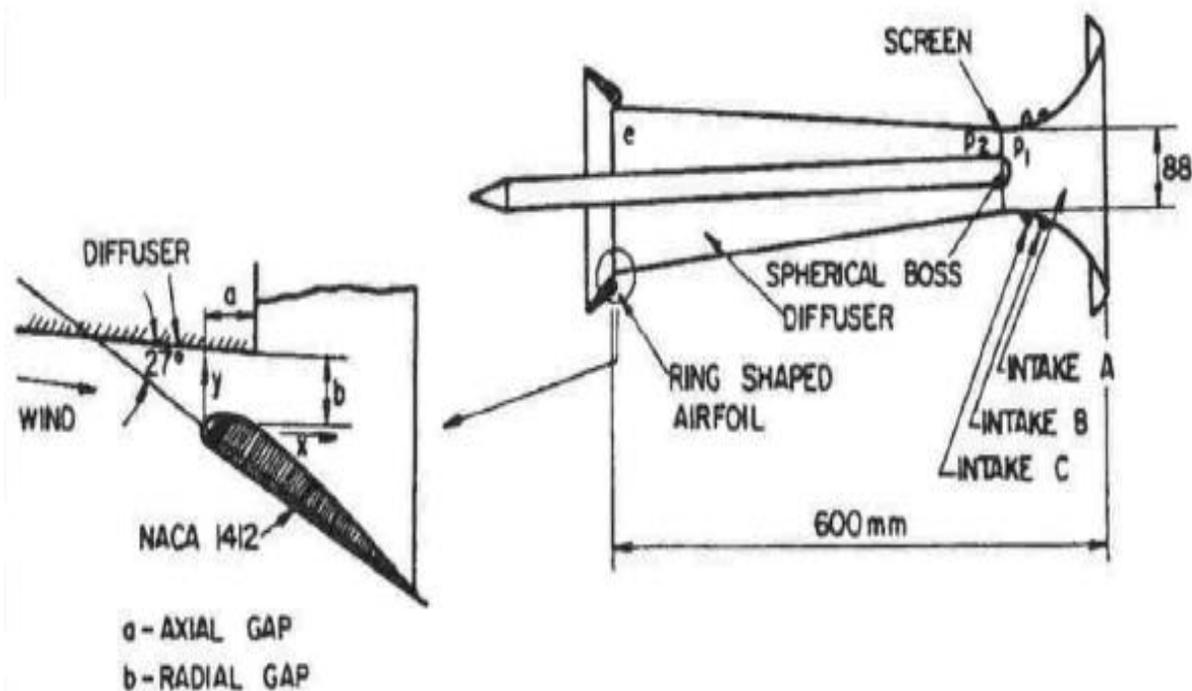


Figure 1.4: Layout of the first phase shroud design of Kogan [Car14].

In 1974, Igra, a student of Kogan, continued on with the DAWT research and realized the importance of reducing the diffuser's length-to-diameter ratio, in order to reduce the cost-of-energy and put the DAWT in a competitive basis, compared to conventional wind turbines. Throughout his research, he explored the reduction of the diffuser's size, based on the work of Kogan et al. [Kog62, Kog63a, Kog63b], and then he proposed the second and third generation designs. According to Igra, the increase in mass flow and therefore power is a result of the pressure drop at the diffuser exit, essentially "sucking" air through the diffuser. By the end of the 1970s, Igra had constructed a prototype DAWT [Car14].

At about the same time, Grumman Aerospace [Oma77] in the United States joined the research effort due to an increase in alternative energy sources caused by the oil crisis in 1974. Foreman, of Grumman Aerospace, focused on controlling the boundary layer in the duct with the use of slots. More than 100 different models were tested in the wind tunnel. The models were initially, like in Igra's research, equipped with a screen in order to simulate the pressure drop over the rotor. In some of Foreman's later research it was determined that swirling flow in the rotor wake increased the energy in the boundary layer and therefore delayed separation in the diffuser due to momentum transfer. Foreman also analyzed the economic and design implications of DAWTs [Hoo09].

Based on the design criteria of a high sub-atmospheric pressure at the diffuser exit plane and a large pressure recovery within a diffuser, taking into account the smallest possible structural

cost, Foreman (along with other researchers at Grumman) reached to the two most promising design concepts for further study. The first one called Boundary Layer Controlled Diffuser, employs the injection of the external air flow for the boundary layer control. As presented in Figure 1.5, the external high energy air, freely available without power penalty from the wind, is injected tangent to the wall, resulting in the increase of axial momentum of the boundary layer within the diffuser's wall. The additional momentum enhances the boundary layer fluid flow against the severe adverse pressure gradient and frictional losses identified along the wall region of large angle diffusers. Eventually, boundary layer control can prevent the flow from separating from the wall, which is the primary cause of flow's failure in large angle diffusers.

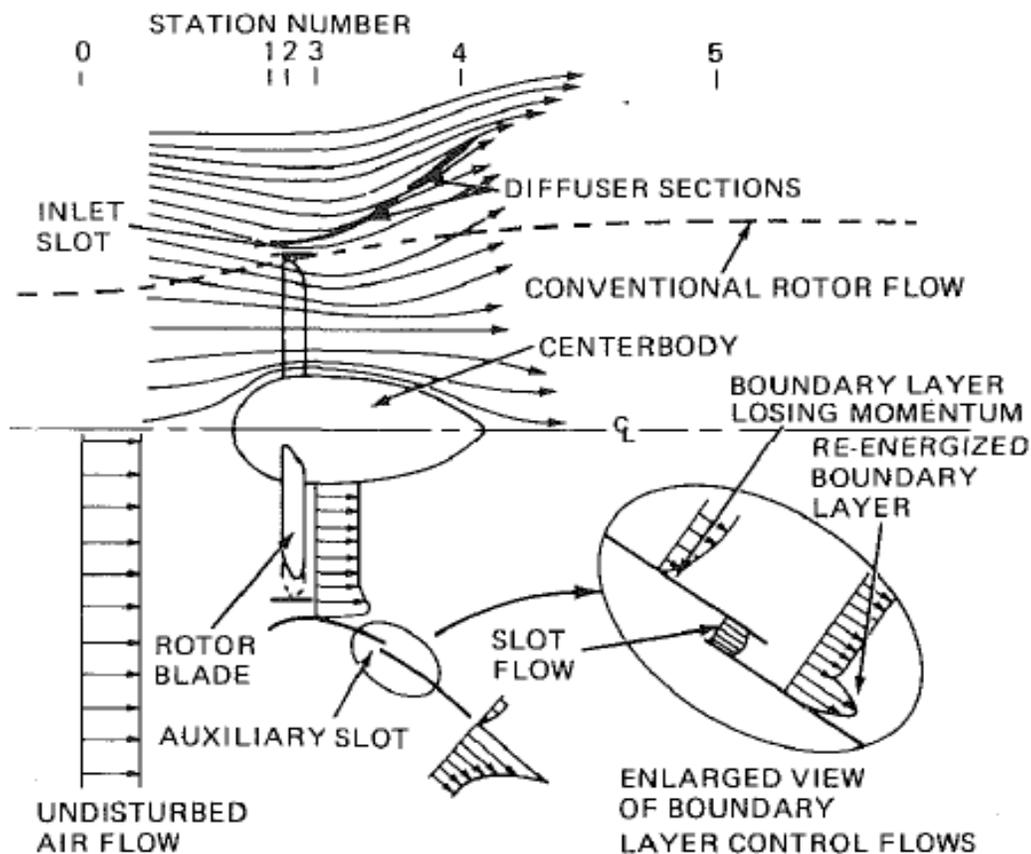


Figure 1.5: Effect of boundary layer control slots, in preventing separation within a DAWT [Gil83].

The second design concept, called Ring Wing Diffuser, was based on the fact that airfoil contours exhibit a generally low pressure field along the upper surface and a high pressure field along the lower contour. In a ring diffuser formation, the inner surface corresponds to the low pressure side. The airfoil section lift, acts towards the centerline and creates a reaction force on the internal flow that tends to direct it radially outward as well as axially downstream. The induced circulation around the lifting airfoil section results in an increased flow rate through the upstream end of the diffuser, in contrast to to a simply cylindrical duct of the same cross

sectional area. As a result, a turbine that is positioned within the ring wing diffuser has an augmented mass flow passing through it and therefore can extract more power than a bare turbine [Oma77]. Overall, Grumman's theoretical work identified the importance of large exit-area-ratios, high diffuser efficiency, peak velocity speed-up, and a strongly negative exit pressure coefficient. However, it was also identified that structural cost increases rapidly along with exit-area-ratio, thus a more tempered ratio would be more cost-effective [Phi03].

By 1976 an English researcher named Lewis joined Igra and Foreman in DAWT research and conducted a number of theoretical and experimental analyses. Lewis concluded through a momentum analysis that a shrouded wind turbine could extract 42.86 % more power compared to a conventional bare turbine. Similarly to Igra and his predecessors, Lewis addressed the importance of trying to reduce the size and cost of the diffuser. As a result of the promising findings of Lewis, Foreman and Igra, a renewed interest in DAWTs was initiated and many theoretical and experimental investigations regarding the DAWT concept were conducted during the 1970s and 1980s [Hoo09, Car14].

1.2.2.2 Final generation of DAWT designs

Having identified that the augmentation achieved by a DAWT, in contrast to a conventional bare wind turbine, is a direct consequence of the high subatmospheric pressure at the diffuser exit, both Igra and Grumman entered to the final stage of their investigations. Igra's diffusers, although more compact in contrast to those investigated in the beginning, remained significantly larger than Grumman's, resulting in a slightly higher augmentation. Thus, Igra [Igr76, Igr81] tried to find ways to reduce the diffuser size and overcome the resulting inherent separation. Consequently, Igra launched a research in order to demonstrate ways for the reduction of the diffuser size through boundary layer control (BLC).

The dominant contributor to the shroud length is the structure downstream of the diffuser. This component has an increasing cross-section as one progresses downstream; however, a fast rate of area divergence will cause flow separation and as a result a significant reduction in output power. Igra's aim was to demonstrate ways to tackle this shortcoming, either by a proper diversion and introduction of the shroud's external flow into the diffuser's inner boundary layer, or alternatively by simply using a ring-flap [Lel16].

During this investigation, Igra examined three BLC shroud configurations, denoted as Model 1, 2 and 3. As shown in Figure 1.6, the particular BLC models employed a single inlet section for the first third of the shroud, while the remaining two thirds were adapted to produce three different shrouds with exit-area-ratios of 2.4, 3.5 and 4.5 respectively [Igr81]. The length-to-diameter ratio was equal to 2.27 for all three models. The blowing or sucking procedure was achieved through the use of three holes drilled into the front section of the shroud. Hole 1 was located closest to the leading edge whereas holes 2 and 3 were placed further downstream on the shroud's outer surface. These holes could be linked either individually or simultaneously to all discharge holes A, B, C and D, located on the inner surface of the diffuser, downstream of the screen. In each case there were 24 holes of 3 mm diameter equally spaced around the respective circumference, except from hole 1, with diameter equal to 1 mm [Phi03].

According to Igra's results, an increase in diffuser efficiency and augmentation can be achieved for a given model geometry, by having a proper treatment of the boundary layer flow. In comparison with Igra's two models with approximately equal exit-area-ratios, the BLC DAWT showed an increase in augmentation between 3% and 20%. Furthermore, an increase in diffuser

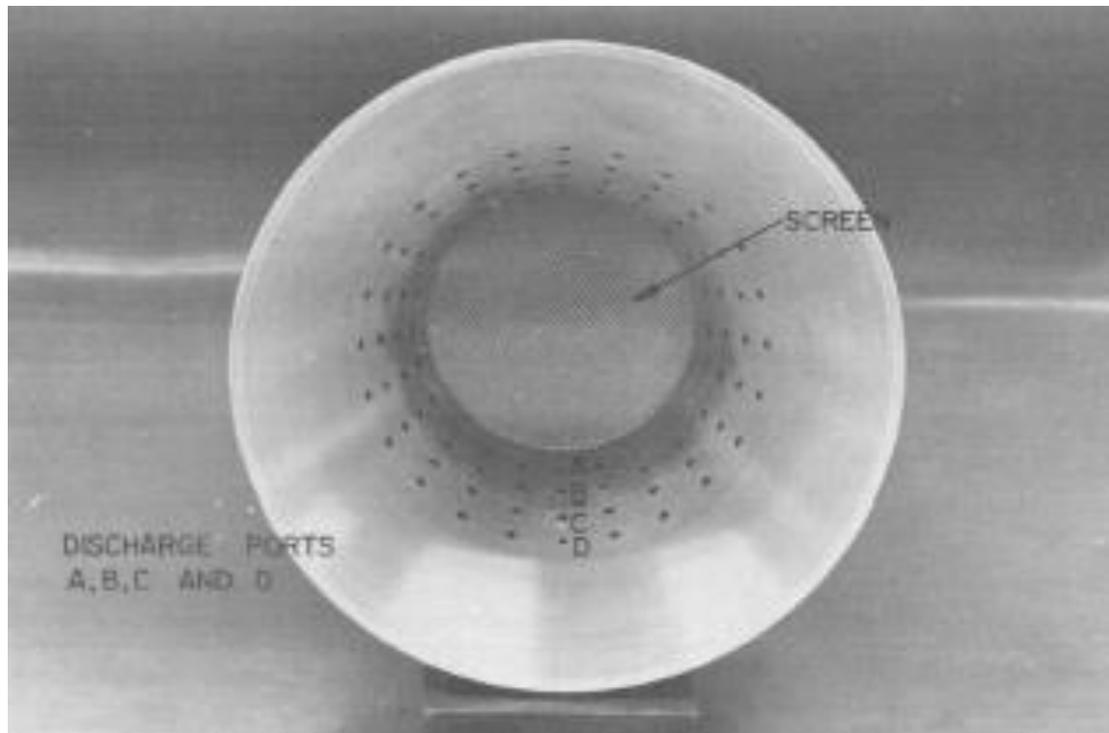


Figure 1.7: Igra's Model 3 rear view [Igr77].

On the other hand, Grumman identified through structural analysis and inviscid flow modelling that further performance gains were possible by increasing the diffuser included angle and incorporating an exit flap, or a dump flange [Phi03]. Specifically, a new diffuser model, based on the Baseline one, was explored and tested in the Virginia Polytechnic Institute open jet wind tunnel for a great variety of blade angles, tip-speed ratios and yaw angles up to 40° . The featured model, called DAWT 45, employed an exit flange in order to provide additional power output for a fixed diffuser length. The examined model (DAWT 45), was characterized by a conical included angle of 90° , a length-to-diameter ratio of 0.488 and exit-area-ratios of 2.62, 2.78 and 3.26, which were obtained by a range of the exit flange width from 1.5% of the diffuser's diameter, to 3% and 7.5% respectively [For83, Lel16].

The peak augmentation measured for the DAWT 45, while operating at 0° yaw angle and having blades with pitch angle of 28° , was equal to 3.25. Furthermore, when the wind turbine was operating at the optimum point, under a disk loading coefficient of 1.10, the generated velocity speed-up ratio of DAWT 45 was equal to 1.82. However, according to Grumman, the obtained results have to be corrected due to a 35% overestimation of the inlet velocity. The corrected results for the DAWT 45 showed a peak available augmentation of 2.19 and an averaged peak of 2.02, while the corresponding values for the Baseline DAWT were calculated to be 1.99 and 1.85 respectively [Lel16]. As a result, the DAWT 45 achieved around a 10% increase in both the average augmentation as well as in the peak values obtained. The cause of this improvement, could be due to the 25% more negative exit pressure. Concerning the effect of the different exit flanges on DAWT 45 performance, Grumman noticed there was a slight augmentation advantage in using the 1.5% exit flap, although if structurally required, the 7.5% exit flap would only impose a slight performance penalty. [For83].

The final observation in this phase of work refers to the performance of the DAWT under yaw. Grumman wind tunnel tested the DAWT 45, Baseline DAWT, along with rotor operating as a conventional bare turbine in 10° yaw increments up to a maximum yaw angle of 40° . The available augmentations for the rotor operating as a conventional bare turbine not only demonstrated an insensitivity to yaw but also an increase in power up to 80% at a yaw angle of 30° . Grumman concluded that the long upstream centrebody channeled the onset flow towards the upwind blade while at the same time shielding the downwind blade, the net result being the observed insensitivity to yaw. Both the DAWT 45 and Baseline geometries showed a negligible or small change in power over the same 30° yaw range while a dramatic reduction in power was found once a yaw angle of 40° was reached. However, it was not possible to resolve whether the yaw insensitivity was attributable to the centrebody effect or DAWTs themselves, therefore Grumman concluded that these yaw tests were inconclusive [For83, Phi03].

1.2.2.3 Further DAWT investigations

Despite the fact that all the results of Igra and Grumman Aerospace seemed quite promising, no attempts had been undertaken to turn the investigated DAWT concepts into commercial competitive products. In 1995 Vortec Energy Limited, the first private company which tried to commercialize the DAWT concept bought the rights to the DAWT design from Grumman Aerospace. The 17 meter prototype Vortec 7 was built in 1997. The prototype DAWT was based on Foreman's DAWT configuration and had a blade diameter of 7.3 meters with an expected power output of 1 MW. However, during the initial testing of the Vortec 7 it became apparent that the power generated did not match the predicted data made in Grumman Aerospace. In order to address this problem the DAWT design was optimized by post-graduate students of the University of Auckland with the use of CFD and small scale experiments to test out the possible gains from a variety of alterations and to investigate a number of new diffuser designs [Car14].

The velocity profile at the rotor plane showed lower speeds closer to the hub, whereas a high speed region was found beyond the blade tips. As a result, a lower power output than predicted was produced, as the initial assumption of a uniform velocity profile along the blade plane was inadequate to model the problem with precision[Hoo09, Car14]. The performance of the Vortec 7 seemed to be affected by separation caused due to some of the diffuser's geometric elements; thus, a development program was undertaken in order to explore modifications concerning the inlet of the Vortec 7, the secondary boundary-layer control slot and the centrebody geometry. The outcome of this program was the reduction of separation and the increase of Vortec 7 augmentation. The above modifications, included an elliptical nosecone, enclosure of the inlet structural ring with an asymmetrically profiled bull-nose, extension of the secondary diffuser upstream with an elliptical bull-nose, as depicted in Figure 1.8. Unfortunately, the data of the modified DAWT resulted in only a 39% increase compared to the initial augmentation, which was still far away the predicted augmentation. Due to this miscalculation, Vortec Energy Limited was led to a non-definitive closure. Nevertheless, due to the experimental and computational study performed by Phillips, a deep understanding concerning the governing principles of DAWTs had been achieved. The modified Vortec 7 DAWT is shown in Figure 1.9 [Phi03].

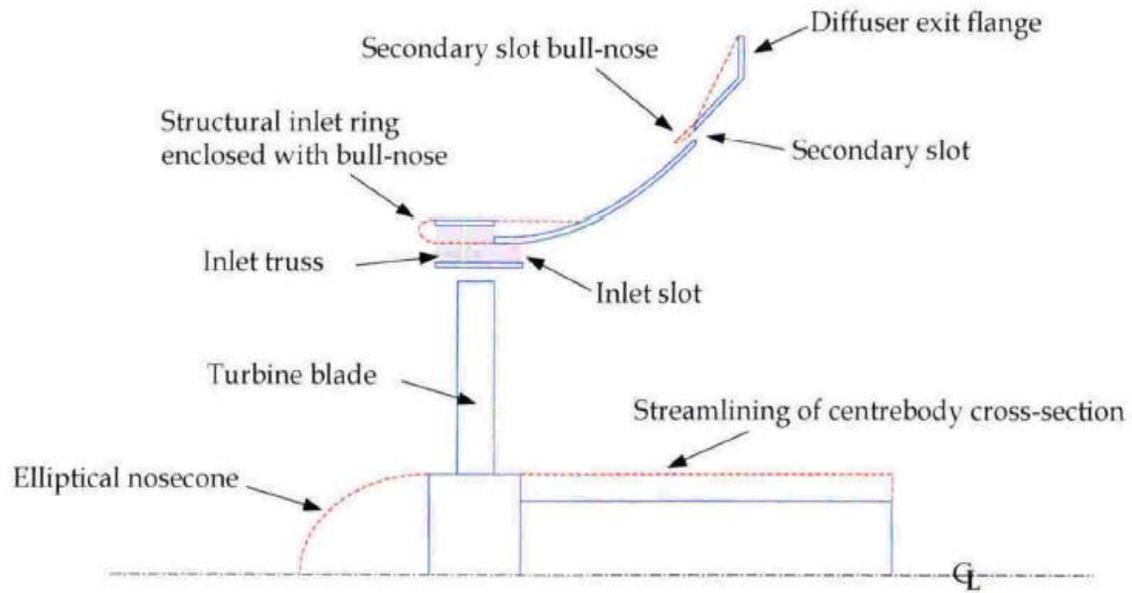


Figure 1.8: Schematic diagram of the modifications made to the Vortec 7 geometry [Phi03].



Figure 1.9: The full-scale modified Vortec 7 [Phi03].

Despite the failure of Vortec Energy, this did not discourage the DAWT research. Around the same time as Vortec Energy, van Bussel [Bus98a, Bus98b, Bus07] readdressed the DAWT theories developed so far, and concluded that the optimal operation conditions of a DAWT are the same as for a bare wind turbine. He also concluded that the pressure drop across the rotor is independent of the diffuser's exit-area-ratio and found that the thrust of the rotor and shroud are dependent on each other; however, the thrust on the rotor, unlike the diffuser, is not proportional to the mass flow. Finally, van Bussel noted that rotor power coefficients equal to 2.5 could be obtained by reducing the diffuser exit pressure. Additionally, it can be concluded from van Bussel's work that the augmentation expectations of Vortec were unrealistic [Hoo09, Le16].

After van Bussel's work, an increasing number of publications released and several attempts were made to commercialize the idea. Specifically, Bet and Grassmann [Bet03], throughout a computational study, developed a shrouded wind turbine characterized by a wing-profile ring structure, which was estimated to achieve an increase in power output by a factor of 2. Additionally, Grassmann et al. [Gra03], performed a preliminary experimental work, based on the DAWT configuration developed by Bet and Grassmann [Bet03]. The outcome revealed an increase of power output by a factor of 1.55 and 2 for high and low wind speeds respectively, confirming previous fluid dynamic simulations [Bet03].

Wang et al. [Wan08] investigated the possibility of improving wind energy capture, under low wind speed conditions, in a built-up area along with the design of a small wind generator for domestic use in such areas. During their research experimental tests and CFD computations were carried out in order to evaluate the effect of a convergent-divergent scoop on the power output of a small wind turbine. The obtained results revealed that the scoop increases the airflow speed resulting in a power output 2.2 times higher compared to a conventional wind turbine. Finally, Wang et al. [Wan08] concluded that the error between the computational and wind tunnel results was within 5%, and therefore CFD methods can be used as an effective design tool for shrouded wind turbines. The convergent-divergent scoop is shown in Figure 1.10 [Le16].

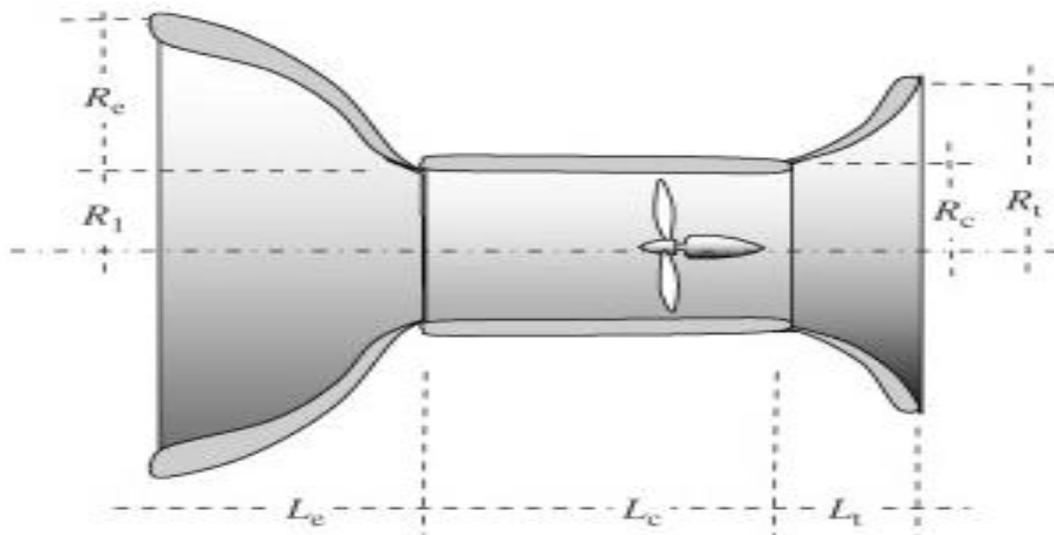


Figure 1.10: Illustration of the scoop design proposed by Wang et al. [Wan08].

Ohya & Karasudani [Ohy10] reported a remarkable increase in the output power of approximately 4-5 times relative to the Betz limit. This is succeeded by the employment of different configurations of a shroud concept called “the brimmed diffuser”, which is depicted in Figure 1.11. After wind tunnel and field experiments, a significant increase in power output was found, which was attributed to the low-pressure region that generates a zone of strong vortex formation behind the broad brim, which in turn draws more airflow to the wind turbine inside the diffuser.

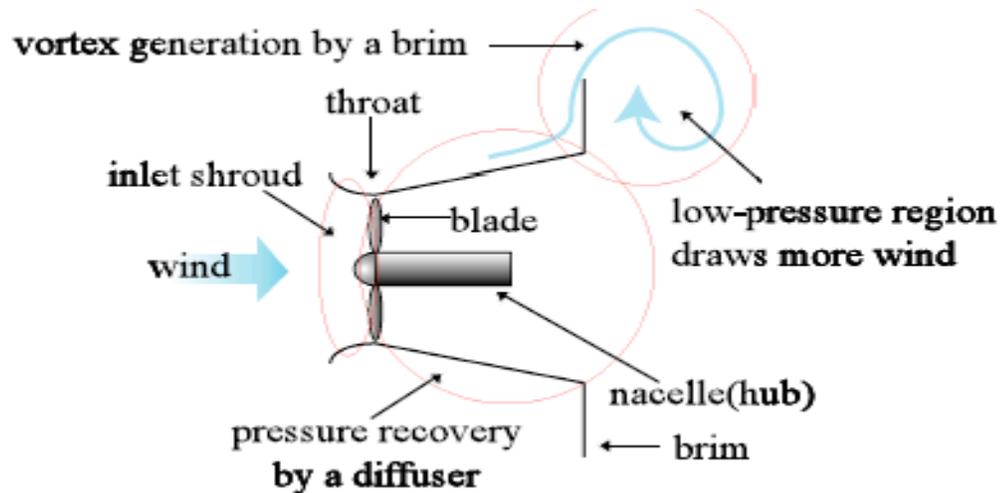


Figure 1.11: Flow around a brimmed diffuser [Ohy10].



Figure 1.12: 500W wind-lens turbine [Ohy10].

In the work of Toshimitsu et al. [Tos08], flow velocity measurements with a flanged diffuser were performed by utilizing Particle-Image-Velocimetry visualizations. The obtained results showed that the rotating effects of the turbine blades suppress the turbulence and the flow separation near the inner diffuser surface. Additionally, at downstream some vortices were consistently found; specifically the one behind the flange produces a suction effect on wind to the diffuser increasing the inlet flow velocity. As a result the diffuser device enhances the wind power in 2.6 times relative to standard wind turbine. Furthermore, Aranake et al. [Ara15] performed computational analysis of shrouded wind turbine configurations by using a 3D RANS solver, in order to investigate airfoil sections for shrouds of shrouded wind turbines and estimate their performance. Among a variety of airfoil shapes considered for shroud cross sectional profiles, the Selig S1223 attained the greatest amplification of mass flow. The following 3D analysis of full shrouded turbine systems identified the benefit of increased mass flow through the plane of the turbine which could lead to a power extraction up to 90% beyond the Betz limit. According to Aranake et al. [Ara15], the improvement in power extraction beyond the bare turbine was substantial; the NACA0006 shroud improved power over the bare turbine by a factor of 1.93 and the S1223 improved it by a factor of 3.39 at freestream velocity equal to 5 m/s. These results enhanced the utility of the shrouded wind turbine as a device that can be used effectively at low cut-in speeds [Lel16].

Abe & Ohya [Abe04], Ohya & Karasudani [Ohy10] performed an extensive experimental and numerical work in order to develop a high performance flanged diffuser applied on small wind turbines. The pressure in the wake downstream of the diffuser, while having a flange around the trailing edge of the diffuser, causes the separation of the flow and creates a large low pressure region downstream of the diffuser, which creates a suction effect through the diffuser. On the other hand, Takahashi et al. [Tak12] worked on development of Wind-Lens turbine. The induced vortex formed, probably by blade tip vortex within the boundary layer of the inner surface of the diffuser, causes the suppression of the flow separation from the inner diffuser surface, resulting in the augmentation of the wind's acceleration. Finally, Jafari & Kosasih [Jaf14] reported that flow separation in the diffuser may lead to a reduction of the overall power coefficient. However, this phenomenon can be mitigated by adapting the length of the diffuser [Lel16].

The last decade a renewed interest in the DAWT configuration reappeared in the United States of America. Specifically, Werle and Pretz proposed a momentum based formulation, in which a duct thrust coefficient is defined in order to control the behavior of a DAWT. Werle and Pretz are also affiliated to the Massachusetts-based company Flodesign, who claimed in 2008 to have developed a shrouded wind turbine generating 3 to 4 times more power than a conventional wind turbine (Figure 1.13). The design is different in the sense that they employ a lobed slotted diffuser, which should enhance mixing and thus the power output [Hoo09, Lel16]. Recently, DAWT research has been done by Ten Hoopen and van Dorst of Delft University of Technology and Widnall of the Massachusetts Institute of Technology, using axisymmetric vorticity method in order to calculate the impact of the duct and rotor on the flow field. Ten Hoopen focused on placing vortex generators at the trailing edge of the duct to enhance the wake mixing behind the rotor. On the other hand, Van Dorst improved the blades of an existing DAWT [Hoo09, Car14].

In conclusion, although during the last decades the DAWT configuration has evolved considerably, no big commercial success has been achieved so far by any company. Furthermore, no commercial DAWT design tools have been developed so far and it seems that the scientific

community still needs to agree on a integrated DAWT theory [Hoo09]. However, a worth-mentioning exception is DONQI, which started developing a small scale urban DAWT in 2008. The wind energy department of Delft University of Technology also succored in developing the turbine by allowing students to graduate on the DONQI URBAN WINDMILL (Figure 1.14). Until today, DONQI has sold turbines in various countries and is on the verge of up-scaling their production to be able to meet the current changing demands [Dor11].



Figure 1.13: DAWT concept by Flodesign [Flod].



Figure 1.14: DONQI Urban Windmill [Dor11].

2. Methodology

In this chapter, a thorough description of the developed algorithm is undertaken, while some of the resulted designs are presented, to demonstrate its features and capabilities.

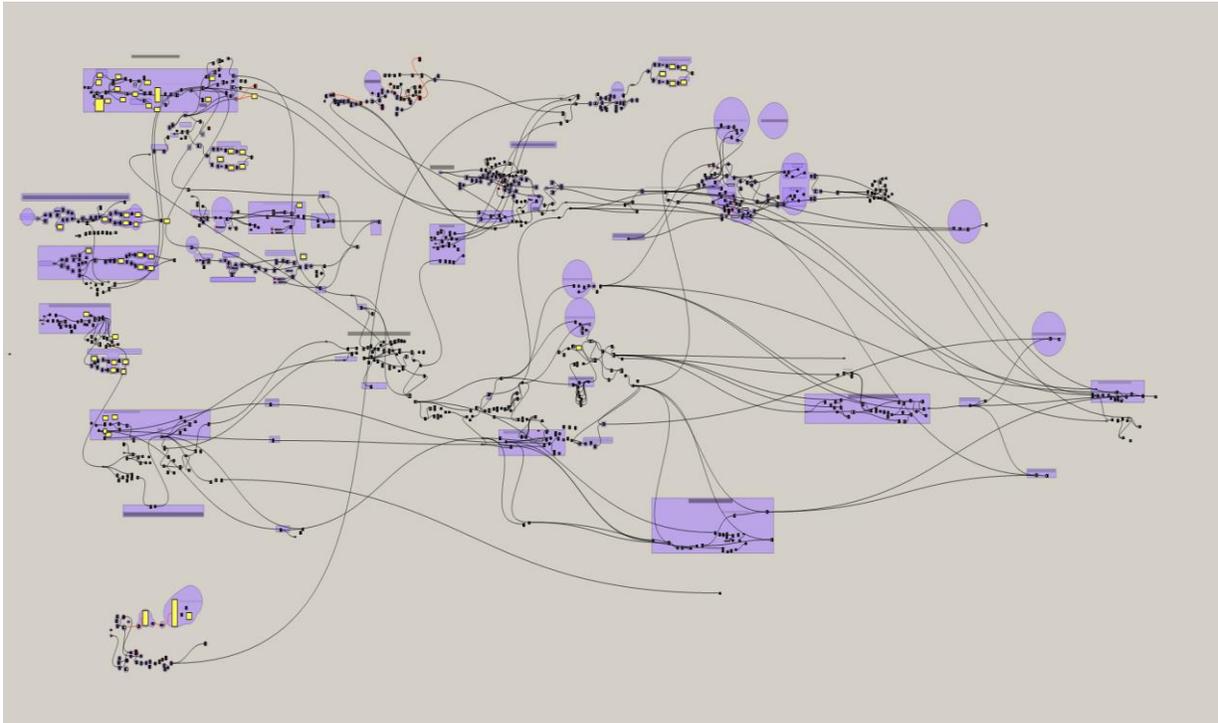


Figure 2.1: An overview of the developed graphical algorithm within the Grasshopper® environment.

2.1 Introduction to Grasshopper®

This subsection presents a brief description of the utilized software and an introduction to the design features for the parametric construction of the DAWT, so as to familiarize the user and the reader with the GUI and some of the software's features. The algorithm was developed entirely in Grasshopper®, which is a graphical algorithm editor, tightly integrated with Rhino's 3D modeling tools. However, this embodiment allows for the interaction with Rhino software without using any tool from Rhino's toolbar [Gra3D], while the utilization of Grasshopper® requires no knowledge of programming or scripting, a characteristic that makes it more user friendly.

Grasshopper® has its own main window and drawing tools, which are separate from those of Rhino®, while maintaining the same capabilities as those of Rhino®. All of its tools are in a form of small boxes, the so called *components*, which may have one or multiple input and output ports. Each component performs a specific function, while the connection and cooperation of a number of elements is succeeded by the use of connecting wires, as illustrated in Figure 2.2. Furthermore, each tool can be selected from the main window and then added to the "drawing" canvas (workspace) for the formation of a graphical algorithm. As mentioned above,

Grasshopper[®] is directly connected to the main window and graphical interface of Rhinoceros[®], allowing the user to create the algorithm in Grasshopper and obtain instantly the geometrical result in the workspace of Rhino [Gras3D].

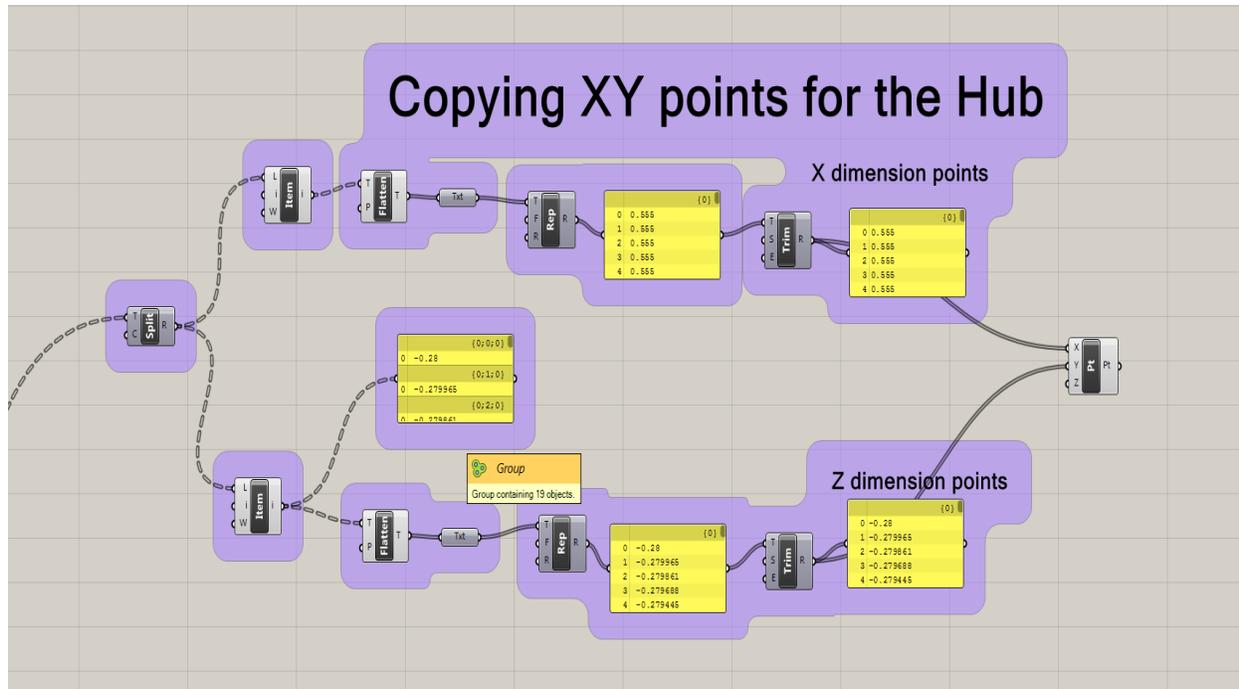


Figure 2.2: Grasshopper[®] components and wires [Cha14].

Another characteristic of Grasshopper[®] is the ability to directly import data through text files. The algorithm generated in this work, imports all the necessary data through a number of different input files in text format (*text files*), and each one of them contains information for a particular algorithm modeling stage. For example, the first input file (*curves points file*) used in this work contains all the points of the different 2D airfoils forming the blades in a dimensionless form. After completing with all of the necessary data input, the user can initiate the operation of the algorithm by simply picking the specific files through Grasshopper[®]. Then, all the data files are automatically stored within appropriate data structures, called *Data Trees*. Grasshopper[®] uses lists to place the imported data, where each data list is configured in a tree structure [Tree1].

However, the data management of Grasshopper[®] may cause access-related complications. Therefore, there are several additional tools, which support the processing and selection of the tree data content, and how to configure the structure of a data tree. A good knowledge of the functions and techniques of the specific tools is essential to develop a Grasshopper[®] algorithm in a simpler way, especially when the management of a large amount of data is required, as in this case [Tree1].

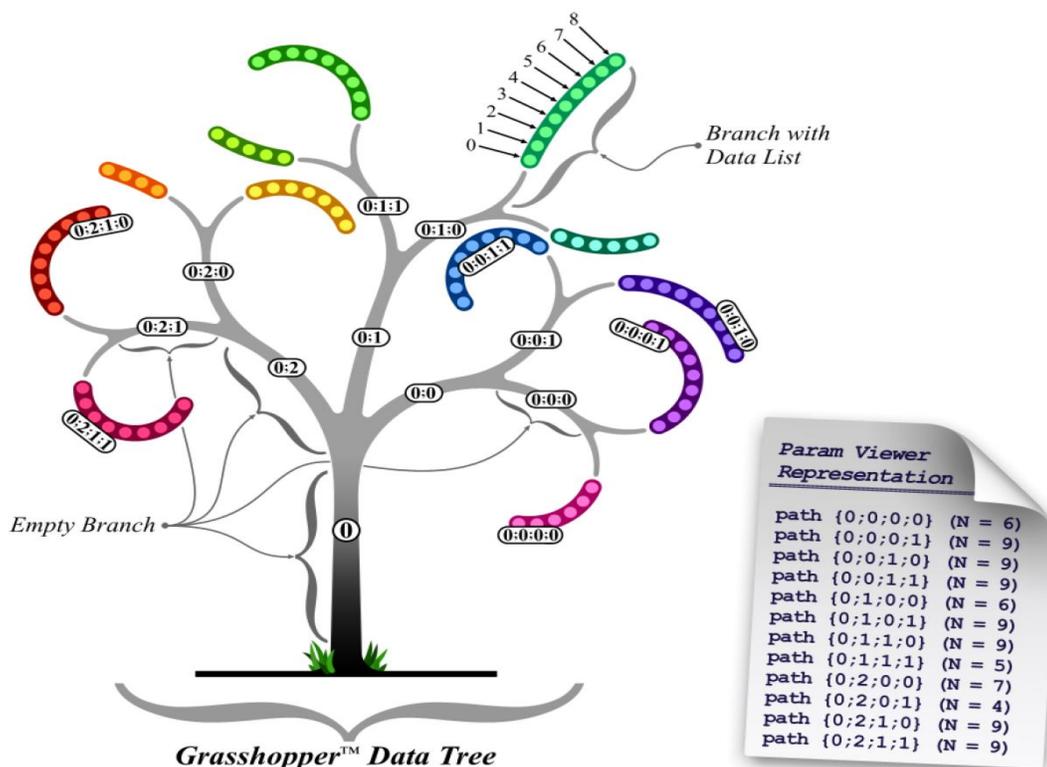


Figure 2.3 : Illustration of Grasshopper® data tree structure [Tree2].

Another important feature in both Grasshopper® and Rhinoceros® is the definition of the model tolerance. Tolerance is generally defined as the precision required to make a mechanical drawing, or vice versa, the error rate that can be accepted for the final model by the designer. The required precision depends on the nature of the object to be designed and is proportional to its complexity and size. In Rhino the absolute tolerance represents the maximum permissible distance between two objects in order to be considered as a one part or solid, i.e. to allow their union. As mentioned in the beginning of the chapter, Grasshopper® is directly connected to Rhinoceros®, for this reason, the absolute tolerance is defined through the main window of Rhino. There are three tolerance settings available, absolute, relative and angular tolerance, which can have a direct effect in some of Grasshopper's basic drawing tools. For example, commands using intersecting curves or surfaces are very sensitive and their outcome depends greatly on the respective values of tolerance, established by the user [Tree1].

Additionally, further commands such as the cutting/ removal of a surface or volume (Boolean Operations) are also very sensitive and directly dependent on the established tolerance; very low value of tolerance for example, can cause a failure of the model. To achieve the best possible result for the purposes of this work, the absolute tolerance was set equal to 0.001 m , while the relative and angular tolerances were set equal to 0.1 m and 0.001 m respectively. These values resulted from a trial and error procedure, to achieve a geometrically acceptable final product. During the construction of the algorithm, it was typical that a number of design tools, such as those mentioned above, to result in bad quality surfaces and volumes or even to lead Grasshopper® to crash. It was found later, that the cause of their problematic operation was due

to the incorrect definition of tolerance. This highlights the importance of tolerance for the final result as well as for the smooth operation of the software [Tol].

2.2 Introduction to the algorithm

In the following subsections, the methodology for the parametric design of a complete DAWT system, including the turbine blades, the centrebody, the diffuser geometry, as well as the internal flap, the tower and the central column structure is presented.

2.2.1 Blades

This subsection presents the methodology for the parametric definition of the turbine blades. The specific procedure was developed within the author's Diploma Thesis; thus a detailed description can be found in [Char14].

2.2.1.1 External surface

A wind turbine blade is shaped by several airfoil sections, which eventually form the 3D outer surface of the blade, through the procedure of lofting; the 2D airfoil profiles are defined by a direct interpolation of a set of points. For the definition of the blade's external surface the user needs to create and import two distinct text files. The first one (airfoils_points.txt), consists of the airfoils' data as a sequence of discrete points, whilst the second input file (airfoils_parameters.txt), includes all the necessary design parameters for the definition of the blade geometry, such as the chord length and the twist angle for each airfoil section. The degree of the interpolation curves, describing the corresponding cross-sections (and consequently the degree of the blade surface), is defined by the user. However it should be always higher than 2, so a continuity of the 1st derivative to be fulfilled, which is a very important feature for the later construction of the 3D solid blade [Cha14].

The coordinates of each airfoil are provided in a dimensionless form (as a function of the section's chord) and are converted to a dimensional form by multiplying with the corresponding chord length at the specific stacking position of the airfoil, along the blade's length. The construction of the blade using planar airfoils is performed with respect to a stacking line; this is usually a straight line, however the algorithm provides the ability to define a curve to be used as a stacking line, in order to produce pre-bend blades. All the cross-sections are distributed in perpendicular planes along the stacking line and they are stacked with respect to their centers of gravity. Additionally, the twist angle of the blade at each radial position can be defined by applying the desirable twist angle in the corresponding airfoil section. The twist angle for each airfoil section is defined through the second input file, along with the stacking positions and the chord length. Consequently, by applying the appropriate translation and rotation matrices to each cross-section, the section curves are accordingly positioned in the 3D space.

The second input file also includes two extra features, which are very useful when the user wants to import a large number of different types of airfoils in the algorithm. In wind turbine blades design, and particularly in large wind turbines design, it is common to use more than once the same airfoil type along the blade, which either may maintain all of its design parameters unchanged or may have modified values for some of its design parameters (for example the twist angle and the chord length). Thus, the first additional feature is the number of "copies"

(duplicates) of a particular type of airfoil along the length of the blade. The second added feature is the curve stacking order, which is complementary to the first one, allowing the user to designate the order in which airfoils are stacked along the stacking curve. In this way, the preparation of the airfoils' parameters file is facilitated. For example, in cases where the same type of airfoil is frequently used at non-consecutive positions, the designer does not need to import the design parameters of each airfoil according to the stacking order used for the airfoils.

The two input files mentioned above constitute the database for the construction of the blade's outer surface, which is essential for the subsequent definition of the internal structure. Once the airfoils are properly positioned, the next step is the creation of the outer surface of the blade through the provided cross-sections. The presented methodology creates the blade's surface through the procedure of lofting. In order to succeed this, the imported curves have to be properly constructed, so that the generated surface is as "clean" as possible. By the term "clean" we mean a high quality surface without discontinuities and failures. A sample design of a 3D blade surface is presented in Figure 2.4 along with the blade's cross sections, which were used to define its external surface [Cha14].

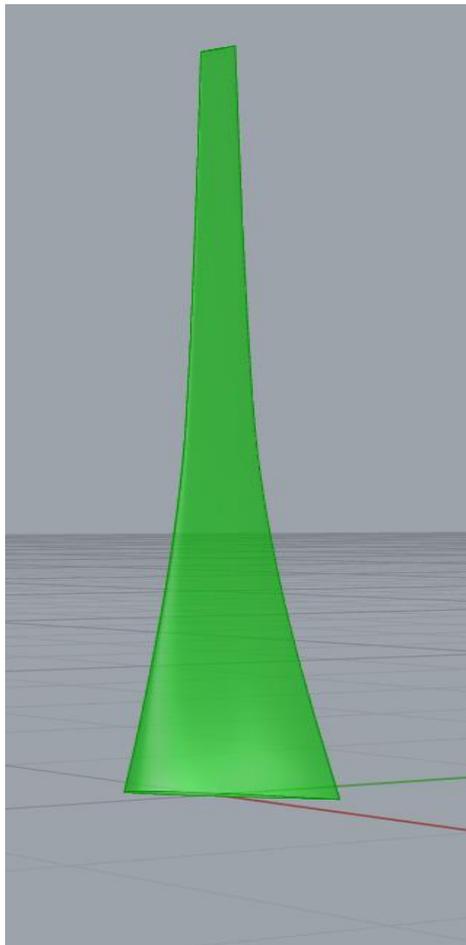


Figure 2.4: The external surface of a small-scale wind turbine blade.

2.2.1.2 The Internal Structure

During the last years, the formation and construction of wind turbine blades' internal structure has been greatly developed. Therefore, various ideas and methodologies have been proposed for the configuration of the blade's internal structure and for the appropriate selection of materials, allowing the blades to maintain high aerodynamic efficiency, along with the required structural integrity. The structural definition approach, used in this algorithm, is general and representative of the current commercial blade designs trends [Cha14]. Modern wind turbine blades are characterized by variable thickness along their span. Assigning different thickness is perhaps the most difficult and complex step in the construction of the blade, because of the sensitivity that some Grasshopper's design tools show, concerning the produced outer surface.

As a result of this weakness, an alternative methodology was developed. In order to define the blade's internal structure, the outer surface of the blade needs to be configured properly. The configuration of the outer surface is accomplished by taking into account the data of a third input file (`internal_structure.txt`). The particular file is divided in two sections: the upper section, which includes all the data concerning the thickness of the inner surface, and the lower section, which includes all the necessary data for the creation of the shear webs [Cha14]. In order to assign different thicknesses along the blade, the number of regions in which the blade is to be split should be specified through the aforementioned third input file. Such regions are defined by selecting a pair of cutting planes for each one of them. The cutting planes are perpendicular to the stacking curve. Subsequently, the blade surface is sectioned by the specified cutting planes and each resulting region is manipulated separately, in order to assign different laminate thicknesses (Figure 2.5).

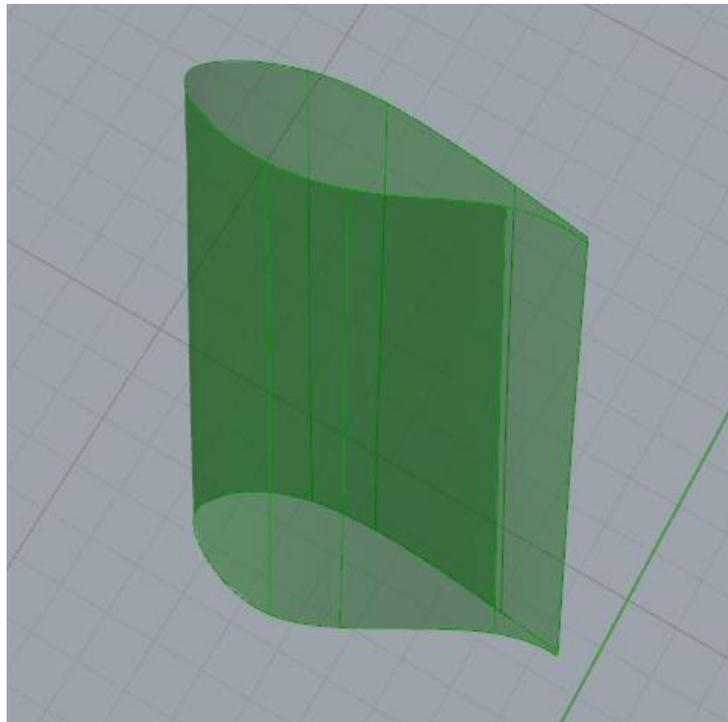


Figure 2.5: A separate blade region [Cha14].

The next step is the creation of surface patches for each blade region. Therefore, a set of curves are mapped on the 3D surface of the blade, creating 3D NURBS curves that split each region of the blade. These curves define separate surface patches and different thicknesses can be created for each blade region, as shown in Figure 2.6. Having completed the blade's surface configuration properly, the surface patches (faces) may obtain the corresponding values for their thicknesses. The thickness of the blade may vary along its span. Thus, the internal of the blade is formed with a thickness which varies linearly from the start to the end of each region. Furthermore, each patch of a blade region may obtain different value for its thickness.

The process described above could be achieved very easily and simply, with an "offset" procedure. However, all of the offsetting design tools in Grasshopper® were unable to displace the surface patches at a certain distance without deformation of the original surface. This inability, combined with the limited number of design tools for this task, led to the development of an alternative methodology; the methodology developed emerged after testing several other methodologies, which did not produce the desired results [Cha15].

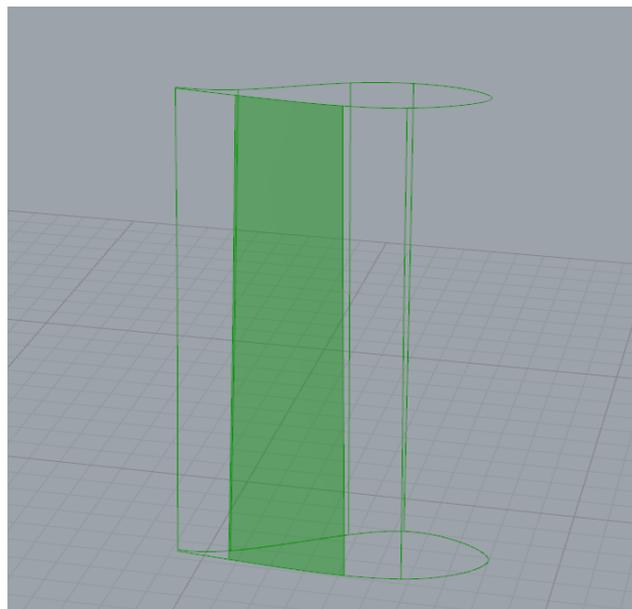


Figure 2.6: The division of a blade region in 7 surface patches.

At first, the wireframes of all surface patches of the blade are computed. The wireframe of each patch consists of four edges, two in the horizontal direction and two in the vertical direction. Afterwards, the vertical edges of each wireframe are divided by a predefined number and their division points are obtained. The points are appropriately sorted into groups, so that their transfer to the $u - v$ parametric space is feasible. In this way the points can be mapped on the outer surface of the blade. Subsequently, from each group, a single 3D curve is produced; each curve is fully fitted on the blade's outer surface. The curves are organized into groups so that every group corresponds to a surface patch. Finally, the curves are divided by a predefined number and their division points are obtained.

In order to achieve a uniform thickness distribution along each patch, a distribution of values is computed with a linear interpolation between the aforementioned limit values. Thus, for each

pair of values, the algorithm generates a series of decreasing values, while the thickness of the blade decreases uniformly as we approach the blade's tip. The displacement of these points determines the distance between the internal surface to be created and the outer surface of the blade and, therefore, the thickness of the blade. Subsequently, each group of the displaced points produces a single 3-dimensional curve. Note that, for the construction of solid tail at the ending of each surface region, the first patch on the pressure side and the first patch on the suction side of the blade region (starting from the trailing edge), are not used at this stage of the algorithm. With the creation of these curves, the construction of the internal surface of the blade is now feasible [Cha15].

The methodology presented above, is effective and fast, as the generated curves can be easily "transformed" into surfaces, which are qualitatively acceptable and do not create any dysfunctions in the algorithm. Furthermore, through this procedure, a smooth thickness distribution in the blade's interior is achieved. Nevertheless, reliability is the main advantage offered by this methodology; although is based on the produced external surface, the effectiveness of the methodology does not depend on the quality of the produced external surface of the blade. After completing the procedure described above, each surface patch is now characterized by a specific number of points that fit totally with the blade's outer surface. Eventually, the internal surface of the blade is produced through lofting the produced curves. The external and internal patch surfaces are then transformed into solid volumes. The "difference" (Boolean Operation) of these volumes (external and internal) produces the blade as a solid, consisted of solid patches, as shown in Figure 2.7. Having completed this stage, the blade is now characterized by uniformly varying thickness along its span [Cha15].

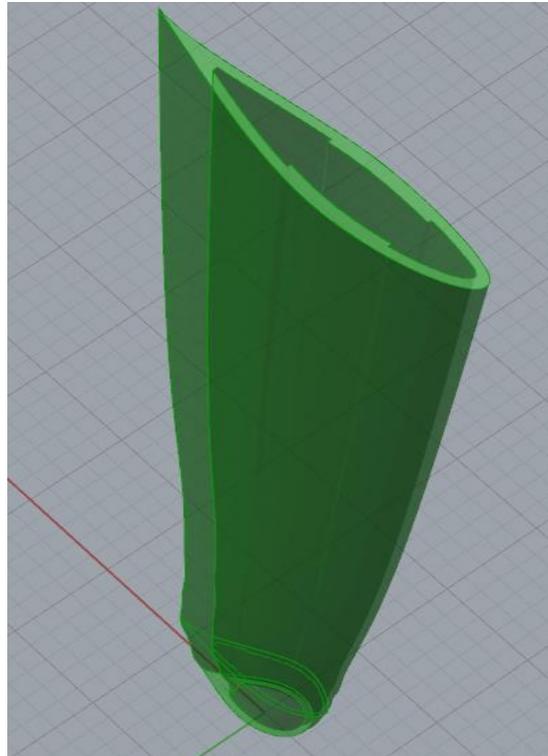


Figure 2.7: Resulting solid blade (Volume Difference).

The last step in order to complete the definition of the blades' internal surface is the creation of the shear webs. This algorithm computes the shear webs as sweep surfaces to which the desirable thickness may be applied to both sides. To create the shear webs, the data from the lower section of the third input file (`internal_structure.txt`) is used. The algorithm provides the ability to assign a uniformly varying thickness along the webs. For this purpose, each shear web is determined by a number of regions with different thicknesses; the number of regions is unlimited. For each region, a starting and an ending plane section are defined, in a similar way to blade regions. In addition, for each region, the user needs to define its position along sections' chord length for the two bounding planar sections (the lower and the upper one respectively); the user may also assign a twist to the shear webs. Eventually, the webs are fused into the blade and the blade obtains its final form, characterized as a single solid. A blade consisting of two shear webs is presented in Figure 2.8; the webs were defined by three regions with different thicknesses and twists [Cha15].

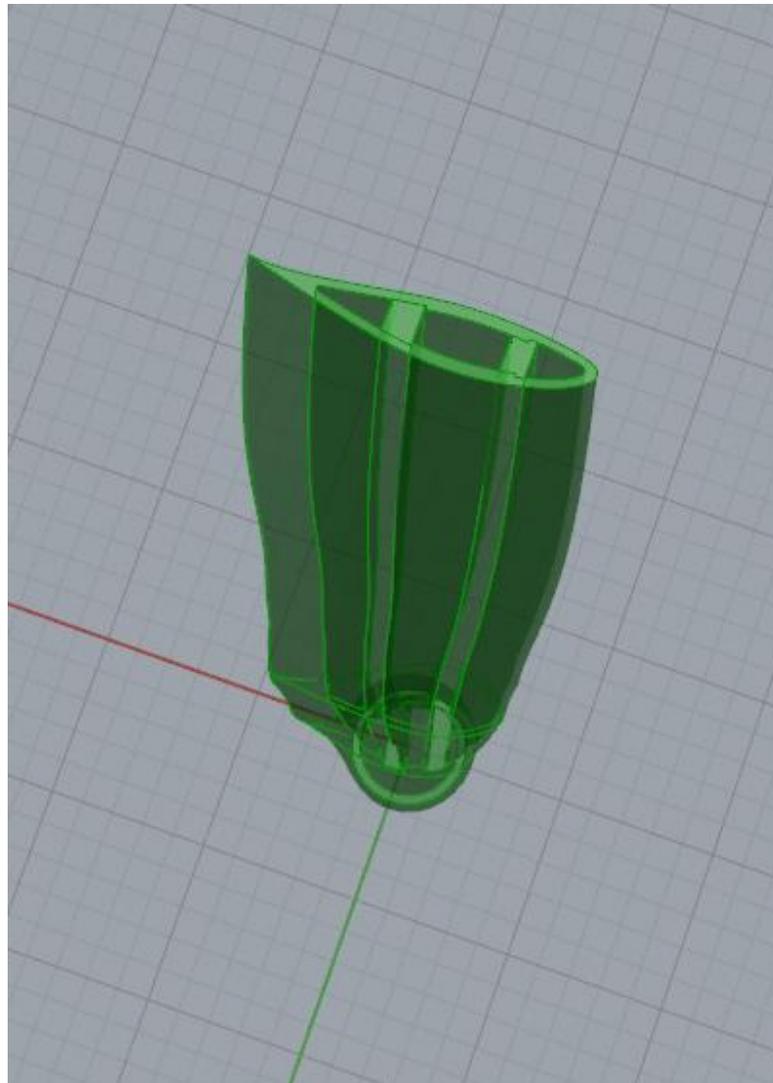


Figure 2.8: Example of a blade with two shear webs [Cha15].



Figure 2.9: Illustration of the final solid blade geometry [Cha15].

2.2.2 DAWT Design

Besides the turbine blades, a typical DAWT system consists of various other elements, such as the diffuser, the hub (centrebody), the internal flap (if any), the tower and the central column. After the completion of the blade construction, the algorithm is able to produce all the aforementioned DAWT elements in order to have an integrated DAWT as a single solid. The procedure is fully parametric and similar to the one used for the creation of the blades' external surface, while the user needs to create and import one text file for the definition of the elements' external surfaces.

2.2.2.1 The Diffuser

The diffuser geometry is defined through a profile curve which results in the 3-dimensional outer surface of the diffuser through the procedure of lofting. The profile curve can be prescribed by a direct interpolation of a set of points, imported through a text file. The particular file (*elements_points.txt*) consists of all the DAWT's elements data, including the diffuser, as a sequence of discrete points. The degree of the interpolation curves is user-defined, being always higher than 2, thus allowing for the continuity of the 1st derivative. This is very important for the construction of the 3D diffuser geometry [Cha17].

After the interpolation of the imported points the user may transform the profile curve through two additional functions offered by the algorithm. The first function allows the user to scale the diffuser's profile curve in order to create larger or smaller diffusers. Unlike the procedure followed for the creation of the profile curve, the scale procedure takes place through Grasshopper's canvas, where the user can modify directly the scale factor, making the whole procedure faster. In this way the user can choose a scale factor in Grasshopper® and get an instant result in Rhino workspace, enabling the construction of different diffusers. Finally, each curve is scaled according to its center of gravity, resulting in a uniformly scaled geometry, as the one shown in Figure 2.10.

The second additional function enables the user to translate the diffuser's profile curve. In this way the user can orient the profile curve in a different location and thus place the diffuser to the desired 3D position. As in the case of the scaling procedure, the user can choose a translation vector through Grasshopper® and instantly see the translated profile curve in Rhino's workspace. Furthermore, the translation procedure for the profile curve is made by taking into consideration its center of gravity, as in the case of the scaling feature (Figure 2.11). Both of the aforementioned features offer the user the capability to transform and translate the diffuser's profile curve in the 3D space, thus the creation of a variety of diffusers. At this point, it should be mentioned that the profile curve is automatically converted into meters, in case that the interpolation points have been imported with different units [Ansa].

Once the profile curve is properly positioned, the next step is the creation of the outer surface of the diffuser. The algorithm creates the diffuser's surface through the procedure of lofting. The procedure is similar to the one used for the creation of the blades' external surface. In order to succeed this, the diffuser's profile curve is rotated by 360 degrees around the x-axis (the axis of symmetry), so as a circular set of curves is created, as shown in Figure 2.12. This procedure is necessary in order to generate a smooth surface of revolution. As a result, the diffuser's external surface is generated without any discontinuities. As understood from above, the resulted diffuser is constructed in a fully parametric way. Eventually, the diffuser's outer surface can be transformed to a solid and added to Rhinoceros® through the *bake* option. A

number of 3D diffuser designs, characterized by different size and shape are depicted in Figure 2.13 and Figure 2.14.



Figure 2.10: Demonstration of the “Scale” operation.

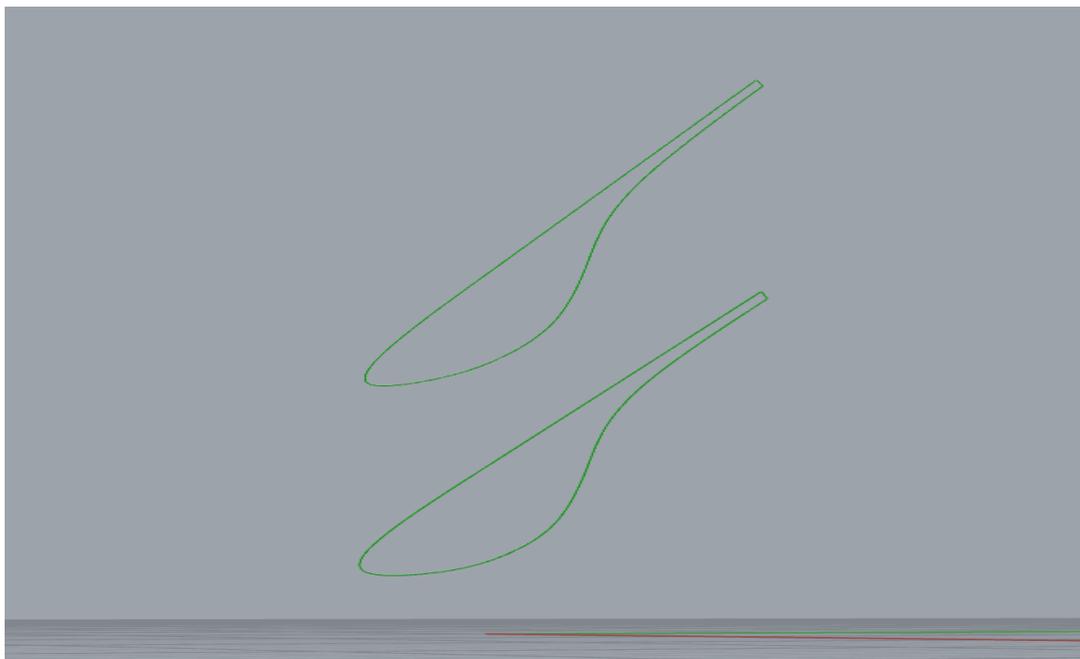


Figure 2.11: Demonstration of the “Translate” operation.

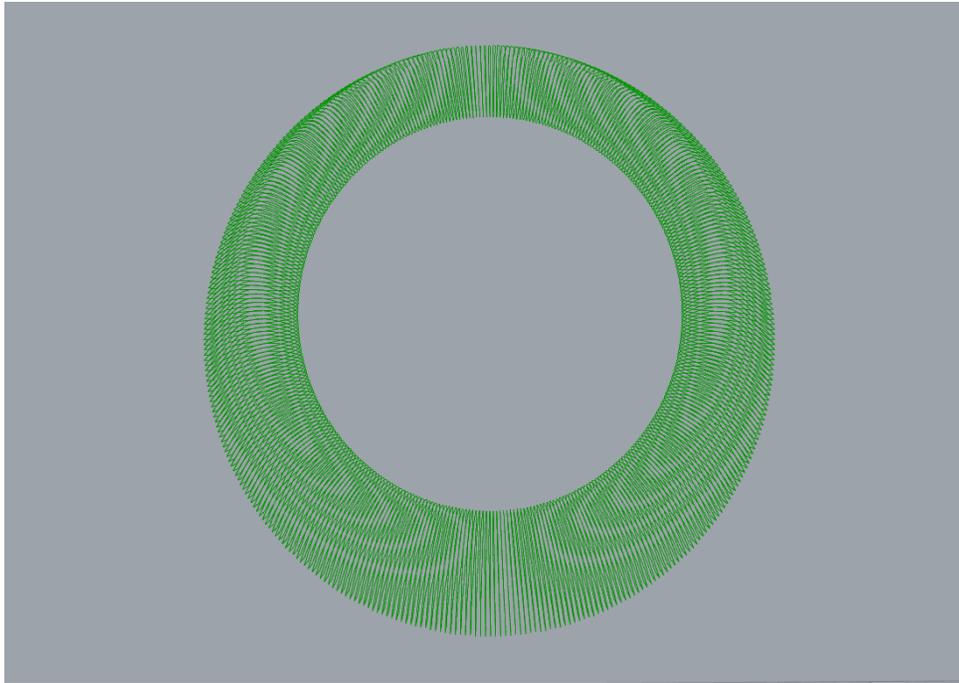


Figure 2.12: Illustration of the curves generated by the 360° rotation of the original diffuser profile curve.

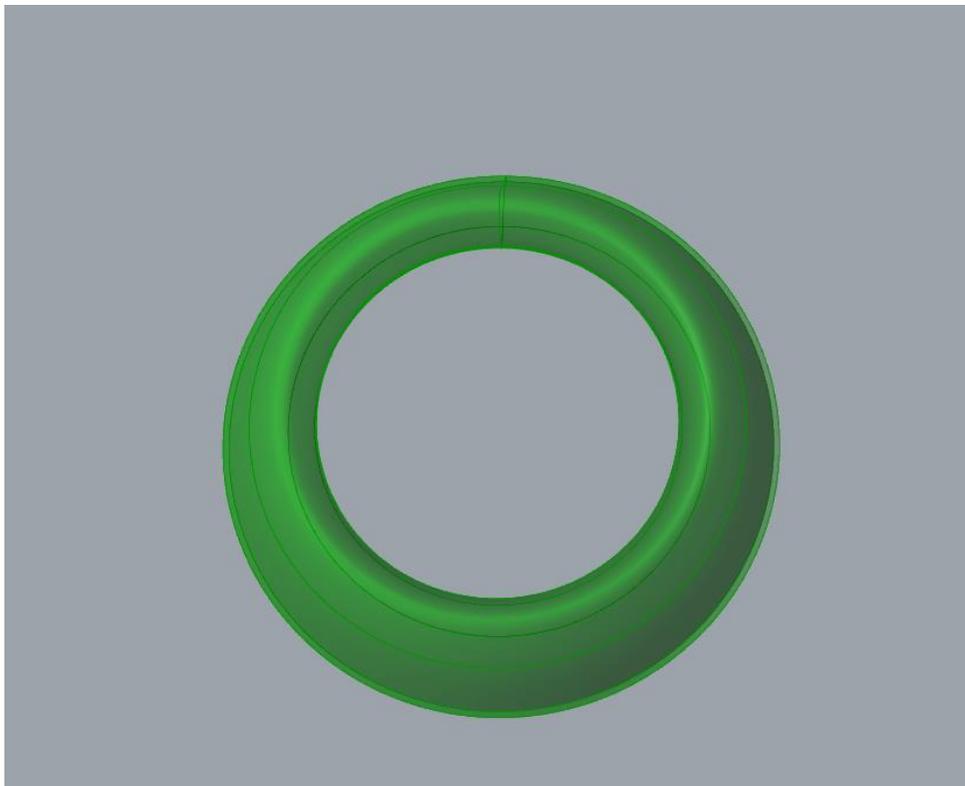


Figure 2.13: Illustration of the external diffuser surface.

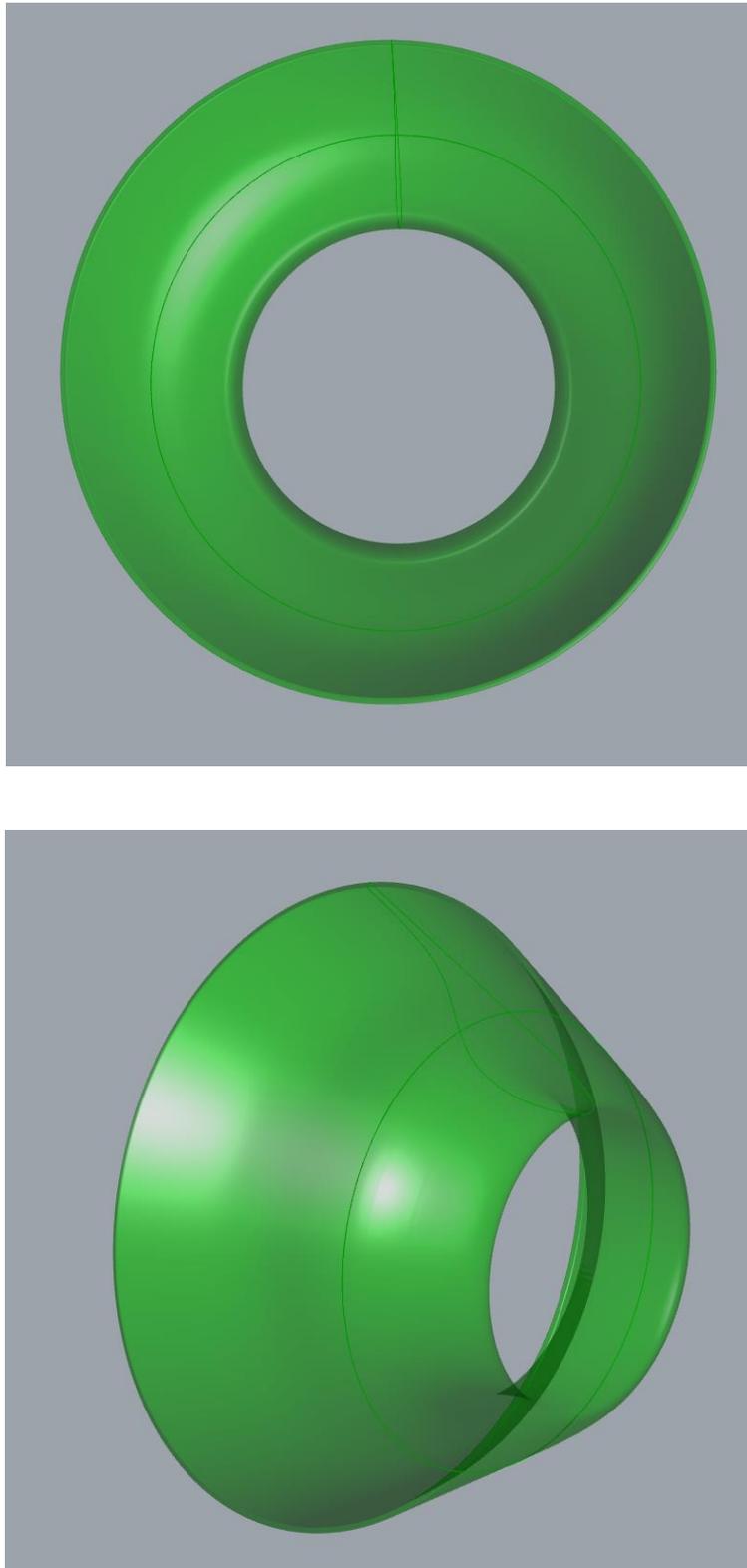


Figure 2.14: Various diffuser geometries produced by the algorithm.

2.2.2.2 The Internal Flap

The next element to be constructed after the diffuser is the internal flap. In order to generate the flap, the algorithm uses a similar methodology to the one employed for the construction of the diffuser. However, the flap's geometry is characterized by high curvature near the leading and trailing edges; thus an alternative approach is followed in order to create a smooth surface, which can then lead to the generation of a fine computational mesh. Initially, the flap's profile curve is defined by a direct interpolation through a set of points, imported from the same text file (*elements_points.txt*) used for the creation of the diffuser. By following the same steps as for the creation of the diffuser, the flap's profile curve may be transformed or translated through the procedures of scaling and translating respectively, as presented in Figure 2.15. Consequently, the algorithm allows the user to create a variety of flaps as well as define the 3D position of the internal flap.

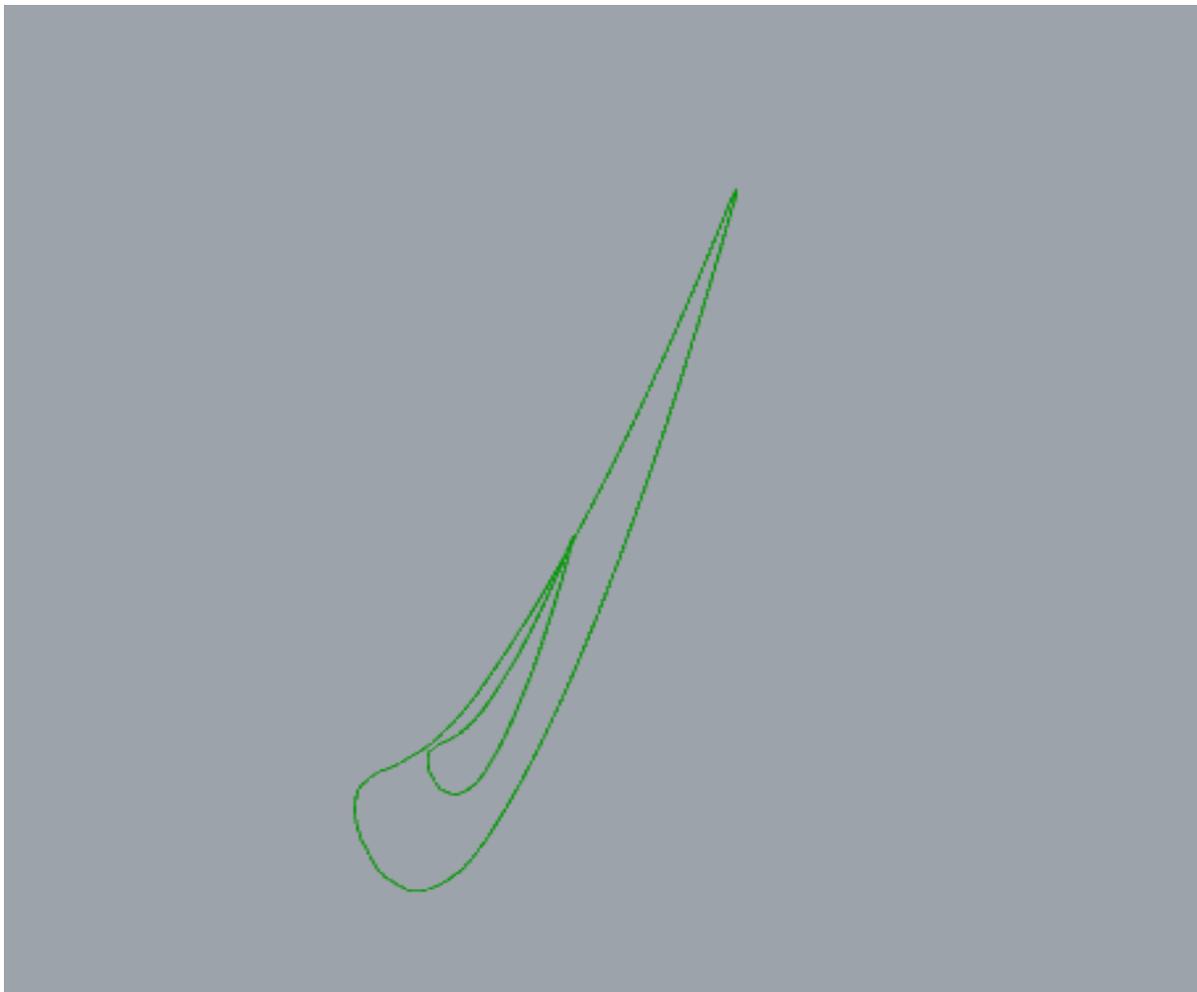


Figure 2.15: Application of the “Scale” operation in the case of a flap profile.

After completing the above procedures, the profile curve is being rotated by 360 degrees around the x -axis (axis of symmetry), generating a circular set of profile curves as in the case of the diffuser. This is essential in order to provide a smooth lofting procedure in the next step. Nevertheless, this does not ensure that the produced surface is ideal for the generation of a high quality mesh. The above conclusion emerged after meshing a number of different flaps resulting in non-satisfactory results, especially near the flap's leading edge, as mentioned in the beginning. For this reason, the algorithm uses a series of design components, which partially alter the procedure for the generation of the internal flap's surface. Initially, the points located at the leading and trailing edges of the profile curves are calculated and then each curve is divided into two separate sections, as shown in Figure 2.16. As a result, the number of the profile curves is doubled and each curve-section is sorted into two lists. The first list includes all the curves that correspond to the flap's pressure side whereas the second list includes all the curves corresponding to the suction side of the internal flap [Cha17].

The next step is to generate the flap's surface through the divided curves. Therefore, the surfaces corresponding to the flap's pressure and suction side are generated through Grasshopper revolution command. As a result, the surfaces are created in a simple way without any discontinuities near the leading and trailing edges. Eventually, the two surfaces are joined together resulting in the internal flap's outer surface. The produced flap is characterized by a smooth surface near the leading and trailing edges; therefore they can be easily meshed as explained in the next chapter. Finally, as mentioned in the beginning of this chapter, the right choice of tolerance is a very important parameter for the proper operation of the Boolean commands, used for the creation of the flap. A sample design of the internal flap is depicted in Figure 2.17.

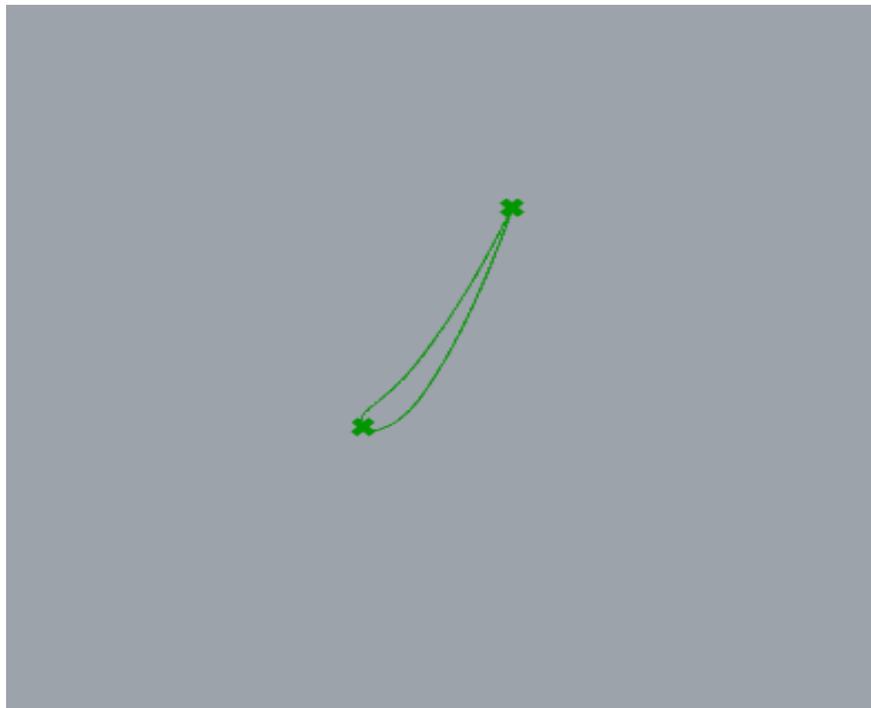


Figure 2.16: *The division of the internal flap profile curve based on the points located at the leading and trailing edges.*

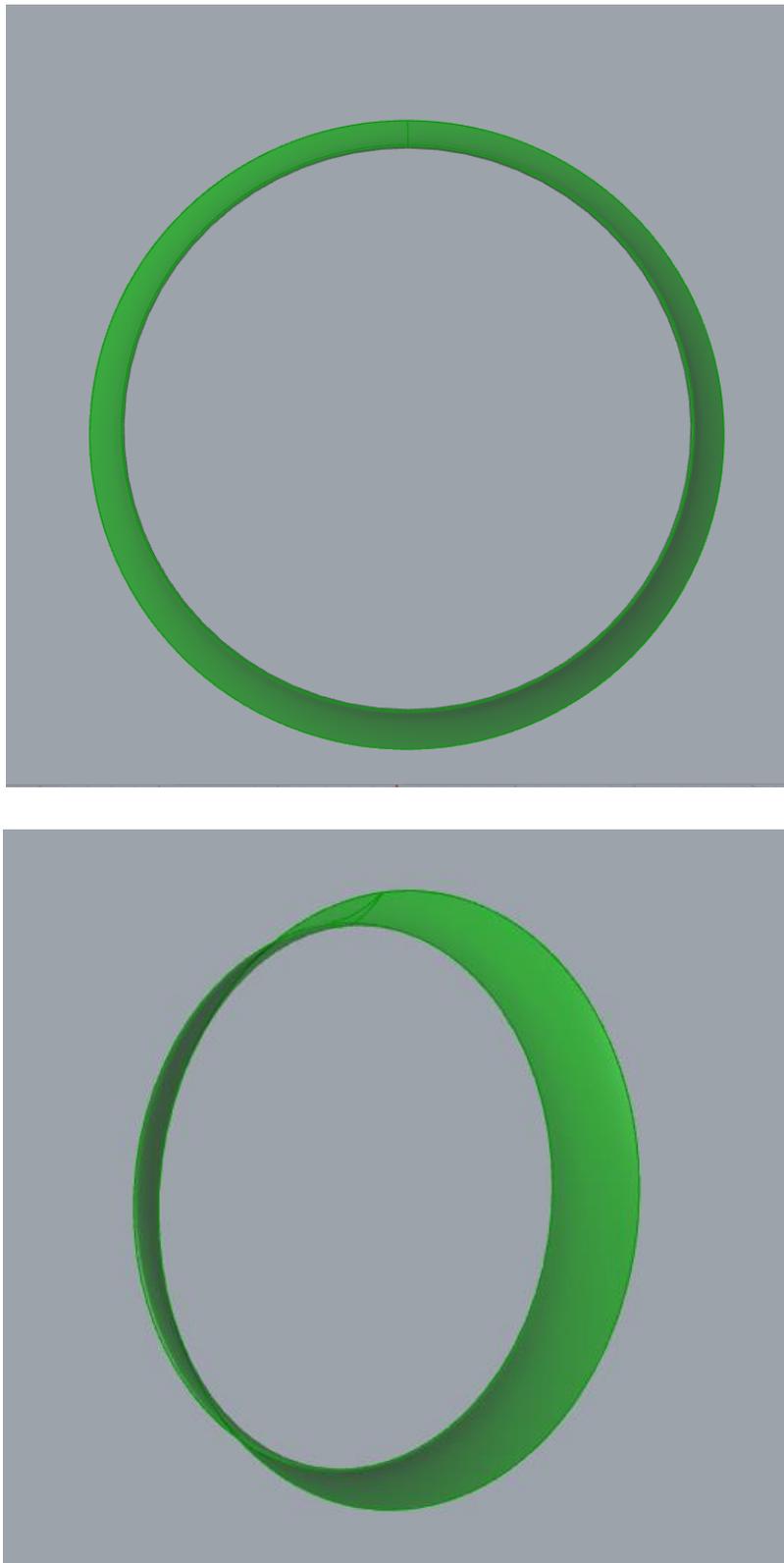


Figure 2.17: The resulted flap surface.

2.2.2.3 The Hub

The next geometry produced by the algorithm is the hub. The DAWT's hub is created through a direct interpolation of a set of points, which are also imported through the text file (*elements_points.txt*) used for the construction of the diffuser and the internal flap. After the interpolation of the imported points, the algorithm allows the user to transform the hub's profile curve through the scaling and translation function, similarly to the diffuser and internal flap profile curves. As mentioned previously, the profile curve (Figure 2.18) is automatically converted into meters in case that the interpolation points are imported in other units. As a result, the user can create a variety of hub geometries as well as change its location in the 3D space. Once the hub's profile curve is generated, the outer surface is produced through a 360 degrees rotation, as presented in Figure 2.19.

Eventually, the algorithm automatically joins the hub with the DAWT's blades. At this point it should be noted that the blades are fused into the hub, after the distance between the blade's tip and the diffuser's inner surface is calculated. In this way, the algorithm ensures that the blades will not intersect with the diffuser's inner surface, after being incorporated into the DAWT's hub. Again, a good choice of tolerance ensures the success of the union procedure (Boolean operation) between the turbine blades and the hub. Finally, the algorithm automatically moves the resulted solid to the correct position, so that the hub (with the blades) is positioned exactly in the center of the diffuser's inner circle. The outer surface of the hub joined with the turbine blades is shown in Figure 2.20.

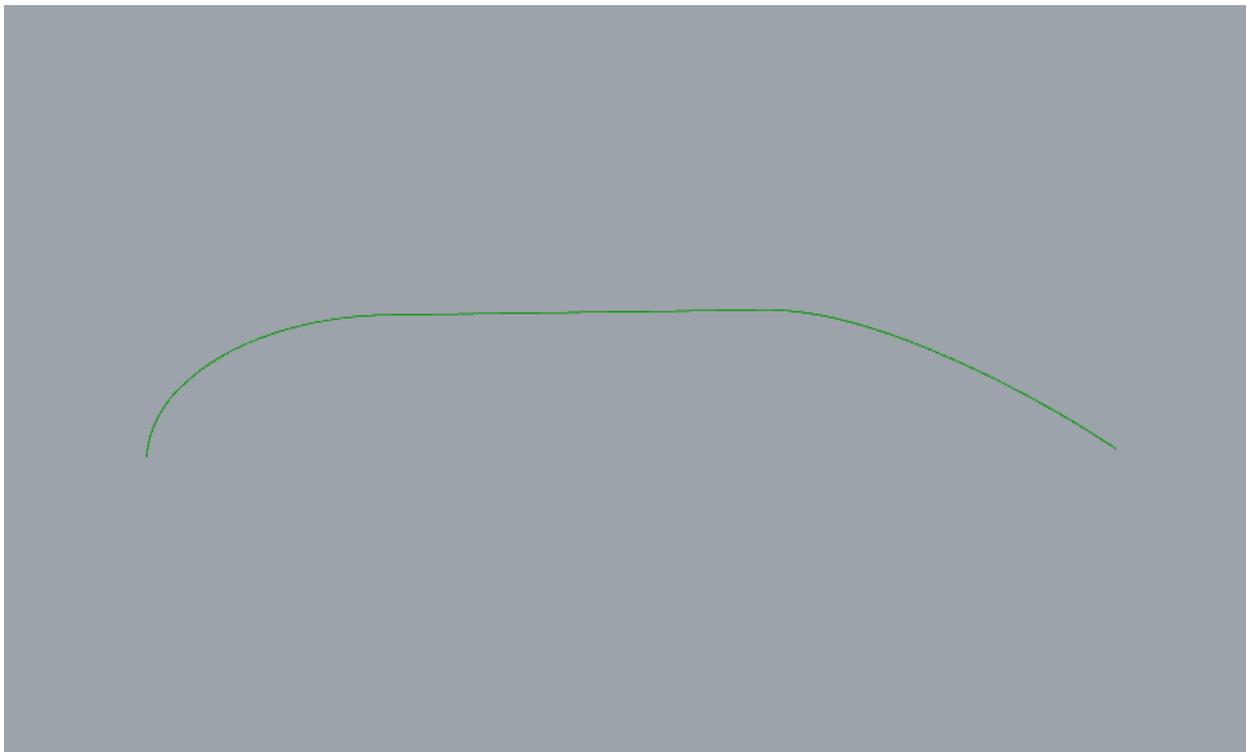


Figure 2.18: The hub profile curve.

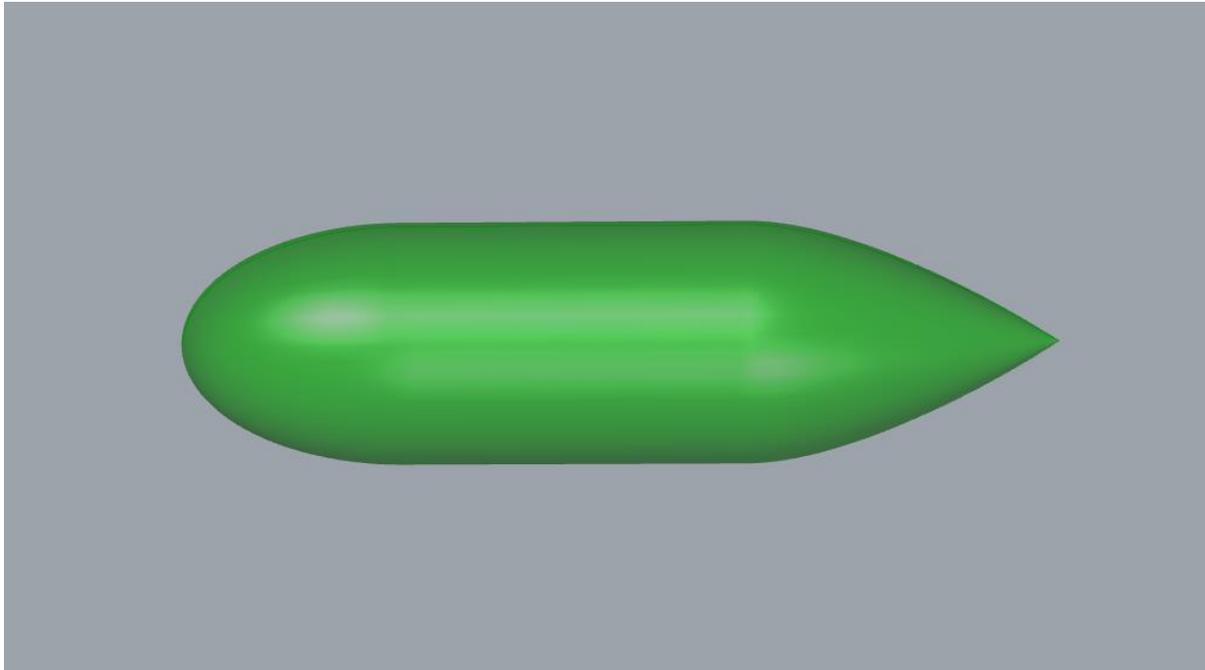


Figure 2.19: The resulted hub surface.

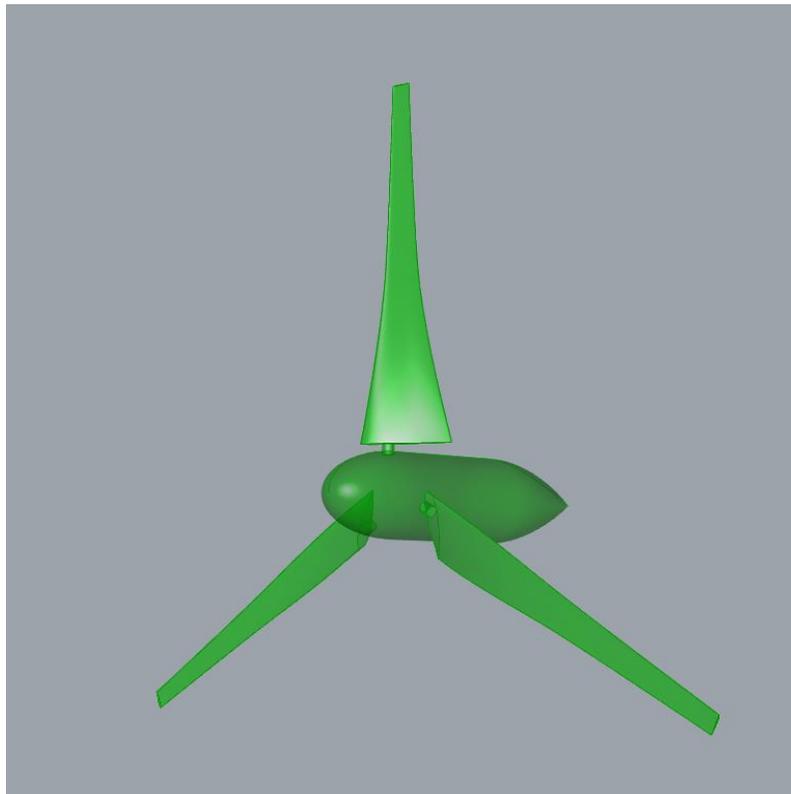


Figure 2.20: Illustration of the blades' connection to the hub.

2.2.2.4 Central Column and Tower

The last two elements designed in order to have an integrated DAWT, are the central column and the tower structures. The configuration of the particular elements was also made by taking into account the data of a text file (*elements_points.txt*), as previously mentioned. The algorithm automatically generates the tower's profile curve (circle) and afterwards produces the tower's 3D surface through the process of lofting, using three design features. Therefore, the user needs to define through the imported text file, the radius and the height for the construction of the tower, due to its cylindrical shape (for simplicity). Additionally, the user may define one extra feature regarding the tower definition, which is the translation distance along the x -axis. Finally, the profile curve is automatically converted into meters in case that the imported radius has been provided in different units; the same procedure is followed, if needed, regarding the tower's height. A sample design of the tower's 3D surface is shown in Figure 2.21.

After the creation of the tower, the algorithm produces the DAWT's central column. In the case of the central column, its profile curve is created through a direct interpolation of a set of points, which are also imported through the text file (*elements_points.txt*). The next step is to translate the produced curve, if necessary, along the x -axis, as in the case of the tower. In order to do so, the user needs to prescribe the translation distance through the imported text file. In this way the user may change the column's position along the x -axis, in order to align it with the DAWT's hub [Cha17]. Furthermore, the profile curve is automatically converted into meters in case that the interpolation points have been provided in different units. Eventually, the central column of the DAWT is created through the procedure of lofting, resulting in the integration of every individual DAWT part. The produced 3D surface of the central column is depicted in Figure 2.22.

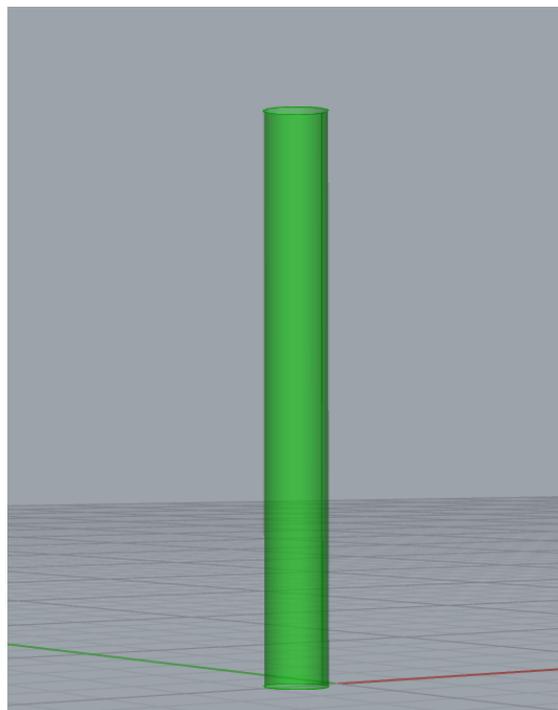


Figure 2.21: *The tower surface.*

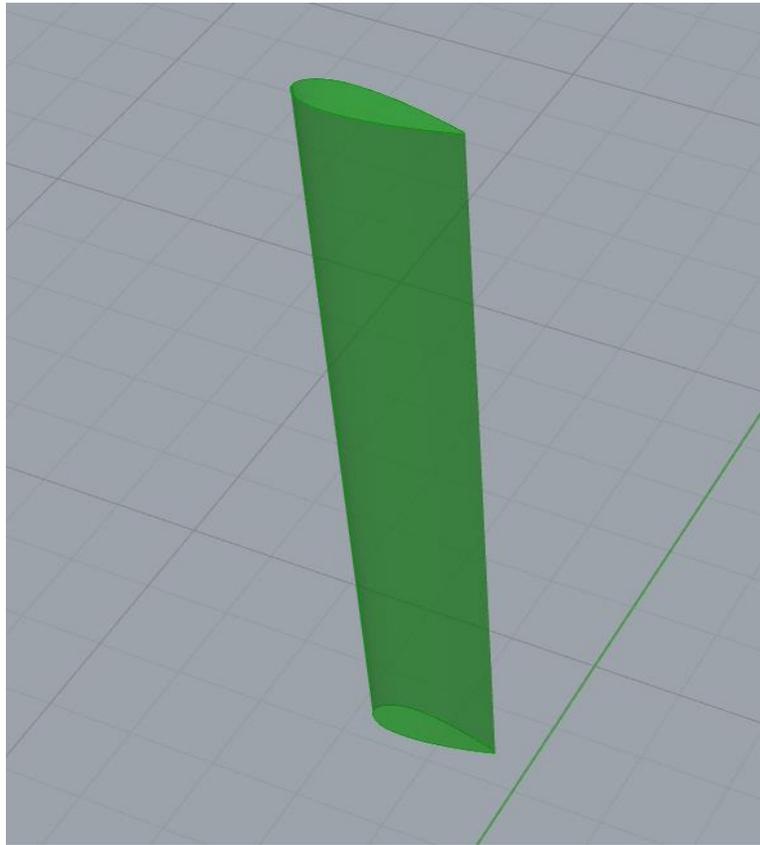


Figure 2.22: The surface of the central column structure.

2.2.2.5 DAWT Assembly

The final step in order to complete the 3D creation of the DAWT is to put together every single part following a specific order. At this point, it is important to mention that all the parts are joined together in pairs, in order to avoid any failures during this “sensitive” procedure. Eventually, the algorithm joins the pairs between them, creating the final form of the DAWT as a single solid. A key factor for the success of the joining procedure, as mentioned before, is the tolerance definition; thus the right choice of tolerance ensures the successful operation of the union commands (Boolean commands) between the DAWT’s individual parts [Cha17].

The first two parts to be joined are the central column with the internal flap. In order to achieve the fusion of the first geometry into the other, the algorithm uses the *Solid Union* command (Figure 2.23), which is a Boolean component of Grasshopper® for the union of surfaces and volumes. However, before joining the two individual parts, an essential condition is needed so that the corresponding volumes are properly positioned, allowing for a smooth union procedure to take place. Therefore, the algorithm automatically moves the particular volumes at the correct 3D positions, ensuring that there is not overlapping between the surfaces of the two parts; thus the volumes intersection is properly configured. As a result, the two individual volumes are joined together, resulting in a new single solid, which forms the first pair of joined DAWT parts, as presented in Figure 2.23.

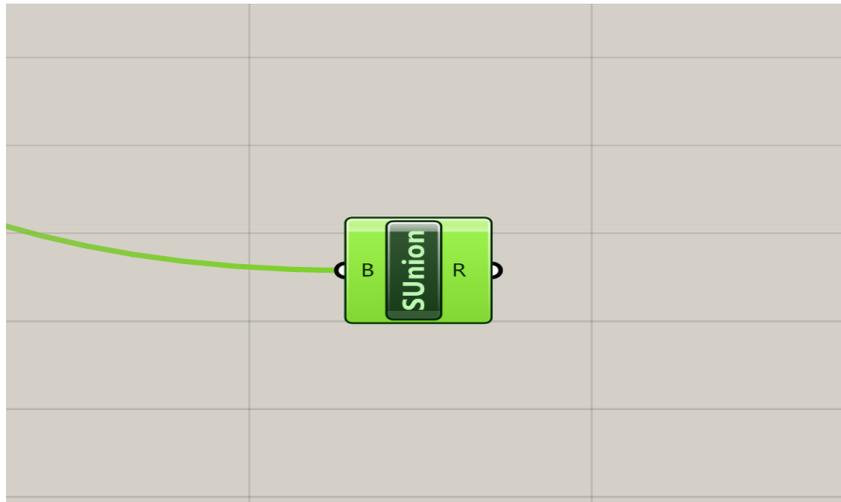


Figure 2.23: The Solid union component.

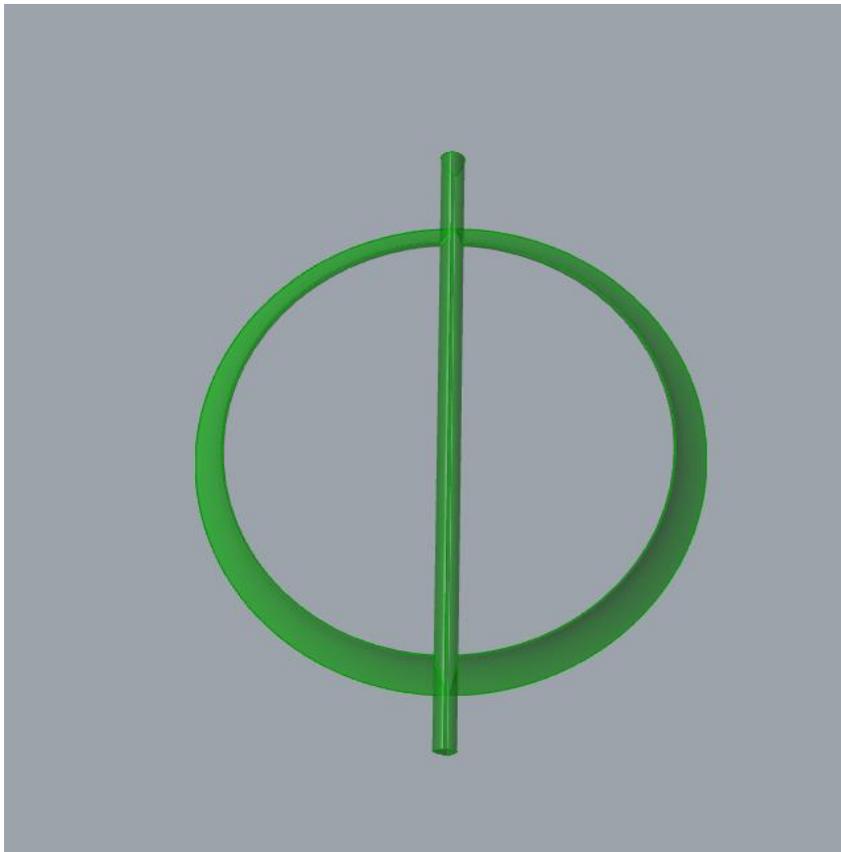


Figure 2.24: The central column joined with the internal flap (1st pair).

The next two parts to be joined are the diffuser along with the DAWT's tower. As mentioned above, at first the corresponding volumes are automatically positioned, so that the union procedure can take place properly. The resulting volume, after joining the diffuser with the tower, consists a new solid, which corresponds to the second pair of DAWT geometries. The emerged solid is shown in Figure 2.25. After completing the formation of the particular pair, the algorithm joins the first with the second pair. The resulted solid, which forms the third union pair, is shown in Figure 2.26.

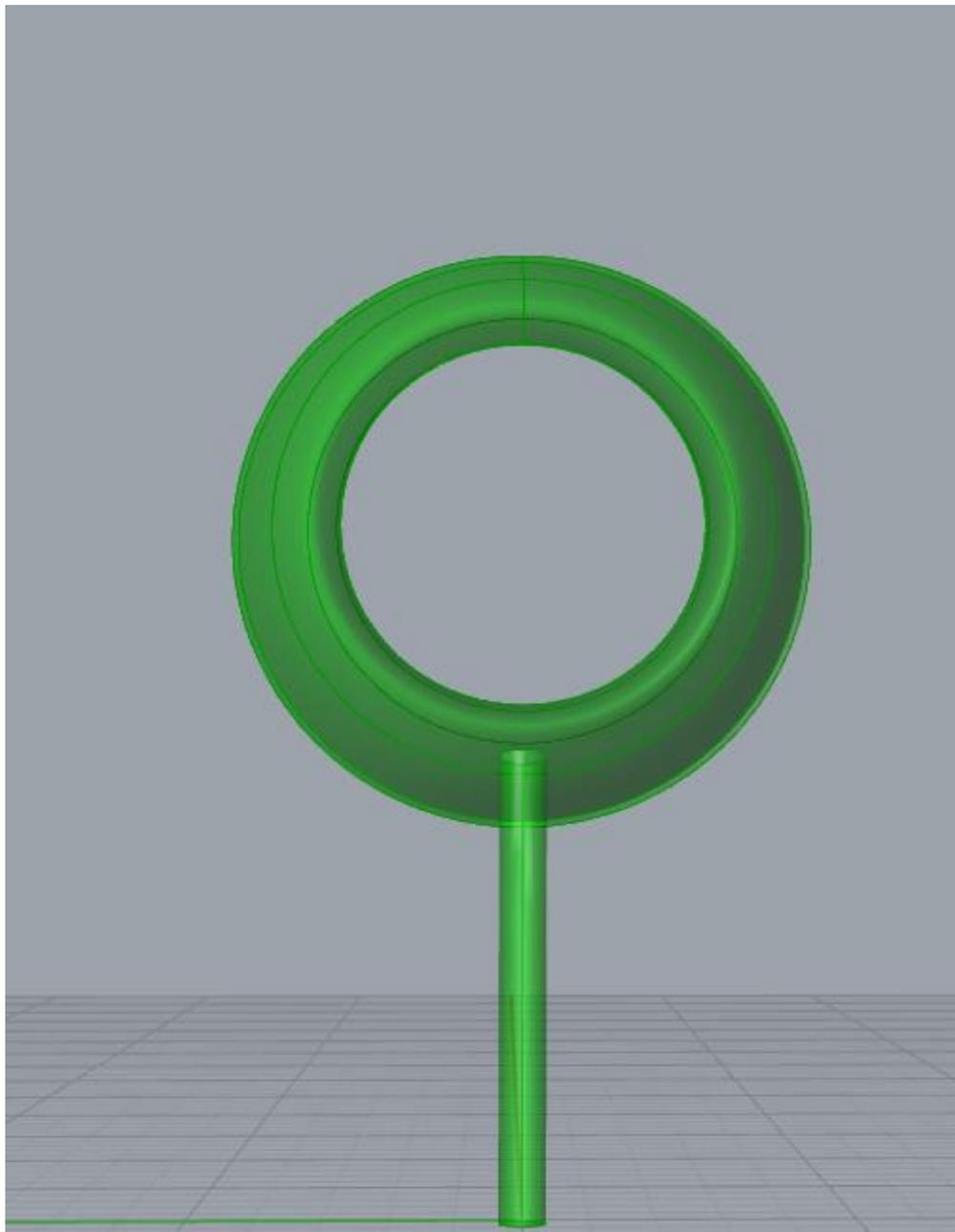


Figure 2.25: The diffuser joined with the tower (2nd pair).

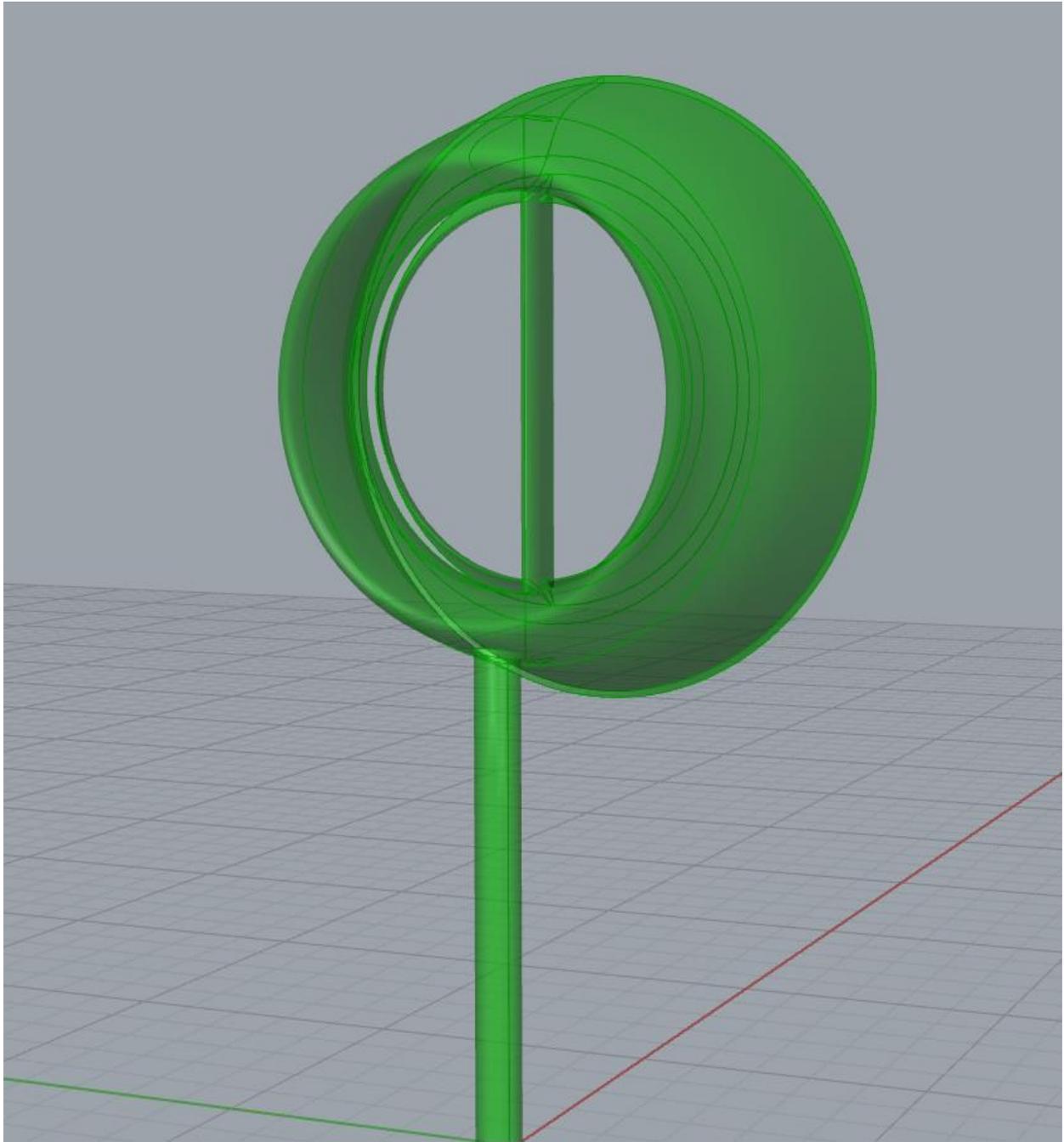


Figure 2.26: Union of the first two geometry pairs.

As mentioned in a previous subsection, the turbine blades are automatically fused into the hub; this can be considered as the fourth pair of DAWT parts. Therefore, the last step in order to complete the DAWT assembly, is to join the remaining two pairs (the third pair with the fourth pair). Eventually, all of the DAWT geometries are joined together, creating an integrated DAWT as one single solid. A sample design of a complete DAWT system is depicted in the following figures.



Figure 2.27: The complete DAWT system.

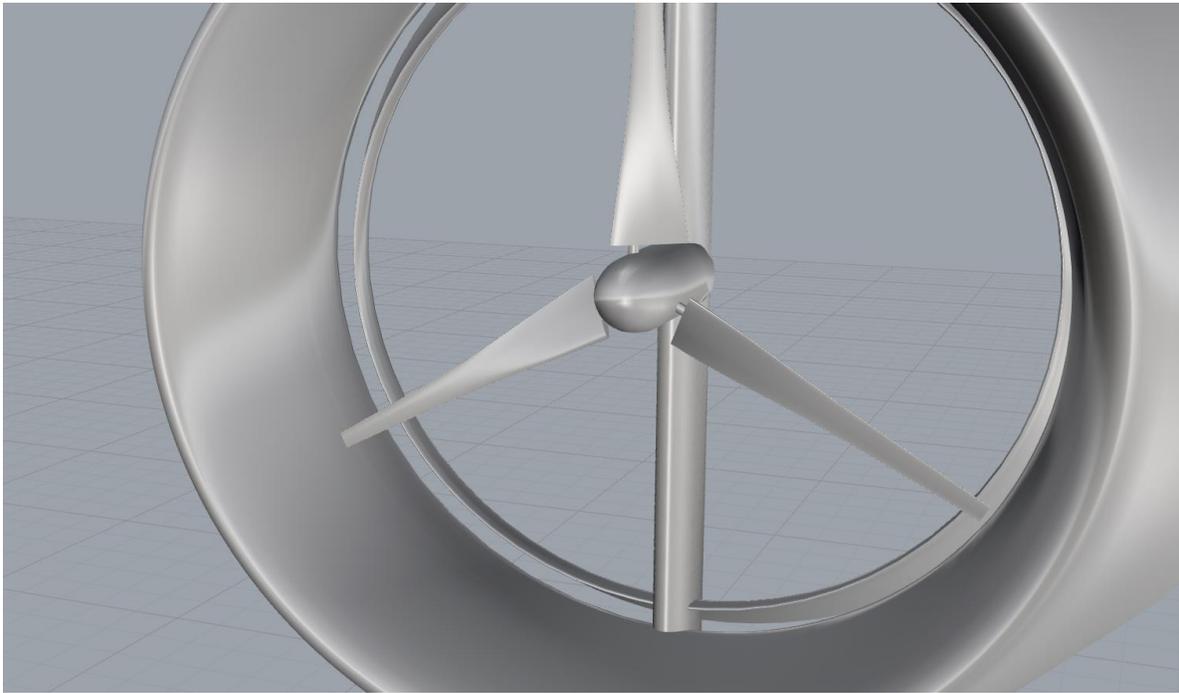


Figure 2.28: Illustration of the final DAWT system [Cha17].

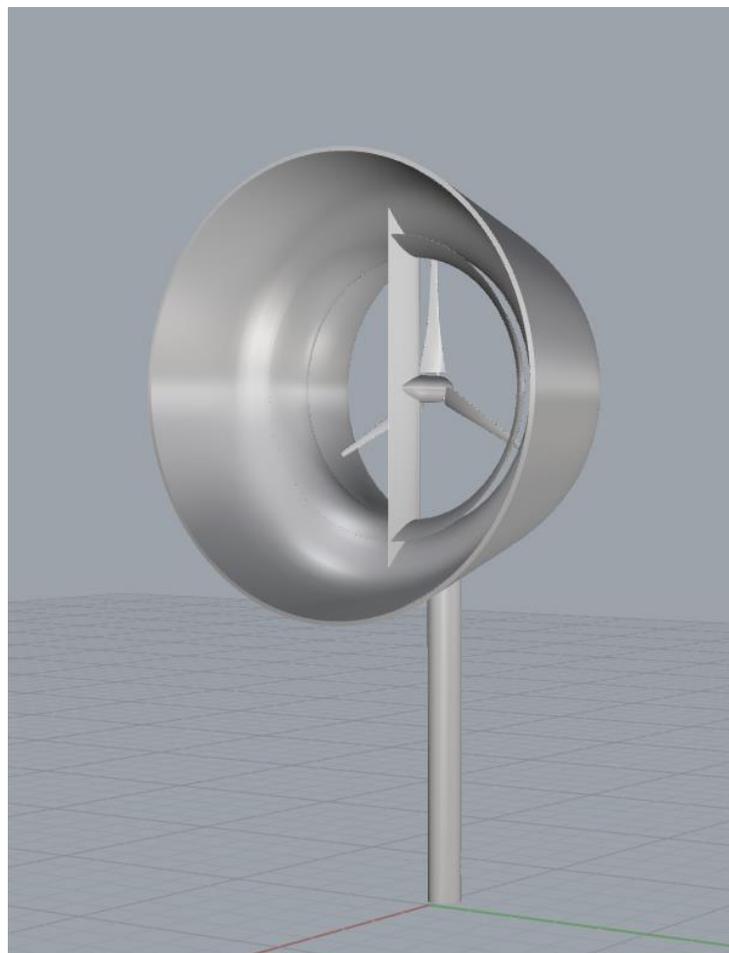
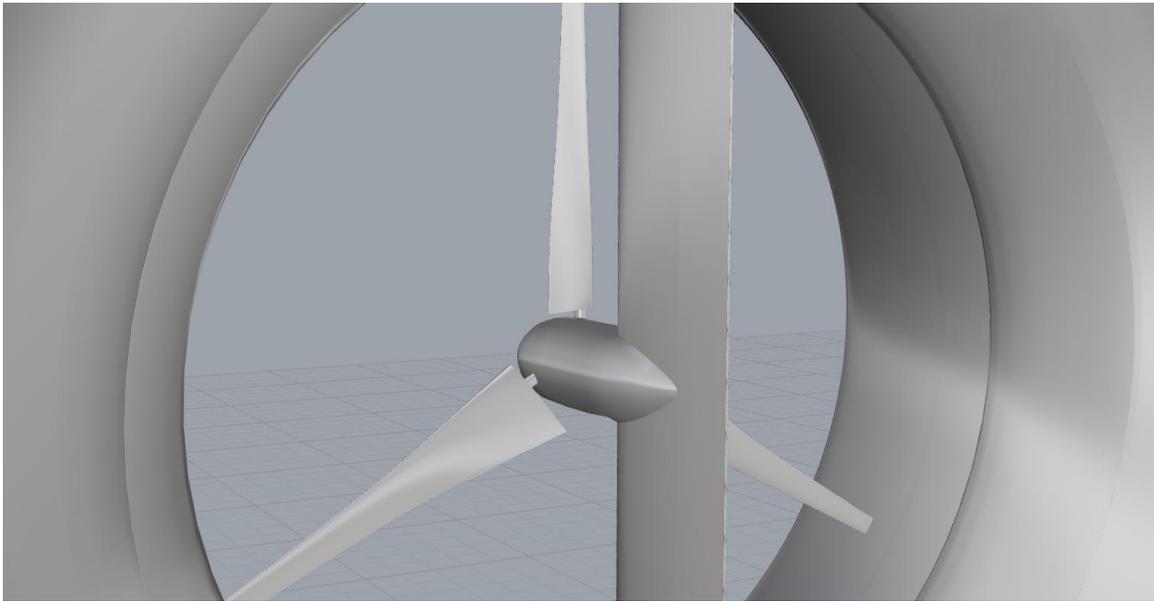


Figure 2.29: Illustration of the final DAWT system.

3. Mesh Generation

In this chapter, the mesh generation for the numerical simulation of the flow around a Diffuser Augmented Wind Turbine (DAWT) is presented. One of the most important stages, during the meshing of such large objects, is the definition of the total number of elements, which is mainly affected by the surface mesh size. As long as complex geometries are concerned, such as the turbine blades, a large number of surface mesh elements is required, in order to model accurately the blade curvature and produce a computational mesh of high quality, suitable for the accurate simulation of the associated physical phenomena. However, the size of the surface mesh must be in line with the capabilities of the computer system used for the mesh generation. Consequently, a fine compromise has to be made, between the mesh quality and the available computational resources, in order to have the desired results. Another key factor during the mesh generation procedure of a wind turbine system is the choice of the number and thickness of prismatic layers, which are constructed between the surface and the tetrahedral volume mesh. The right choice of layers can have a great impact on the results of the CFD analysis [Cha17].

3.1 Surface mesh

The first step to begin the meshing process is to separate the DAWT's parts into two different volumes; each volume is manipulated in a separate ANSA[®] file. The first volume, denoted as *Volume I*, consists of the rotor blades and the nose cone (i.e., the front part of the hub), surrounded by a cylinder, which defines the corresponding flow domain [Cha17]. The cylinder is designed in such a way, so its radius to be equal to the radius of the blade tip, plus the half distance between the blade tip and the diffuser's inner surface (tip clearance). Eventually, the bodies mentioned above are subtracted from the generated cylinder.

The second volume, denoted as *Volume II*, consists of the tower, the diffuser, the internal flap, the domain cylinder, the central column and the rest of the hub geometry that was not included in *Volume I*. All the geometries mentioned above, compose a single solid which is then subtracted from a box defining the flow domain. The domain box is characterized by a height of *30 meters*, a total width of *50 meters* and a length equal to *90 meters*. Additionally, it is noted that the DAWT is positioned *30 meters* behind the inlet and *60 meters* ahead of the outlet of the box. The cylinder including the nose cone and the rotor blades (*Volume I*) defines the rotating frame of reference, while the box containing all the remaining parts defines the inertial reference frame, which are shown in Figures 3.1 and 3.2 respectively [Cha17].

3.1.1 Volume I

3.1.1.1 Blades mesh

A wind turbine blade is the most difficult part to be meshed, due to the complexity of its geometry. Both the leading and trailing edges are characterized by high values of curvature, which makes the meshing process even harder. At first, in order to have a smooth surface mesh near the leading and trailing edges, a large number of surface elements were used, resulting in an excessive total size, which could not be accepted due to the limited available computational resources. In order to overcome this shortcoming, an attempt to increase the growth rate of the

elements was undertaken, however the resulting mesh quality was rendered unacceptable [Cha17].

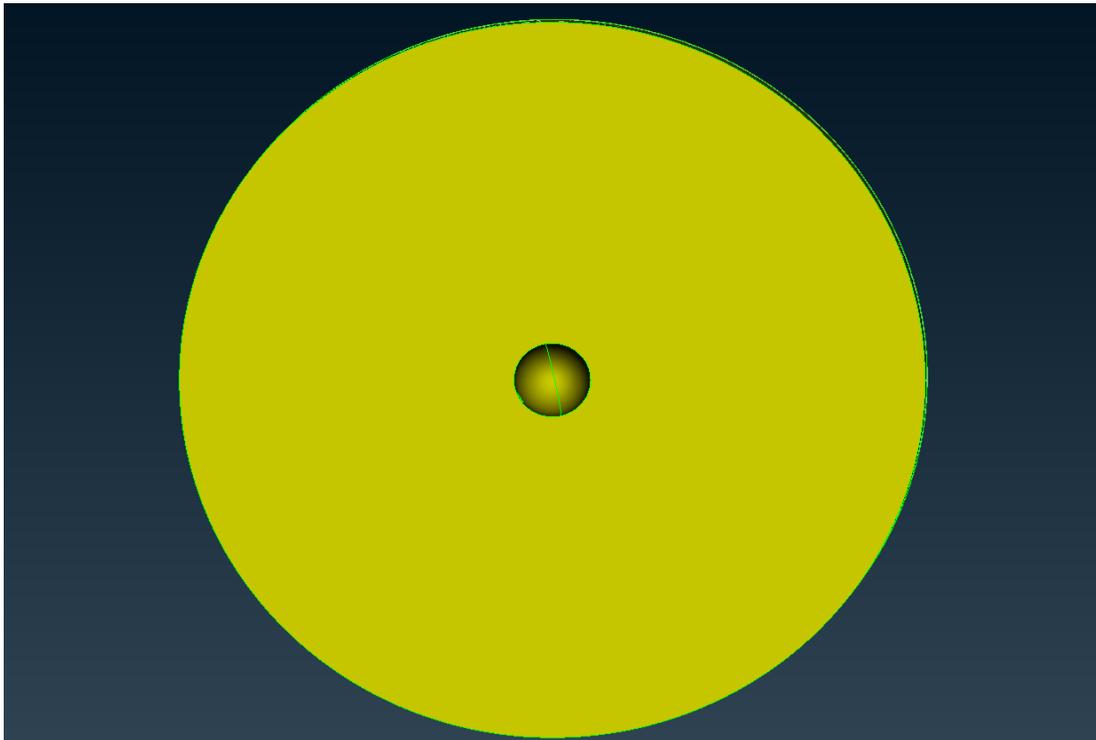


Figure 3.1: Illustration of Volume I (Rotating Frame).



Figure 3.2: Illustration of Volume II (Inertial Frame).

Therefore, an alternative approach was attempted, to reduce the size of the blade surface mesh. According to this approach, every blade was divided into three distinct regions (*root, middle and tip regions* – Figure 3.3) and then a surface mesh was constructed for each blade region, based on different mesh parameters, resulting in a reduced number of surface elements for each blade [Cha17]. The mesh parameters (*CFD parameters*) used for each region, which were obtained based on a trial and error procedure, are presented in Table 3.1.

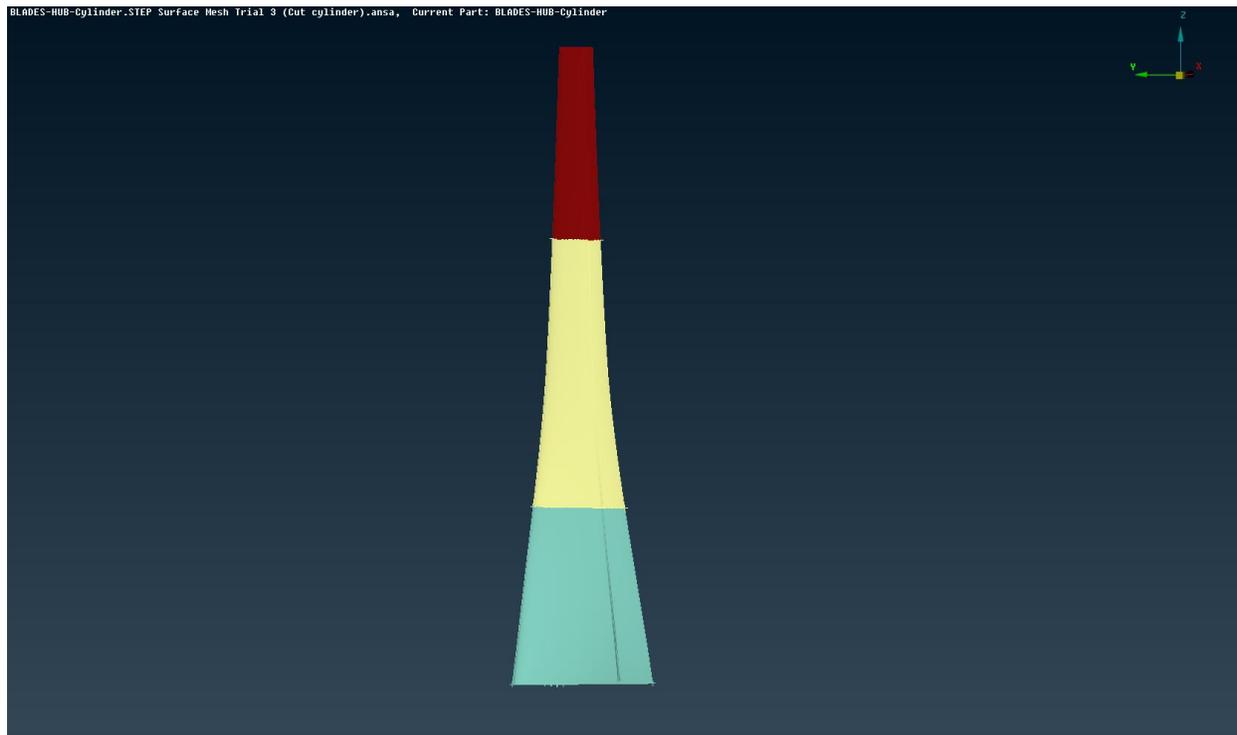


Figure 3.3: The division of the blade into the root (blue), the middle (white) and the tip (red) regions.

Blade Region	Root	Middle	Tip
Elements Type	<i>Triangles</i>	<i>Triangles</i>	<i>Triangles</i>
2nd Order	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>
Interior Growth Rate	<i>1.35</i>	<i>1.25</i>	<i>1.20</i>
Distortion Angle	<i>15</i>	<i>10</i>	<i>10</i>
Minimum Target Length (mm)	<i>10</i>	<i>10</i>	<i>10</i>
Maximum Target Length (mm)	<i>50</i>	<i>40</i>	<i>25</i>

Table 3.1: The parameters for the surface mesh generation of each blade region.

Additionally, in order to achieve a smooth mesh near the leading and trailing edges, the perimeters' length of each region concerning these areas, are defined as shown in Table 3.2. Notice that for the leading edge, a very small perimeter length is used, otherwise the surface mesh cannot follow the blade's high curvature at the particular area. Eventually, all the regions are meshed successfully, resulting in a total number of surface elements for each blade equal to **463,766**. The surface mesh, constructed for the blade configuration, is shown through Figure 3.4 – Figure 3.6.

Blade Region	<i>Root</i>	<i>Middle</i>	<i>Tip</i>
Leading Edge (Perimeter Length)	<i>1.0</i>	<i>0.5</i>	<i>0.2</i>
Trailing Edge (Perimeter Length)	<i>0.6</i>	<i>0.5</i>	<i>0.5</i>

Table 3.2: Leading and trailing edge perimeter lengths.

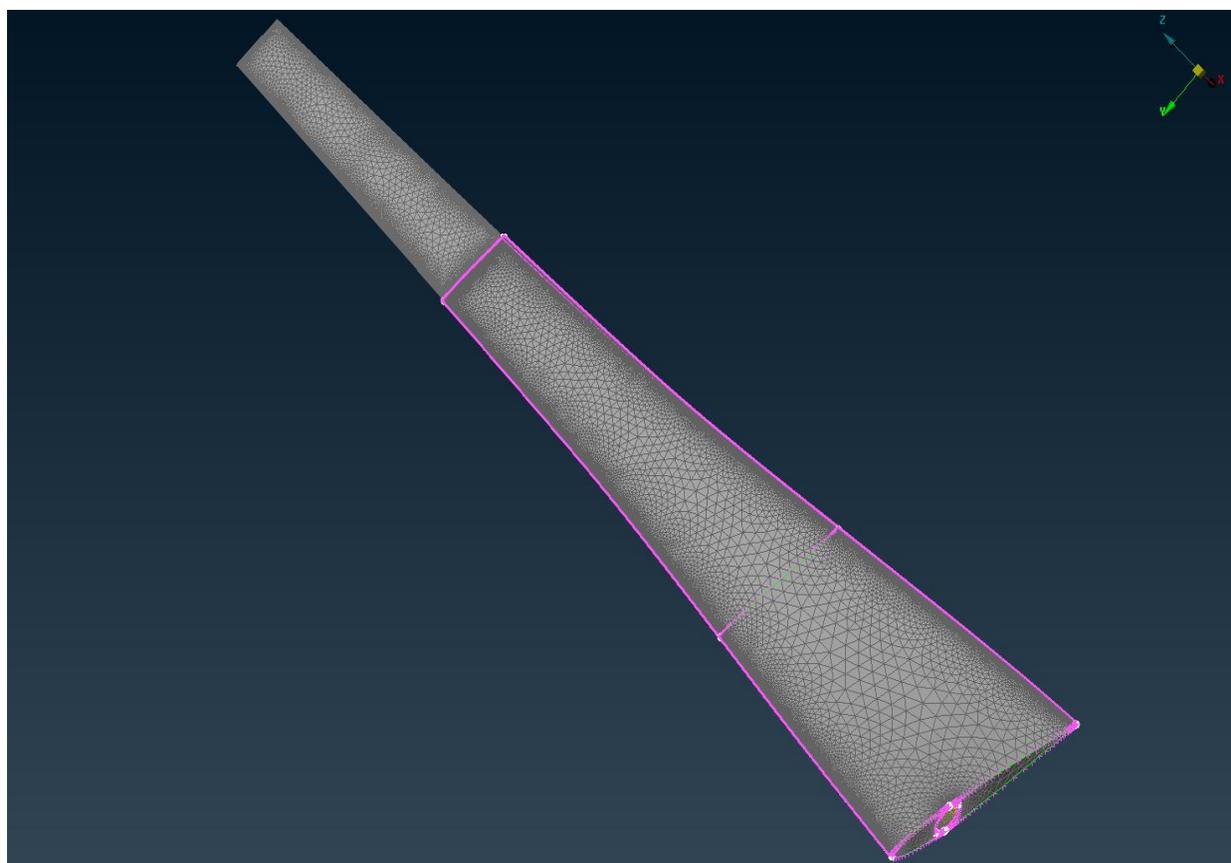


Figure 3.4: The generated surface meshes for each blade region.

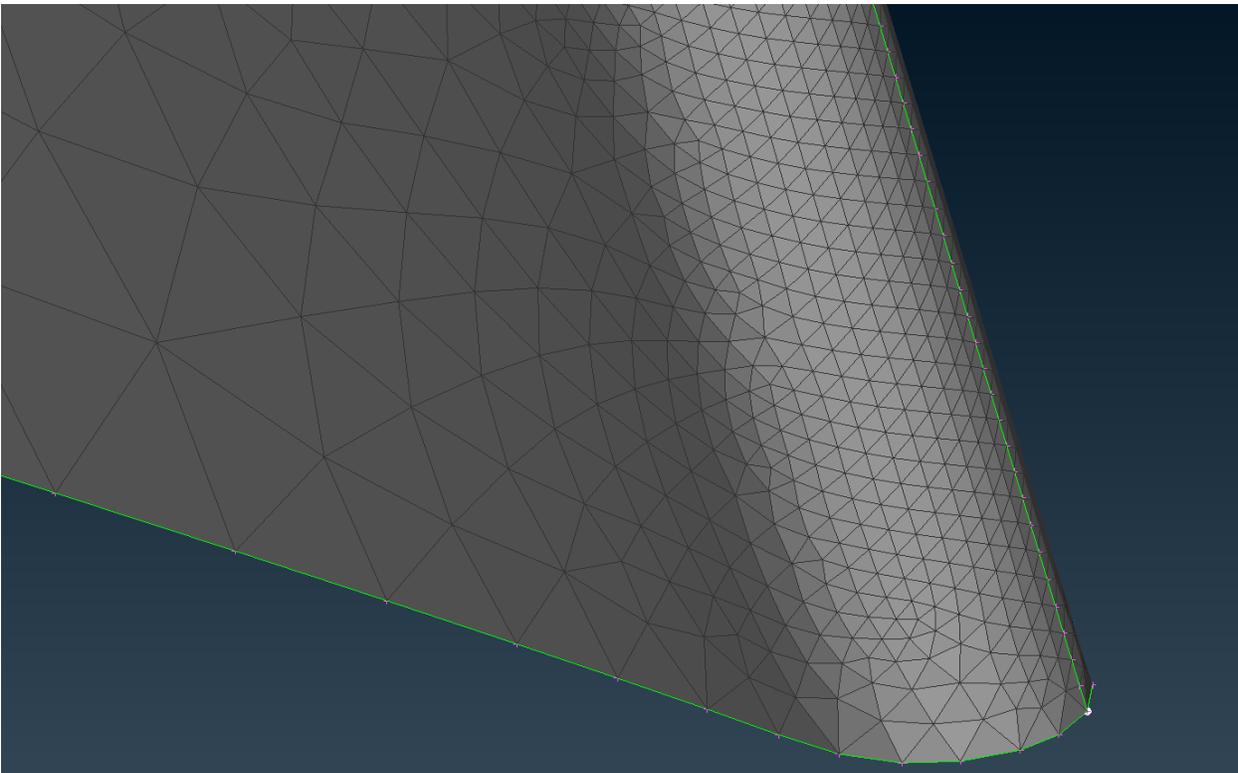
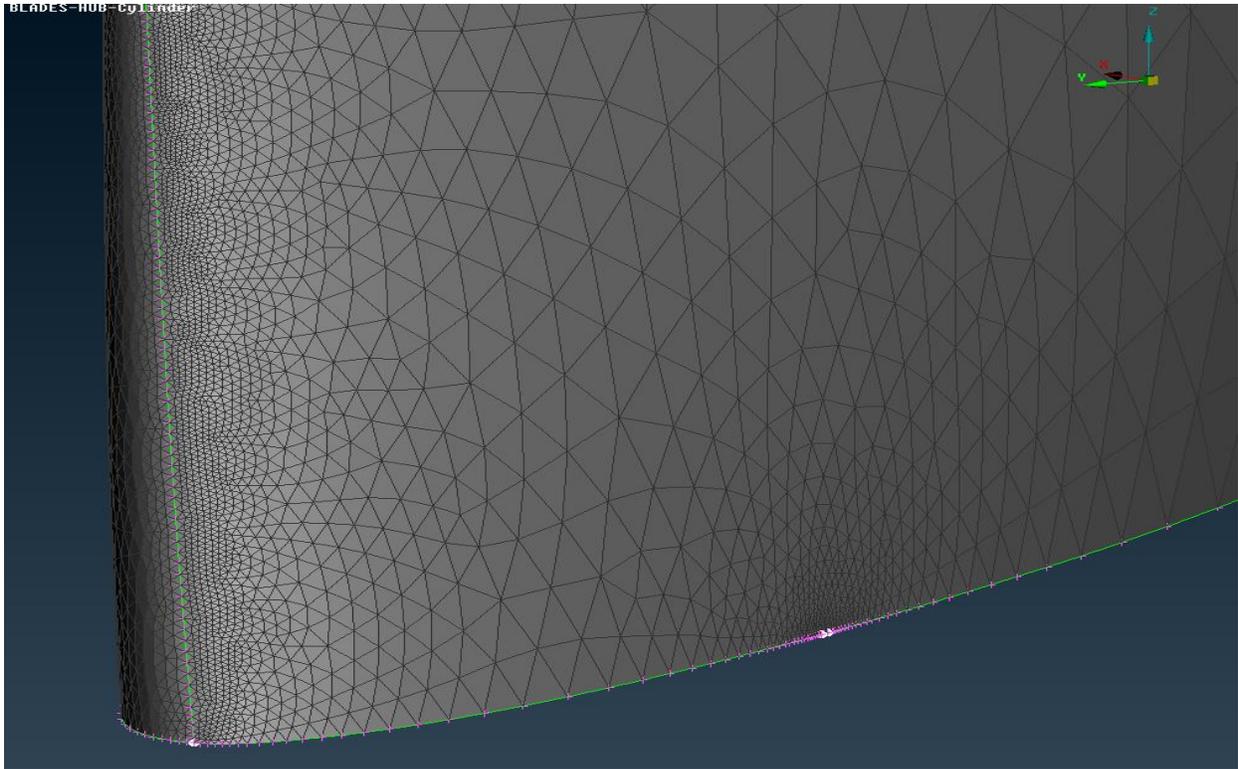


Figure 3.5: Close-up views of the leading (up) and trailing (down) edges at the root region of the blade.

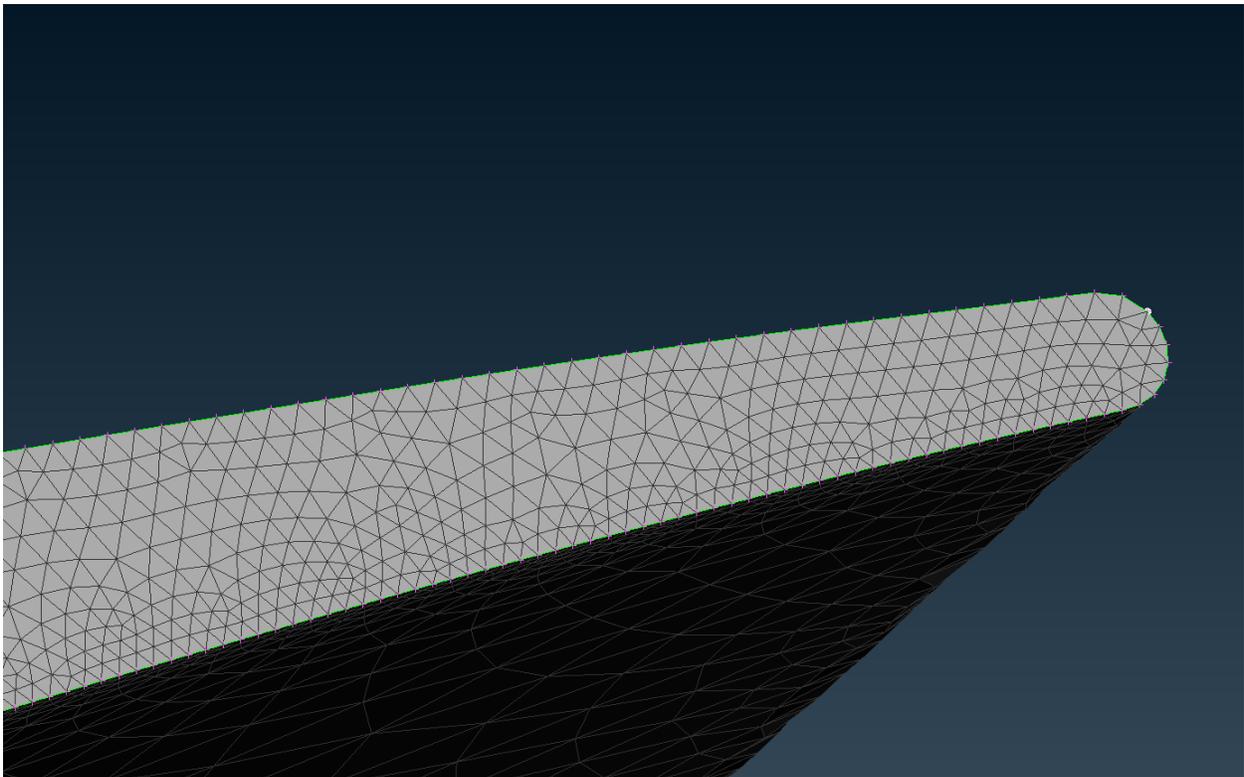
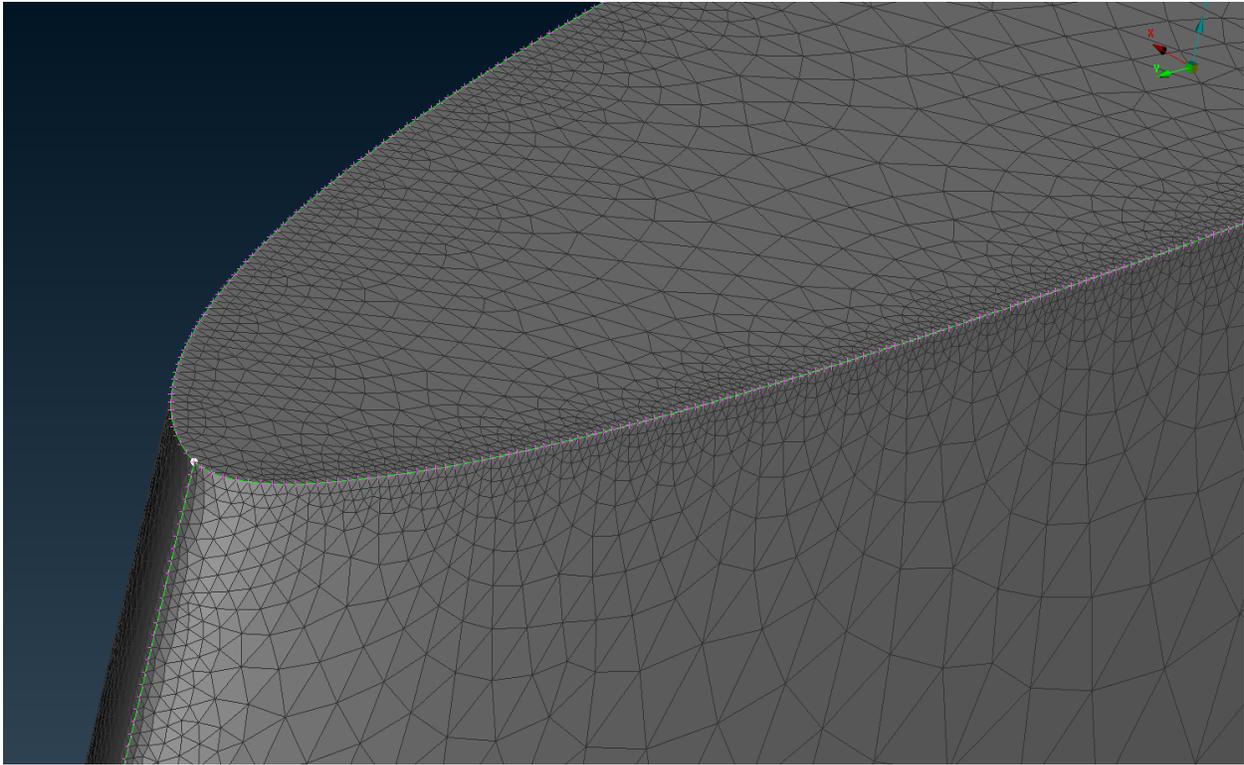


Figure 3.6: Close-up views of the leading (up) and trailing (down) edges at the tip of the blade.

3.1.1.2 Nose Cone and Blade Connectors Mesh

Having completed the meshing of the blades, the next geometries to be meshed are the blade connectors (i.e., the elements connecting the blade to the hub) and the nose cone (i.e., the front part of the hub). Unlike the blades, the particular parts are much easier to be meshed, due to their simple geometry. Initially, the blade connector of each blade is meshed by applying the mesh parameters used for meshing the root region, so a smooth transition between the surface meshes of the two geometries to be achieved. The mesh parameters and the surface mesh of a blade connector are presented in Table 3.3 and Figure 3.7 respectively [Cha17].

Blade Connector Mesh	
Element Type	<i>Triangles</i>
2nd Order	<i>Yes</i>
Interior Growth Rate	<i>1.2</i>
Distortion Angle	<i>15</i>
Minimum Target Length (mm)	<i>10</i>
Maximum Target Length (mm)	<i>20</i>

Table 3.3: The parameters for the surface mesh generation of the blade connector.

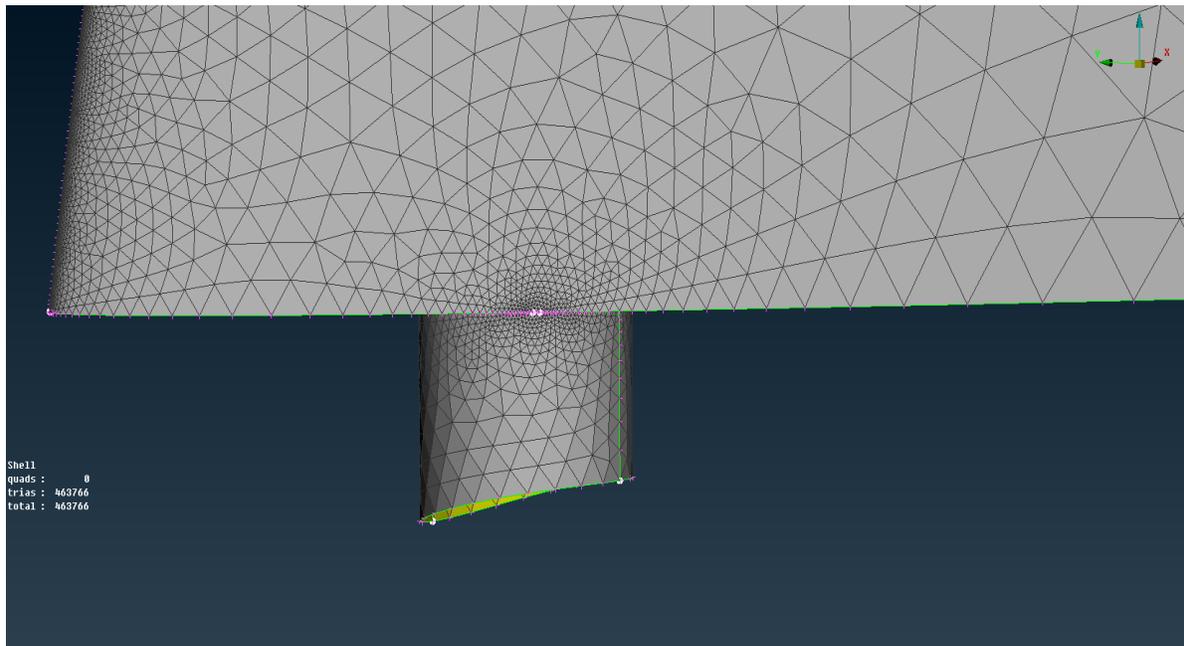


Figure 3.7: The surface mesh of a blade connector.

The next geometry to be meshed is the nose cone. The parameters used for the generation of the nose cone's surface mesh are shown in Table 3.4, whereas the respective surface mesh is presented in Figure 3.8.

Nose Cone Mesh	
Element Type	<i>Triangles</i>
2nd Order	<i>Yes</i>
Interior Growth Rate	<i>1.25</i>
Distortion Angle	<i>15</i>
Minimum Target Length (mm)	<i>10</i>
Maximum Target Length (mm)	<i>40</i>

Table 3.4: The parameters for the surface mesh generation of the nose cone.

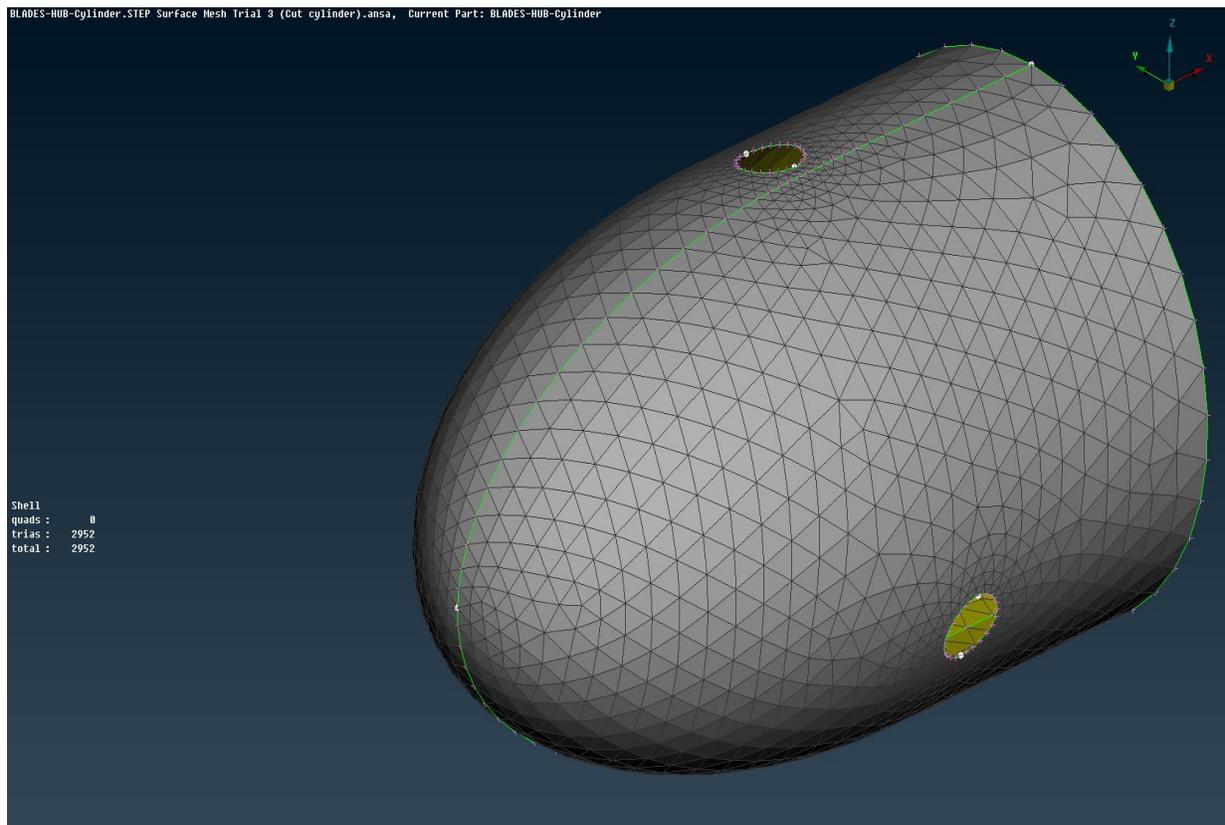


Figure 3.8: The surface mesh of the nose cone.

3.1.1.3 Rotating Cylinder Mesh

The last step in order to complete the surface mesh of *Volume I* is the cylinder's meshing. Although the particular geometry can be easily meshed, there is one extra specification that needs to be taken under consideration. Because of the small distance between the blade's tip and the cylinder's side surface, the areas of the cylinder that are near to the blade's tip need to be meshed with similar parameters used for the tip, in order to avoid any failures during the layers generation process [Cha17]. As a result, the cylinder's side surface is separated into smaller regions, at these areas, where a more dense mesh is required. The mesh parameters used for the cylinder and a cylinder's region are presented in Table 3.5, whilst the resulting surface mesh is shown in Figure 3.9.

	<i>Cylinder</i>	<i>Cylinder Region</i>
Elements Type	<i>Triangles</i>	<i>Triangles</i>
2nd Order	<i>Yes</i>	<i>Yes</i>
Interior Growth Rate	<i>1.4</i>	<i>1.1</i>
Distortion Angle	<i>15</i>	<i>15</i>
Minimum Target Length (mm)	<i>10</i>	<i>10</i>
Maximum Target Length (mm)	<i>400</i>	<i>20</i>

Table 3.5: The parameters for the surface mesh generation of the rotating cylinder.

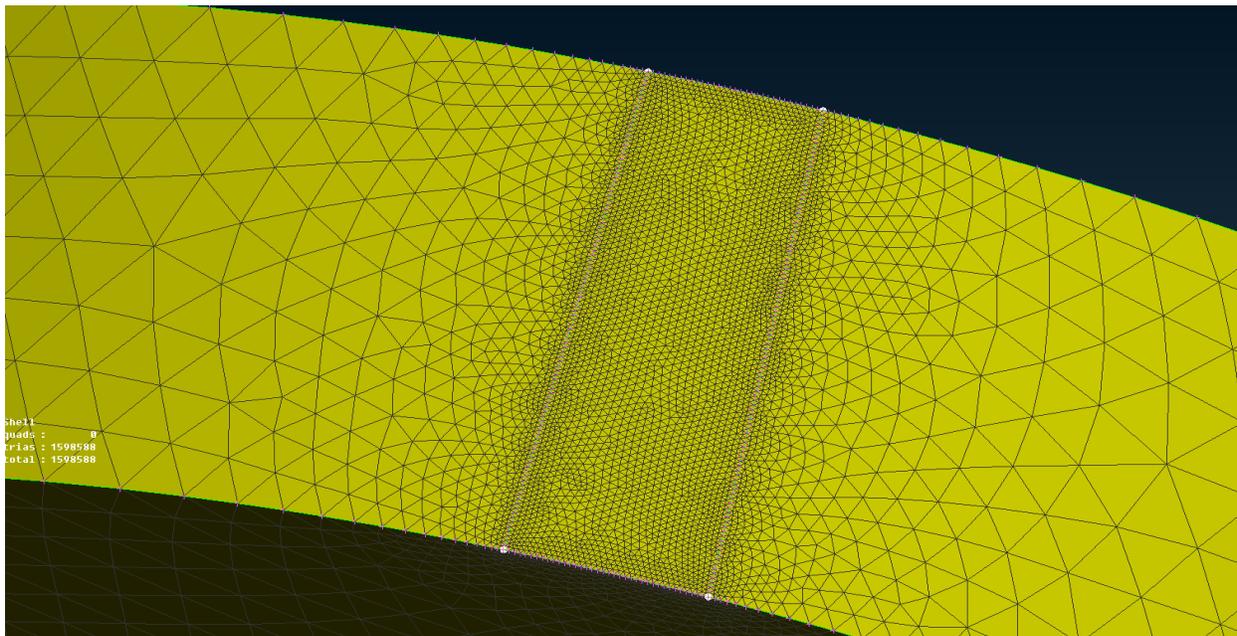


Figure 3.9: The surface mesh of the rotating cylinder, close to the blade tip region.

3.1.2 Volume II

3.1.2.1 Diffuser Mesh

The diffuser is the most difficult part to be meshed among the geometries included in *Volume II*. Initially, the diffuser is divided into smaller regions, as shown in Figure 3.10, to generate a smoother surface mesh near the leading edge and the diffuser's contraction; this also ensures the smooth layers development that will be presented in the next section. In Table 3.6 the mesh parameters used for the construction of the surface mesh of each region are presented, whereas the diffuser's surface mesh is depicted in Figure 3.11.

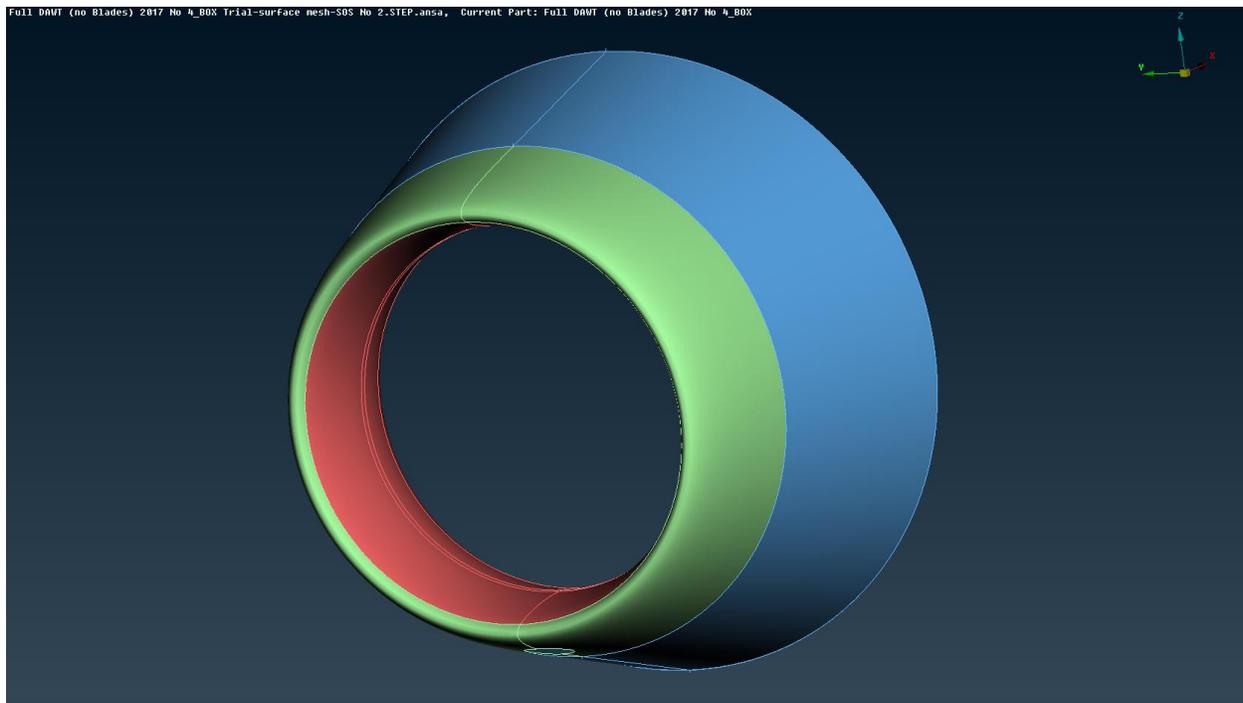


Figure 3.9: The diffuser's surface division into regions of different mesh parameters.

Region	Leading Edge	Contraction	Rest Geometry
Elements Type	Triangles	Triangles	Triangles
2 nd Order	Yes	Yes	Yes
Interior Growth Rate	1.2	1.15	1.3
Distortion Angle	10	10	15
Minimum Target Length (mm)	10	10	50
Maximum Target Length (mm)	50	20	500

Table 3.6: The parameters for the surface mesh generation of the diffuser.

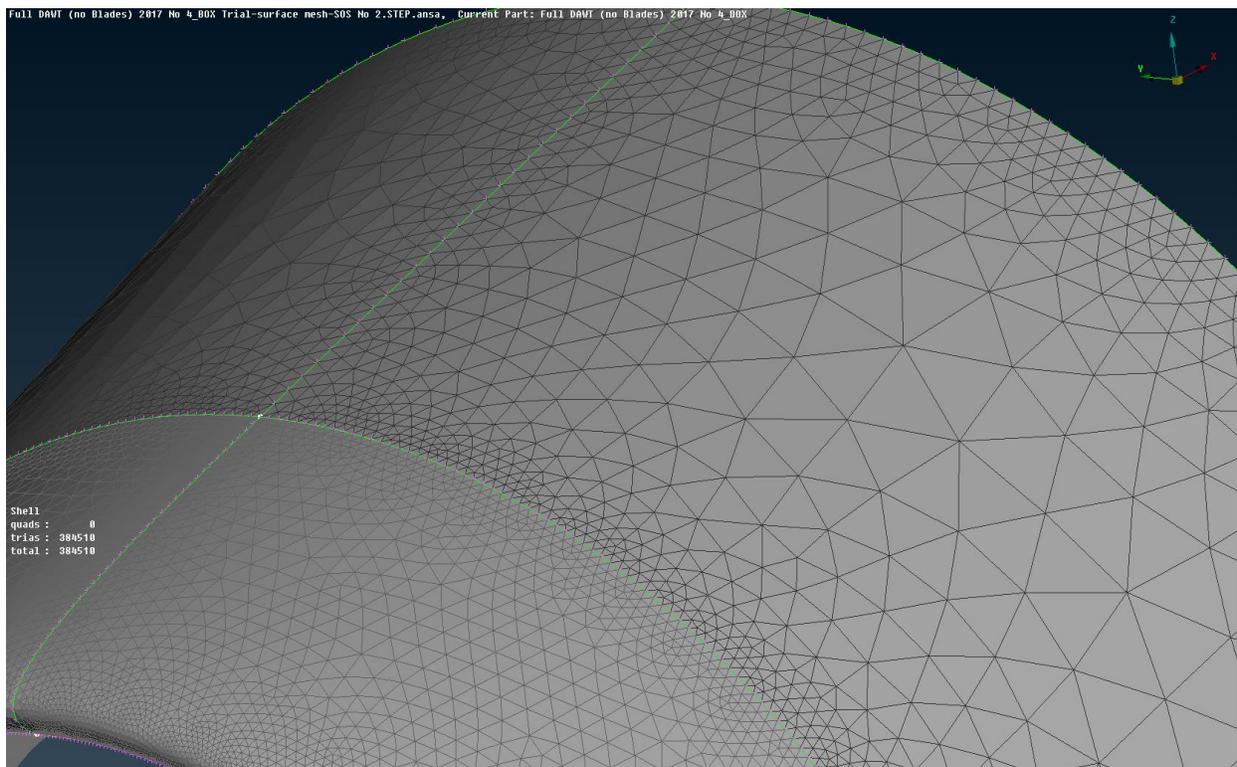
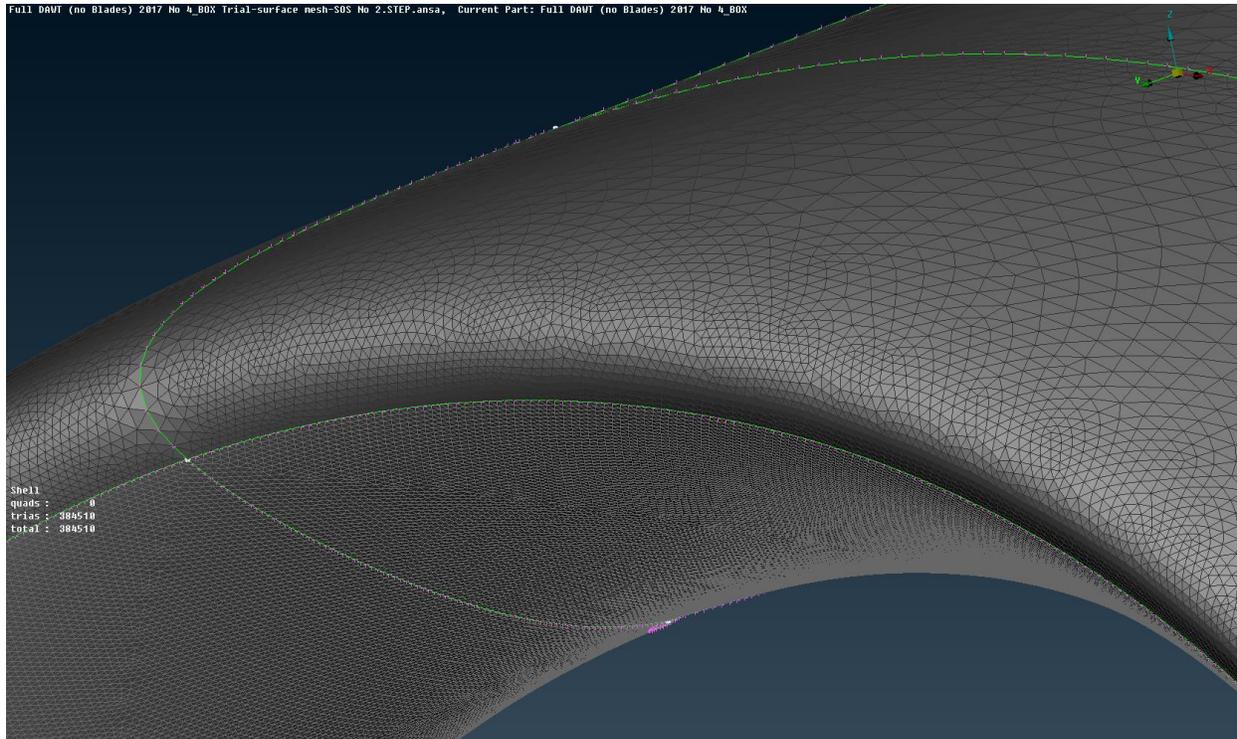


Figure 3.10: The surface mesh at the leading edge (up) and the main body (down) of the diffuser.

3.1.2.2 Internal Flap Mesh

Another part included in *Volume II* that needs a dense surface mesh is the internal flap, due to the high values of curvature at the flap's leading and trailing edges, as explained in the previous chapter. Fortunately, because of the alternative approach used for its construction and its small size, the mesh procedure is easily succeeded, while the total number of the resulting surface elements is manageable. The mesh parameters used in this case are presented in Table 3.7, whereas the surface mesh of the internal flap is shown in Figure 3.12.

Internal Flap Mesh	
Element Type	<i>Triangles</i>
2nd Order	<i>Yes</i>
Interior Growth Rate	<i>1.2</i>
Distortion Angle	<i>10</i>
Minimum Target Length (mm)	<i>10</i>
Maximum Target Length (mm)	<i>80</i>

Table 3.7: The parameters for the surface mesh generation at the internal flap.

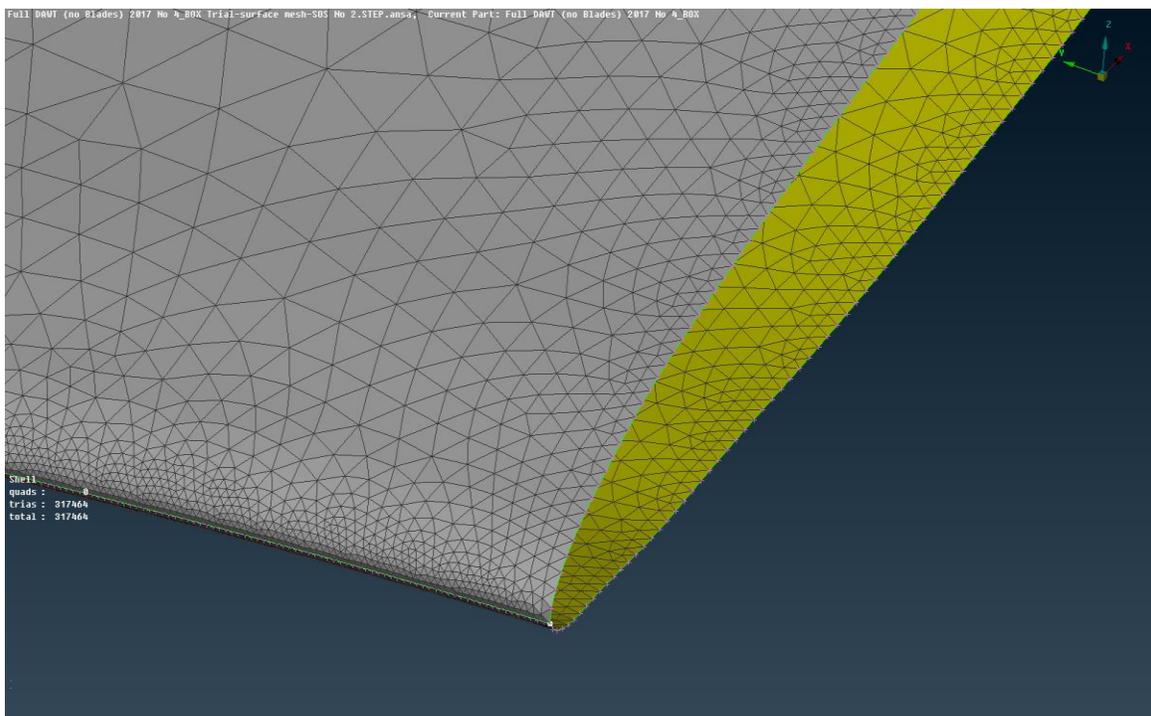


Figure 3.11: Close-up view of the internal flap's leading edge.

3.1.2.3 Central Column and Hub Mesh

The following parts to be meshed are the hub (excluding the nosecone), along with the central column. The mesh parameters used for both individual components are presented in Table 3.8. The surface meshes for the hub and central column geometries, are depicted in Figure 3.13.

Region	Central Column	Hub
Elements Type	Triangles	Triangles
2 nd Order	Yes	Yes
Interior Growth Rate	1.2	1.2
Distortion Angle	15	15
Minimum Target Length (mm)	20	20
Maximum Target Length (mm)	60	60

Table 3.8: The parameters for the surface mesh generation of the hub and central column.

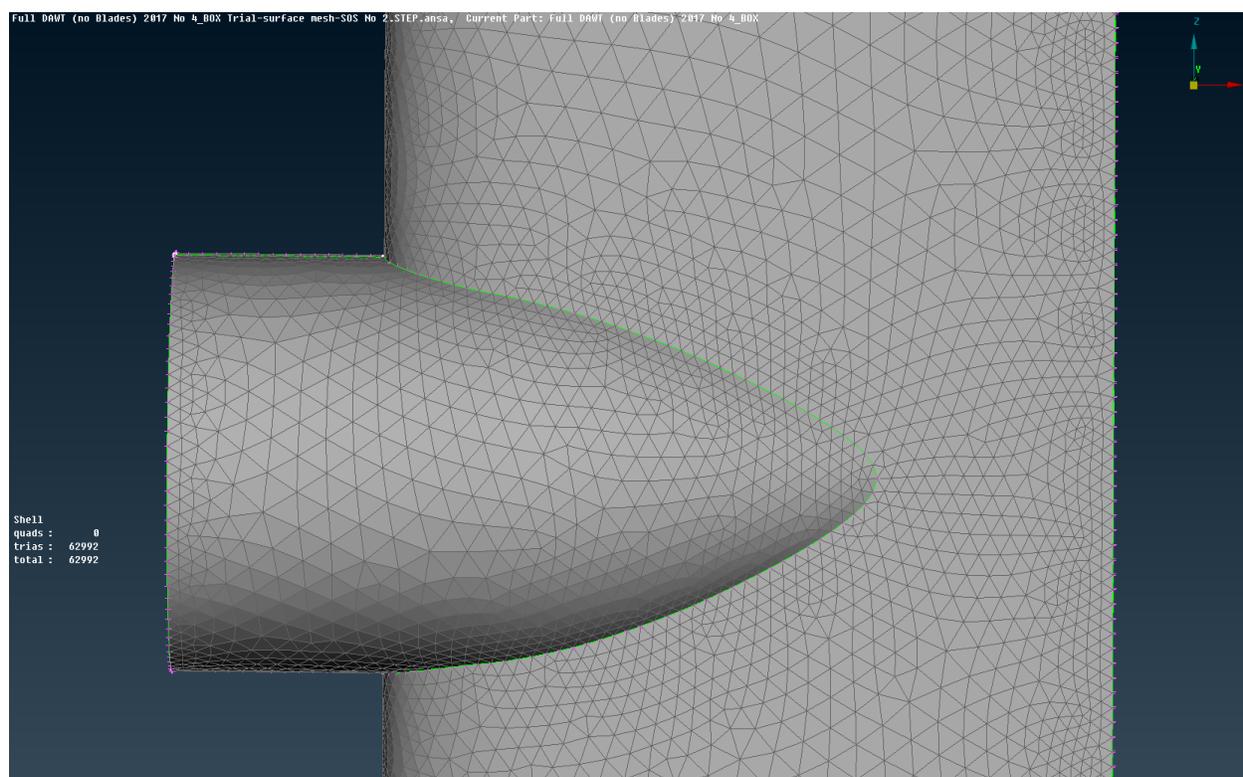


Figure 3.12: The surface mesh of the hub and central column.

3.1.2.4 Tower, Flow Domain (Box) and Cylinder Mesh

The last step in order to finish the DAWT's surface mesh, is to have the tower, the cylinder and the flow domain (box) meshed. The tower and the cylinder can be easily meshed, due to their simple geometry. The flow domain (box) can also be meshed in a simple way; however, in order to include the flow longwise the ground, the corresponding side must be meshed with respect to the layers generation which will take place in the following stage [Cha17]. The mesh parameters used for the tower and the box's bottom surface (ground) are similar, as presented in Table 3.9. The surface mesh for both geometries is shown in Figure 3.14, while the surface mesh of the DAWT components included in *Volume II* is shown in Figure 3.15.

Region	Tower	Cylinder	Bottom Surface	Box
Elements Type	Triangles	Triangles	Triangles	Triangles
2 nd Order	Yes	Yes	Yes	Yes
Interior Growth Rate	1.2	1.3	1.25	1.4
Distortion Angle	15	15	15	15
Minimum Target Length (mm)	20	20	20	400
Maximum Target Length (mm)	200	400	200	2000

Table 3.9: The parameters for the surface mesh generation of the tower and flow domain.

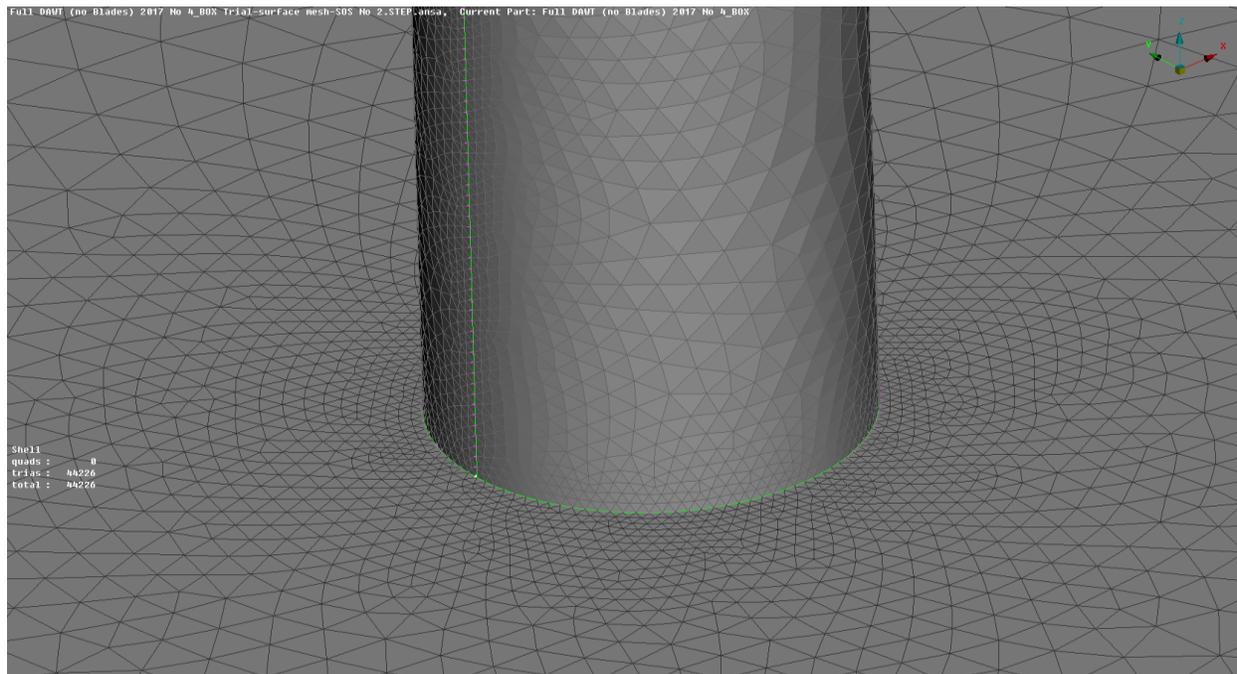


Figure 3.13: The surface mesh of the tower and the flow domain's bottom surface (ground).

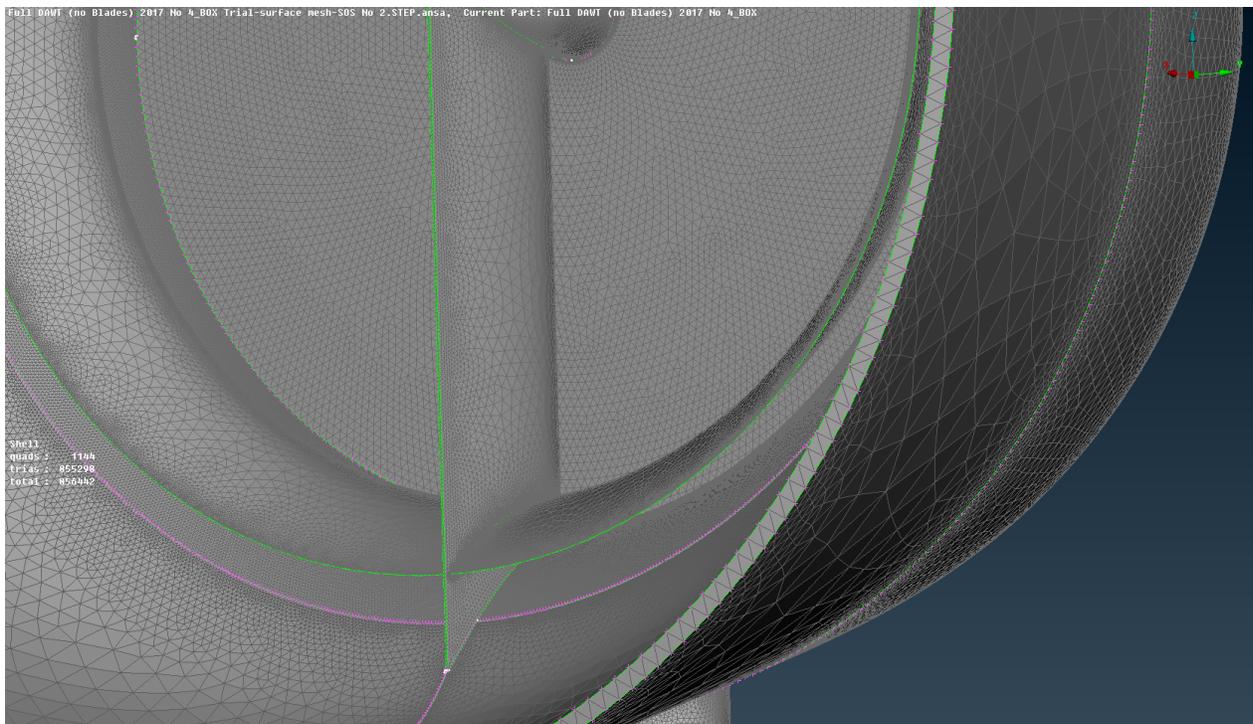
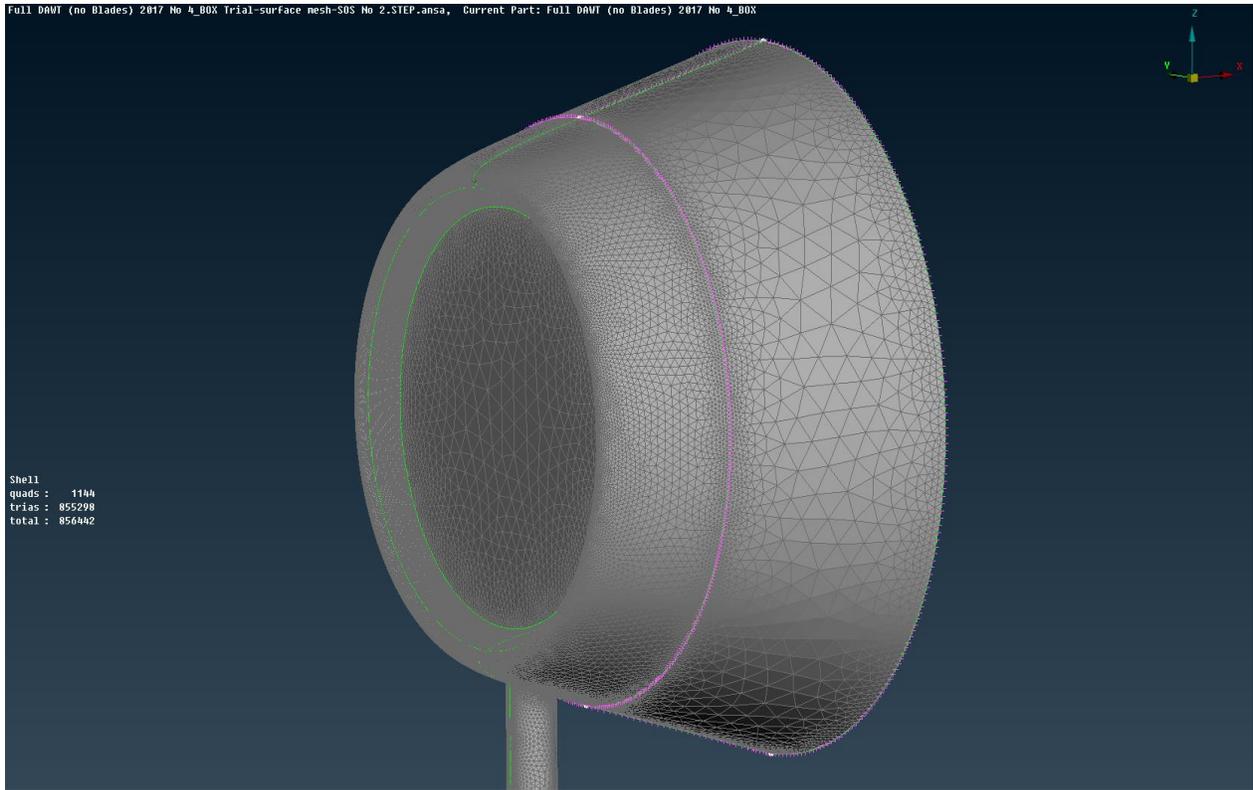


Figure 3.14: The surface mesh of the components included in Volume II.

3.2 Layers Construction

After the completion of the surface meshes construction, the second step is to create the layers of each component. The layers procedure deals with each volume separately. For the case of *Volume I*, where the blades are included, creating layers is not an easy task, due to the blade's geometrical complexity, whilst for the case of *Volume II*, layers can be generated easily for every single part. As mentioned in the beginning of this chapter, prismatic layers play a very important role in creating a volume mesh of good quality, for the effective simulation of the viscous effects close to the solid surfaces. Therefore, the choice of the number along with the total thickness of layers could have a great impact on the following CFD analysis [Cha17].

3.2.1 Volume I

3.2.1.1 Blades layers

The blades' prismatic layers are difficult to be created because of two reasons: the first has to do with the presence of a small gap between the blades' tip and the boundary of *Volume I*, which limits the total height of the corresponding layers; the second factor is that due to the blades' high curvature in some areas (leading and trailing edge), the creation of thick layers can lead to false results. Note that, in order to have a smooth transition without any discontinuities between the blades' regions and the nose cone, the layers of the particular geometries are generated through ANSA[®] *Batch Mode*, which allows the user to define and create layers for multiple geometries at the same time [Cha17]. The parameters used for the layers definition of each blade is presented in the following Table 3.10.

Blades Layers	
First height (Absolute)	<i>0.1 mm</i>
Growth Factor	<i>1.17</i>
Number of Layers	<i>7</i>
Total height (Additional layers not included)	<i>1.177201 mm</i>
Additional outer layers	<i>7</i>
Last Aspect	<i>0.6</i>
Squeeze	<i>Yes</i>
Collapse	<i>Yes</i>
Auto-connect to neigh. mesh	<i>Yes</i>
Auto-connect to neigh. Map-QUAD mesh	<i>Yes</i>

Table 3.10: The parameters for the blades layers construction.

As presented in Table 3.10, the parameters used for the definition of prismatic layers are the same for every blade region. At this point, it is important to mention that additional outer layers are very important, because they enhance the smooth development of layers and also define the size of the first tetrahedral element (shell). Notice that the total height of layers does not include the height of the additional outer layers [Cha17]. Additionally, the number of layers is equal to the number of additional outer layers, which makes a total number of layers equal to 14.

Finally, there are two more available features that should be mentioned, due to their high influence during the generation layers process. The first one is the *Squeeze* option, which allows the squeeze of layers in order to overcome intersection, proximity or quality problems [Ansa]. The second one is the *Collapse* option, which allows the collapse of certain local areas that cause intersection, proximity or quality issues, resulting in the growth of gradually less layers around these areas [Ansa]. The blades' prismatic layers are shown in the following figures:

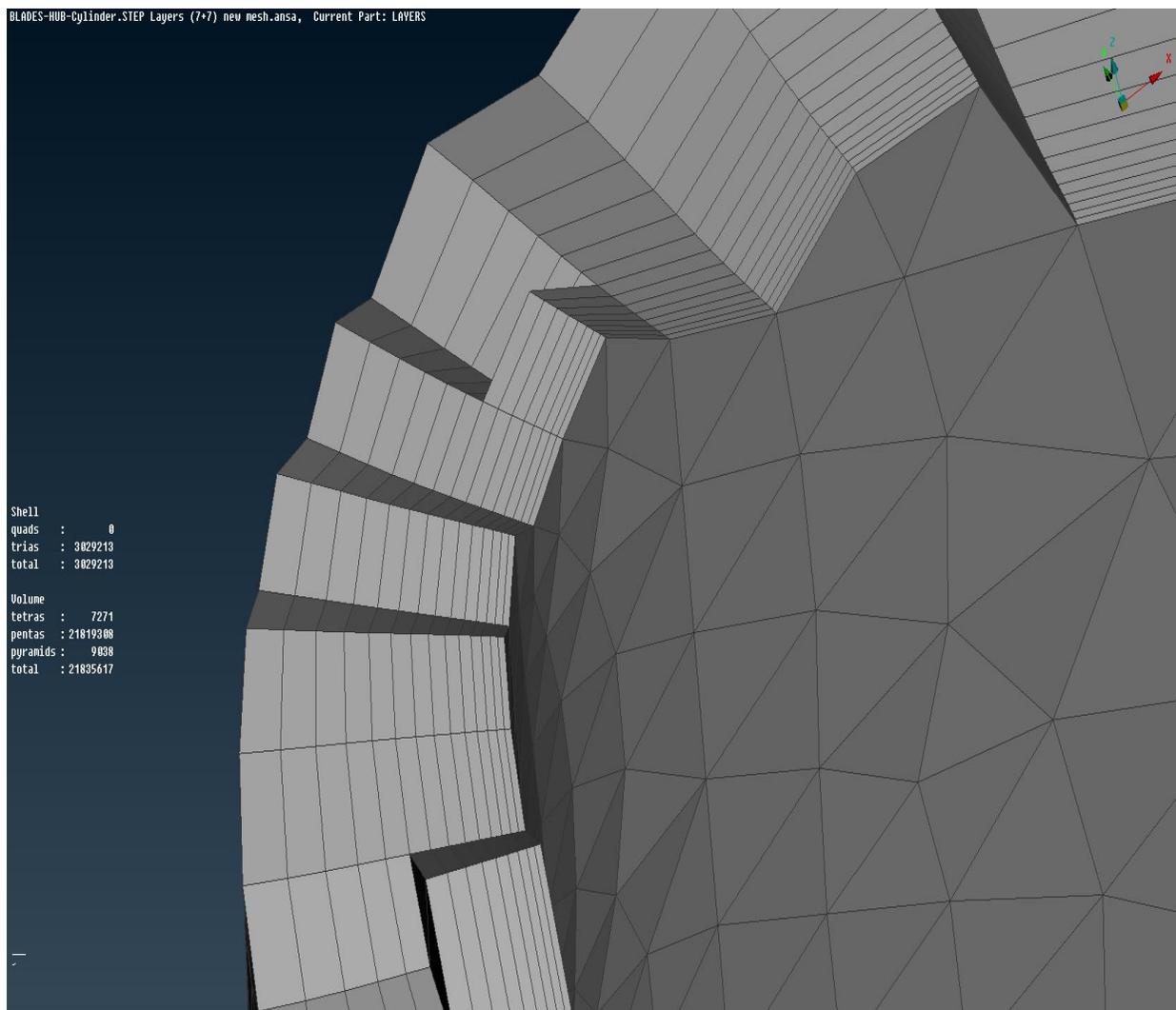


Figure 3.15: Illustration of the generated prismatic layers near the blade's leading edge.

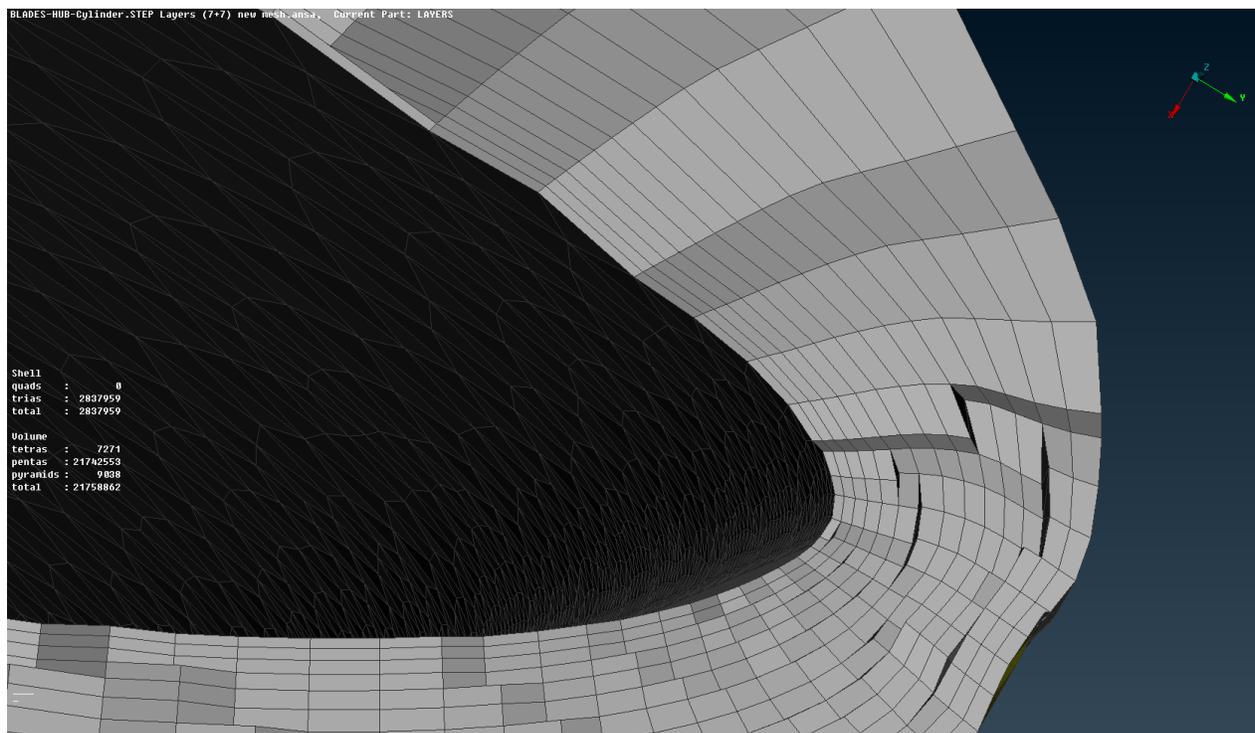
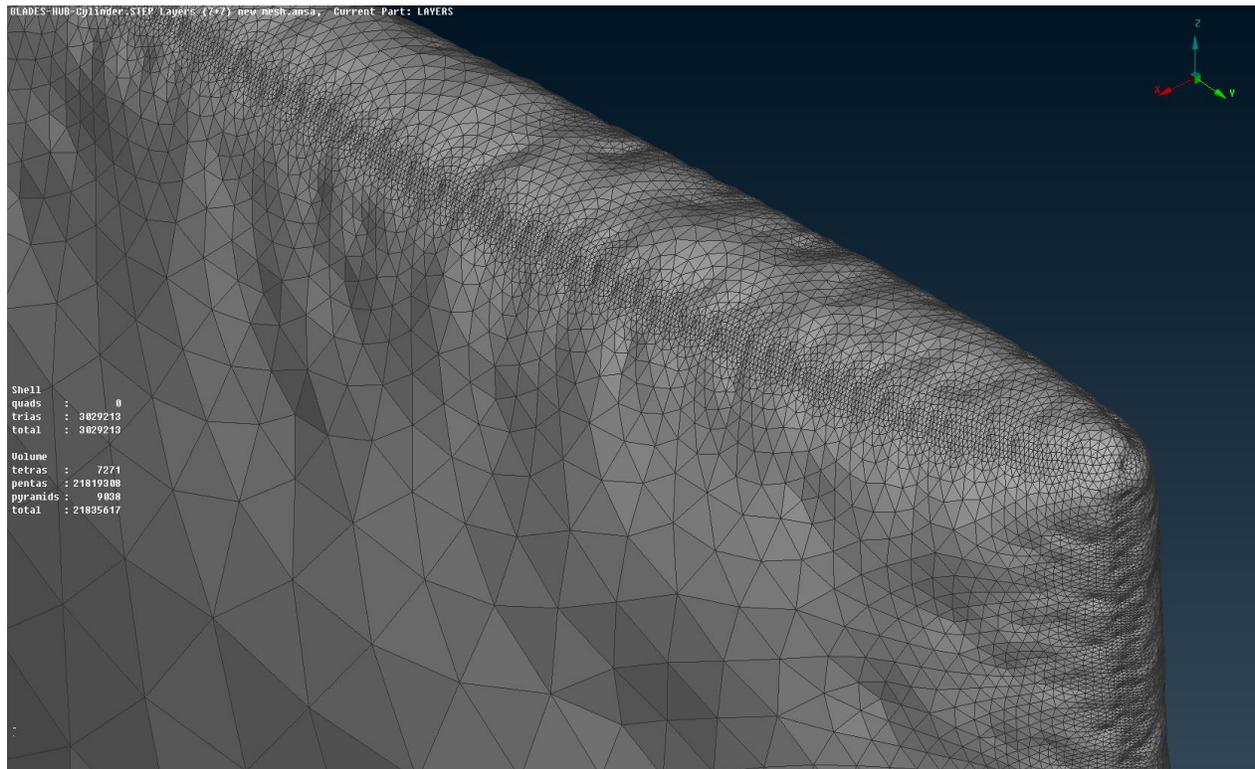


Figure 3.16: Illustration of the generated prismatic layers at the blade tip region.

3.2.1.2 Nose Cone Prismatic Layers Construction

Having completed with the layers development for the blades, the following step is to create the prismatic layers for the nose cone. As mentioned above, the layers generation takes place at the same time in order to have a smooth transition between the prismatic layers of the blades and those of the nose cone [Cha17]. This is succeeded by defining the corresponding parameters through the use of *Batch Mode*. As a result, the parameters used for the nose cone are similar to the ones used for the blades, as presented in Table 3.11.

Nose Cone Layer	
First height (Absolute)	<i>0.3 mm</i>
Growth Factor	<i>1.25</i>
Number of Layers	<i>7</i>
Additional outer layers	<i>7</i>
Last Aspect	<i>0.6</i>
Squeeze	<i>Yes</i>
Collapse	<i>Yes</i>
Auto-connect to neigh. mesh	<i>Yes</i>
Auto-connect to neigh. Map-QUAD mesh	<i>Yes</i>

Table 3.11: The parameters for the nose cone layer construction.

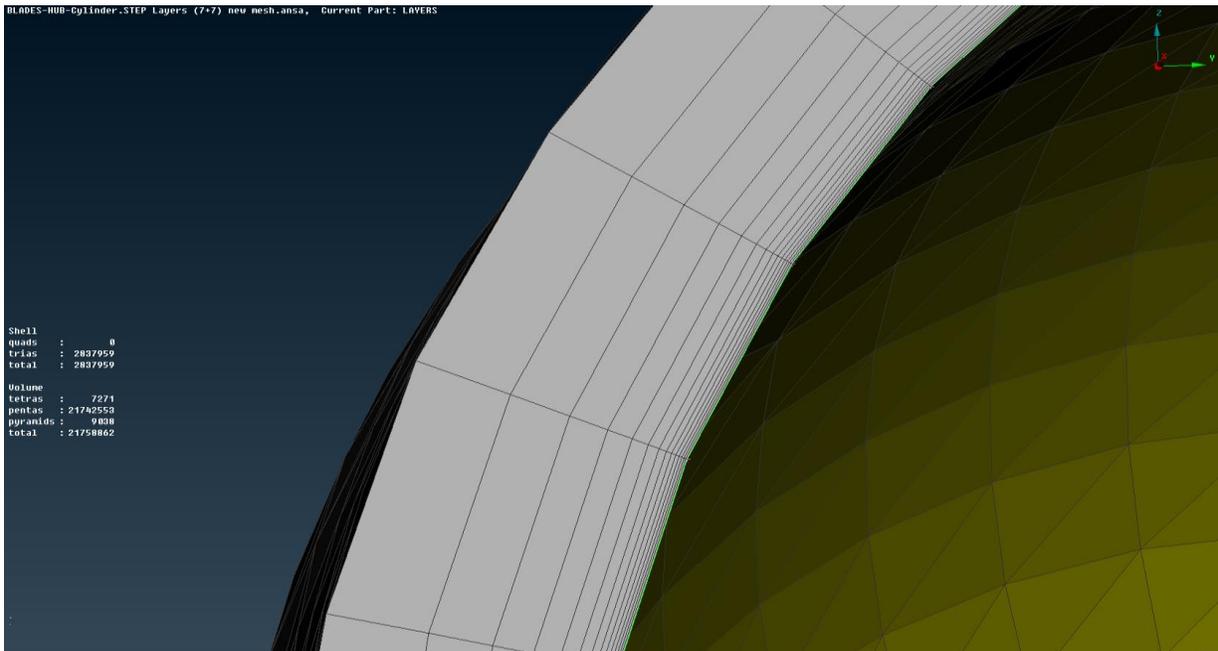


Figure 3.17: Illustration of the generated prismatic layers around the nose cone.

3.2.2 Volume II

The methodology used to create the layers around the geometries included in *Volume II* is similar to the one followed in the case of *Volume I*. The CFD parameters definition along with the layers generation for each individual part takes place through the use of *Batch Mode*. However, in contrast to the layers around the parts of *Volume I*, the layers around the geometries of *Volume II* are much thicker, not only due to the fact that the particular geometries are much simpler, but also because of their bigger size [Cha17].

3.2.2.1 Diffuser Layers

Before defining the diffuser's layers, the gap between the diffuser and the rotating cylinder (*Volume I*) is calculated. Furthermore, the gap between the diffuser and the internal flap needs to be considered, as well. Therefore, the layers around the diffuser geometry are defined based on the parameters presented in Table 3.12. The generated layers around the diffuser are depicted in Figure 3.19.

Diffuser Layers	
First height (Absolute)	<i>1 mm</i>
Growth Factor	<i>1.18</i>
Number of Layers	<i>7</i>
Total height (Additional layers not included)	<i>11.77201 mm</i>
Additional outer layers	<i>7</i>
Last Aspect	<i>0.6</i>
Squeeze	<i>Yes</i>
Collapse	<i>Yes</i>
Auto-connect to neigh. mesh	<i>Yes</i>
Auto-connect to neigh. Map-QUAD mesh	<i>Yes</i>

Table 3.12: The parameters for the diffuser prismatic layers construction.

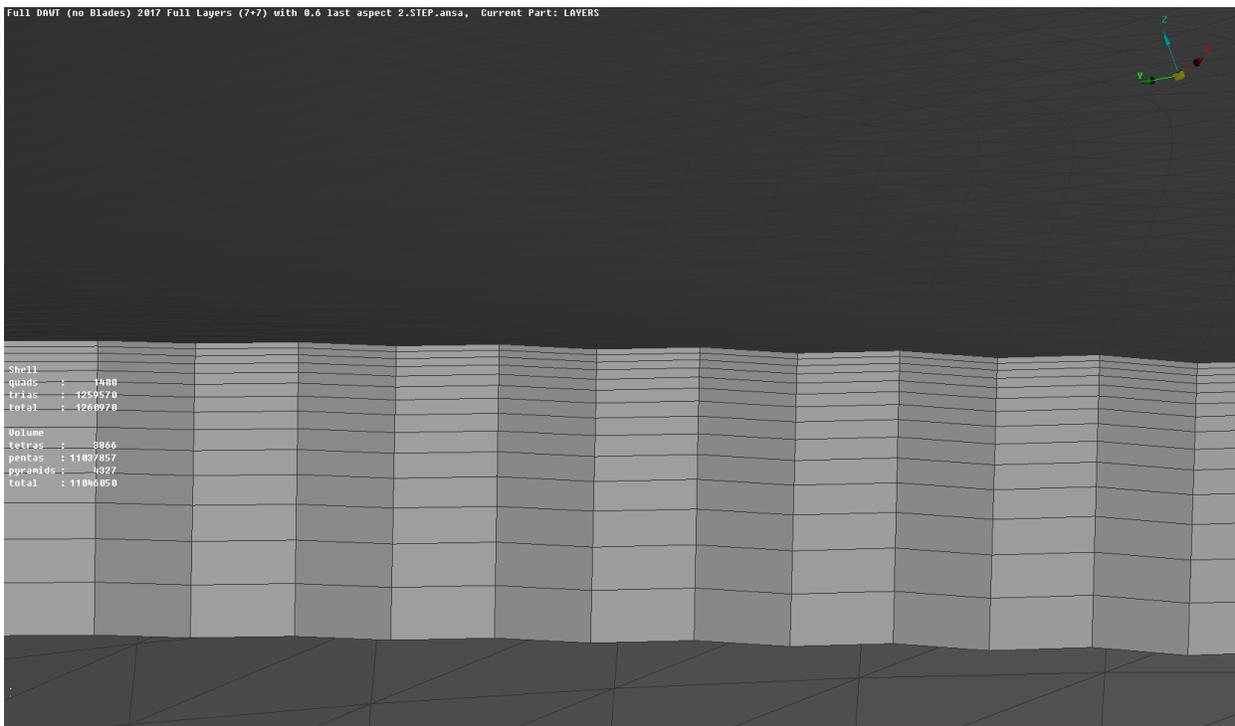
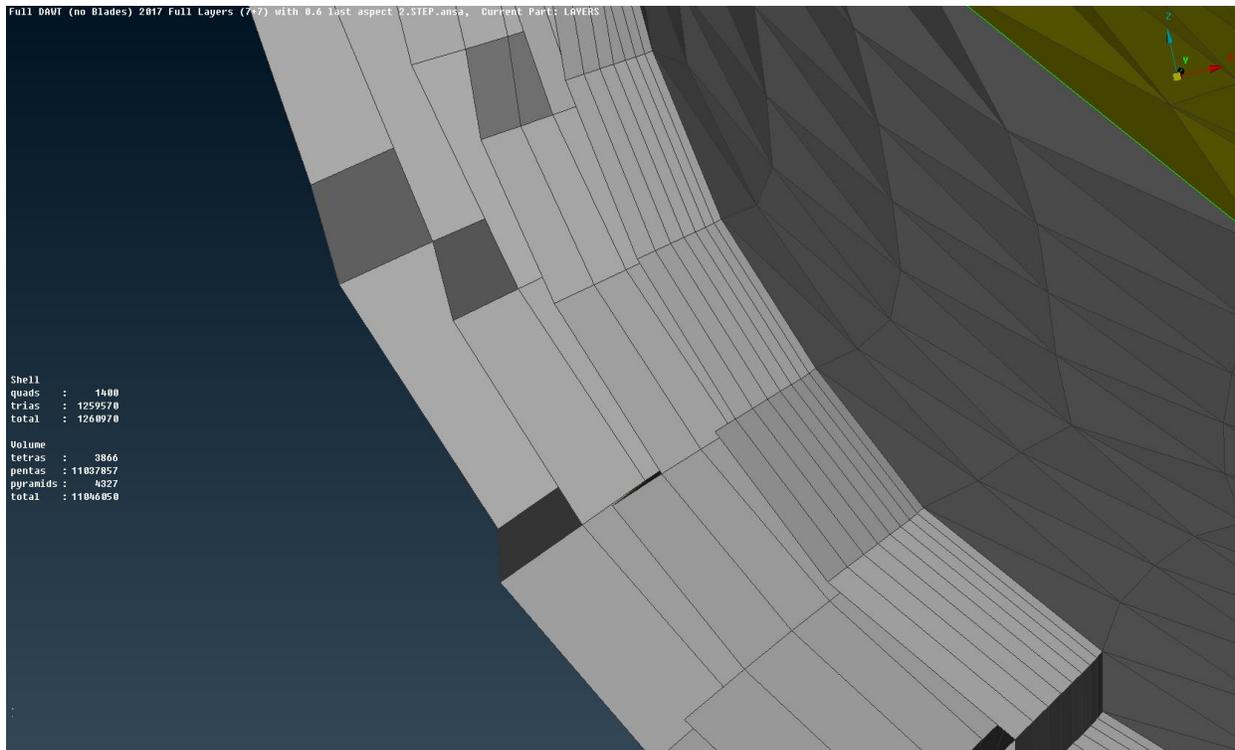


Figure 3.18: The generated prismatic layers at the diffuser's leading edge (up) and middle surface (down).

3.2.2.2 Internal Flap, Central Column and Hub Layers

The same parameters used for the layers construction around the diffuser are used in order to develop the prismatic layers of the internal flap, central column and hub geometries, after their addition to the *Batch Mode* process. In the following Fig. 3.20 and Fig. 3.21, the layers growth of each individual component is demonstrated.

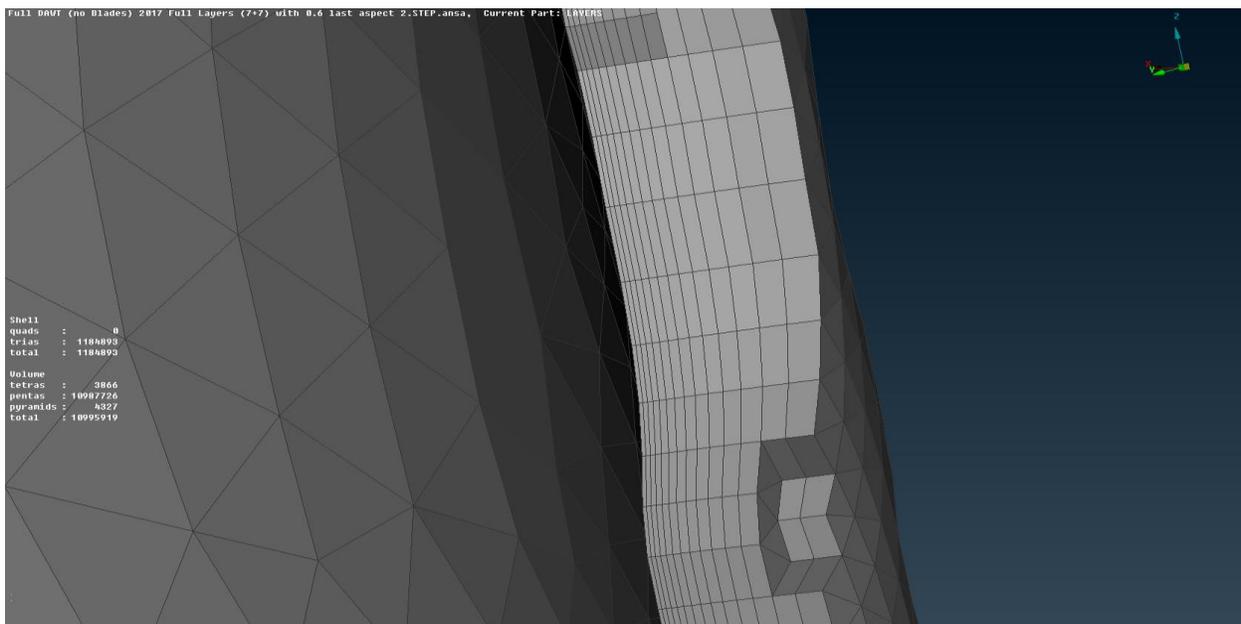
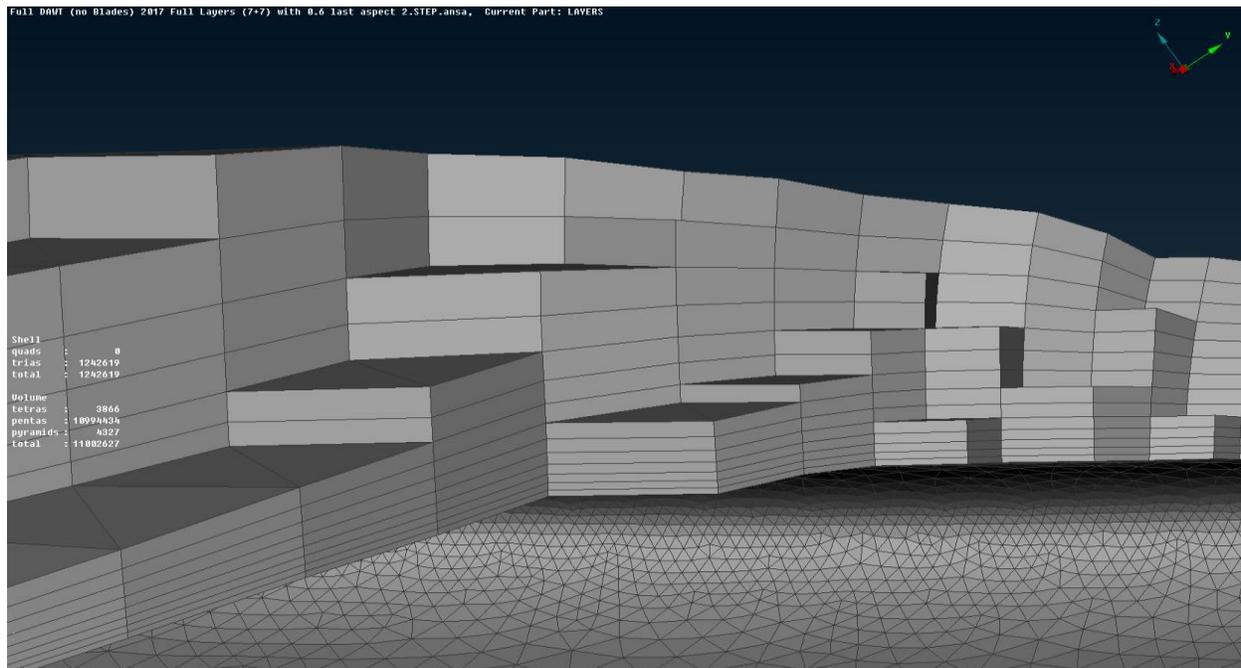


Figure 3.19: The generated prismatic layers around the internal flap (up) and the central column (down).

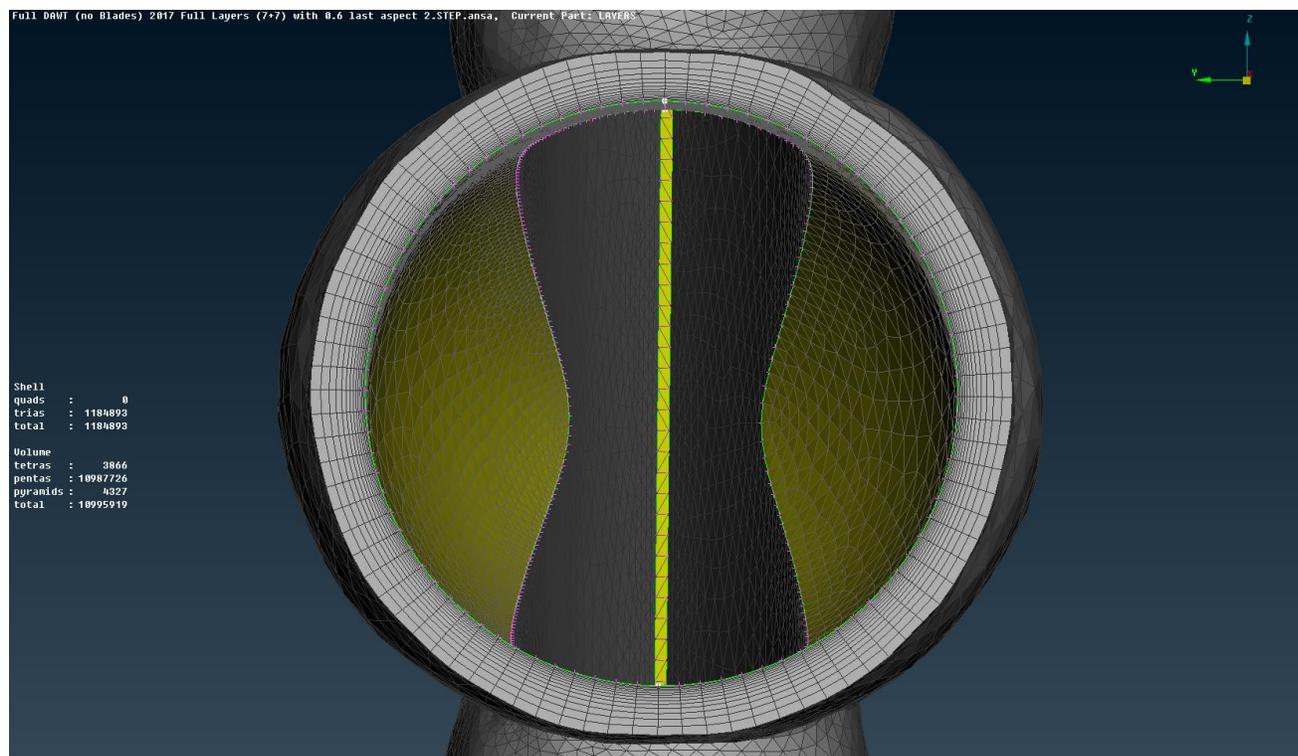


Figure 3.20: Illustration of the generated prismatic layers around the hub.

3.2.2.3 Tower and Flow Domain (Box) Layers

The last step in order to complete the prismatic layers generation of *Volume II* is to produce the layers for the tower and the flow domain (box). The tower and the bottom surface of the flow domain (ground) have been meshed with larger surface elements; therefore the parameters used for the definition of the respective layers are different, compared to those used for the rest components of *Volume II*. Additionally, the prismatic layers generation procedure takes place in *Batch Mode*, along with all the other geometries of *Volume II*, as explained before. The two main differences used for the prismatic layers definition in this case are the increased first height, as well as the slightly increased growth rate, as presented in the Table 3.13. The growth rate of the layers corresponding to the tower and the bottom surface of the flow domain are shown in Figure 3.22.

Tower and Flow Domain Layers	
First height (Absolute)	1.5 mm
Growth Factor	1.2

Table 3.13: The parameters for layers construction of the tower and the flow domain's bottom surface.

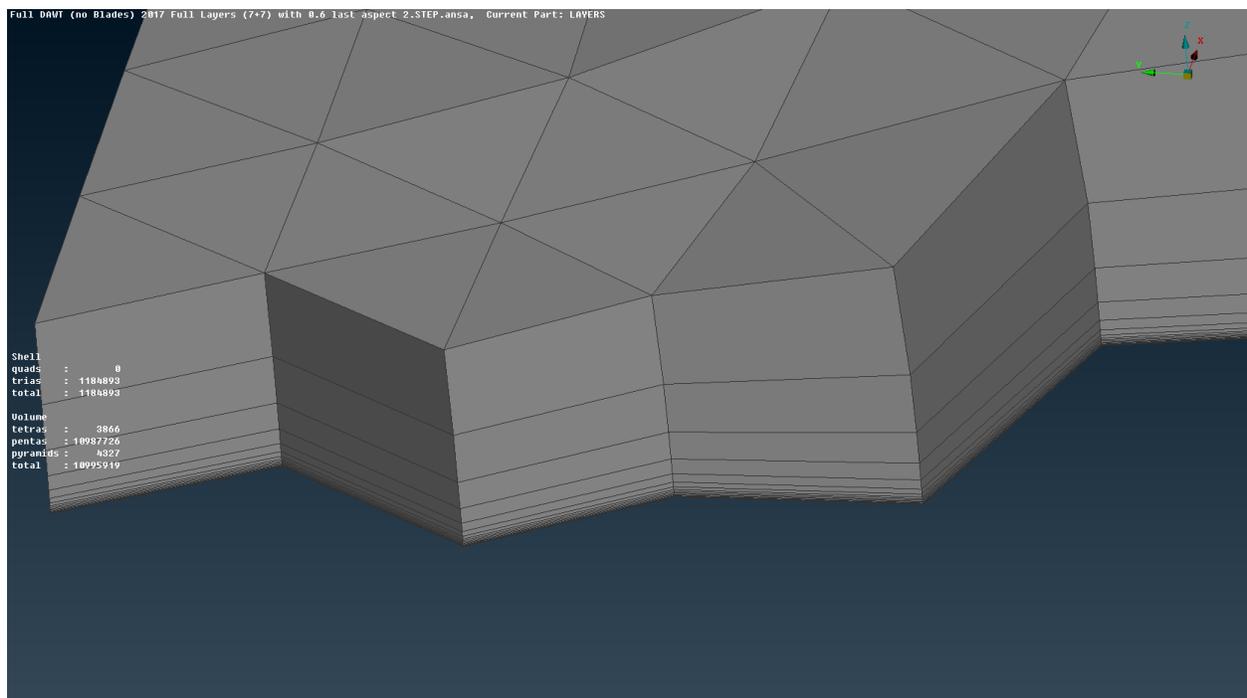
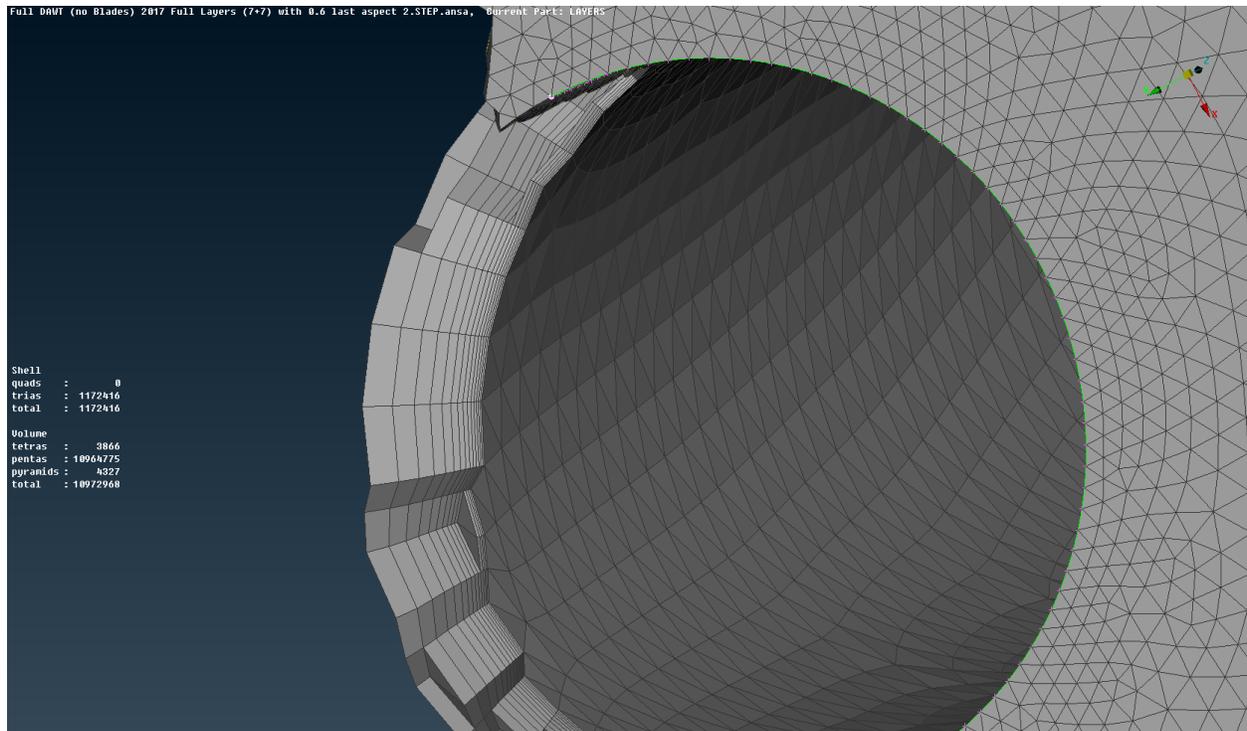


Figure 3.21: The generated prismatic layers of the tower (up) and the flow domain's bottom surface (down).

3.3 Volume Mesh

After the integration of the layers growth, the final stage is to generate the volume mesh for each independent volume. The particular procedure also takes place in *Batch Mode* as in the case of layers generation, enabling the parallel development of both volumes. Eventually, the volumes are produced with the use of *Tetra Rapid* algorithm, which uses tetrahedral elements and pyramids [Cha17, Ansa].

3.3.1 Volume I

As noticed before, *Volume I* is characterized by a large number of surface elements due to its geometries' complexity (blades). In order to avoid a rapid development of shells along the blades, the process of volume creation needs to be done in a slow growth rate. Notice also that for the volume generation there is no minimum shell size option as in the surface mesh case. However, the definition of maximum length is required otherwise ANSA[®] generates shells as large as possible, considering only the growth rate [Cha17]. The volume parameters are presented in the following table, while a sample volume mesh is depicted in Figure 3.23 and Figure 3.24.

Volume I - Mesh Parameters	
Maximum Growth Rate	<i>1.1</i>
Maximum Length	<i>175 mm</i>
Criterion	<i>FLUENT Skewness</i>
Frozen Entities	<i>0</i>
Create Pyramids	<i>Yes</i>
Total Size (Shells)	<i>40,012,464</i>

Table 3.14: The parameters for the volume mesh construction of Volume I.

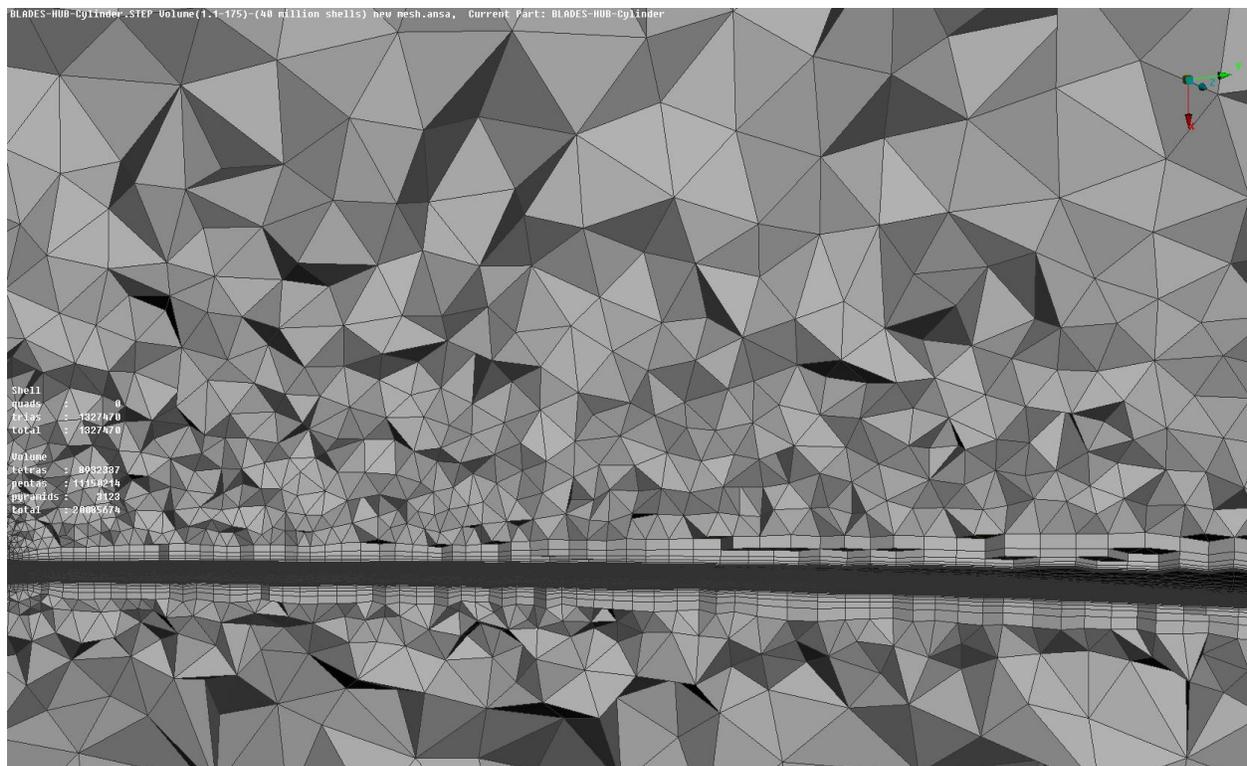
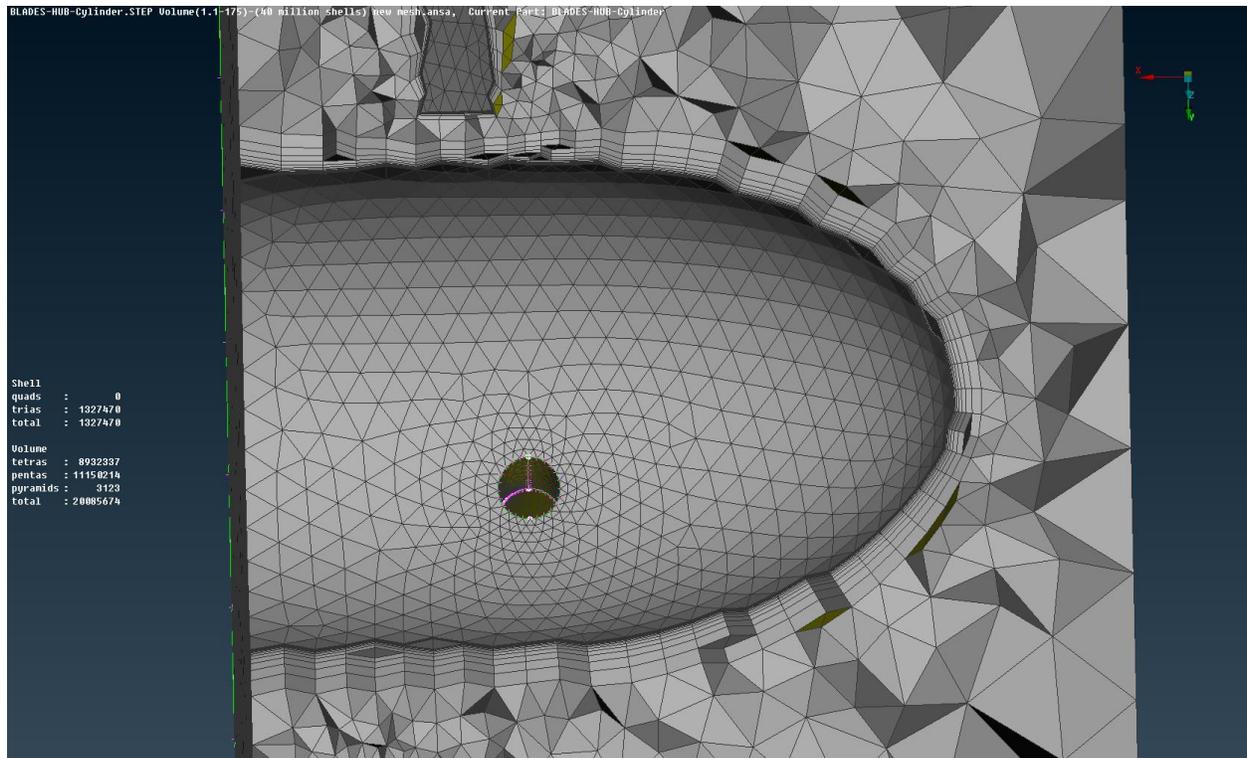


Figure 3.22: Illustration of the volume mesh around the nose cone (up) and lengthwise the blade (down).

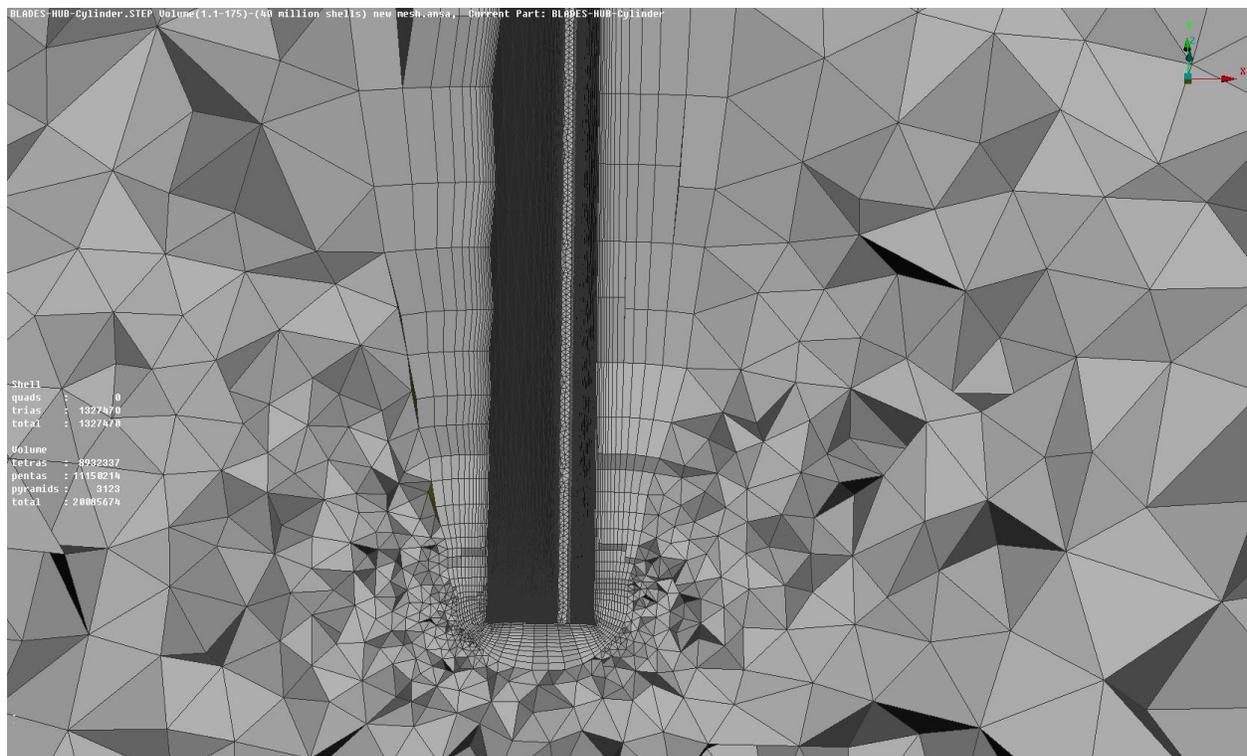
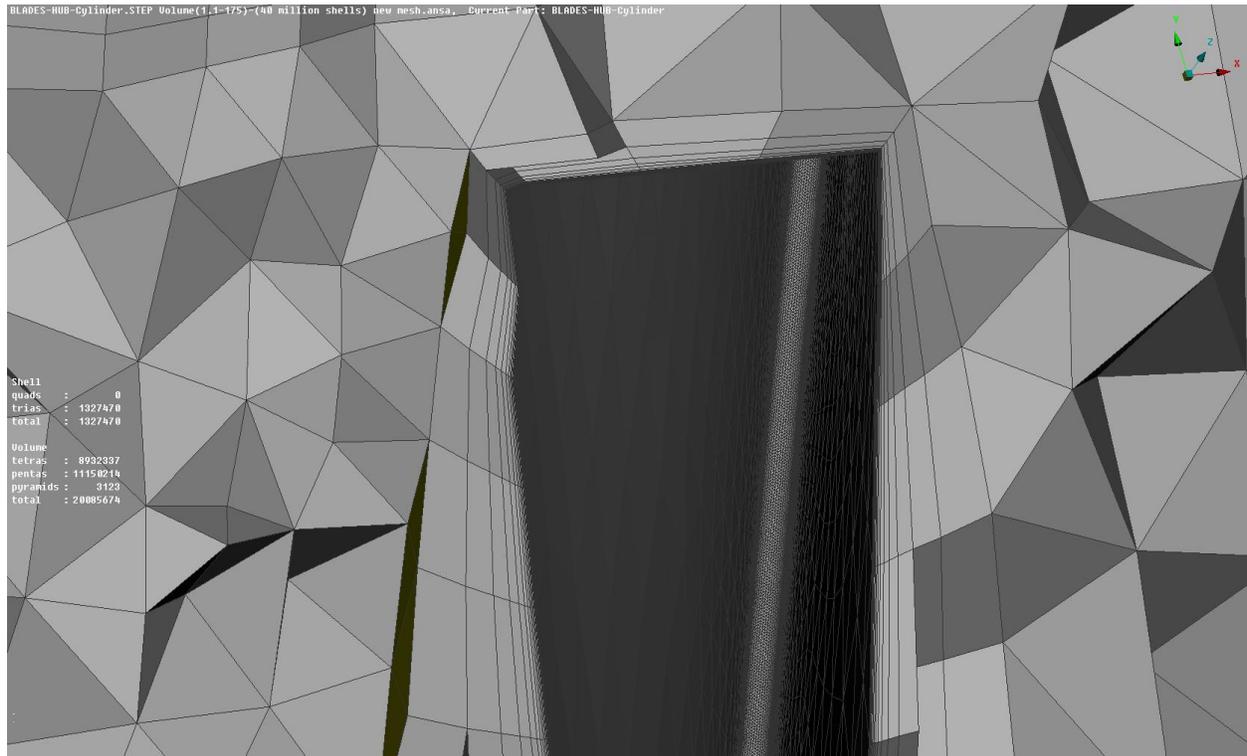


Figure 3.23: Close-up views of the volume mesh around the blade root (up) and near the blade tip (down).

3.3.2 Volume II

As in the case of *Volume I*, the procedure followed for *Volume II* takes place with the use of *Batch Mode*. In contrast to the first volume, here the surface elements are much larger; therefore a higher growth rate can be applied. However, the large size of the domain box can lead to an excessively large number of shells, which would render the CFD analysis in the next stage even slower. As a result, the choice of the growth rate must be done very carefully. The table below presents the options used for the definition of *Volume II*, whereas in Figure 3.25-3.27 the results of the volume generation are shown [Cha17]. Finally, Table 3.16 contains the final statistics of the two individual grids (*Volume I* and *II*).

Volume II – Mesh Parameters	
Maximum Growth Rate	1.18
Maximum Length	2500 mm
Criterion	<i>FLUENT Skewness</i>
Frozen Entities	0
Create Pyramids	Yes
Total Size (Shells)	25,322,474

Table 3.15: The parameters for the volume mesh construction of Volume II.

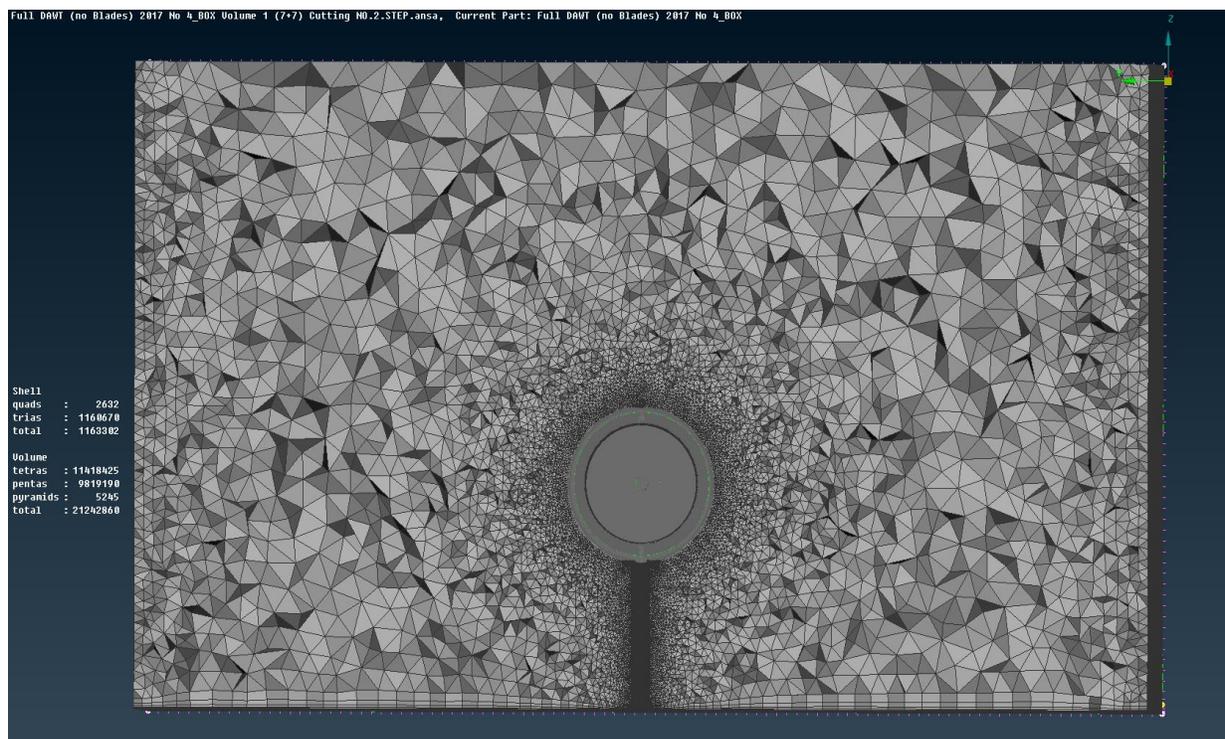


Figure 3.24: Illustration of the volume mesh around the DAWT

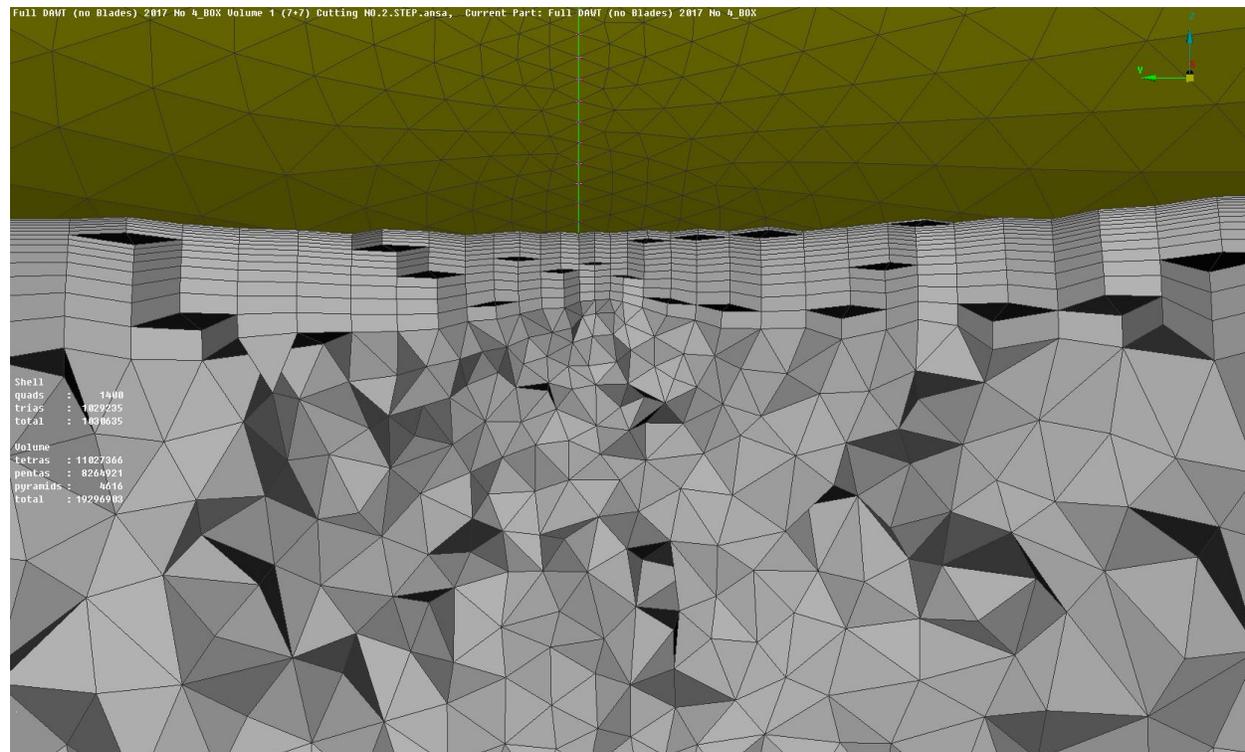
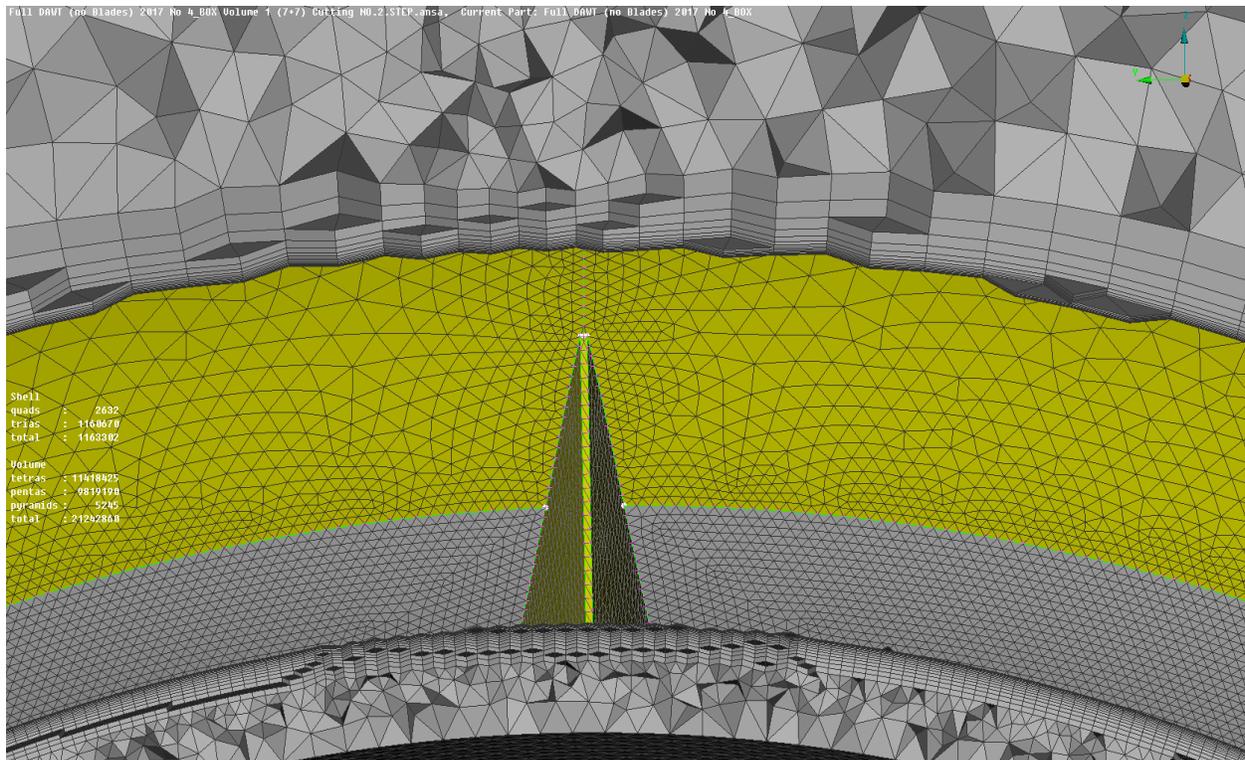


Figure 3.25: Illustration of the volume mesh around the diffuser.

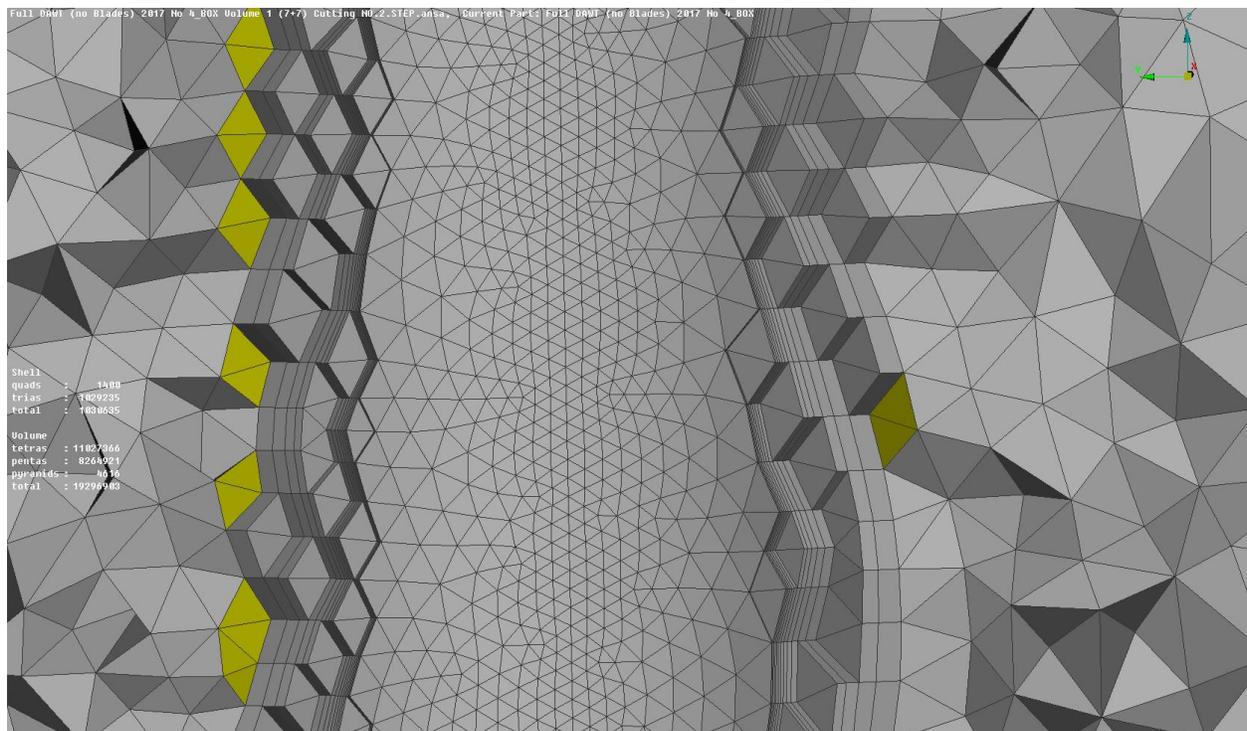
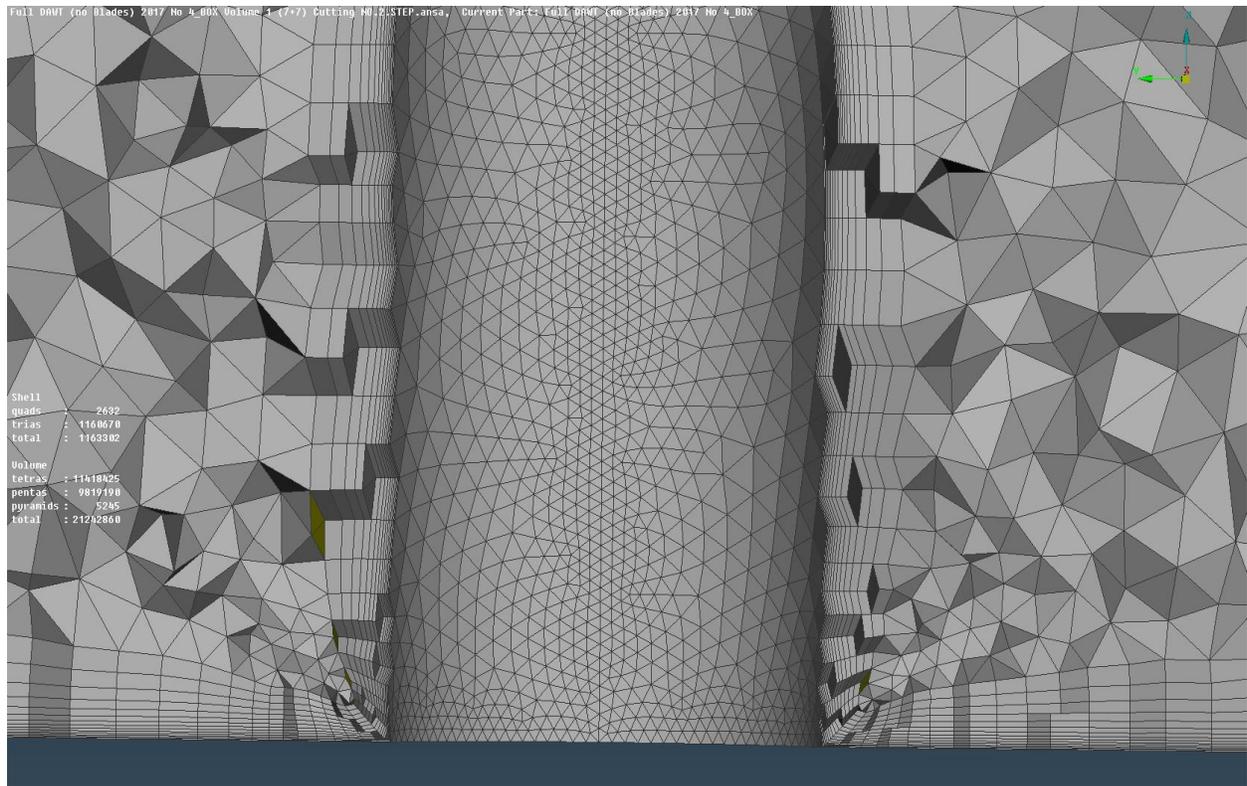


Figure 3.26: Illustration of the volume mesh near the tower's root (up) and along the tower (down).

	<i>Volume I</i>	<i>Volume II</i>
Quads	9072	532
Triangles	1,748,864	3,162,570
Total (surface)	1,757,936	3,163,102
Total number of nodes	8,209,774	14,327,081
Tetras	13,997,346	18,148,702
Prisms	11,320,283	21,854,724
Pyramids	5245	9038
TOTAL (volume)	25,322,874	40,012,464

Table 3.16: Final Mesh Statistics.

4. Conclusions and Recommendations

In this work an algorithm for the parametric design and modeling of Diffuser Augmented Wind Turbines (DAWTs) was developed. The algorithm allows the user to define some of the main parameters that characterize a modern wind turbine resulting in an integrated DAWT system. The specific algorithm was developed in Grasshopper®, a graphical algorithm editor tightly integrated into Rhinoceros 3D CAD application. The drawing tools in Grasshopper® and the way they cooperate in order to achieve the final result give to the user an algorithmic way of thinking although without requiring any programming or scripting skills. Grasshopper® offers the user a friendlier graphical interface, allowing for the construction of the main parts composing a complete DAWT system, including the turbine blades, the centrebody, the diffuser geometry, as well as the internal flap, the tower and the central column structures.

As long as the parametric design procedure is concerned, Grasshopper® offers the user a variety of options and tools in order to manage and process the data through a more clear and effective way. For example, the fact that the data lists are managed as data trees, offers the user the flexibility and speed to develop the algorithm, as there are several commands for their proper formation and organization. Finally, it should be mentioned that during the development of the algorithm neither additional applications nor plug-ins were used, nor any additional design tools built to replace the existing Grasshopper tools, in order to overcome their bad outcomes.

However, during the construction of the algorithm, many times Grasshopper® showed a weakness to perform certain commands, while other times the results were not as expected. Particularly, regarding some specific drawing tools (Boolean operations), the operation was problematic, leading to the development of alternative methods for the realization of this work. The failure of these tools was pivotal for the way this algorithm was developed, since the available alternative design tools in Grasshopper® were either limited or inadequate.

A recommendation for the extension of the algorithm in the future is the definition and addition of a surface cap in order to “fill” the blade tip. Furthermore, a useful addition would be the definition and creation of layers along the DAWT’s internal surface, in order to separate the various materials of construction which form the internal structure of a wind turbine, including the blades. As a result, the user will be able to set different properties for each different layer of material, resulting in a more realistic analysis of the behavior of the blade when combined with finite element analysis software. Additionally, the scaling and translation features could be extended to other DAWT geometries, besides the diffuser and the internal flap, allowing the user to transform and move each part separately in order to create integrated DAWTs of various sizes located in different 3D positions. Eventually, the constructed mesh could be used in order to run CFD analyses of the flow inside and around the DAWT for different conditions. Such analyses can be conducted with a rotating rotor, taking, thus, into account the real effect of the rotor on the flow inside and around the DAWT. This can be easily achieved, as the computational mesh is divided in two parts, one stationary and the other one (around the rotor) capable to be a rotating one.

One last recommendation, in order to upgrade the presented algorithm, is the definition and the reproduction of a prescribed number of DAWTs, resulting in the construction of a small wind farm. Therefore, the user will be able to design and create more than one DAWTs, as well as change their 3D positions. The user will also have the ability to rotate each independent DAWT according to the wind direction. The whole procedure will be fully parametric and representative

of the current commercial DAWT designs and demands, enabling for the construction of a small wind farm.

Bibliography

[Abe04] Abe, K. and Ohya, Y., 2004, “An Investigation of Flow Fields Around Flanged Diffusers Using CFD”, *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 92, pp. 315–330.

[Ansa] BETA CAE Systems, <http://www.beta-cae.com/ansa.htm>

[Ara15] Aranake, A. C., Lakshminarayan, V. K. and Duraisamy, K., 2015, “Computational Analysis of Shrouded Wind Turbine Configurations Using a 3-Dimensional RANS Solver”, *Renewable Energy*, Vol. 75, pp. 818–832.

[Bet03] Bet, F. and Grassmann, H., 2013, “Upgrading Conventional Wind Turbines”, *Renewable Energy*, Vol. 28, pp. 71–78.

[Bus98a] van Bussel, G. J. W., 1998, “Power Augmentation Principles for Wind Turbines”, The World Directory of Renewable Energy.

[Bus98b] van Bussel, G. J. W., 1998, “Development of a Momentum Theory for DAWT”, Mie University, Japan.

[Bus07] van Bussel, G. J. W., 2007, “The Science of Making More Torque from Wind: Diffuser Experiments and Theory Revisited”, *Journal of Physics: Conference Series*, Vol. 75.

[Car14] Carroll, J., 2014, “Diffuser Augmented Wind Turbine Analysis Code”, M.Sc. Thesis, Department of Aerospace Engineering, University of Kansas, Kansas.

[Cha14] Charalampous, K., 2014, “Wind Turbine Blades Parametric Design”, Diploma Thesis, School of Production Engineering and Management, Technical University of Crete, Chania.

[Cha15] Charalampous, K. G., Strofylas, G.A., Mazanakis, G.I. and Nikolos, I.K., 2015 “Wind Turbine Blade Parametric Design Using Grasshopper”, *8th GRACM International Conference on Computational Mechanics*, 12-15 July, Volos, Greece.

[Cha17] Charalampous, K. G., Leloudas, S. N. and I. K. Nikolos, “*Generation of a 3D Computational Mesh for the Flow Simulation Around a DAWT*”, OGT GreenTech Intermediate Report – 6, March 2017, Chania, Greece.

[Dor11] Van Dorst, F. A., 2011, “An Improved Rotor design for a Diffuser Augmented Wind Turbine”, M.Sc. Thesis, Delft University of Technology, Delft.

[Flod] FloDesign Wind Turbines, <http://wattnow.org/1891/flodesign-wind-turbine-theres-change-in-the-wind>.

[For83] Foreman, K. M. and Gilbert, B. L., 1983, “A Free Jet Wind Tunnel Investigation of DAWT Models”, Grumman Research Dept. Report RE-668, SERI/TR 01311-1, New York.

- [Gil78] Gilbert, B. L., Oman, R. A. and Foreman, K. M., 1978, "Fluid Dynamics of Diffuser-Augmented Wind Turbines", *Journal of Energy*, Vol. 2, No. 6, pp. 368–374.
- [Gil79] Gilbert, B. L. and Foreman, K. M., 1979, "Experimental Demonstration of the Diffuser-Augmented Wind Turbine Concept", *Journal of Energy*, Vol. 3, No. 4, pp. 235–240.
- [Gil83] Gilbert, B. L. and Foreman, K. M., 1983, "Experiments With a Diffuser-Augmented Model Wind Turbine", *Journal of Energy Resources Technology*, Vol. 105, pp. 46–53.
- [Gra03] Grassmann, H., Bet, F., Cabras, G., Ceschia, M., Cobai, D. and DelPapa, C., 2003, "A Partially Static Turbine - First Experimental Results", *Renewable Energy*, Vol. 28, pp. 1779–1785.
- [Gra3D] Robert McNeel & Associates, Grasshopper™, <http://www.grasshopper3d.com>.
- [Gur13] Gurit, Wind Energy Handbook, <http://www.gurit.com/>.
- [Han00] Hansen, M. O. L., Sørensen, N. N. and Flay, R. G. J., 2000, "Effect of Placing a Diffuser around a Wind Turbine", *Wind Energy*, Vol. 3, No. 4, pp. 207–213.
- [Hjo15] Hjort, S. and Larsen, H., 2015, "Rotor Design for Diffuser Augmented Wind Turbines", *Energies*, Vol. 8, No. 10, pp. 10736–10774.
- [Hol81] Van Holten, T., 1981, "Concentrator Systems for Wind Energy, with Emphasis on Tipvanes", *Wind Engineering*, Vol. 5, No. 1, pp. 29-45.
- [Hoo09] ten Hoopen, P.D.C., 2009, "An Experimental and Computational Investigation of a Diffuser Augmented Wind Turbine", M.Sc. Thesis, Faculty of Aerospace Engineering, Delft University of Technology, Delft.
- [Igr76] Igra, O., 1976, "Shrouds for Aerogenerators", *AIAA Journal*, Vol. 14, pp. 1481–1483.
- [Igr77] Igra, O., 1977, "Compact Shrouds for Wind Turbines", *Energy Conversion*, Vol. 16, pp. 149–157.
- [Igr81] Igra, O., 1981 "Research and Development for Shrouded Wind Turbines", *Energy Conversion and Management*, Vol. 21, pp. 13–48.
- [Iwa53] Iwasaki, M., *The Experimental and Theoretical Investigations of Windmills*, Kyushu University, Research Institute for Applied Mechanics, 1953.
- [Jaf14] Jafari, S. A. H. and Kosasih, B., 2014, "Flow Analysis of Shrouded Small Wind Turbine with a Simple Frustum Diffuser with Computational Fluid Dynamics Simulations", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 125, pp. 102–110.
- [Kog62] Kogan, A. and Nissim, E., 1962, "Shrouded Aerogenerator Design Study, Two-Dimensional Shroud Performance", *Bulletin of the Research Council of Israel*, pp. 67-88.
- [Kog63a] Kogan, A. and Seginer, A., 1963, "Shrouded Aerogenerator Design Study II, Axisymmetrical Shroud Performance", *Fifth Israel Annual Conference an Aviation and Astronautics*, Israel.

[Kog63b] Kogan, A. and Seginer, A., 1963, “Shrouded Aerogenerator, Design Study II, Axisymmetric Shroud Performance”, Dept. of Aeronautical Engineering, Technion.

[Lil56] Lilley, G.M. and Rainbird, W.J., 1956, “A Preliminary Report on the Design and Performance of Ducted Windmills”, CoA Report No. 102, The College of Aeronautics, Cranfield, p. 72.

[Lel16] Leloudas, S. N. and Nikolos I. K., "A Literature Review on the Diffuser Augmented Wind Turbine Concept", OGT GreenTech Intermediate Report – 2, Chania, Greece, February 2016.

[Ohy10] Ohya, Y. and Karasudani, T., 2010, “A Shrouded Wind Turbine Generating High Output Power with Wind-lens Technology”, *Energies*, Vol. 3, pp. 634–649.

[Oma77] Oman, R. A., Foreman, K. M. and Gilbert, B. L., 1977, “Investigation of Diffuser-Augmented Wind Turbines, Parts I & II”, Grumman Research Department, Report RE-534, ERDA Report C00-2616-2, New York.

[Phi03] Phillips, D. G., 2003, “An Investigation on Diffuser Augmented Wind Turbine Design”, Ph.D. Thesis, University of Auckland.

[Pie97] Piegl, L. and Tiller, W., 1997, *The NURBS Book*, Springer-Verlag Berlin Heidelberg.

[Rhi3D] Robert McNeel & Associates, Rhinoceros™, <http://www.rhino3d.com>.

[Roo04] Rooij, R. and Timmer, N., 2004, Design of Airfoils for Wind Turbine Blades, DUWIND, Netherlands.

[San50] Sanuki, M., 1950, “Studies on Biplane Wind Vanes, Ventilator Tubes and Cup Anemometers”, *Papers in Meteorology and Geophysics*, Vol. 1, pp. 81–132.

[Sho16] Shonhiwa, C. and Makaka, G., 2016, “Concentrator Augmented Wind Turbines: A Review”, *Renewable and Sustainable Energy Reviews*, Vol. 59, pp. 1415–1418.

[Tak12] Takahashi, S., Hata, Y., Ohya, Y., Karasudani, T. and Uchida, T., 2012, “Behavior of the Blade Tip Vortices of a Wind Turbine Equipped with a Brimmed-Diffuser Shroud”, *Energies*, Vol. 5, pp. 5229–5242.

[Tol] Understanding Tolerances, <http://wiki.mcneel.com/rhino/faqtolerances>.

[Tos08] Toshimitsu, K., Nishikawa, K., Haruki, W., Oono, S., Takao, M., and Ohya, Y., 2008, “PIV Measurements of Flows Around the Wind Turbines with a Flanged-Diffuser Shroud”, *Journal of Thermal Science*, Vol. 17, pp. 375–380.

[Tree1] Grasshopper Data Tree Editing, http://wiki.bk.tudelft.nl/toipedia/Grasshopper_Data_Tree_Editing

[Tree2] Data Trees, <http://architectureparametrique.com/data-trees-workshop-darchitecture-parametrique/>

[Wan08] Wang, F., Bai, L., Fletcher, J., Whiteford, J., and Cullen, D., 2008, “The Methodology for Aerodynamic Study on a Small Domestic Wind Turbine with Scoop”, *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 96, pp. 1–24.

Appendix

In this appendix, a brief overview of the **Non-Uniform Rational B-Splines (NURBS)** theory is provided, since the particular theory is widely utilized within the proposed methodology, for the representation of the various diffuser augmented wind turbine components.

Non-Uniform Rational B-Splines (NURBS)

Non-Uniform Rational B-Splines is a mathematical model commonly employed in the field of computer graphics for the generation and representation of curves and surfaces, as it provides great flexibility and precision for handling both analytic and modeled shapes. Some reasons for the use of NURBS are, that they:

- offer one common mathematical form for both standard analytical and free-form shapes.
- provide the flexibility to design a large variety of shapes.
- can be evaluated reasonably fast by numerically stable and accurate algorithms.
- are invariant under affine as well as perspective transformations.
- are generalizations of non-rational B-splines and Bezier curves and surfaces.

However, one of the drawbacks NURBS have, is the need for extra storage to define traditional shapes (e.g. circles). This results from parameters in addition to the control points, but finally allow the desired flexibility for defining parametric shapes. NURBS-shapes are not only defined by control points; weights, associated with each control point are also necessary. A NURBS curve, which is a vector-valued piecewise rational polynomial function, is defined as [Pie97]:

$$C(u) = \frac{\sum_{i=0}^n \mathbf{P}_i N_{i,k}(u) w_i}{\sum_{i=0}^n N_{i,k}(u) w_i} \quad (1)$$

where w_i and \mathbf{P}_i are the weight and the position vector of the i^{th} control point, while $N_{i,k}(u)$ is the corresponding B-Spline basis function of k^{th} degree. The B-Splines basis functions are defined recursively as:

$$N_{i,k}(u) = \frac{u - t_i}{t_{i+k} - t_i} N_{i,k-1}(u) + \frac{t_{i+k+1} - u}{t_{i+k+1} - t_{i+1}} N_{i+1,k-1}(u) \quad (2)$$

and

$$N_{i,0}(u) = \begin{cases} 1, & \text{if } t_i \leq u < t_{i+1} \\ 0, & \text{else} \end{cases} \quad (3)$$

where t_i are the knots forming a knot vector

$$U = (t_0, t_1, \dots, t_m) \quad (4)$$

The Knot Vector

The knot vector uniquely determines the B-Splines. The relation between the number of knots $m+1$, the degree k and the number of control points $n+1$, is given by $m=n+k+1$. The sequence of the knots in the knot vector U is assumed to be nondecreasing, i.e. $t_i \leq t_{i+1}$. Each successive pair of knots represents an interval $[t_i, t_{i+1})$ for the parameter values to calculate a segment of a shape [Pie97]. For NURBS, the relative parametric intervals (knot spans) need not be the same for all shape segments, i.e. the knot spacing is non-uniform, leading to a non-periodic knot vector of the form

$$U = (a, \dots, a, t_{k+1}, \dots, t_{m-k-1}, \dots, b, \dots, b) \quad (5)$$

where a and b are repeated with multiplicity of $k+1$.

The multiplicity of a knot affects the parametric continuity at this knot. Non-periodic B-splines, like NURBS, are infinitely continuously differentiable in the interior of a knot span and $k - M - 1$ times continuously differentiable at a knot, where M is the multiplicity of the knot. (In contrast, a periodic knot vector $U = [0, 1, \dots, n]$ is everywhere $k - 1$ times continuously differentiable.) Considering the knot vector for NURBS, the end knot points (t_k, t_{n+1}) with multiplicity $k + 1$ coincide with the end control points P_0, P_n . Since the knot spacing could be non-uniform, the B-Splines are no longer the same for each interval $[t_i, t_{i+1})$ and the degree of the B-Spline can vary. Considering the whole range of parameter values represented by the knot vector, the different B-Splines build up continuous (overlapping) blending functions $N_{i,k}(u)$, as defined in Eq. (2), over this range (Figure A1). These blending functions have the following properties:

- $N_{i,k}(u) \geq 0$, for all i, k, u
- $N_{i,k}(u) = 0$, if u does not belong in $[t_i, t_{i+k+1})$, meaning local support of $k + 1$ knot spans, where $N_{i,k}(u)$ is nonzero.
- If u in $[t_i, t_{i+1})$, the non-vanishing blending functions are $N_{i-k}(u), \dots, N_{i,k}(u)$
- $\sum_{j=i-k}^i N_{j,k}(u) = \sum_{i=0}^n N_{j,k}(u) = \mathbf{1}$, (Partition of Unity)
- In case of multiple knots, $0/0$ is deemed to be zero.

The 1st and 4th properties result into the convex hull, the control points build up for a shape defined by NURBS. Furthermore, the 2nd and 3rd properties show, that $k+1$ successive control points define a shape segment, and a control point is involved in $k+1$ neighboring shape segments. Therefore, changing a control point or weight influences just $k+1$ shape segments, defined over the interval given in Eq. (2).

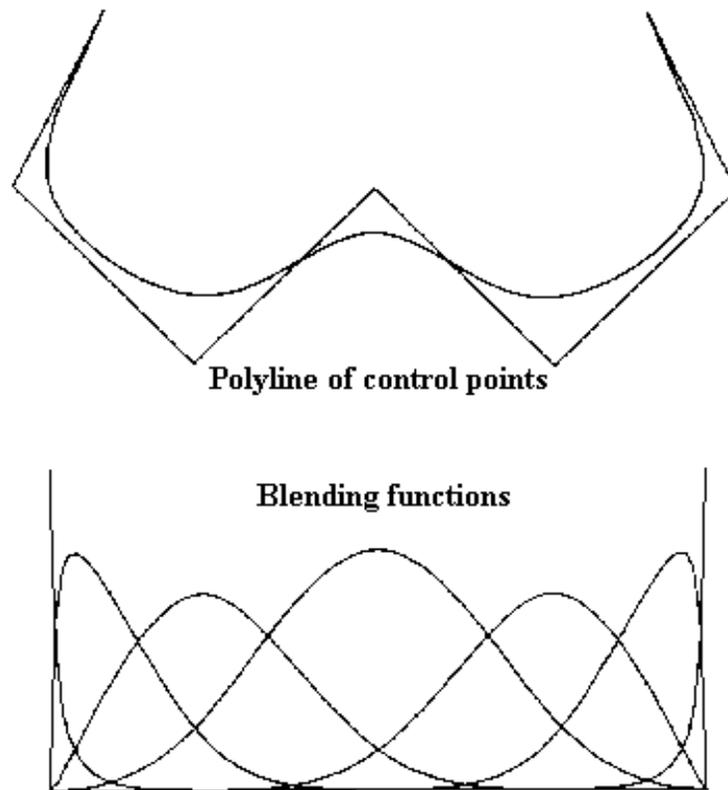


Figure A.1: Cubic NURBS curve with associated blending functions.

Curve/Surface Definition

The previous definition of a NURBS curve (Eq. (1)) can be rewritten by using rational basis functions

$$C(u) = \sum_{i=0}^n P_i R_{i,k}(u) \quad (6)$$

where

$$R_{i,k}(u) = w_i N_{i,k}(u) / \sum_{j=0}^n w_j R_{j,k}(u) \quad (7)$$

A NURBS-surface is define in a similar way:

$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} R_{i,k}(u, v) \quad (8)$$

where

$$R_{i,k}(u, v) = w_i N_{i,k}(u) N_{j,p}(v) / \sum_{r=0}^n \sum_{s=0}^m w_{r,s} N_{r,k}(u) N_{s,p}(v) \quad (9)$$

The rational basis functions have the same properties as the blending functions. One point to emphasize, is their invariance under affine and (even) perspective transformations. Therefore, only the control points have to be transformed to get the appropriate transformation of the NURBS shape.

Computational Algorithm

NURBS can be evaluated effectively by using homogeneous coordinates [Pie97]. The following steps perform the evaluation:

1. add one dimension to the control points (e.g. $P = (x, y) \rightarrow P'(x, y, 1)$) and multiply them by their corresponding weights, i.e. in 2D: $P_i = (x_i, y_i) \rightarrow P'_i(x_i w_i, y_i w_i, w_i)$.
2. calculate NURBS in homogeneous coordinates:

$$C'(u) = \sum_{i=0}^n P'_i N_{i,k}(u) \quad (10)$$

3. map "homogeneous" NURBS back to original coordinate system with:

$$\text{map}(X_1, X_2, \dots, X_n, W) = \begin{cases} (X_1/W, X_2/W, \dots, X_n/W), & \text{if } W \neq 0 \\ (X_1, X_2, \dots, X_n), & \text{if } W = 0 \end{cases} \quad (11)$$

$$C(u) = \text{map}(C'(u)) = \sum_{i=0}^n P_i N_{i,k}(u) w_i / \sum_{i=0}^n N_{i,k}(u) w_i \quad (12)$$

For u in $[t_i, t_{i+1})$, the only existing blending functions to consider in evaluation of the curve at u are $N_{i-k,k}(u), \dots, N_{i,k}(u)$.

The Weights

As mentioned above, changing the weight w_i of a control point P_i affects only the range $[t_i, t_{i+k+1})$ (in case of a curve) [Pie97]. The geometric meaning of the weights is shown as follows (Figure A2).

Defining the points:

$$\begin{aligned} B &= C(u; w_i = 0) \\ N &= C(u; w_i = 1) \\ B_i &= C(u; w_i \notin [0, 1]) \end{aligned}$$

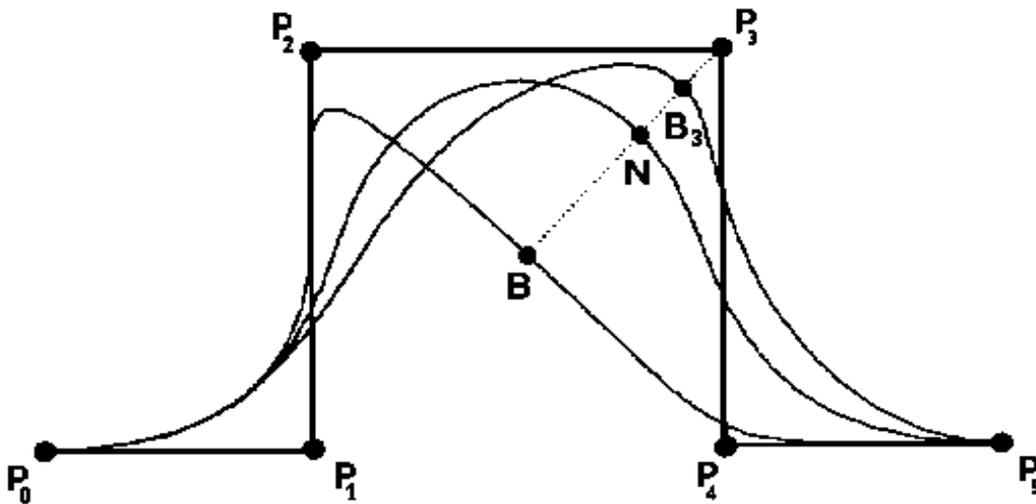


Figure A.2: Geometric meaning of weights.

N and B_i can also be expressed as:

$$N = (1 - a)B + aP_i$$

$$B_i = (1 - b)B + bP_i,$$

where

$$a = R_{i,k}(u; w_i = 1)$$

$$b = R_{i,k}(u).$$

The following identity is obtained from the expression of a and b :

$$1 - a/a : 1 - b/b = P_i N / N : P_i B_i / B B_i = w_i,$$

which is called the cross- or double ratio of the four points P_i, B, N, B_i . From these expressions, the effect of shape modification can be derived:

B_i sweeps out on a straight line segment

if $w_i = 0$ then P_i has no effect on shape

if w_i increases, so b and the curve is pulled toward P_i and pushed away from P_j , for $j \neq i$

if w_i decreases, so b and the curve is pushed away from P_i and pulled toward P_j , for $j \neq i$

if $w_i \rightarrow \infty$ then $b \rightarrow 1$ and $B_i \rightarrow P_i$, if $u \in [t_i, t_{i+k+1})$