

POLITECNICO DI MILANO

Facoltà di Ingegneria Industriale

Corso di Laurea in  
Ingegneria Aeronautica



Un approccio *libero* all'ottimizzazione di un bulbo per un'imbarcazione classe 950

Relatore : Prof. Luca FORMAGGIA

Co-relatore : Dr. Nicola PAROLINI

Tesi di Laurea di :

Vittorio BISSARO Matr.753359

Anno Accademico 2011 - 2012



# Ringraziamenti

Il primo ringraziamento lo devo al Prof. Luca Formaggia che mi ha permesso di conciliare in questa tesi lo studio e la passione per la vela, mostrandosi incline fin da subito alla genesi un po' particolare di questo lavoro.

Ho anche avuto la fortuna di essere stato direttamente seguito dal miglior correlatore possibile, mi sento in grande debito con il Dr. Nicola Parolini perchè è stato con me paziente e disponibile, entusiasmante e appassionato. Non potevo sperare in una presenza migliore al mio fianco in questa tesi. Voglio ringraziare anche l'Ing. Matteo Lombardi per essere stato un costante riferimento nel momento in cui ho voluto approfondire la conoscenza di OpenFOAM.

Un grazie lo devo a Davide Conti per avermi introdotto alla CFD, per avermi guidato nei primi mesi di brancolamento nel buio e per avermi fatto compagnia in questo percorso di compenetrazione tra vela e università.

Grazie a Giuseppe Aloe e Luca Paglieri per il supporto morale e pratico durante gli afosi mesi estivi in cui il cluster non ne voleva sapere di funzionare, ma più in generale grazie a tutto il dipartimento di matematica MOX perchè è stato un piacere l'aver studiato in una realtà d'eccellenza come la vostra.

Voglio ringraziare anche il prof. Andrea Ratti del MYD, perchè è stata la figura che mi ha fatto vivere le più belle esperienze di vela al Politecnico, non ultima questa stessa tesi.

Ringrazio anche Umberto Felci e Lorenzo Giovannozzi di Felci Yachts Design per i preziosi consigli e l'interessante punto di vista pratico che sono riusciti a darmi nelle purtroppo poche occasioni di incontro.

Il ringraziamento più grande lo voglio fare ai miei compagni d'università. Siamo riusciti a vivere questi 6 anni alternando leggerezza e impegno, serenità e sudate gratificazioni. Dagli esami scorrevoli della triennale fino ai mastodontici progetti della specialistica abbiamo sempre cercato che il motore fosse la passione e non il dovere, trovando sempre del carburante non convenzionale alle nostre ore di studio. Ho l'ambizione di credere di aver lavorato con i mi-

## Ringraziamenti

---

glieri studenti del mio anno e i traguardi raggiunti insieme a voi sono quelli che mi rendono più orgoglioso nel mio percorso universitario.

Un sincero grazie lo devo ai miei amici: quelli della notte, quelli della vela, dei viaggi, delle pazzie o di tutte le cose insieme. Essere stato vostro amico mi ha permesso di vivere con il giusto distacco quelle giornate in cui tornavo a casa con la testa che faceva male per le incomprensibili formule trascritte su un quaderno, o per le alienanti notti passate davanti ad un pc. Mi avete reso un'ingegnere aerospaziale non-convenzionale e di questo ne vado molto fiero.

Grazie ai miei genitori, perchè mi hanno dimostrato di desiderare più di ogni altra cosa la mia felicità; perchè hanno messo da parte un po' di legittimo orgoglio scolastico pur di permettermi di coltivare la mia passione per la vela.

Ai miei fratelli e sorelle e a tutti i miei parenti, perchè ogni occasione di festa è un motivo per sentirsi parte di una famiglia incredibilmente unita, forte e felice.

A Giulia infine, perchè sei la compagna di ogni viaggio che mi immagino davanti.

# Indice

<b>Ringraziamenti</b>	<b>iii</b>
<b>Sommario</b>	<b>xi</b>
<b>Introduzione</b>	<b>xv</b>
<b>1 Definizione del problema</b>	<b>1</b>
1.1 Presentazione <i>Classe 950</i> . . . . .	1
1.2 Definizione delle specifiche di progetto . . . . .	4
1.3 Geometria del bulbo . . . . .	6
1.3.1 Parametrizzazione . . . . .	7
1.4 Flusso di ottimizzazione . . . . .	10
1.4.1 Scelta delle analisi . . . . .	11
<b>2 Analisi fluidodinamica</b>	<b>13</b>
2.1 Modello numerico . . . . .	13
2.1.1 Equazioni RANS . . . . .	14
2.2 Approssimazione numerica del problema . . . . .	17
2.2.1 Soluzione del sistema algebrico . . . . .	18
2.2.2 Scelte risolutive . . . . .	19
2.2.3 Calcolo parallelo . . . . .	20
2.3 Griglie di calcolo . . . . .	22
2.3.1 Condizioni al contorno . . . . .	24
2.3.2 Analisi di convergenza spaziale . . . . .	26
2.4 Deformazione della griglia . . . . .	28
2.4.1 Algoritmo <i>sin-2D</i> . . . . .	28
2.4.2 IDW e implementazione . . . . .	31
2.4.3 Confronto degli algoritmi di deformazione . . . . .	35

<b>3</b>	<b>Tecniche di ottimizzazione</b>	<b>39</b>
3.1	Introduzione . . . . .	39
3.1.1	Formulazione matematica . . . . .	40
3.2	Metodo del gradiente coniugato . . . . .	42
3.2.1	Lineare . . . . .	43
3.2.2	Non lineare: algoritmo di Fletcher-Reeves . . . . .	45
3.2.3	Esempio di funzionamento . . . . .	46
3.3	Metodi genetici . . . . .	48
3.3.1	Algoritmo Genetico Multi Obiettivo (MOGA) . . . . .	49
<b>4</b>	<b>Analisi dei risultati</b>	<b>53</b>
4.1	Poppa . . . . .	53
4.1.1	Termini di resistenza . . . . .	54
4.2	Allround: compromesso bolina / poppa . . . . .	57
4.3	Multi obiettivo . . . . .	60
	<b>Conclusioni</b>	<b>65</b>
	<b>Appendici</b>	<b>69</b>
	<b>Bibliografia</b>	<b>72</b>

# Elenco delle tabelle

1.1	Riepilogo analisi di ottimizzazione condotte . . . . .	11
2.1	Schemi utilizzati nella soluzione di equazioni RANS tramite <i>simpleFoam</i> . . . . .	19
2.2	Solutori utilizzati per i sistemi risultanti dalle equazioni . . . . .	19
2.3	Condizioni al contorno <i>base</i> . . . . .	25
2.4	Condizioni al contorno per la velocità U . . . . .	25
2.5	Confronto differenti griglie . . . . .	26
2.6	Confronto differenti metodi di deformazione - indicatori di qualità	36
4.1	Caratteristiche geometriche diverse geometrie . . . . .	53
4.2	Termini di resistenza per le diverse geometrie . . . . .	55
4.3	Confronto termini di resistenza per $Geo_0$ e $Geo_2$ ai diversi assetti	58
4.4	Componenti supplementari di forze su $Geo_2$ in condizione di bolina	59
4.5	Confronto fra le geometrie selezionate nella famiglia $Geo_3$ con la geometria indeformata $Geo_0$ . . . . .	62
4.6	Licenze dei programmi utilizzati . . . . .	70



# Elenco delle figure

1	Schema a blocchi del <i>toolbox</i> di ottimizzazione creato utilizzando software <i>liberi</i> . . . . .	xvi
1.1	Imbarcazione classe 950 . . . . .	2
1.2	Geometria del bulbo di partenza . . . . .	6
1.3	Deformazioni dovute al parametro P1: <i>strizione</i> . . . . .	8
1.4	Deformazioni dovute al parametro P2: <i>asimmetria</i> . . . . .	9
1.5	Deformazioni dovute al parametro P3: <i>allungamento</i> . . . . .	9
1.6	Calcolare <i>displacement</i> dei nodi del bulbo . . . . .	12
2.1	Zone e scale tipiche di uno strato limite turbolento . . . . .	16
2.2	Studio di scalabilità . . . . .	21
2.3	Indicatori di qualità di una griglia . . . . .	23
2.4	Griglia <i>medium</i> . . . . .	27
2.5	Distribuzione di $y^+$ a parete . . . . .	27
2.6	Suddivisione dominio per l'algoritmo <i>sin-2D</i> . . . . .	29
2.7	Funzione di transizione . . . . .	30
2.8	Deformazione di una sezione attraverso l'algoritmo <i>sin-2D</i> . . . . .	30
2.9	Raggi di influenza . . . . .	32
2.10	Effetto di una generica deformazione - traslazione perpendicolare al piano di simmetria . . . . .	34
2.11	Correzione piano di simmetria . . . . .	35
2.12	Confronto differenti metodi di deformazione - validità della griglia . . . . .	37
3.1	Funzionamento FR - CG: grafico 3D (a) e curve di livello (b) . . . . .	47
3.2	Funzionamento di un algoritmo genetico . . . . .	51
4.1	Geometria del bulbo ottimizzato $Geo_1$ : vista frontale (a) e prospettica (b) . . . . .	54

## Elenco delle figure

---

4.2	Evoluzioni della funzione obiettivo e dell'errore percentuale sul vincolo di momento . . . . .	55
4.3	Evoluzione dei parametri . . . . .	56
4.4	Distribuzione di velocità per $Geo_0$ : vista laterale (a) e superiore (b)	56
4.5	Distribuzione di velocità per $Geo_1$ : vista laterale (a) e superiore (b)	56
4.6	Geometria del bulbo ottimizzato $Geo_2$ : vista frontale (a) e prospettica (b) . . . . .	57
4.7	Linee di flusso intorno a $Geo_2$ all'assetto di bolina . . . . .	58
4.8	Distribuzione di pressione su $Geo_2$ vista dall'alto . . . . .	59
4.9	Fronte di pareto $Geo_3$ . . . . .	60
4.10	Geometria del bulbo ottimizzato $Geo_{3a}$ : vista frontale (a) e prospettica (b) . . . . .	61
4.11	Geometria del bulbo ottimizzato $Geo_{3b}$ : vista frontale (a) e prospettica (b) . . . . .	61
4.12	Geometria del bulbo ottimizzato $Geo_{3c}$ : vista frontale (a) e prospettica (b) . . . . .	62

# Sommario

La disponibilità di risorse computazionali ha aumentato negli anni l'interesse per l'uso della CFD come strumento di analisi. Le ottimizzazioni di forma nella nautica sono diventate un metodo sempre più comune per la scelta della geometria di particolari componenti.

In questa tesi si affronta un problema di ottimizzazione per un bulbo di un'imbarcazione *Classe 950* sviluppando una procedura di analisi generale per l'emissione di specifiche e costruendo un *toolbox* integrato utilizzando solo software *liberi*. Lo studio CFD viene condotto con equazioni RANS su griglie strutturate, indagando il problema della deformazione di griglia e sviluppando sia algoritmi *ex-novo* che adattando algoritmi esistenti al problema specifico. Il processo di ottimizzazione viene condotto con metodi basati sul gradiente e con metodi genetici, sviluppando algoritmi per il vincolo della geometria e evidenziando i diversi studi singolo o multi-obiettivo. I risultati vengono divisi sulla base dell'analisi condotta, mostrando le diverse geometrie risultanti per approssimazioni sempre più complesse del problema affrontato.

**Parole chiave:** ottimizzazioni di forma, *Classe 950*, analisi CFD, software *liberi*, *OpenFOAM*, *DAKOTA*



# Abstract

The availability of computational resources has increased over the years the interest in the use of CFD as an analysis tool. Shape optimization in nautical field has become a more and more common method for the choice of the geometry of particular items.

This thesis faces a bulb optimization problem for a boat *Classe 950* developing a general analysis procedure to produce a list of requisites and building an integrated *toolbox* using only open-source software. The study is performed solving the RANS equations on structured mesh, investigating the problem of mesh deformation and developing both newly algorithms and adapting existing algorithms to the specific problem. The optimization process is conducted using gradient-based methods and genetic ones, developing algorithms for the imposition of constraints on the geometry and discussing different optimization studies (single or multi-objective). The results are divided on the basis of the analysis performed, showing the different geometries resulting for increasingly complex approximations of the problem.

**Keywords:** shape optimization, *Classe 950*, CFD analysis, open-source software, *OpenFOAM*, *DAKOTA*



# Introduzione

Gli strumenti utilizzati per le scelte progettuali nel mondo della nautica hanno subito una forte evoluzione negli ultimi anni.

L'introduzione della fluidodinamica computazionale (CFD, Computational Fluid Dynamics) nel ciclo di progetto ha permesso di analizzare problemi fluidodinamici complessi fornendo risultati sempre più accurati.

L'obiettivo di questa tesi è quello di effettuare uno studio di ottimizzazione di forma per un bulbo di un'imbarcazione *Classe 950* tramite la creazione di un *toolbox* basato su software *libero*.<sup>1</sup>

## Inquadramento generale

Il disegno nautico è stato per molto tempo una materia più di esperienza che di vera scienza. La stessa progettazione di appendici, derive, timoni e altre parti venivano basate su considerazioni qualitative o su studi statistici di progetti di successo.

Alcune realtà isolate di grandi sindacati partecipanti a competizioni come la Coppa America hanno stanziato grandi budget per questi studi ma i risultati venivano celati perchè avversari non ne entrassero in possesso. Anche a causa di ciò l'approccio progettuale non è cambiato molto nel tempo.

Negli ultimi anni ricerche condotte sulla progettazione di appendici o di imbarcazioni da regate sono diventate di grande interesse accademico. Parallelamente la CFD ha acquisito sempre maggiore credibilità prima nell'ambito navale e poi in quello delle imbarcazioni a vela; a questo proposito alcuni dei contributi più importanti si trovano in [1], [2], [3] e [4].

La grande crescita delle potenze di calcolo ha poi aperto la strada ad una metodologia progettuale più scientifica e rigorosa. Studi di ottimizzazione so-

---

<sup>1</sup>Si veda l'appendice per una definizione di software libero.

no diventati sempre più comuni come metodo per scegliere la forma di diversi componenti nell'industria e nel mondo della nautica in particolare. Tra questi la forma del bulbo è sicuramente uno degli oggetti di indagine più interessanti e più richiesti.

Questa tesi si ricollega ad un progetto di un'imbarcazione *Classe 950* presentato dagli studenti Marco Achler, Giuseppe Musca e Laura Rizzo per l'esame finale del MYD Master in Yacht Design al Politecnico di Milano. A partire dalla geometria di bulbo selezionata in quella sede si è sviluppata una metodologia di indagine il più generale possibile perchè potesse essere estesa a problemi simili. Parallelamente si è creato un *toolbox* per la soluzione del problema di ottimizzazione del bulbo utilizzando strumenti di analisi a codice aperto.

In riferimento ad uno schema a blocchi tipico di uno studio di ottimizzazione, rappresentato in figura 1, si sono utilizzati:

1. il solutore fluidodinamico a volumi finiti *OpenFOAM* (Open Field Operation and Manipulation) [5] per la parte di calcolo CFD;
2. l'ottimizzatore *DAKOTA* (Design Analysis Kit for Optimization and Tera-scale Applications) [6] che implementa diversi algoritmi di ottimizzazione;
3. *GNU Octave* [7] come strumento generico di calcolo matriciale, responsabile dell'interfaccia tra i due codici sopracitati.

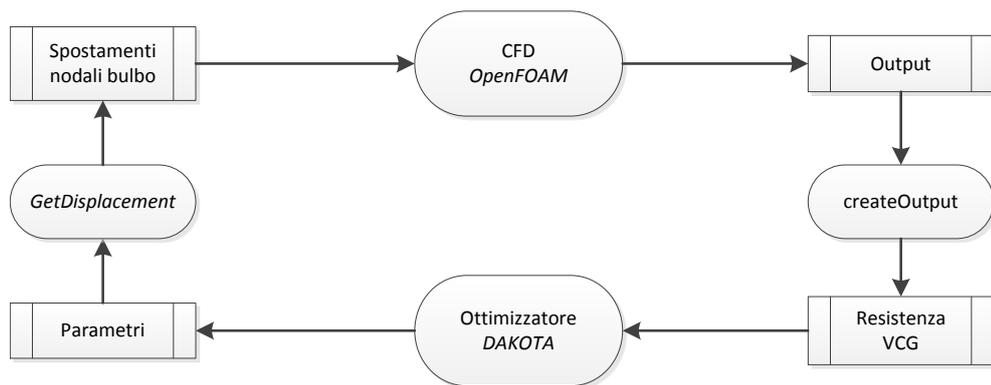


Figura 1: Schema a blocchi del *toolbox* di ottimizzazione creato utilizzando software *liberi*

---

## Scelta degli strumenti

La scelta di utilizzare *OpenFOAM* è stata presa per la curiosità di avvicinarsi alla CFD cercando di apprezzarne al massimo la complessità: il codice obbliga l'utente a confrontarsi con un formalismo e una vastità di solutori/modelli/funzionalità altrimenti nascoste dietro a gradevoli interfacce grafiche.

*OpenFOAM* è principalmente un toolbox C++ per personalizzare ed estendere alcune soluzioni software per la simulazione. È un solutore basato sulla teoria della meccanica dei continui che include la fluidodinamica computazionale. Viene fornito con un'estesa libreria di solutori, arricchita ad ogni nuova release, applicabile ad una sempre più vasta generalità di problemi. È diventato un riferimento nel campo della CFD per i software *liberi* ed è sempre più apprezzato anche al confronto di codici commerciali. Viene distribuito con il post processore *paraView* e una grande varietà di convertitori di griglie tridimensionali anche da pacchetti commerciali.

In quanto software *libero* presenta alcune difficoltà tipiche della maggior parte di software di quel tipo come la comprensione del linguaggio C++, l'utilizzo della Shell di Linux in alternativa ad un'interfaccia grafica e richiede una comprensione approfondita della modellistica matematico/numerica per poter correttamente rappresentare un problema.

*GNU Octave* è un'applicazione software per l'analisi numerica in gran parte compatibile con *Matlab*, la sintassi è molto simile e la maggior parte delle funzioni esistenti sulla soluzione commerciale sono poi presenti anche su *GNU Octave*. La dimestichezza accumulata in questi anni con il linguaggio di programmazione ha reso agevole l'utilizzo del programma come strumento generico di calcolo e per la costruzione dell'interfaccia. *GNU Octave* è scritto in linguaggio C++ e conserva la flessibilità di questo genere di librerie, è estendibile attraverso moduli o funzioni scritte dall'utente e può lavorare assieme a *gnuplot* per disegnare grafici o tabelle.

*DAKOTA* è un toolkit scritto in C++ concepito come una flessibile ed estendibile interfaccia tra codici di simulazione o blackbox esterne, come ad esempio *OpenFOAM*, e una varietà di metodi di analisi iterativa, inclusi l'ottimizzazione, metodo ai minimi quadrati o analisi di sensitività.

Per quel che riguarda il generatore di griglia in principio si è cercato di utilizzare il codice *libero enGrid* [8]. Sono emerse alcune problematiche legate alla deformabilità della griglia (discusse nella sezione 2.3) che hanno fatto optare

per una griglia costruita con il codice commerciale *Ansys Icem CFD* [9].

## Struttura della tesi

La prima macro-sezione della tesi consiste nella costruzione del *toolbox* di ottimizzazione, in riferimento allo schema a blocchi in figura 1 i primi tre capitoli andranno rispettivamente a presentare la costruzione dell'interfaccia, il blocco della CFD e quello dell'ottimizzazione.

Più nello specifico nel **capitolo 1** si presenta il problema da affrontare a partire dallo studio della tipologia di imbarcazione e del regolamento di stazza per l'emissione di specifiche per la ricerca dell'ottimo. Viene quindi presentata la geometria di bulbo da ottimizzare e motivata la parametrizzazione scelta. Infine si mostra il flusso di ottimizzazione andando ad evidenziare i vari blocchi di interfaccia costruiti per la comunicazione fra i vari software e presentate le analisi condotte.

Nel **capitolo 2** si analizza il blocco della CFD presentando il modello numerico RANS utilizzato e le ipotesi effettuate nella sua implementazione numerica con particolare riferimento al formalismo di *OpenFOAM*. Si mostra quindi la griglia di calcolo scelta per l'analisi evidenziandone le caratteristiche in termini di condizioni al contorno e dimensioni. Infine si introduce il problema della deformazione della griglia presentando i due algoritmi implementati nello studio.

Nel **capitolo 3** si presentano i tratti salienti di uno studio di ottimizzazione mostrando la formulazione matematica e la sua discretizzazione numerica. In seguito si mostrano le due tipologie di algoritmo utilizzate; il metodo del gradiente coniugato viene presentato prima nel caso lineare e poi non lineare, mostrando con un esempio grafico il funzionamento. Il metodo multi-obiettivo genetico viene presentato prima attraverso uno studio dei metodi genetici in generale, quindi presentando le particolarità dell'algoritmo utilizzato.

Nel **capitolo 4** si presentano i risultati dello studio mostrando le diverse geometrie ottimizzate per ogni analisi condotta. Si è cercato inoltre di analizzare i risultati evidenziando i motivi fisici alla base dei risultati ottenuti.

# Capitolo 1

## Definizione del problema

In questo capitolo viene presentato il problema specifico di ottimizzazione del bulbo per l'imbarcazione esaminata. Si presenta inizialmente la tipologia di imbarcazione, il contesto in cui si troverà ad operare e vengono tracciate le specifiche che sono emerse dallo studio. Viene quindi presentata la geometria di partenza del bulbo da ottimizzare, motivata la parametrizzazione scelta e vengono mostrate le varie geometrie risultanti introducendo un vincolo da rispettare. Infine si mostra il percorso di ottimizzazione implementato e le analisi condotte.

### 1.1 Presentazione *Classe 950*

La classe 950 è una categoria di imbarcazioni monoscafo a vela ideata nel 2007 da Jean-Marie Vidal.

Come si può vedere in figura 1.1 queste imbarcazioni sono caratterizzate da un piano velico sovradimensionato e da linee d'acqua plananti che le rendono imbarcazioni dalle elevate prestazioni, soprattutto alle andature portanti.

Come genere di imbarcazione si colloca tra il *Mini Transat 650* e il *Class 40*<sup>1</sup>, tra le imbarcazioni di piccole/medie dimensioni più diffuse nella *course au large*<sup>2</sup>.

L'idea alla base è stata quella di creare una classe più abitabile dei *Mini* e più compatta dei *40*, il cui grosso limite è il raggiungimento del budget necessario

---

<sup>1</sup>Si faccia attenzione all'ambiguità della scala utilizzata: i *Mini Transat 650* sono lunghi 6.5 metri, i *Classe 950* sono 9.5 metri mentre i *Class 40* sono 40 piedi, poco più di 12 metri

<sup>2</sup>Nome evidentemente di origine francese che indica una serie di prove d'altura o di navigazione oceanica collocate tipicamente su diversi giorni



Figura 1.1: Imbarcazione classe 950

al sostenimento di una stagione. In quest'ottica di contenimento dei costi il regolamento di stazza esclude certi tipi di materiali nella costruzione e nelle appendici, e limita l'equipaggiamento. Per questo nonostante il difficile periodo finanziario la classe ha avuto una risposta positiva dal mercato.

### **Regolamento di stazza**

Si riportano di seguito gli estratti del regolamento di stazza [10] interessanti in uno studio di ottimizzazione del bulbo:

**101.Appendici** Le appendici sono limitate ad una chiglia, fissa in navigazione (no flap o alette di trim) e due timoni (fissi o sollevabili) massimo. Nota: canards e derive sono proibite.

**204.Pescaggio massimo** Il massimo pescaggio non deve eccedere 2.40 metri (barca in assetto di misura <sup>3</sup>).

**206.Dislocamento** La massa dell'imbarcazione deve essere di almeno 2700 Kg.

### **Calendario 2012 per la Classe 950**

Qui di seguito si riporta il calendario per la stagione 2012, estratto dal sito della classe [11]:

- Roma per Due: (15 – 22 Aprile 2012). Sito: [www.romaper2.com](http://www.romaper2.com). 530 miglia non stop (in doppio). Percorso: Riva di Traiano / Ventotene / Lipari / Ventotene / Riva di Traiano.
- Antipolis: (4 – 6 Maggio 2012) Sito: [www.yc-antibes.net](http://www.yc-antibes.net) 120 miglia non stop (en Équipage) Percorso: Antibes / Porquerolles / Antibes
- Croisiere Bleue: (16–19 Maggio 2012) Sito: [www.yc-antibes.net](http://www.yc-antibes.net) Percorso: Antibes / Calvi et Calvi / Antibes 190 miglia in 2 tappe (in equipaggio).
- Giraglia: (13 – 16 Giugno 2012) Sito: [www.yachtclubitaliano.it](http://www.yachtclubitaliano.it) Percorso: St-Tropez / Genova 240 miglia non stop (in equipaggio)
- Mini-Max: ( 30 Giugno – 08 Luglio 2012) Sito: [www.sngrpc.com](http://www.sngrpc.com) 500 miglia in due tappe (in doppio) Percorso: Port Camargue / Porquerolles / Ajaccio e Ajaccio / Porquerolles / Port Camargue .
- Le grand huit, La grande Motte. 25/08/2012 Corsa in doppio della classe mini. Percorso 700 miglia Sito: <http://www.ycgm.fr/regates>
- Tour della Corsica: (18 – 21 Ottobre 2012) Sito: [www.tourdecorse.org](http://www.tourdecorse.org) 250 miglia non stop (in equipaggio) Percorso: Bonifacio / Bonifacio

---

<sup>3</sup>Nel regolamento di stazza [10] sono presentati i vari assetti di misura.

## 1.2 Definizione delle specifiche di progetto

Come emerge dall'analisi del calendario di regate, il contesto operativo di un'imbarcazione *Classe 950* è una regata d'altura di media-breve distanza e spesso di andata e ritorno.

Nelle regate oceaniche si massimizza il tempo di navigazione alle andature portanti avendo a disposizione una vasta superficie di mare su cui è possibile tracciare la traiettoria ideale tenendo conto delle polari dell'imbarcazione e delle condizioni meteo-marine. Molto spesso tale traiettoria si allontana considerevolmente da quella di minima distanza per l'interesse dei *routier*<sup>4</sup> a portare l'imbarcazione in quelle zone di mare dove si potrà navigare alle andature portanti e quindi a velocità più elevate. Un progetto di un'imbarcazione oceanica è tipicamente specifico per questo tipo di navigazione.

In regate come quelle della *Classe 950* non c'è un'andatura preponderante sulle altre. Nel corso della competizione (soprattutto in quelle andate e ritorno) è probabile che andature di bolina o portanti siano equamente condotte in termini di distanza.

A fronte dell'eterogeneità delle condizioni che è possibile incontrare si è optato per un progetto conservativo che si ponga come utilizzo medio una via di mezzo tra andature di bolina e di poppa. Nel seguito tale filosofia assumerà un'importanza quantitativa nella scelta della funzione da minimizzare.

Analizzando il regolamento di stazza invece evidenziamo alcuni tratti interessanti:

- il massimo pescaggio è limitato
- la chiglia è fissa
- esiste un dislocamento minimo

Il fatto che la chiglia sia fissa ci permette di evidenziare in maniera sufficientemente chiara la condizione di utilizzo del bulbo. L'assetto tipico di navigazione di una barca è noto perchè suggerito dal progettista come condizione di ottimo per la navigazione. La tipologia di analisi risulta chiara in termini di angolo di incidenza e velocità una volta individuata l'andatura da caratterizzare.

Il limite sul pescaggio apre la strada a bulbi con sezioni non simmetriche, in grado di generare maggiori capacità raddrizzanti fissato un volume o lo stesso momento raddrizzante a fronte di una diminuzione del volume. Siccome

---

<sup>4</sup>Il responsabile del carteggio e della scelta della rotta nella *course au large*

non disponiamo di informazioni sufficienti in merito, per scegliere l'analisi da condurre si è scelto di preservare entrambi gli approcci, ponendo il vincolo sul volume o il momento e fornendo risposte diverse a seconda del caso.

Il vincolo sul dislocamento minimo potrebbe far supporre che il peso del bulbo sia scelto come differenza tra il peso complessivo dell'imbarcazione senza bulbo e la massa minima imposta. Sarebbe verosimile credere che l'ottimizzazione debba essere condotta perciò a peso (e quindi volume costante), variando potenzialmente il momento raddrizzante al variare della geometria.

Nel nostro contesto di studio non abbiamo strumenti per apprezzare il miglioramento delle prestazioni ottenibile a fronte di un cambiamento di resistenza o di momento raddrizzante. Questo genere di risposte è possibile ottenerle solo interfacciando il codice di ottimizzazione con un programma VPP (Velocity Prediction Program) che metta in evidenza il peso specifico delle variabili legate al cambiamento della forma del bulbo.

Si è scelto però di caratterizzare comunque un trade-off tra il contributo al miglioramento della performance dovuto ad una diminuzione di resistenza o ad un miglioramento delle capacità raddrizzanti specifiche.

Per mantenere quanto più generale il campo di applicazione del modello presentato in questa tesi si è deciso di considerare diversi possibili studi di ottimizzazione: in un primo caso andremo a minimizzare la sola resistenza, mettendo a confronto geometrie capaci di garantire lo stesso momento raddrizzante e quindi potenzialmente con volumi diversi (al variare del VCG (Vertical Center of Gravity)).

In un secondo momento verrà effettuata una minimizzazione nei due obiettivi di resistenza e VCG, in questo caso il vincolo di geometria verrà imposto sul volume.

Riassumendo i tratti salienti dell'ottimizzazione che conduciamo sono:

- minimizzazione di resistenza del bulbo isolato
- possibilità di scegliere l'assetto a cui valutare la resistenza
- possibilità di vincolare il volume o il momento raddrizzante nell'ottimizzazione
- VCG come indice di prestazione raddrizzante

### 1.3 Geometria del bulbo

La geometria del bulbo di partenza è stata disegnata in fase di progetto di un'imbarcazione della *Classe 950* per un esame finale del MYD<sup>5</sup>; in figura 1.2 si presenta la geometria.

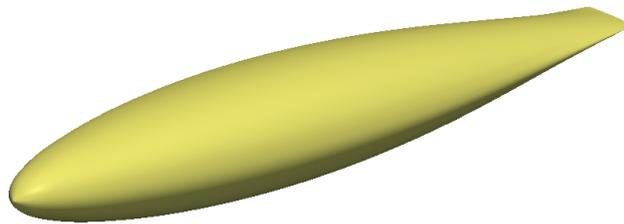


Figura 1.2: Geometria del bulbo di partenza

Nel progetto preliminare tipicamente si prendono decisioni conservative e dettate molto spesso dall'esperienza o da regressioni storico-statistiche su modelli simili.

Significa che la geometria è probabilmente un buon punto di partenza per un'imbarcazione di questo tipo, ma allo stesso tempo è verosimile credere che possa essere migliorata o semplicemente resa più specifica per il caso in esame.

Negli ultimi anni la progettazione del bulbo si è divisa in due scuole di pensiero differenti rispetto alla trattazione o meno dell'eventuale transizione da strato limite laminare a turbolento. Le forme che privilegiano un ritardo nella transizione sono quelle caratterizzate da uno spessore massimo relativamente arretrato, garantendo così un gradiente di pressione favorevole per una elevata percentuale della lunghezza del bulbo e stabilizzando lo strato limite.

Nel campo fluidodinamico computazionale questa eventualità è stata approfondita con alcuni codici commerciali come *CFX*, si veda [12] e [3], e solo nell'ultimissimo periodo si è presentata la possibilità di utilizzare lo stesso *OpenFOAM* per caratterizzare la transizione. In fase di impostazione della te-

---

<sup>5</sup>Per i dettagli sugli autori si veda il capitolo di introduzione.

si questa potenzialità era ancora poco sviluppata e quindi non è stata presa in considerazione nella nostra analisi.

Non potendo apprezzare una eventuale transizione è sembrato poco interessante valutare l'influenza di un parametro come la posizione relativa del massimo spessore.

Riprendendo invece i tratti salienti individuati in sezione 1.2 sull'emissione delle specifiche si è scelta una parametrizzazione che permettesse di deformare una sezione circolare verso una ellittica o rendendo asimmetrica la sezione. In tal modo si sono potute valutare geometrie di bulbo con prestazioni raddrizzanti differenti.

Allo stesso tempo si è voluto cogliere l'importanza della variazione di lunghezza, ottenendo bulbi relativamente più tozzi o allungati, per andare a caratterizzare il contributo di resistenza dovuto alla pressione o agli sforzi tangenziali viscosi (legati in prima approssimazione alla sezione frontale e alla superficie bagnata) e per valutarne l'effetto al variare dell'angolo di incidenza.

### 1.3.1 Parametrizzazione

I tre diversi gradi di libertà che abbiamo evidenziato sono tutti ottenibili attraverso deformazioni globali della geometria, ovvero attraverso semplici leggi algebriche che modificano le coordinate dei nodi superficiali.

A differenza di una parametrizzazione locale, per esempio attraverso dei punti di controllo di spline che modellano la superficie, questa scelta permette con un numero contenuto di parametri di poter rappresentare una vasta famiglia di bulbi differenti.

I tre parametri sono perciò:

- P1: strizione sezione
- P2: asimmetria sezione
- P3: allungamento longitudinale

I vari parametri modificano le coordinate della geometria  $\mathbf{x} = (x, y, z)$  con il seguente algoritmo<sup>6</sup>:

---

<sup>6</sup>La geometria deformata risente della collocazione spaziale di partenza. L'algoritmo è scritto per una geometria centrata nell'origine i cui assi baricentrici principali d'inerzia coincidono con gli assi cartesiani. Attraverso opportune rotazioni o traslazioni è possibile ricondurre qualsiasi geometria in questo sistema di riferimento.

## Definizione del problema

---

$$\begin{aligned}x &\leftarrow \frac{x \cdot (P1 + 1)}{\sqrt{(P3 + 1)}} \\y &\leftarrow (P3 + 1) \cdot y \\z &\leftarrow \frac{z}{(P1 + 1) \cdot \sqrt{(P3 + 1)}}\end{aligned}$$

**if**  $z < 0$

$$z \leftarrow \frac{z}{(P2 + 1)}$$

**else**

$$z \leftarrow z \cdot \left(1 + \frac{1}{2} \sin\left(P2 \cdot \frac{\pi}{2}\right)\right)$$

**end**

Si noti che un valore nullo dei tre parametri corrisponde alla configurazione originale. Si riportano nelle figure 1.3-1.5 gli effetti delle varie deformazioni della geometria negli intervalli di variazione di interesse. Tutte le deformazioni sono applicate con il vincolo di volume costante, condizione che viene sviluppata in sezione 1.4.

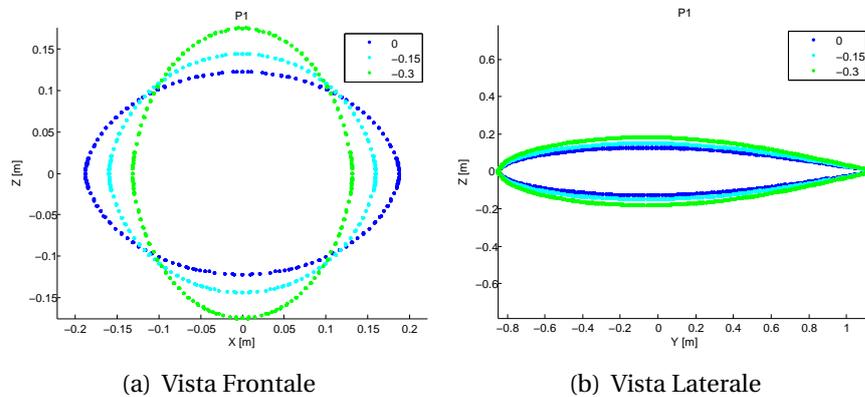


Figura 1.3: Deformazioni dovute al parametro P1: *strizione*

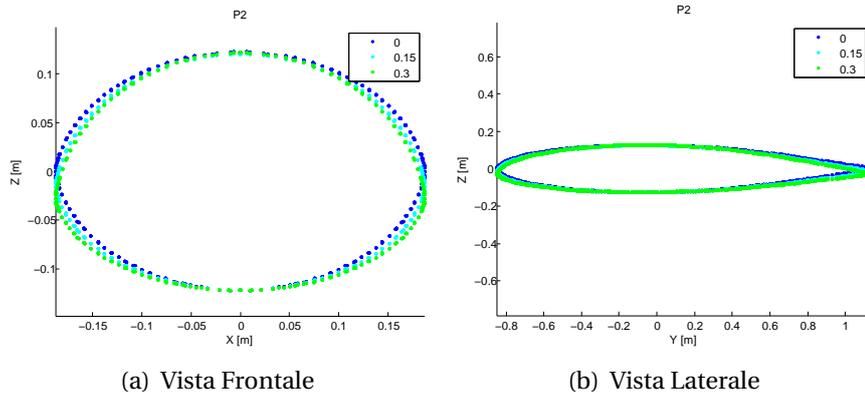


Figura 1.4: Deformazioni dovute al parametro P2: *asimmetria*

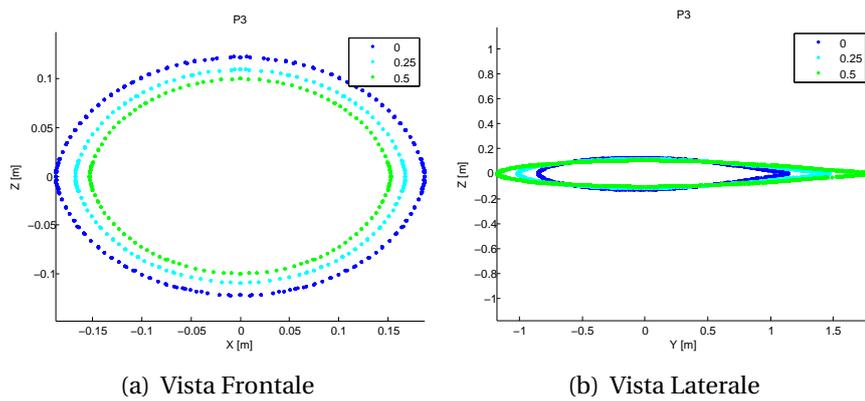


Figura 1.5: Deformazioni dovute al parametro P3: *allungamento*

## 1.4 Flusso di ottimizzazione

Lo schema a blocchi tipico del flusso di ottimizzazione è quello mostrato in figura 1. Per poter interfacciare *DAKOTA* e *OpenFOAM* è necessario sviluppare i blocchi di *createOutput* e *GetDisplacement*.

Il blocco *createOutput* è limitato ad una ricostruzione del valore di resistenza e di VCG in una forma leggibile da *DAKOTA*, risultato ottenibile facilmente manipolando i file di testo risultanti.

Il blocco *GetDisplacement*, mostrato in figura 1.6, racchiude un algoritmo che a partire da un set di parametri suggerito da *DAKOTA* fornisce in output un vettore contenente lo spostamento di ogni nodo della mesh del bulbo tenendo conto delle deformazioni e del vincolo imposto. L'intero codice del blocco *GetDisplacement* è stato scritto con *GNU Octave*.

Una prima parte del codice si occupa di raccogliere le informazioni necessarie sfruttando le funzioni di gestione/lettura di file di testo. Si leggono i parametri forniti da *DAKOTA*, si raccolgono quindi le coordinate dei punti di contorno del bulbo sfruttando i file di testo scritti in formato ASCII da *OpenFOAM* nella descrizione della mesh.

Con il set di parametri si applicano le deformazioni ad un file STL (STereoLithography) molto raffinato descrivente la geometria. A valle della deformazione si verifica che la geometria deformata rispetti il vincolo imposto. Si calcola quindi il volume o il momento in maniera agevole sfruttando la discretizzazione superficiale triangolare tipica dei file STL, ricavando il volume come sommatoria dei volumi dei tetraedi costruiti a partire da uno stesso centro aventi come area di base il triangolo superficiale.

Se il vincolo non è rispettato si itera il ciclo andando ad aggiornare un fattore di scala  $s$ , inizialmente uguale ad 1, con cui poi vengono scalate le dimensioni del bulbo. L'aggiornamento sul volume viene calcolato con l'algoritmo seguente:

### Check volume

$$\Delta V = \frac{V(\mathbf{x}_{def})}{V(\mathbf{x}_0)}$$
$$s = 1$$

**while**  $|\Delta V - 1| > toll$

$$s = s \cdot \sqrt[3]{\Delta V}$$

$$\mathbf{x}_{def} = \mathbf{x}_0 \cdot s$$

$$V_{def} = V(\mathbf{x}_{def})$$

$$\Delta V = \frac{V(\mathbf{x}_{def})}{V(\mathbf{x}_0)}$$

**end(while)**

dove  $V(\cdot)$  è una funzione che dato un vettore di coordinate superficiali nodali restituisce il volume del bulbo,  $\Delta V$  è un errore tra il volume della geometria indeformata e quella deformata,  $s$  è un fattore di scala,  $\mathbf{x}_0 = (x_0, y_0, z_0)$  è il vettore di coordinate nodali indeformate e  $\mathbf{x}_{def} = (x_{def}, y_{def}, z_{def})$  è il vettore di coordinate nodali deformate.

### 1.4.1 Scelta delle analisi

Evidenziati i vari blocchi del processo di ottimizzazione si ricapitolano le varie analisi condotte.

ID	Descrizione	Vincolo	Obiettivo
1	Poppa	Momento	R1
2	Allround	Momento	R1+R2
3	MultiObiettivo	Volume	R1+R2, VCG

Tabella 1.1: Riepilogo analisi di ottimizzazione condotte

Gli obiettivi minimizzati sono:

**R1** Minimizzazione di resistenza del bulbo su metà dominio di calcolo a  $V_\infty = 5m/s$  all'assetto di  $0^\circ$ .

**R2** Minimizzazione di resistenza del bulbo sull'intero dominio di calcolo a  $V_\infty = 5m/s$  all'assetto di  $20^\circ$  di rollio e  $3^\circ$  di scarroccio.

**VCG** Minimizzazione del VCG.

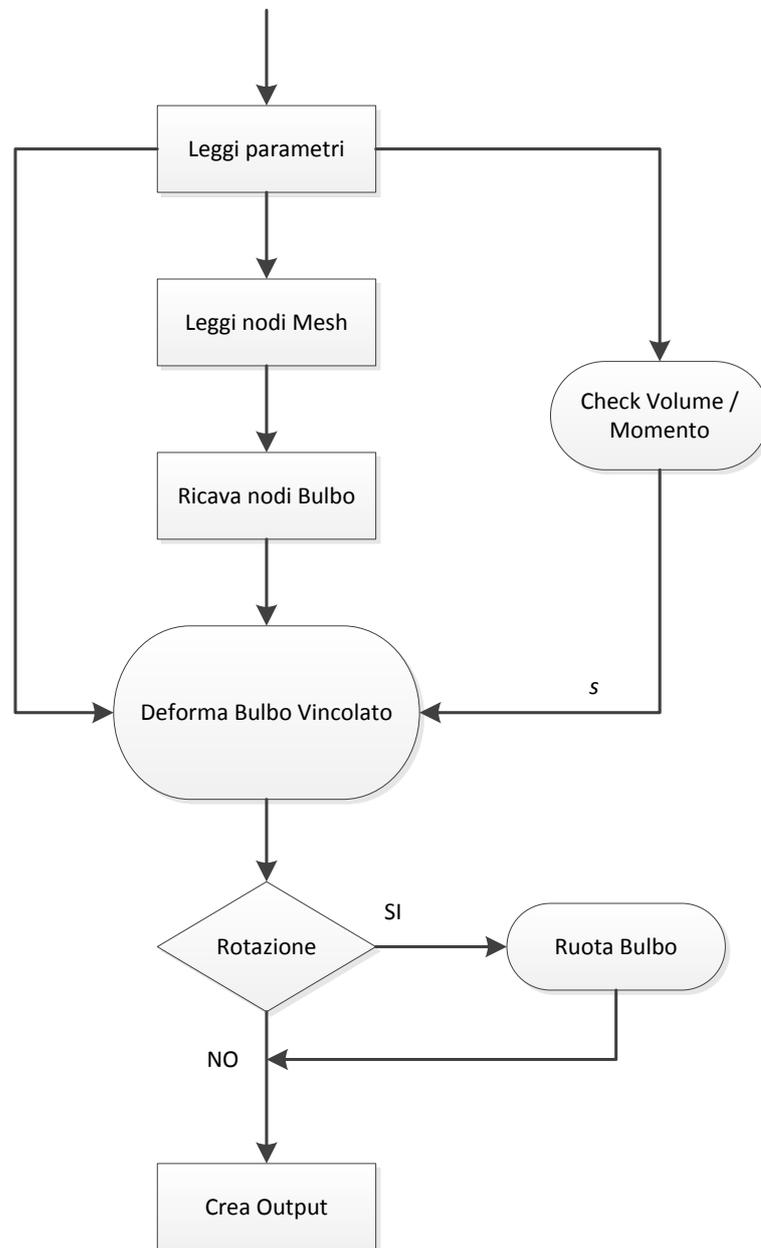


Figura 1.6: Calcolare *displacement* dei nodi del bulbo

# Capitolo 2

## Analisi fluidodinamica

L'analisi fluidodinamica è stata condotta per via numerica utilizzando un modello che tiene conto della turbolenza senza risolverla in tutte le scale, ovvero le equazioni RANS (Reynolds Averaged Navier-Stokes).

Innanzitutto viene presentato il modello alla base di questa trattazione con le varie ipotesi assunte. Quindi viene descritta la sua implementazione nel codice di calcolo *OpenFOAM* e il tipo di solutore utilizzato; si ricorda che non è interesse di questo lavoro focalizzarsi sul solutore CFD, presentato in maniera piuttosto sintetica, per ogni dettaglio si può consultare [13].

Si presentano poi le griglie di calcolo utilizzate nell'analisi CFD e l'imposizione delle condizioni al contorno.

Infine viene analizzato il problema della deformazione della griglia in un contesto di ottimizzazione numerica con i diversi algoritmi utilizzati.

### 2.1 Modello numerico

Le equazioni che governano il moto dei fluidi sono note come equazioni di *Navier-Stokes* ed esprimono il bilancio di massa, di quantità di moto e di energia della corrente. Nel caso semplificato di flussi incomprimibili l'equazione di bilancio dell'energia risulta disaccoppiata dall'equazione di continuità e da quella della quantità di moto, non viene perciò considerata in questa trattazione.

Nell'ulteriore ipotesi di regime di moto stazionario le equazioni di bilancio che governano il problema si riducono a:

$$\nabla \cdot \mathbf{u} = 0 \quad (2.1)$$

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{F} \quad (2.2)$$

dove  $\mathbf{u}$  è il vettore velocità,  $p$  è la pressione,  $\rho$  è la densità,  $\mu$  è la viscosità dinamica e  $\mathbf{F}$  sono le eventuali forze di campo.

### 2.1.1 Equazioni RANS

Avendo a che fare con flussi ad elevato numero di Reynolds non è possibile dal punto di vista computazionale risolvere problemi complessi utilizzando un'integrazione diretta delle equazioni di *Navier-Stokes*<sup>1</sup>.

Si ricorre invece ad un metodo che caratterizza la componente turbolenta della corrente attraverso una sua ricostruzione a partire dalle componenti medie del moto.

In altre parole la velocità viene suddivisa in un contributo medio e uno turbolento:

$$\mathbf{u} = \bar{\mathbf{u}} + \mathbf{u}' \quad (2.3)$$

dove il contributo medio è dato da:

$$\bar{\mathbf{u}} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_T^0 \mathbf{u}(t) dt \quad (2.4)$$

Nell'ipotesi che le fluttuazioni  $\mathbf{u}'$  siano a media nulla è possibile riscrivere le equazioni di partenza nelle quali è stata sostituita l'espressione della velocità.

$$\nabla \cdot \bar{\mathbf{u}} = 0 \quad (2.5)$$

$$\rho \bar{\mathbf{u}} \cdot \nabla \bar{\mathbf{u}} = -\nabla \bar{p} + \mu \nabla^2 \bar{\mathbf{u}} - \rho \nabla \cdot \overline{(\mathbf{u}' \mathbf{u}'^T)} + \mathbf{F} \quad (2.6)$$

Nell'equazione della quantità di moto compare il termine degli *sforzi di Reynolds*  $\overline{(\mathbf{u}' \mathbf{u}'^T)}$  che si aggiunge ad un'equazione altrimenti del tutto simile a quella originale riguardante però ora solo i termini medi. In altre parole tutto il contributo turbolento è annidato nel termine degli *sforzi di Reynolds*.

Il problema di chiusura delle equazioni RANS consiste quindi nel cercare di esprimere l'azione dei termini fluttuanti sul moto medio, riconducendo però la valutazione del tensore degli sforzi turbolenti a solo grandezze relative al moto medio stesso.

---

<sup>1</sup>Attraverso la tecnica definita DNS (Direct Numerical Simulation).

## Modelli di Turbolenza

Esistono numerosi modelli di turbolenza e possono essere schematizzati nelle seguenti categorie:

- modelli algebrici
- modelli basati su equazioni differenziali (modelli a una o due equazioni)
- modelli basati sull'equazione di trasporto del tensore degli *sforzi di Reynolds*

Le prime due categorie si basano sull'ipotesi di Boussinesq [14] che assume di poter esprimere il termine degli sforzi di Reynolds in funzione del tensore della velocità di deformazione del moto medio attraverso l'introduzione di una viscosità turbolenta isotropa. La viscosità turbolenta è uno scalare analogamente alla viscosità molecolare, ma non ha nulla a che fare con quest'ultima. Non trae origine infatti da proprietà fisiche del fluido ma dalle fluttuazioni del suo moto, separate dal moto medio.

Nell'ambito dell'ipotesi di Boussinesq, il problema della modellazione degli sforzi di Reynolds è quindi ricondotto alla determinazione della viscosità turbolenta. Problema affrontato come detto da semplici chiusure di tipo algebrico (modelli di ordine zero) a modelli differenziali ad una o più equazioni, dove la viscosità turbolenta è determinata sulla base di una o più proprietà scalari della corrente valutate con equazioni differenziali di trasporto ricavate direttamente dalle equazioni di *Navier-Stokes*.

Si è scelta quindi una modellazione che trova ampio uso in ambito industriale nel campo della nautica ovvero il *modello a due equazioni  $k - \omega$ SST* (Shear Stress Transport) [15].

Il modello  $k - \omega$ SST è una combinazione dei due modelli  $k - \epsilon$  e  $k - \omega$ . Il modello di turbolenza  $K - \omega$  fornisce risultati molto accurati in prossimità di una parete ma la qualità diminuisce allontanandosi da quest'ultima. Il modello  $k - \epsilon$  invece fornisce buoni risultati nelle correnti libere.

Il modello  $k - \omega$ SST permette di selezionare l'uno o l'altro nelle regioni in cui si comportano meglio, caratterizzando al meglio il flusso in tutta la regione di moto.

Per definire completamente il modello è necessario quantificare i coefficienti di chiusura che compaiono nelle equazioni differenziali. Non è l'obiettivo di questa tesi una validazione di tale modello, le equazioni e i valori assunti

dai coefficienti non vengono quindi riportati per sintesi: per ulteriori dettagli si rimanda a [15] e [19].

### Leggi di parete

Lo strato limite turbolento è suddiviso in diverse zone caratterizzate da profili di velocità e scale differenti, evidenziate in figura 2.1.

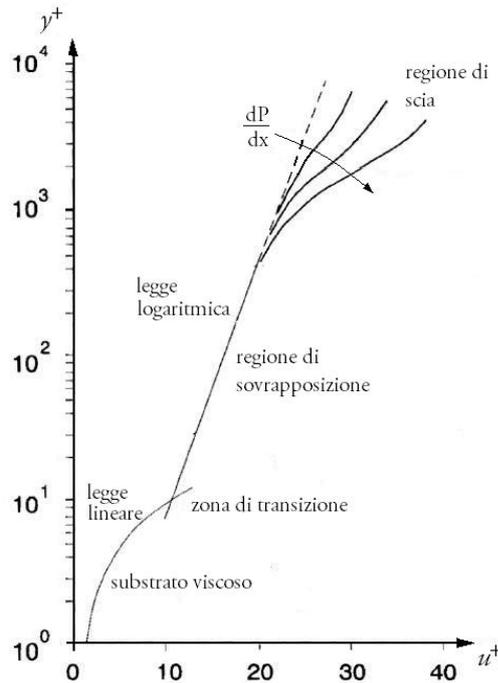


Figura 2.1: Zone e scale tipiche di uno strato limite turbolento

La grandezza adimensionale  $y^+$  è definita:

$$y^+ = \frac{u^* y}{\nu} \quad (2.7)$$

dove  $\nu$  è la viscosità cinematica,  $y$  è l'altezza della prima maglia a parete e  $u^* = \sqrt{\tau_w / \rho}$  è la velocità di frizione a parete. La distribuzione di  $y^+$  è un utile indicatore della corretta discretizzazione superficiale di una griglia.

Nella zona di parete (substrato viscoso, zona di transizione e regione logaritmica) lo sforzo totale tangenziale, dato dalla somma di uno sforzo viscoso e di uno turbolento, è circa costante; si veda [17] per una formulazione numerica completa. A parete solo lo sforzo viscoso è non nullo, allontanandosi da parete

lo sforzo turbolento aumenta con l'aumentare della lunghezza di rimescolamento disponibile. A  $y^+ = 10$  circa si equivalgono e nella regione logaritmico il primo è circa trascurabile rispetto al secondo.

L'approccio più corretto per risolvere uno strato limite turbolento è quello di infittire la discretizzazione a parete nella zona interna e di utilizzare in questa zona modelli di turbolenza adatti a caratterizzare regioni in cui gli effetti viscosi sono dominanti, ovvero i cosiddetti modelli *low-Reynolds*. In termini di risorse computazionali non è sempre possibile seguire tale approccio, è più efficiente usare griglie più fitte dove i gradienti del moto medio sono elevati e cercare di stimare il flusso attraverso delle leggi di parete.

Le leggi di parete sono derivate per flussi semplici attaccati e in equilibrio, in questo caso nella regione logaritmica i termini di produzione e di dissipazione sono molto maggiori di quello diffusivo e convettivo, si ipotizza quindi una legge logaritmica semplificata e si ipotizzano delle relazioni approssimate fino a parete per alcune grandezza di interesse come  $k$ ,  $\omega$  (si veda [17]). Con una prima cella delle dimensioni di  $100 > y^+ > 30$  tale processo di stima delle grandezze di interesse nella zona interna fornisce risultati ragionevolmente accurati a patto che il regime di moto della corrente sia simile a quello per cui si sono ricavate tali relazioni; in figura 2.5 si riporta la distribuzione di tale grandezza sulla superficie. L'approssimazione nel nostro caso risulta accettabile perchè il regime di moto è caratterizzato da gradienti di pressione limitati e una trascurabile percentuale della superficie del bulbo interessata da fenomeni di distacco di scia.

## 2.2 Approssimazione numerica del problema

Le equazioni introdotte non sono in generale risolvibili per via analitica, si ricorre perciò ad una tecnica di approssimazione numerica. Tale approssimazione consiste in due passi: la discretizzazione spaziale (e temporale se il problema è non stazionario) e il trattamento delle non linearità e della natura accoppiata delle equazioni di *Navier-Stokes*. Questo permette di trasformare un sistema di equazioni differenziali in un sistema di equazioni algebriche, le cui soluzioni sono i valori delle variabili assunte in pre-definiti punti nello spazio.

La discretizzazione spaziale può essere condotta con diverse tecniche, nel nostro caso è implementata un'approssimazione ai volumi finiti (FVM, Finite Volume Method); il dominio viene suddiviso in un numero finito di volumi di

controllo, le equazioni di governo sono scritte in forma integrale e risolte per ogni cella.

Le incognite sono quindi i valori mediati della grandezza di interesse nel centro della cella che approssima il valore continuo. Naturalmente esistono diversi schemi per estrapolare il valore delle incognite in un punto preciso o per valutare i gradienti o derivate ed è necessario spesso prendere alcune accortezze; in riferimento per esempio ad un generico problema di diffusione-trasporto se approssimiamo il termine di derivata spaziale con uno schema centrato si può incorrere in instabilità nella soluzione se il problema è a trasporto dominante, per maggiori dettagli si rimanda a [16]. Per le derivate spaziali si utilizzano perciò schemi decentrati o *upwind* che sfruttando la conoscenza della direzione del flusso stabilizzano la soluzione abbattendo però di un ordine l'approssimazione.

### 2.2.1 Soluzione del sistema algebrico

Dopo aver discretizzato il sistema di RANSE si ottiene un'equazione algebrica per ogni volume di controllo, tale sistema è accoppiato ed è costituito da equazioni non lineari.

È possibile la risoluzione sia in forma accoppiata che segregata; per motivi di efficienza l'approccio usato è il secondo, più nello specifico lo schema adottato è quello noto con l'acronimo PISO (Pressure Implicit Solution). L'algoritmo PISO funziona schematicamente nel seguente modo:

- il gradiente di pressione non è noto, si utilizza il campo di pressione all'iterata precedente per risolvere l'equazione del momento.
- nota una prima stima della velocità si valuta la pressione.
- il campo di velocità viene corretto sulla base della nuova distribuzione di pressione.

I problemi algebrici ottenuti sono tipicamente caratterizzati da matrici sparse di grandi dimensioni che vengono risolti con metodi iterativi come, ad esempio, il metodo di Gauss-Seidel o metodi multigrid algebrici.

Nel caso di simulazioni stazionarie l'algoritmo utilizzato è una semplificazione del metodo PISO ovvero il metodo SIMPLE (Semi Implicit Method for Pressure Linked Equations). Nonostante non sia necessaria una derivazione

nel tempo nell'equazione del momento viene comunque mantenuta tale pseudo-integrazione per migliorare le caratteristiche di stabilità. Ottenuto un campo di velocità e pressione con l'algoritmo SIMPLE si aggiornano infine le quantità scalari trasportate dal campo di velocità.

### 2.2.2 Scelte risolutive

In questa sezione si vogliono riassumere le scelte effettuate per la risoluzione dell'equazioni.

Tutte le simulazioni sono stazionarie, si è optato quindi per il solutore *simpleFoam* del software *OpenFOAM*.

Per la soluzione con la deformazione della griglia implementata all'interno di *OpenFOAM* si è scritto un solutore ad hoc *SimpleFoamWithMotion* che facesse una chiamata di *mesh update* all'inizio del calcolo, salvo poi proseguire con i medesimi solutori del caso stazionario.

I vari termini dell'equazione sono stati risolti con gli schemi riportati in tabella 2.1.

ddtSchemes	steadyState
gradSchemes	Gauss linear / upwind
laplacianSchemes	Gauss linear corrected
interpolationSchemes	linear
snGradSchemes	corrected

Tabella 2.1: Schemi utilizzati nella soluzione di equazioni RANS tramite *simpleFoam*

In tabella 2.2 sono invece riportati i solutori delle equazioni utilizzati.

p	GAMG
U	GaussSeidel
k/ $\omega$	GaussSeidel

Tabella 2.2: Solutori utilizzati per i sistemi risultanti dalle equazioni

### 2.2.3 Calcolo parallelo

La dimensione del problema e le numerose valutazioni che devono essere condotte per poter effettuare uno studio di ottimizzazione hanno reso indispensabile l'utilizzo di più processori per il calcolo CFD. *OpenFOAM* implementa funzioni per il calcolo parallelo basate sulla decomposizione del dominio, in cui la geometria e i campi associati vengono separati in sotto-domini e allocati a diversi processori.

Il processo di calcolo parallelo coinvolge quindi la decomposizione della griglia e dei campi, il calcolo in parallelo e la valutazione a posteriori del caso.

La decomposizione del dominio viene effettuata con la funzione *decomposePar* che prevede diverse tipologie di divisione e richiede un certo numero di parametri, informazioni presenti in un dizionario specifico. I nostri casi sono stati divisi in 8 parti utilizzando il metodo *scotch*, metodo che non prevede nessuna informazioni geometrica di ingresso dall'utente e cerca di minimizzare la necessità di comunicare tra i vari sotto-domini.

Un caso decomposto viene lanciato in parallelo utilizzando l'implementazione di dominio pubblico *openMPI* dell'interfaccia di passaggio dei messaggi MPI (Message Passing Interface). *openMPI* prevede di poter lanciare calcoli in parallelo sia su diversi processori di una stessa macchina sia su diverse macchine all'interno di un cluster.

Infine per valutare i risultati si può sia visualizzare ogni sottodominio alla volta sia ricostruire l'intero dominio attraverso la funzione *reconstructPar*. Un caso viene ricostruito unendo i risultati contenuti nelle cartelle di ogni processore in un'unica cartella.

### Analisi di scalabilità

Nella valutazione di algoritmi per il calcolo parallelo è importante tenere conto della legge di Amdahl. Tale legge parte dal presupposto che in ogni applicazione ci sia una parte non parallelizzabile, che sostanzialmente limita l'aumento di velocità di calcolo ottenibile in maniera asintotica. Il vantaggio ottenuto aumentando il numero di processori utilizzati può infatti incidere solamente sui tempi di calcolo della parte parallelizzabile.

Nel caso particolare della CFD, un calcolo parallelo viene lanciato partizionando il dominio; perciò la risoluzione necessita anche di un processo di comunicazione tra i vari sotto-domini che diventa tanto più costoso quanto più aumenta il numero di partizioni. In altre parole all'aumentare del numero di

sotto-domini i tempi di calcolo diminuiscono mentre aumentano i tempi di comunicazione.

Si definisce lo *Speedup*  $S$ , indicatore di prestazione del calcolo parallelo, come il rapporto fra il tempo di calcolo seriale  $t_{seq}$  e quello parallelo  $t_{par}$ :

$$S = \frac{t_{seq}}{t_{par}} \quad (2.8)$$

Per i motivi evidenziati si è deciso di effettuare uno studio di scalabilità, ovvero si è lanciato il medesimo calcolo su un numero via via crescente di processori, cercando quindi di evidenziare una relazione tra le risorse utilizzate e la velocità di calcolo; per ulteriori dettagli si veda [18].

Nella figura seguente 2.2 seguente si mostra lo *Speedup* in funzione del numero dei processori, si nota come il massimo aumento di prestazioni si ottiene per 8 processori. Oltre tale numero ogni ulteriore partizione diventa addirittura nociva: il costo di comunicazione tra le varie partizioni diventa maggiore del vantaggio legato alla suddivisione del calcolo. Si decide quindi per lo studio di limitare il numero di processori a 8.

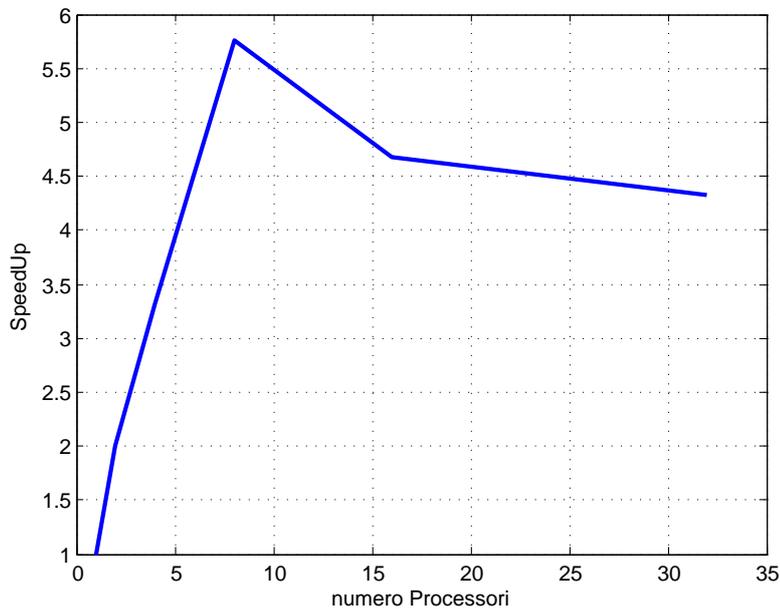


Figura 2.2: Studio di scalabilità

## 2.3 Griglie di calcolo

Esistono due grandi famiglie di tipologie di griglie di calcolo, ovvero quelle *strutturate* e *non strutturate*.

Le griglie *strutturate* sono composte tipicamente da esaedri e la loro costruzione segue una logica a blocchi, basata sulla creazione di una mappa fra un dominio cubico di riferimento e il dominio fisico della geometria.

La distribuzione dei nodi è quindi ricostruibile agevolmente, i solutori risultano più efficienti ed è limitata la propagazione di errori numerici. I problemi dell'utilizzo di queste griglie risiedono nella difficoltà di costruzione attorno a geometrie complesse e nella ridotta capacità di addensare i nodi in zone di interesse.

Le griglie *non strutturate* sono costituite solitamente da tetraedri e sono molto più flessibili rispetto a geometrie complesse o ad addensamenti specifici, tuttavia è necessario costruire un'apposita indicizzazione per risalire alle posizioni dei vari nodi, rendendo più lenta la risoluzione numerica e più sensibile alla propagazione di errori numerici. Questo tipo di griglia è perciò fortemente sconsigliato in regioni come quelle di strato limite.

Un compromesso interessante è realizzato dalle cosiddette griglie *ibride* dove la zona di strato limite viene discretizzata in forma strutturata mentre la zona esterna viene risolta in maniera non strutturata.

### Indicatori di qualità

Per garantire un risultato accurato non basta avere la giusta risoluzione spaziale in aree specifiche, ma la qualità stessa della griglia deve essere garantita. Si evidenziano tre parametri, in accordo con [20], che permettono di caratterizzare la qualità di una griglia, riportati nelle figure 2.3:

- Non-ortogonalità (Non-orthogonality): è il complementare dell'angolo formato tra la normale di una faccia e la congiungente i centri delle due facce
- Distorsione (Skewness): il rapporto delle distanze tra la distanza tra i centri di due facce e la distanza tra il centro della faccia e la congiungente i due centri.
- Allungamento (Aspect-ratio): il rapporto tra le dimensioni di una cella.

Una griglia è considerata di buona qualità se tutte le celle hanno basse non-ortogonalità e distorsioni ridotte. Sono accettabili invece celle con elevati allungamenti a patto che il flusso sia all'incirca parallelo alle facce allungate<sup>2</sup>; relazione ottenibile esclusivamente attraverso l'utilizzo di griglie strutturate.

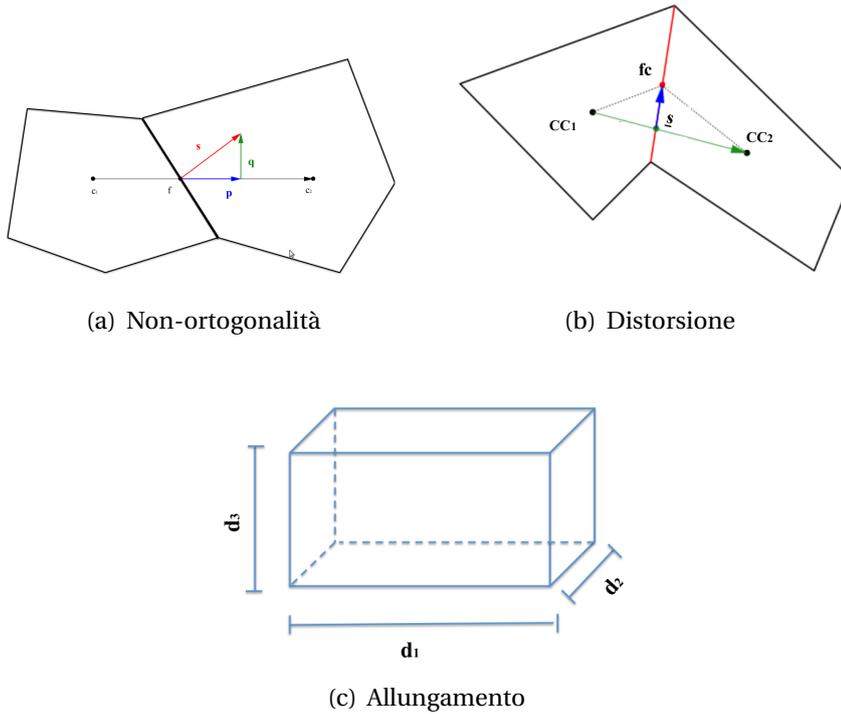


Figura 2.3: Indicatori di qualità di una griglia

<sup>2</sup>Discretizzazione spaziale tipicamente effettuate per lo strato limite.

### Scelta della griglia di calcolo

La scelta ottimale della griglia è un problema non banale perchè diverse considerazioni sulla bontà di una scelta possono essere fatte solo a posteriori.

Nel nostro caso specifico esistono alcuni vincoli che devono essere rispettati per poter fornire risultati credibili:

- garantire una buona qualità di griglia iniziale per avere margine rispetto alla deformazione richiesta dai vari step di ottimizzazione
- garantire  $y^+$  entro certi valori per avere una rappresentazione corretta dello strato limite e quindi della stima di resistenza

Mentre la prima richiesta è facilmente verificabile una volta definita la griglia, la distribuzione della grandezza adimensionale  $y^+$  a parete è un'informazione che è possibile ottenere solo a valle dell'analisi.

Nell'ottica di costruire un pacchetto interamente *libero* si è optato inizialmente per la costruzione di una griglia di tipo ibrido con *enGrid*. La qualità della griglia, misurata attraverso gli indicatori evidenziati in sezione 2.3, non era soddisfacente in relazione soprattutto alle deformazioni applicate; perciò tale approccio non è stato sviluppato ulteriormente.

Si è quindi scelto il software commerciale *ICEM CFD* per la creazione delle griglie strutturate utilizzate in questa tesi.

#### 2.3.1 Condizioni al contorno

Le equazioni di Navier Stokes necessitano di opportune condizioni al contorno per la buona posizione del problema.

In OpenFoam ci sono tre attributi associati ad un contorno ordinati in maniera gerarchica. L'attributo *base* è quello che descrive il contorno in termini puramente geometrici. Se assegniamo una condizione numerica semplice ad un contorno caratterizzato da un campo variabile viene definito *primitivo*, se la condizione è complessa si dice *derivato*.

Le condizioni al contorno base per i vari contorni della griglia sono presentate in tabella 2.3 e sono rispettivamente:

- *patch* è la tipologia standard di un contorno
- *symmetryPlane* caratteristica di un contorno che è piano di simmetria

- *wall* è tipica di un contorno a cui viene applicato un modello di turbolenza

ID	attributo <i>base</i>
bulb	wall
symmetry	symmetryPlane
inlet, outlet	patch
top, bottom	patch
side	patch

Tabella 2.3: Condizioni al contorno *base*

Per ogni contorno e ogni grandezza fisica di interesse  $p$ ,  $U$ ,  $k$ ,  $\omega$  è stato necessario andare a definire degli attributi di tipo primitivo o derivato.

A titolo d'esempio si riporta in tabella 2.4 l'imposizione delle condizioni al contorno per la velocità.

ID	attributo	tipo	valore
bulb	fixedValue	primitivo	uniform ( 0 0 0 )
symmetry	symmetryPlane	base	-
inlet	fixedValue	primitivo	uniform ( 5 0 0 )
top, bottom, side	slip	derivato	-
outlet	zeroGradient	primitivo	-

Tabella 2.4: Condizioni al contorno per la velocità  $U$ 

L'attributo *fixedValue* quindi forza il campo di velocità al valore imposto, nel nostro caso quindi otterremo una velocità nulla a parete sul bulbo mentre nel contorno di ingresso definiamo le condizioni asintotiche a monte.

Il contorno simmetria è totalmente caratterizzato dall'attributo base *symmetryPlane*. Nel contorno outlet imponiamo semplicemente un gradiente nullo normale del campo di velocità *zeroGradient*.

L'attributo *slip* è un attributo derivato e nel caso di un campo vettoriale la componente normale è imposta nulla (*fixedValue* zero) mentre su quella tangenziale si impone la componente normale del gradiente nulla.

### 2.3.2 Analisi di convergenza spaziale

Si è effettuata un'analisi di convergenza spaziale per valutare la qualità della discretizzazione. Sono state create tre griglie strutturate differenti: *coarse*, *medium* e *fine* su metà problema, sfruttando la simmetria della geometria. Le tre griglie differiscono per numero di elementi ma conservano la discretizzazione a parete, dettata dal corretto utilizzo delle leggi di parete, e le coordinate del contorno esterno.

Si è quindi lanciato il calcolo per le tre griglie e si sono confrontati i risultati in termini di:

- $D_y$  resistenza complessiva longitudinale
- $t$  tempo di calcolo su un processore

tipologia	numero celle	$D_y$ [N]	$t$ [s]
<i>coarse</i>	88956	38.22	277
<i>medium</i>	194108	37.77	619
<i>fine</i>	591598	37.76	2046

Tabella 2.5: Confronto differenti griglie

E' ragionevole credere che ci sia una convergenza verso la soluzione esatta all'infittirsi di una griglia. Il fatto che i risultati ottenuti con la griglia *medium* siano simili a quelli ottenuti con la griglia *fine* ci fa supporre che tale convergenza sia già ottenuta per le due griglie in questione contrariamente a quanto ottenuto con la griglia *coarse*. Allo stesso tempo la griglia *medium* permette di avere una riduzione del costo computazionale piuttosto importante, quantificato dal valore di  $t$ , rispetto alla *fine*.

Scegliamo perciò la griglia *medium* per l'analisi di ottimizzazione, che riportiamo nelle figure 2.4

Nella figura 2.5 è evidenziata la distribuzione di  $y^+$  a parete.

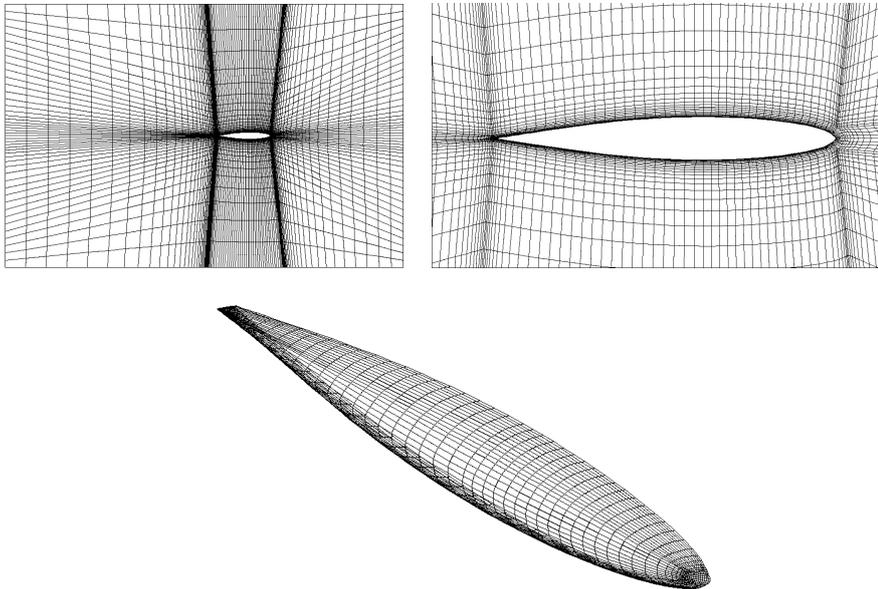


Figura 2.4: Griglia *medium*

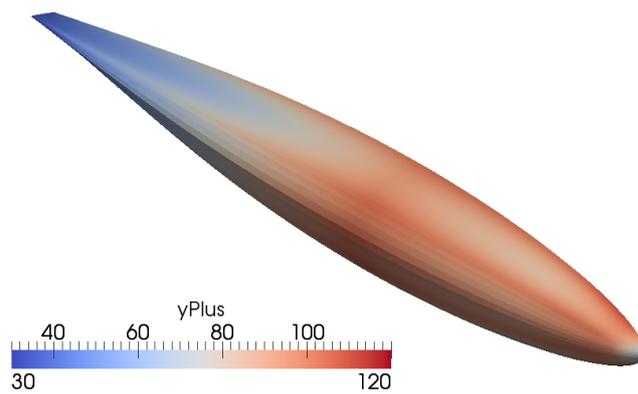


Figura 2.5: Distribuzione di  $y^+$  a parete

## 2.4 Deformazione della griglia

Parte integrante del processo di ottimizzazione è inevitabilmente quella di valutare diverse geometrie per poter stimare una direzione di miglioramento di una certa funzione di merito.

Le strade percorribili sono essenzialmente quella di deformare la geometria e costruire la griglia ex-novo sul nuovo contorno o quella di deformare direttamente la griglia data una deformazione superficiale della geometria, si veda ad esempio [21] e [22].

Questo secondo approccio conduce ad un notevole risparmio computazionale ma implica lo sviluppo di algoritmi che permettano di trasmettere l'informazione di deformazione superficiale a tutti i nodi della griglia senza peggiorare eccessivamente la qualità di quest'ultima.

In principio è stata elaborata una tecnica di deformazione della griglia piuttosto rudimentale che permette deformazioni generiche alle varie sezioni del bulbo, questa tecnica chiamata *sin-2D* viene presentata nella sezione 2.4.1.

Per far fronte a deformazioni nelle tre dimensioni, eventuali rotazioni o traslazioni generiche e garantire una maggiore robustezza alle deformazioni imponibili si è poi utilizzata una tecnica più raffinata basata sul metodo IDW (Inverse Distance Weighting) [19].

### 2.4.1 Algoritmo *sin-2D*

La filosofia alla base di questo algoritmo di deformazione è quella di partizionare il dominio in diverse zone e applicare diverse leggi di deformazione alle varie zone. L'intero codice è stato scritto con *GNU Octave* ed è stato implementato inizialmente come deformatore di griglia test per la messa a punto del processo di ottimizzazione. L'algoritmo lavora in un ambiente esterno ad *OpenFOAM*, non utilizza quindi le librerie interne al codice CFD ma fornisce un nuovo vettore di coordinate dei nodi di tutta la griglia deformata che viene sostituito nelle cartelle di informazioni della griglia da risolvere.<sup>3</sup>

Il codice risale alle informazioni necessarie a partire dai file di testo di descrizione della griglia forniti da *OpenFOAM*, nello specifico necessità di tutte le coordinate nodali dell'intera griglia e di quelle della geometria da deformare.

---

<sup>3</sup>Il flusso è quindi leggermente diverso da quello presentato in 1 in quanto l'informazione che viene passata ad *OpenFOAM* non è lo spostamento dei nodi di superficie del bulbo ma dell'intera griglia.

Con queste informazioni si divide il dominio nelle tre diverse zone a cui poi viene applicata la movimentazione vera e propria.

Le tre zone individuate, riportate nelle figure 2.6, sono:

- zona di deformazione pura (in rosso)
- zona di transizione (in arancione)
- zona indeformata (in giallo)

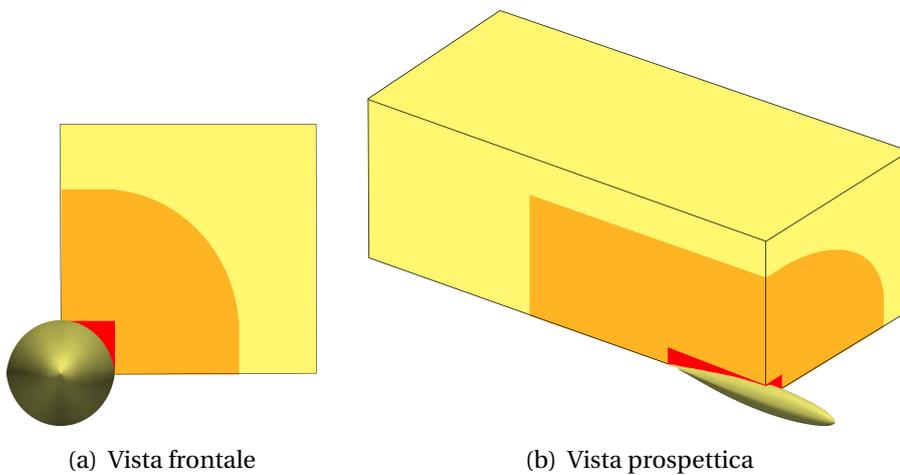


Figura 2.6: Suddivisione dominio per l'algoritmo *sin-2D*

La zona di deformazione pura è il parallelepipedo che circoscrive la geometria da deformare, in questa zona la deformazione viene applicata in maniera completa ad ogni punto della griglia. La zona di transizione viene definita tramite una distanza di buffer<sup>4</sup> ed è quel volume in cui la deformazione viene dissipata fino a diventare nulla al confine con la zona indeformata.

La legge di decadimento della deformazione che da il nome all'algoritmo è di tipo sinusoidale, analiticamente è:

$$f(x) = \frac{1 + \cos(\pi \cdot x)}{2}$$

con  $x = d/R_{buffer}$  distanza adimensionalizzata sulla dimensione della zona di transizione. Tale funzione si è mostrata particolarmente adatta per la presenza dei due flessi a tangenza orizzontale che garantiscono un'adeguata continuità nel passaggio da una zona ad un'altra.

<sup>4</sup>Selezionata dall'utente in funzione della dimensione del dominio di calcolo.

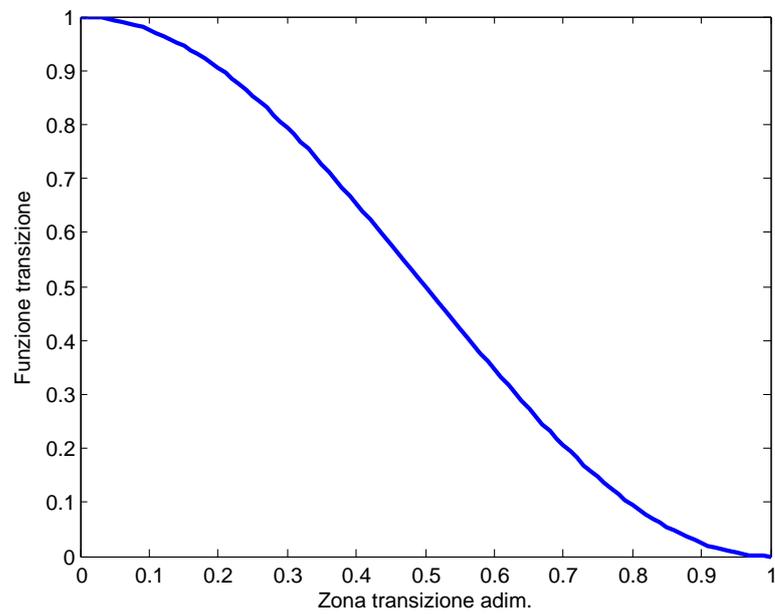


Figura 2.7: Funzione di transizione

In figura 2.8 si rappresenta una sezione di una griglia deformata attraverso il metodo *sin-2D*.

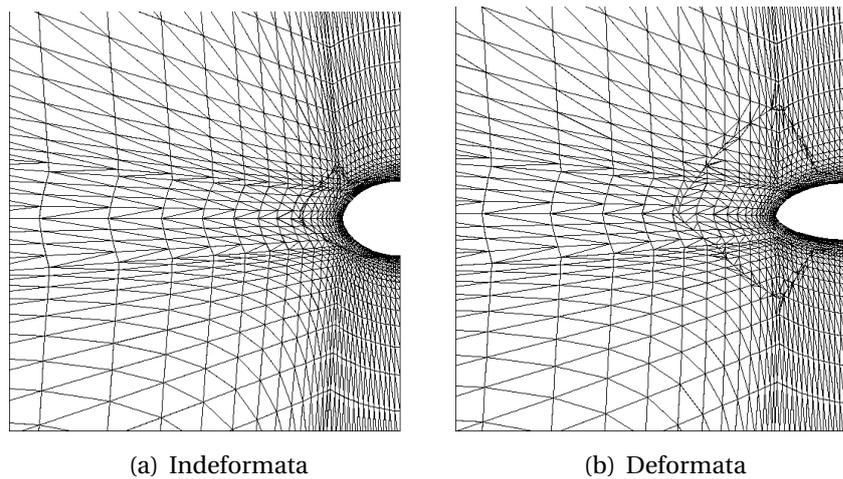


Figura 2.8: Deformazione di una sezione attraverso l'algoritmo *sin-2D*

### 2.4.2 IDW e implementazione

IDW è una tecnica di interpolazione spaziale che permette di creare una mappa noti gli spostamenti di alcuni punti di controllo dati che estende tali informazioni in qualsiasi punto del dominio. Data la semplice implementazione e il ridotto costo computazionale è largamente usata in diverse applicazioni, si veda ad esempio [23] e [24].

Il nome stesso suggerisce la filosofia alla base del metodo: l'influenza di ogni punto su di un altro dipende dall'inverso della distanza. Lo spostamento  $\mathbf{p}$  di un generico punto della griglia  $\mathbf{x}$  è dato dalla sommatoria sugli  $N$  spostamenti noti  $\mathbf{c}_i$  dei punti di controllo  $\mathbf{x}_i$  pesati dall'inverso della distanza, elevata ad un fattore di potenza  $P$ , tra il punto interno e il punto di controllo:

$$\mathbf{p}(\mathbf{x}) = \frac{\sum_{i=1}^N \frac{1}{|\mathbf{x}-\mathbf{x}_i|^P} \mathbf{c}_i}{\sum_{i=1}^N \frac{1}{|\mathbf{x}-\mathbf{x}_i|^P}} \quad (2.9)$$

L>IDW calcola quindi l'importanza di ogni punto di controllo noto sui punti da interpolare attraverso la distanza elevata ad un parametro di potenza. Con una potenza più alta si pone più importanza ai punti vicini e l'interpolazione finale risulterà più dettagliata. Una potenza bassa invece anche ai punti più distanti garantisce una certa influenza, risultando in una superficie più smooth.

<sup>5</sup>

La deformazione della griglia si riduce ad un prodotto tra la matrice  $A$  di dimensioni  $m \times n$ , con  $n$  numero dei punti di controllo  $c_i$  e  $m$  numero dei punti della griglia  $p_j$ , e il vettore  $c$  dei punti di controllo nell'incognita vettore  $p$  degli spostamenti nodali:

$$A\mathbf{c} = \mathbf{p} \quad (2.10)$$

#### Funzione di cut-off

La deformazione predetta dall>IDW viene ulteriormente moltiplicata per una funzione di *cut-off* il cui obiettivo è quello di limitare la sfera di influenza di ogni punto di controllo di spostamento noto.

<sup>5</sup>Una potenza di 2 generalmente è un buon punto di partenza ma nel nostro caso si sono utilizzate potenze più elevate per soddisfare i criteri di qualità della griglia, le varie deformazioni sono state condotte con una potenza pari a 7.

I due raggi  $R_{min}, R_{max}$ <sup>6</sup> evidenziano la zona di transizione tra le due zone in cui viene applicata interamente la deformazione calcolata dall'IDW e quella in cui viene posta nulla.  $R_{max}$  è quindi quel raggio che permette di determinare quali sono i punti che hanno influenza su un determinato punto da interpolare, mentre il contributo di tutti gli altri punti viene trascurato.

La funzione cubica di cut-off rappresentata in figura 2.9 è data da:

$$b(x) = \begin{cases} 1 & \text{se } t < 0 \\ 1 - t^2(3 - 2t) & \text{se } 0 < t < 1 \\ 0 & \text{se } t > 1 \end{cases}$$

con  $t = \frac{\| |x - x_c| \| - R_{min}}{R_{max} - R_{min}}$ .

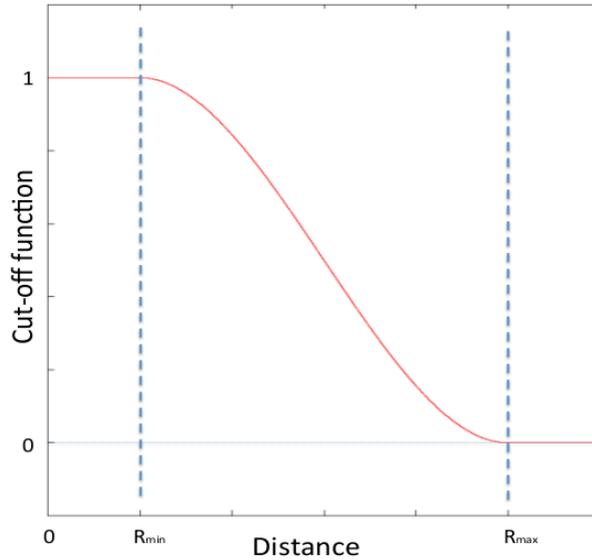


Figura 2.9: Raggi di influenza

In riferimento alla matrice evidenziata nel prodotto (2.10), la generica componente matriciale risulta infine:

$$A_{j,i} = \frac{\frac{b(|x_j - x_i|)}{|x_j - x_i|^P}}{\sum_{i=1}^N \frac{1}{|x_j - x_i|^P}} \quad (2.11)$$

dove  $b(\cdot)$  è la funzione di cut-off appena introdotta.

<sup>6</sup>In altri testi noti come  $R_{blend}, R_{cutOff}$ .

Questo approccio permette di trasformare la base IDW in una base a supporto compatto, la matrice risultante è quindi *sparsa* e la richiesta di memoria e il costo computazionale associato al prodotto risultano inferiori<sup>7</sup>.

Nel caso particolare di una geometria che si vuole deformare all'interno di un dominio che non deve essere deformato, come nel caso del bulbo, sarà sufficiente scegliere un  $R_{max}$  di poco inferiore alla distanza del contorno dalla geometria per mantenere fisso il contorno esterno. Questo approccio permette di non dover selezionare come punti di controllo a spostamento imposto nullo i punti del contorno limitando la dimensione del prodotto (2.10).

### Piano di simmetria

L'approccio dovuto all'introduzione della funzione di cut-off presenta il vantaggio di poter mantenere fisse alcune patch della griglia di calcolo semplicemente selezionando in maniera opportuna  $R_{max}$ . Nel caso in cui i punti di un contorno da mantenere fisso fossero relativamente vicini alla geometria deformabile non sarebbe possibile questo approccio per la necessità di scegliere un  $R_{max}$  troppo piccolo; tale caso si presenta, ad esempio sul piano di simmetria in un problema simmetrico.

Poichè una parte delle nostre analisi è stata condotta su metà dominio di calcolo, si è rivelato necessario implementare una correzione all'algoritmo che mantenesse il piano di simmetria effettivamente planare senza considerare come punti di controllo quelli appartenenti al piano di simmetria ai quali si sarebbe dovuto imporre spostamento normale nullo. In figura 2.10 si mostra l'effetto di una generica deformazione del contorno del bulbo che coinvolge la coordinata a cavallo dal piano di simmetria e di come si vada a perdere la planarità.

Per presentare la procedura di correzione si espande formalmente il prodotto (2.10) nel caso di un problema simmetrico andando a evidenziare i sottoblocchi:

$$\begin{bmatrix} A_{ll} & A_{lr} \\ A_{rl} & A_{rr} \end{bmatrix} \begin{Bmatrix} \mathbf{c}_l \\ \mathbf{c}_r \end{Bmatrix} = \begin{Bmatrix} \mathbf{p}_l \\ \mathbf{p}_r \end{Bmatrix}$$

I pedici  $l$  e  $r$  si riferiscono all'elemento caratteristico di un lato e dall'altro del piano di simmetria (*left / right*). Sono messi in evidenza i punti control-

<sup>7</sup>Si ricorda che in prima approssimazione il costo computazionale associato ad un prodotto matrice-vettore per una matrice quadrata di dimensione  $n$  è dell'ordine di  $n^2$ .

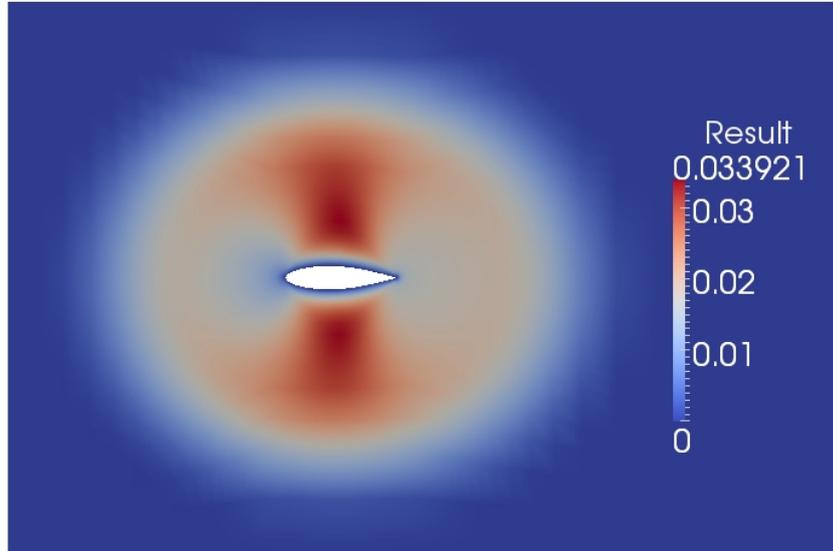


Figura 2.10: Effetto di una generica deformazione - traslazione perpendicolare al piano di simmetria

lo  $\mathbf{c}_l, \mathbf{c}_r$  i cui spostamenti sono noti e i punti della griglia  $\mathbf{p}_l, \mathbf{p}_r$  da calcolare. Le sottomatrici  $A_{ll}, A_{lr}, A_{rl}, A_{rr}$  sono calcolate utilizzando sempre la formula (2.11) mettendo in evidenza i coefficienti di influenza del punto di controllo  $\mathbf{c}_i$  sul punto della griglia  $\mathbf{p}_j$ .

Siamo interessati naturalmente solo al calcolo degli spostamenti dei nodi della griglia da un solo lato, senza perdere di generalità scegliamo quello sinistro  $\mathbf{p}_l$ .

Nel caso di un problema simmetrico i punti di controllo soddisfano la relazione  $\mathbf{c}_r = \mathbf{c}_l$  per tutte le coordinate a meno di quella normale al piano di simmetria dove  $\mathbf{c}_r = -\mathbf{c}_l$ . Definiamo quindi il vettore fittizio  $\bar{\mathbf{c}}$  globale che rispetta tali relazioni geometriche.

Sempre nel caso di problema simmetrico vale la relazione tra le sottomatrici  $A_{ll} = A_{rr}$ .

Abbiamo implementato nella libreria il calcolo della sottomatrice  $A_{lr}$ , costruita a partire dalla conoscenza dei punti di controllo  $\mathbf{c}_r$ .

Il problema del calcolo degli spostamenti della griglia  $\mathbf{p}_l$  tenendo conto della correzione dovuta alla simmetria si riduce al prodotto seguente:

$$\begin{bmatrix} A_{ll} & A_{lr} \end{bmatrix} \bar{\mathbf{c}} = \mathbf{p}_l$$

Con questa procedura è quindi possibile applicare il metodo IDW ad un generico dominio simmetrico rispettando la planarità del piano di simmetria e garantendo un risparmio di risorse computazionali rispetto al problema completo; il prodotto evidenziato infatti coinvolge una matrice con la metà delle righe del problema intero di partenza.

In figura 2.11 viene presentato un taglio della griglia per una generica deformazione che coinvolge la coordinata a cavallo del piano di simmetria. Si mostra la griglia iniziale e quella finale, prima e dopo la correzione.

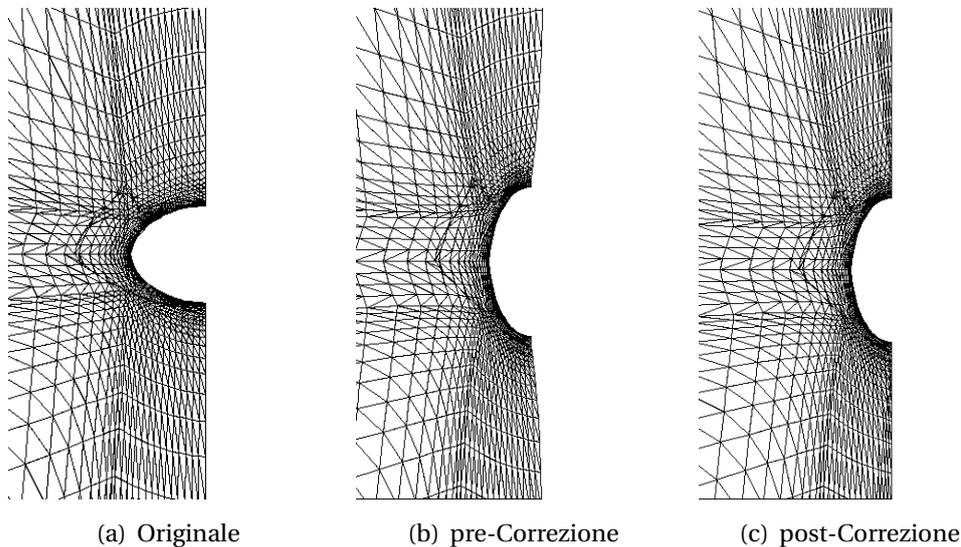


Figura 2.11: Correzione piano di simmetria

### 2.4.3 Confronto degli algoritmi di deformazione

In questa sezione si vuole presentare un confronto fra i due metodi di deformazione utilizzati in riferimento agli indicatori di qualità evidenziati in sezione 2.3. Si riportano i valori di tali indicatori in tabella 2.6, riferiti ad una deformazione estrema applicata alla geometria di parametri  $P1=0.3$ ,  $P2=0.3^8$ . Appare evidente come l'algoritmo IDW garantisca una qualità di griglia più elevata, visualizzabile attraverso valori degli indicatori di qualità mediamente più bassi e un minor numero di celle fallite.

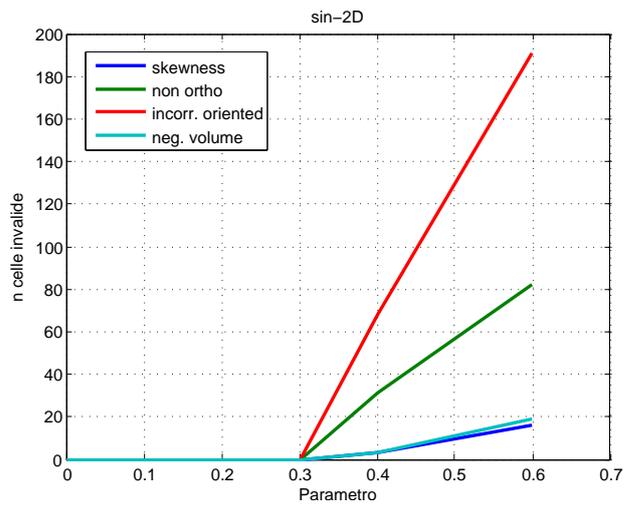
<sup>8</sup>Il parametro P3 non è mai stato implementato nell'algoritmo *sin-2D*

indice di qualità	<i>sin-2D</i>	IDW
max allungamento (n° celle fallite)	62.481 (0)	53.839 (0)
max distorsione (n° celle fallite)	3.392 (0)	1.457 (0)
max non-ortogonalità (n° celle fallite)	87.325 (212)	71.742 (16)

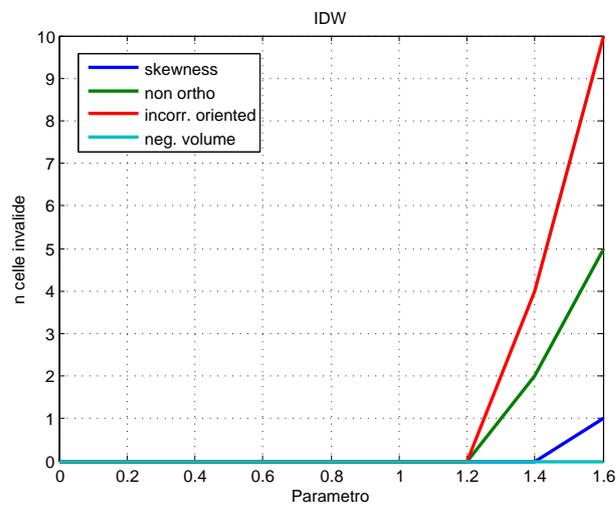
Tabella 2.6: Confronto differenti metodi di deformazione - indicatori di qualità

Si è anche evidenziato come il metodo IDW permetta un'escursione dei parametri più ampia rispetto all'algoritmo *sin 2-D*. Nel grafico 2.12 viene riportato il numero di celle invalide rilevate all'aumentare della deformazione imposta; nello specifico si è applicata una deformazione di parametri  $P1 = x$  e  $P2 = x$  con  $x$  via via crescente.

Da un punto di vista pratico inoltre l'algoritmo IDW necessita in ingresso degli spostamenti di un set di punti di controllo, mentre *sin-2D* ha direttamente implementate le leggi di deformazioni del contorno. Questo significa che l'algoritmo IDW oltre ad essere più robusto è anche più generale; è stato per esempio possibile estendere le deformazioni possibili attraverso l'introduzione del parametro  $P3$  e prevedere la rotazione del bulbo nel dominio. Per questi motivi si è scelto di effettuare lo studio esclusivamente con la tecnica IDW.



(a) *sin 2-D*



(b) IDW

Figura 2.12: Confronto differenti metodi di deformazione - validità della griglia



# Capitolo 3

## Tecniche di ottimizzazione

In questo capitolo si formula inizialmente un generico problema di ottimizzazione.

Quindi si mostra il metodo del gradiente coniugato lineare e non, con la presentazione specifica e l'analisi dell'algoritmo utilizzato applicato ad un problema semplificato.

Infine vengono presentate le particolarità dei metodi genetici, fino a presentare l'algoritmo multiobiettivo MOGA.

### 3.1 Introduzione

L'ottimizzazione è un importante strumento nel campo delle scelte industriali o nell'analisi di sistemi fisici.

Per utilizzarlo al meglio è innanzitutto necessario identificare degli *obiettivi*, cioè una misura quantitativa della performance del sistema studiato. L'obiettivo dipenderà da alcune caratteristiche del sistema chiamate *gradi di libertà* o *parametri*.

Uno studio di ottimizzazione consiste quindi nel trovare quei valori dei gradi di libertà che ottimizzino l'obiettivo. Spesso però le variabili sono *vincolate* in qualche modo tra di loro per mantenere la fisicità del sistema o semplicemente per rispettare alcune richieste dell'utente.

Il processo di individuazione di obiettivi, gradi di libertà e vincoli di un problema si chiama *modellazione*.

Solo a valle di un corretto studio del modello si può procedere ad un'analisi di ottimizzazione scegliendo l'opportuno algoritmo risolutivo. La scelta cor-

retta infatti non solo può comportare tempi di soluzione diversi ma potrebbe condurre anche a soluzioni diverse.

### 3.1.1 Formulazione matematica

Matematicamente parlando ottimizzare significa minimizzare o massimizzare una funzione soggetta a vincoli nei suoi gradi di libertà.

Il problema di ottimizzazione si può quindi scrivere come:

$$\min_{x \in \mathcal{R}^n} f(x) \text{ soggetta a } \begin{cases} g_L \leq g(x) \leq g_U \\ h(x) = h_T \\ a_L \leq A_i x \leq a_U \\ A_e x = a_T \\ x_L \leq x \leq x_U \end{cases} \quad (3.1)$$

L'obiettivo dell'ottimizzazione è quindi minimizzare la funzione obiettivo,  $f(x)$ , soddisfacendo allo stesso tempo i vincoli. I vincoli possono essere divisi in vincoli lineari o non lineari e in vincoli d'uguaglianza o di disequaglianza.

I vincoli non lineari di disequaglianza,  $g(x)$ , possono avere un limite superiore e uno inferiore, rispettivamente  $g_L$  e  $g_U$ . I vincoli non lineari d'uguaglianza,  $h(x)$ , hanno il valore obiettivo  $h_T$ . I vincoli lineari di disequaglianza creano un sistema lineare  $A_i x$  dove  $A_i$  è la matrice dei coefficienti per il sistema lineare. Questi vincoli hanno anch'essi un limite inferiore e uno superiore  $a_L$  e  $a_U$  rispettivamente. I vincoli lineari d'uguaglianza creano un sistema lineare  $A_i x$  dove  $A_i$  è la matrice dei coefficienti per il sistema lineare e  $a_T$  i valori obiettivo.

#### Tipologia del problema

La distinzione più importante che si può fare su un problema di ottimizzazione è la presenza o meno di vincoli sui gradi di libertà.

Una *ottimizzazione non vincolata* può essere insita nella natura del problema o altre volte ottenuta. Se ci sono vincoli naturali sui gradi di libertà è spesso utile non considerarli se si può avanzare l'ipotesi che non abbiano impatto sulla soluzione. Si può anche andare a riformulare il problema, dove i vincoli sono sostituiti da una penalizzazione della funzione obiettivo avvicinandosi a tali valori nei gradi di libertà, riducendo così le probabilità di una violazione del vincolo. Questi artifici vengono messi in atto perchè è sempre preferibile utilizzare algoritmi risolutivi tipici di problemi non vincolati.

L' *ottimizzazione vincolata* invece evidenzia dei vincoli espliciti sul modello. Possono essere vincoli semplici sull'insieme di appartenenza dei gradi di libertà, vincoli lineari fino a relazioni non lineari tra i gradi di libertà.

Inoltre se i vincoli e la funzione stessa sono lineari possiamo definire il problema di *ottimizzazione lineare*, altrimenti diventa un problema di *ottimizzazione non lineare*.

### **Obiettivo**

Il minimo di una funzione può essere *globale* o *locale*.

Gli algoritmi tipicamente più veloci riescono a trovare solo minimi locali, un punto cioè in cui la funzione è effettivamente minore di tutti i punti possibili nelle vicinanze ma che non necessariamente coincide con la soluzione migliore.

Il minimo globale da un punto di vista computazionale è molto più oneroso da calcolare e spesso se non si utilizza l'algoritmo opportuno non può essere localizzato.

### **Metodo di ricerca**

Con metodo di ricerca intendiamo l'approccio utilizzato nell'algoritmo di ottimizzazione per individuare una nuova stima di variabili che caratterizza una funzione obiettivo minore dell'iterata precedente.

I metodi di ricerca si possono sostanzialmente dividere in *algoritmi basati sul gradiente* o *non basati sul gradiente*.

Negli algoritmi basati sul gradiente una ricostruzione del gradiente della funzione obiettivo viene utilizzata per trovare la direzione di miglioramento. Sono i metodi che presentano le migliori prestazioni per la ricerca di minimi locali.

Tuttavia non è sempre possibile utilizzarli, lo stesso gradiente potrebbe non essere calcolabile o troppo costoso, in questi casi vengono utilizzati metodi non basati sul gradiente. Esistono numerosi metodi di questo tipo, come i *pattern search method* per minimi locali o gli algoritmi genetici per la ricerca di minimi globali.

### **Algoritmi utilizzati**

Nel caso specifico di studio del bulbo esistono diverse tipologie di analisi che possono essere condotte.

L'obiettivo di questa ottimizzazione è il miglioramento delle prestazioni dell'imbarcazione ottenibile scegliendo diverse funzioni obiettivo da minimizzare. Riprendendo la scelta delle analisi presentate in sezione 1.4.1 come prima analisi si è scelto di minimizzare la resistenza del bulbo a zero gradi di incidenza soggetto al vincolo di rispetto di un momento raddrizzante minimo soddisfatto direttamente dalla geometria deformata.

La seconda analisi è analoga alla precedente con la differenza che la funzione da minimizzare non è più la resistenza a zero gradi di incidenza ma diventa una resistenza mediata tra una configurazione tipica di bolina e di poppa secondo un peso scelto dall'utente. Queste due prime analisi sono un esempio di un problema vincolato che viene risolto con algoritmi tipici di problemi non vincolati adottando artifici ad hoc, nello specifico la generazione automatica di geometrie che rispettano il vincolo imposto.

La terza analisi invece propone come miglioramento di prestazioni dell'imbarcazione un'ottimizzazione della geometria rispetto a due funzioni obiettivo, una minimizzazione di resistenza insieme alla minimizzazione del centro di gravità verticale (VCG) del bulbo. Tutte le soluzioni rispettano in questo caso un vincolo sul volume.

Nelle prime due analisi verrà utilizzato l'algoritmo di Fletcher-Reeves implementato in *DAKOTA*, tale algoritmo appartiene alla famiglia dei metodi del gradiente coniugato per risolvere problemi non lineari non vincolati.

L'ultima analisi invece deve utilizzare un algoritmo tipico di problemi multi-obiettivo, in accordo con [25] scegliamo l'algoritmo genetico MOGA.

## **3.2 Metodo del gradiente coniugato**

L'interesse nel metodo del gradiente coniugato è duplice. E' uno dei metodi più adatti per risolvere grandi sistemi di equazioni lineari e può essere adattato per risolvere problemi di ottimizzazione non lineare.

Il metodo del gradiente coniugato lineare fu proposto da Hestenes and Stiefel nel 1950 ed è un metodo iterativo risolvete sistemi lineari per matrici definite positive. Le prestazioni del metodo sono legate alla distribuzione de-

gli autovalori delle matrici dei coefficienti, ecco perchè preconditionando tali matrici è possibile aumentarne le prestazioni.

Il primo metodo del gradiente coniugato non lineare fu introdotto da Fletcher e Reeves negli anni '60. E' una delle prime tecniche per risolvere problemi di ottimizzazione non lineare di grande scala. Negli anni diverse varianti allo schema originale sono state introdotte nella scelta di alcuni parametri del ciclo di ricerca.

Il tratto chiave di questi algoritmi è che non richiedono un elevato spazio di memoria e sono più veloci del metodo di massima discesa. Deriviamo i tratti salienti del gradiente coniugato lineare, quindi lo estendiamo al caso non lineare e ne mostriamo il comportamento.

### 3.2.1 Lineare

Come detto il metodo del gradiente coniugato è un metodo per risolvere sistemi lineari

$$Ax = b, \quad (3.2)$$

dove la matrice  $A$  è una matrice simmetrica e definita positiva. Tale problema può essere egualmente espresso come un problema di minimizzazione nella forma:

$$\phi(x) = \frac{1}{2}x^T Ax - b^T x, \quad (3.3)$$

Entrambe le formulazioni hanno la stessa soluzione e per questo possiamo considerare il metodo del gradiente come un algoritmo per risolvere sistemi di equazioni lineari o come una tecnica per minimizzare funzioni quadratiche convesse.

Si mette in evidenza inoltre come il gradiente  $\phi$  equivale al residuo del sistema lineare.

$$\nabla\phi(x) = Ax - b = r(x). \quad (3.4)$$

Una delle caratteristiche che dona il nome al metodo è l'abilità di generare, in maniera economica, un insieme di vettori coniugati rispetto ad una matrice SDP (Simmetrica Definita Positiva), ovvero:

$$p_j^T A p_j = 0, \text{ per tutti gli } i \neq j. \quad (3.5)$$

Una proprietà del metodo quindi è quella di costruire una direzione coniugata senza avere la necessità di immagazzinare tutti i vettori utilizzati fino a quel momento ma semplicemente avendo a disposizione le informazioni note al passo precedente.

Ogni direzione  $p_k$  è scelta essere una combinazione lineare tra la direzione di massima discesa  $-\nabla\phi(x_k)$  (che è equivalente al residuo  $-r_k$ ) e la direzione precedente.

Quindi la direzione k-esima è:

$$p_k = -r_k + \beta_k p_{k-1} \quad (3.6)$$

dove  $\beta_k$  è da determinare con la necessità di rispettare la condizione su  $p_{k-1}$  e  $p_k$  coniugati rispetto ad A.

$$\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k} \quad (3.7)$$

La lunghezza del passo è invece:

$$\alpha_k = \frac{r_k^T r_k}{p_k^T A p_k} \quad (3.8)$$

necessario per calcolare la stima successiva, banalmente:

$$x_{k+1} = x_k + \alpha_k p_k \quad (3.9)$$

Riassumiamo qui di seguito i vari passi dell'algorithm.

### Algoritmo CG

Dato  $x_0$ ;

$r_0 \leftarrow Ax_0 - b, p_0 \leftarrow -r_0, k \leftarrow 0$ ;

**while**  $r_k \neq 0$

$$\begin{aligned}
\alpha_{k+1} &\leftarrow \frac{r_k^T r_k}{p_k^T A p_k} \\
x_{k+1} &\leftarrow x_k + \alpha_k p_k \\
r_{k+1} &\leftarrow r_k + \alpha_k A p_k \\
\beta_{k+1} &\leftarrow \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k} \\
p_{k+1} &\leftarrow -\nabla f_{k+1} + \beta_{k+1}^{FR} p_k; \\
k &\leftarrow k + 1
\end{aligned}$$

**end(while)**

### 3.2.2 Non lineare: algoritmo di Fletcher-Reeves

Il metodo del gradiente coniugato può essere visto come una tecnica di minimizzazione della generica funzione quadratica convessa  $\Phi$ . Può essere esteso ad una minimizzazione di generica funzione convessa, fino ad una generica funzione non lineare  $f(x)$ . Il metodo implementato in *DAKOTA* è l'algoritmo di Fletcher-Reeves, che andremo a presentare.

E' sufficiente fare due modifiche semplici al metodo del gradiente coniugato lineare per estenderlo al caso non lineare. Innanzitutto al posto di scegliere come lunghezza del passo  $\alpha_k$  come riportato nell'equazione (3.8) per il gradiente coniugato lineare dobbiamo effettuare una *line search* che possa identificare un'approssimazione del minimo della funzione non lineare  $f$  lungo  $p_k$ . Questo significa che rispetto al CG lineare la scelta dello step  $\alpha_k$  non sarà ottenuta in una iterazione ma necessita di alcune iterazioni per selezionare un buon compromesso nella lunghezza del passo.

Il residuo  $r$ , che è semplicemente il gradiente di  $\phi$  nell'algoritmo del metodo lineare, viene sostituito con il gradiente dell'obiettivo non lineare  $f$ .

#### Algoritmo FR - CG

Dato  $x_0$ ;  
 Valuta  $f_0 = f(x_0), \nabla f_0 = \nabla f(x_0)$ ;  
 $p_0 = -\nabla f_0, k \leftarrow 0$ ;

**while**  $\nabla f_k \neq 0$

Calcola  $\alpha_k$  e imponi  $x_{k+1} = x_k + \alpha_k p_k$ ;

Valuta  $\nabla f_{k+1}$ ;

$$\beta_{k+1}^{FR} \leftarrow \frac{\nabla f_{k+1}^T \nabla f_{k+1}}{\nabla f_k^T \nabla f_k}$$

$$p_{k+1} \leftarrow -\nabla f_{k+1} + \beta_{k+1}^{FR} p_k;$$

$$k \leftarrow k + 1$$

**end(while)**

La scelta del parametro di line search  $\alpha_k$  può essere esatto o inesatto. Nel caso inesatto, ovvero quello implementato in *DAKOTA*, si dice che  $\alpha_k$  minimizza in maniera *debole* la funzione lungo la direzione  $p_k$ .

Una modifica che è spesso implementata è la capacità di *restart* ogni  $n$  passi imponendo  $\beta_k$ , equivalente a prendere la massima direzione di discesa. Questa modifica serve per aggiornare periodicamente l'algoritmo, cancellando quelle informazioni che potrebbero non essere benefiche.

### 3.2.3 Esempio di funzionamento

In questo capitolo mostriamo e commentiamo il funzionamento dell'algoritmo di Fletcher-Reeves utilizzato da *DAKOTA* per l'ottimizzazione di un problema semplificato caratterizzato da:

- funzione obiettivo: resistenza *Drag*
- vincolo: geometrie aventi lo stesso volume
- parametri liberi:  $P1$  e  $P3$ , rispettivamente strizione e allungamento.

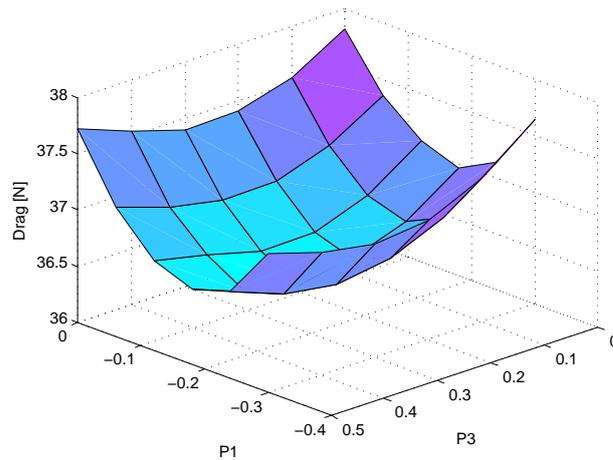
Nelle figure 3.1 si presenta in (a) uno studio nei due parametri per tracciare la superficie rappresentante la resistenza. Quindi nel grafico (b) riportiamo le curve di livello su cui sono proiettati i vari step dell'ottimizzazione.

Un ciclo base nella stima della soluzione  $x_{k+1}$  a partire da  $x_k$  è il seguente:

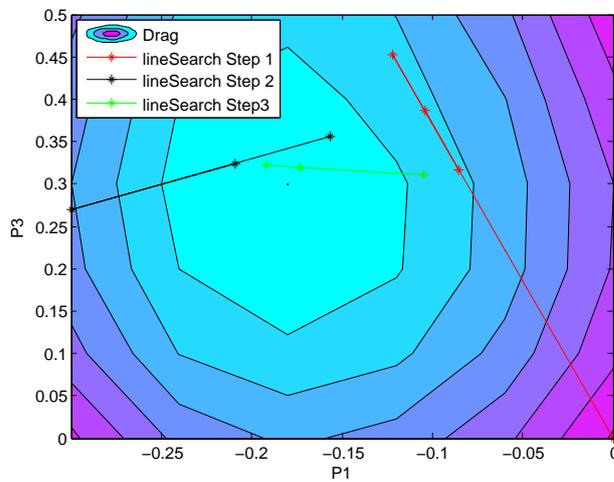
1. si calcola il gradiente in  $x_k$  per differenze finite nelle due direzioni, se la prima iterata nel punto caratterizzato dai parametri  $P1 = 0, P3 = 0$
2. si calcola una direzione di miglioramento  $p_k$  e attraverso una *line search* si minimizza debolmente la soluzione su quella linea (necessarie 3-4 iterate)

3. si ottiene una nuova stima  $x_{k+1}$
4. si verifica la tolleranza, se non viene rispettata si rientra nel ciclo

Si ricorda che ogni geometria valutata nei vari step di ottimizzazione rispetta intrinsecamente il vincolo di volume attraverso l'algoritmo presentato in sezione 1.4.



(a)



(b)

Figura 3.1: Funzionamento FR - CG: grafico 3D (a) e curve di livello (b)

### 3.3 Metodi genetici

Gli algoritmi genetici sono una famiglia di tecniche di ottimizzazione di tipo globale a differenza dei metodi del gradiente che sono locali. Si ispirano alla teoria evuzionistica di Charles Darwin la quale spiega come gli individui possano sopravvivere e svilupparsi adattandosi progressivamente all'ambiente che li circonda.

La caratteristica alla base di questa teoria consiste nella capacità degli individui migliori di sopravvivere in un ambiente avverso e di essere quindi adatti a riprodursi combinando tra loro le informazioni genetiche.

Tra una generazione e l'altra la popolazione evolve evidenziando differenze dovute alla combinazione dei genitori e ad un mutamento casuale. Questi cambiamenti possono essere più o meno adatti all'ambiente circostante, mutevole di generazione in generazione e sempre più selettivo.

Questo meccanismo evolutivo prende il nome di *selezione naturale* ed è caratterizzato da alcuni tratti salienti:

- ogni individuo ha una codifica genetica (genotipo) che evidenzia delle caratteristiche fisiche (fenotipo)
- l'evoluzione agisce sui cromosomi dell'individuo (genotipo)
- la selezione naturale favorisce la riproduzione degli individui più forti: si riprodurrà quell'individuo che trasporta il genotipo che ha prodotto il fenotipo migliore
- la riproduzione è alla base dell'evoluzione: un figlio presenterà un genotipo dovuto alla combinazione dei genotipi dei genitori più eventuali mutazioni casuali

Gli algoritmi genetici operano su una popolazione di cromosomi artificiali che vengono fatti riprodurre selettivamente in base alle prestazioni di un fenotipo a cui danno origine. Di generazione in generazione la popolazione sarà caratterizzata dal proprio genotipo o vettore di parametri che convergerà verso il set o i set che realizzano il minimo del fenotipo associato.

Dal punto di vista numerico ci sono tre parametri che calibrano l'evoluzione della popolazione:

- riproduzione selettiva

- crossover
- mutazione

La riproduzione selettiva consiste nella maggior capacità degli individui migliori di trasmettere il patrimonio genetico. A livello riproduttivo tipicamente è necessario definire il numero di genitori e di prole che realizzano ogni incrocio.

I cross-over e la mutazione invece sono due parametri che gestiscono il peso relativo del genotipo dei genitori e del caso sulla scrittura del genotipo del figlio.

Risulta evidente come algoritmi di questo tipo siano particolarmente adatti alla risoluzione di problemi con diversi minimi locali o nei quali il calcolo del gradiente non sia possibile.

### 3.3.1 Algoritmo Genetico Multi Obiettivo (MOGA)

Una ottimizzazione multi obiettivo significa che ce ne sono due o più funzioni obiettivo che si vogliono minimizzare allo stesso tempo. Spesso questi sono obiettivi in conflitto tra di loro, come possono essere costo e performance.

La risposta di un'ottimizzazione di questo tipo non è un singolo punto ma un set di punti chiamato *fronte di Pareto*. Ogni punto appartenente a questa curva soddisfa la condizione di ottimo di Pareto: localmente non esiste un altro vettore di parametri che possa migliorare un obiettivo senza peggiorarne simultaneamente un altro.

Una soluzione rispetto alla quale è possibile fare dei miglioramenti di una funzione obiettivo senza peggiorare le altre non rispetta la condizione di ottimo di Pareto. Tale soluzione è detta essere *dominata* mentre i punti lungo il fronte di Pareto sono detti *non-dominati*.

Spesso delle analisi multiobiettivo vengono ridotte ad analisi singolo obiettivo fornendo un peso a ciascuna funzione da minimizzare, potendo quindi risolvere il problema con le diverse tecniche a disposizione. Il limite di questo approccio consiste nel fatto che si ottiene una sola soluzione che rappresenta la miglior risposta a partire dai pesi selezionati.

Per avere quindi una buona conoscenza del problema sarebbe necessario lanciare numerose ottimizzazioni, procedimento che potrebbe risultare costoso dal punto di vista computazionale. Ecco perchè le ottimizzazioni multiobiettivo hanno comunque trovato la loro nicchia di utilizzo.

L'idea alla base è che la popolazione evolve secondo la teoria genetica, le soluzioni che non sono dominate sono quelle che hanno il fenotipo adatto alla sopravvivenza.

Nonostante la popolazione sia teoricamente capace di trovare i vari minimi locali una popolazione di partenza limitata tende a convergere verso un singolo ottimo rendendo il fronte di Pareto non uniformemente rappresentato. Questo fenomeno noto con il nome di *deriva genetica* è un problema che i vari algoritmi genetici multi obiettivo hanno dovuto risolvere.

Per far fronte a tale problema si corregge l'evoluzione definendo un indicatore di densità che rappresenta il numero di individui nelle vicinanze. Nel momento della riproduzione individui caratterizzati da un genotipo particolarmente isolato hanno maggiori probabilità di riprodursi degli altri, favorendo perciò la diversificazione e limitando le alte densità specifiche di popolazione.

Un'altra caratteristica del metodo Moga è l'*elitarismo*, ad ogni generazione gli individui che hanno presentato il miglior fenotipo fino a quel momento sono congelati fino alla generazione successiva.

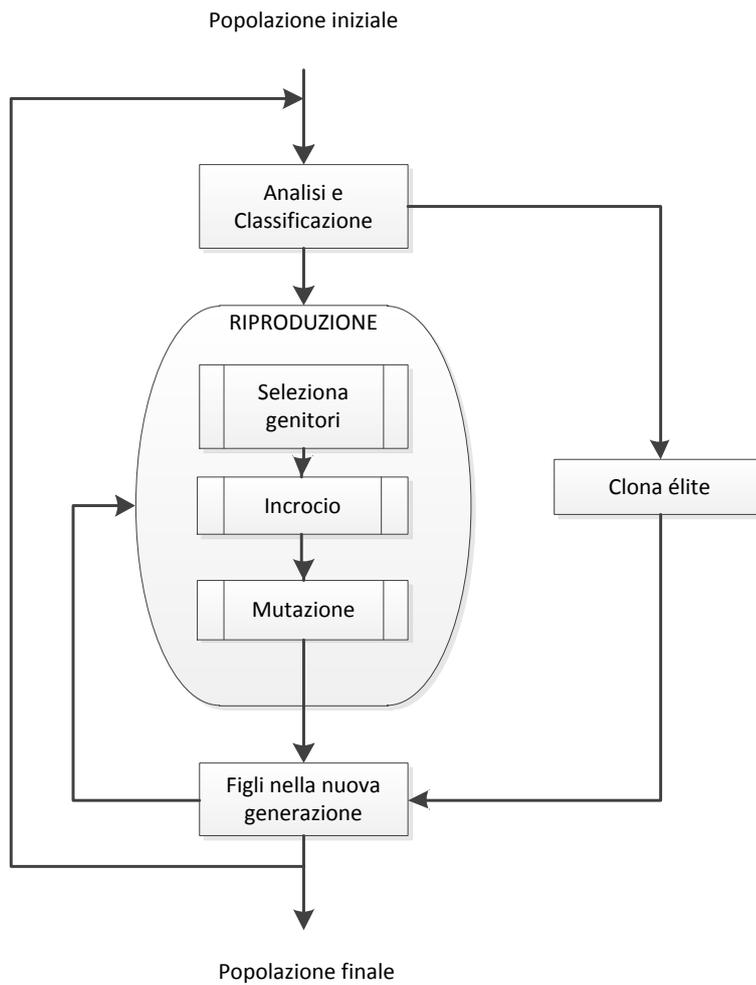


Figura 3.2: Funzionamento di un algoritmo genetico



# Capitolo 4

## Analisi dei risultati

In questo capitolo si mostrano i risultati delle varie analisi di ottimizzazioni condotte. In riferimento alla tabella 1.1 della sezione 1.4.1 si presentano:

- la geometria  $Geo_1$  specifica per l'andatura di poppa in sezione 4.1
- la geometria  $Geo_2$  per un progetto allround in sezione 4.2
- una famiglia di geometrie  $Geo_3$  per lo studio multiobiettivo in sezione 4.3

tutte le geometrie ottimizzate verranno confrontate con la geometria di partenza  $Geo_0$ .

### 4.1 Poppa

Il risultato dell'analisi è la geometria di bulbo ottimizzata  $Geo_1$ , presentata in figura 4.1 e definita dal set di parametri:

- $P1 = -0.1778$ ;  $P2 = 0.0907$ ;  $P3 = 0.3417$ ;

geometria	Volume	Momento
$Geo_0$	0.085002	0.193188
$Geo_1$	0.085091	0.193184

Tabella 4.1: Caratteristiche geometriche diverse geometrie

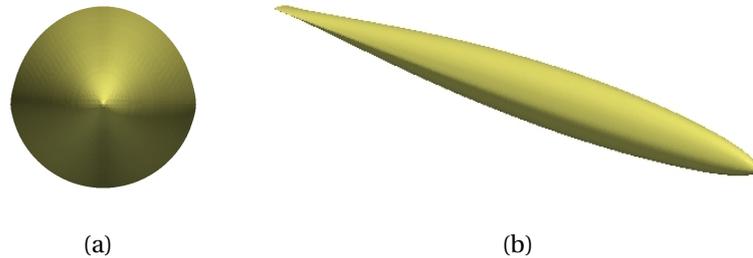


Figura 4.1: Geometria del bulbo ottimizzato  $Geo_1$ : vista frontale (a) e prospettica (b)

La risposta del sistema più evidente è la necessità di rendere più circolare la sezione attraverso il parametro P1 e allungarla attraverso il parametro P3. Il parametro P2 viene utilizzato in maniera meno invasiva ma coerentemente con il vincolo si ha la convergenza verso una geometria con sezioni asimmetriche. Il volume di  $Geo_1$  è superiore a  $Geo_0$  mentre il momento raddrizzante è inferiore. Il vincolo sul momento risulta contenuto nella tolleranza imposta a  $toll = 10^{-4}$ .

Nel grafico 4.2 sono riportate le evoluzioni della funzione obiettivo e del vincolo alle diverse iterazioni. Interessante apprezzare come nel problema non lineare da risolvere siano necessarie diverse iterate per ogni *line search*, in accordo con quanto riportato nella simulazione in sezione 3.2.3

In figura 4.3 si riportano invece i valori assunti dai parametri alle diverse iterazioni di decision Function, ovvero il set di parametri scelto per ogni *line search*.

#### 4.1.1 Termini di resistenza

In tabella 4.2 si mostra la diminuzione di resistenza complessiva tra la geometria indeformata e quella ottimizzata per la poppa. Si ricorda che questa prima analisi è stata condotta su metà dominio di calcolo, i risultati riportati in tabella sono quindi raddoppiati per essere confrontabili con le analisi successive presentate nei capitoli 4.2 e 4.3. In particolare vengono mostrati i diversi contributi di resistenza viscosa e di pressione, rispettivamente  $D_{visc}$  e  $D_{press}$ .

La dimensione della scia è legata all'entità del termine di resistenza di pressione. Nelle figure 4.4 e 4.5 si mostra come la scia sviluppata dalla geometria ottimizzata  $Geo_1$  sia sensibilmente inferiore a quella di partenza  $Geo_0$  e questo spiega l'abbattimento di resistenza di pressione riportato in tabella 4.2;

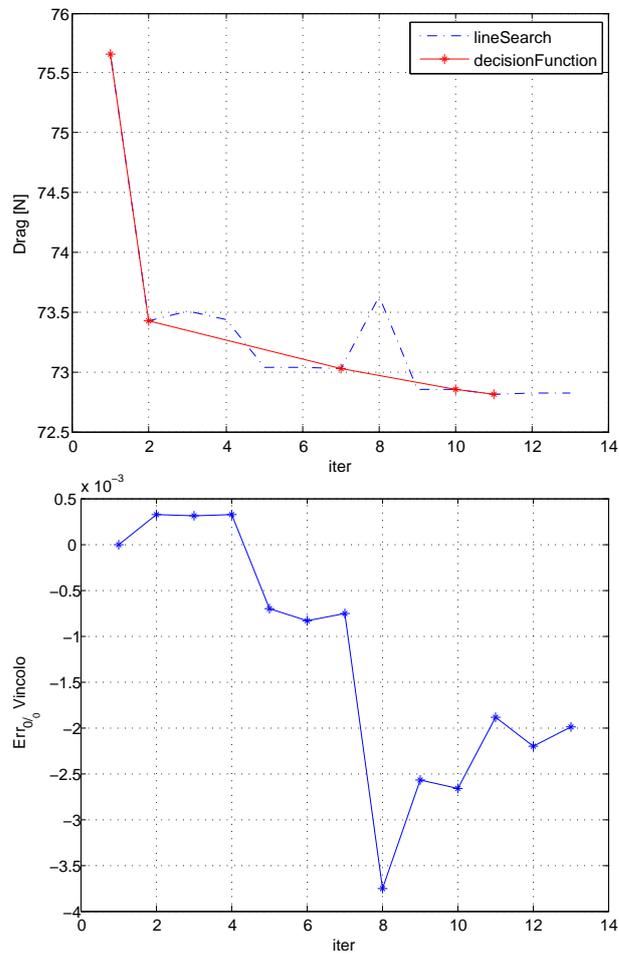


Figura 4.2: Evoluzioni della funzione obiettivo e dell'errore percentuale sul vincolo di momento

geometria	$D_{visc}$ [N]	$D_{press}$ [N]	$D$ [N]
$Geo_0$	56.68	18.58	75.26
$Geo_1$	60.44	12.38	72.82

Tabella 4.2: Termini di resistenza per le diverse geometrie

naturalmente la resistenza viscosa è invece aumentata perchè è aumentata la superficie bagnata compressiva.

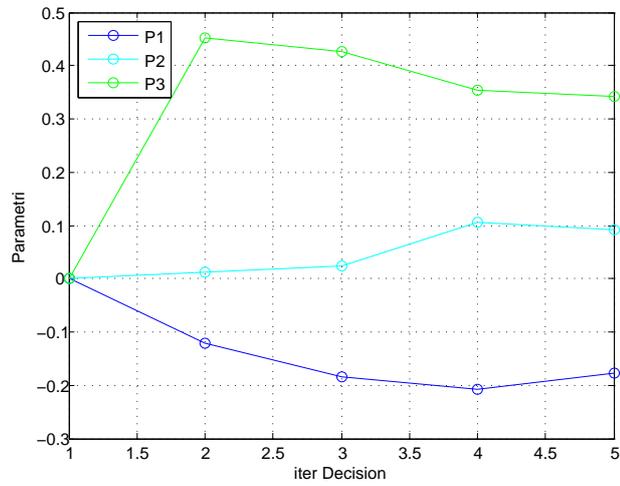


Figura 4.3: Evoluzione dei parametri

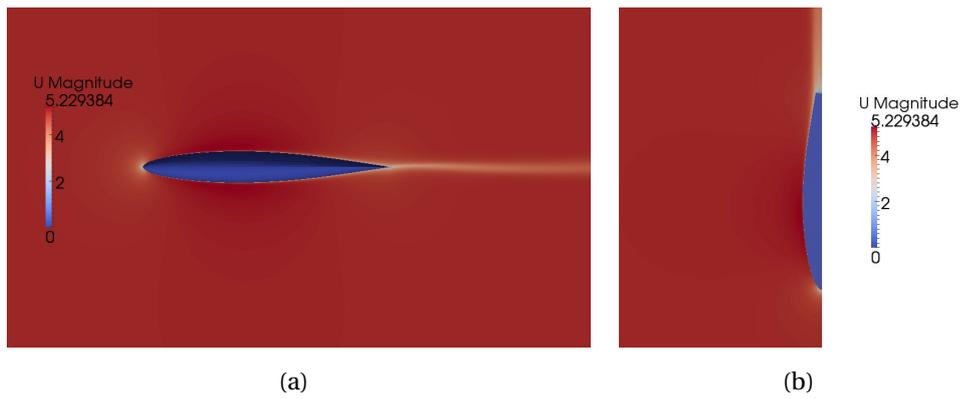


Figura 4.4: Distribuzione di velocità per  $Geo_0$ : vista laterale (a) e superiore (b)

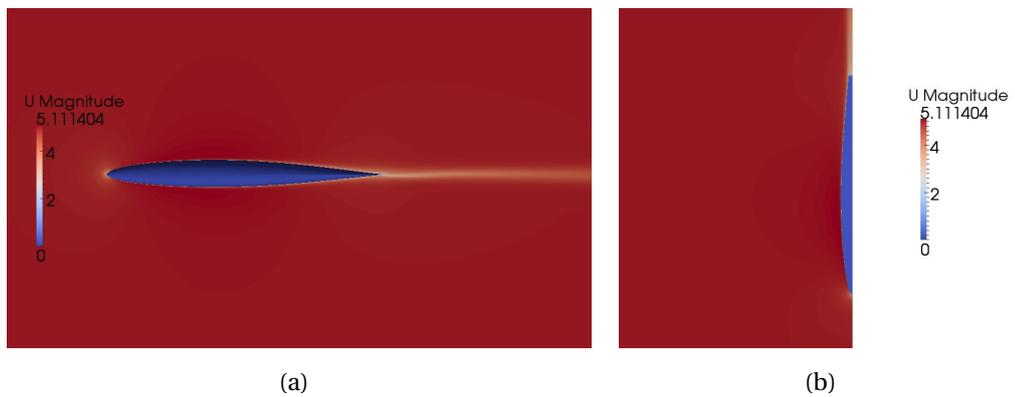


Figura 4.5: Distribuzione di velocità per  $Geo_1$ : vista laterale (a) e superiore (b)

## 4.2 Allround: compromesso bolina / poppa

Il risultato dell'analisi è la geometria di bulbo ottimizzata  $Geo_2$ , presentata in figura 4.6, e definita dal set di parametri:

- $P1 = -0.17468$  ;  $P2 = 0.00237$ ;  $P3 = 0.06339$ ;

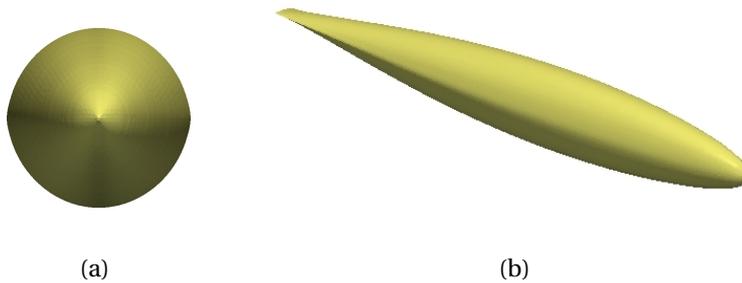


Figura 4.6: Geometria del bulbo ottimizzato  $Geo_2$ : vista frontale (a) e prospettica (b)

La scelta del parametro più interessante in questo caso risulta quella effettuata per  $P3$ , mentre per gli altri due parametri la risposta del sistema è simile a quella ottenuta per  $Geo_1$ .

Questa analisi, oltre allo studio per la poppa presentato al capitolo 4.1, aggiunge un'analisi ad un angolo di assetto tipico di un'andatura di bolina.

Ad un angolo di assetto non nullo la superficie frontale del bulbo, a cui può venire associata in prima approssimazione la resistenza di pressione, risente anche dello sviluppo della geometria nella dimensione longitudinale. Per queste geometrie caratterizzate da parametri  $P3$  elevati sono poco adatte a questa tipologia di navigazione.

Nelle figure 4.7 si mostrano le linee di flusso intorno alla geometria ottimizzata  $Geo_2$  che mettono in evidenza l'assetto di bolina.

In tabella 4.3 si riportano i vari termini di resistenza per la geometria  $Geo_2$  confrontati con  $Geo_0$  ai diversi assetti di navigazione. La diminuzione di resistenza per la geometria ottimizzata rispetto a quella originale in condizione di poppa supera il 3% mentre il miglioramento per la bolina si attesta intorno al 2%. In riferimento alla geometria  $Geo_2$  (per  $Geo_0$  si possono fare considerazioni analoghe) nel caso di assetto di bolina la resistenza di pressione aumenta di quasi l'8% rispetto alla poppa, mentre quella viscosa cambia di appena

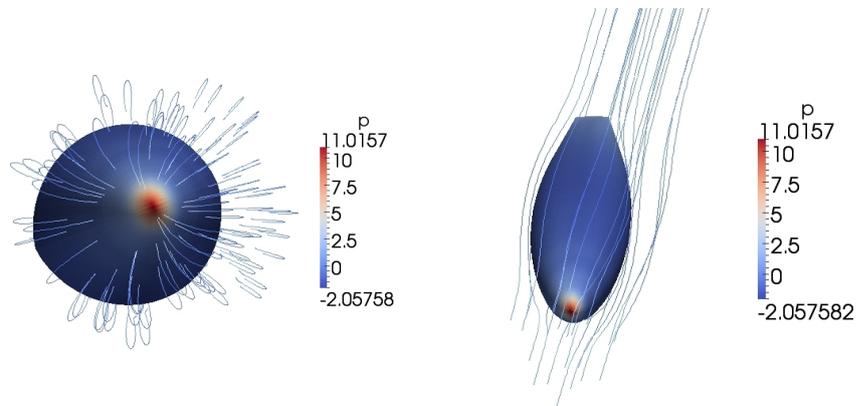


Figura 4.7: Linee di flusso intorno a  $Geo_2$  all'assetto di bolina

il 5 ‰. Questo si può spiegare perchè mentre la superficie bagnata non varia quella frontale, a cui è legato il termine di resistenza di pressione, risente dell'incidenza del bulbo.

geometria	condizione	$D_{visc}$ [N]	$D_{press}$ [N]	$D$ [N]
$Geo_0$	poppa	56.68	18.58	75.26
	bolina	56.87	19.48	76.35
$Geo_2$	poppa	55.82	17.17	72.99
	bolina	56.11	18.63	74.74

Tabella 4.3: Confronto termini di resistenza per  $Geo_0$  e  $Geo_2$  ai diversi assetti

Naturalmente la risultante di forze applicata al bulbo in assetto di bolina non è solo longitudinale, lungo  $y$ , ma ha anche due contributi lungo  $x$  e  $z$ . Tale risultato trova coerenza in figura 4.8 dove è possibile apprezzare, scalando opportunamente il range di pressione visualizzato, la distribuzione fortemente asimmetrica di pressione sul contorno del bulbo. In 4.4 si riportano le diversi componenti di resistenza nelle due direzioni laterale *side force* e verticale *vertical force*.

direzione	$F_{visc}$ [N]	$F_{press}$ [N]
<i>side force</i>	-0.26	40.33
<i>vertical force</i>	-0.11	11.76

Tabella 4.4: Componenti supplementari di forze su  $Geo_2$  in condizione di bolina

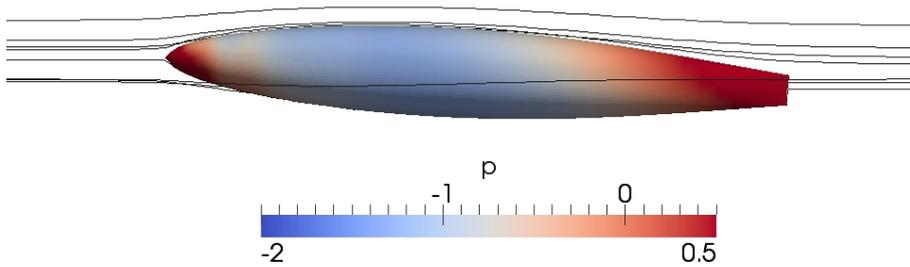


Figura 4.8: Distribuzione di pressione su  $Geo_2$  vista dall'alto

### 4.3 Multi obiettivo

L'algoritmo MOGA è stato lanciato creando una popolazione di 10 geometrie iniziali, che sono evolute per 10 generazioni. Sono stati usati i seguenti valori dei parametri: probabilità di cross-over 85%, probabilità di selezione 5% e mutazione 10%, in accordo con [25].

Il numero di valutazioni effettuate è stato di 100 su un periodo complessivo di 13 ore.

In figura 4.9 sono mostrati i risultati dell'ottimizzazione per i due obiettivi: resistenza e VCG. Il grafico mostra chiaramente come l'algoritmo abbia generato un discreto numero di geometrie non dominate; vista la complessità del problema nulla ci garantisce che queste geometrie rappresentino il vero fronte di Pareto, l'elevato numero di generazioni rende tuttavia plausibile la vicinanza di tale situazione.

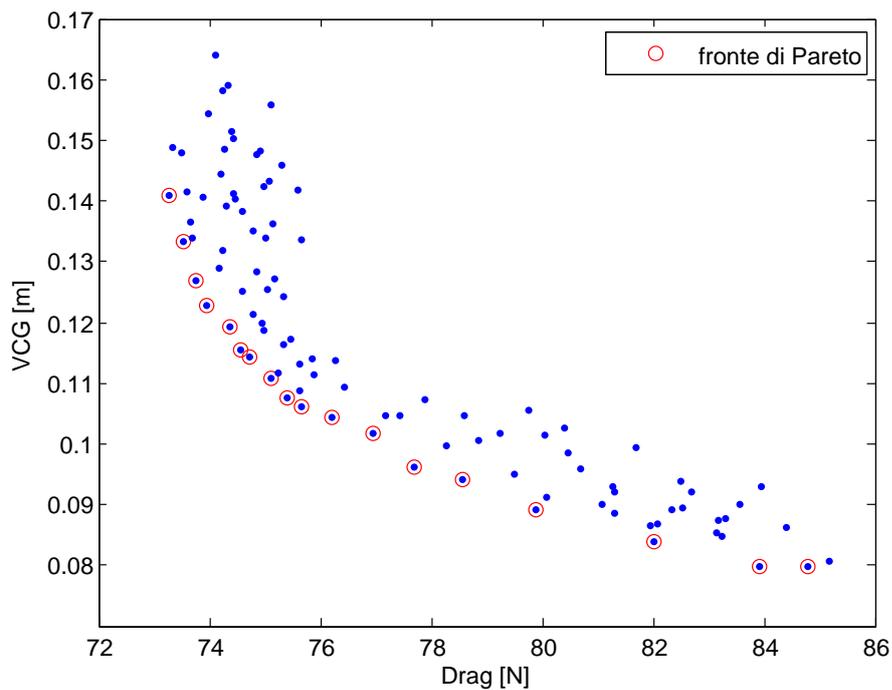


Figura 4.9: Fronte di pareto  $Geo_3$

Visto che si sono selezionate due funzioni obiettivo in conflitto che devono essere ottimizzate simultaneamente, la soluzione di tale algoritmo non è singola ma come detto è una famiglia di soluzioni plausibili. Si ricorda inoltre che il vincolo in questo caso è il volume e non il momento; l'eventuale confronto

con la analisi precedentemente condotte in termini di parametri e funzioni obiettivo deve essere fatto tenendo conto che il bulbo risultante è scalato.

Mostriamo tre soluzioni selezionate tra quelle non dominate nelle figure 4.10-4.12: le due soluzioni estreme appartenenti al fronte di Pareto caratterizzate dalla minima resistenza  $Geo_{3a}$ , dal minor VCG  $Geo_{3c}$  e il miglior compromesso tra le due funzioni obiettivo  $Geo_{3b}$  identificata come quella di minima distanza dal *punto di utopia*. Il *punto di utopia* è definito come quel punto fittizio nel piano di figura 4.9 che ha come coordinate i minimi di ogni funzione obiettivo.

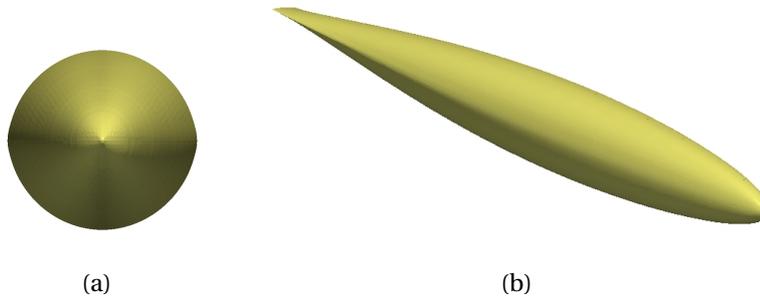


Figura 4.10: Geometria del bulbo ottimizzato  $Geo_{3a}$ : vista frontale (a) e prospettica (b)

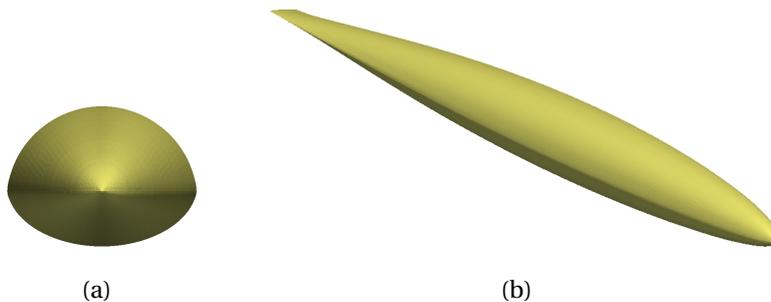


Figura 4.11: Geometria del bulbo ottimizzato  $Geo_{3b}$ : vista frontale (a) e prospettica (b)

In tabella 4.5 riportiamo le prestazioni e i parametri di ogni geometria confrontata con quella indeformata  $Geo_0$ .

La geometria  $Geo_{3b}$  può essere vista come la soluzione di compromesso migliore evidenziata nella più completa delle analisi condotte, tenente conto

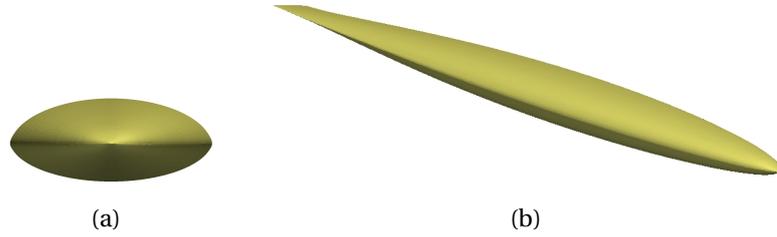


Figura 4.12: Geometria del bulbo ottimizzato  $Geo_{3c}$ : vista frontale (a) e prospettica (b)

geometria	P1	P2	P3	$D$ [N]	VCG [m]
$Geo_0$	0	0	0	75.653	0.1298
$Geo_{3a}$	-0.161	0.008	0.153	73.2837	0.140879
$Geo_{3b}$	-0.0506	0.271	0.361	75.3988	0.107666
$Geo_{3c}$	0.272	0.098	0.483	84.7870	0.0797183

Tabella 4.5: Confronto fra le geometrie selezionate nella famiglia  $Geo_3$  con la geometria indeformata  $Geo_0$

di bassa resistenza in navigazione a diversi assetti e di prestazioni raddrizzanti allo stesso tempo.  $Geo_{3b}$  riesce a migliorare entrambe le funzioni obiettivo rispetto alla geometria originale  $Geo_0$ . Studiando la risposta del sistema appare migliore come parametro di abbassamento del baricentro l'utilizzo dei gradi di libertà P2 e P3 rispetto a P1, che addirittura viene utilizzato in controfase (rendendo più circolare la sezione); risultato quindi in controtendenza con il disegno originale  $Geo_0$ .

La geometria risultante  $Geo_{3a}$  non deve venire confusa con la  $Geo_2$ : nonostante la funzione obiettivo sia la stessa in un caso si è utilizzato un metodo genetico, mentre nell'altro un metodo del gradiente. In merito al primo caso non è previsto un criterio di tolleranza che garantisca che il set evidenziato sia effettivamente sul fronte di Pareto, ovvero che la geometria minimizzi la funzione obiettivo di resistenza (si ricorda che  $Geo_{3a}$  è stata scelta come la geometria che minimizza la resistenza sul fronte di Pareto). Per questo la convergenza su un set di parametri leggermente diverso dall'analisi 4.2 non deve essere visto come un errore ma semplicemente come una tolleranza sulla soluzione.

Inoltre è opportuno evidenziare come la resistenza più bassa calcolata in

questo caso sia coerente con la diversa imposizione del vincolo. Un vincolo sul momento obbliga geometrie con VCG superiori al caso indeformato ad un aumento di volume, la  $Geo_2$  ha un volume superiore quindi a  $Geo_0$ . Il vincolo sul volume garantisce che la  $Geo_{3a}$  abbia lo stesso volume di  $Geo_0$  e quindi inferiore a  $Geo_2$ .



# Conclusioni

In questa tesi si è sviluppato un *toolbox* per l'ottimizzazione di bulbi di imbarcazioni utilizzando software *liberi*. A partire dallo studio della tipologia di imbarcazione *Classe 950* si sono cercate di evidenziare alcune specifiche di analisi da cui sviluppare le potenzialità del *toolbox*. Si è voluto perciò costruire lo strumento a valle della richiesta; l'estensione e la flessibilità del linguaggio di programmazione di *GNU Octave* hanno reso possibile la comunicazione interna tra i codici CFD *OpenFOAM* e l'ottimizzatore *DAKOTA*.

Si è presentato in forma generale il modello numerico che governa il moto attorno al bulbo a partire dalle equazioni di *Navier-Stokes*. Quindi si è posta l'attenzione sul lato computazionale mostrando i solutori utilizzati, le condizioni al contorno e la discretizzazione del problema attraverso le griglie di calcolo. Una parte importante del lavoro si è concentrata nello studio dei metodi di deformazione di griglie, sviluppando prima un algoritmo *ex-novo* chiamato *sin-2D* e quindi utilizzando delle librerie C++ già scritte per *OpenFOAM* che implementano il metodo IDW. Si è sviluppato quindi una correzione all'algoritmo che permettesse di mantenere la planarità del piano di simmetria nel caso di problemi simmetrici.

Si sono quindi mostrate le generalità di uno studio di ottimizzazione e gli algoritmi implementati in *DAKOTA* e utilizzati nella nostra analisi. Viste le specifiche di analisi evidenziate si sono utilizzati sia metodi basati sul gradiente che metodi non basati sul gradiente (metodi genetici).

Infine sono stati condotti gli studi di ottimizzazione mostrando le geometrie ottenute e commentando i risultati alla luce della fisica che regge il problema. La risposta del sistema più evidente è la grande dipendenza della geometria ottimizzata dalla definizione della funzione obiettivo; ogni analisi condotta ha fornito risposte molto diverse dalle altre evidenziando ancora una volta come la corretta imposizione degli obiettivi sia un punto fondamentale di uno studio di ottimizzazione. Le analisi condotte, di complessità e accuratezza via

via crescente, hanno mostrato come la geometria di partenza fosse comunque una buona forma; allo stesso tempo hanno evidenziato quali sono i parametri più influenti nell'ottica di migliorare una prestazione specifica.

## **Limiti dello studio**

Si ritiene utile riassumere qui le ipotesi e le semplificazioni effettuate che possono limitare l'accuratezza della soluzione. Si ricorda che l'ambizione di questa tesi non è quella di presentare la geometria migliore in assoluto quanto piuttosto una metodologia di studio per un problema di ottimizzazione. I vari passi di analisi del problema, di modellazione e di studio sono bilanciati; le ipotesi sono quindi parte integrante dello studio.

Il tema dell'interferenza nell'innesto tra deriva e bulbo è un argomento di ricerca molto interessante e tutt'altro che esaurito [28]. Tipicamente il punto di innesto è punto di ristagno e può portare alla separazione della corrente. Tale separazione che viene a trovarsi sul bulbo e che non viene rappresentata nell'analisi del bulbo isolato, produce un moto fortemente tridimensionale provocando una resistenza supplementare del corpo affusolato. Un'analisi di bulbo isolato trascura tutto l'effetto di interferenza.

Come si è già discusso, il solutore utilizzato per la CFD non tiene conto di eventuali laminarità nello strato limite. Non si sono quindi potute apprezzare quelle geometrie che a fronte di un aumento di superficie bagnata riescono a garantire resistenze inferiori per una maggiore estensione della regione di laminarità.

Anche la parametrizzazione della geometria scelta presenta dei limiti. Una parametrizzazione globale è, per esempio, molto sensibile alla geometria di partenza. Se ci fosse una caratteristica geometrica del bulbo particolarmente penalizzante ma associata ad una zona limitata, come potrebbe essere il naso o la dimensione della coda, ci si ridurrebbe ad un'ottimizzazione poco interessante perchè anche il risultato finale manterrebbe eventuali limiti della geometria iniziale.

## **Sviluppi futuri**

Alcuni sviluppi futuri che prevediamo sono volti più che altro a rivedere alcune ipotesi o scelte che possono inficiare la bontà dell'analisi, ripercorrono

cioè alcune problematiche evidenziate nella sezione 4.3 mostrandone un approccio migliorativo. Altri invece sono vere e proprie estensioni dell'analisi, andando a caratterizzare il problema in maniera sempre più completa e accurata. Gli sviluppi futuri prospettati sono i seguenti.

### **Ottimizzazione**

Potrebbe essere interessante effettuare un'ottimizzazione controllando in maniera diversa la forma, ovvero gestendo deformazioni locali. La maggior parte dei software CAD (Computer Aided Design) costruisce geometrie tipo quelle del bulbo attraverso l'unione di NURBS/SPLINES parametrizzate che rappresentano curve e superfici in due o tre dimensioni. I parametri in questo caso possono essere visti come i gradi di libertà di alcuni punti di controllo attraverso i quali la curva deve passare, il numero necessario per garantire un'adeguata mobilità alla geometria è sicuramente almeno un ordine di grandezza superiore a quello utilizzato in questa tesi. Sarebbe perciò necessario sviluppare uno studio di ottimizzazione in parallelo (e non solo la valutazione CFD) su almeno 2-4 geometrie concorrenti nello stesso tempo. Lo stesso *DAKOTA* prevede tale possibilità, si veda ad esempio [27].

### **CFD**

L'analisi CFD può essere sviluppata su almeno due fronti: per prima cosa sarebbe interessante mantenere lo stesso problema migliorando le condizioni iniziali di studio. L'analisi potrebbe essere lanciata per esempio su uno spettro di velocità e angoli di incidenze più ampio, in condizioni di navigazione ricavate da uno studio delle polari di imbarcazioni simili.

In secondo luogo un'analisi che tenga conto della presenza della pinna di deriva caratterizzerebbe meglio l'interfaccia, modellando cioè la geometria del bulbo integrata. In questo caso potrebbe essere interessante andare a definire come parametro supplementare anche il punto di innesto della pinna nel bulbo.

### **Dinamica imbarcazione**

Introdurre un'interfaccia con un VPP porterebbe a tenere conto delle diverse funzioni obiettivo in maniera più accurata. Le prestazioni idrodinamiche (calcolo CFD) e le prestazioni di stabilità (VPP) verrebbero considerate direttamente, senza imporre vincoli di volume o momento come è stato fatto in questa tesi. Si potrebbe pensare di simulare con il VPP un percorso di regata obiettivo a bastone, caratterizzato da un lato di bolina e uno di poppa, su cui minimizzare il tempo di navigazione per un set discreto di diverse intensità di vento. Naturalmente il VPP dovrebbe in questo caso essere integrato nel ciclo

di ottimizzazione, in riferimento alla figura 1 dovrebbe essere posto a valle della simulazione CFD.

La forza propulsiva generata dal piano velico di un'imbarcazione dipende fortemente dalle oscillazioni applicate, le perturbazioni più critiche sono quelle associate al moto ondoso. Il comportamento dinamico dell'imbarcazione forzato da un sistema di onde a frequenza differente può essere più o meno smorzato a seconda che la frequenza tipica di beccheggio di imbarcazione completa, in cui entra in gioco il momento d'inerzia del bulbo, sia più o meno vicina alla frequenza delle onde. Significa che un bulbo che è risultato come la geometria ottimale in assenza di moto ondoso potrebbe non essere la migliore in un contesto perturbato. Questo genere di studi sulla dinamica longitudinale dell'imbarcazione viene tipicamente condotto per via sperimentale in una vasca navale, si veda ad esempio [25].

# Appendici

## Software libero

Un software libero è un programma tipicamente disponibile in rete, rilasciato con licenza e con una filosofia che incentiva lo studio, le modifiche e la diffusione. La definizione potrebbe essere ambigua: con la parola libero non bisogna intendere solamente gratuito perchè si porrebbe l'attenzione sul lato economico che è uno solo dei tratti caratteristici di questa tipologia di software.

L'idea alla base del software libero risale al 1984 ed è da attribuirsi a R. Stallman quando creò una nuova tipologia di licenza, la GNU *General Public License* (GPL), la prima di tipo copyleft. Per la prima volta il detentore del copyright poteva, se lo desiderava, assicurare che il massimo numero di diritti si trasferisse in maniera perpetua agli utenti del programma, a prescindere dalle eventuali modifiche successivamente apportate da chiunque al programma originale.

Il copyleft altro non è che una modalità di esercizio del diritto d'autore che sfrutta i principi di base del diritto d'autore non per controllare la circolazione dell'opera bensì per stabilire un modello virtuoso di circolazione dell'opera, che si contrappone al modello detto proprietario. La licenza vuole comunque preservare gli autori della proprietà intellettuale del programma, significa che i nomi degli autori originali devono essere preservati tutte le volte che vengono modificate o aggiornate versioni del software.

Una licenza basata sui principi del copyleft trasferisce a chiunque posseda una copia dell'opera alcuni dei diritti propri dell'autore. Inoltre consente la redistribuzione dell'opera stessa solo se tali diritti vengono trasferiti assieme ad essa.

Fondamentalmente, questi diritti sono le quattro libertà fondamentali indicate da Stallman:

- libertà di eseguire il programma per qualsiasi scopo

- libertà di studiare il programma e modificarlo, l'accesso al codice sorgente ovviamente ne è un prerequisito
- libertà di ridistribuire copie del programma in modo da aiutare il prossimo
- libertà di migliorare il programma e di distribuirne pubblicamente i miglioramenti, in modo che tutta la comunità ne tragga beneficio

La licenza GNU *General Public License* (GPL) è molto restrittiva perchè obbliga che ogni applicazione derivata sia distribuita con la stessa licenza. In alternativa esiste una versione meno restrittiva chiamata GNU *Lesser General Public License* (LGPL) che consente di utilizzare il programma anche nell'ambito di software proprietari.

I vantaggi del software libero rispetto a quello proprietario sono diversi, innanzitutto la disponibilità del codice sorgente permette di adattare in modo flessibile il programma alle particolari esigenze dell'utente; inoltre il fatto che il codice sorgente sia sottoposto alla continua revisione di una base di utenti molto maggiore rende la fase di individuazione e correzione di errori molto efficiente. Tuttavia l'assenza di documentazione aggiornata o la mancanza spesso di un'interfaccia grafica ne rendono l'utilizzo ma soprattutto l'avvicinamento molto difficile.

Nella tabella 4.6 si riportano le licenze sotto le quali sono distribuiti i vari programmi *liberi* utilizzati.

programma	licenza
<i>OpenFOAM</i>	GPL
<i>DAKOTA</i>	LGPL
<i>GNU Octave</i>	GPL

Tabella 4.6: Licenze dei programmi utilizzati

# Lista degli acronimi

CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
DAKOTA	Design Analysis Kit for Optimization and Terascale Applications
DNS	Direct Numerical Simulation
FVM	Finite Volume Method
GNU	GNU's Not Unix
GPL	General Public License
IDW	Inverse Distance Weighting
LGPL	Lesser General Public License
MYD	Master Yacht Design
OpenFOAM	Open Field Operation and Manipulation
RANS	Reynolds Averaged Navier-Stokes
VCG	Vertical Center Gravity
VPP	Velocity Prediction Program



# Bibliografia

- [1] R. AZCUETA, Computation of turbulent free-surface flows around ships and floating bodies, *PhD thesis, Technical University of Hamburg-Harburg, 2001.*
- [2] D. DETOMI ET AL., Numerical models and simulations in sailing yacht design, *In M. Peters, editor, Computational Fluid Dynamics for Sport Simulation, pages 1-31, Springer, 2009.*
- [3] N. PAROLINI AND A. QUARTERONI, Mathematical models and numerical simulations for the America's cup, *Computer Methods in Applied Mechanics and Engineering, 194 pp 1001-1026, 2005.*
- [4] P. J. ZWART ET AL., Simulation of unsteady free-surface flow around a ship hull using a fully coupled multi-phase flow method, *Journal of Marine Science and Technology, 13 pp 346-355, 2008.*
- [5] <http://www.openfoam.org>.
- [6] <http://dakota.sandia.gov/index.html>.
- [7] <http://www.gnu.org/software/octave/>.
- [8] <http://engits.eu/en/engrid>.
- [9] <http://www.ansys.com/Products/Workflow+Technology/ANSYS+Workbench+Platform/ANSYS+Meshing>.
- [10] [http://www.classe950.com/IMG/pdf/Jauge\\_engl\\_122010.pdf](http://www.classe950.com/IMG/pdf/Jauge_engl_122010.pdf).
- [11] <http://www.classe950.com/spip.php?article46>.

- [12] K. BOCHON, W. WROBLEWSKY, S. DYKAS, Modelling of influence of turbulent transition on heat transfer conditions, *Task, Quarterly 12 No 3, 173-184, Gliwece (Poland), Silesian University of Technology, 30 June 2008.*
- [13] J. H. FERZIGER, M. PERIC, Computational Methods for Fluid Dynamics, *Springer, second edition, 1999.*
- [14] J. BOUSSINESQ, Theorie de l'ecoulement tourbillant, *memoria presentata all'Accademia della Scienze di Francia, N°XXIII, 46, Parigi, 1877.*
- [15] F. R. MENTER ET AL., Ten years of industrial experience with the SST turbulence model, *Heat and mass transfer, 4 pp 625-632, 2003.*
- [16] A. QUARTERONI, R. SACCO, F. SALERI, Matematica numerica, *Springer, 3a edizione, 2008.*
- [17] L. DAVIDSON, An introduction to Turbulence Models, *Publication 97/2, Goteborg (Sweden), Chalmers University of Technology, November 2003.*
- [18] S. IACOBUCCI, Calcolo parallelo e sistemi multicore, *Embedded 34, Febbraio 2010.*
- [19] M. LOMBARDI, Numerical simulation of a sailing boat: free surface, fluid structure interaction and shape optimization, *PhD Thesis, Lausanne (Swiss), EPFL, 2012.*
- [20] H. JASAK, Error analysis and estimation for the finite volume method with application to fluid flows, *PhD Thesis, London (England), Imperial College, June 1996.*
- [21] A. O. GONZALEZ, Mesh motion alternatives in *OpenFOAM CFD with OpenSource Software, Assignment 3, Goteborg (Sweden), Chalmers University of Technology, December 2009.*
- [22] H. JASAK, Z. TUKOVIC, Dynamic mesh handling in *OpenFOAM* applied to fluid-structure interaction simulation *V European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010, Lisbon (Portugal), 14-17 June 2010.*
- [23] G. ROMANELLI, E. SERIOLI, Un approccio libero alla moderna aeroelasticità computazionale, *MSc Thesis, Milano, Politecnico di Milano, 2008.*

- [24] J. WITTEVEEN, H. BIJL, Explicit mesh deformation using inverse distance weighting interpolation *19th AIAA Computational Fluid Dynamics, 22-25 June 2009, San Antonio, Texas.*
- [25] P. GEREMIA, M. POIAN, S. POLES, Genetic Optimization for Yacht Design *GECCO '07, July 7-11 2007.*
- [26] J. NOCEDAL, S. J. WRIGHT, Numerical Optimization, *Springer 1999.*
- [27] SANDIA NATIONAL LABORATORIES, DAKOTA, A Multilevel Parallel Objected-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis - Version 5.2 User's Manual *Updated November 30, 2011.*
- [28] G. SCANDELLI, Ottimizzazione numerica del raccordo tra un corpo assialsimmetrico ed un corpo aerodinamico a generatrice cilindrica, *MSc Thesis, Politecnico di Milano (2011/2012).*
- [29] K. LJUNGQVIST, Shape optimisation of an integrated bulb-keel, *MSc Thesis, Goteborg (Sweden), Chalmers University of Technology, 2011.*