POLITECNICO DI MILANO

FACOLTÀ DI INGEGNERIA INDUSTRIALE

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA AERONAUTICA

OpenNOSE: an open-source procedure for the simulation of nasal aerodynamics

RELATORE:
Prof. Maurizio QUADRIO

TESI DI LAUREA DI:
Chiara PESCI
Matricola 755599

ANNO ACCADEMICO 2012 - 2013

"Fu così che a sei anni io rinunziai a quella che avrebbe potuto essere la mia gloriosa carriera da pittore. Il fallimento del mio disegno numero uno e del mio disegno numero due mi aveva disanimato. I grandi non capiscono mai niente da soli e i bambini si stancano di spiegargli tutto ogni volta. Allora scelsi un'altra professione e imparai a pilotare gli aeroplani".

da *Il Piccolo Principe*
– Antoine De Saint-Exupéry [1943]

Contents

List of Figures

List of Tables

Abstract

The present work deals with the numerical simulation of nasal cavity aerodynamics in patients affected by nasal pathologies. It represents the beginning of a feasibility study of virtual surgery. In particular, it is a first step towards implementing the virtual surgery concept; here we consider a specific case of a patient with a severe septal deviation.
The work results from a collaboration with the otolaryngology ward of San Paolo Hospital in Milan. The long-term aim of this cooperation is to provide a robust and reliable open-source procedure for patient-specific treatment of nasal pathologies, such as septal deviation and turbinate hypertrophy.
The first part of the work is focused on turbulence modelling. RANS equations are chosen to describe the flow and the transitional $k_T - k_L - \omega$ model is compared to the more classical one, the $k - \omega - SST$ model.
Then attention is given to virtual surgery to investigate the consequences of the operation. The septal deviation is removed through geometry editing and pre- and post-virtual surgery cases are compared.
Geometry is reconstructed from CT images, in case of virtual-surgery it has been modified, and, finally the mesh is created with the purpose of simulating the nasal cavity flow.
The results suggest the $k_T - k_L - \omega$ turbulence model ability of representing the transitional state of the flow. Unfortunately, no in-vivo information is available to ascertain the validity of the previous suggestion. Further studies are required to validate these results; in other words, in-vitro experimental verification and more accurate simulations such as LES or DNS could offer a quantitative basis for comparison.

Key-words: fluid dynamics, nasal cavity flow, nasal pathology, virtual surgery, CFD, 3D Slicer, FreeCAD, Blender, OpenFOAM, RANS equations, laminar model, turbulence model, $k - \omega - SST$, $k_T - k_L - \omega$.

Sommario

Il presente lavoro riguarda la simulazione numerica dell'aerodinamica all'interno della cavità nasale in pazienti affetti da patologie nasali. In particolare, questo è il primo passo verso l'implementazione del concetto di *virtual surgery*; a tale scopo si considera il caso specifico di un paziente con deviazione del setto piuttosto severa.
Questo lavoro prende le mosse dalla collaborazione con il reparto di otorinolaringoiatria del Policlinico San Paolo di Milano. L'obiettivo a lungo termine di questa cooperazione è di fornire ai chirurghi una procedura *open-source* robusta e affidabile per lo studio di ogni specifico caso affetto da patologie nasali, come deviazione del setto e ipertrofia dei turbinati.

Negli ultimi anni gli avanzamenti nelle capacità di calcolo e modellazione hanno permesso alla fluidodinamica computazionale (CFD) di affrontare geometrie anche molto complesse come la cavità nasale. Inoltre, grazie alle moderne tecniche di *imaging*, è possibile ricostruire esattamente la geometria permettendo un approccio *patien-specific* al fine di trovare il miglior trattamento personalizzato tramite *virtual surgery*.
La CFD permette quindi di studiare localmente il flusso all'interno della cavità nasale ad un livello di dettaglio che le tecniche sperimentali in-vivo e in-vitro non possono fornire.

Gli studi in questo ambito nascono dalla mancanza di conoscenze riguardo la correlazione tra i sintomi riportati dai pazienti e il loro effettivo bisogno di un trattamento chirurgico. Fino ad ora, infatti, i chirurghi hanno basato le loro scelte cliniche sull'esperienza in quanto in un intervento di resezione del setto o riduzione di turbinato ipertrofico non vi sono standard procedurali che stabiliscano quanto tessuto del turbinato o quanto osso del setto nasale vadano rimossi in modo tale da ristabilire una corretta fisiologia nasale. Non è nemmeno noto il limite oltre il quale tale rimozione potrebbe causare danni alla fisiologia del naso.
Una conoscenza approfondita del legame tra morfologia e fisiologia della cavità nasale, associata ad un approccio specifico tramite virtual surgery per ogni paziente affetto da patologie nasali, costituirebbero le basi per una rinologia più funzionale.

Lo scopo di questo lavoro quindi è duplice. Inizialmente si è puntato al raffinamento di alcuni aspetti della procedura in modo da migliorare la simulazione del flusso all'interno della cavità nasale; in particolare ci si è focalizzati sulla creazione di una buona mesh, ovvero la griglia di calcolo, tramite la *utility* snappyHexMesh di OpenFOAM, e sulla scelta di un adeguato modello di turbolenza.

In secondo luogo, si è cercato e applicato uno strumento di modifica della geometria adeguato alla simulazione dell'intervento chirurgico su un paziente specifico. Chiaramente questa parte del lavoro deve essere supportata dai chirurghi, i quali hanno il compito di stabilire gli standard per il trattamento clinico delle patologie nasali coinvolte nello studio.

Con riferimento al primo obiettivo, la procedura si basa su simulazioni di tipo RANS. Il flusso nella cavità nasale è supposto stazionario, in una situazione corrispondente ad una inspirazione a riposo. Nella scelta del modello di turbolenza saranno valutate le prestazioni di un modello relativamente nuovo ($k_T - k_L - \omega$) che dovrebbe essere capace di trattare gli aspetti transizionali di questo flusso, confrontandolo con un modello di turbolenza più classico ($k - \omega - SST$).
I risultati delle simulazioni numeriche suggeriscono che il modello transizionale $k_T - k_L - \omega$ applicato a questo tipo di flussi abbia buone potenzialità. Sfortunatamente non ci sono dati in-vivo disponibili per accertare la validità della precedente assunzione. Sono quindi richiesti ulteriori studi per validare questi risultati; in altre parole, verifiche sperimentali in-vitro e simulazioni più accurate come LES o DNS potrebbero offrire una base quantitativa per il confronto.

Il secondo obiettivo, che considera la *virtual surgery*, coinvolge una procedura ben definita il cui punto di partenza sono le immagini derivanti dalla tomografia computerizzata (CT) del paziente. Dall'osservazione delle immagini CT si identificano le principali patologie nasali e si stabilisce il modo di procedere. Si prosegue quindi con la ricostruzione tridimensionale della geometria nasale che potrà essere in seguito modificata dal chirurgo per simulare l'intervento di rinologia. Una volta create le griglie di calcolo e applicata la fluidodinamica computazionale al problema tramite il software OpenFOAM, è possibile uno studio comparativo dei casi pre- e post-operazione virtuale.
Da questo confronto per il caso in esame emerge che la resezione del setto induce le attese modifiche nel flusso. Si evidenzia inoltre che il caso preoperatorio non è caratterizzato da forti anomalie nelle quantità globali del flusso, ad esempio nelle portate. Ulteriori investigazioni sulle altre quantità fluidodinamiche come campi di pressione e velocità, quantità turbolente e sforzi a parete hanno evidenziato differenze localizzate in prossimità della deviazione del setto che potrebbero influire sulla fisiologia nasale. Tuttavia rimane da determinare se queste considerazioni indichino che l'intervento non sia realmente necessario, o se le funzionalità di quest'organo così delicato, qual è il naso, dipendano dalle caratteristiche di piccola scala del flusso.

Parole chiave: fluidodinamica, cavità nasale, virtual surgery, CFD, 3D Slicer, FreeCAD, Blender, OpenFOAM, equazioni RANS, modello laminare, modello di turbolenza, $k-\omega-SST$, $k_T-k_L-\omega$.

1 Introduction

1.1 The project

Over the last years the increase in computational power and improvements in algorithmic and modelling capability has allowed computational fluid dynamics to tackle complex geometries as the nasal cavity. Thus, in recent years CFD has come to support *in-vivo* and *in-vitro* techniques to better understand the flow behaviour in the upper respiratory tract, as demonstrated by the first meaningful CFD by Zhao et al. in [47]. In fact, well resolved CFD simulations offer the possibility to locally investigate the flow, a capability that is missing in other techniques. Nevertheless, only nasal pathologies related to the geometry, such as septal deviation and turbinate hypertrophy, can be studied by CFD. Other nasal diseases like inflammations cannot be approached with this tool.

A computational fluid dynamics study of the nasal cavity consists in a well-structured procedure, from the production of suitable computed tomography scans or magnetic resonance images to the simulation itself. In fact advanced imaging techniques allow an exact geometry reconstruction and its change into a suitable form to interface with CFD. These capabilities offer the possibility of a *patient-specific* approach, that is on a case-by-case basis study of nasal pathologies to understand the flow behaviour and to eventually find the best personalised treatment through virtual surgery.
In fact, these topics come from the lack of knowledge about the correlation between patient reported symptoms and their effective need of surgical treatments. Until now, surgeons have based their clinical choices on experience because in a rhino-surgery [1] there is no standard of how much turbinate tissue or septal bone has to be removed to re-establish

[1] Rhino-surgery is a reconstructive surgical approach that reshapes the external nose and/or inner structures of the nasal cavities with the objective of re-establishing normal nasal breathing.

physiological breathing. Nor is it known how much tissue or bone can be removed before it causes other damage to the physiology of the nose. Thus, an in-depth knowledge of normal nose breathing as well as the understanding of the relationship between morphology and physiology of the nose will be the foundation for functional rhino-surgery. Moreover,as said before, virtual surgery could offer the possibility to test different ways for a specific operation to find the most suitable one without corrupting nasal physiology.

These arguments, belonging to a medical field, have to be placed side by side with the computational fluid dynamics topics. Thus, attention is focused on numerical modelling, to apply the correct simplifying hypotheses, to find the best suitable model to describe this type of flow, i.e. transitional flow, and finally to validate the simulation results.

1.2 Objectives

The aim of the present work is twofold. First, we would like to improve some aspects of the procedure for the nasal cavity flow simulation, in particular the creation of the mesh and the key choice of the turbulence model. Second, we aim at finding and applying suitable mesh editing tools in order to simulate the surgical operation on a specific patient. Clearly this part of the work must be supported by the ENT[2] surgeons, who set-up the standard of surgical treatment of nasal pathologies. It is important to remark that all the steps described in this work are carried out with open-source software.

With reference to the first goal, the procedure is based on RANS simulations. In choosing the turbulence model, we will evaluate the performances of a model introduced quite recently, that should be capable to deal with transitional features of the flow, and compare them with a more standard choice.

The second goal, where virtual surgery is considered, involves starting from the CT scans of the patient, identifying the main nasal pathologies and then establishing how to proceed. Then, the three-dimensional nasal geometry is reconstructed, and subsequently modified by the surgeon at the computer. Once the meshes are created, a CFD study comparing the

[2]Otolaryngology or ENT (Ear, Nose and Throat).

pre- and post-operation geometries can be carried out.

Both goals are somewhat hindered by the all-encompassing problem that validation of results is impossible, owing to the lack of proper data. It is our long-term plan to remedy this situation and carry out an experimental campaign (backed up by high-fidelity numerical simulations) to produce a reliable database of measurements for validation. For the time being, we will focus ourselves on the procedural aspects of the analysis.

1.3 Thesis structure

The thesis follows the structure outlined in this section.

1. *Introduction.* Overview of the work, specifying the objectives of the thesis and the long term goals. The chapter contains also the state of the art of nasal cavity simulations and a brief description of the patient specific procedure.

2. *Nasal cavity flow.* The chapter introduces the reader to the nasal cavity issue, both from medical and computational point of view.

    - *Physiology and anatomy.* Here are briefly described the physiology and anatomy of the human upper respiratory tract to have background informations for the interpretation of the simulation results.

    - *Experiments and numerical simulations.* Studying the literature about experiments and flow simulations in the nasal cavity, it is outlined the state of the art in these fields.

3. *Numerical simulation of the nasal cavity flow.* The chapter contains the fluid-dynamics description of the nasal cavity flow with its governing equations, the simplifying hypotheses and the turbulence models applied to problem closure.

4. *Simulation setup.* The chapter contains the *toolchain* from the creation of the mesh to the CFD simulations. It is divided into two

main sections:

- *Creation of the geometry.* In this chapter are summarised the main steps to reconstruct the nasal cavity geometry from CT scans and to modify the region interested by the nasal pathology.

- *OpenFOAM.* Here OpenFOAM software is introduced, meshes are created following the required criteria and case variables are initialised.

5. *Results.* This chapter consists of two important parts:

   - comparison between laminar and turbulent cases to find the most suitable turbulence model;

   - comparison between pre and post surgery analyses to understand the effect of a possible operation.

6. *Conclusion.* The chapter ends this thesis specifying future developments.

In the appendix are better explained the procedures to obtain the geometry *stl* file from the CT of the patient, how to modify parts of the nasal cavity with Blender software and how to create the mesh for the simulation through the *snappyHexMesh* utility in OpenFOAM.

2 Nasal Cavity Flow

2.1 The Upper Human Respiratory Tract

To better understand the steps of virtual surgery and simulation results it is important to familiarize first with nasal architecture; the interested reader refer to articles by Doorly et al. [7], Jones [21] and Zachov et al. [43] for further details.

2.1.1 Physiology

The nose is the first component of the human respiratory tract and it has to protect the delicate tissues of the lungs. The functions of the human nose are diverse and range from breathing and smelling to humidification, warming and cleaning of the inhaled air.

Defence mechanisms, which include alterations in passage size, and mucosal secretion, highlight the adaptability of the nasal airways to respond to external challenges. As an olfactory organ the nose samples inspired air, directing a portion to sensory receptors where it is temporarily retained to facilitate odourant molecule capture. With respect to humidification and warming, the nose manages to humidify the air to a humidity over 80% before it enters the lungs. Air is heated through conduction, convection and radiation with blood flow in the opposite direction to incoming airflow, that improves the efficiency of warming. Notice that the temperature of the air in the nasopharynx only varies by $2\,^{\circ}\mathrm{C} - 3\,^{\circ}\mathrm{C}$ within reasonable variation of the temperature of the outside air.
That the nose accomplishes such disparate tasks is remarkable, since efficient air conditioning requires a rapid transit, whereas chemical sensing benefits from longer time scales.

2.1.2 Anatomy

The main anatomical features are identified in figures 2.1, 2.2 and 2.3. The nasal cavity is a space situated above the oral cavity and hard palate and below the skull base and intracranial compartment. The septum divides the nasal cavity into left and right sides of approximately equal volumes, from the nares, twin anterior openings at the nostrils, to the nasal choanae. In particular, the nasal septum is composed of cartilage in its front-end and bone towards the back of the nose.

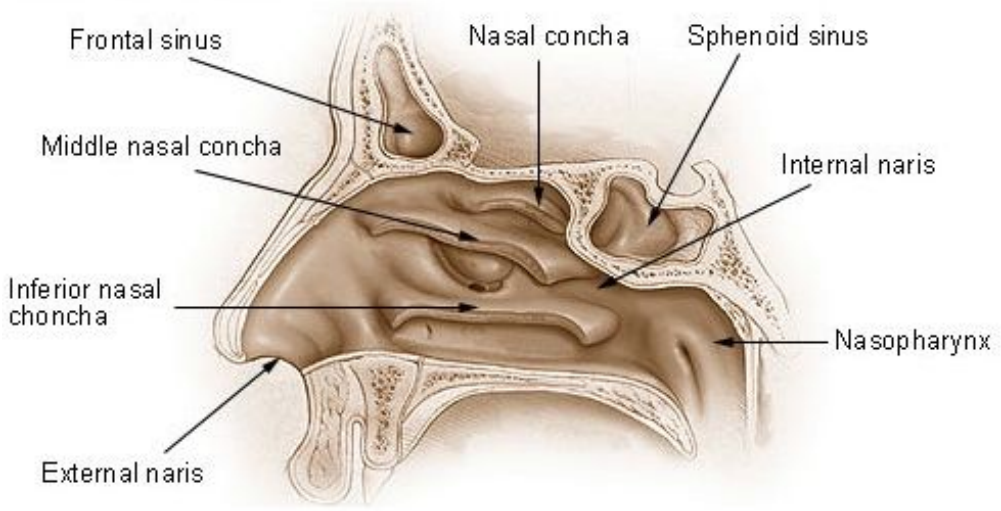
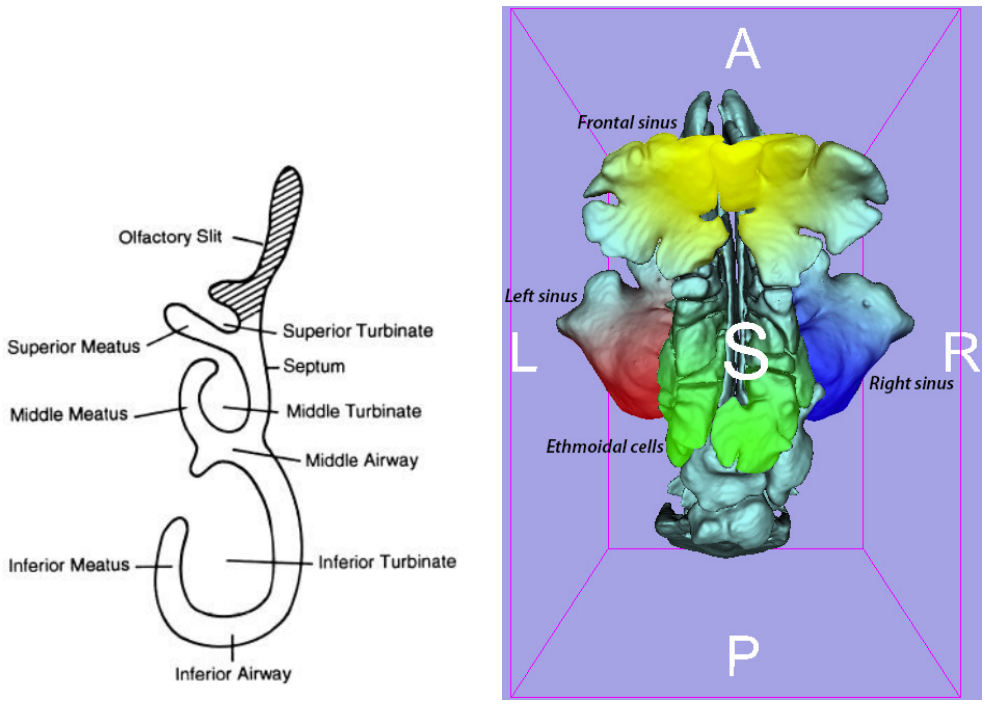


Figure 2.1: Anatomical features of the nose

There are three contributors to the bony septum: the perpendicular plate of the ethmoid bone, vomer bone, and maxilla bone. The left and right nasal cavities become continuous in the back of the nose via the opening to the nasopharynx, termed the choana. In this area, the nasal cavity transits into the nasopharynx. The nasopharynx contains a collection of centrally located lymphoid tissue called the adenoids.
In the cavity of each side of the septum (fig. 2.2), the passageway is much reduced by the turbinates, curled bony structures covered with soft tissue and mucosa that project into the airway. Turbinates are important because they serve to increase the mucosal surface area of the nasal cavity and regulate nasal airflow. Below each turbinate runs the correspondingly named passageway or meatus, for instance the inferior and middle meatuses.



(a) *Coronal slice of half nasal cavity, image from [10].*

(b) *3D view of nasal cavity from the top.*

Figure 2.2: Views of nasal cavity

The inferior turbinate is the largest of the three paired turbinates, and runs along the entire length of the lateral nasal wall, adjacent to the nasal floor. It has a lateral vascular cushion which acts like a radiator to the nose, warming the air as well as humidifying it. The nasolacrimal (tear) duct, which collects tears from the eye, drains beneath the inferior turbinate into the inferior meatus, which explains why crying leads to a rapid onset of nasal discharge.
The middle turbinate projects into the central nasal cavity and resides next to the nasal septum. It is attached to the lateral nasal wall posteriorly just above the inferior turbinate but behind the maxillary sinus. Superiorly, it inserts along the lateral nasal wall and skull base.
The frontal sinus, maxillary sinus, and anterior ethmoid sinus cells (fig. 2.3) drain beneath the middle turbinate into the middle meatus. Sometimes an ethmoid sinus cell can expand within the normally thin walls of the middle turbinate, and form an enlarged structure termed *concha bullosa.*
The superior turbinate is the smallest of the turbinates. It resides just

above and behind the middle turbinate, and also attaches to the skull base superiorly and nasal wall laterally. The sphenoid sinus and posterior ethmoid sinus cells drain into spaces between the nasal septum and superior turbinate called the sphenoethmoid recess.

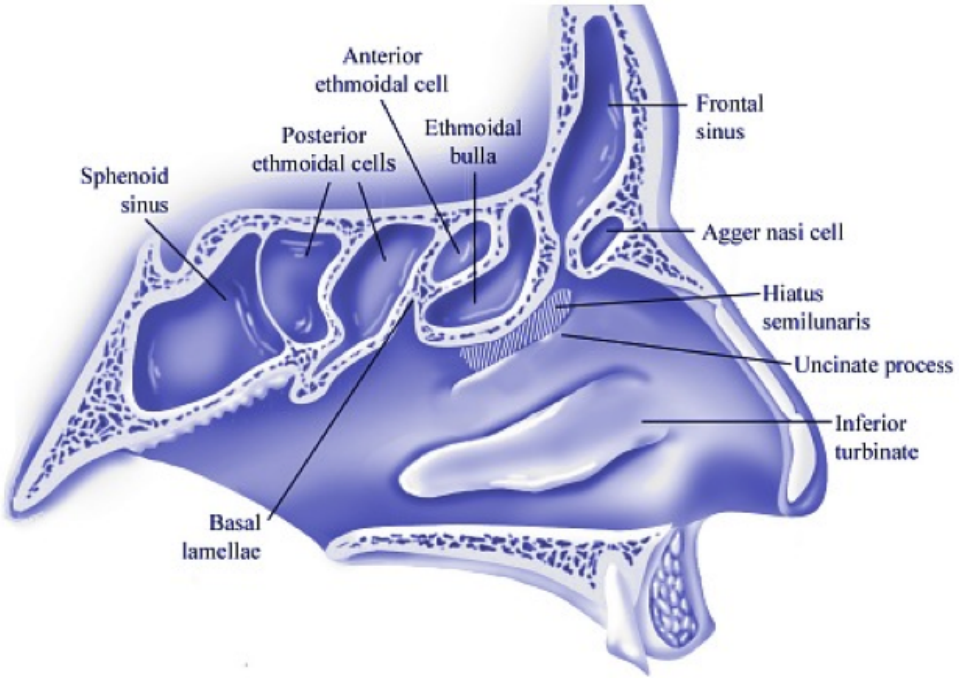


Figure 2.3: Sagittal view of nasal cavity: cells and sinuses highlighted

Finally, the nasal valve area is the area of maximum airflow resistance in the nasal airway and is formed by the returning or overlap between the upper and lower lateral cartilages, the inferior turbinate and the septum.

Another complicating feature is that nasal geometry is never fully fixed; for example, natural cyclic variation in airway passages associated with alternating states of congestion/decongestion is reported to occur in a significant proportion of the population; furthermore, responses to irritants from the external environment can cause changes. In the present work physiological changes in the geometry of the nasal passageways are not considered.

2.2 Experiments and numerical simulation

The investigation of the nasal airflow is a relatively recent issue. It represents an extended topic, with different types of flow simulations, *in-vivo* or *in-vitro* measurements, analysis of the particle transport in the respiratory tract and virtual surgery. As said before, this work is focused on a feasibility study of virtual surgery, in terms of real application of the procedure by surgeons before operation; thus the following brief literature review is based towards flow simulation and virtual surgery.

2.2.1 Experimental results

The first attempts to study the nasal cavity flow were supported only by in-vivo exams, i.e. acoustic rhinometry, or in-vitro laboratory experiments, i.e. with laser anemometry or more recently with particle image velocimetry. Discussing such subjects is not the main task of this work. The interested reader is advised to consult the papers mentioned below, which are very briefly reviewed here. Remember, however, that experimental results are the main and more reliable way of validating CFD results.
Early studies had the aim to understand the aerodynamic effects of shapes and dimensions of the nasal fossa on the velocity field, Arbour et al. [1]. These first results showed also that the airflow is moderately turbulent, but for normal breathing laminar flow is present in most of the nasal cavity, Hahn et al. [10]. Another key phenomenon identifiable through flow visualisation was the emergence of the jet from the internal nasal valve into the main cavity, how it impacts on the middle turbinate, and the large enhancement of dispersion that accompanies the initial appearance of flow instability, Doorly et al. [6].
Further investigations of nasal velocity profiles provide important information necessary to characterise all nasal airway transport processes of pollutants, odorants and aerosol particle, Cheng et al. [4].
In particular, in the last years, attention is focused on stereoscopic PIV in the nasal cavity with high flow (*NHF*) therapy because the flow pattern with *NHF* was found to differ significantly from natural breathing, as reported by Spence et al. in [34] and [33].
Articles by Hahn et al. [10] and Spence et al. [34], [33] are important for this thesis because they represent an important source of data useful for validation of numerical works, since they provide complete 3D, phase-averaged velocity fields.

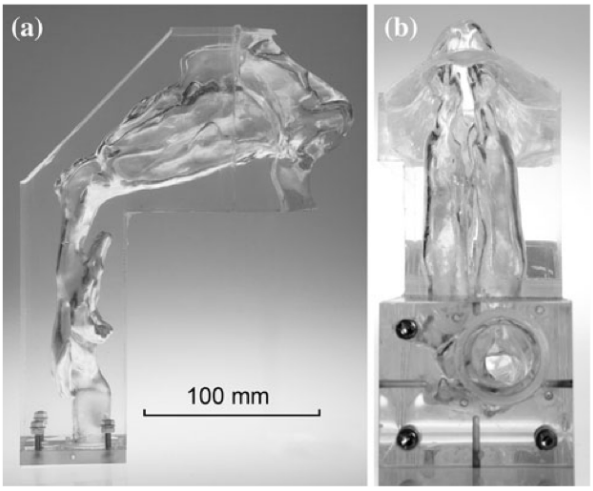


Figure 2.4: *(a).* Silicone nasal cavity flow phantom viewed sagittally from the left and *(b)* axially from the bottom. Figure from Spence et al. [33]

Other articles refer to the peak of nasal inspiratory flow measurement as a screening tool for patients with, for example, nasal sinus diseases. Refer to [11], [25], [35] and [44] for more details.

2.2.2 CFD results

Henceforth the attention is focused on numerical simulation of nasal flow through computational fluid dynamics. CFD is used for various purposes: to understand the flow in a healthy nose, to model particle and odorant transport and to simulate the heating and humidifying abilities of the nose, or to investigate the impact of nasal pathologies on the quality of the flow.
In any case, a CFD simulation requires a physical model to be chosen: the flow could be considered as steady, unsteady or represented by a sequence of quasi-steady states; it could be laminar, turbulent or transitional. The choice of one model rather than another should be done through reference to previous knowledge and studies of nasal cavity flows.

Understanding the flow in the nasal cavity

In the paragraphs below the studies on healthy nasal cavity are analysed. Most of them consider a steady flow.
The steady state hypothesis for nasal cavity flows is supported by some studies by Martonen et al. [19], Croce et al. [5], Hoerschler et al. [12], Tan et al. [36] and Zhu et al. [48].
The Strouhal number has been taken as an indicator for the unsteadiness of the developing flow in the nasal cavity. Tan et al. [36] demonstrate that for a quiet restful breathing frequency and a half-nasal flow rate of less than 12 l/min, a typical value for respiratory cycle, the Strouhal number is less than 0.2, indicating that a quasi-steady approximation is reasonable. The flow is then considered as a series of steady conditions, in particular [36] only considers a steady inspiration and a steady exhalation. They find that during the inhalation phase, vortices and turbulence are located at anterior and bottom parts of the nasal cavity; but there is no vortex in the whole nasal cavity during the expiratory phase. The maximum airflow velocity occurs around the plane of palatine velum; after the nasal valve, the peak velocity of inhaled airflow decreases and it increases again at the post-naris. Vice versa, the exhaled airflow decelerates after it has passed the post-naris and it accelerates again at nasal valve. Martonen et al. [19] notice a jet formation immediately at the nostrils, too; the jets' strengths and the sizes of flow recirculation zones were proportional to velocity values. Hoerschler et al. [12] remark that only at transition from inspiration to expiration the unsteady results sensitively differ from the steady state solutions. The comparison with rhinomanometry measurements confirms their numerical findings.
On the other hand, some authors claim that the unsteadiness of nasal cavity flow is important and should be taken into account. The analysis by Elad et al. [9] demonstrate that, even during quiet breathing, wall shear stress and temperature anywhere in the nose vary significantly with time, as high as 0.5 Pa/s and 20 °C/s, respectively. In the article by Lee et al. [17] a large eddy simulation *(LES)* suggests that the major difference between the steady- and unsteady-state results could be the inertial effect resulting from the variation of mass flux. A comparison between steady-state and unsteady-state computations showed a considerable difference in flow characteristics especially in the expiration phase. In particular, temperature profiles were more sensitive to the inertial effect than the velocity field was. The streamline pattern in the nasal cavity shows the presence of a recirculation zone which forces the fluid to remain in the olfactory region most of the time during the inspiration phase, and another

secondary flow separation zone in the nasopharynx during the expiration period.

Regarding turbulence models, the choice of a laminar model is made, for instance, by Elad et al. [9], Martonen et al. [19], Zhu et al. [48] and Croce et al., [5].
In the work by Tan [36] instead, RANS equations with the standard $k-\varepsilon$ turbulence model were employed to better detect recirculation regions and describe the streamlines distribution among the cavity; even if they found that airflow in the nasal cavity was usually laminar and very similar in both sides.
More accurate turbulence models are employed by Luo et al. [18], Weinhold et al. [40] and Lee et al. [17]. Luo opted for a LES simulation, but to maintain a limited computational effort he applied the LES modelling to a simplified model of the geometry which consists of a single asymmetric bifurcation model of human upper airway. The results demonstrate that the LES model is capable of capturing instantaneous eddy formation and flow separation in laminar, transitional and turbulent flow regimes, and hence it may be used as a powerful tool to trace the particle trajectories and particle deposition in human airways.
Weinhold et al. [40], knowing that in the range of physiological breathing laminar, turbulent and transitional flow regimes may exist at the same time, compute the velocity and pressure fields in the reconstructed cavities for the entire range of physiological nasal inspiration using an enhanced two-equation turbulence model of the k - ε type. To conclude this first review of previous works other two articles are recommended to the reader, one by Ishikawa [13], and one by Zhao and Dalton [45].

Particle tracking and nose conditioning capacity

One of the main functions of the human nose is conditioning of inhaled air, which guarantees optimal air conditions to the lower respiratory tract. In contrast to in vivo measurements, numerical simulations are an effective method to explore the impact of airflow distribution on nasal air heating and humidifying ability.
Naftali et al. [24] and, recently, Sommer et al., [32], demonstrate that a healthy nose can efficiently provide about 90% of the heat and the water fluxes required to modify the inspired air to near-alveolar conditions from different environmental states.
It has been demonstrated that during quiet breathing through the nose,

ambient air at 20-25 °C was conditioned up to 34 °C and 80% relative humidity by the time it reached the pharynx. In particular, the inspired airstream is compressed and accelerated; after passing the nasal valve it is disrupted, slowed down and dispersed consecutively. Dispersion and hence turbulence created by this barrier within the airstream enable the mucosa to transfer heat and moisture to the inhaled air due to a close contact between air and nasal wall. The increase of air temperature within the anterior nasal segment directly behind the nasal valve area is much more effective than along the rest of the nasal airways; more precisely, the majority of heating and wetting was completed before the air left the turbinates region.
Other authors focused their attention on particle and odorant transport patterns in nasal airflow. Kleinstreuer et al. [16] make a review in this field.
Zhao et al. in [47] and [46], take advantage of CFD to predict airflow and odorant transport, which may ultimately determine olfactory sensitivity. Results suggest that anatomical changes in the olfactory region (upper meatus) and in the nasal valve region will strongly affect airflow patterns and odorant transport with subsequent effects on olfactory function.
Knowing particle transport and deposition patterns provide accurate information about local and regional aerosol deposition in the human upper respiratory tract. Results would be of great value for researches of dosimetry and health effect studies.
The LES numerical results by Jin et al. [14] demonstrate that the particles deposition efficiencies are strongly dependent on the particle diameter, particle density, breathing intensity and respiration mode. The unsteady respiration mode and the increasing of particle diameter, particle density and breathing intensity improve the particle deposition efficiencies in the human upper respiratory tract. Shi et al. [31] introduced the wall roughness effect for the first time in nasal simulations, taking into account the presence of gravity and particle inertia, too.

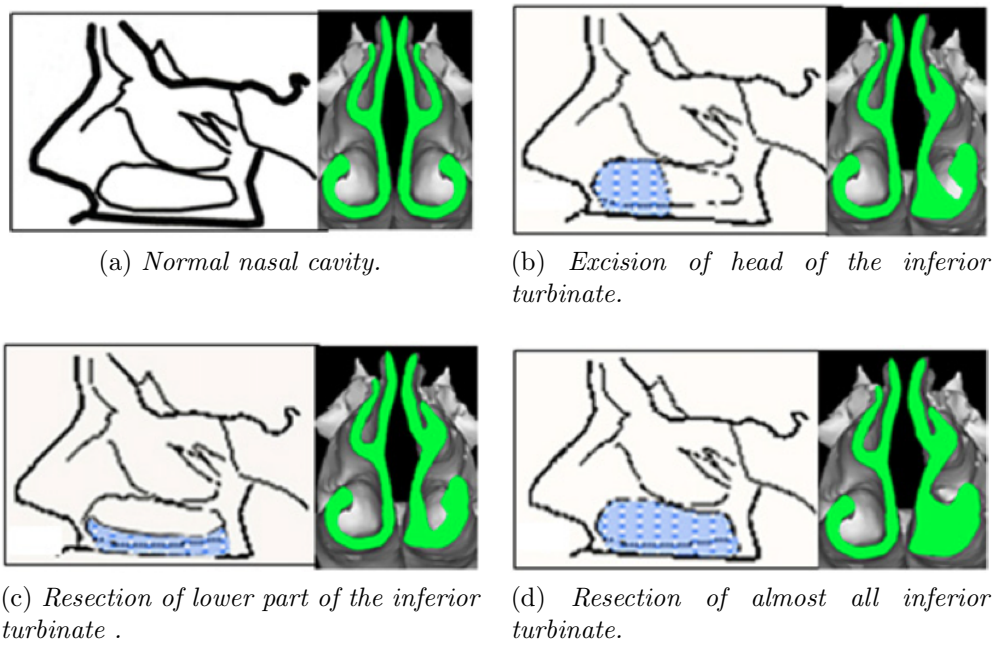
Nasal pathologies and virtual surgery

Computational fluid dynamics studies provide detailed information on the aerodynamic effects of nasal pathologies on nasal airflow patterns and their associated physiological functions. Nasal pathologies that need surgical treatment usually are septal deviation, turbinate hypertrophy and chronic rhinosinusitis. To investigate the effects of these pathologies, surgeons might take advantage of CFD simulations. Moreover, applying virtual surgery they would be able to choose the best patient-specific nasal

surgery. These topics, well examined by Kim in his review [15], also represent the core of the current work.
Some authors, such as Xiong [42] and Chen [3], focused their attention to Functional Endoscopic Sinus Surgery, *FESS*, as a common surgical treatment for refractory chronic rhinosinusitis. This operation involves initial uncinate and ethmoid excision, followed by an opening of the maxillary, sphenoid and frontal sinuses. The procedure facilitates increased sinus ventilation and improved sinus draining through the opened middle meatus-anterior ethmoid complex.
Xiong et al. [42] perform a pre- and post-FESS simulation with a laminar steady-state CFD analysis. In the post-FESS model, as expected, there was an increase in airflow distribution in the maxillary, ethmoid and sphenoid sinuses, and a 13% increase through the area connecting the middle meatus and the surgically opened ethmoid. There was a gradual decrease in nasal resistance in the posterior ethmoid sinus region after FESS. The article by Chen et al. [3] responds to the need of more accurate investigations, extending Xiong's study. They evaluate the effects of a particular FESS case with unsteady nasal aerodynamic flow patterns to attempt to answer whether there were circulations inside the sinus regions. Reynolds averaged Navier-Stokes equations were solved for the unsteady turbulent flow with $k-\omega$ SST model. Although the enlarged airflow passageway helped to decrease nasal resistance and alleviated original nasal blockages (as shown in the previous study), the nasal cavity with FESS had induced airflow into these sinus regions because of the enlarged ostia and removed upper ethmoidal cells. The current transient modelling results also showed different airflow patterns and existences of continuous local circulations with a same airflow rate or at quiet statuses, but at different breathing phases. To avoid other flow-related nasal symptoms with changes of aerodynamic patterns inside, such FESS procedures are suggested to implement with smaller enlargements of ostia and fewer removal of ethmoid cells.
In the same field of maxillary sinus, Na [23] investigated about the effects of an accessory ostium (AO) in addition to the natural one. The effects of the AO are unknown, but it is reported that patients with chronic rhinosinusitis and antrochoanal polyp show a higher rate of occurrence of the AO.
The turbinate hypertrophy, usually regarding the inferior one, has been discussed by Wexler in [41] and in the last years by Chen [3] and Na [22]. Inferior Turbinate Reduction (ITR) surgery is often performed for chronic nasal obstruction attributed to refractory turbinate enlargement. There are numerous techniques for ITR and most studies indicate a significant

rate of clinical success in improving nasal patency. Nevertheless, the available outcome data consist of patient's measures of satisfaction. Thus, not only is the physiologic interpretation of such subjective measures limited, but also there is a lack of correlative geometric information regarding the location and the amount of tissue reduction that produces specific local and transnasal aerodynamic effects. Therefore, it is desirable to quantify the effects of ITR on nasal airflow and pressure-flowrate relationships, maybe through CFD. Wexler, [41], was one of the first to study the effects of a unilateral ITR through the simulation of a circumferential removal of soft tissue bulk along the left inferior turbinate. He obtained a a broad reduction of pressure along the nasal airway, including the regions distant from the inferior turbinate vicinity. In contrast, relative airflow changes were regional: airflow was minimally affected in the valve region, increased in the lower portion of the middle and posterior nose, and decreased dorsally.
The study by Chen et al. [3] provides information about the impact of ITR on the airflow aerodynamic patterns, especially for the existence and distribution changes of turbulence inside the nasal cavity. Indeed, the average turbulent intensity increased rapidly, especially in a severe blocked nose; it affects the nasal valve region and more studies are needed in order to understand the optimal range of the turbulent airflow, which is the most favourable for achieving the basic functions of the nose.
Finally, the article by Na [22] presents one of the most complete studies of ITR. Three types of turbinectomy, both with laminar and turbulent flow simulation are analysed, and temperature and humidity are taken into account, too. The left turbinates are normal and right nasal cavity is modified by (1) excision of the head of the inferior turbinate; (2) resection of the lower fifth of the inferior turbinate; (3) resection of almost the entire inferior turbinate, as shown in figure 2.5.
Results show that the turbinectomy altered the main stream direction. The flow rate in the upper airway near the olfactory slit decreased in models (1) and (3). This may weaken the olfactory function of the nose. Fluid and thermal properties are dependent on turbinate resection volume, position, and type; widening of the inferior airway does not always result in decreased flow resistance or wall heat transfer. The gains and losses of inferior turbinectomy were considered by analysis of the post-surgery model results. Nasal resistance was increased in model (1) due to sudden airway expansion. Nasal resistance increased and the wall heat transfer decreased in model (3) due to sudden airway expansion and excessive reduction of the mucosal wall surface area. Local shear stress and pressure gradient levels were increased in models (1) and (3). Thus, in

this circumstance, model (2) was found to be the best one.



(a) *Normal nasal cavity.*

(b) *Excision of head of the inferior turbinate.*

(c) *Resection of lower part of the inferior turbinate .*

(d) *Resection of almost all inferior turbinate.*

Figure 2.5: Geometry and the post-surgery models, figures from [22]

Turbinate reduction should be taken into account also as a complementary part of septal deviation reduction or resection operation.

Finally, the two reference articles for this study are presented. Both Chen et al. [2] and Rhee et al. [29] deal with CFD analysis of septal deviation and virtual surgery.
Usually for a healthy nose, the septum that separates the two sides of the nose is straight; instead, a deviated septum is not straight, but shifted to one side. Based on the type and severity of the deviation, it can be classified into linear (I-shaped), C-shaped, and S-shaped. A deviation in the septum of the nose can cause narrowing of the nasal passage, lessening the amount of secretions that can drain, and sometimes lead to sinus infections.
The imbalance of the nasal cavity due to septal deviation is thought to be a typical aetiology of nasal airway obstruction (NAO). Since septal deviation is so commonly observed as an anatomic variation in healthy adults, clinicians must often decide whether or not a septal deviation is clinically relevant to nasal airway obstruction. Surgical correlation of septal devi-

ations has been one of the most common procedures in otolaryngology, but, the objective evidence to support the efficacy of surgery is controversial. Despite septoplasty and turbinectomy being two of the most common performed surgical procedures for relieving nasal airway obstruction, the long-term outcomes of such treatments are still not satisfactorily assessed. Without any standard criteria, the decision to perform septoplasty was usually based on clinical observations alone. The main reason behind this was the lack of knowledge on the correlation between patient-reported symptoms and objective findings, including fluid mechanical properties (Refer to [15] for more details).
Chen et al. [2] analysed the effects of septal deviation on the aerodynamic flow pattern compared with that of a normal nose by both laminar and turbulent ($k-\omega$ SST) CFD models with respect to different flow rates. Simulations were done on an healthy nose and a septal deviated one. For the healthy case, the air flow is almost uniform for the left and right cavities; for the deviated one, there is less air going through the obstructed part. The nasal resistance of the deviated nose is almost twice that of the healthy one.
Regarding velocity and pressure fields, the septal deviation causes the main airflow to follow a narrowed path with greater pressure gradient or abrupt pressure jump posterior to the side of the deviation. Moreover, vortex areas are detected in the upper passageway in the cavity and beyond the deviation, and higher shear stress distribution is found with respect to the healthy nose. Nevertheless, there is still no statistical relationship between septal deviation and sinus disease.
In the article by Rhee et al. [29] virtual surgery is performed on a 3D reconstructed model of a severe septal deviated nose to predict post-surgical outcomes. Pre-surgery and post-surgery CT scans were available for comparison between real and virtual septoplasty and right ITR. The pre-surgery model was digitally altered to generate three virtual surgery models to reproduce the anticipated surgical changes: right ITR only, septoplasty only, and septoplasty with right ITR, refer to figure 2.6 for clearness. The real treatment of the patient was based on clinical presentation and the standard of medical care. In this case, surgeons decided for a septoplasty using standard septoplasty techniques and right inferior turbinate reduction (ITR). The anterior one-half of the inferior turbinate was de-bulked by performing submucosal resection of the bone and removal of the submucosal tissue with sharp dissection.
The authors conduced a steady-state inspiratory airflow simulation for flow rates corresponding to normal resting breathing.
Both virtual and actual surgery resulted in decreases in unilateral nasal

resistances; however, the effect was more pronounced on the patient's left side, which was affected by the septal deviation. Virtual septoplasty alone

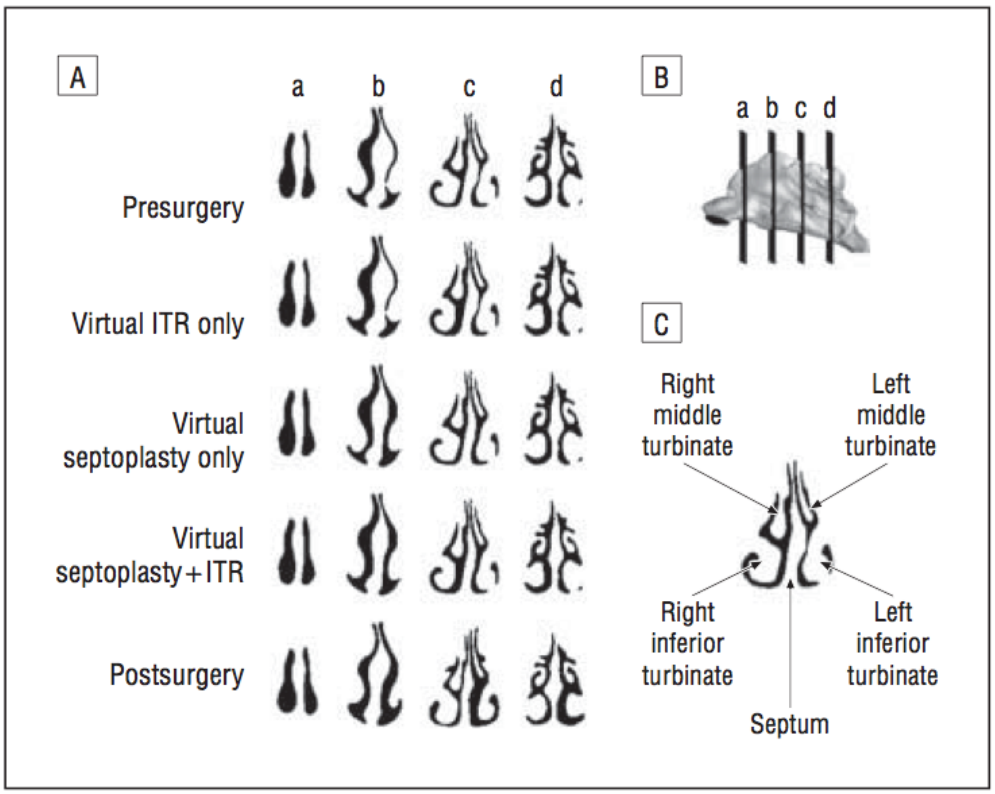


Figure 2.6: Virtual surgery models and real pre- and post-surgery, figure from [29]

accounted for almost all of the drop in resistance on that side. In the pre-surgery CFD model, the majority of the airflow passed through the right middle region; the post-surgery and virtual surgery models predicted a more balanced airflow distribution between the two sides. The virtual septoplasty with ITR model and the septoplasty only model resulted in a very similar distribution; in contrast, the ITR-only model did not alter the airflow allocation between sides. The actual post-surgery CFD model resulted in an airflow allocation slightly favouring the left side when compared with the CFD-predicted results for the virtual surgery models, but this difference in airflow allocation may have been due to the nasal cycle. This analysis suggests that septoplasty alone could achieve improved airflow allocation without the need of ITR for this patient. The decision to perform turbinate reduction is typically based on the surgeon's clinical judgment, and the exact method may vary from surgeon to surgeon.

The use of CFD modelling for surgical planning could help determine which patients may benefit from ITR in the setting of septal deviation. In this particular patient case, the use of pre-surgical CFD planning may have changed the decision to perform ITR using this particular surgical technique, since simulations would have predicted little impact on nasal resistance.

3 Numerical simulation of the nasal cavity flow

In this chapter are briefly presented the governing equations for the fluid motion, in their Reynolds-averaged form. To solve the problem the Reynolds' average is adopted to obtain the RANS equations.
Finally, to solve the closure problem, the turbulence models used in this work are introduced.

3.1 Navier-Stokes Equations

The governing equations for the fluid motion descend from the conservation laws for mass, momentum and energy under the thermodynamic quasi equilibrium hypothesis. This system of equations is called *Navier-Stokes equations* and its validity is very general, [28].
In the most general case, the Navier-Stokes equations with the state equation, a relation between thermodynamic variables such as density, internal energy and pressure, form a closed system of partial differential equation. The system assumes the form below (3.1), where equation (3.1a) is the continuity equation, relations (3.1b) and (3.1c) stands for momentum and energy conservation, respectively.

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{V}) = 0 \tag{3.1a}$$

$$\frac{\partial (\rho \mathbf{V})}{\partial t} + \boldsymbol{\nabla} \cdot \left(\rho \mathbf{V}\mathbf{V} + p\mathbf{I} + \mathbf{J}^d_{\mathbf{Q}}\right) = \mathbf{F} \tag{3.1b}$$

$$\frac{\partial}{\partial t}\left[\rho\left(e + \frac{V^2}{2}\right)\right] + \boldsymbol{\nabla} \cdot \left[\rho\left(e + \frac{V^2}{2}\right)\mathbf{V} + p\mathbf{V} + \mathbf{J}^d_{\mathbf{Q}} \cdot \mathbf{V} + \mathbf{J}^d_E\right] = L \tag{3.1c}$$

where ρ is the density, $\mathbf{V}$ is the velocity vector, p is the pressure, $\mathbf{F}$ the external forces resultants, e the internal energy and L the sum of external

forces power. The terms $\mathbf{J}^d_{\mathbf{Q}}$ and $\mathbf{J}^d_E$ represents, respectively, dissipation fluxes of momentum and energy calculated through the following (3.2) e (3.3):

$$\mathbf{J}^d_{\mathbf{Q}} = -\mu\left(\boldsymbol{\nabla}\mathbf{V} + (\boldsymbol{\nabla}\mathbf{V})^T - \frac{2}{3}(\boldsymbol{\nabla}\cdot\mathbf{V})\,\mathbf{I}\right) - \lambda\,(\boldsymbol{\nabla}\cdot\mathbf{V})\,\mathbf{I} \tag{3.2}$$

$$\mathbf{J}^d_E = -k\boldsymbol{\nabla}T \tag{3.3}$$

where μ is the dynamic viscosity, λ is the second viscosity coefficient (for newtonian fluid $\lambda = -2/3\mu$), k is the thermal conductivity and T the temperature.

In case of incompressible flows, such the nasal cavity one, the system (3.1) could be simplified as follows. The state equation reduces to $\rho = const$, the energy equation is decoupled from the rest of the system and the remaining relations take the form of the *Navier-Stokes imcompressible equations* with constant properties (3.4):

$$\nabla\cdot\mathbf{V} = 0 \tag{3.4a}$$

$$\frac{\partial\mathbf{V}}{\partial t} + (\mathbf{V}\cdot\boldsymbol{\nabla})\,\mathbf{V} + \frac{1}{\rho}\boldsymbol{\nabla}p = \nu\nabla^2\mathbf{V} + \mathbf{f} \tag{3.4b}$$

where $\nu = \mu/\rho$ is the cinematic viscosity and $\mathbf{f} = \mathbf{F}/\rho$ is the body force acceleration.

3.1.1 Reynolds averaged Navier-Stokes equations

The system (3.4) fully describes the dynamics of every Newtonian incompressible flow with constant properties. Nevertheless, for Reynolds number high enough, the solutions to (3.1) show a chaotic behaviour called *turbulence*. A fundamental characteristics of turbulent motion, as a chaotic system, is the strong sensitivity to initial conditions which increases with the Reynolds number. Turbulence tackled with statistics, but with the added difficulty of total lack of scale separation; in fact turbulent flows involve all length scales, from microscopic to the characteristic length scale of the problem. Thus, a deterministic approach would imply solving for all length and time scales with too high computational effort[1].

[1]This very expensive procedure from the computational point of view, is called *Direct Numerical Simulation, DNS*. It implies the numerical integration of the differential system (3.1) or (3.4) over a suitably resolved domain.

Usually, the Reynolds decomposition (3.5) is applied to the Navier-Stokes equations, which are solved only for the mean part; the fluctuating field is neglected. Indicating with $\langle\cdot\rangle$ the mean statistical operator, the Reynolds decomposition for the velocity field looks like

$$\mathbf{u}(\mathbf{x},t) = \langle\mathbf{u}(\mathbf{x},t)\rangle + \mathbf{u}'(\mathbf{x},t) \tag{3.5}$$

After the Reynolds decomposition, equations (3.4), without body forces, remain:

$$\nabla\cdot\langle\mathbf{V}\rangle = 0 \tag{3.6a}$$

$$\rho\frac{\partial\mathbf{V}}{\partial t} + \rho\boldsymbol{\nabla}\cdot\langle\mathbf{V}\mathbf{V}\rangle + \boldsymbol{\nabla}\langle p\rangle + \boldsymbol{\nabla}\cdot\langle\mathbf{J}^d_{\mathbf{Q}}\rangle = 0 \tag{3.6b}$$

where it is assumed that mean and differentiation commute, and $\langle\mathbf{J}^d_{\mathbf{Q}}\rangle$, that is the dissipative part of stress tensor for newtonian fluid with constant properties, is written by component as

$$J^d_{\mathbf{Q},ij} = -\mu\left(\frac{\partial\langle V_i\rangle}{\partial x_j} + \frac{\partial\langle V_j\rangle}{\partial x_i}\right) \tag{3.7}$$

As can be seen from equations (3.6) and (3.7), all the linear terms are represented by mean variables; instead, the non linear term $\langle\mathbf{V}\mathbf{V}\rangle$ of (3.6b) will be

$$\langle\mathbf{V}\mathbf{V}\rangle = \langle\mathbf{V}\rangle\langle\mathbf{V}\rangle + \langle\mathbf{V}'\mathbf{V}'\rangle \tag{3.8}$$

The quantity $\rho\langle\mathbf{V}'\mathbf{V}'\rangle$ has the dimensions of a stress and it is called Reynolds stress tensor $\mathbf{R}$. Thus, defining $\mathbf{R} = \rho\langle\mathbf{V}'\mathbf{V}'\rangle$, it is possible to introduce the total stress tensor $\mathbf{T}$:

$$\mathbf{T} = \langle\mathbf{J}^d_{\mathbf{Q}}\rangle + \mathbf{R} \tag{3.9}$$

By components, the tensor $\mathbf{T}$ could be written as

$$T_{ij} = -\mu\left(\frac{\partial\langle V_i\rangle}{\partial x_j} + \frac{\partial\langle V_j\rangle}{\partial x_i}\right) + \rho\langle V_i'V_j'\rangle$$

The introduction of $\mathbf{T}$ allows rewriting equation (3.6b) as follows:

$$\rho\frac{\partial\mathbf{V}}{\partial t} + \rho\boldsymbol{\nabla}\cdot\langle\mathbf{V}\rangle\langle\mathbf{V}\rangle + \boldsymbol{\nabla}\langle p\rangle + \boldsymbol{\nabla}\cdot\mathbf{T} = 0 \tag{3.10}$$

Equations (3.6a) and (3.10) are called Reynolds averaged equations or *Reynolds Averaged Navier-Stokes* equations, *RANS*.

RANS equations presents a closure problem: there is no relationship between the Reynolds stress tensor and the mean velocity field thus $\langle \mathbf{V} \rangle$ could not be determined. A model expressing the Reynolds stress tensor (and thus $\mathbf{T}$) in terms of the mean field is required. Turbulence models offer a solution to the closure problem.

3.2 Turbulence models

The closure problem for RANS equation can be solved through:

- *eddy-viscosity models*, which are based on the Boussinesq hypothesis; the Reynolds stresses are assumed to be proportional to mean velocity gradients. These models can be algebraic or composed by one or more differential equations.

- *Reynolds stress models*, where model transport equations are solved for the individual Reynolds stresses component $\langle V_i' V_j' \rangle$ and for the dissipation ε of the turbulent kinetic energy. These models are no more considered in the present work.

Taking into account the *eddy-viscosity models*, the Boussinesq hypothesis implies:

$$R_{ij} = -2\nu_t \overline{S}_{ij} \tag{3.11}$$

which explains how R_{ij}, the generic component of the Reynolds stress tensor, is proportional through the eddy viscosity [2] to the tensor $\overline{\mathbf{S}}$ that is formed with the spatial derivation of the mean velocity field:

$$\overline{S}_{ij} = \frac{1}{2}\left(\frac{\partial \langle V_i \rangle}{\partial x_j} + \frac{\partial \langle V_j \rangle}{\partial x_i}\right) \tag{3.12}$$

If the turbulent-viscosity hypothesis is accepted as an adequate approximation, all that remains to be determined is an appropriate specification

[2] The eddy viscosity is not a fluid property, but a flow property. This parameter is only an artifice based on the parallel relationship between viscous stresses and rate-of-strain tensor.

of the turbulent viscosity ν_t. This can be written as the product of a velocity u^* and a length ℓ^*:

$$\nu_t = u^* \ell^*,$$

and the task of specifying ν_t is generally approached through specification of u^* and ℓ^*.
In the sections below the two turbulence model used in this work are presented: the $k-\omega-SST$ model, one of the classical two-equation models; and the $k_T - k_L - \omega$ model, a relatively new three-equation model.
The choice of $k_T - k_L - \omega$ model rather than more classical ones is dictated by the transitional nature of the nasal cavity flow.

3.2.1 $k-\omega-SST$ Model

The $k-\omega-SST$ turbulence model is a two-equation eddy-viscosity model. It is an evolution of the standard $k-\omega$ model, where a shear stress transport (SST) formulation is included to combine the best of the $k-\omega$ and $k-\varepsilon$ models. The use of a $k-\omega$ formulation in the inner part of the boundary layer makes the model directly usable all the way down to the wall through the viscous sub-layer, hence the SST $k-\omega$ model can be used as a Low-Re turbulence model without any extra damping functions. The SST formulation also switches to a $k-\varepsilon$ behaviour in the free-stream and thereby avoids the common $k-\omega$ problem that the model is too sensitive to the inlet free-stream turbulence properties. The SST $k-\omega$ model produces a bit too large turbulence levels in regions with large normal strain, like stagnation regions and regions with strong acceleration. This tendency is much less pronounced than with a normal $k-\varepsilon$ model though.

This model adds two equations to the RANS equations, one for the turbulent kinetic energy k, and one for the turbulence frequency $\omega = k/\varepsilon$. The turbulent-viscosity is calculated through k and ω. Rather than being based on the exact equations for the turbulent quantities, the model equations for k and ω are best viewed as being entirely empirical.
Recalling the definition of the substantial or material derivative $D(\cdot)/Dt$:

$$\frac{D}{Dt}(\cdot) = \frac{\partial}{\partial t}(\cdot) + \langle V_i \rangle \frac{\partial}{\partial x_i}(\cdot)$$

the model equations reduce to:

$$\frac{Dk}{Dt} = \mathcal{P}_k - \beta^* \omega k + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_k \nu_t)\frac{\partial k}{\partial x_j}\right] \tag{3.13a}$$

$$\frac{D\omega}{Dt} = \alpha S^2 - \beta\omega^2 + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_\omega \nu_t)\frac{\partial \omega}{\partial x_j}\right] + 2(1 - F_1)\,\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i} \tag{3.13b}$$

and kinematic eddy viscosity is

$$\nu_t = \frac{\alpha_1 k}{max\,(\alpha_1\omega, SF_2)} \tag{3.14}$$

The closure coefficients and the auxiliary relations are as follows:

$$F_2 = tanh\left[\left[max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right)\right]^2\right]$$

$$\mathcal{P}_k = min\left(\tau_{ij}\frac{\partial V_i}{\partial x_j}, 10\beta^* k\omega\right)$$

$$F_1 = tanh\left\{\left\{min\left[max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{4\sigma_{\omega 2}k}{CD_{k\omega}y^2}\right]\right\}^4\right\}$$

$$CD_{k\omega} = max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i},\ 10^{-10}\right)$$

$$\phi = \phi_1 F_1 + \phi_2\,(1 - F_1)$$

$$\alpha_1 = \frac{5}{9};\ \alpha_2 = 0.44$$

$$\beta_1 = \frac{3}{40};\ \beta_2 = 0.0828$$

$$\beta^* = \frac{9}{100}$$

$$\sigma_{k1} = 0.85;\ \sigma_{k2} = 1$$

$$\sigma_\omega = 0.5;\ \sigma_{\omega 2} = 0.856$$

All these empirical relations demonstrates how obtaining a good method is a delicate issue, and for this reason results could have an inadequate accuracy sometimes.

3.2.2 $k_T - k_L - \omega$ Model

The $k_T - k_L - \omega$ model was developed by D. K. Walters and D. Cokljat in 2008 [38]. As said before, it is an eddy-viscosity turbulence model employing three additional transport equations. The model is based on the $k - \omega$ framework and represents a substantial refinement to a transition-sensitive model according to the previous work by D. K. Walters and J. H. Leylek in 2004 [39].
The Reynolds stress is typically interpreted as the *turbulent stress*, although it is important to note that the Reynolds stress arise as a consequence of the averaging process and is nonzero for any time-varying velocity field, even if velocity fluctuations are not due to *turbulence* in strict sense. In this theory, thus, transition as well as turbulent fluctuations may be modelled through the Reynolds stress tensor and the model presented here is based on this assumption.

Model Concepts and Development

A pretransitional boundary layer is effectively laminar in terms of the mean velocity profile.
At the beginning of transition phenomena, when freestream turbulence intensities are lower than about 1%, the development of low amplitude pretransitional velocity fluctuations is dominated by self-sustained instability mechanism, the Tollmien-Schlichting waves. As freestream turbulence intensity increases, the mean velocity in the pretransitional boundary layer becomes noticeably distorted with respect to the typical Blasius profile with an increase in momentum in the inner region and a decrease in the outer. This shift in mean velocity profile is accompanied by the development of relatively high-amplitude streamwise fluctuations, which can reach intensities several times the freestream level. This process causes an increase of skin friction and heat transfer in the pretransitional region and eventually leads to transition through the breakdown of the streamwise fluctuations. This process is known as *by-pass transition.*
These streamwise fluctuation represent Klebanoff modes and are not turbulence in the usual sense of the word. This distinction was firstly made by Mayle and Schulz [20], who developed the laminar kinetic energy concept to describe the development of pretransitional fluctuation leading to by-pass transition. Such fluctuation are very different from turbulent fluctuation, both structurally and dynamically. In fact, the energy is almost entirely contained in the streamwise component and the classical energy cascade from larger to smaller scales is not present, instead fluctuations

are amplified at certain scales determined by the boundary layer itself and remain at relatively small frequency. Dissipation is therefore also expected to be relatively low, except very near the wall due to the no-slip condition. Thus Mayle and Schulz proposed to add a second kinetic energy equations to govern these fluctuations and this approach was used both by Walters and Cokljat in [38] and Walters and Leylek in [39].

Laminar Kinetic Energy Production and model overview At present, the dynamics of laminar kinetic energy production are not entirely understood, but a number of researchers have underlined two critical aspects:

1. selectivity of boundary layer to certain freestream eddy scales;

2. amplification of low-frequency disturbances in the boundary layer by the mean shear.

A reasonable explanation of the k_L behaviour is due to Volino [37]. He considered the possibility that growth of k_L is caused by a *splat mechanism*; in other words, the wall redirects the normal fluctuation into a streamwise component. At the same time local pressure gradients in the boundary layer are created, leading to disturbance amplification. Since *splats* are likely to occur only for eddies with large length scale relative to the wall distance, the turbulent energy spectrum could be divided into two sections in the near-wall region, as can be seen from figure 3.1, where λ_{eff} (the effective turbulent length scale for the small scale turbulence) stands for the cutoff eddy size. Scales smaller than λ_{eff} interact with the mean flow as typical turbulence and larger scales contribute to the production mechanism for k_L. The small and large scale components are $k_{T,s}$ and $k_{T,l}$, respectively. Note that far from walls, in the free stream, $k_{T,s} \rightarrow k_T$ and $k_{T,l} \rightarrow 0$ and the *splat mechanism* does not exist.
In this model, the transition process is represented by a transfer of energy from the laminar kinetic energy k_L to the turbulent kinetic energy k_T; where k_T represents the magnitude of fluctuations in a fully turbulent tridimensional flow.
The beginning of transition process is based on the concept of *shear-sheltering* and consideration for the relevant time-scales for nonlinear dis-

turbance amplification and dissipation. The effect of shear-sheltering[3] is to inhibit nonlinear turbulence breakdown mechanism, such as it occurs in the pretransitional boundary layer. Once transition initiates, the effects of shear-sheltering are restricted to the viscous sublayer in the turbulent boundary layer.

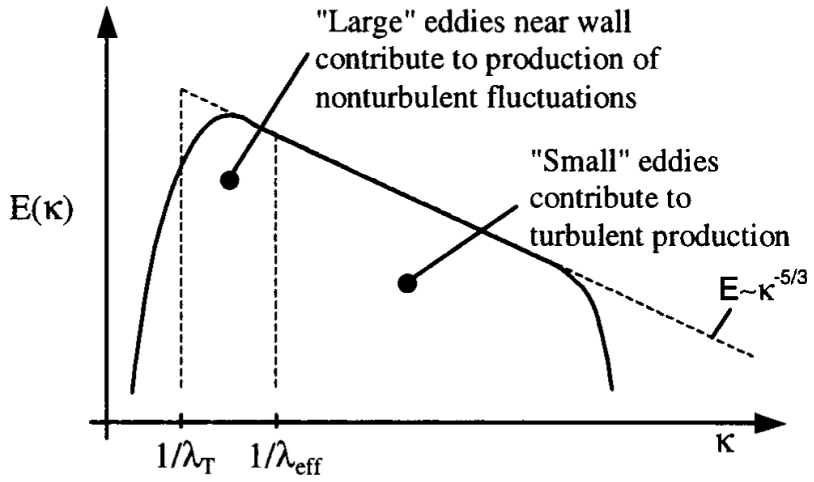


Figure 3.1: Illustration of wall-limiting concept for k_L production, figure from [39]

In this model, a production term includes shear sheltering effect while transition initiation is represented by transfer terms in the k_L and k_T equations. Then, the authors define a dimensionless quantity to represent the transition inception, that is the ration between turbulent production and molecular diffusion time-scales. They choose these two time-scales because entrained disturbances in the developing boundary layer undergo nonlinear breakdown and amplification when turbulent production time-scale is short enough with respect to the time-scale associated with molecular diffusion. Therefore, the onset of transition would occur at a critical value of this time-scales ratio (*by-pass* transition concept).
Note that this model is also extended to include natural transition, by assuming that disturbances associated with Tollmien-Schlichting waves are characterised by a time-scale proportional to the inverse of the vorticity within the pretransitional boundary layer. Here, the ratio between Tollmien-Schlichting time-scale and molecular diffusion time-scale sets the natural transition criterion.

[3]Shear-sheltering refers to the dumping of turbulence dynamics that occurs in thin regions of high vorticity.

Downstream of either by-pass or natural transition, the model should correctly predict a fully turbulent boundary layer.

Model Equations

This section summarise the governing equations for the model; an incompressible single-phase flow with no body forces is considered. The flow is governed by the RANS equations and a linear eddy-viscosity model [4] is adopted for the Reynolds stresses. Three additional model transport equations are solved for the turbulent kinetic energy k_T (3.15), the laminar kinetic energy k_L (3.16), and the scale-determining variable ω (3.17), defined here as $\omega = \varepsilon/k_T$ where ε is the isotropic dissipation. The transport equations are

$$\frac{Dk_T}{Dt} = \mathcal{P}_{k_T} + \mathcal{R}_{\mathrm{BP}} + \mathcal{R}_{\mathrm{NAT}} - \omega k_T - \mathcal{D}_T + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\sigma_k}\right)\frac{\partial k_T}{\partial x_j}\right] \tag{3.15}$$

$$\frac{Dk_L}{Dt} = \mathcal{P}_{k_L} - \mathcal{R}_{\mathrm{BP}} - \mathcal{R}_{\mathrm{NAT}} - \mathcal{D}_L + \frac{\partial}{\partial x_j}\left[\nu\frac{\partial k_L}{\partial x_j}\right] \tag{3.16}$$

$$\begin{aligned}\frac{D\omega}{Dt} = C_{\omega 1}\frac{\omega}{k_T}\mathcal{P}_{k_T} + \left(\frac{C_{\omega R}}{f_W} - 1\right)\frac{\omega}{k_T}\left(\mathcal{R}_{\mathrm{BP}} + \mathcal{R}_{\mathrm{NAT}}\right) - C_{\omega 2}\omega^2 \\ + C_{\omega 3} f_\omega \alpha_T f_W^2 \frac{\sqrt{k_T}}{d^3} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right]\end{aligned} \tag{3.17}$$

The various terms in the model equations represent production, destruction and transport mechanism. In the ω equation, first, third and fifth terms on the right-hand side of equation (3.17) stand for fully turbulent production, destruction and gradient transport. The transition production term (second term of right-hand side of (3.17)) is intended to produce a reduction in turbulence length scale during the transition breakdown process. The fourth term on the right-hand side of (3.17) was included in order to decrease the length scale in outer region of the turbulent boundary layer, which is necessary to ensure correct prediction of

[4] The simplest RANS models assume a linear relationship between the Reynolds stresses and the strain rate tensor, the so called Boussinesq hypothesis:
$\rho\langle V_i V_j\rangle - \frac{1}{3}\rho\langle V_k V_k\rangle\delta_{ij} = -2\nu_T S_{ij}$, where ν_T is the dynamic turbulent, or eddy, viscosity.

the boundary layer wake region.

The total fluctuation kinetic energy is

$$k_{\mathrm{TOT}} = k_T + k_L \tag{3.18}$$

Considering each component of model equations in detail, the production of turbulent kinetic energy by mean strain is modeled as:

$$\mathcal{P}_{k_T} = \nu_{T,s}\overline{S}^2 \tag{3.19}$$

where the small-scale eddy-viscosity is defined as

$$\nu_{T,s} = \nu_{T,I} f_W f_{\mathrm{INT}} C_\mu \sqrt{k_{T,s}}\lambda_{\mathrm{eff}} \tag{3.20}$$

and $k_{T,s}$ is the effective small scale turbulence

$$k_{T,s} = f_{\mathrm{SS}} f_w k_T \tag{3.21}$$

$\overline{S}$ is the module of the tensor $\overline{\mathbf{S}}$ with components expressed as in (3.36). The kinematic wall effect is included through an effective (wall-limited) turbulence length scale λ_{eff} and a dumping function f_W:

$$\lambda_{\mathrm{eff}} = \min\left(C_\lambda d,\, \lambda_T\right) \tag{3.22}$$

$$f_W = \left(\frac{\lambda_{\mathrm{eff}}}{\lambda_T}\right) \tag{3.23}$$

with

$$\lambda_T = \frac{\sqrt{k_T}}{\omega} \tag{3.24}$$

Through an effective turbulence Reynolds number, a viscous damping function is computed to take into account of the viscous wall effect.

$$f_v = 1 - \exp\left(-\frac{\sqrt{\mathrm{Re}_T}}{A_v}\right) \tag{3.25}$$

$$\mathrm{Re}_T = \frac{f_W^2 k_T}{\nu\omega} \tag{3.26}$$

The shear-sheltering effect discussed previously is included in the damping function f_{SS}:

$$f_{\mathrm{SS}} = \exp\left[-\left(\frac{C_{\mathrm{SS}}\nu\overline{\Omega}}{k_T}\right)^2\right] \tag{3.27}$$

The turbulent viscosity coefficient C_μ is defined to satisfy the realizability constraint:

$$C_\mu = \frac{1}{A_0 + A_s \left(\frac{\overline{S}}{\omega} \right)} \tag{3.28}$$

An empirical intermittency damping function introduce the effect of intermittency on the turbulence production:

$$f_{INT} = \min \left(\frac{k_L}{C_{\mathrm{INT}} k_{\mathrm{TOT}}}, 1 \right) \tag{3.29}$$

The production of laminar kinetic energy k_L is assumed to be governed by the large-scale near-wall turbulent fluctuations, based on the correlation of pretransitional fluctuation growth with freestream low-frequency wall-normal turbulent fluctuations.
The large-scale turbulence contribution is

$$k_{T,l} = k_T - k_{T,s} \tag{3.30}$$

where the small scale contribution is defined by equation (3.21). The production of laminar kinetic energy is

$$\mathcal{P}_{k_L} = \nu_{T,l} \overline{S}^2 \tag{3.31}$$

where

$$\nu_{T,\mathrm{I}} = \min \left\{ f_{\tau \mathrm{I}} C_{11} \left(\frac{\overline{\Omega} \lambda_{\mathrm{eff}}^2}{\nu} \right) \sqrt{k_{T,\mathrm{I}}} \lambda_{\mathrm{eff}} + \beta_{\mathrm{TS}} C_{12} \mathrm{Re}_\Omega d^2 \overline{\Omega},\ \frac{0.5 \cdot (k_L + k_{T,\mathrm{I}})}{\overline{S}} \right\} \tag{3.32}$$

The production term is comprised of two parts, the first addresses the development of Klebanoff modes and the second addresses natural modes. Here there are:

$$\mathrm{Re}_\Omega = \frac{d^2 \overline{\Omega}}{\nu} \tag{3.33}$$

$$\beta_{TS} = 1 - \exp \left(- \frac{\max \left(\mathrm{Re}_\Omega - C_{\mathrm{TS},crit},\ 0 \right)^2}{A_{\mathrm{TS}}} \right) \tag{3.34}$$

$$f_{\tau,\mathrm{I}} = 1 - \exp \left[-C_{\tau,\mathrm{I}} \frac{k_{T,\mathrm{I}}}{\lambda_{\mathrm{eff}}^2 \overline{\Omega}^2} \right] \tag{3.35}$$

where $\overline{\Omega}$ is the module of the tensor $\overline{\mathbf{\Omega}}$ defined by:

$$\overline{\Omega}_{ij} = \frac{1}{2}\left(\frac{\partial\langle V_i\rangle}{\partial x_j} - \frac{\partial\langle V_j\rangle}{\partial x_i}\right) \tag{3.36}$$

The anisotropic near-wall dissipation terms for $\mathcal{D}_T$ and $\mathcal{D}_L$ are in the form:

$$\mathcal{D}_T = \nu\frac{\partial\sqrt{k_T}}{\partial x_j}\frac{\partial\sqrt{k_T}}{\partial x_j} \tag{3.37}$$

$$\mathcal{D}_L = \nu\frac{\partial\sqrt{k_L}}{\partial x_j}\frac{\partial\sqrt{k_L}}{\partial x_j} \tag{3.38}$$

The turbulent transport terms in the k_T and ω equations include an effective diffusivity α_T defined as

$$\alpha_T = f_v C_{\mu,\mathrm{std}}\sqrt{k_{T,s}}\lambda_{\mathrm{eff}} \tag{3.39}$$

In the boundary layer production term is included a kinematic damping function to reproduce proper behaviour of the boundary layer wake region

$$f_\omega = 1 - \exp\left[-0.41\cdot\left(\frac{\lambda_{\mathrm{eff}}}{\lambda_T}\right)^4\right] \tag{3.40}$$

The remaining terms in the transport equations are related to the laminar-to-turbulent mechanism in the model. As already noticed, transition occurs as a transfer energy from k_L to k_T, meanwhile there is a reduction in turbulence length scale from freestream value to the value found in an equilibrium turbulent boundary layer. The model terms $\mathcal{R}_{\mathrm{BP}}$ and $\mathcal{R}_{\mathrm{NAT}}$ appear with opposite signs in the k_T and k_L equations and represents bypass and natural transition, respectively.

$$\mathcal{R}_{\mathrm{BP}} = C_R\beta_{\mathrm{BP}}k_L\omega/f_W \tag{3.41}$$

$$\mathcal{R}_{\mathrm{NAT}} = C_{\mathrm{R,NAT}}\beta_{\mathrm{NAT}}k_L\overline{\Omega} \tag{3.42}$$

Transition initiation is governed by the threshold functions β_{BP} and β_{NAT}. As discussed previously, transition in both cases is assumed to initiate when the characteristic time-scale for turbulence production is smaller than the viscous diffusion time-scale of the pretransitional fluctuation. The forms used are

$$\beta_{\mathrm{BP}} = 1 - \exp\left(-\frac{\phi_{\mathrm{BP}}}{A_{\mathrm{BP}}}\right) \tag{3.43}$$

$$\phi_{\mathrm{BP}} = \max\left[\left(\frac{k_T}{\nu\overline{\Omega}} - C_{\mathrm{BP,crit}}\right),\ 0\right] \tag{3.44}$$

$$\beta_{\mathrm{NAT}} = 1 - \exp\left(-\frac{\phi_{\mathrm{NAT}}}{A_{\mathrm{NAT}}}\right) \tag{3.45}$$

$$\phi_{\mathrm{NAT}} = \max\left(\frac{\mathrm{Re}_\Omega - C_{\mathrm{NAT,crit}}}{f_{\mathrm{NAT,crit}}},\ 0\right) \tag{3.46}$$

$$f_{\mathrm{NAT,crit}} = 1 - \exp\left(-C_{\mathrm{NC}}\frac{\sqrt{k_L}d}{\nu}\right) \tag{3.47}$$

Note that through the function $f_{\mathrm{NAT,crit}}$ the amplitude of pretransitional fluctuations influences the initiation of natural transition in an appropriate manner.
The model allows to include heat transfer effects, too. The turbulent heat flux vector can be modelled using a turbulent thermal diffusivity α_θ

$$-\langle V_i T\rangle = \alpha_\theta \frac{\partial T}{\partial x_i} \tag{3.48}$$

$$\alpha_\theta = f_W\left(\frac{k_T}{k_{\mathrm{TOT}}}\right)\frac{\nu_{T,s}}{\mathrm{Pr}_\theta} + (1 - f_W)\, C_{\alpha,\theta}\sqrt{k_T}\lambda_{\mathrm{eff}} \tag{3.49}$$

Finally, the turbulent viscosity used in the momentum equations is the sum of the small-scale and large-scale contributions defined above.

$$\nu_T = \nu_{T,s} + \nu_{T,\mathrm{I}} \tag{3.50}$$

To conclude, model constants are listed below in the table 3.1.

Summary of model constants		
$A_0 = 4.04$	$C_{\mathrm{INT}} = 0.75$	$C_{\omega 1} = 0.44$
$A_s = 2.12$	$C_{\mathrm{TS,crit}} = 1000$	$C_{\omega 2} = 0.92$
$A_v = 6.75$	$C_{\mathrm{R,NAT}} = 0.02$	$C_{\omega 3} = 0.3$
$A_{\mathrm{BP}} = 0.6$	$C_{11} = 3.4 \cdot 10^{-6}$	$C_{\omega R} = 1.5$
$A_{\mathrm{NAT}} = 200$	$C_{12} = 1.0 \cdot 10^{-10}$	$C_\lambda = 2.495$
$A_{\mathrm{TS}} = 200$	$C_R = 0.12$	$C_{\mu,\mathrm{std}} = 0.09$
$C_{\mathrm{BP,crit}} = 1.2$	$C_{\alpha,\theta} = 0.035$	$Pr_\theta = 0.85$
$C_{\mathrm{NC}} = 0.1$	$C_{\mathrm{SS}} = 1.5$	$\sigma_k = 1$
$C_{\mathrm{NAT,crit}} = 1250$	$C_{\tau,\mathrm{I}} = 4360$	$\sigma_\omega = 1.17$

Table 3.1: $k - k_L - \omega$ model constants

4 Simulation setup

CFD of the nasal cavity involves various passages, starting from manipulation of CT scans to the effective simulation start-up. In the subsequent sections each step of the procedure is briefly seen. The interested reader is addressed to the appendix for more details.

4.1 Creation of the geometry

The final geometry is the results of further manipulations. The starting point is the images acquisition followed by the 3D reconstruction; then geometry is split into the required parts and if virtual surgery is required, the user acts on geometry through the Blender software.
All these steps can be done on a personal computer with standard capabilities, a minimum of 4 GB of memory is advised.

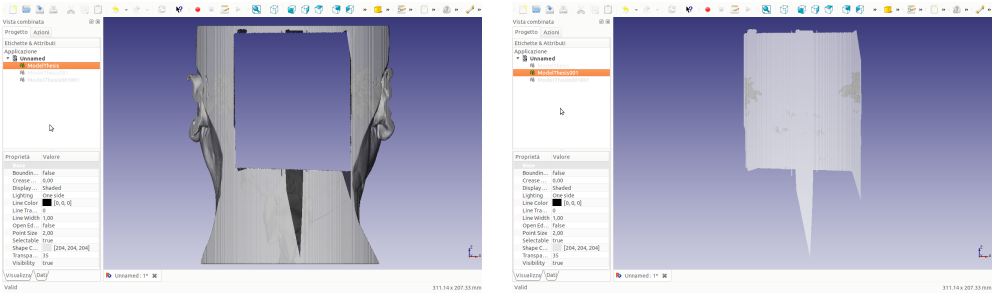
4.1.1 3D-Slicer

The raw material on which the user starts the study of the nasal cavity consists of CT scans, with adequate resolution and spatial spacing. The series of axial CT slices is collected by the 3D-Slicer software. Here the most important issue is to set the correct HU[1] threshold. This value defines which anatomical components will be reconstructed, assuming a key-role to include all tissues and bones.
Once the user has set the HU threshold and he eventually applies some modifications, i.e. to exclude ears from the reconstruction, the software builds the 3D surface, see figure A.6. The resultant geometry can now be exported in a suitable format for OpenFOAM, that is *stl* format.

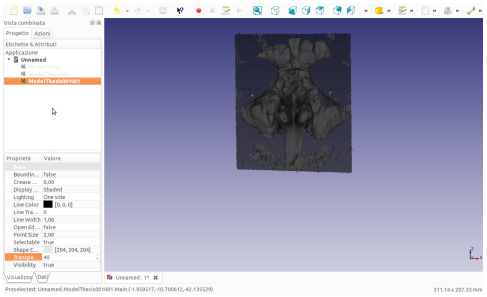
[1]Hounsfield unit, refer to appendix A.1 for further details.

4.1.2 FreeCAD

The resultant geometry from 3D-Slicer is very complex and large. Thus, the editing of the all geometry will be more difficult for the user and more expensive from the computer memory *(RAM)* point of view. Another reason to divide the geometry is represented by mesh's requirements; this point will be clarified later in section 4.2.1.
The open-source software used to split geometry in the required parts is FreeCAD. The interested reader can find all the steps in appendix B.
Here is recalled figure B.4 from the appendix B to show the resulting parts after geometry decomposition (figure4.1). In the post-surgery case a further part is created in order to make easier the geometry editing (see figure C.4).



(a) *First part.* (b) *Second part.*



(c) *Nasal cavity.*

Figure 4.1: Three surfaces divided, back view

4.1.3 Blender

The open-source software exploited to edit edit the geometry in order to simulate a surgery is Blender. It is a graphics software product with

huge capabilities, but in this work only *sculpt* and *edit* modes are used. The basic steps of geometry editing are resumed below:

- to identify and select the regions affected by the nasal pathology;
- to delete the faces which form the obstruction or the turbinate hypertrophy;
- to reconstruct the geometry resembling to a sane one.

This procedure should imitate the surgeon's steps during an operation. Notice that in this part of the all procedure the collaboration with surgeons is fundamental.
Then, the user can deal with mesh generation and CFD simulations.

4.2 OpenFOAM

The OpenFOAM (Open Field Operation and Manipulation) CFD Toolbox is a free, open source CFD software package produced by OpenCFD Ltd. To run an application in OpenFOAM the user has to create a case directory containing the minimum set of files required. The *constant* directory contains a full description of the mesh in the subdirectory *polyMesh* and files specifying physical properties for the application concerned.
The *system* directory contains the dictionary associated with the solution procedure; that is, at least, *controlDict*, *fvSchemes* and *fvSolutions*. In the *controlDict* run control parameters are set including start/end time, time step and parameters for data output; in *fvSchemes* discretisation schemes used in the solution are specified and they may be selected at run-time; finally, in *fvSolution* the equation solvers, tolerances and other algorithm controls are set for the run.
A *time* directory initial values and boundary conditions that the user must specify to define the problem; or, results written to file by OpenFOAM.

The cases studied in this work are organised as reported below:

1. pre-surgery:

- laminar;
- turbulent: $k - \omega - SST$;
- turbulent: $k_T - k_L - \omega$;

2. post-surgery:

- laminar;
- turbulent: $k - \omega - SST$;
- turbulent: $k_T - k_L - \omega$.

The *constant* directory contains the meshes generated by the utility *snappyHexMesh* for the two different geometries. Here are specified also the type of simulation, laminar or turbulent, and the turbulence model used, together with physical properties, e.g. *transportProperties*.
In *controlDict*, positioned in the *system* directory, is specified the case solver, i.e. *simpleFoam*. To be more brief the *fvSchemes* and *fvSolutions* dictionaries are not discussed here.

This part of the work is done exploiting Cineca HPC resources: PLX and Fermi[2]. To gain the best advantages from each machines, PLX is used to create the meshes while simulations run on Fermi.

4.2.1 Mesh generation

The resulting meshes are a compromise between the need of a well resolved computational domain and the possibility of post-processing the simulation results on personal computers yet (In this case 8 *BG* of RAM are required). This limitation is thought to allow surgeons to post-process the results without the further effort to interface with CINECA facility.
The mesh generation starts from the creation of the background mesh

[2]For details on Cineca HPC SuperComputing Applications and Innovation see the webpage http://www.cineca.it

through the OpenFOAM utility *blockMesh*. A box composed by hexes surrounding the whole geometry is created, figure D.1. It is important to underline that the mesh consists of all the patient's head and the box walls are distant enough from him. In this way, the flow in the nasal cavity should be not affected by the boundary conditions.
Then, the *snappyHexMesh* utility is used to adjust the existing mesh to the surface described by the *stl* files.
SnappyHexMesh consists of three steps:

1. *castellatedMesh*. The initial block mesh is refined according to surface and volumetric refinement settings supplied by the user.

2. *snap*. Patch faces are projected onto the surface geometry to remove the jagged castellated surfaces from the mesh.

3. *addLayers*. One or more layers of hexahedral cells are added to a specified set of boundary patches. In this case, the adding layers phase acts only on the nasal cavity. Layer addition on a specified part of the geometry is one of the reasons for splitting the geometry in more than one *stl* files.

In appendix D are reported and explained all the dictionaries necessary to create the mesh.

The mesh for the pre-surgery simulations have the characteristics resumed below. Note that walls are supposed to be solid. The post-surgery mesh is almost equal, hence its *checkMesh* is not reported here.

```
Time = 0

Mesh stats
    points:           9034265
    faces:            24647734
    internal faces:   23873141
    cells:            7868708
    boundary patches: 10
    point zones:      0
    face zones:       0
    cell zones:       0

Overall number of cells of each type:
    hexahedra:     6451582
    prisms:        111017
```

```
    wedges:         4
    pyramids:       0
    tet wedges:     3333
    tetrahedra:     91
    polyhedra:      1302681

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces ...
    Patch               Faces    Points   Surface topology
    bottomWall          10211    10937    ok (non-closed singly connected)
    topWall             7680     7857     ok (non-closed singly connected)
    backWall            5120     5265     ok (non-closed singly connected)
    frontWall           5120     5265     ok (non-closed singly connected)
    leftWall            6144     6305     ok (non-closed singly connected)
    rightWall           6144     6305     ok (non-closed singly connected)
    p1_vcg              197493   219794   ok (non-closed singly connected)
    p2_vcg              18442    19020    ok (non-closed singly connected)
    NasalCavity_vcg     516930   621334   multiply connected (shared edge)
    throat              1309     1365     ok (non-closed singly connected)
  <<Writing 34 conflicting points to set nonManifoldPoints

Checking geometry...
    Overall domain bounding box (-0.1 -0.11 -0.07) (0.1 0.13 0.09)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (-1.1276086e-15 -2.2151533e-15 -2.8376956e-14) OK.
    Max cell openness = 1.4092531e-15 OK.
    Max aspect ratio = 52.871729 OK.
    Minumum face area = 1.1756992e-10. Maximum face area = 7.4335653e-06.
Face area magnitudes OK.
    Min volume = 3.3546727e-14. Max volume = 1.7759361e-08.
    Total volume = 0.0050965915.  Cell volumes OK.
    Mesh non-orthogonality Max: 59.998406 average: 10.113049
    Non-orthogonality check OK.
    Face pyramids OK.
    Max skewness = 3.8375845 OK.
    Coupled point location match (average 0) OK.

Mesh OK.
End
```

As can be seen from the code, the mesh obeys to all quality requirements. In figure 4.2 is showed a global view of the mesh; the darkest region is where the mesh is more refined.

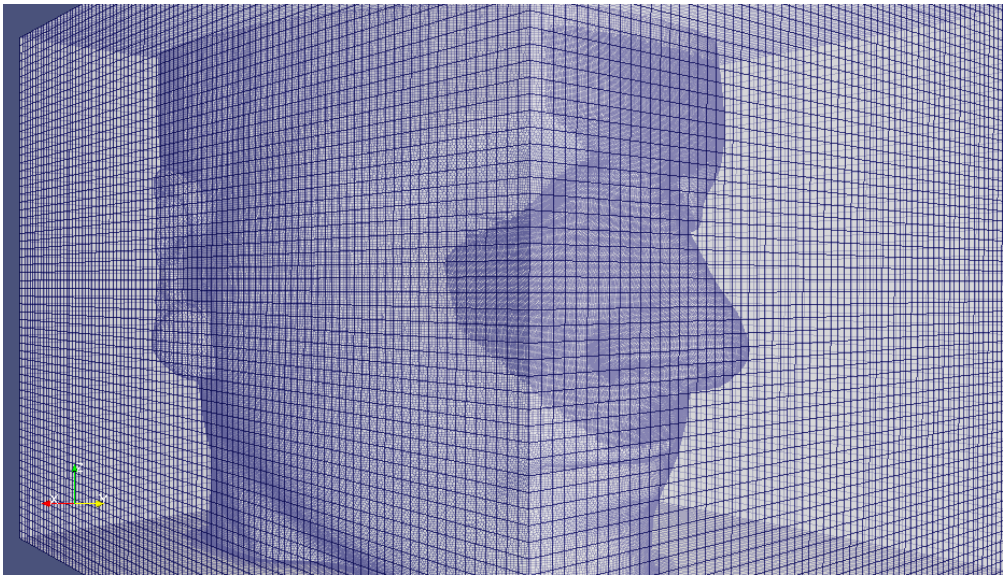


Figure 4.2: Pre-surgery mesh

4.2.2 Initial and boundary condition

Note that the OpenFOAM fields must always be initialised, even when the solution does not strictly require it, as in steady-state problems. The initial and boundary conditions for each case studied are specified below.

Laminar simulation

For the laminar simulation only pressure and velocity fields need to be specified through boundary and initial conditions. The aim is to simulate a quiet restful inspiration; hence, a pressure difference between the wall of the box and the throat is set. This condition was chosen to better mimics the physics of the problem, in fact the resulting flow is more realistic with a Δp imposed, rather than the imposition of a velocity profile at the nostrils. Note that the pressure reported in OpenFOAM is not the *real* pressure, but it is the quantity p/ρ $[m^2/s^2]$, where ρ is the fluid density, in this case $\rho = 1.225\ kg/m^3$. Initial pressure in the nasal cavity is set to zero.
For the velocity field a *zeroGradient* condition is applied to the throat section; velocity is set to zero to the walls to apply no-slip and no-penetration conditions; initial velocity is set to zero too.
In general, p and U dictionaries are contained in the *0* time directory and

they read as follows.

Pressure

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *//
dimensions      [0 2 -2 0 0 0 0];
internalField   uniform 0;
boundaryField
{
    leftWall
    {
        type            fixedValue;
        value           uniform 130;
    }

......

    topWall
    {
        type            fixedValue;
        value           uniform 130;
    }
    p1_vcg
    {
        type            zeroGradient;
    }
    p2_vcg
    {
        type            zeroGradient;
    }
    NasalCavity_vcg
    {
        type            zeroGradient;
    }
    throat
    {
        type            fixedValue;
        value           uniform 0;
    }
}
```

Velocity

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
```

```
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *//
dimensions      [0 1 -1 0 0 0 0];
internalField   uniform (0 0 0);
boundaryField
{
    leftWall
    {
                        type inletOutlet;
                        inletValue uniform (0 0 0);
                        value uniform (0 0 0);
    }

......

    topWall
    {
                        type inletOutlet;
                        inletValue uniform (0 0 0);
                        value uniform (0 0 0);
    }
    p1_vcg
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    p2_vcg
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    NasalCavity_vcg
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    throat
    {
        type            zeroGradient;
    }
}
```

Turbulent simulations

For the turbulent simulations starting pressure and velocity fields are those from the laminar analysis. Starting from a non zero velocity condition should help the convergence of the turbulent cases. The other fields who need to be initialised vary with the turbulence model.
For $k-\omega-SST$ model the user has to specify k, ω, and *nut* initial and boundary conditions. *nut* is set to zero at wall and at the beginning of simulation in the whole nasal cavity; a *zeroGradient* condition is imposed in the throat section.

Turbulent kinetic energy

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      k;
}
// * * * * * * * * * * * * * * * * * * * * * * * * *//

dimensions      [ 0 2 -2 0 0 0 0 ];

internalField   uniform 1e-6;

boundaryField
{
    leftWall
    {
        type            turbulentIntensityKineticEnergyInlet;
        intensity       0.01;
        value           uniform 1e-6;
    }

.......

    topWall
    {
        type            turbulentIntensityKineticEnergyInlet;
        intensity       0.01;
        value           uniform 1e-6;
    }

.......

    NasalCavity_vcg
    {
        type            fixedValue;
        value           uniform 1e-6;
    }
    throat
    {
        type            zeroGradient;
    }
}
```

Turbulence frequency

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      omega;
}
```

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *//

dimensions      [ 0 0 -1 0 0 0 0 ];

internalField   uniform 1;

boundaryField
{
    leftWall
    {
        type            fixedValue;
        value           uniform 1;
    }

........


    topWall
    {
        type            fixedValue;
        value           uniform 1;
    }

........

    NasalCavity_vcg
    {
        type            fixedValue;
        value           uniform 1;
    }
    throat
    {
        type            zeroGradient;
    }
}
```

For $k_T - k_L - \omega$ model there is a further dictionary needed, the *kl* one. While *kt*, the turbulent kinetic energy, corresponds to the classical k dictionary.

Laminar kinetic energy

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      kl;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *//

dimensions      [ 0 2 -2 0 0 0 0 ];
```

```
internalField   uniform 1e-6;

boundaryField
{
    leftWall
    {
        type            fixedValue;
        value           uniform 1e-6;
    }

........

    topWall
    {
        type            fixedValue;
        value           uniform 1e-6;
    }

........

    NasalCavity_vcg
    {
        type            fixedValue;
        value           uniform 1e-6;
    }
    throat
    {
        type            zeroGradient;
    }
}
```

5 Results

The chapter is organised into two sections, the first regarding numerical turbulence models applied to the geometry under investigation and the second comparing the results between pre and post virtual surgery.
All the simulations are performed through the open source platform *OpenFOAM* on the meshes generated by its tool *snappyHexMesh*, see appendix D for details. The steady-state condition is adopted like in the works by Martonen et al. [19], Croce et al. [5], Hoerschler et al. [12], Tan et al. [36] and Zhu et al. [48]. Hence, the OpenFOAM solver *simpleFoam* is used, that is a steady-state solver for incompressible, turbulent flow.

5.1 Laminar and turbulence models

In this section the results for laminar and turbulent simulations are compared; however, the attention is given to turbulence modelling.
As said in section 3.2, the most suitable turbulence models for this flow are the SST $k-\omega$ model and the $k_T - k_L - \omega$ model. To study these models only the pre-surgery case is considered.
Note that the little blank spaces appearing in some slices are only a consequence of the *slice* filter in ParaView; geometry is continuous as can be tested with an *extract cell by region* filter.

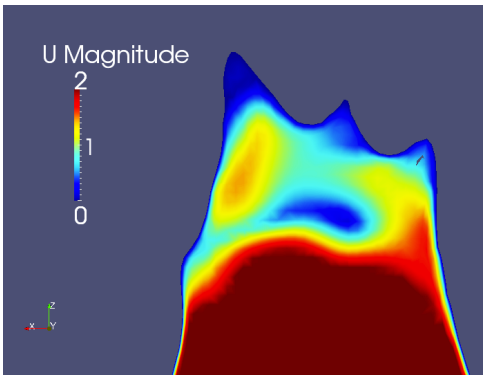
5.1.1 Velocity and Pressure Fields

Results in terms of velocity and pressure fields are very similar among the different simulations. From one model to another the distribution changes only locally, in particular in the region after the obstruction and at the beginning of the pharynx.
For clearness, in figure 5.1 is highlighted the section represented in figures 5.2 and 5.3.

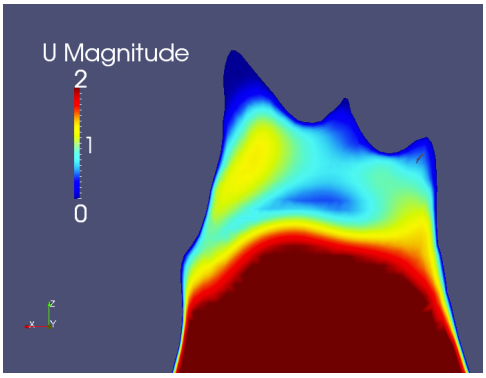
As can be seen from the figure 5.2, the $k-\omega-SST$ results present a lower velocity modules where the flow changes its direction in the pharynx; instead the velocity magnitudes for the other two models are almost equal. Velocity is slightly higher in the transitional case than in the laminar one. This difference is supported by the study of ν_T behaviour, see figure 5.3. In fact for the $k_T - k_L - \omega$ model, ν_T in this region is almost zero, while for the $k - \omega - SST$, ν_T reaches its maximum. Such ν_T values, for the $k_\omega - SST$ model, demonstrate the presence of turbulence in this region, affecting its velocity field.

Figure 5.1: Coronal section highlighted in the nasal cavity.

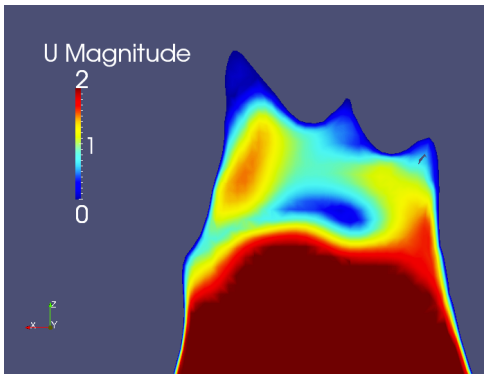
These differences are also visible in terms of the velocity component in the mean flow direction, U_Y, as reported in the sagittal views in the rear part of the nasal cavity, see figure 5.4. In figure 5.4(a) the section under investigation is specified.



(a) *Laminar model.*

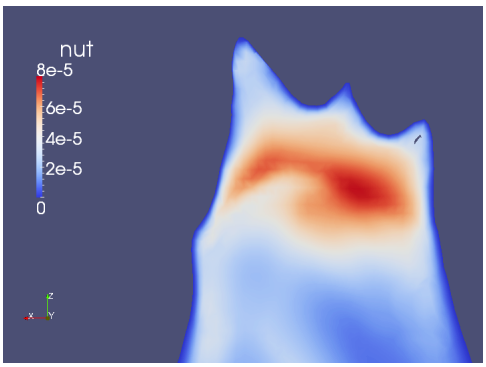


(b) $k-\omega-SST$ *model.*

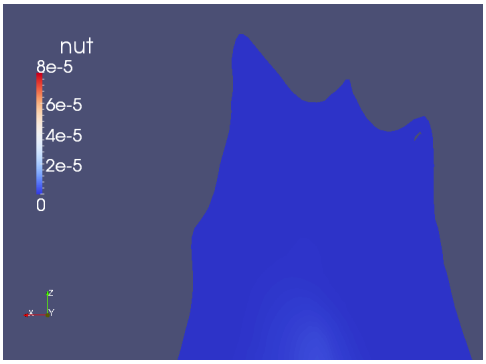


(c) $k_T-k_L-\omega$ *model.*

Figure 5.2: Velocity magnitude $[m/s]$, coronal section.

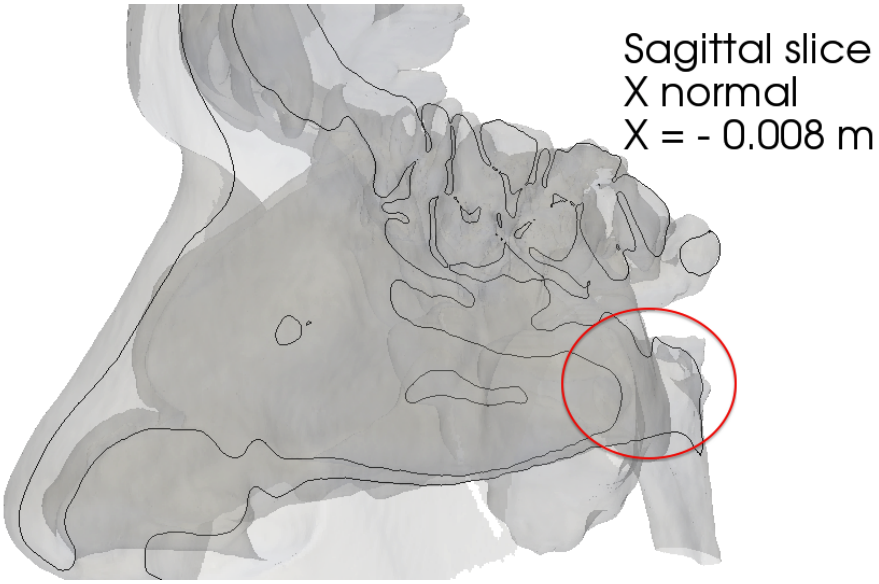


(a) $k-\omega-SST$ *model.*

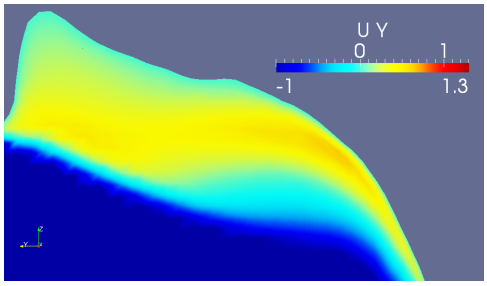


(b) $k_T-k_L-\omega$ *model.*

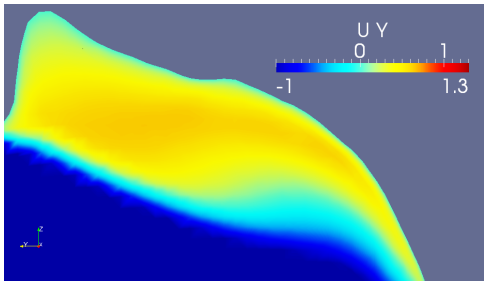
Figure 5.3: ν_T $[m^2/s]$, coronal section.



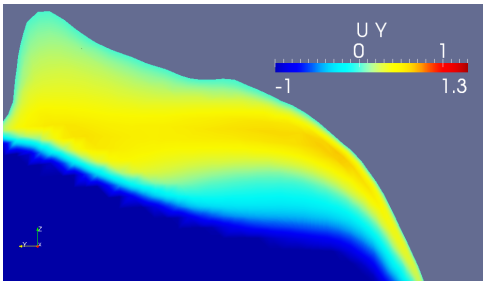
(a) *Sagittal section highlighted in the nasal cavity..*



(b) *Laminar model.*



(c) $k-\omega-SST$ *model.*



(d) $k_T-k_L-\omega$ *model.*

Figure 5.4: U_Y velocity component $[m/s]$, sagittal section.

In the section of figure 5.4 a recirculation zone appears. Its intensity is higher in the $k-\omega-SST$ simulation, in the laminar case it is not so remarkable, while the transitional one presents intermediate values.
Notice that U_Y velocity during inspiration has the minus sign because of

the reference system orientation.

The other important region that needs to be investigated is where turbulent fluctuations with respect to the mean flow are the highest, that is the zone circled in figure 5.5. The region is located just before the pharynx, on the very right of the patient, the septal deviation side. In figure 5.6 can be highlighted a difference in the mean flow component of the velocity, U_Y. In this case the $k-\omega-SST$ model shows lower velocity (in module) with respect to the other two simulations.
These results suggest that in terms of velocity field, the $k_T - k_L - \omega$ case behaves like the laminar one. This fact implies the presence of low turbulence levels for transitional simulation. Thus, here, the $k_T - k_L - \omega$ behaviour is correctly very similar to the laminar one. Anyway, this consideration does not mean that laminar simulation is preferable. This is due to the presence of fluctuations which assume significant values in some regions of the nasal cavity. Their identification could be helpful to have a more complete description of the dynamics of the problem; especially if studies with higher pressure differences are carried out.

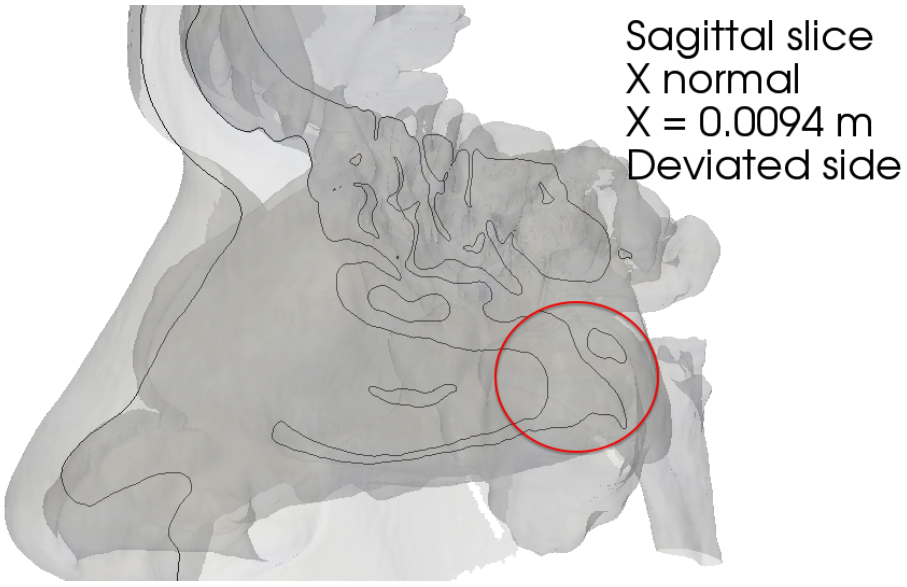
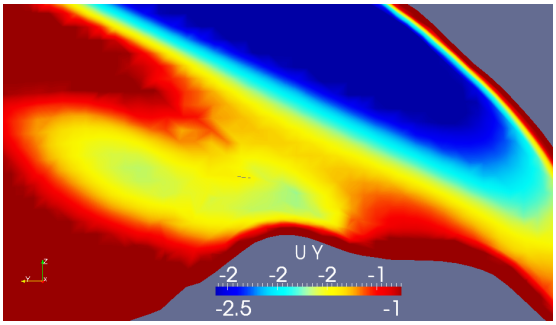
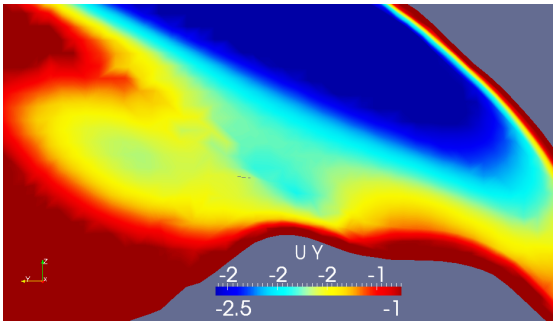


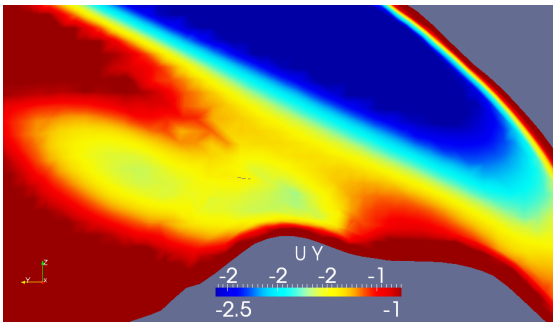
Figure 5.5: Sagittal section highlighted in the nasal cavity.



(a) *Laminar model.*



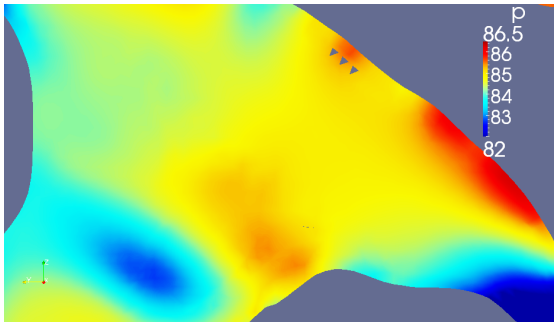
(b) $k-\omega-SST$ *model.*



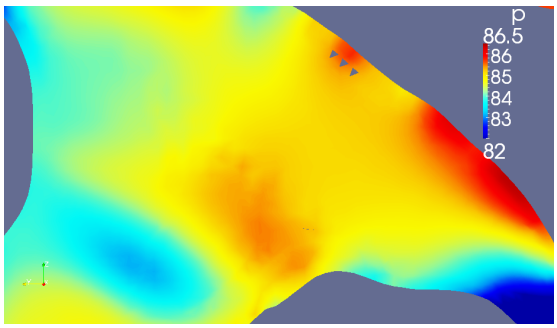
(c) $k_T-k_L-\omega$ *model.*

Figure 5.6: U_Y velocity component $[m/s]$, sagittal section.

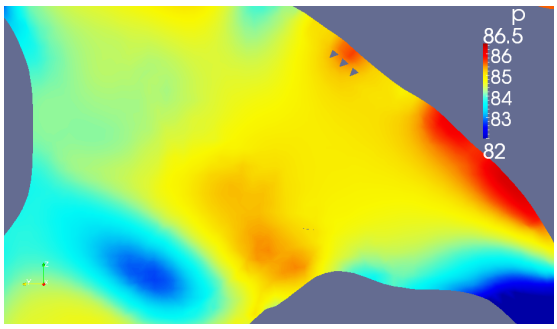
The pressure field does not suffer too much the variation of turbulence model, only local differences are noticeable in some regions of the flow (figure 5.7) such as in the highest k location, identifiable in figure 5.5. Remember that these pressure values are p/ρ $[m^2/s^2]$.
Another way to compare the three types of simulations is through wall shear stresses, as done in the next paragraph.



(a) *Laminar model.*



(b) $k-\omega-SST$ *model.*

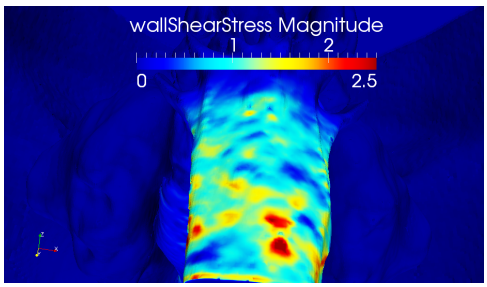


(c) $k_T - k_L - \omega$ *model.*

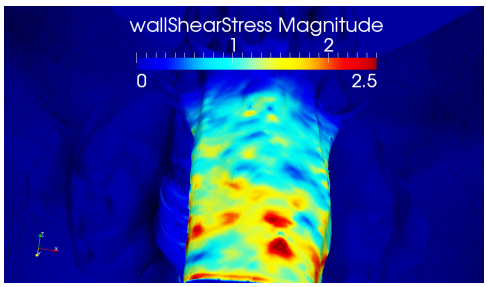
Figure 5.7: Pressure field $[m^2/s^2]$, sagittal section.

5.1.2 Wall Shear Stress

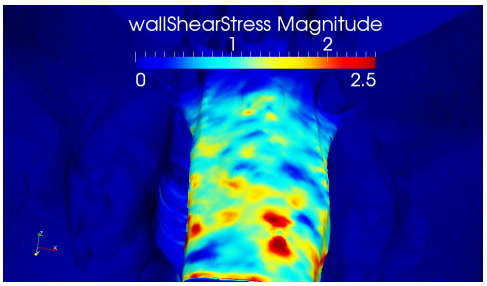
Since the flow is transitional, or, in other words, almost laminar, it is worth comparing turbulent simulation results with the laminar case in terms of wall shear stresses, too. In fact, through wall shear stresses some differences near wall are noticeable, especially between $k-\omega-SST$ and $k_T-k_L-\omega$ models. Studying this quantity, remember that *WSSs* are directly the stresses on the nasal cavity and they should affect breathing quality, i.e. causing mucosal inflammations. The stresses are concentrated in the highest velocity regions, that is the nasal valve and the pharynx. Wall shear stresses computed for the laminar case and for the two turbulence models are reported in figures 5.8 and 5.9. Moreover, in table 5.1 are resumed the minimum and maximum values in the three simulations. Note that these values are all divided by density, i.e. τ_W/ρ. For turbulent or transitional cases, wall shear stresses are slightly higher than the laminar results.



(a) *Laminar model.*



(b) $k-\omega-SST$ *model.*



(c) $k_T-k_L-\omega$ *model.*

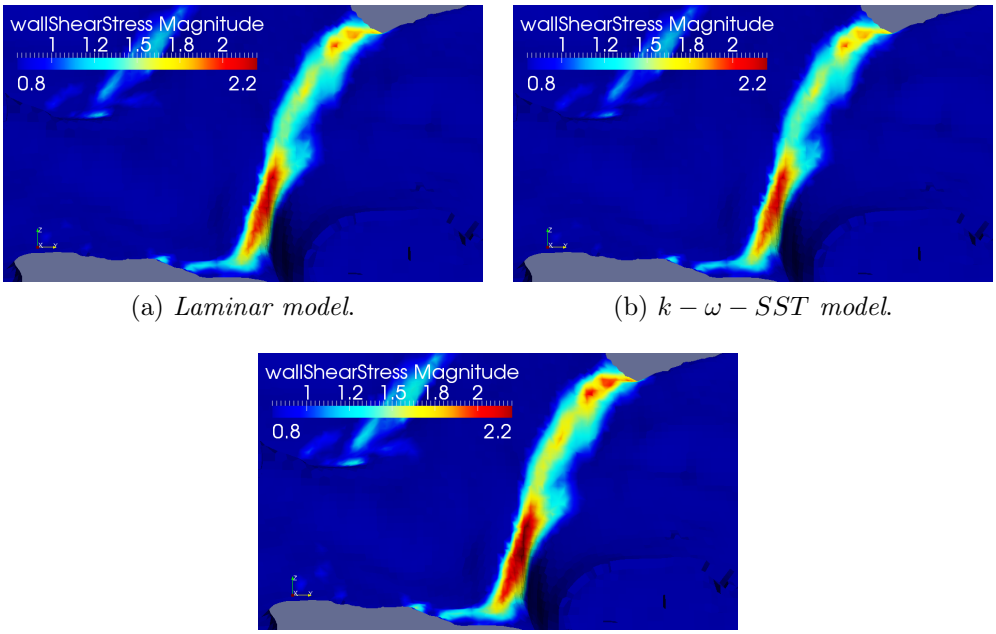
Figure 5.8: Wall Shear Stresses, particular of the pharynx.

As can be seen from figures 5.8 and 5.9, WSS distribution in laminar and $k_T-k_L-\omega$ cases are very similar, with higher peak values in the second model. In the $k-\omega-SST$ simulation stresses are more spread in

the pharynx but with similar magnitudes with respect to the other results. Finally, it is important to underline that these results are in agreement with the works by Lee et al. [17], Han et al. [36] and Chen et al. [3].

	Laminar	$\mathbf{k-\omega-SSTmodel}$	$\mathbf{k_T-k_L-\omega}$ model
$\mathbf{WSS_{magnitude}}$ $[Pa]$	$[0 \div 3.42]$	$[0 \div 3.36]$	$[0 \div 4.37]$
$\mathbf{WSS_x}$ $[Pa]$	$[-1.77 \div 1.82]$	$[-1.77 \div 2.10]$	$[-4.15 \div 2.04]$
$\mathbf{WSS_y}$ $[Pa]$	$[-1.59 \div 1.82]$	$[-1.86 \div 2.10]$	$[-3.89 \div 3.60]$
$\mathbf{WSS_z}$ $[Pa]$	$[-1.35 \div 3.38]$	$[-1.34 \div 3.32]$	$[-1.44 \div 3.60]$

Table 5.1: Wall shear stresses



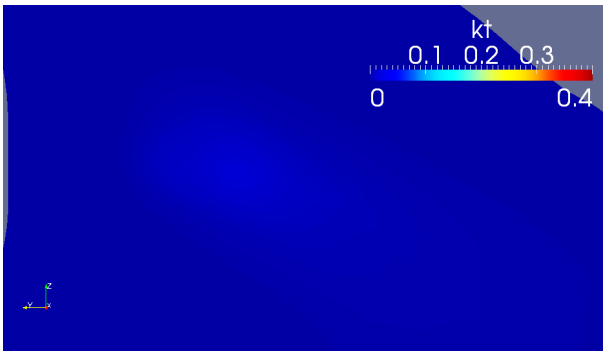
(a) *Laminar model.* (b) $k-\omega-SST$ *model.*

(c) $k_T-k_L-\omega$ *model.*

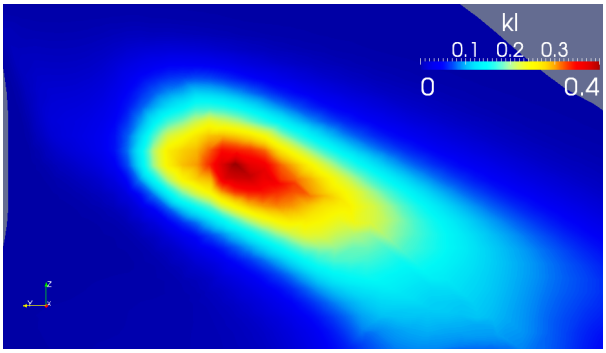
Figure 5.9: Wall Shear Stresses, particular of the nasal valve.

To conclude, for the case in which a pressure difference is imposed corresponding to a restful breath, turbulence is not fully developed and the discrepancies between laminar and turbulent simulations are not so remarkable with respect to mean flow quantities.
A more intense inspiration would have effectively underlined the presence of turbulent motion. In fact, considering the $k_T-k_L-\omega$ model, fluctuations are mostly due to instabilities in laminar kinetic energy and not

strictly speaking turbulent fluctuations. In figure 5.10 is reported the kinetic energy distributions in a particular of the rear of nasal cavity.



(a) *Turbulent kinetic energy.*



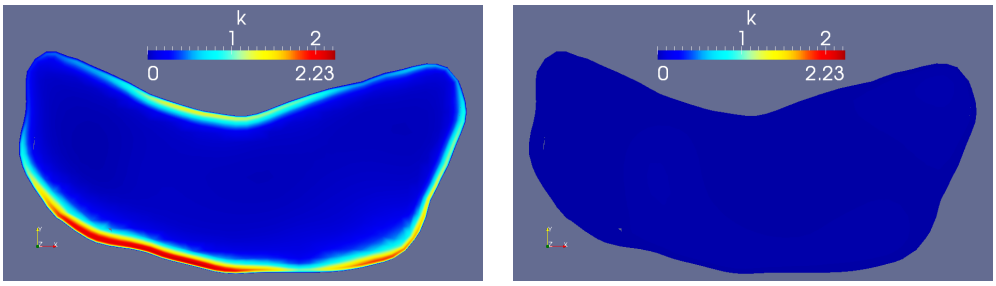
(b) *Laminar kinetic energy.*

Figure 5.10: Turbulent and laminar fluctuations $[m^2/s^2]$ with $k_T - k_L - \omega$ model. Sagittal section, refer to figure 5.5 for the position in the nasal cavity.

5.1.3 Turbulent Kinetic Energy

The core of the comparison between turbulence models resides in the behaviour of turbulent kinetic energy, that is the turbulent fluctuations with respect to the mean flow. For the SST $k-\omega$ simulation, the turbulent kinetic energy has the maximum in 2.23 m^2/s^2. Results from $k_T - k_L - \omega$ model show a peak value of 0.42 m^2/s^2. However, considering only these two values could be misleading; in fact, the most important issue for the SST $k - \omega$ model is the appearance of turbulent fluctuation in the rear part of the pharynx, see figure 5.11. Moving towards the throat these fluctuations reach the peak value reported before. Note that in the rest of

the nasal cavity turbulent kinetic energy values do not overcome the unity. Thus, this very high peak could be attributed to an under-resolution of the grid or to an incapacity of the turbulence model to represents correctly the transitional situation.

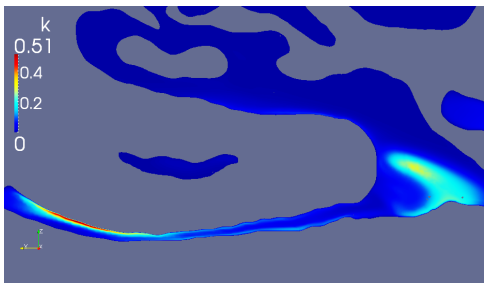


(a) $k-\omega-SST$ *model.* (b) $k_T-k_L-\omega$ *model.*

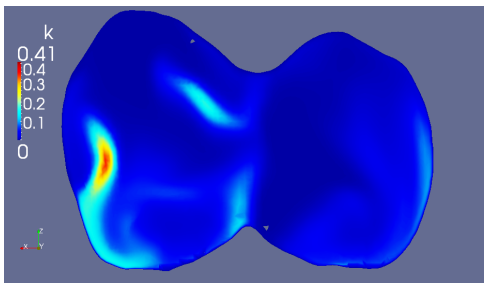
Figure 5.11: Turbulent fluctuations with turbulence models, throat section

Considering the $k_T-k_L-\omega$ simulation, fluctuations are concentrated in the region after the septal deviation and they decrease along the pharynx, as can be seen in figures 5.12(c) and (d). This aspect will be deepened in section 5.2.
In the other region of the nasal cavity the turbulent kinetic energy is practically zero.
The intensity of fluctuations for the $k-\omega-SST$ simulation in this region is less than the transitional case, see figure 5.12(a) and (b). In the sagittal section fluctuations in the inferior turbinate are also visible.
In fact, studying in details the $k-\omega-SST$ simulation, it is found that turbulent fluctuations appear not only after the obstruction, but also in other regions of the nasal cavity such as inferior and superior turbinates, and in all the pharynx. Figure 5.13 supports this statement.

To conclude this section, it could be said that the transitional model $k_T-k_L-\omega$ should be more reliable in describing nasal cavity flows, because of its ability to discern between laminar and turbulent fluctuations. Instead, the other model shows those regions with turbulent fluctuations which seem not fully consistent with the physics of the problem.

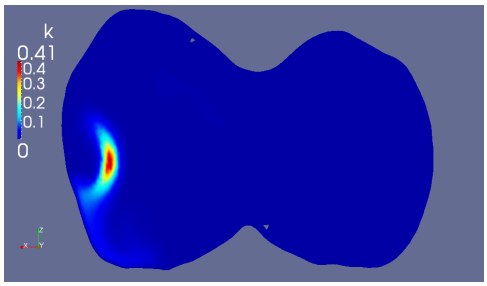


(a) $k-\omega-SST$ *model, sagittal view.*

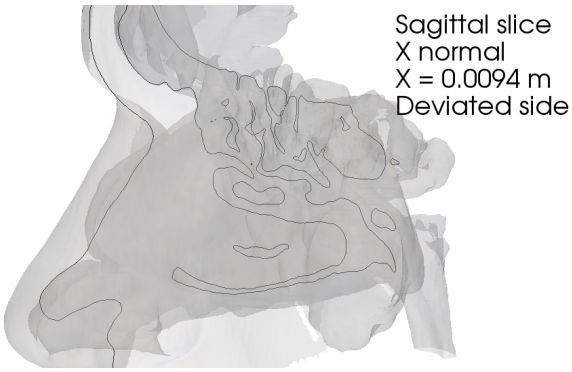


(b) $k-\omega-SST$ *model, coronal view.*

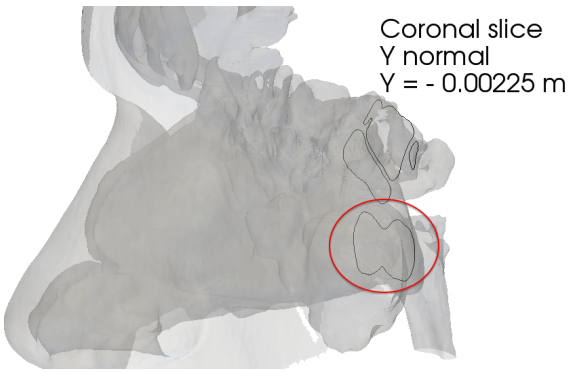
(c) $k_T-k_L-\omega$ *model, sagittal view.*



(d) $k_T-k_L-\omega$ *model, coronal view.*

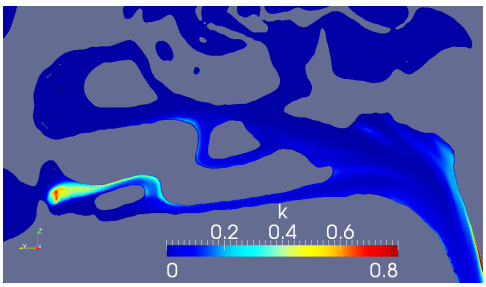


(e) *Position of the sagittal view.*

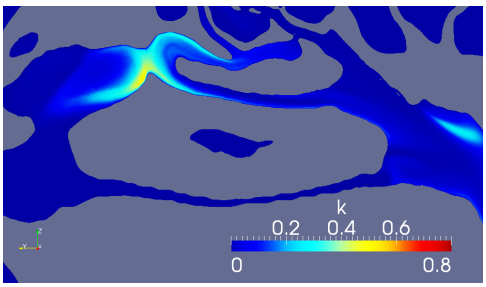


(f) *Position of the coronal view.*

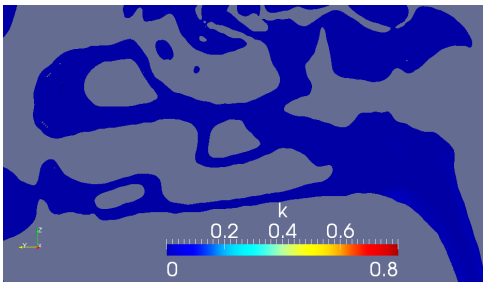
Figure 5.12: Turbulent kinetic energy $[m^2/s^2]$.



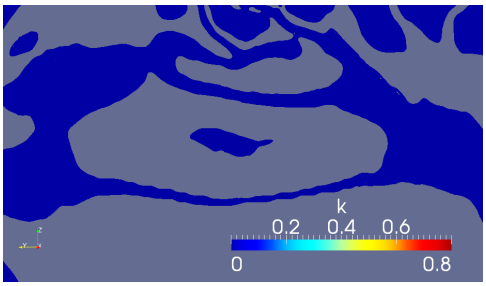
(a) $k-\omega-SST$ *model, sagittal view.*



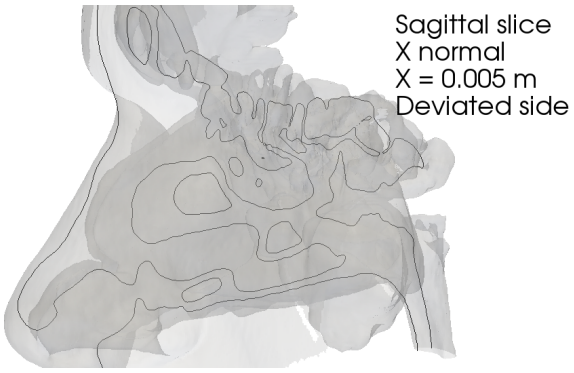
(b) $k-\omega-SST$ *model, sagittal view.*



(c) $k_T-k_L-\omega$ *model, sagittal view.*



(d) $k_T-k_L-\omega$ *model, sagittal view.*



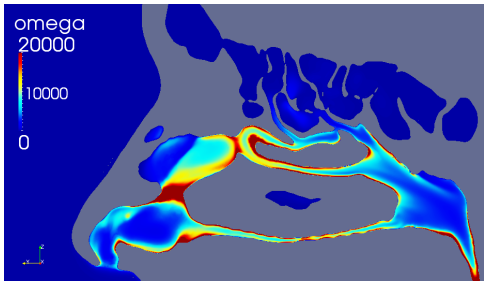
(e) *Position of the sagittal view, deviated side.*

(f) *Position of the sagittal view.*

Figure 5.13: Turbulent kinetic energy $[m^2/s^2]$.

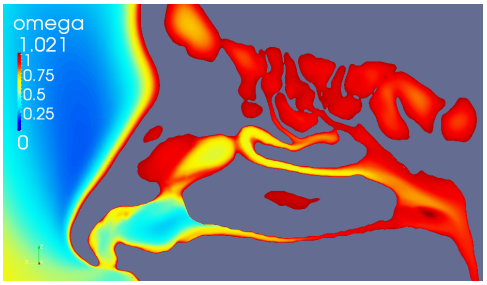
5.1.4 Turbulence Frequency ω

The quantity ω, defined as the turbulence frequency or the specific dissipation, confirms the non-fully reliability of the $k-\omega-SST$ model with respect to the flow involved in this study. The simulation done with $k_T-k_L-\omega$ model shows reasonable values and distributions of ω in accordance with the turbulent fluctuations. In such simulation the turbulence frequency reaches its maximum after the septal deviation, $83s^{-1}$, and behaves similarly to the turbulent kinetic energy in the rest of the nasal cavity, as will be seen in figure 5.27.
Results for $k-\omega-SST$ analysis show values for ω that need to be rescaled (because of grid under-resolution). Turbulence frequency is very high near wall in almost all the nasal cavity, for instance see figures 5.14. Hence, this model represents, perhaps incorrectly, a turbulent situation that does not exist in reality. On the other hand, $k_T-k_L-\omega$ model shows little turbulence frequency values according to the transitional nature of this flow. That is, $k_T-k_L-\omega$ simulation maintains "switched-off" turbulence to represent the *quasi*-laminar situation. Thus, also these quantity suggests that here the most suitable turbulence model is the $k_T-k_L-\omega$ one.



(a) $k-\omega-SST$ *model, sagittal view.*

(b) $k_T-k_L-\omega$ *model, sagittal view.*



(c) $k_T-k_L-\omega$ *model, not rescaled values.*

Figure 5.14: Turbulence frequency ω $[s^{-1}]$, section from figure 5.13(f).

5.2 Pre- and Post-Virtual Surgery

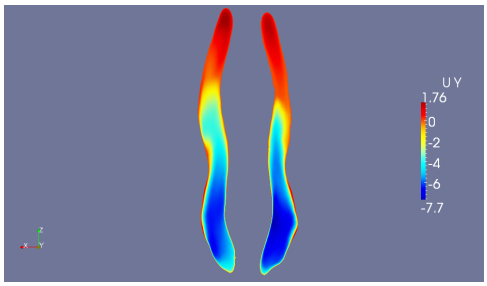
The turbulence models described in the previous section have been applied to the pre-surgery case and to the post-surgery one. To be more concise, the comparison between the two cases refers to the results of the $k_T - k_L - \omega$ simulation only, except for the flow rates computation.
As already seen, the nasal pathology under investigation in this work is a septal deviation. From an anatomical point of view, a deviated septum is shifted to one side of the nose causing the narrowing of the nasal passage, imbalance in flow rate and sometimes infections. The clinical consequences of this pathology should depend on the type and severity of the deviation (paragraph 2.2.2).

5.2.1 Global Quantities

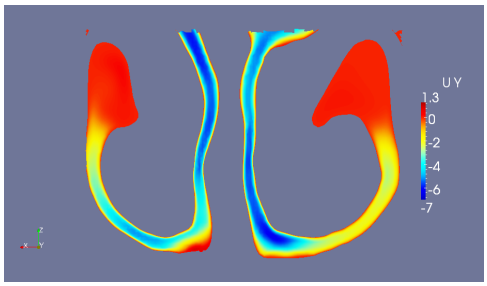
To investigate the global effects of this specific septal deviation and septoplasty, quantities like flow rates and pressure drops are studied firstly.

Flow Rates

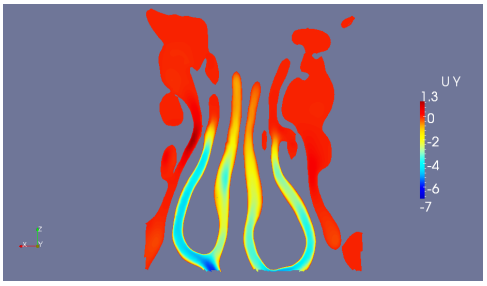
Since the flow under investigation is steady, the flow rate will be constant in all the nasal cavity. For simplicity, the total flow rate is computed in a throat section for pre- and post-surgery cases. Other relevant region where flow rates computation is useful are in proximity of the nostrils, at inferior, medium and superior turbinates, see figures 5.15 and 5.16. In table 5.2 are resumed flow rates for each case indicated above. To reduce computational errors flow rates for the middle and superior turbinates are computed together. In the analysis of these results the reader has to take into account some uncertainty of the procedure and computation. Uncertainty could derive from the fact that the effective component of the velocity vector is not perfectly aligned with the section's normal where flow rates are calculated. Moreover, the integration process in ParaView for middle and superior meatuses is affected by the regions with zero velocity corresponding to ethmoidal cells, figure 5.15(c) and 5.16(c). Hence the sum of the flow rates in the three turbinates slightly differs from the value in the corresponding nostril.



(a) *Nostrils.*

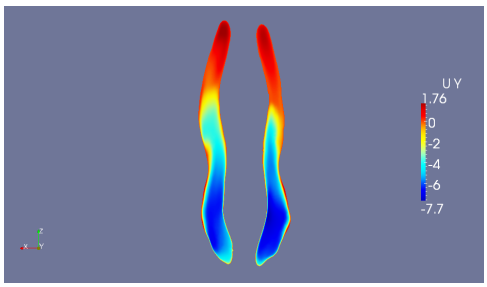


(b) *Inf. turbinates.*

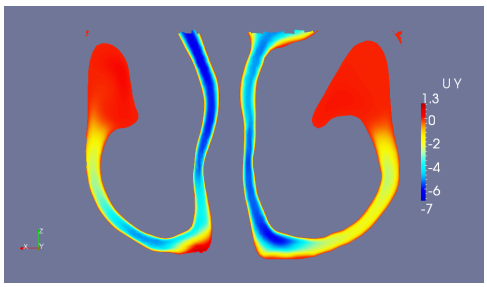


(c) *Med/sup. turbinates.*

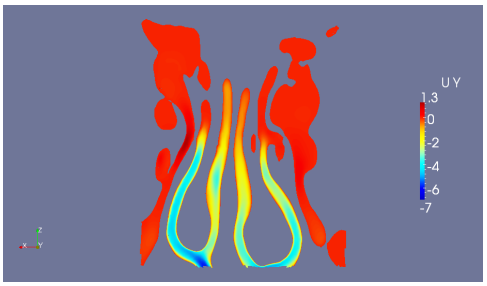
Figure 5.15: Pre-surgery. U_y for flow rates computation.



(a) *Nostrils.*



(b) *Inf. turbinates.*



(c) *Med/sup. turbinates.*

Figure 5.16: Post-surgery. U_y for flow rates computation.

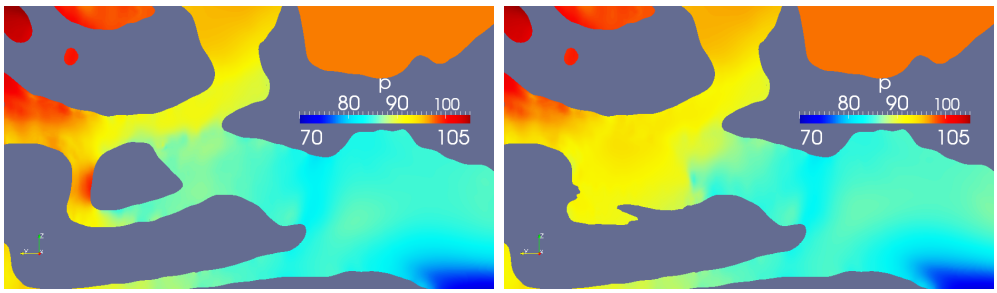
From these flow rate values it can be said that this particular septal deviation would not be so depreciatory, but nothing tells what effects could cause on patient's breath quality and if it would make other physiological implication happen.
A checking with surgeons highlighted that the patient suffered from this nasal breathing stenosis. Therefore, this septal resection does not directly affect flow rates, but should improve the global qualities of the flow because after the real surgery patient's reported symptoms disappeared.
Studying the velocity field would deepen the understanding of flow behaviour. In other words, with this septal deviation, perhaps, there is not the whole blocking of the channel; in such case, flow rates are rightly almost equal. But this situation is enough to make the patient suffering.

		PRE-surgery Flow rates		**POST-surgery Flow rates**	
Sections	**Area** $\mathbf{A}\ [\mathbf{m^2}]$	$\mathbf{k-\omega-SST}$ $\mathbf{Q}$ [l/s]	$\mathbf{k_T-k_L-\omega}$ $\mathbf{Q}$ [l/s]	$\mathbf{k-\omega-SST}$ $\mathbf{Q}$ [l/s]	$\mathbf{k_T-k_L-\omega}$ $\mathbf{Q}$ [l/s]
Throat	$6.31 \cdot 10^{-5}$	0.699	**0.705**	0.701	**0.707**
Left nostril	$1.08 \cdot 10^{-4}$	0.345	**0.347**	0.343	**0.345**
Right nostril	$1.21 \cdot 10^{-4}$	0.354	**0.358**	0.358	**0.363**
Inf. turbinate Left	$1.05 \cdot 10^{-4}$	0.199	**0.192**	0.191	**0.191**
Inf. turbinate Right	$0.83 \cdot 10^{-4}$	0.167	**0.170**	0.170	**0.175**
Med. and sup. turbinate - Left	$3.58 \cdot 10^{-4}$	0.137	**0.140**	0.137	**0.139**
Med. and sup. turbinate - Right	$3.23 \cdot 10^{-4}$	0.176	**0.176**	0.176	**0.175**

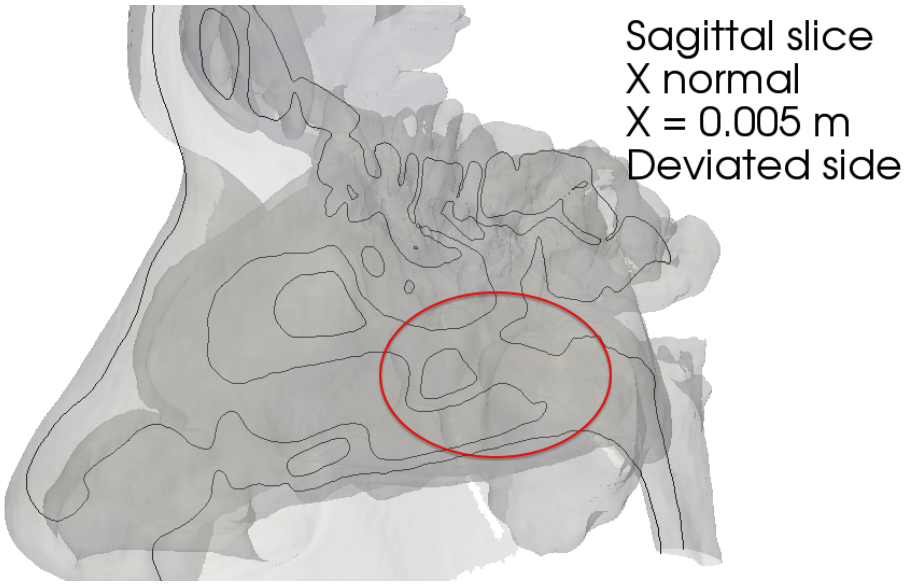
Table 5.2: Flow rates. Left and right are intended for the the patient; septal deviation at right side

Pressure

Consider the pressure field in the septal deviation region, figure 5.17. The user can see an increase of pressure before the obstruction where the flow impact against the obstacle, which does not appear anymore in the operated nose. After virtual surgery the pressure field varies more gradually.



(a) *Pre-surgery.* (b) *Post-virtual surgery.*



(c) *Position in the nasal cavity.*

Figure 5.17: Pressure $[m^2/s^2]$, sagittal view.

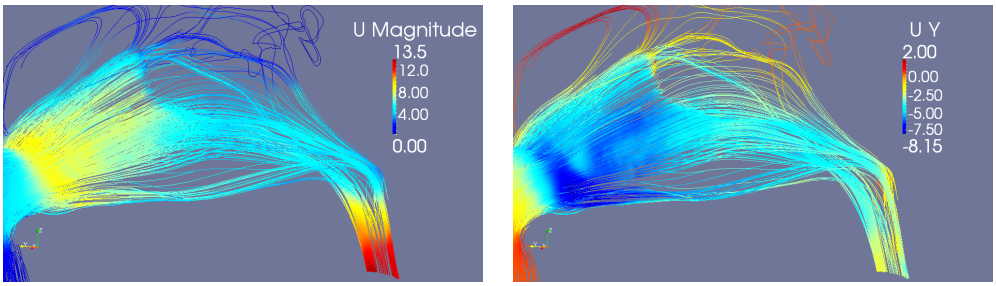
5.2.2 Local Quantities

Since flow rates are not so effective in the evaluation of the consequences of septal resection, the study of velocity fields and turbulent quantity distributions is worth to a better understanding of the physics of the problem.

Velocity Fields

In these section there are presented the results in terms of velocity. First, figure 5.18 recalls the velocity behaviour in the nasal cavity.

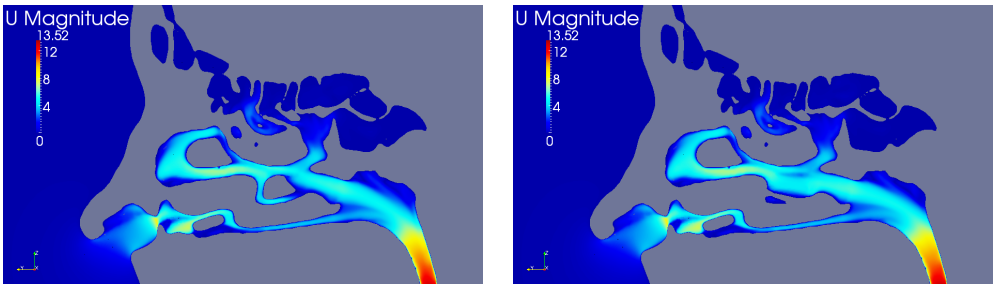
As usual, during inspiration, the flow enters the nasal cavity through the nostrils with a velocity magnitude about 3 m/s, it is accelerated in the nasal valve region until 8 m/s. Going on in the nasal cavity, there is a decrease in mean velocity, the most of the flow passes through the medium meatus, a secondary part passes in the inferior meatus and a very little quantity stands in the olfactive region, that is the superior meatus.



(a) *Post-virtual surgery,* $U_{magnitude}$. (b) *Post-virtual surgery,* U_y.

Figure 5.18: Velocity streamlines.

After the meatuses the right and left flows join again, in the rhinopharynx, bending of almost 90°. Here, it could be observed a further acceleration reaching $5 - 6$ m/s and then the velocity increases through the pharynx. In the throat region there is the highest velocity, about 13 m/s, see 5.19.

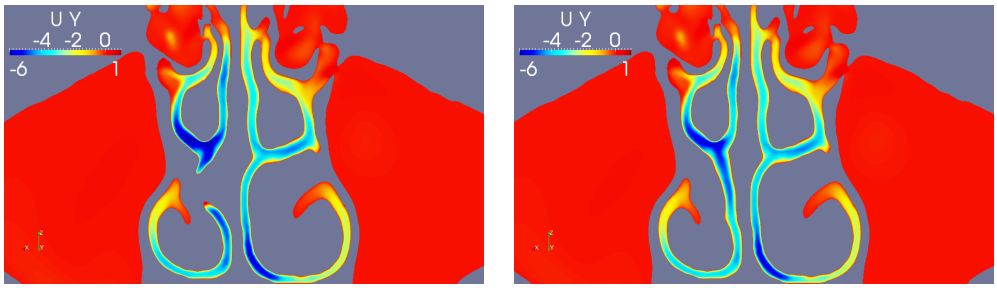


(a) *Pre-surgery.* (b) *Post-virtual surgery.*

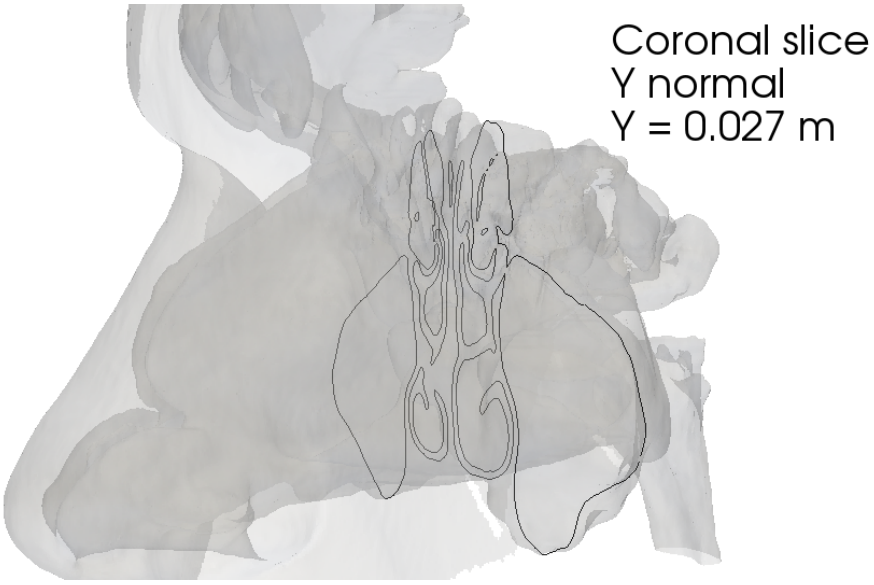
Figure 5.19: Example of velocity magnitude $[m/s]$, sagittal view, location in the nasal cavity highlighted in figure 5.13(e).

Nevertheless, here, the aim is to investigate the local velocity near the septal deviation. As can be seen from figures 5.20 and 5.21, in the

pre-surgery case, the flow is forced to overcome the deviation with an increasing U_y just above the obstacle and a changing U_y direction after the obstruction. These findings are also supported by the path of velocity streamlines 5.22.

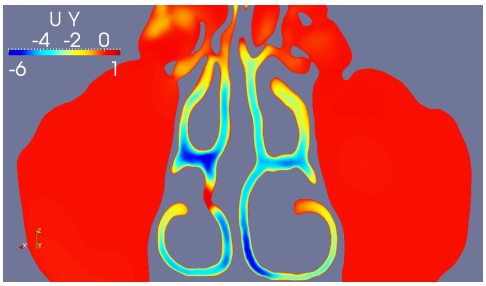


(a) *Pre-surgery.* (b) *Post-virtual surgery.*

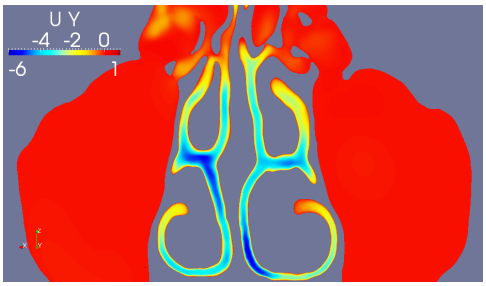


(c) *Position in the nasal cavity.*

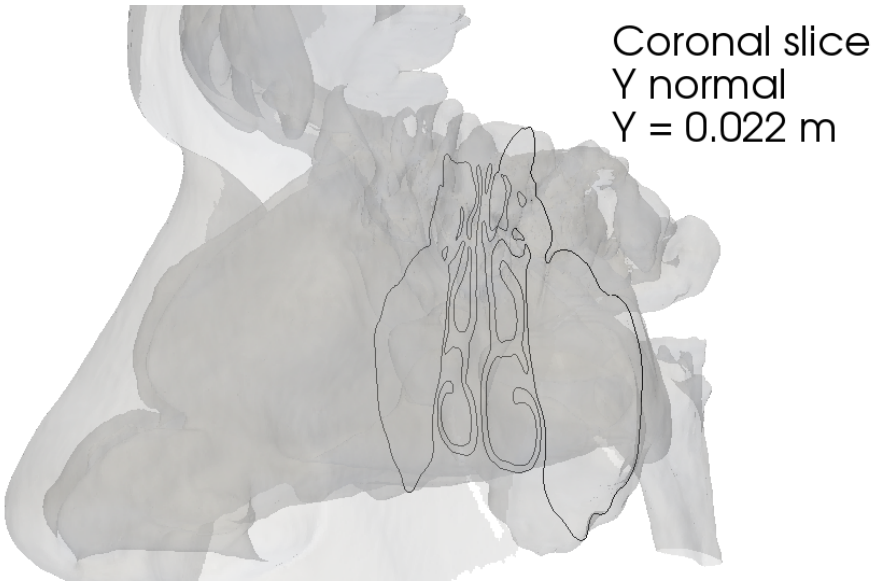
Figure 5.20: U_y at the septal deviation $[m/s]$, coronal section.



(a) *Pre-surgery.*

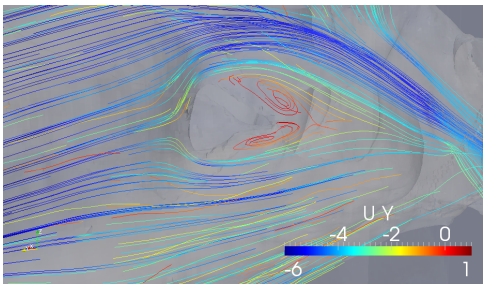


(b) *Post-virtual surgery.*

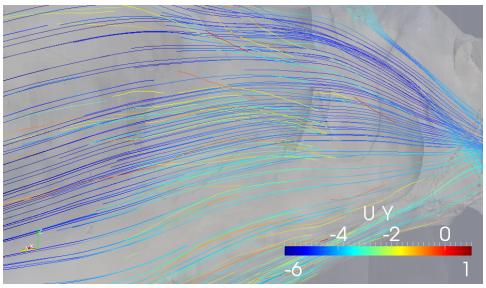


(c) *Position in the nasal cavity.*

Figure 5.21: U_y after the septal deviation $[m/s]$, coronal section.



(a) *Pre-surgery.*

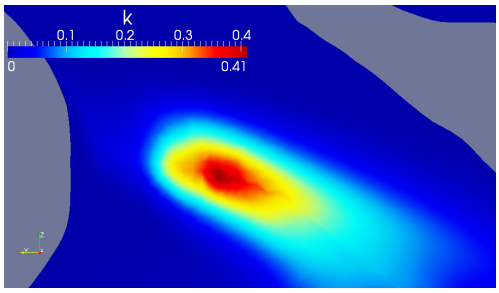


(b) *Post-virtual surgery.*

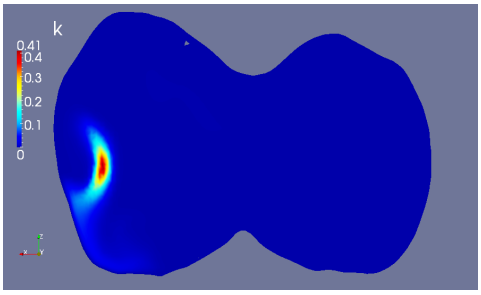
Figure 5.22: U_y streamlines in the obstruction region.

Turbulent Kinetic Energy

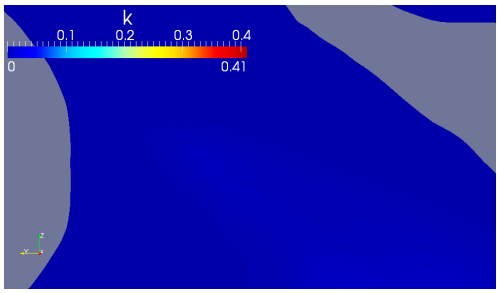
Consider now the turbulent kinetic energy k. As said before, the total turbulent kinetic energy obtained from the $k_T - k_L - \omega$ model is the sum of laminar and turbulent kinetic energy, $k = k_T + k_L$.
Comparing this energy in the two cases, it is evident the reduction of fluctuations after the virtual surgery in the coronal sections where k reaches its maximum. This changing could be interpreted as a reduction in turbulence intensity. In particular, the maximum value of fluctuations passes from 0.42 m^2/s^2 to 0.16 m^2/s^2, as can be seen from figure 5.23 where the two slices are in proximity of maximum k.



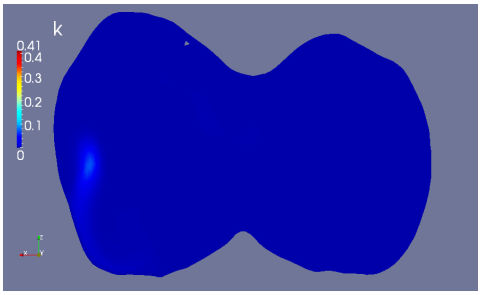
(a) *Pre-surgery, sagittal section 5.5.*



(b) *Pre-surgery, coronal section 5.12(f).*



(c) *Post-virtual surgery, sagittal section 5.5.*

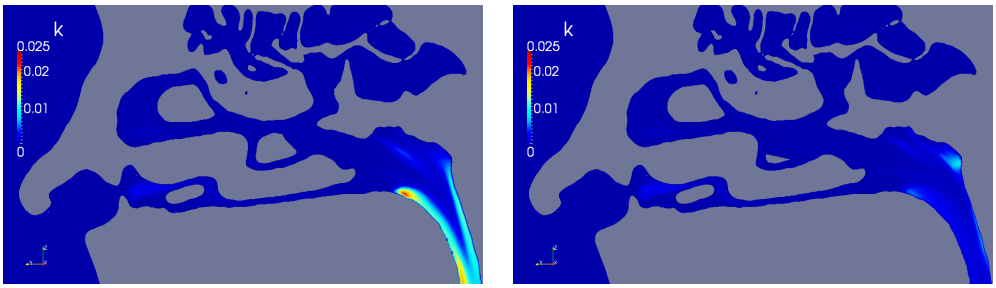


(d) *Post-virtual surgery, coronal section 5.12(f).*

Figure 5.23: Highest turbulent kinetic energy regions $[m^2/s^2]$.

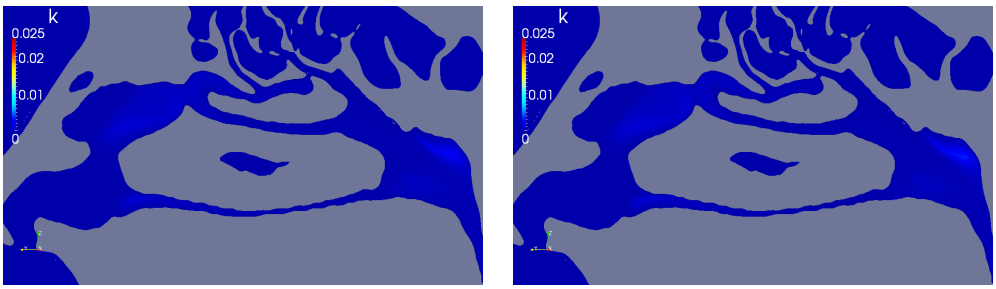
In figures 5.24 and 5.25 the k distributions in two sagittal sections is shown. Turbulent fluctuations are not present in the left side of the nasal cavity, the sane one, and they appear after the obstruction only near wall in the pre-surgery case.
The variation of fluctuations distributions in the rest of nasal cavity is less intuitive, and its representation through 2D slices is more difficult.
The peak value of fluctuations, as already seen, appears for both cases in

the region after the septal deviation (operated or not); then the fluctuations behave in different ways. In pre-surgery case fluctuations are concentrated near the right wall after the obstruction; they decrease moving towards the pharynx (about 10^{-2} order of magnitude) where they appear also in the middle of the channel. This behaviour is maintained until the throat.
After virtual surgery, at the beginning, that is after the resection, fluctuations behave as in the pre-surgery case (but more than halved); going to the pharynx, fluctuations are almost zero near wall, but they appear in centre of the the pharynx with an order of magnitude about 10^{-2} and with values double of the pre-surgery case. Moving to the throat they gradually decrease. Refer to figure 5.26 for more clearness.



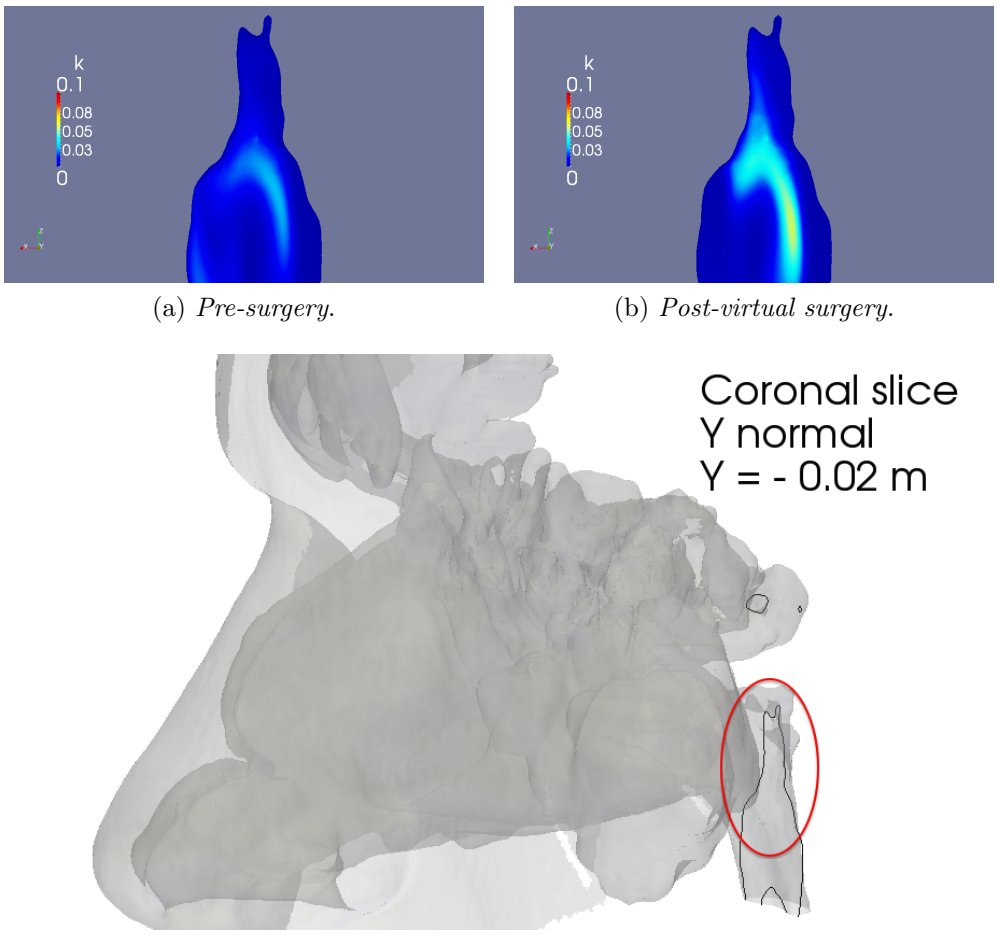
(a) *Pre-surgery.* (b) *Post-virtual surgery.*

Figure 5.24: Turbulent kinetic energy $[m^2/s^2]$; sagittal section, refer to figure 5.13(e).



(a) *Pre-surgery.* (b) *Post-virtual surgery.*

Figure 5.25: Turbulent kinetic energy $[m^2/s^2]$; sagittal section, 5.13(f).



(a) *Pre-surgery.*

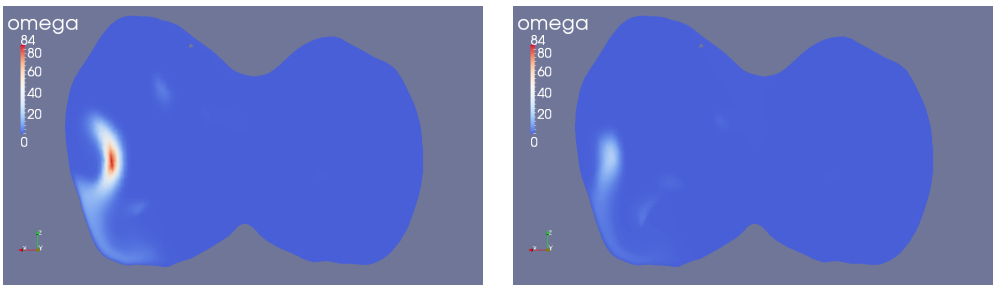
(b) *Post-virtual surgery.*

(c) *Position in the nasal cavity.*

Figure 5.26: Turbulent kinetic energy $[m^2/s^2]$; coronal section.

Turbulence Frequency ω

The turbulence frequency behaves exactly as the turbulent kinetic energy. In pre-surgery case it starts from a peak value about 84 s^{-1} and it decrease through the pharynx. After the virtual surgery the maximum value of ω is reduced to 58 s^{-1} and it follows a similar distribution, see figure 5.27.

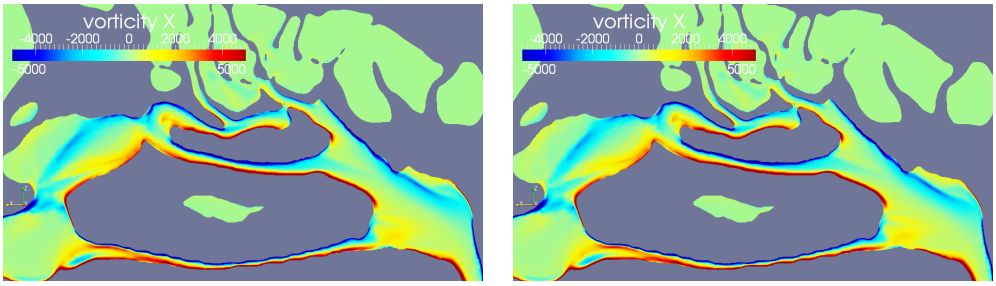


(a) *Pre-surgery.* (b) *Post-virtual surgery.*

Figure 5.27: Turbulence frequency ω $[s^{-1}]$; coronal section, refer to figure 5.12(f).

Vorticity

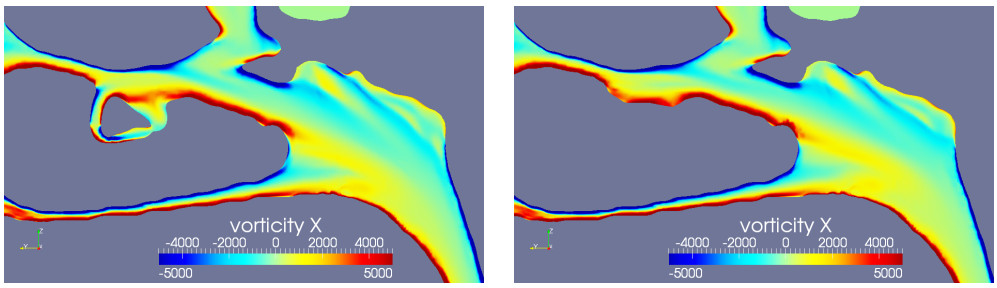
The vorticity, that is $\omega = \nabla \times \mathbf{U}$ (different from turbulence frequency in spite of the same notation), offers the possibility to investigate the flow's shear layers. Neglecting the highest values near the walls, it is interesting to identify separation and recirculation regions in the nasal cavity flow. Figure 5.28 shows the ω_X component of the vorticity, that is the transversal component with with respect to the mean flow direction. Separation regions are noticeable in the nostrils zone and after the turbinates.



(a) *Pre-surgery.* (b) *Post-virtual surgery.*

Figure 5.28: Vorticity component ω_X $[s^{-1}]$; sagittal section.

Figure 5.29(a) highlights the flow separation and recirculation after the septal deviation. This flow behaviour is no longer visible in the post-virtual surgery case where the obstruction was removed, 5.29(b).

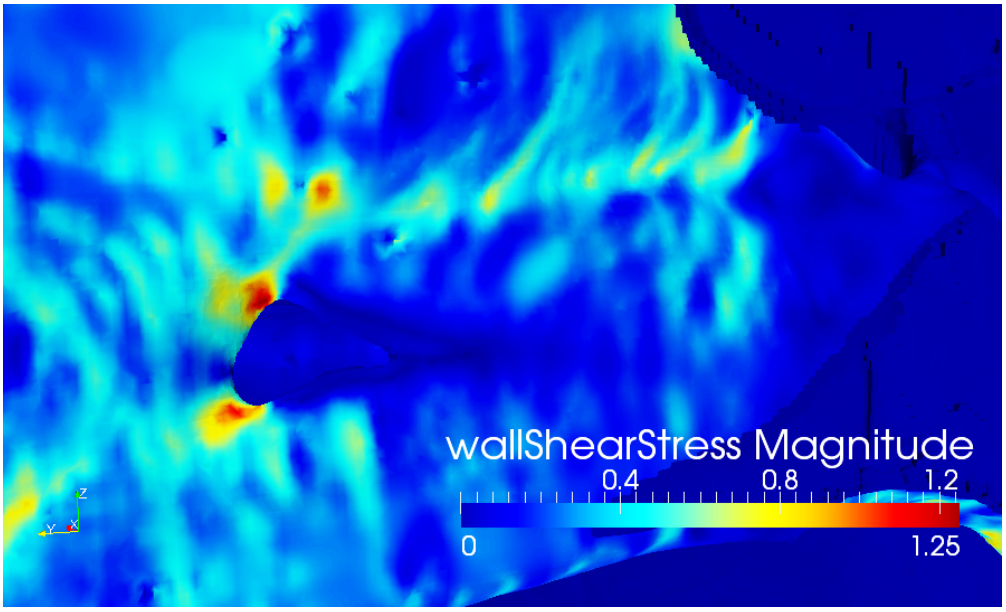


(a) *Pre-surgery.* (b) *Post-virtual surgery.*

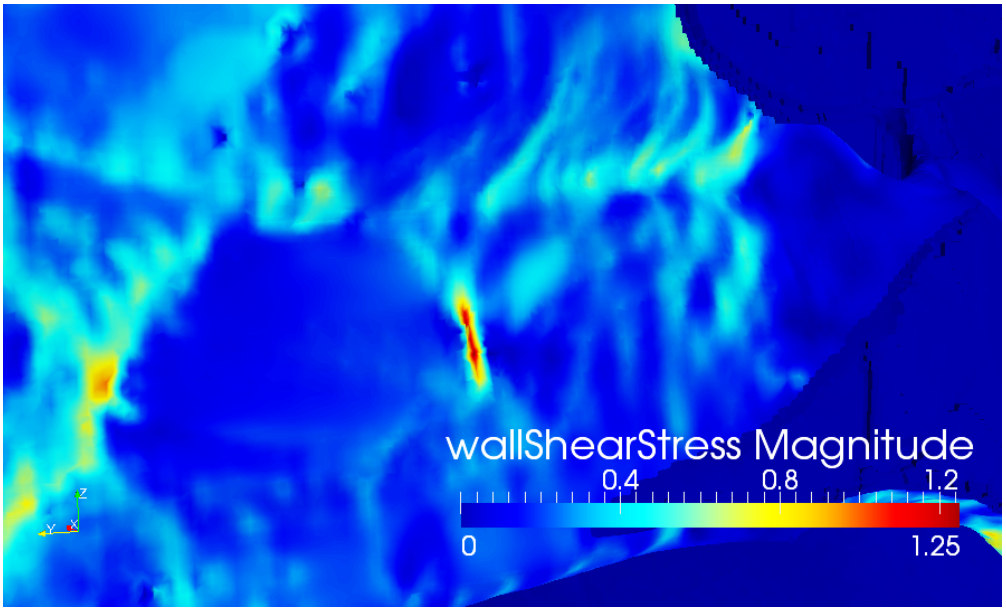
Figure 5.29: Vorticity component ω_X $[s^{-1}]$; sagittal section, deviated side.

Wall Shear Stress

Consider now the distribution and intensity of wall shear stresses. As expected, the highest shear stresses are close by the fastest region, that is the nasal valve and the pharynx. However, the more interesting region is near the septal deviation.
In fact, looking to the two distributions, between the two cases there are no substantial differences, except in the septal deviation region where septal resection drastically change WSSs distribution. Figure 5.30 represents an extract of the septal deviation region, here it is evident the reduction of wall shear stresses after the virtual septal resection. It is interesting to note that after virtual surgery the stresses are notably reduced in the region where the flow was impacting on the obstruction; but stresses appears in the rear part of septal resection. This WSS trend demonstrates the delicacy of the surgery, in fact the surgeon must be very careful to not create high WSS regions when dealing with septal resections or turbinate reductions.



(a) *Pre-surgery.*



(b) *Post-surgery.*

Figure 5.30: Wall shear stresses.

6 Conclusions

6.1 Achievements

The present work has been carried out within the framework of the standing collaboration with the San Paolo Hospital in Milan. The main goals of this thesis work, i.e. the setting up of an open-source procedure for patient-specific virtual surgery in the nasal cavity and critically examining the performance of the available turbulence models have been successfully achieved.

The first point we have addressed has been the reconstruction of the 3D surface of the nasal cavity and the head starting from the patient's CT scan. The geometry is then modified in order to reproduce an endoscopic nasal surgery. Although these steps cannot be fully automated, the time required for this part of the procedure ranges from one to two hours; we consider this as an extremely satisfying result.

Once the meshes for pre- and post-virtual surgery cases are created through the OpenFOAM utilities (blockMesh and snappyHexMesh), our attention is focused on numerical simulations. The nasal cavity flow is assumed steady, in a situation corresponding to a quiet restful inspiration. We use the RANS equations to compute the flow field, and evaluate the effect of the choice of a turbulent model. In particular, we contrast the laminar, no-model approach, the classical $k-\omega-SST$ turbulence model, and the relatively new transitional $k_T - k_L - \omega$ turbulence model. This last model should be capable of catching the transitional nature of the flow, properly switching between laminar and turbulent conditions.

The comparison of the results shows that the results by the transitional model closely resemble those by the laminar ones. This behaviour is reasonable, owing to the low turbulence levels developed by this flow in the present conditions. Instead, the $k-\omega-SST$ model tends to represent a fully developed turbulent flow, and is unable to discern between laminar and turbulent fluctuations.

Indeed, for such a low-intensity inspiration, the instabilities in the flow are not too relevant. Nevertheless, a turbulence model is required should the full respiratory cycle be considered, and/or if the flow has higher intensities.

The comparison between pre- and post-virtual surgery flows shows that the septal resection induces the expected modification in the flow, but also that the pre-surgery situation was not characterized by any outstanding anomaly in global flow quantities, like for example flow rates. Further investigations on pressure and velocity fields, turbulent quantities and wall shear stresses have highlighted localised differences in the septal deviation region which could affect nasal physiology. It remains to be determined whether this points to a case where surgery was not really necessary, or to a scenario where proper functioning of such a delicate organ as the nose depends on the small-scale features of the airflow.

6.2 Future development

The CFD results still need a validation. This can be done through comparison with in-vitro experimental results and more accurate CFD simulations, such as LES or DNS. An experimental campaign is about to take place: its results will offer a reliable database of measurements to validate CFD results. Validation can also derive from an higher-fidelity CFD study. This implies setting up first a LES simulation, and then possibly a DNS too, with proper space and time resolution.

Along a different line, the RANS simulations can be improved by considering air temperature and humidity, as well as the transport of odorants, pollutants and aerosol particles. Moreover, a full respiratory cycle should be simulated through unsteady boundary conditions, and breathing at higher intensity should be considered.

For patient-specific virtual surgery to become an effective tool in the clinical treatment of nasal pathologies, a large number of clinical cases need to be investigated, involving severe septal deviations, turbinate hypertrophy, and other common alterations of the inner-nose geometry. The comparative analysis of such results should help the surgeon to determine whether a surgical treatment is recommended and which operating modality would be best, from the perspective of both fluid-dynamics and physiology.

A 3D-Slicer

Slicer, or 3D Slicer, is a free, open source software package for visualisation and image analysis. 3D Slicer is natively designed to be available on multiple platforms, including Windows, Linux and Mac Os X. The Slicer version used in this work is the 3.6.3, for more information about released version see the web site *http://www.slicer.org* and pubblications [26], [27]. Slicer has a double function, it allows both the study of the clinical features of the patient through visualisation of computed tomography and the reconstruction of a 3D model of the entire part of the body scanned.

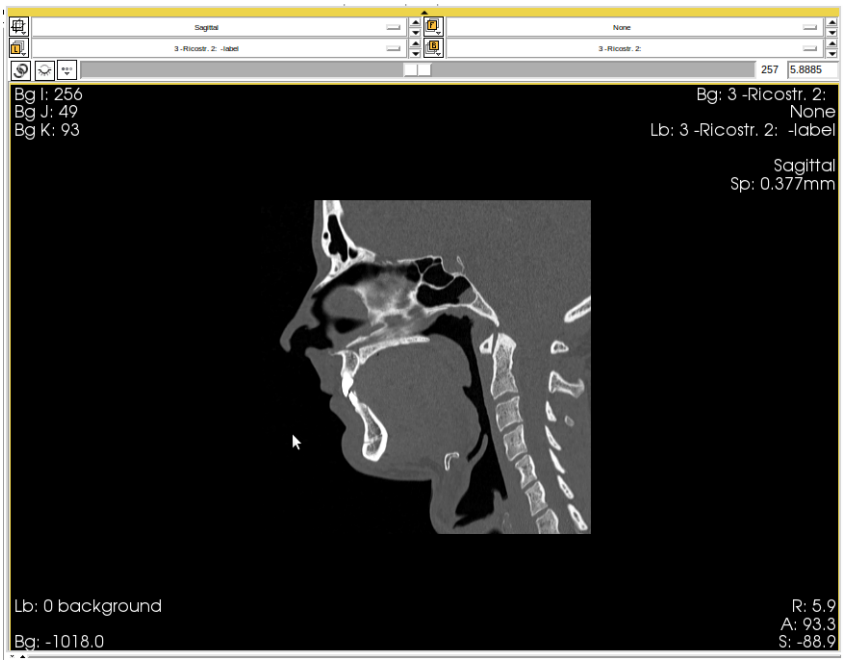


Figure A.1: Sagittal view in gray scale from a CT

A.1 Introduction to CT

X-ray computed tomography, also computed tomography (CT scan) or computed axial tomography (CAT scan), is a medical imaging procedure that uses computer-processed X-rays to produce tomographic images or *slices* of specific areas of the body. These cross-sectional images are used for diagnostic and therapeutic purposes in various medical disciplines. Digital geometry processing is used to generate a three-dimensional image of the inside of an object from a large series of two-dimensional X-ray images taken around a single axis of rotation.
CT produces a volume of data that can be manipulated, through a process known as *windowing*, in order to demonstrate various bodily structures based on their ability to block the X-ray beams. X-ray slice data is generated using an X-ray source that rotates around the object, X-ray sensors are positioned on the opposite side of the circle from the X-ray source. Once the scan data has been acquired, the data must be processed using a form of tomographic reconstruction, which produces a series of cross-sectional images.

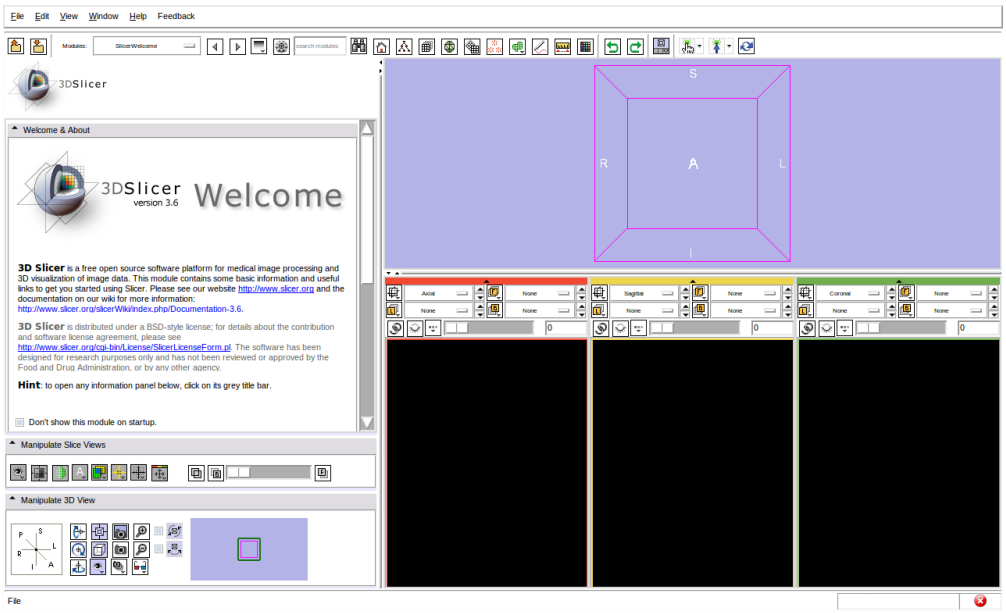


Figure A.2: Slicer start-up module

Pixels in an image obtained by CT scanning are displayed in grey scale, from black to white, in terms of relative radiodensity, that is the relative transparency property of a material to the radiation transfer. The grey scale is related to the mean attenuation of the tissue, starting from

+3071,the most attenuating (white, e.g. bones), to -1024, the least attenuating (black), on the Hounsfield scale. When the CT slice thickness is also factored in, the unit is known as a Voxel, which is a three-dimensional unit. The Hounsfield unit (HU) scale is a linear transformation of the original linear attenuation coefficient measurement into one referred to the radiodensity of distilled water and air at standard pressure and temperature. In a voxel with average linear attenuation coefficient μ_x, the corresponding HU value is therefore given by:

$$HU = \frac{\mu_x - \mu_{water}}{\mu_{water} - \mu_{air}} \cdot 1000 \tag{A.1}$$

where μ_{water} and μ_{air} are the linear attenuation coefficients of water and air, respectively. Thus, a change of one Hounsfield unit (HU) represents a change of 0.1% of the attenuation coefficient of water since the attenuation coefficient of air is nearly zero.
Water has an attenuation of 0 HU, while air is -1000 HU, cancellous bone is typically +400 HU, cranial bone can reach 2000 HU or more and can cause artefacts. It is very important to keep in mind these attenuations, because they play a key role in geometry reconstruction with Slicer.

In this case the CT scans concern the patient cranium, from the poll to the larynx. There are available 313 axial images, 0.625 mm spaced, composed by 512×512 *pixels*; this implies the reconstruction of 512 images, 0.377 mm space, in both sagittal and coronal planes.
Furthermore, looking at figure A.1, it can be seen on the lower right, the cursor coordinates and on the lower left the HU value in that point, which is about -1000, thus air.

A.2 Slicer Procedure

In this section the procedure to reconstruct surfaces from CT scans (saved as *.DICOM* files) with Slicer is presented.
The user's operations are as follows.

1. The Slicer start-up modules appears as in figure A.2. The user has to select *File* → *Add Volume*.
2. Choose the directory where *.DICOM* files are located, clic on *Parse Directory* → *Apply*.
3. In Slicer modules select the *Editor* module and confirm *Generic Anatomic Colors*. The frame would be similar to figure A.3.

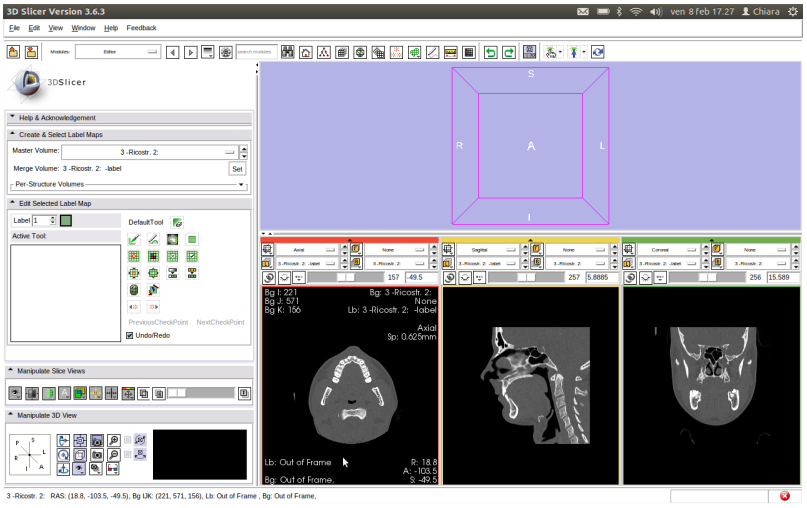


Figure A.3: Interface of Slicer after applying the Editor module

4. It is time to choose the HU threshold, the software will rebuild the CT volume inside the given range of HU. Thus, to include all the tissues and bones the lower threshold is set as -220 HU, instead, the upper limit is set to its maximum. In practice,with reference to figure A.4 select the *Threshold* button → *Set the lower HU value* → *press Enter*→ *Apply*. It is important to highlight that the HU lower bound comes from a sensitivity analysis, for more details see [30]. Taking this study into account, reasonable values for the lower threshold are included into the range $[-230 \div -200]$HU.

5. To avoid the reconstruction of the head support, the user has to remove any connection between the head and the support through the *Paint* tool. In other word, select the *Paint* tool → change *Label* to number 9, *foreign-objects*→ paint over any connection in each slice it is visible.

6. Still with *Label 9*, select the *Level Tracing* tool, go to the two sagittal views where the ears first appear and paint that region to avoid the air from going into the ears during simulations, see figure A.5.

7. Change the label into number 29, *gas*, select the *Change Island* tool and clic on a black point, that is air.

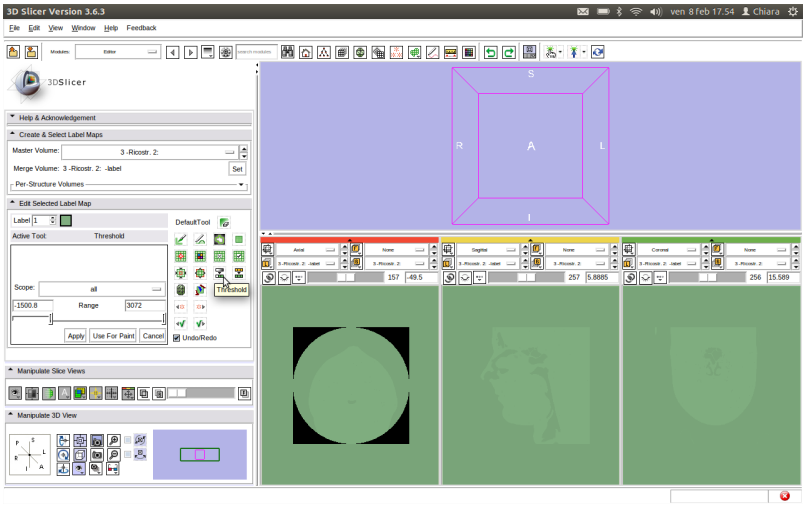


Figure A.4: Slicer frame meanwhile the user is setting the HU threshold

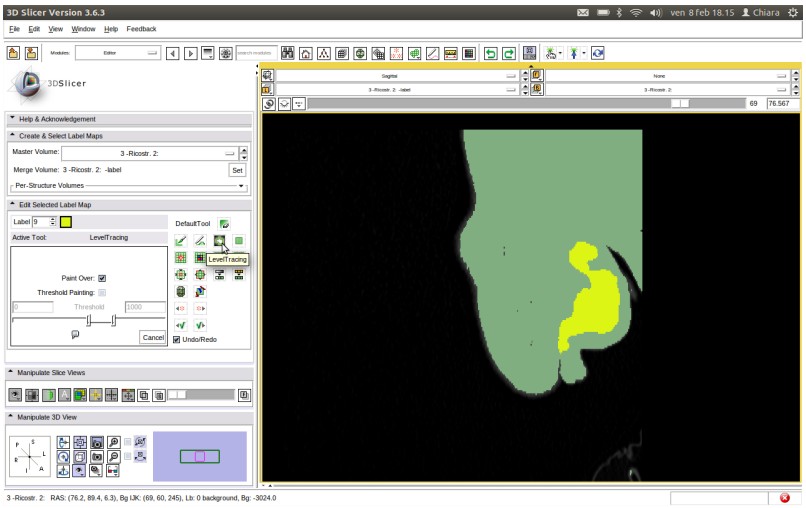


Figure A.5: Slicer frame meanwhile the user is excluding ears

8. Return to the previous *Label*, number 1, *tissue*; select the *Save Island* tool and clic on a green part, that is tissue. This coloured region represents the part which would be reconstructed.

9. Finally, select the box on the left *Per-Structured Volumes* → *Merge and Build*. The frame would appear like to figure A.6.

10. Save the reconstructed surface as an *.stl* file; this is the suitable format for OpenFOAM.

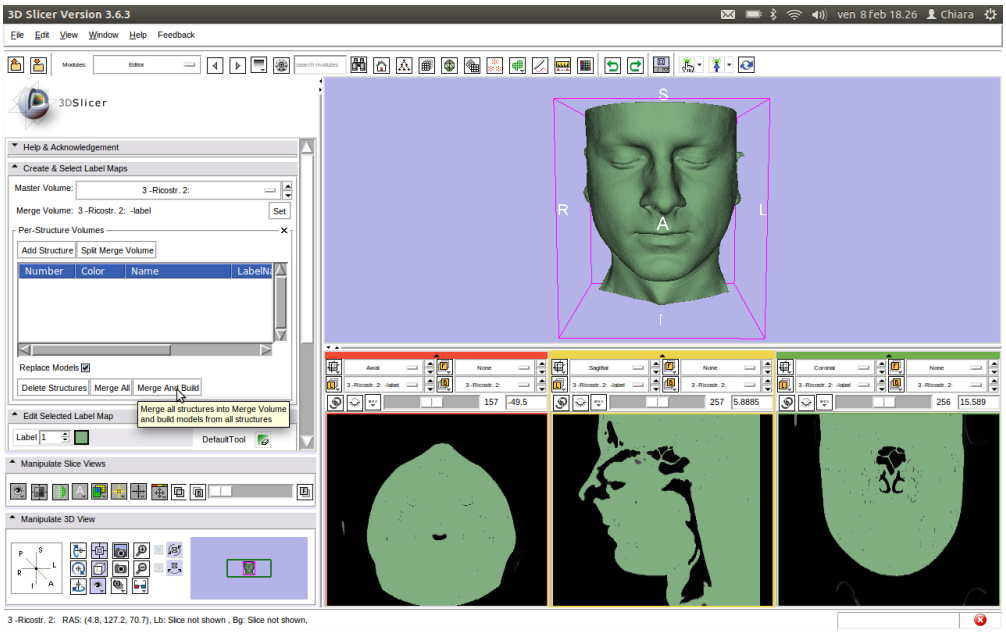


Figure A.6: Slicer frame with 3D reconstruction

B FreeCAD Geometry Decomposition

FreeCAD is a multi-platform general purpose 3D CAD modeller whose development is completely Open Source (GPL and LGPL License). Currently, FreeCAD can import and display CAD models in IGES, STEP, and B-Rep formats and meshes in STL, BMS, AST and Wavefront OBJ formats.

This software is used in the current work because of the need to divide the geometry into different *stl* file. In fact, when adding layers only in the nasal cavity to save on cells number, geometry must be split into different files. Below it is described how the geometry decomposition is carried out within FreeCAD.

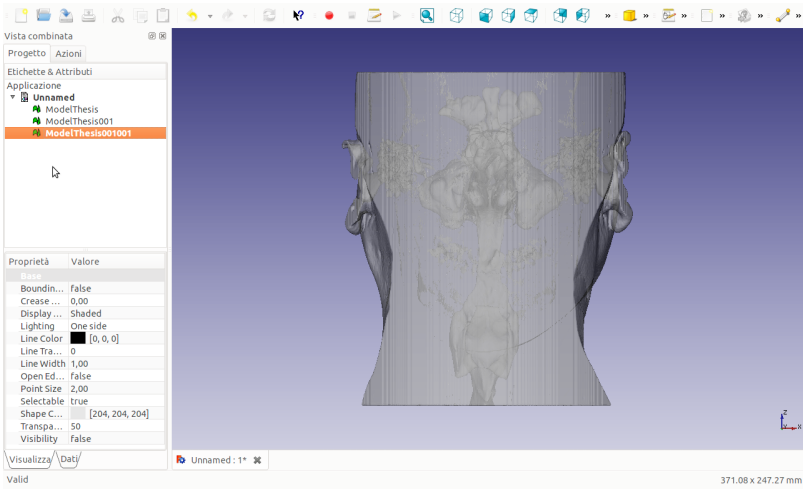
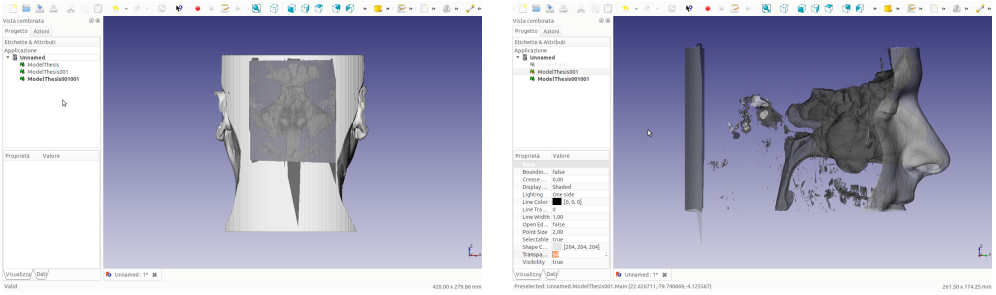


Figure B.1: Back view of entire surface in transparency

1. *File* → *Import* → select the $\langle geometry - name \rangle$.*stl* file and open it.
2. Select the *back view* and modify the surface transparency to work better (e.g. figure B.1).

3. *Tool* → *Mesh* → *Divide Mesh* → Select with the cursor a rectangular region containing the nasal cavity and with a left mouse clic close the selection and choose the option *internal*. The initial surface is then split into two, according to figure B.2.



(a) *First divided part in transparency.* (b) *Highlight of the first separated surface.*

Figure B.2: FreeCAD frames

4. Apply the same commands to the resultant surface after first division, to obtain the nasal cavity alone, as reference see figure B.3.

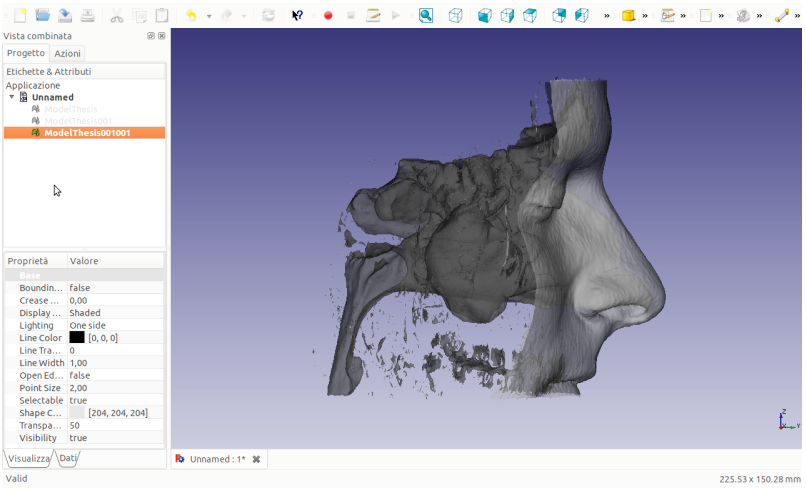
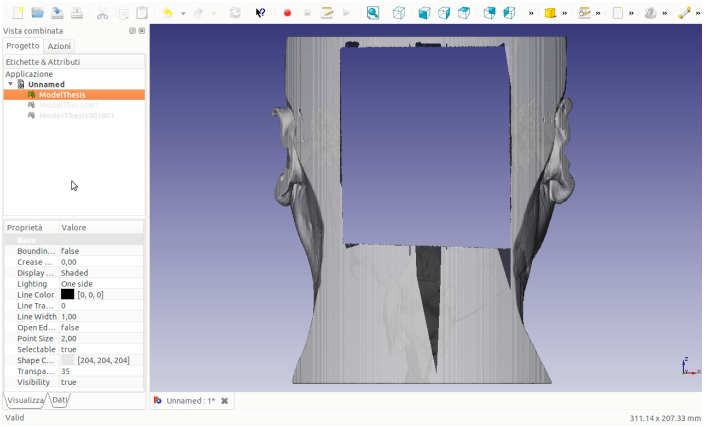
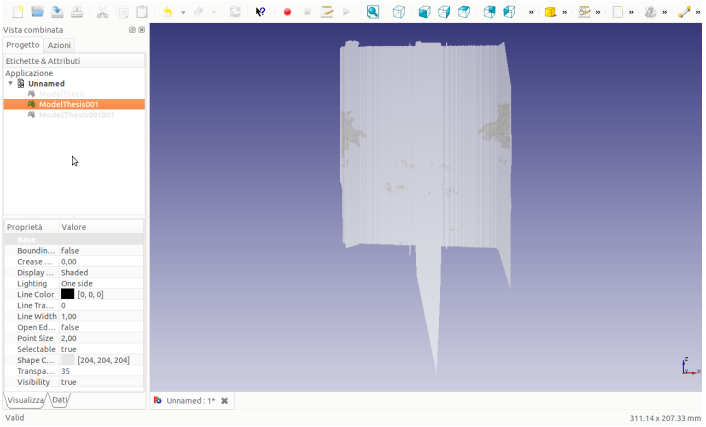


Figure B.3: Side view of nasal cavity in transparency

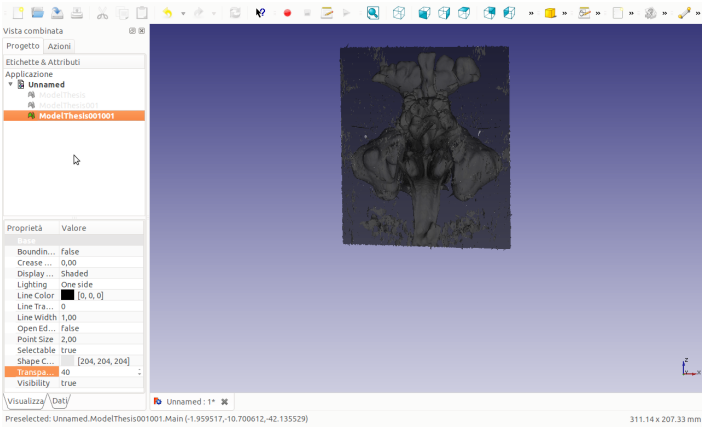
5. At the end the user will have three separated surfaces (e.g. figure B.4) which must be saved in stl format: *File* → *Export* → *Mesh formats* → ⟨*name*⟩*.stl*. These files will be imported in OpenFOAM for the simulations.



(a) *First part.*



(b) *Second part.*



(c) *Nasal cavity.*

Figure B.4: Three surfaces divided, back view

C Geometry Morphing: Blender

The aim of this chapter is to introduce the reader to the software Blender, the tool chosen for mesh editing to the end of virtual surgery.

Blender is a free and open-source 3D computer graphics software product used to creating animated films, visual effects, interactive 3D applications or video games. Here the 2.62.0 Blender version is used. Despite the huge capabilities of this software, in the present work its use is limited to mesh editing. To this end, the two primary modes of work are *Object Mode* and *Edit Mode*. Object mode is used to manipulate individual objects as a unit, while edit mode is used to manipulate the actual object data. In other words, object mode can be used to move, scale, and rotate entire polygon meshes, and edit mode can be used to manipulate the individual elements of a mesh, such as vertices, edges and faces. There is also another mode, *Sculpt Mode*, which will be used to edit the geometry, without adding or removing mesh elements. These modes, in relation to virtual surgery, will be presented in details in the following sections.
The mesh to be edited in the present work comes from the *3D Slicer* reconstruction (see Appendix A). It is very large and complex, thus there are two choices:

1. to modify the geometry as it comes from *Slicer*;

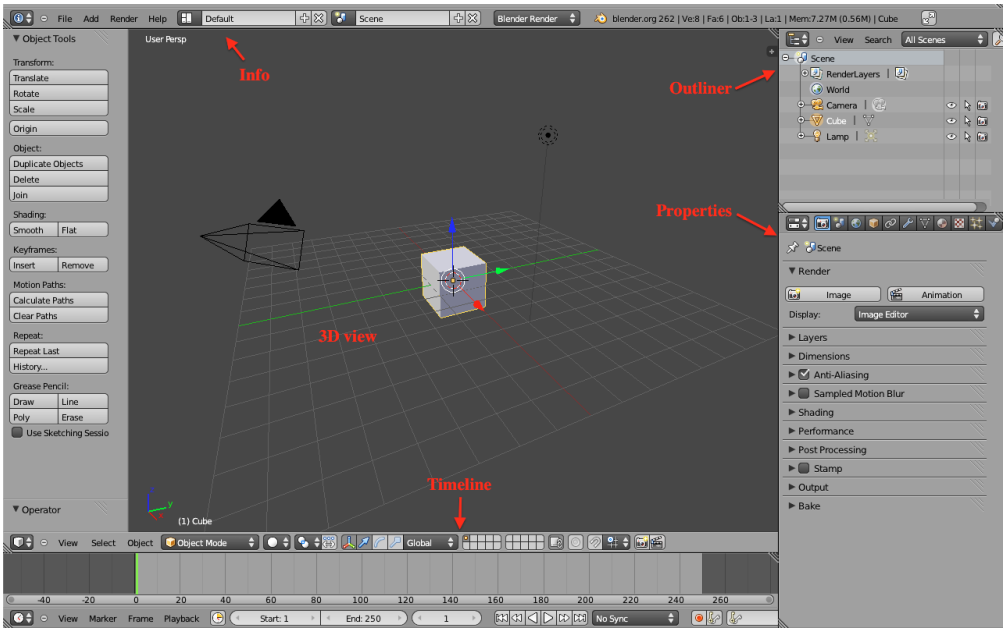
2. to subdivide the geometry with *freeCAD* [B] and apply the changes to a sub-mesh. In this case the only recommendation is to pay attention to changing reference system in Blender; this point will be clarified later.

In order to simplify the work of mesh editing to the surgeon, the second way is chosen. Thus the procedure so far looks like this:

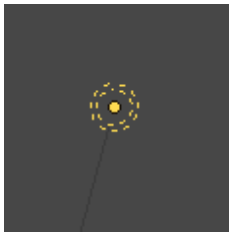
$$CT \rightarrow 3D\ Slicer \rightarrow FreeCAD \rightarrow Blender$$

C.1 Introduction to Blender

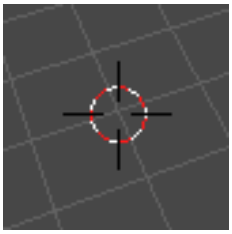
The Blender opening scene is like figure C.1, where can be seen three starting objects: a cube, a camera and a lamp.



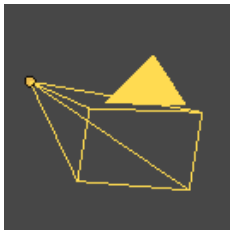
(a) *Blender opening scene.*



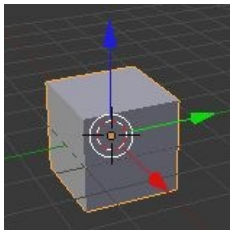
(b) *Lamp.*



(c) *3D cursor.*



(d) *Camera.*



(e) *Mesh.*

Figure C.1: Opening scene of Blender with default objects

Here the cube is a simple mesh; the camera is useful to render, but also to change the point of view; and the lamp simply lights up the scene. The 3D cursor is a movable object useful to define a reference point for transformations; press "N" to see its properties.

The starting scene is made of five windows, or boards. Each board is composed by two parts, a header, containing a menu with its selectors;

and a unit containing the instruments.
At the top there is the *info* board; in the middle the *3D view* window; at the bottom of the scene there is the *timeline* board with animation control. Finally, on the left there are the *outliner* and *properties* windows with the list of the objects participating the scene and the object modifiers, respectively. The last board is the most important for mesh editing. On the left, there is the *tool shelf* which changes with the active mode. Regarding the purpose of this work, the user can forget the timeline window.
Note that in *Object mode* transformations using the *tool shelf* commands implies a change in the local reference system of the object.
Since the user can add cameras on the scene, to have more than one view could help, thus dragging the icon on the top right of the scene, a new view is created; see figure C.2.

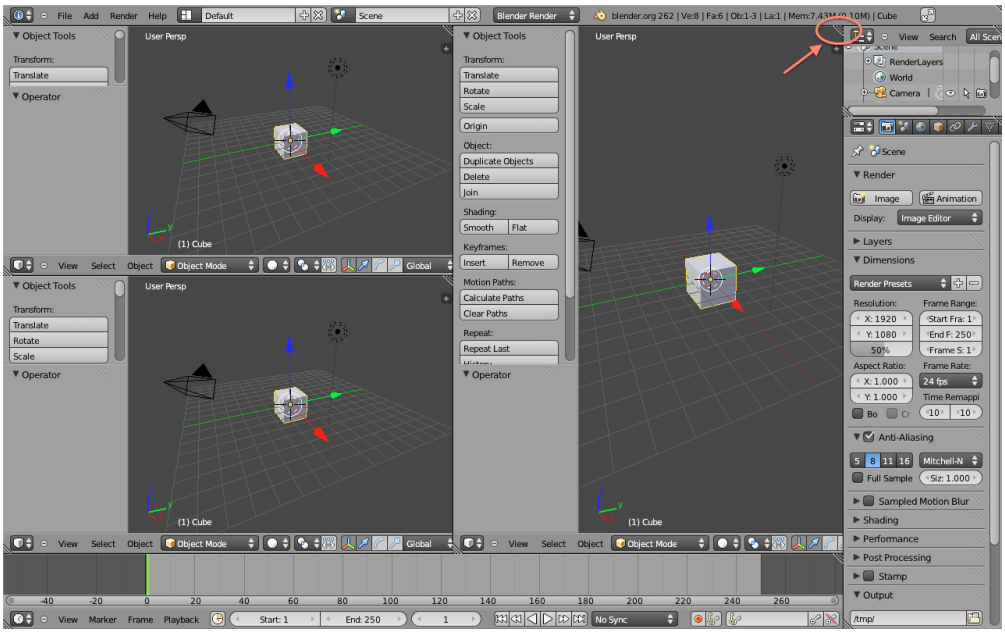


Figure C.2: Blender scene with more views, icon highlighted

To become familiar with the software, the user must know some basic controls, which are summarised in the next list.

- Move all the 3D scene: "shift + mouse central button" or refer to *view* → *navigation* → *one of the action.*

- Move the object: select *Transform* in the *tool shelf*, choose one action, drag the mouse, *left clic* to place the object. Remember that at the beginning the global and local reference systems correspond, but when one of these transformations is applied there is no more match between the two (see the *Origin* menu on the left).

- Selection/deselection:
  - one object: mouse *right clic*;
  - more objects: "shift + right clic" on each object to select;
  - all objects of the scene: "A";
  - objects in rectangular area: "B" → *left clic* → drag the mouse for the dimension of selection;
  - objects in circular area: "C" → drag the mouse for positioning the circle → *roller for selection dimensions* → *left clic to select* → *right clic to Esc*;

C.2 Sculpt Mode

The less intrusive mode to edit the mesh is the *Sculpt mode* because it does not add or remove vertices, but deforms the existing ones. To this end, the mesh should be dense enough. As the mode name says, the mouse is used exactly as a scalpel on the geometry surface.
This mode is used here where the surface needs to be modified locally, but no vertices, edges or faces cutting/adding are needed. The first thing to set is the *Brush* type. The drawings in figure C.3 are explanatory of the effects.
Then, the user could set the radius of influence and the strength of the brush. Notice that the brush could be used not only in *add* mode (default), but also in *subtract* mode. Left clic and drag on mesh to see the effects.
Another important setting is the *Stroke method*, usually *airbrush* or *drag-dot* are chosen. The *airbrush* applies the brush effects until the mouse left button is pressed; instead, the *drag-dot* limits the effects to the cursor position.
Moreover, the user can force these effects with respect to one or more symmetry axes with the tool *Symmetry* → *Mirror*.
Once the changes are done, it is possible to go back to *Object mode* and apply a *shading smooth* to the mesh to smooth possible edges introduced with the sculpt.

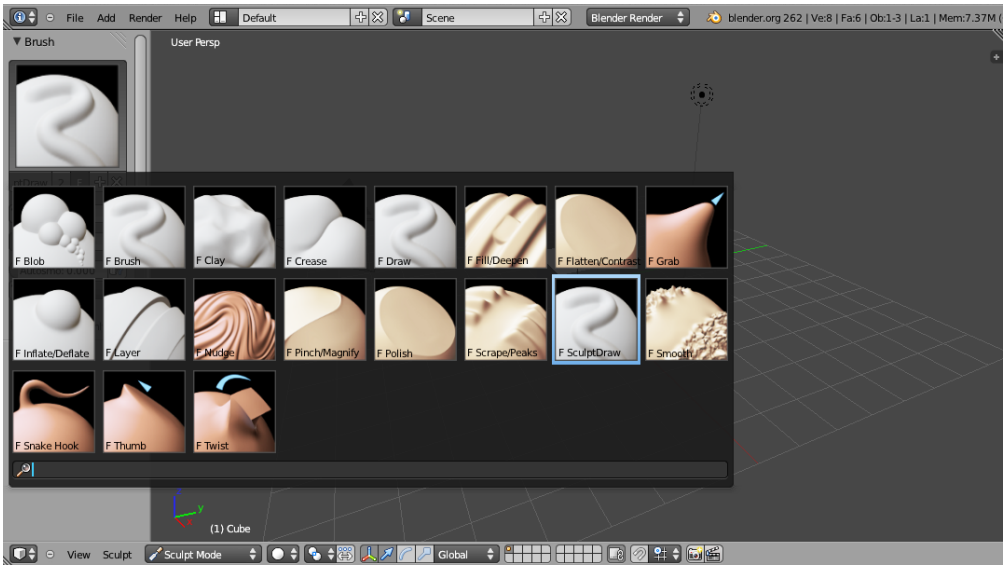


Figure C.3: Sculpt mode: brush types

This way of mesh editing, in relation to virtual surgery aims to reproduce the effects of little surface refinement done by the surgeon.

C.3 Edit Mode

In order to investigate the effects of a septoplasty and an eventual turbinate reduction, the most suitable tool is the *Edit mode*. Here the difficulty is not the mesh modification itself, once the user gets used to Blender, but it is the geometry complexity that delays the procedure.
Since the mesh is complex the procedure for virtual surgery should be applied to a sub-part of the geometry. Surgeon's steps would be the same with a sub-part of the mesh or with the entire one, unless positioning at the septal deviation with further cameras and applying these steps to it.
Before modifying the nasal cavity mesh, some basic tools of this mode are listed below:

- starting from *object mode*, select the mesh and press "Tab" to switch to *edit mode*, or select this mode from the command line above the timeline. At the moment the mesh is all highlighted.

- Selection and deselection tools work as in object mode, with the possibility to choose vertices, edges and faces or groups of them, instead of the entire object.

- A further selection tool in edit mode is *lazo* selection, that is a free-hand selection. Keeping "ctrl + left mouse button" pressed, drag the mouse around the region to select.

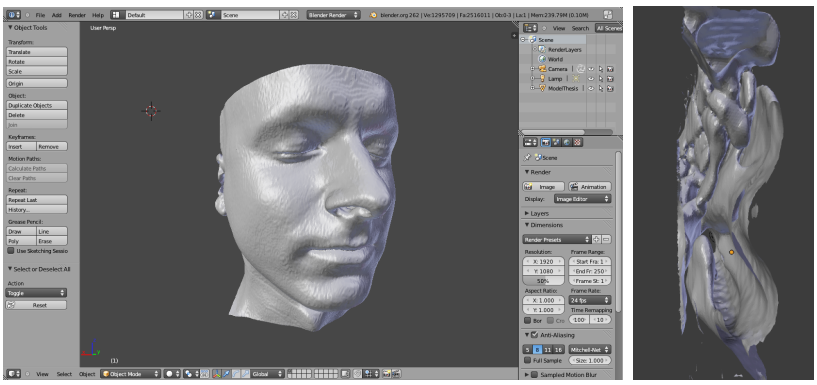
- Remove vertices, edges and faces: highlight the entities and press "X", a menu appears, choose one of these voices to delete elements.

- See the menu *Mesh* at the bottom board for all the mesh instruments.

During a septal resection the surgeon re-shapes or removes cartilage and bone until he obtains a well aligned septum. Moreover a septoplasty could be supported by an inferior turbinate reduction. The mesh editing will reflect this procedure.
Below are reported the steps to modify the mesh and the resultant geometry, from Blender opening to saving the edited mesh.

1. Open Blender and delete the default mesh, that is the cube.

2. *File* $\rightarrow$ *Import* $\rightarrow$ *Stl* $\rightarrow$ choose the stl file and clic on *Import STL* on the left.



(a) *Full geometry.* (b) *Extracted geometry.*

Figure C.4: Stl geometry imported in Blender

3. The aim here is to remove the septal deviation highlighted in figures C.4 and C.5. Once the user knows from CT the part of the nasal cavity affected by the septal deviation or other pathologies suitable for virtual surgery, he can extract the interested part of the geometry through *freeCAD* B and easily work on it.

Figure C.5: Particular of the geometry

4. Pass in edit mode, deselect all and choose the suitable view for the editing. For this septal resection the user must find the best point of view to apply this changing, as a surgeon.

5. Select the inner vertices or directly the faces of the region of interest with one of the methods cited before, the *lazo* could be the easiest, as shown in figure C.6

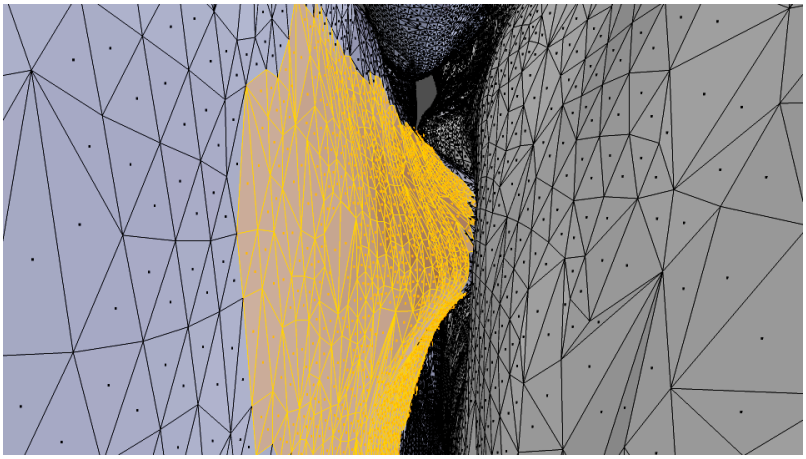


Figure C.6: Particular of the region to modify

6. Delete the selected vertices or faces, in order to rebuild the septum, but now aligned.

7. Select the border vertices of the hole obtained with vertices removal, select *Mesh* at bottom of the scene → *Faces* → *Fill*.

8. In order to make the new surface smoother, when the new faces are still highlighted, use the tool *Deform* → *Smooth Vertex* and, eventually, *Shading* → *Smooth*. This action has a local effect in edit mode.

9. Repeat these actions until the required simulation of septal resection is obtained.

10. The edited mesh looks like figure C.7.

11. Return to *object mode*.

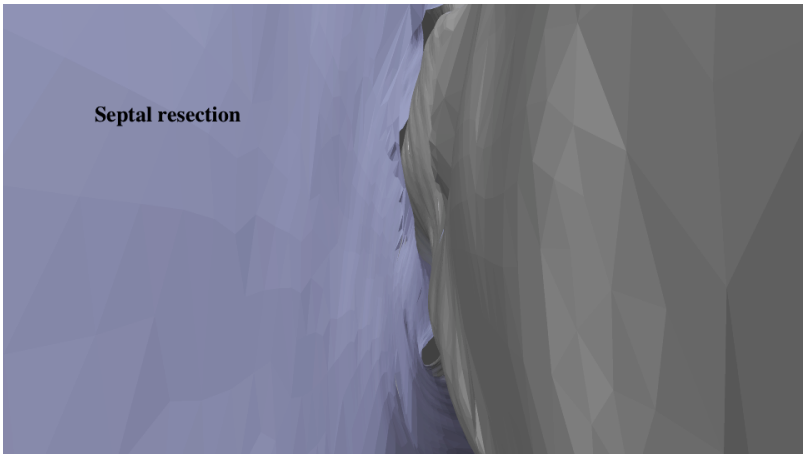


Figure C.7: Particular of the geometry modified

12. If the user accidentally move the geometry from the native reference system, the last step before saving is to bring back the geometry to the global reference system. This can be easily done selecting in the *tool shelf*: *Origin* → *Geometry to Origin*. In this way the modified mesh can fit again with the other pieces of geometry subdivided before.

13. *File* → *Export* → *Stl*.

The user should know that, because of some very heavy meshes, the software could be slow, thus he must be patient and wait for the conclusion of each action he does.
Note that it is very important that the original geometry and the modified one perfectly coincide, obviously except in the region of septal deviation. Thus the user has to control this aspect before going on with CFD. In figure C.8 is reported a coronal section where the superimposition is excellently verified.

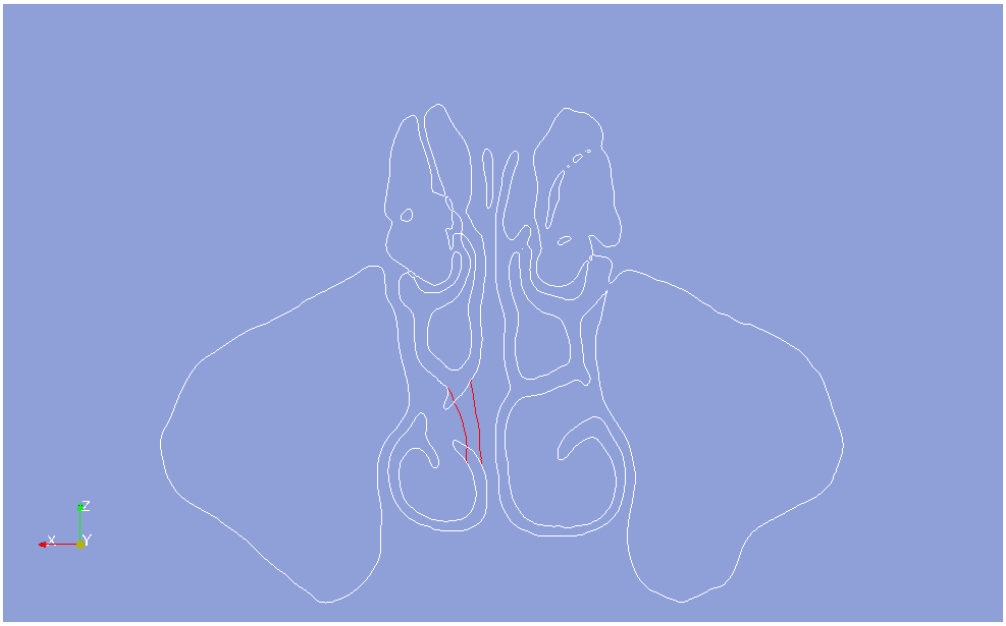


Figure C.8: Particular of the geometry modified

Unfortunately, the mesh editing steps could not be automated, because of patient-specific application; that is each nasal pathology is different from the other both in terms of form and position along the nasal cavity. Once the user is familiar with the softwares applied, the mesh editing procedure takes from one to two hours.

D Generation of the Mesh

In this chapter it is described how to transform an input surface geometry, in this case the one from 3D Slicer, into a volume mesh. This is one of the most delicate issue, because the mesh is an integral part of the numerical solution and must satisfy certain quality criteria to ensure a valid, and hence accurate, solution.
The procedure to obtain the final mesh is composed essentially by four steps:

- freeCAD geometry decomposition;
- Blender geometry morphing;
- the *blockMesh* utility;
- the *snappyHexMesh* utility.

Note that *blockMesh* and *snappyHexMesh* are both OpenFOAM utilities.

D.1 *blockMesh* Utility

Before *snappyHexMesh* is executed the user must create a background mesh of hexahedral cells that fills the entire region around the surface boundary. This can be done simply using *blockMesh*, but the following criteria must be observed when creating the background mesh:

- the mesh must consist purely of hexes;
- the cell aspect ratio should be approximately 1, otherwise the convergence of the snapping procedure is slow, possibly to the point of failure;
- there must be at least one intersection of a cell edge with the STL surface.

The blockMesh utility creates parametric meshes generated from a dictionary file named blockMeshDict located in the *constant/polyMesh* directory of a case. BlockMesh reads this dictionary, generates the mesh and writes out the mesh data to *points and faces*, *cells* and *boundary* files in the same directory.
The principle behind blockMesh is to decompose the domain geometry into a set of 1 or more three dimensional, hexahedral blocks. Edges of the blocks can be straight lines, arcs or splines. Each block of the geometry is defined by 8 vertices, one at each corner of a hexahedron. The vertices are written in a list so that each vertex can be accessed using its label, remembering that OpenFOAM always uses the C++ convention that the first element of the list has label "0". Each block has a local coordinate system (x1 , x2 , x3) that must be right-handed.
Hereafter is presented the *blockMeshDict* used in this work, for a better understanding of each input, the reader is addressed to the OpenFOAM User Guide.

```
convertToMeters 1000;
vertices
(
    (-0.10 -0.11 -0.07)
    ( 0.10 -0.11 -0.07)
    ( 0.10  0.13 -0.07)
    (-0.10  0.13 -0.07)
    (-0.10 -0.11  0.09)
    ( 0.10 -0.11  0.09)
    ( 0.10  0.13  0.09)
    (-0.10  0.13  0.09)
);
blocks
(
    hex (0 1 2 3 4 5 6 7) (80  96  64) simpleGrading  (1 1 1)
)
edges
(
);
boundary
(
    bottomWall
    {
        type patch;
        faces
        (
            (0 1 2 3)
        );
    }
........
    rightWall
    {
        type patch;
        faces
```

```
        (
            (1 2 6 5)
        );
    }
);
mergePatchPairs
(
);
```

After running blockMesh, the user obtains the background hex mesh which defines the extent of the computational domain and a base level mesh density. This will be the starting point for running snappyHexMesh in order to obtain the volume mesh.

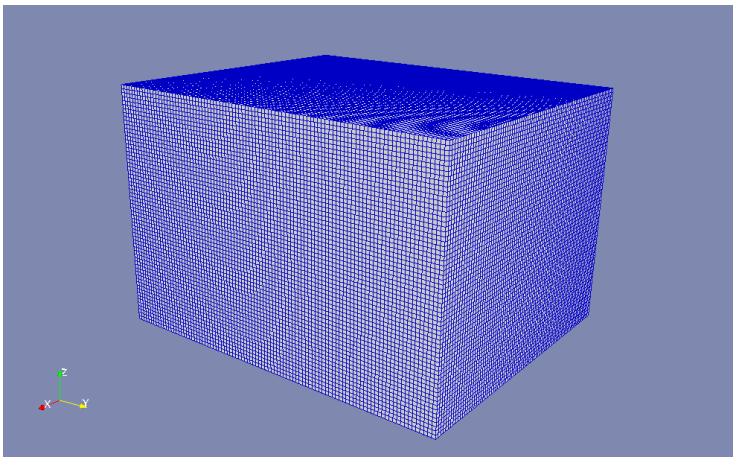
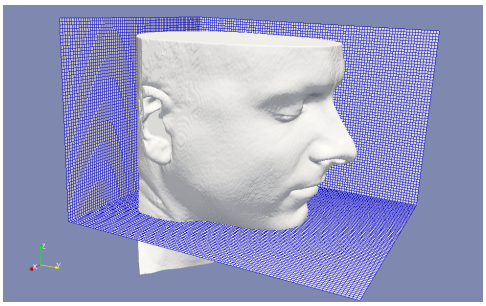
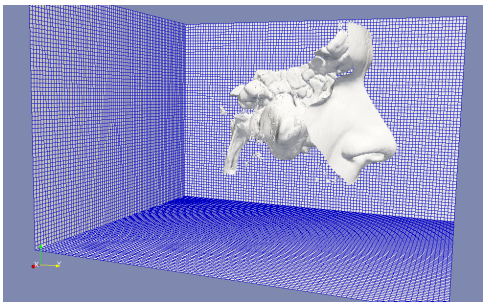


Figure D.1: Resultant block mesh for this work



(a) *blockMesh with geometry to be modeled.*



(b) *blockMesh and highlight of nasal cavity geometry.*

Figure D.2: ParaView frames

D.2 *snappyHexMesh* Utility

The objective is to mesh a rectangular shaped region surrounding an object described by and STL surface. Thus, in order to run snappyHexMesh, the user requires the following:

- surface data files in STL format, either binary or ASCII, located in a *constant/triSurface* sub-directory of the case directory;
- a background mesh generated using blockMesh, as said before;
- a snappyHexMeshDict dictionary, with appropriate entries, located in the *system* sub-directory of the case.

Notice that this utility could run in parallel.
Following, it is reported the *snappyHexMeshDict* used in this thesis. The entries will be illustrated in the next paragraphs.
SnappyHexMesh consists of three steps:

- castellatedMesh;
- snap;
- addLayers.

Before starting with them, the utility requires the definition of geometry, which is specified in a geometry sub-dictionary of snappyHexMesh. The geometry could be specified through an STL surface or bounding geometry entities in OpenFOAM; where *STL* surfaces are used to *"snap"* the mesh boundary to the surface, instead *refinementBox* is used to specify refinement for any mesh cell intersecting it or inside/outside it.

```
// Geometry. Definition of all surfaces.
geometry
{
    p1.stl
    {
        type triSurfaceMesh;
        name p1;
    }
    p2.stl
    {
        type triSurfaceMesh;
        name p2;
    }
    NasalCavity.stl
    {
        type triSurfaceMesh;
```

```
        name NasalCavity;
    }
    refinementBox1
    {
        type searchableBox;
        min (-41 -25 -25);
        max ( 41  45 45);
    }
};
```

D.2.1 Castellated Mesh

In sub-dictionary *castellatedMeshControls* there are the settings for the castellatedMesh generation. During this phase the initial block mesh is refined according to surface and volumetric refinement settings supplied by the user in this sub-dictionary.

```
castellatedMeshControls
{
    // Refinement parameters

   maxLocalCells 6000000;
    maxGlobalCells 15000000;
    minRefinementCells 1;
    nCellsBetweenLevels 4;
```

These first four entries are the global mesh size controls and buffer layers. In particular, the meaning of *nCellsBetweenLevels* is illustrated in figure D.3.

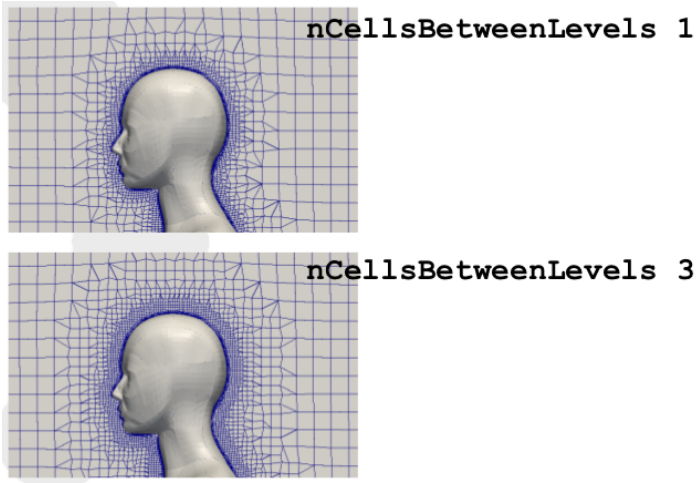


Figure D.3: Different values for *nCellsBetweenLevels*

In this treatment, *features* are not considered, so this entry is skipped. Consider now the surface based refinements, these entries provide the levels of refinement of the backgroundMesh intersecting the STL surface. It can be seen from figure D.4 how cells are refined according to the level.

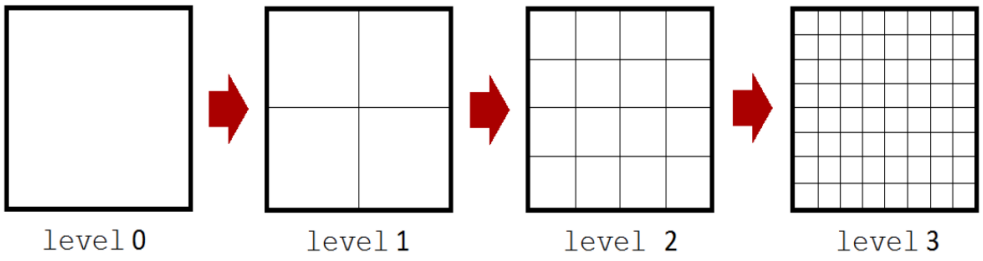
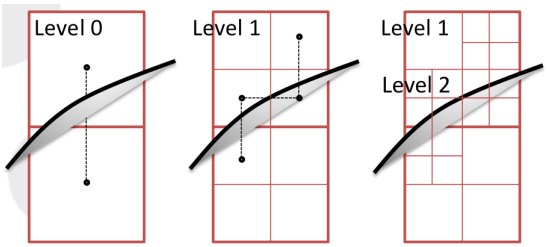


Figure D.4: Examples of level entries producing different refinement

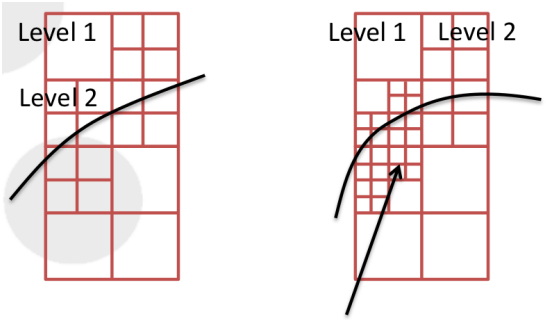
Knowing how cells are refined, apply this to the surface.

```
// Explicit feature edge refinement
features ();

// Surface based refinement
refinementSurfaces
{
    p1
    {
        level (2 2);
    }
    p2
    {
        level (2 2);
    }
    NasalCavity
    {
        level (3 3);
    }
}

resolveFeatureAngle 30;
```

In this sub-dictionary the user has to specify two levels for each surface. The first is the minimum level; every cell intersecting a surface gets refined up to the minimum level. The second level is the maximum level; cells that "see" multiple intersections, where the intersections make an angle bigger than the *resolveFeatureAngle* get refined up to the maximum level. For instance, if the user sets the minimum and maximum level as *level* (2 3), there will be not only a global surface refinement as shown in figure D.5(a), but also a local curvature based refinement, as it can be seen from figure D.5(b).



(a) *Surface based refinement.*



(b) *Additional feature refinement.*

Figure D.5: Refinement levels

The next passage is volume refinements, with the possibility to choose between different modes: *inside*, *ouside* or *distance*. The regions where cells are refined are the *refinementBoxes* defined in geometry sub-dictionary.

```
// Region-wise refinement
refinementRegions
{
    refinementBox1
    {
        mode inside;
        levels ((1 3));
    }
}

// Mesh selection
locationInMesh (0 120 -60);

allowFreeStandingZoneFaces false;}
```

In this case the level entries have a different meaning. In *inside/outside* mode:

levels(<*distance*> <*level*>)

where the entry <*distance*> is ignored; *level* represents effectively the level of refinement of the volume.
Instead, in *distance* mode the user could have a situation like this:

levels((<*distance*> <*level*>)...( )...(<*distance*> <*level*>))

where the first entry inside each round bracket represents the absolute distance from surface and *level* is still the wanted refinement level for that region. Note that distances need to be specified in descending order.
After refinement the section reachable from the *locationInMesh* is kept; in other words, this point must belong to the volume mesh the user wants to retain.
The last entry of this sub-dictionary is *allowFreeStandingZoneFaces*. This keyword is used when an internal face is on one boundary of the solution domain. This usually occurs when an internal face coincides with the background mesh boundary. The user can set this entry to be true or false, but keeping it true can cause quality issues during the snapping process. In the present study the castellated mesh appears as follows, figures D.6 and D.7.

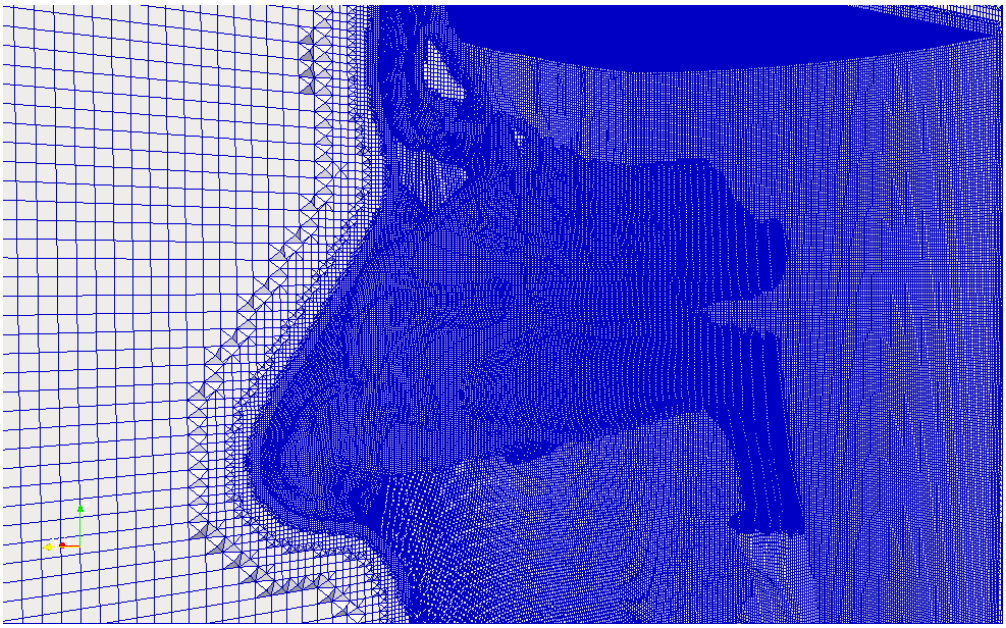
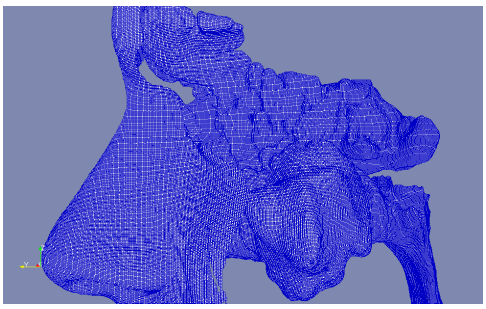
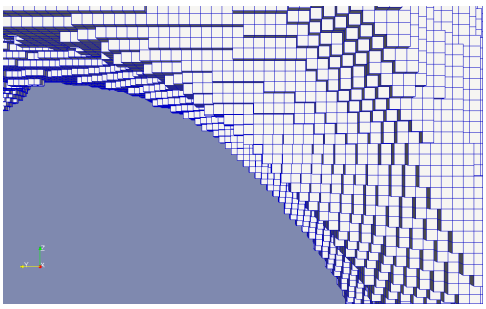


Figure D.6: Resultant castellated mesh, sagittal view of the nasal cavity



(a) *Castellated mesh, extract of the nasal cavity.*



(b) *Castellated mesh, particular of the pharynx.*

Figure D.7: ParaView frames, castellated mesh

D.2.2 Snap Phase

The second meshing stage is called *snapping* where patch faces are projected onto the surface geometry. This process involves moving cell vertex points onto surface geometry to remove the jagged castellated surface from the mesh. Schematically, the process is:

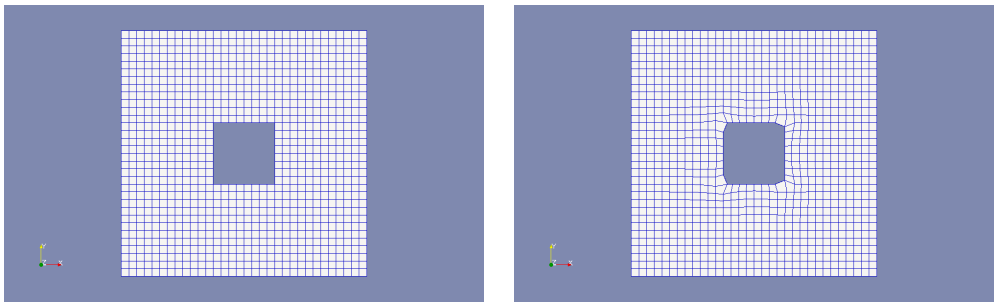
1. displace the vertices in the castellated boundary onto the STL surface;
2. solve for relaxation of the internal mesh with the latest displaced boundary vertices;
3. find the vertices that cause mesh quality parameters to be violated;
4. reduce the displacement of those vertices from their initial value and repeat from 2 until mesh quality is satisfied.

Considering this work:

```
// Settings for the snapping.
snapControls
{
    nSmoothPatch 2;

    // Relative distance for points to be
    // attracted by surface feature point
    //  or edge. True distance is this
    // factor times local maximum edge length.
    tolerance 4.0;

    // Number of mesh displacement
```

```
    // relaxation iterations.
    nSolveIter 50;

    // Maximum number of snapping
    // relaxation iterations.
    nRelaxIter 5;
}
```

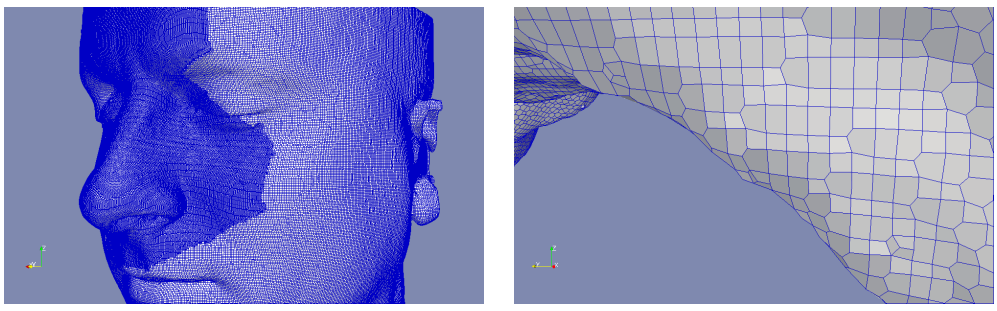
In this section the most relevant entry is *nSmoothPatch*, or better the number of pre-smoothing iterations of patch points before projection to the surface is performed. Refer to figure D.8 for clarifications.



(a) nSmoothPatch = *0.* (b) nSmoothPatch = *3.*

Figure D.8: Comparison between different *nSmoothPatch*

After the snapping procedure, the mesh under investigation appears as follow.



(a) *View of the head after snapping.* (b) *Snapped mesh, particular of the pharinx.*

Figure D.9: ParaView frames, snapped mesh

D.2.3 Adding layers

The final optional meshing stage is the layer addition, where one or more layers of hexahedral cells are added to a specified set of boundary patches. This stage is controlled by the *addLayersControls* sub-dictionary. The process of mesh layer addition involves shrinking the existing mesh from the boundary and inserting layers of cells, with respect to mesh quality controls.

It is fundamental to highlight that the layers addition relates to the existing mesh, not the surface geometry; hence applied to a patch, not a surface region. This implies that the starting mesh must have a good quality.
In the following, thickness parameters are expressed in relative sizes, that is the final layer thickness and minimum thickness can be defined as being relative to the background spacing ΔS, see figureD.10.

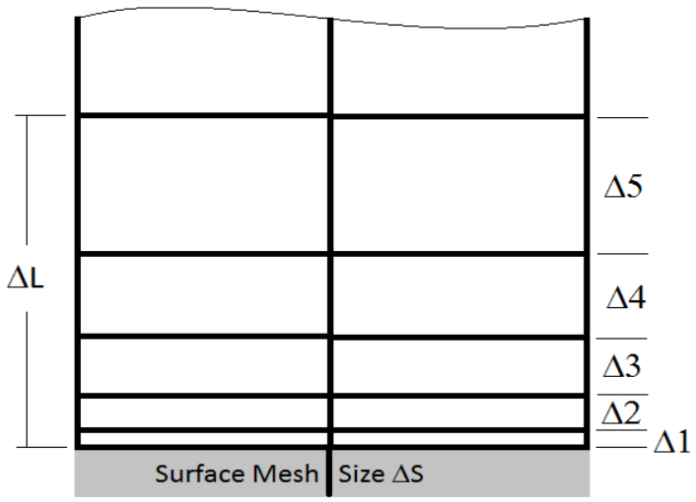


Figure D.10: Layers distrubution scheme

Specification of the number of layers, the final layer thickness and expansion ratio uniquely defines the layer profile and they are used to calculate the first cell height and total layer thickness. In particular:

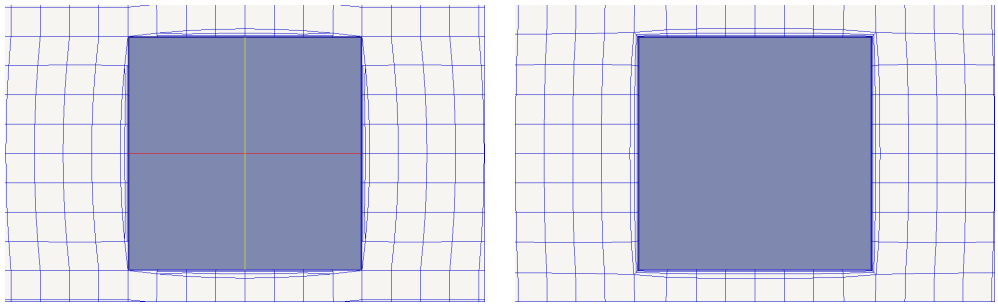
- *nSurfaceLayers* is the number of layers added.

- *expansionRatio* is the ratio of the heights from one layer to the next consecutive layer in the direction away from the surface, i.e. $\frac{\Delta 2}{\Delta 1} = \frac{\Delta 3}{\Delta 2} = \frac{\Delta 4}{\Delta 3} = \frac{\Delta 5}{\Delta 4}$.
- *finalLayerThickness* is the wanted thickness of final added cell layer relative to the adjacent surface mesh size, i.e. $\frac{\Delta 5}{\Delta S}$. If multiple layers are added, it is the thickness of the layer furthest away from the wall.
- *minThickness* is the specification of a minimum layer thickness below which height layers will automatically be collapsed. If for any reason layer cannot be above minThickness do not add layer.
- *nGrow* is a parameter related to the features, which is not considered here.

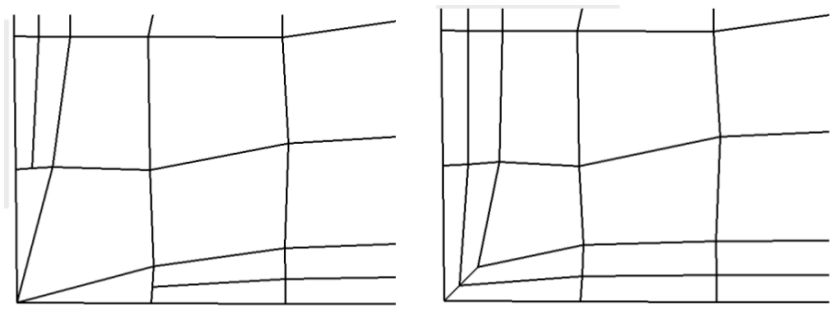
```
// Settings for the layer addition.
addLayersControls
{
    relativeSizes true;

    layers
    {
        NasalCavity_vcg
        {
            nSurfaceLayers 4;
        }
    }
    expansionRatio 1.6;
    finalLayerThickness 1.0;
    minThickness 0.04;

    nGrow 0;

    // Advanced settings

    featureAngle 180;

    //- Maximum number of snapping relaxation iterations
    nRelaxIter 5;
    // Number of smoothing iterations of surface normals
    nSmoothSurfaceNormals 1;
    // Number of smoothing iterations of interior mesh movement direction
    nSmoothNormals 3;
    // Smooth layer thickness over surface patches
    nSmoothThickness 10;
    // Stop layer growth on highly warped cells
    maxFaceThicknessRatio 1.3;
    // Reduce layer growth where ratio thickness to medial
    // distance is large
    maxThicknessToMedialRatio 0.3;
    // Angle used to pick up medial axis points
    minMedianAxisAngle 90;
    // Create buffer region for new layer terminations
```

```
    nBufferCellsNoExtrude 0;
    // Overall max number of layer addition iterations
    // mesh.
    nLayerIter 50;
}
```

Regarding the *Advanced settings*, the most important parameter is the feature angle, it is the angle above which layers are collapsed automatically. For instance, see figure D.11 with the difference between a feature angle of 45° and 180°.



(a) featureAngle = 45°. (b) featureAngle = 180°.



(c) featureAngle = 45°. (d) featureAngle = 180°.

Figure D.11: Comparison between differen *featureAngle*

To conclude, there are the mesh quality controls. Here the *maxNonOrtho* value is fundamental to have a good mesh in terms of convergence of the simulation. The lower the maximum non orthogonality is, the better the convergence is.
For further information the reader is addressed to the OpenFoam user guide and the paper [8].

```
meshQualityControls
{
    // Maximum non-orthogonality allowed.
    maxNonOrtho 60;
```

```
    // Max skewness allowed. Set to <0 to disable.
    maxBoundarySkewness 20;
    maxInternalSkewness 4;
    // Max concaveness allowed.
    maxConcave 80;
    // Minimum pyramid volume.
    minVol 1e-13;

    minTetQuality -1e30;
    // Minimum face area.
    minArea -1;
    // Minimum face twist.
    minTwist 0.05;
    // Minimum normalised cell determinant
    minDeterminant 0.001;

    minFaceWeight 0.05;
    minVolRatio 0.01;
    minTriangleTwist -1;

    // Advanced
    // Number of error distribution iterations
    nSmoothScale 4;
    // Amount to scale back displacement at error points
    errorReduction 0.75;
}
// Advanced

// Flags for optional output
// 0 : only write final meshes
// 1 : write intermediate meshes
// 2 : write volScalarField with cellLevel for
//      postprocessing
// 4 : write current intersections as .obj files
debug 0;

mergeTolerance 1e-6;
```

In the present work, as said before, layers are added only in the nasal cavity. On the average, there are four layers of cells with an expansion ratio of 1.6 and a final layer thickness equal to unity, all in relative sizes. The mesh with layers appears as follow. Figure D.12 refers to a throat section to get a simpler and better view of layers. A representation of the nasal cavity is given in figure D.13.

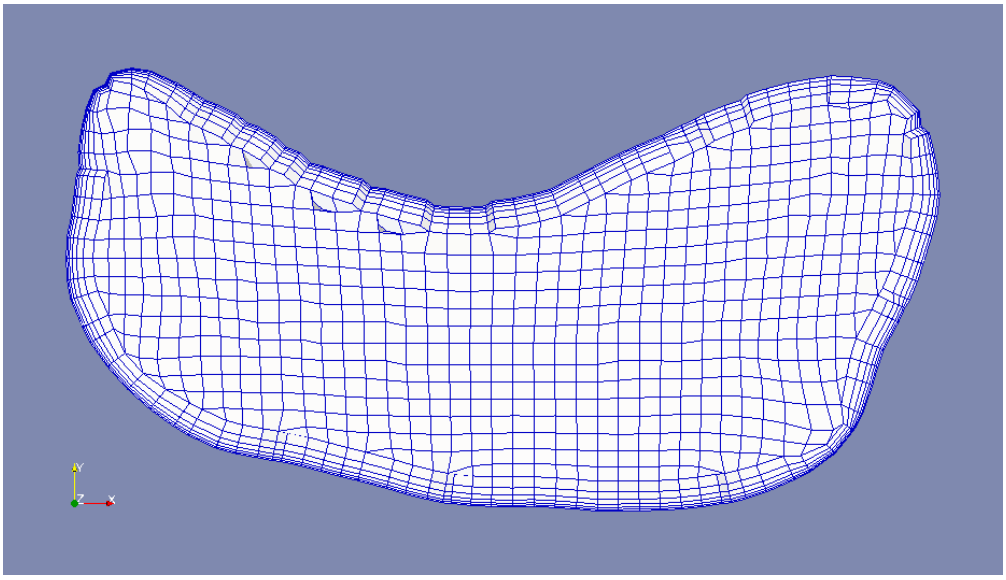


Figure D.12: Throat, mesh with layers

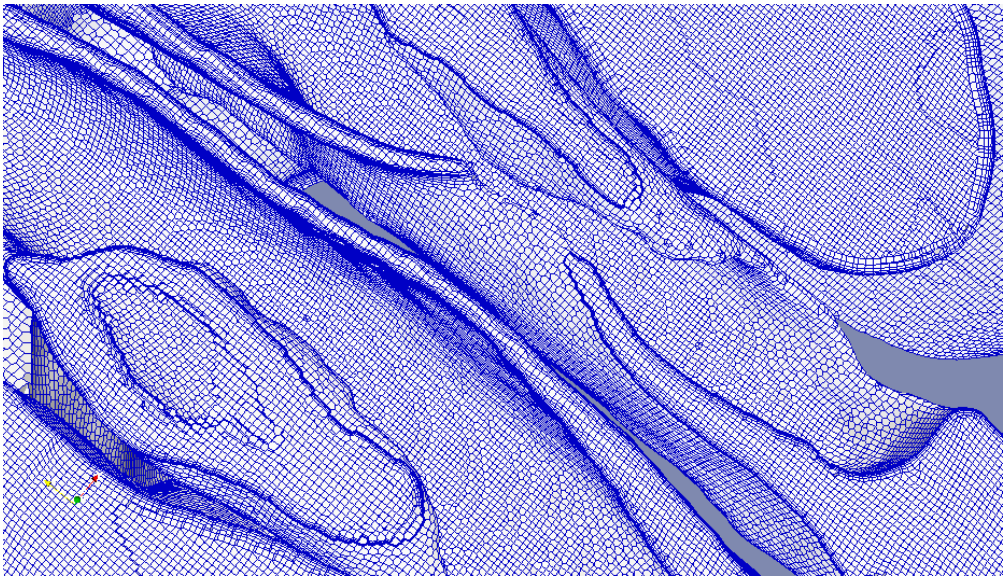


Figure D.13: Mesh with layers, axial view

Bibliography

[1] P. Arbour, E. Bilgen, and M. Girardin. Experimental study of nasal velocity fields in a human nasal fossa by laser anemometry. *Rhinology*, 23:201–207, 1985.

[2] X.B. Chen, H.P. Lee, V.F.H. Chong, and D.Y. Wang. Assessment of septal deviation effects on nasal air flow: a computational fluid dynamics model. *American Laryngological Rhinological and Otological Society*, 119:1730–1736, 2009.

[3] X.B. Chen, H.P. Lee, V.F.H. Chong, and D.Y. Wang. Impact of inferior turbinate hypertrophy on the aerodynamic pattern and physiological functions of the turbulent airflow – a CFD simulation model. *Rhinology*, 48(163-168), 2010.

[4] K.H. Cheng, Y.S. Cheng, H.C. Yeh, R.A. Guilmette, S.Q. Simpson, Y.H. Yang, and D.L. Swift. In vivo measurements of nasal airway dimensions and ultrafine aerosol deposition in the human nasal and oral airways. *J. of Aerosol Science*, 27(5):785–801, 1996.

[5] C. Croce, R. Fodil, M. Durand, and G. Sbrilea-Apiou. In vitro experiments and numerical simulations of airflow in realistic nasal airway geometry. *Annals of Biomedical Engineering*, 34(6):997–1007, 2006.

[6] D. Doorly, D.J. Taylor, P. Franke, and R.C. Schroter. Experimental investigation of nasal airflow. *J. Engineering in Medicine*, 222:439–453, 2007.

[7] D. Doorly, D.J. Taylor, A.M. Gambaruto, R.C. Schroter, and N. Tolley. Nasal architecture: form and flow. *Philosophical Transaction of the Royal Society*, 366:3225–3246, 2008.

[8] De Villiers E. A comprehensive tour of snappyhexmesh. In *7th OpenFOAM Workshop, 25 June 2012*, 2012.

[9] D. Elad, S Naftali, M. Rosenfeld, and M. Wolf. Physical stresses at the air-wall interface of the human nasal cavity during breathing. *J. of Applied Physiology*, 100:1003–1010, 2006.

[10] I. Hahn, P.W. Scherer, and M.M. Mozell. Velocity profiles measured for airflow through a large-scale model of the human nasal cavity. *J. Appl. Physiol.*, 75:2273–2287, 1993.

[11] O. Hilberg. Objective measurement of nasal airway dimensions using acoustic rhinometry: methodological and clinical aspects. *Allergy*, 57(60):5–39, 2002.

[12] I. Hoerschler, W. Schoroeder, and M. Meinke. On the assumption of steadiness of nasal cavity flow. *Biomechanics*, 43:1081–1085, 2010.

[13] S. Ishikawa, T. Nakayama, M. Watanabe, and T. Matsuzawa. Visualization of flow resistance in physiological nasal respiration. *Arch Otolaryngol Head Neck Surg.*, 132:1203–1209, 2006.

[14] H.H. Jin, J.R. Fan, M.J. Zeng, and K.F. Cen. Large eddy simulation of inhaled particle deposition within the human upper respiratory tract. *J. of Aerosol Science*, 38:257–268, 2006.

[15] Kim J.I. Chung S.K. Kim S. K., Na Y. Patient specific CFD models of nasal airflow: Overview of methods and challenges. *Journal of Biomechanics*, 46:299–306, 2013.

[16] C. Kleinstreuer and Z. Zhang. Airflow and particle transport in the human respiratory system. *Ann. Rev. Fluid Mech.*, 42:301–334, 2009.

[17] J.-H. Lee, Y. Na, S.-K. Kim, and S.-K. Chung. Unsteady flow characteristics through a human nasal airway. *Respiratory Physiology & Neurobiology*, 2010.

[18] X. Y. Luo, J. S. Hinton, T. T. Liew, and K. K. Tan. LES modelling of flow in a simple airway model. *Medical Engineering & Physics*, 26(5):403–413, 2004.

[19] T.B. Martonen, L. Quan, Z. Zhang, and C.J. Musante. Flow simulation in the human upper respiratory tract. *Cell Biochem Biophys*, 37:27–36, 2002.

[20] R. E. Mayle and A. Schulz. The path to predicting bypass transition. *Journal of Turbomachinery.*, 119:405–411, 1997.

[21] Jones N. The nose and paranasal sinuses physiology and anatomy. *Advanced Drug Delivery Reviews*, 51:5–19, 2001.

[22] Y. Na, K.S. Chung, S.-K. Chung, and S.K. Kim. Effects of single-sided inferior turbinectomy on nasal function and airflow characteristics. *Respiratory Physiology & Neurobiology*, 180:289–297, 2012.

[23] Y. Na, K. Kim, S.K. Kim, and S.-K. Chung. The quantitative effect of an accessory ostium on ventilation of the maxillary sinus. *Respiratory Physiology & Neurobiology*, 181:62–73, 2012.

[24] S. Naftali, M. Rosenfeld, M. Wolf, and D. Elad. The air-conditioning capacity of the human nose. *Annals of Biomedical Engineering*, 33(4):545–553, 2005.

[25] A.R. Nathan, R. Eccles, P.H. Howarth, S.K. Steinsvag, and A. Togias. Objective monitoring of nasal patency and nasal physiology in rhinitis. *J Allergy Clin Immunol*, 115(3):S442–S459, 2005.

[26] Kikinis R. Pieper S., Halle M. 3D SLICER. *Proceedings of the 1st IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 632–635, 2004.

[27] Schroeder W. Kikinis R. Pieper S., Lorensen B. The NA-MIC Kit: ITK, VTK, Pipelines, Grids and 3D Slicer as an Open Platform for the Medical Image Computing Community. *Proceedings of the 3rd IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 698–701, 2006.

[28] S. B. Pope. *Turbulent Flows*. Cambridge University Pres, 2000.

[29] J.S. Rhee, S.S. Pawar, G.J.M. Garcia, and J.S. Kimbell. Toward Personalized Nasal Surgery Using Computational Fluid Dynamics. *Arch Facial Plast Surg*, pages E1–E6, 2011.

[30] Corti S. *Simulazione numerica del flusso nelle cavità nasali*. PhD thesis, Politecnico di Milano, 2011.

[31] H. Shi, C. Kleinstreuer, and Z. Zhang. Modeling of inertial particle transport and deposition in human nasal cavities with wall roughness. *J. of Aerosol Science*, 38:398–419, 2007.

[32] F. Sommer, R. Kroeger, and J. Lindemann. Numerical simulation of humidification and heating during inspiration within an adult nose. *Rhinology*, 2012.

[33] C.J.T. Spence, N.A. Buchmann, and M.C. Jermy. Unsteady flow in the nasal cavity with high flow therapy measured by stereoscopic PIV. *Exp Fluids*, 52:569–579, 2012.

[34] C.J.T. Spence, N.A. Buchmann, M.C. Jermy, and S.M. Moore. Stereoscopic PIV measurements of flow in the nasal cavity with high flow therapy. *Exp Fluids*, 50(1005-1017), 2011.

[35] M. Stewart, B.J. Ferguson, and L. Fromer. Epidemiology and burden of nasal congestion. *Int. J. General Medicine*, 3:37–45, 2010.

[36] J. Tan, D. Han, J. Wang, T. Liu, T. Wang, H. Zang, Y. Li, and X. Wang. Numerical simulation of normal nasal cavity airflow in Chinese adult: a computational flow dynamics model. *Eur Arch Otorhinolaryngol*, 2011.

[37] R. J. Volino. A new model for free-stream turbulence effects on boundary layers. *Journal of Turbomachinery.*, 120:613–620, 1998.

[38] D. K. Walters and D. Cokljat. A three-equation eddy-viscosity model for reynolds-averaged navier-stokes simulations of transitional flow. *Journal of Fluids Engineering.*, 130:121401.1–121401.14, 2008.

[39] D. K. Walters and J. H. Leylek. A new model for boundary layer transition using a single-point rans approach. *Journal of Turbomachinery.*, 126:193–202, 2004.

[40] I. Weinhold and G. Mlynski. Numerical simulation of airflow in the human nose. *Eur. Arch. Otorhinolaryngol.*, 261:452–455, 2004.

[41] D. Wexler, R. Segal, and J. Kimbell. Aerodynamic effects of inferior turbinate reduction. *Arch Otolaryngol Head Neck Surg.*, 131:1102–1107, 2005.

[42] G. Xiong, J. Zhan, K. Zuo, J. Li, L. Rong, and G. Xu. Numerical flow simulation in the post-endoscopic sinus surgery nasal cavity. *Medical and Biological Engineering*, 46:1161–1167, 2008.

[43] S. Zachov, P. Muigg, T. Hildebrandt, H. Doleisch, and H.C. Hege. Visual exploration of nasal airflow. *IEEE Transaction on Visualization and Computer Graphics*, 15(6):1407–1414, 2009.

[44] K. Zhao, K. Blacker, Y. Luo, B. Bryant, and J. Jiang. Perceiving nasal patency through mucosal cooling rather than air temperature or nasal resistance. *PLoS ONE*, 6(10):e24618, 2001.

[45] K. Zhao and P. Dalton. The way the wind blows: implications of modeling nasal airflow. *Current Allergy and Asthma Reports*, 7:117–125, 2007.

[46] K. Zhao, P. Dalton, G.C. Yang, and P.W. Scherer. Numerical modeling of turbulent and laminar airflow and odorant transport during sniffing in the human and rat nose. *Chem. Senses*, 31:107–118, 2006.

[47] K. Zhao, P.W. Scherer, S.A. Hajiloo, and P. Dalton. Effect of anatomy on human nasal air flow and odorant transport patterns: implications for olfaction. *Chem. Senses*, 29:365–379, 2004.

[48] J.H. Zhu, H.P. Lee, K.M. Lim, S.J. Lee, and D.Y. Wang. Evaluation and comparison of nasal airway flow patterns among three subjects from Caucasian, Chinese and Indian ethnic groups using computational fluid dynamics simulation. *Respiratory Physiology & Neurobiology*, 175:62–69, 2011.

Ringraziamenti

Alla fine del mio percorso di studi in università sono molte le persone che vorrei ringraziare.

Prima di tutto ringrazio il Professor Quadrio per tutto quello che mi ha insegnato e per avermi dato la possibilità di lavorare con lui.

Un doveroso ringraziamento va anche ai miei predecessori in questo ambito, Stefano Corti e Samuele Zampini, che sono sempre stati molto disponibili nei miei confronti.

Ringrazio anche il Professor Felisati e i suoi collaboratori, Francesco Messina e Carlotta Pipolo, che sono i destinatari di questo lavoro, con la speranza che i pazienti possano trarre beneficio da questo studio e dai suoi sviluppi futuri.

Un caloroso ringraziamento va anche alla Professoressa Spreafichi che mi ha aiutato nella revisione linguistica, mostrando tanta disponibilità.

Ringrazio i miei genitori, mia sorella e la nonna Adele, che mi hanno permesso di portare avanti gli studi e mi hanno sempre incoraggiato e sostenuto in qualsiasi modo. Con loro vanno ringraziati anche il nonno Lino, lo zio Andrea e la zia Claudia, Massimo, Nicola e Diego che mi sono sempre vicini. Ovviamente c'è anche Michele, basta stare con lui un solo minuto per risollevarsi di morale in qualsiasi momento. E poi c'è la nonna Lidia, che a causa della sua malattia non può gioire con me di questo risultato, ma sono sicura che ne sarebbe molto fiera. Colgo anche l'occasione per ricordare il nonno Andrea e la nonna Cesarina, quanto vorrei fossero ancora qui con me.

Vorrei ringraziare anche lo zio Piero, lo zio "acquisito", per la gentilezza con cui ha accolto me e la Gio nella sua casa e per i suoi racconti di vita e di viaggi.

Un saluto speciale va a tutti i compagni di università, con cui ho condiviso le gioie e dolori del Poli. Menzione speciale per Mattia, la Mary e il Verons, ottimi amici oltre che fidati compagni d'avventura all'università. Poi ci sono le amiche di sempre: Fresh, Fra, Vally, Lara, Fuso, Nini, Bonny, Very, Skyzzo e la Gio, con cui ho condiviso i miei anni milanesi. Loro andrebbero ringraziate ogni giorno perché mi sopportano, ma soprattutto perché ci sono sempre, nei momenti felici e in quelli più tristi.

Infine ringrazio tutti gli altri amici e quelli che probabilmente ho dimenticato.