

UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA
MECCANICA

Development of a parametric tool for shape manipulation and fluid dynamics optimization of an UHBR engine nacelle

Laureando:
Denis BUOSI

Relatore:
Prof. Ernesto BENINI

A handwritten signature in blue ink, appearing to read 'Ernesto Benini', is written over the printed name of the supervisor.

Anno accademico 2018 / 2019

*Success is not final,
failure is not fatal:
it is the courage to continue that counts.*

*Il successo non è mai definitivo,
il fallimento non è mai fatale:
è il coraggio di continuare che conta.*

Winston Churchill

NOMENCLATURE

A [-]	Matrice dei vincoli di disuguaglianza
A_{eq} [-]	Matrice dei vincoli di uguaglianza
AIP [-]	Piano di Interfaccia Aerodinamica
AR [-]	Aspect Ratio
$B_{i,n}$ [-]	Funzione base generica per una Bézier
b [-]	Vettore dei termini noti dei vincoli di disuguaglianza
b_{eq} [-]	Vettore dei termini noti dei vincoli di uguaglianza
BC [-]	Boundary Conditions
BPR [-]	Bypass Ratio
C_L [-]	Coefficiente di portanza
C_P [-]	Coefficiente di pressione
C_{ref} [mm]	Corda di riferimento
CFD [-]	Computational Fluid Dynamics
CRM [-]	Common Research Model
CST [-]	Class Shape Trasformation
$C(u)$ [-]	Curva parametrica
D_{fan} [mm]	Diametro del fan
D_{Hi} [mm]	Diametro di Highlight
DPW [-]	Drag Prediction Workshop
ε [mm]	Errore locale di fitting
$\varepsilon_{fitting}$ [mm]	Errore di fitting
ε_{max} [mm]	Errore massimo di fitting
ε_{mean} [mm]	Errore medio di fitting
FPR [-]	Fan Pressure Ratio
$H_{by,core}$ [mm ²]	Sezione d'uscita del condotto caldo
$H_{by,fan}$ [mm ²]	Sezione d'uscita del condotto freddo
$H_{th,core}$ [mm ²]	Sezione di gola del condotto caldo
$H_{th,fan}$ [mm ²]	Sezione di gola del condotto freddo
JSM [-]	Jaxa Standard Model
K [-]	Coefficiente proporzionale per determinare il numero di individui
L_{AIP} [mm]	Distanza assiale della nacelle esterna dal punto di raggio di gola all'AIP
L_{cc} [mm]	Lunghezza core cowl
$L_{nacelle}$ [mm]	Lunghezza assiale della nacelle esterna
$L_{nozzle,core}$ [mm]	Lunghezza dell'ugello supersonico del condotto caldo
$L_{nozzle,fan}$ [mm]	Lunghezza dell'ugello supersonico del condotto freddo
L_{post} [mm]	Distanza assiale della nacelle esterna dal punto di raggio massimo al bordo d'uscita
L_{plug} [mm]	Lunghezza della spina
L_{pre} [mm]	Distanza assiale della nacelle esterna dal bordo d'ingresso al punto

	di raggio massimo
$L_{spinner}$ [mm]	Lunghezza assiale dello spinner
L_{throat} [mm]	Distanza assiale della nacelle esterna dal bordo d'ingresso al punto di raggio di gola
lb [-]	Vettore dei limiti inferiori di ricerca
LE [-]	Leading Edge
M_{free} [-]	Numero di nodi liberi della curva parametrica
$m_{th,core}$ [-]	Coefficiente angolare della sezione di gola del condotto caldo
$m_{th,fan}$ [-]	Coefficiente angolare della sezione di gola del condotto freddo
\dot{m}_{bypass} [m^3/kg]	Portata di massa del turbofan
\dot{m}_{bypass} [m^3/kg]	Portata di massa nel condotto freddo
\dot{m}_{core} [m^3/kg]	Portata di massa nel condotto caldo
m [-]	Indice dell'ultimo nodo di una curva parametrica
M [-]	Numero di nodi della curva parametrica
Ma [-]	Indice di Mach di volo
$MFCR$ [-]	Mass Flow Capture Ratio
$N_{generazioni}$ [-]	Numero di generazioni
$N_{individui}$ [-]	Numero di individui
$N_{i,n}$ [-]	Funzione base generica per una BSpline
$N_{variabili}$ [-]	Numero di variabili
$N_{vincoli}$ [-]	Numero di vincoli
n [-]	Numero di punti di controllo della curva parametrica
$NURBS$ [-]	Non Uniform Rational B-Spline
OPR [-]	Overall Pressure Ratio
P_i [-]	Punto di controllo generico di una curva parametrica
$P_{i,x}$ [-]	Coordinata assiale di un punto di controllo generico
$P_{i,y}$ [-]	Coordinata radiale di un punto di controllo generico
p [-]	Grado della curva parametrica
P [-]	Matrice dei punti di controllo di una curva parametrica
R_{AIP} [mm]	Raggio di AIP
$R_{core,hub}$ [mm]	Raggio all'uscita della BC dopo la turbina, lato mozzo
$R_{core,shroud}$ [mm]	Raggio all'uscita della BC dopo la turbina, lato corona
$R_{exit,core}$ [mm]	Raggio della nacelle interna all'uscita del condotto caldo
$R_{exit,fan}$ [mm]	Raggio della nacelle esterna all'uscita del condotto freddo
$R_{fan,hub}$ [mm]	Raggio all'uscita della BC dopo il fan, lato mozzo
$R_{fan,shroud}$ [mm]	Raggio all'uscita della BC dopo il fan, lato corona
R_{Hi} [mm]	Raggio di Highlight
R_{max} [mm]	Raggio massimo della nacelle esterna
$R_{i,n}$ [-]	Funzione base generica per una NURBS
$R_{spinner}$ [mm]	Raggio dello spinner
R_{throat} [mm]	Raggio di gola della nacelle esterna
Re [-]	Numero di Reynolds
S_{ref} [mm^2]	Superficie di riferimento
TE_1 [mm]	Trailing Edge della nacelle esterna
TE_2 [mm]	Trailing Edge della nacelle interna
TE_3 [mm]	Trailing Edge della spina
$TE_{1,angle}$ [rad]	Angolo del Trailing Edge della nacelle esterna

$TE_{2,angle}$ [rad]	Angolo del Trailing Edge della nacelle interna
$TE_{3,angle}$ [rad]	Angolo del Trailing Edge della spina
T_{max} [K]	Temperatura massima di ciclo all'uscita della turbina
TE [-]	Trailing Edge
TFN [-]	Through-Flow Nacelle
u_k [-]	Nodo generico di una curva parametrica
ub [-]	Vettore dei limiti superiori di ricerca
U [-]	Vettore dei nodi di una curva parametrica
UBS [-]	Uniform B-Spline
$UHBR$ [-]	Ultra High Bypass Ratio
UI [-]	User Interface
V_{state} [-]	Vettore con lo stato delle variabili della nacelle
V_{value} [-]	Vettore dei valori delle variabili della nacelle
w_i [-]	Peso generico di una curva parametrica
W_{fan} [mm]	Larghezza del occupata dal fan nella nacelle
W_{core} [mm]	Larghezza del occupata dal core nella nacelle
WB [-]	Wing Body configuration
$WBNP$ [-]	Wing Body Nacelle Pylon configuration
$x_{th,core}$ [mm]	Posizione della gola nel condotto caldo rispetto al bordo d'attacco della nacelle
$x_{th,fan}$ [mm]	Posizione della gola nel condotto freddo rispetto al bordo d'attacco della nacelle
X_{LE} [mm]	Distanza assiale tra il bordo d'attacco della nacelle e il bordo d'attacco dell'ala
Y_{LE} [mm]	Distanza radiale tra il bordo d'attacco della nacelle e il bordo d'attacco dell'ala
y'_0 [-]	Tangente nel punto iniziale di una curva parametrica
y''_0 [-]	Curvatura nel punto iniziale di una curva parametrica
y'_1 [-]	Tangente nel punto finale di una curva parametrica
y''_1 [-]	Curvatura nel punto finale di una curva parametrica
λ [-]	Taper Ratio

CONTENTS

1	Introduction	1
1.1	CRM	3
2	Curve Parametriche	9
2.1	Introduzione	9
2.2	Confronto curve parametriche	10
2.2.1	Bézier	10
2.2.2	BSpline	11
2.2.3	Nurbs	14
2.3	Vincoli curve parametriche	16
2.3.1	Punti estremali	16
2.3.2	Nodi centrali	17
2.3.3	Vincoli di tangenza	18
2.3.4	Vincoli di curvatura	18
3	Parametrizzazione	21
3.1	Quote geometriche	23
3.1.1	Dimensioni Globali	23
3.1.2	Dimensioni Locali	24
3.2	Aspetti specifici di ogni curva	25
3.2.1	Curva 1	26
3.2.2	Curva 2	26
3.2.3	Curva 3	27
3.2.4	Curva 4	28
3.2.5	Curva 5	29
3.2.6	Curva 6	29
3.2.7	Curva 7	31
3.2.8	Curva 8	32
3.2.9	Curva 9	33
3.2.10	Curva 10	34
3.2.11	Curva 11	35
3.2.12	Curva 12	35
3.2.13	Curva 13	36
3.3	Costruzione della baseline	37
3.4	Valutazione sensibilità di ogni curva	40
4	Struttura del codice	43
4.1	Blocco di parametrizzazione	43
4.1.1	UI selection	44
4.1.2	Optimization problem setting	51

4.1.3	Nacelle fitting	53
4.2	Blocco di ottimizzazione	54
4.2.1	Optimization Bounds	54
4.2.2	Optimization Constraint	56
4.2.3	Impostazione del ga	61
4.2.4	Ottimizzazione	62
5	Conclusions	67
Appendices		
A	Tangente ad una NURBS	69
A.1	Casi particolari	71
B	Curvatura di una NURBS	73
B.1	Casi particolari	77
C	Codice di calcolo per curve NURBS	79
C.1	Bspline basis function	80
D	Codice per costruire un vincolo di disuguaglianza generico	85

LIST OF FIGURES

1.1	Andamenti di pressione nel CRM in configurazione WB e WBNP	5
1.2	C_P calcolati con CFD per il CRM in configurazione WB e WBNP 2	6
1.3	C_P testati e simulati per il CRM in configurazione WB	7
1.4	C_P testati e simulati per il CRM in configurazione WBNP	7
1.5	C_P della nacelle del CRM	8
2.1	Esempio di curva di Bézier con poligono di controllo a gradino	11
2.2	Esempio di curva di BSpline con poligono di controllo a gradino, al variare del nodo centrale	12
2.3	Estensione BSpline di Figura 2.2	13
2.4	Esempio di effetto locale di una BSpline per la modifica di un punto di controllo	13
2.5	Esempio di effetto locale di una BSpline per la modifica di un nodo centrale	14
2.6	Confronto curve parametriche con un circonferenza	15
3.1	Profilo 2D di riferimento	21
3.2	Profilo 2D di riferimento suddiviso nelle varie parti	22
3.3	Dimensioni globali utilizzate nella parametrizzazione	24
3.4	Sistema di coordinate utilizzato nella generazione del profilo 2D	25
3.5	Curva 1 e relative dimensioni locali	26
3.6	Curva 2 e relative dimensioni locali	27
3.7	Curva 3 e relative dimensioni locali	28
3.8	Curva 4 e relative dimensioni locali	28
3.9	Curva 5 e dimensioni locali relative al punto iniziale	29
3.10	Curva 5 e dimensioni locali relative al punto finale	30
3.11	Curva 6 e relative dimensioni locali	30
3.12	Curva 7 e relative dimensioni locali	32
3.13	Curva 8 e relative dimensioni locali	32
3.14	Curva 9 e relative dimensioni locali	33
3.15	Curva 10 e relative dimensioni locali	34
3.16	Curva 11 e relative dimensioni locali	35
3.17	Curva 12 e relative dimensioni locali	36
3.18	Curva 13 e relative dimensioni locali	37
3.19	Dimensioni della TFN del CRM [23]	38
3.20	Profilo nacelle JSM	40
3.21	Errore di fitting per la curva 1 al variare del numero di variabili	41
3.22	Errore di fitting, rispetto all'errore minimo, per la curva 1 al variare del numero di variabili	42
4.1	Flowchart principale del codice	43
4.2	Flowchart della parametrizzazione	44

4.3	Interfaccia grafica principale	46
4.4	Interfaccia grafica secondaria	47
4.5	Schema a blocchi dell'interfaccia grafica principale	49
4.6	Schema a blocchi dell'interfaccia grafica secondaria	50
4.7	Schema a blocchi per l'impostazione del problema di ottimizzazione	52
4.8	Schema a blocchi per il fitting della nacelle di partenza	53
4.9	Flowchart per il blocco di ottimizzazione	54
4.10	Esempio di limiti per le variabili dimensionali	55
4.11	Flowchart per ottenere i limiti di ricerca dell'ottimizzatore	56
4.12	Schema logico per la costruzione del primo vincolo dimensionale	58
4.13	Schema logico per la costruzione di un vincolo generico	59
4.14	Flowchart per la costruzione dei vincoli	60
4.15	Flowchart per la valutazione di un individuo dell'ottimizzazione	63
4.16	Esempio di mesh generata attorno ad una nacelle	64
4.17	Esempio del boundary layer della mesh generata	65
A.1	NURBS di esempio con relativa derivata prima	71
B.1	NURBS di esempio con relativa derivata prima e seconda	77

LIST OF TABLES

1.1	Dimensioni di riferimento per il CRM	4
1.2	Dimensioni di riferimento per il CRM in millimetri	4
1.3	Condizioni di volo di design per il CRM	4
1.4	Condizioni di volo di design per il DPW 6	5
2.1	Riassunto parametri necessari per generazione di una curva 2D	15
2.2	Riassunto parametri necessari per una curva 2D con passaggio per i punti estremali	17
2.3	Riassunto parametri necessari per generazione di una curva 2D con passaggio per i punti estremali, caso pratico con n e p fissati	17
A.1	Parametri usati per generazione di NURBS per esempio di calcolo di derivata prima	71

SUMMARY

The objective of this thesis is the development of a calculation code aimed at the geometric parameterization of an axial symmetrical nacelle for turbofan UHBR and its fluid-dynamic optimization.

For the geometric modeling of the nacelle, aimed at controlling the shape, size and position of the same, we have handled the corresponding 2D profile, through which we obtained the 3D geometry with a revolution.

For a better control, the 2D profile has been divided into 13 parts, each of which has been parameterized with BSpline or NURBS, in order to guarantee local control over each curve and maximum flexibility.

Subsequently, through 3D fluid dynamic simulations, it is possible to evaluate the performance of the nacelle, and therefore of the engine, considering, in particular, the interactions of the fluid with the aircraft.

A multi-objective optimization was then set up in order to find the optimal 2D nacelle profile to improve fluid-dynamic performance and the nacelle-wing interaction. In particular, the goal is the minimization of the additional drag, given by the installed nacelle, in different flight conditions.

L'obiettivo di questa tesi è lo sviluppo di un codice di calcolo finalizzato alla parametrizzazione geometrica di una nacelle assialsimmetrica per turbofan UHBR e la sua ottimizzazione fluidodinamica.

Per la modellazione geometrica della nacelle, mirata a controllare la forma, le dimensioni e la posizione della stessa, si è maneggiato il corrispettivo profilo 2D, attraverso il quale si è ottenuta la geometria 3D con una rivoluzione.

Per un miglior controllo il profilo 2D è stato suddiviso in 13 parti ognuna delle quali è stata parametrizzata con BSpline o NURBS, in modo da garantire il controllo locale su ogni curva e la massima flessibilità.

Successivamente, attraverso simulazioni fluidodinamiche 3D, è possibile valutare le prestazioni della nacelle, e quindi del motore, considerando, in particolare, le interazioni del fluido con l'aereo.

È stata quindi impostata un'ottimizzazione multi-obiettivo con lo scopo di ricercare il profilo 2D ottimale della nacelle per migliorare le prestazioni fluidodinamiche e l'interazione nacelle - ala. In particolare, l'obiettivo è la minimizzazione del drag aggiuntivo, dato dalla nacelle installata, in diverse condizioni di volo.

INTRODUCTION

Quando camminerete sulla terra dopo aver volato, guarderete il cielo perché là siete stati e là vorrete tornare. Leonardo da Vinci

Volare, da Icaro in poi volare è sempre stato un desiderio ancestrale dell'uomo. Questa ambizione accompagnò l'umanità fino al 17 Dicembre 1903, quando i fratelli Wright riuscirono a percorrere i primi metri volati, e non planati, della storia umana. I due pionieri si resero conto di come l'aspetto chiave per volare fosse la presenza di un propulsore nel velivolo. Da quel giorno in poi, come predetto da Leonardo da Vinci, in cui l'uomo calcò i cieli per i primi secondi della sua storia, non lo lasciò più e, anzi, cercò di farlo sempre in modo più efficiente.

Il primo propulsore della storia era un motore a combustione interna che metteva in rotazione un'elica in legno. Questa tecnologia dimostrò però, fin dai primi anni della sua storia, il suo limite più grande: le limitate velocità massime raggiungibili. La seconda pietra miliare nella storia del volo moderno, volto a superare proprio questa limitazione, si deve a Frank Whittle, il quale nel 1928 ideò il primo turbogetto moderno. Questo propulsore, con il passare degli anni, sostituì il motore ad elica usato dai fratelli Wright. Ma il primo vero volo con un turbogetto equipaggiato si ebbe solo il 27 Agosto 1939 quando Erich Warsitz, riuscì a far volare l'He 178 fino ad una velocità di circa 600 km/h.[1]

Il terzo fondamentale passo, nella storia del volo terrestre, è stata l'intuizione di sfruttare un propulsore "ibrido" tra il motore ad elica dei fratelli Wright e il turbogetto, nacque così il turbofan, tecnologia che tutt'oggi viene impiegata nella maggior parte degli aerei civili. Questo motore, formato da un grande fan anteposto ad un classico turbogetto, ha permesso all'uomo di ottenere efficienze sempre più alte e quindi voli più lunghi ed economici, mantenendo, tuttavia, le velocità tipiche di crociera del motore a getto. Il primo turbofan moderno, ad alto rapporto di bypass, fu testato il 13 Febbraio 1964 dall'azienda AVCO-Lycoming e poco dopo la General Electric creò il primo modello destinato alla produzione, il TF39, dotato di un rapporto di bypass BPR pari a 8 circa e un rapporto totale di compressione pari a 25 circa. [2]

I principali parametri di design che descrivono e caratterizzano questi propulsori sono i seguenti 3:

- BPR : rapporto tra la portata di massa \dot{m} che viene elaborata dal solo fan \dot{m}_{bypass} e quella elaborata anche dal turbogetto \dot{m}_{core}
- T_{max} : temperatura massima di ciclo raggiunta in uscita dalla turbina.

- OPR : rapporto tra la pressione totale a valle del compressore e la pressione totale indisturbata

Nelle fasi iniziali di sviluppo, per migliorare le prestazioni di questi propulsori, ci si concentrò principalmente sulla parte in comune con il turbogetto, andando quindi a lavorare sui componenti principali (compressore, camera di combustione e turbina) e il loro accoppiamento. In particolare, si notò come al crescere di T_{max} e di OPR migliorava progressivamente l'efficienza del propulsore. Si raggiunsero però velocemente i limiti tecnici imposti dai materiali usati, ad esempio, T_{max} non deve superare la temperatura massima di lavoro dei materiali usati per costruire il motore. Questo costrinse, per cercare di migliorare ulteriormente le prestazioni, a porre l'attenzione anche su altri fattori, come il BPR . Infatti, ci si accorse che aumentando questo rapporto (specifico di un turbofan), si otteneva un importante aumento dell'efficienza complessiva del propulsore.

A partire dagli anni '80 si manifestarono i principali problemi legati agli alti valori (> 10) di questo parametro. Infatti, all'aumentare del BPR , e quindi al crescere della portata di massa complessiva ingerita dal motore, si ebbe un sensibile aumento dell'area frontale del motore stesso. Mentre inizialmente i motori erano inclusi nella fusoliera o nelle ali degli aeroplani, si rese necessario, per questo effetto, il passaggio alla tipica configurazione odierna, ossia con i motori inseriti in gondole posizionate sotto le ali degli aeroplani. [3]

Fino ad una decina di anni fa si riteneva che il massimo valore per questo parametro fosse circa 12, infatti grazie ad analisi monodimensionali sul ciclo termodinamico del motore sembrava che un ulteriore aumento del BPR causasse un peggioramento dell'efficienza complessiva. Grazie all'avvento dei codici CFD, invece, ci si rese conto man mano come questa convinzione non fosse completamente corretta. Difatti, nelle precedenti analisi monodimensionali non potevano essere inclusi gli effetti di interferenza tra la nacelle e l'ala dell'aereo, effetti però che, specie alle velocità di crociera odierne, sono tutt'altro che trascurabili.

Questi nuovi strumenti spinsero la NASA ad intraprendere una serie di workshop finalizzati al potenziamento delle tecniche CFD che portarono, nel 2008, allo sviluppo di un aereo, il Common Research Model CRM , che potesse essere un punto di riferimento per lo studio computazionale fluidodinamico aeronautico. In particolare, l'aereo è stato pensato per riprodurre i moderni modelli transonici con le dimensioni tipiche di un aeroplano di 300 posti. [4]

Lo sviluppo di questo modello, e la disponibilità dei suoi dati pubblici [5] [6], ha permesso uno studio molto più intensivo delle prestazioni aerodinamiche di un aeromobile: sia, in specifico, per quanto riguarda l'interferenza tra ala e gondola sia, più in generale, per le prestazioni fluidodinamiche dell'aereo stesso in varie condizioni. Questo è stato possibile grazie alle attività di ricerca promosse [7] [8] [9] [10] dalla NASA e la successiva divulgazione dei risultati.

Negli ultimi anni, si sono susseguiti alcuni studi specifici su questo modello, non più promossi dalla NASA. Dapprima per determinare approfonditamente gli effetti di interferenza nel CRM di una Through-Flow Nacelle TFN se la natura del drag aggiuntivo [11]. Successivamente invece, si è cercato di compiere una valutazione fluidodinamica della

nacelle coinvolgendo il sistema propulsivo, simulato attraverso la presenza di opportune Boundary Conditions BC interne [12]. Infine si è valutato l'effetto dell'interferenza con l'ala al variare della posizione della nacelle con il motore [13].

Dopo aver esaminato la nacelle del CRM dal punto di vista aerodinamico, sono nati una serie di studi per la sua ottimizzazione. In particolare, si è cercato, attraverso curve CST, di ricostruire una nacelle 2D assialsimmetrica con BC interne e successivamente di ottimizzare la sua forma, dal punto di vista fluidodinamico, attraverso algoritmi genetici [14] [15] [16] [17] [18] [19].

Questa tesi ha l'obiettivo di compiere un ulteriore passo in avanti in questo ambito così importante per il trasporto civile. In particolare, si pone lo scopo di creare un codice di calcolo per parametrizzare una nacelle assialsimmetrica, con relativo sistema propulsivo, nel caso di turbofan $UHBR$, ed eseguirne la sua ottimizzazione fluidodinamica con un algoritmo genetico multi-obiettivo. Gli aspetti innovativi sono duplici: il primo consiste nell'utilizzare varie tipologie di curve quali BEZIER, B-SPLINE o NURBS per la parametrizzazione geometrica e poter controllare le dimensioni della nacelle indipendentemente dalle curve usate (a differenza delle CST). Il secondo, invece, riguarda la simulazione fluidodinamica 3D della nacelle installata nel CRM considerando, quindi, anche gli effetti di installazione nel CRM.

In particolare, l'obiettivo dell'ottimizzatore è di minimizzare il drag aggiuntivo, causato dall'installazione della nacelle nel velivolo, in due diverse condizioni di volo:

1. $Ma = 0.8$
2. $Ma = 0.85$

1.1 CRM

È fondamentale, prima di procedere con l'esposizione della tesi, approfondire il modello di aereo su cui si basa il codice. L'aereo sviluppato dalla NASA, basandosi sui moderni modelli transonici, ha le dimensioni principali [4] riportate in Tabella 1.1.

Per rendere più agevole la comprensione delle precedenti dimensioni, le medesime sono riportate in Tabella 1.2 in millimetri.

Viene qui descritto il processo progettuale che è stato seguito per ottenere il CRM [4]. L'aereo, dovendo riprodurre i modelli ordiarni per il trasporto civile, è stato progettato considerando i parametri riportati in Tabella 1.3.

Mentre la fusoliera è stata disegnata in modo tale da riprodurre le tipiche configurazioni adottate nel trasporto civile, per l'ala si è adottata una procedura più elaborata. Il design delle ali, adatto per le condizioni riportate in Tabella 1.3, ha l'obiettivo di garantire prestazioni abbastanza elevate in un ragionevole intorno del punto di progettazione [4]. Sono state adottate le seguenti coppie di valori di (M, C_L) : $(0.85, 0.50)$, $(0.85, 0.48)$, $(0.85, 0.52)$, $(0.84, 0.50)$, e $(0.86, 0.50)$.

Si è scelto questo approccio per i due seguenti motivi: il primo riguarda gli *step climbs*. Per ridurre il carburante consumato l'aereo viene, con salti discreti, portato ad altitudini

S_{ref}	594,720.0 in^2	4,130.0 ft^2
Trap-Wing Area	576,000.0 in^2	4,000.0 ft^2
C_{ref}	275.80 in	
Span	2,313.50 in	192.8 ft
X_{ref}	1,325.90 in	
Y_{ref}	468.75 in	
Z_{ref}	177.95 in	
λ	0.275	
$\Lambda_{C/4}$	35°	
AR	9.0	

Table 1.1: Dimensioni di riferimento per il CRM

S_{ref}	38368955.2 mm^2
Trap-Wing Area	371612160 mm^2
C_{ref}	7005.32 mm
Span	58762.9 mm
X_{ref}	33677.86 mm
Y_{ref}	11906.25 mm
Z_{ref}	4519.93 mm

Table 1.2: Dimensioni di riferimento per il CRM in millimetri

Ma	0.85
Re (based on C_{ref})	$40 \cdot 10^6$
C_L	0.5
Altitude	10866 m

Table 1.3: Condizioni di volo di design per il CRM

più alte dove, per la diminuzione del carburante e quindi il minor peso, si ha il nuovo punto ottimale di volo. Un tipico trasporto commerciale sperimenterà, in queste manovre, coefficienti di portanza che variano nell'ordine di ± 0.02 rispetto al valore di crociera nominale [4]. Il secondo aspetto riguarda, invece, la tendenza tipica delle compagnie aeree a far volare i velivoli con un numero di Mach superiore a quello ottimale, bruciando fino all'1% di carburante in più [4]. Per questi motivi si è deciso un approccio di design multi-punto.

Un ulteriore requisito aerodinamico era che le configurazioni senza (WB) e con gruppo nacelle / pilone (WBNP) fossero entrambe soddisfacenti. Quindi, per garantire ciò, l'approccio adottato è quello di sviluppare prima un'ala ad alte prestazioni nella presenza solo della fusoliera, e dopo integrare il gruppo gondola più pilone senza introdurre effetti indesiderati eccessivi di installazione [4]. La nacelle aggiunta è di tipo TFN, ossia senza il motore interno. In questo modo è stato possibile valutare i corrispondenti effetti di installazione nel CRM. In Figura 1.1 sono riportati i contour di pressione, ottenuti nelle simulazioni fluidodinamiche condotte dalla NASA, per evidenziare gli effetti di interferenza [4].

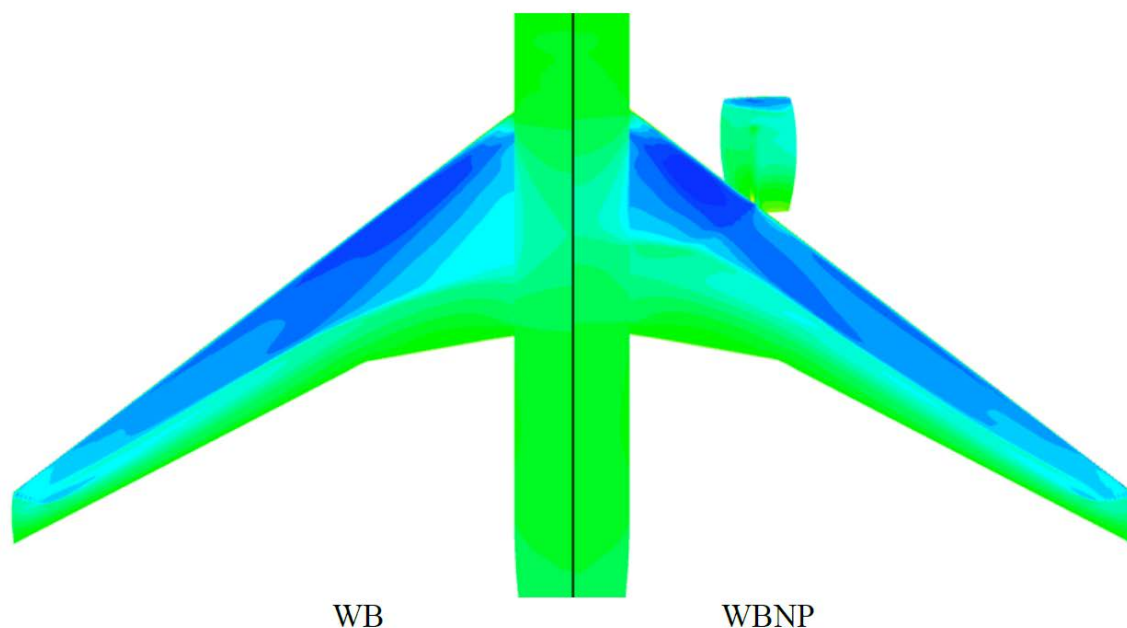


Figure 1.1: *Andamenti di pressione nel CRM in configurazione WB e WBNP*

Vengono inoltre riportati, al variare dello span, i coefficienti di pressione C_P lungo l'ala nelle due configurazioni per poter valutare l'effetto di interferenza con la nacelle installata[4]. In Figura 1.2 è possibile valutare come i principali effetti di disturbo siano concentrati nella prima metà dello span dell'ala.

I risultati riportati in queste immagini fanno però riferimento a simulazioni fluidodinamiche condotte dalla NASA. Di particolare importanza è stato l'ultimo DPW promosso dalla NASA [10]. Uno dei compiti, per i partecipanti, era la quantificazione dell'aumento del drag a causa della nacelle, con i parametri di volo in Tabella 1.4. L'aspetto fondamentale, per questa tesi, per cui è stato preso in considerazione questo workshop della NASA, riguarda però i dati sperimentali, ottenuti in galleria del vento, forniti al termine di esso [5] [6] [20].

Ma	0.85
Re	$5 \cdot 10^6$
C_L	0.5 ± 0.0001
T_{ref}	100 ° F
AoA	2.7500°

Table 1.4: *Condizioni di volo di design per il DPW 6*

Vengono riportati nelle Figure 1.3 e 1.4 , al variare dello span, i coefficienti di pressione C_P lungo l'ala nel caso WB e WBNP misurati sperimentalmente e calcolati con simulazioni fluidodinamiche [21].

Si nota, per entrambe le configurazioni, come i coefficienti di pressione nel primo 50% dello span siano ben catturati dalle simulazioni fluidodinamiche. Le differenze principali

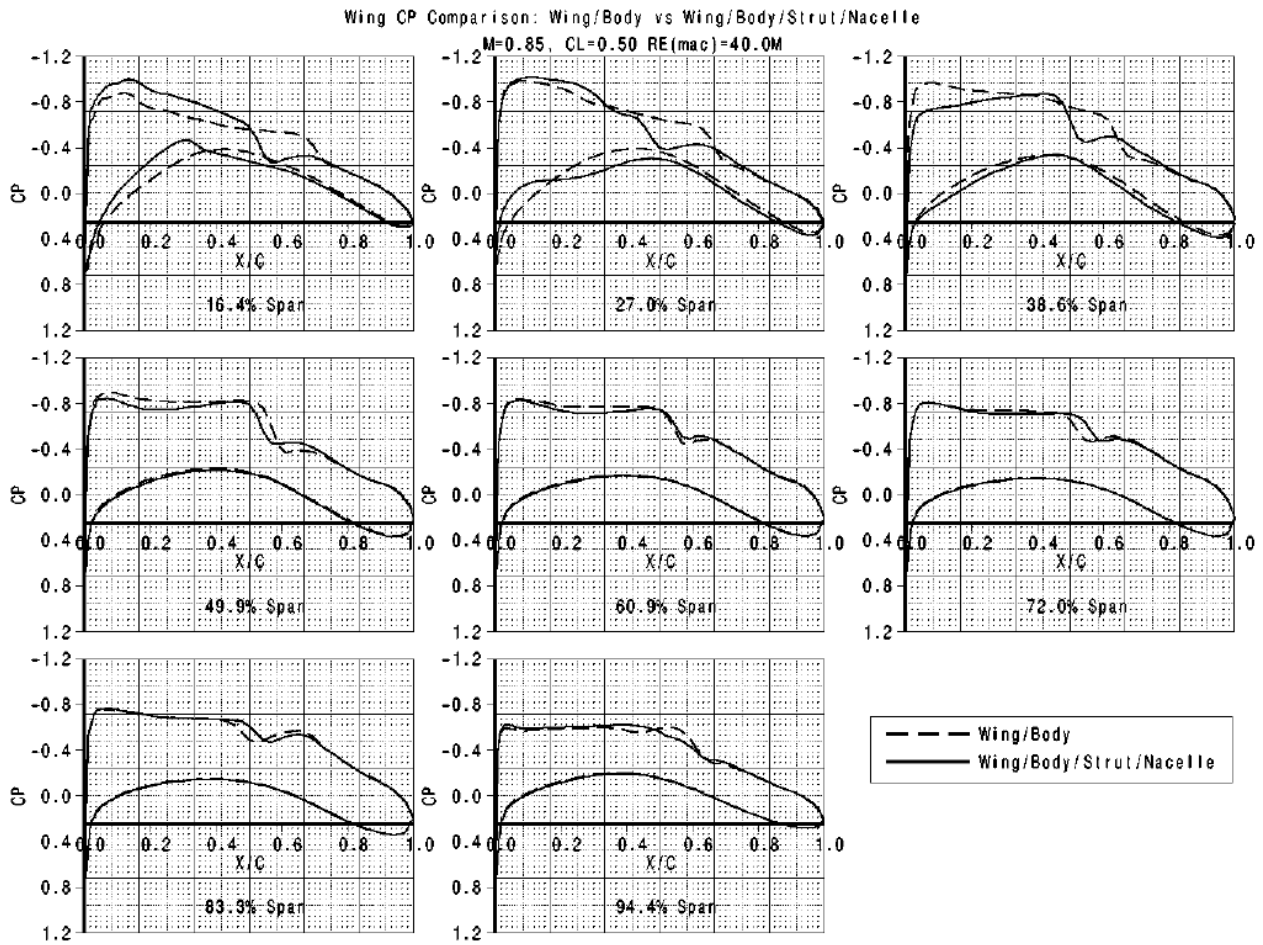


Figure 1.2: C_P calcolati con CFD per il CRM in configurazione WB e WBNP 2

nei successivi diagrammi di C_P riguardano soprattutto la posizione dello shock [21]. A conferma di quanto già notato nelle precedenti simulazioni fluidodinamiche, in Figura 1.2, è evidente l'effetto di interferenza della nacelle nella prima metà dell'ala.

Sono inoltre riportati i C_P della sola nacelle al variare dell'angolo considerato in Figura 1.5 [21].

I dati sperimentali e computazionali riportati in questa sezione sono fondamentali per poter validare il codice di calcolo che verrà descritto dal Capitolo 2, in modo da effettuare un'ottimizzazione fluidodinamica il più veritiera possibile.

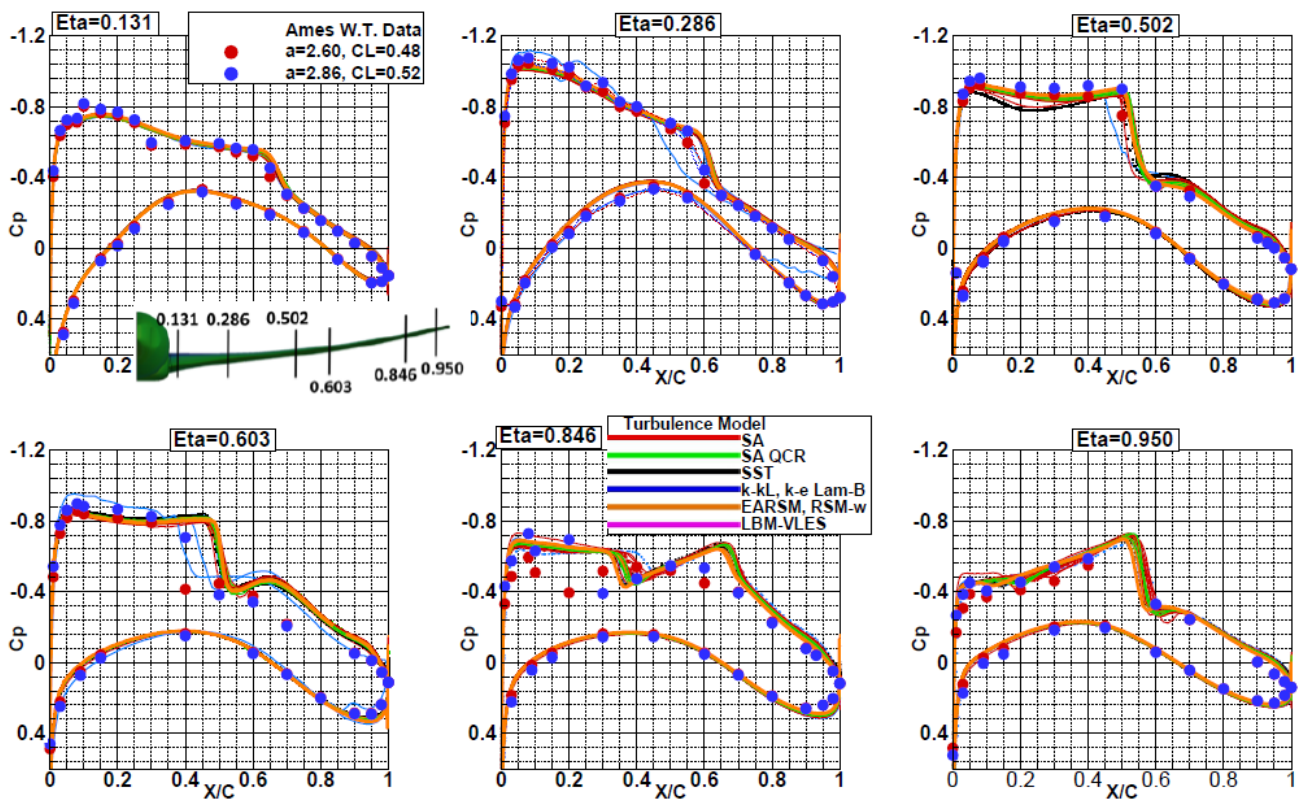


Figure 1.3: C_p testati e simulati per il CRM in configurazione WB

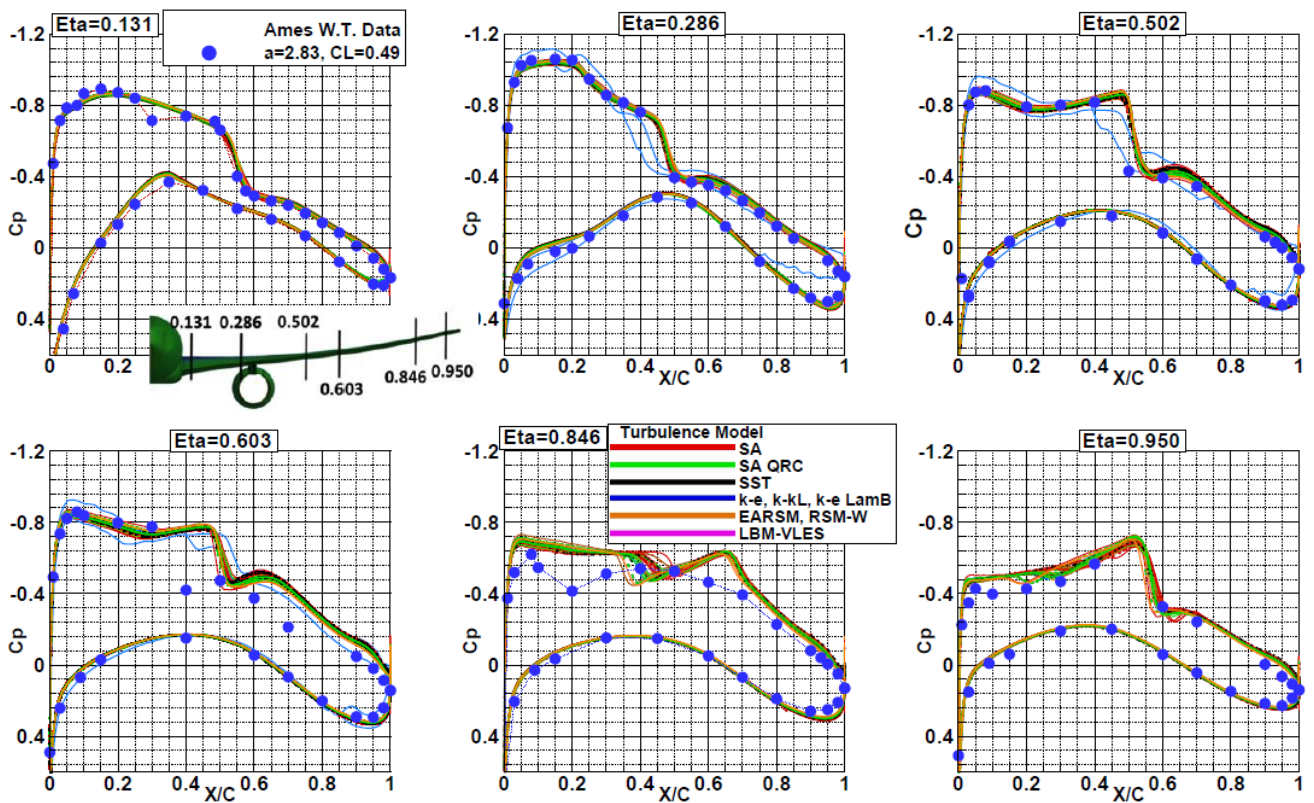


Figure 1.4: C_p testati e simulati per il CRM in configurazione WBNP

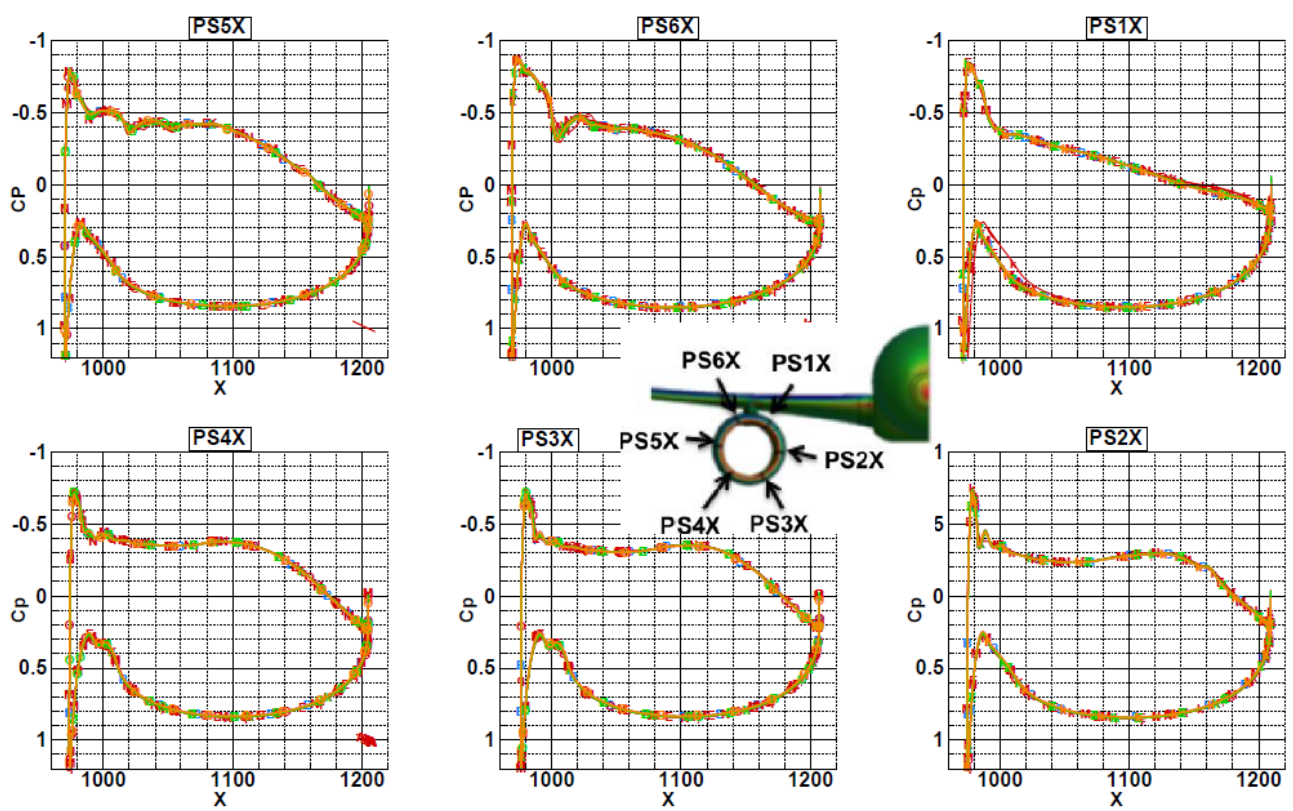


Figure 1.5: C_p della nacelle del CRM

CURVE PARAMETRICHE

2.1 INTRODUZIONE

L'obiettivo principale di una parametrizzazione è quello di cercare di rappresentare, con un numero opportuno di parametri gli elementi di un insieme. In riferimento a questa tesi, quindi, una parametrizzazione geometrica deve riuscire a rappresentare le curve delineanti la forma della nacelle. Per chiarire il significato di parametrizzazione geometrica viene di seguito riportato un caso molto semplice. L'obiettivo dell'esempio è quello di replicare, attraverso una parametrizzazione, la forma di una circonferenza unitaria centrata nell'origine, la cui equazione cartesiana è:

$$x^2 + y^2 = 1 \quad 2.1$$

Per poter definire univocamente un punto appartenente alla circonferenza in questione è necessario definire entrambe le sue coordinate, in quanto, definendo ad esempio la sola x , rimarrebbe il dubbio sul segno della corrispondente y . Le potenzialità della parametrizzazione risultano evidenti quando si introduce il parametro t ed l'Equazione 2.1 viene elaborata come:

$$x = \cos(t) \quad y = \sin(t) \quad 2.2$$

Infatti ora, per identificare univocamente un punto appartenente alla circonferenza è sufficiente scegliere un valore, del parametro t , compreso nell'intervallo $[0 ; 2\pi]$. È evidente quindi, come attraverso una semplice parametrizzazione, sia stato possibile ridurre il numero di parametri necessari per rappresentare la medesima curva.

È necessario notare però, come attraverso la nuova formulazione della circonferenza, si sia ristretto l'insieme di curve che possono essere rappresentate. Infatti, i due parametri nell'equazione 2.1 permettono di rappresentare una qualunque curva sul piano, viceversa il solo parametro t non ha la stessa flessibilità.

Gli aspetti messi in luce sono chiave in un'ottimizzazione attraverso un algoritmo genetico. Poiché, tanto maggiore sarà il numero di parametri a disposizione, tanto migliore sarà la capacità di "esplorazione" dell'algoritmo stesso. Al contrario però tanto minore sarà la capacità di "convergenza" e quindi aumenteranno i tempi di elaborazione. Vengono quindi proposte le proprietà fondamentali che deve avere una parametrizzazione, soprattutto alla luce di una successiva ottimizzazione:

- *Accuratezza*: la parametrizzazione deve riuscire a riprodurre nel modo più fedele possibile la geometria di partenza;
- *Flessibilità*: la parametrizzazione deve riuscire a generare il maggior numero possibile di geometrie diverse, garantendo quindi un'alta "esplorazione" dell'algoritmo;
- *Agilità*: la parametrizzazione deve riuscire ad avere il minor numero possibile di parametri in gioco, garantendo quindi una veloce "convergenza" al punto ottimale.

È immediato notare come i primi due obiettivi siano diametralmente opposti al terzo. Inizialmente, pertanto, si è deciso di ricercare una parametrizzazione che fosse il più generale possibile in modo da soddisfare al meglio i primi due requisiti. Successivamente, per ottenere un buon compromesso tra tutte e tre le proprietà, si è proceduto vincolando la formulazione matematica ottenuta.

2.2 CONFRONTO CURVE PARAMETRICHE

Le curve parametriche che meglio si adattano alla richiesta di alta flessibilità sono state individuate nelle *NURBS* (Non Uniform Rational B-Spline). Viene qui di seguito proposto un confronto tra le principali famiglie di curve parametriche per giustificare questa affermazione.

2.2.1 BÉZIER

Le curve parametriche più semplici valutate sono le Bézier (sulle quali sono costruite le curve CST) che sono definite come [22]:

$$C(u) = \sum_{i=0}^n B_{i,n}(u)P_i \quad 2.3$$

Dove P_i è il generico punto di controllo della curva, appartenente a P matrice dei punti di controllo $[P_0, P_1, \dots, P_{n-1}, P_n]$, e $B_{i,n}$ è la generica funzione base con cui è costruita la curva parametrica. Per questa tipologia di curve, le funzioni base utilizzate sono i polinomi di Bernstein di grado massimo pari a n . La variabile parametrica della curva è u ed appartiene all'intervallo $0 \leq u \leq 1$.

Viene riportata in Figura 2.1 la curva di Bézier ottenuta con un poligono di controllo a gradino.

Il vantaggio di utilizzare queste curve è che il numero di parametri necessario per controllarle è strettamente legato al numero di controllo, infatti nel caso bidimensionale i parametri sono il doppio dei punti di controllo, in quanto ogni punto di controllo ha due coordinate. Inoltre, a differenza delle successive curve, dato un poligono di controllo,

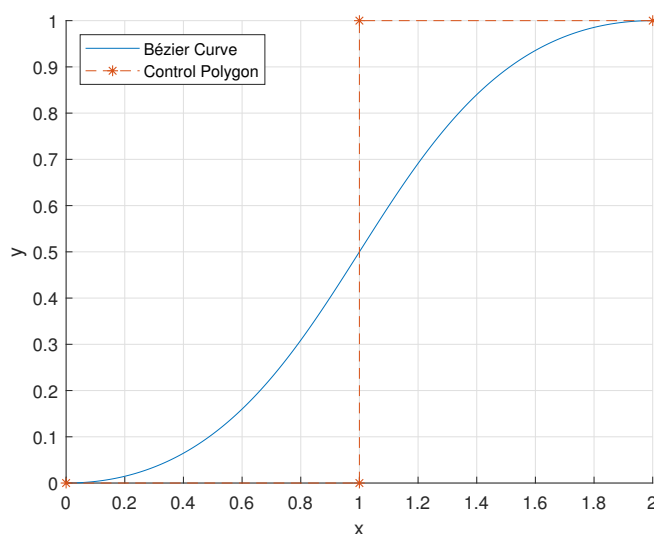


Figure 2.1: Esempio di curva di Bézier con poligono di controllo a gradino

esiste una e una sola curva generabile. Il grado p delle curve di Bézier è fortemente legato al numero di punti n di controllo, e viene espresso dall'Equazione 2.4.

$$p = n - 1 \tag{2.4}$$

2.2.2 BSPLINE

Le successive curve valutate sono le B-Spline, le quali sono definite come [22]:

$$C(u) = \sum_{i=0}^n N_{i,p}(u)P_i \tag{2.5}$$

Dove P_i è il generico punto di controllo della curva, appartenente a P matrice dei punti di controllo $[P_0, P_1, \dots, P_{n-1}, P_n]$, e $N_{i,n}$ è la generica funzione base con cui è costruita la curva parametrica. Per questa tipologia di curve, le funzioni base utilizzate sono costruite con l'algoritmo di deBoor [22].

Viene riportata in Figura 2.2 la BSpline ottenuta con il poligono di controllo a gradino usato in Figura 2.1.

È evidente, considerando la Figura 2.2, come sia possibile ottenere con questa tipologia di curve una maggiore flessibilità. Infatti, a parità di poligono di controllo ed a parità di grado della curva, cambiando il valore del nodo centrale si modifica l'intera curva. Quindi, a differenza di quanto visto nel Paragrafo 2.2.1, dato un poligono di controllo la BSpline non è univocamente determinata.

Inoltre, il grado di queste curve è definito arbitrariamente e deve soddisfare la seguente relazione:

$$1 \leq p \leq n - 1 \tag{2.6}$$

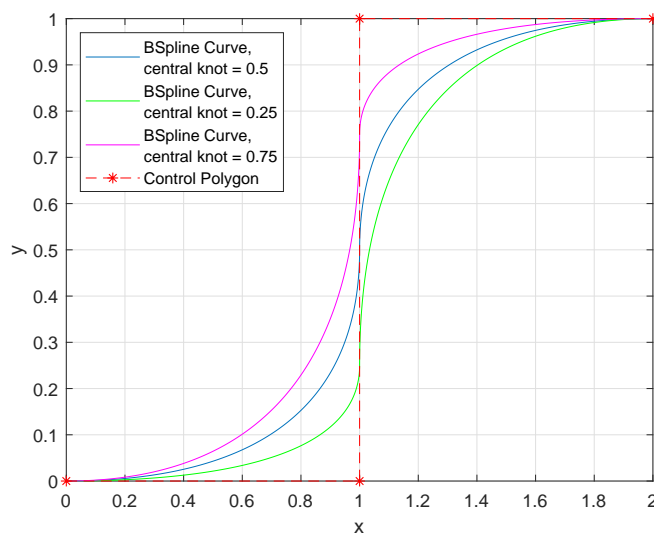


Figure 2.2: Esempio di curva di BSpline con poligono di controllo a gradino, al variare del nodo centrale

Nel caso in cui p sia 1 la curva coincide con il poligono di controllo (cosa non possibile per una curva di Bézier con $n > 2$), mentre se $p = n - 1$ la BSpline degenera in una curva Bézier.

Rispetto alle curve precedenti si ha una maggiore flessibilità grazie all'introduzione di ulteriori parametri: i nodi. Infatti, oltre ad i parametri in comune con le curve di Bézier, risultano fondamentali altri M nodi, dove M è calcolato secondo l'Equazione 2.7, per poter tracciare univocamente la curva.

$$M = n + p + 1 \quad \text{and} \quad m = M - 1 \quad 2.7$$

Il generico nodo normalizzato u_k , che appartiene a U vettore dei nodi $[u_0, u_1, \dots, u_{m-1}, u_m]$, è compreso nell'intervallo $[0 ; 1]$ ed assume importanza nel calcolo delle funzioni base $N_{i,n}$.

Il secondo principale vantaggio delle curve BSpline, rispetto alle curve precedenti, non viene messo in luce nella Figura 2.2. Infatti la modifica del nodo centrale, causa la variazione globale della curva. Questo effetto, che è tipico delle Bézier, è solamente un caso fortuito per le BSpline. Queste curve, infatti, sono dette a controllo locale, perché una modifica dei punti di controllo o una modifica dei nodi della curva, generalmente, influenza solo una parte di essa.

Viene riportata in Figura 2.3 un'estensione della curva precedente in cui sono evidenziati anche i nodi della curva. L'obiettivo, nella parte seguente, è quello di dimostrare l'effetto locale di una modifica della curva in Figura 2.3.

In Figura 2.4 viene riportato il primo caso in cui si manifesta evidentemente la modifica locale di queste curve. Nella figura sono riportate la curva di partenza ed una curva modificata in un punto del poligono di controllo. Per poter valutare correttamente l'effetto locale sono stati aggiunti anche i nodi di entrambe le curve. Per una valida comprensione è necessario considerare la natura geometrica delle BSpline e dei nodi. Infatti, queste curve sono nate come una serie di curve, una Spline, di curve di Bézier collegate tra loro. Il

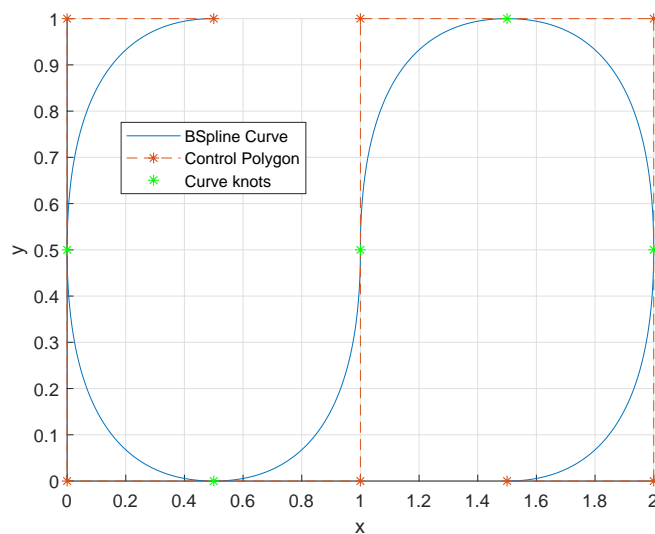


Figure 2.3: Estensione BSpline di Figura 2.2

punto di collegamento delle "sottocurve", rispetto alla curva principale, è esattamente un nodo della BSpline. Quindi ogni "sottocurva" è delimitata da due nodi attraverso i quali si collega alle porzioni adiacenti della BSpline.

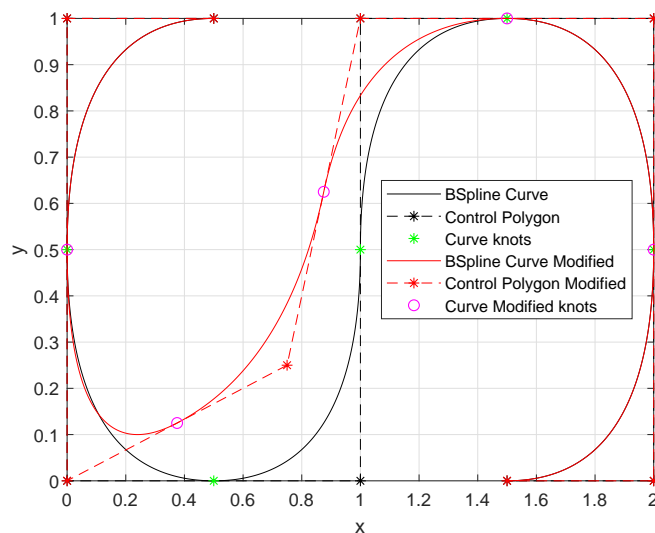


Figure 2.4: Esempio di effetto locale di una BSpline per la modifica di un punto di controllo

Si può facilmente osservare come la modifica di un punto di controllo del poligono modifichi solo in parte la BSpline di partenza. In particolare, si nota che la variazione interessa solo la "sottocurva" più vicina al punto modificato, ossia quella compresa tra i due nodi che comprendono il punto di controllo. Inoltre, la nuova curva differisce da quella originale anche per le due parti adiacenti a quella direttamente interessata dalla modifica, mentre tutte le altre porzioni della BSpline non sono affette dalla modifica. Pertanto, la modifica di un punto di controllo influenza solamente le tre "sottocurve" più vicine ad esso.

Viene riportato di seguito, in Figura 2.5, un ulteriore caso in cui è evidente il secondo caso in cui si manifesta la modifica locale di queste curve. Al contrario, ora è stato mantenuto costante il poligono di controllo ma è stata variata la distribuzione dei nodi della curva di partenza.

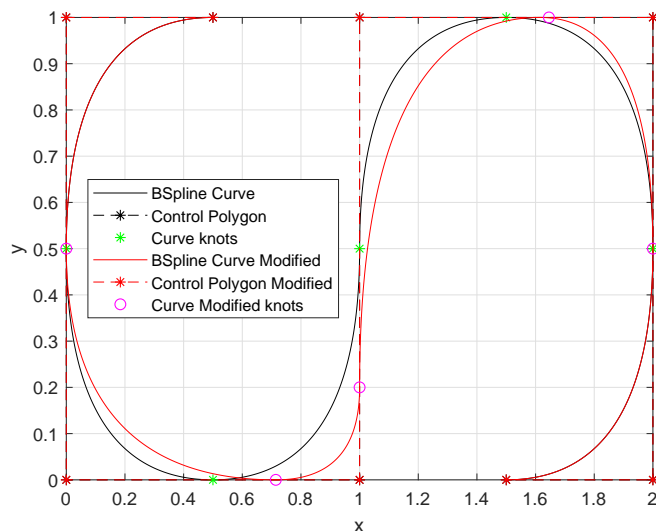


Figure 2.5: Esempio di effetto locale di una BSpline per la modifica di un nodo centrale

Si può, anche in questa seconda valutazione, osservare come la modifica del nodo centrale influenzi solo una parte della BSpline. In particolare, in questo caso, si nota che la variazione influenza le due parti che sono delimitate dal nodo modificato ed anche per le due "sottocurve" adiacenti a queste, mentre tutte le altre porzioni non sono influenzate dalla variazione. Pertanto la modifica di un nodo influenza le quattro "sottocurve" più vicine al nodo stesso.

2.2.3 NURBS

Il limite principale della precedente tipologia di curve è quello di non riuscire a rappresentare le coniche. Considerando, ad esempio, un arco della circonferenza rappresentato dall'Equazione 2.1 e riportato nella Figura 2.6, si evidenzia come la BSpline non riesca a riprodurre fedelmente la circonferenza (rappresentata dai puntini neri) a differenza della NURBS.

Le curve NURBS vengono ottenute attraverso le seguenti relazioni [22]:

$$C(u) = \sum_{i=0}^n R_{i,n}(u)P_i \tag{2.8}$$

$$R_{i,n}(u) = \frac{N_{i,p}(u)w_i}{\sum_{j=0}^n N_{j,p}(u)w_j} \tag{2.9}$$

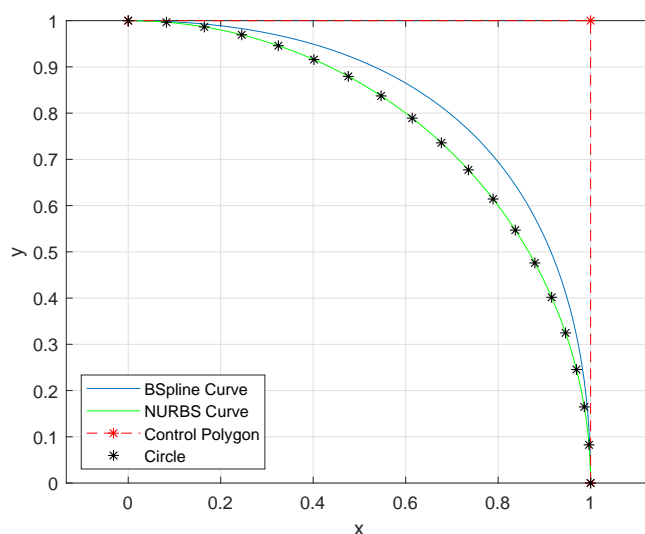


Figure 2.6: Confronto curve parametriche con un circonferenza

Queste curve, come le BSpline, si basano su punti di controllo e nodi, ciò che viene cambiato sono le funzioni base $R_{i,p}$. Infatti, la singola funzione base è costruita come una media pesata delle funzioni base $N_{i,p}$ della corrispondente BSpline. I pesi adottati w_i , appartenenti al vettore dei pesi $[w_0, w_1, \dots, w_{n-1}, w_n]$, sono generalmente compresi nell'intervallo $[0; 1]$ e sono in numero pari ai punti di controllo utilizzati. Nel caso in cui tutti i pesi siano assunti con lo stesso valore una curva NURBS si riduce ad una BSpline e, ulteriormente, se è verificata l'Equazione 2.4 la NURBS si riduce ad una Bézier. Le NURBS non sono quindi altro che l'estensione delle curve precedenti ma che consentono la massima flessibilità.

Viene proposto in Tabella 2.1 un confronto, nel caso più generale possibile, che evidenzia il numero di parametri necessari per la generazione di ogni curva parametrica. Viene inoltre riportato il relativo rapporto con i parametri necessari per generare una Bézier, ossia la curva che ne richiede il minor numero possibile.

	Bézier	BSpline	NURBS
Coordinate x	n	n	n
Coordinate y	n	n	n
Nodi	0	n+p+1	n+p+1
Pesi	0	0	n
Parametri	2n	3n+p+1	4n+p+1
Rapporto Parametri con Bézier	1	$\approx \frac{3}{2}$	≈ 2

Table 2.1: Riassunto parametri necessari per generazione di una curva 2D

Per quanto esposto nei precedenti paragrafi è evidente come le curve NURBS garantiscano la maggiore flessibilità possibile a cui è accoppiata, generalmente, la migliore ac-

curatezza nel riprodurre una curva data. La controindicazione principale però è legata all'accettare più del doppio del numero di parametri di una Bézier, aspetto di fondamentale importanza in un'ottimizzazione con un algoritmo genetico.

2.3 VINCOLI CURVE PARAMETRICHE

Diventa ora interessante valutare quali parametri di ogni curva siano vincolati da condizioni sulle curve e quali invece siano liberi e quindi a disposizione dell'ottimizzatore, ossia iniziare a valutare gli aspetti che riguardano l'*agilità* di una curva parametrica in questo contesto.

2.3.1 PUNTI ESTREMALI

Nella parametrizzazione di una curva $f(x,y)=0$ viene richiesto, solitamente, che la curva parametrica $C(u)=0$ garantisca il passaggio per il punto iniziale e finale della curva di partenza. Questa, apparentemente semplice, richiesta comporta una significativa limitazione per i parametri che definiscono la curva parametrica. Il seguente ragionamento viene proposto per una curva NURBS in quanto, essendo la più generale, comprende anche le altre tipologie più semplici.

Per garantire che la curva NURBS, di grado p , passi per il punto iniziale e punto finale della curva $f(x,y)=0$ sono necessarie le seguenti condizioni:

1. Il primo punto di controllo della NURBS, P_0 , deve coincidere con il punto iniziale della curva $f(x,y)=0$;
2. L'ultimo punto di controllo della NURBS, P_n , deve coincidere con il punto finale della curva $f(x,y)=0$;
3. I primi $p+1$ nodi, $[u_0, u_1, \dots, u_p]$, devono avere valore normalizzato pari a 0;
4. Gli ultimi $p+1$ nodi, $[u_{m-p}, u_{m-p+1}, \dots, u_m]$, devono avere valore normalizzato pari a 1;
5. Il primo peso, w_0 , deve essere diverso da 0;
6. L'ultimo peso, w_n , deve essere diverso da 0;

I nodi ripetuti $p+1$ volte consentono di avere gli estremi della curva coincidenti con gli estremi del poligono del controllo. In caso contrario la curva, che verrebbe comunque generata, avrebbe i punti estremali di posizione incognita a priori. Vista questa imposizione, il numero di nodi effettivamente liberi di variare durante l'ottimizzazione è:

$$M_{free} = n - p - 1 \quad 2.10$$

Sebbene gli ultimi due vincoli siano meno "stringenti", in quanto non determinano il valore preciso per una variabile, le condizioni precedentemente esposte limitano il numero

\	Bézier	BSpline	NURBS
Coordinate x	n-2	n-2	n-2
Coordinate y	n-2	n-2	n-2
Nodi	0	n-p-1	n-p-1
Pesi	0	0	n
Parametri	2n-4	3n-p-5	4n-p-5

Table 2.2: Riassunto parametri necessari per una curva 2D con passaggio per i punti estremali

delle variabili sensibilmente. Si può quindi considerare il numero di parametri effettivamente liberi, ottenendo la situazione riportata in Tabella 2.2.

Per rendere più evidenti i rapporti tra le variabili necessarie, viene riportato in Tabella 2.3 un caso pratico. Considerando di voler ottenere una generica curva con $n = 5$ e $p = 3$ ($p = 4$ imposto dall'Equazione 2.4 per la Bézier), si ottiene la seguente situazione riguardo i parametri liberi:

	Bézier	BSpline	NURBS
Parametri	6	7	12
Rapporto Parametri con Bézier	1	≈ 1	2

Table 2.3: Riassunto parametri necessari per generazione di una curva 2D con passaggio per i punti estremali, caso pratico con n e p fissati

É immediato notare come rispetto al caso generale, in Tabella 2.1, i rapporti tra il numero di parametri liberi in un'ottimizzazione tra le varie curve è sostanzialmente costante, eccetto per le curve BSpline. Questa tipologia di curve risulta essere particolarmente favorita dall'imposizione dei vincoli rispetto al caso generale.

2.3.2 NODI CENTRALI

Per quanto riguarda gli M_{free} nodi centrali, il cui numero è espresso dall'equazione 2.10, sono parametri liberi. Generalmente, soprattutto nelle BSpline, si tende a equispaziare i nodi centrali tra $[0 ; 1]$ e il loro valore viene mantenuto costante. Da questo punto in avanti della tesi verranno quindi suddivise due tipologie di BSpline:

- *UBS*: Uniform BSpline con i nodi equispaziati e fissi;
- *BS*: BSpline generiche con i nodi centrali liberi.

Osservando le Figure 2.2 e 2.5 è possibile capire l'influenza della posizione di un nodo centrale. Fissare quindi la posizione dei nodi di una curva ha i seguenti effetti:

- Rendere la suddivisione tra le parti fissa;
- Rendere la curva modificabile dai soli punti di controllo.

In particolare quindi, adottando una curva UBS, si rinuncia in parte alla flessibilità per ridurre (seppur di poco) il numero di parametri coinvolti.

2.3.3 VINCOLI DI TANGENZA

Nella parametrizzazione di una curva $f(x,y)=0$, può essere richiesto che la curva parametrica $C(u)=0$ garantisca di mantenere la tangente iniziale e finale uguale a quelle della curva di partenza. La trattazione per il calcolo della tangente ad una curva NURBS nei punti estremali viene riportata in Appendice A, vengono riportate qui le relazioni ottenute. Le Equazioni 2.11 e 2.12 sono fondamentali in quanto permettono di controllare le tangenti estremali della curva a priori, grazie ad un controllo dei suoi parametri.

$$y'_0 = \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \quad 2.11$$

$$y'_1 = \frac{P_{n,y} - P_{n-1,y}}{P_{n,x} - P_{n-1,x}} \quad 2.12$$

Entrambe le relazioni sono legate quindi a quattro parametri e, scelto il valore della tangente, permettono di ottenerne uno fissando gli altri tre. Adottando contemporaneamente entrambe le condizioni può quindi essere ridotto di due il numero delle variabili libere. Queste relazioni sono particolarmente utili in un'ottimizzazione anche per i seguenti aspetti:

- Sono indipendenti dalla tipologia della curva adottata;
- Sono lineari, dopo aver stabilito il valore richiesto della tangente.

2.3.4 VINCOLI DI CURVATURA

Nella parametrizzazione di una curva $f(x,y)=0$, può essere richiesto che la curva parametrica $C(u)=0$ garantisca di mantenere la curvatura iniziale e finale uguale a quelle della curva di partenza. La trattazione per il calcolo della curvatura di una curva NURBS nei punti estremali viene riportata in Appendice B, vengono riportate qui le relazioni ottenute. Le Equazioni 2.13 e 2.14 sono fondamentali in quanto permettono di controllare le tangenti estremali della curva a priori, grazie ad un controllo dei suoi parametri.

$$y''_0 = \frac{w_2 \cdot w_0}{w_1^2} \cdot \frac{p-1}{p} \cdot \frac{u_{p+1}}{u_{p+2}} \cdot \frac{P_{2,x} - P_{0,x}}{(P_{1,x} - P_{0,x})^2} \cdot \left(\frac{P_{2,y} - P_{0,y}}{P_{2,x} - P_{0,x}} - \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \right) \quad 2.13$$

$$y''_1 = \frac{w_{n-2} \cdot w_n}{w_{n-1}^2} \cdot \frac{p-1}{p} \cdot \frac{1-u_{m-p-1}}{1-u_{m-p-2}} \cdot \frac{P_{n-2,x} - P_{n,x}}{(P_{n-1,x} - P_{n,x})^2} \cdot \left(\frac{P_{n-2,y} - P_{n,y}}{P_{n-2,x} - P_{0,x}} - \frac{P_{n-1,y} - P_{n,y}}{P_{n-1,x} - P_{n,x}} \right) \quad 2.14$$

Adottando contemporaneamente entrambe le condizioni può quindi essere ridotto ulteriormente di due il numero delle variabili libere ma, rispetto a quanto esposto nel Paragrafo 2.3.3, queste relazioni sono particolarmente complesse, solitamente non lineari e fortemente dipendenti dalla curva usata (le relazioni ad hoc per ogni curva sono riportate in Appendice B).

PARAMETRIZZAZIONE

Dopo aver approfondito quanto necessario le varie curve parametriche coinvolte in questa tesi, in questo capitolo verranno discusse le scelte che sono state eseguite per riuscire a gestirle parametricamente:

- la forma del profilo 2D della nacelle assialsimmetrica;
- le dimensioni del profilo 2D della nacelle;
- la posizione rispetto al bordo d'attacco dell'ala dell'aereo (rappresentata dalla linea puntinata in Figura 3.1) della nacelle.

La geometria di riferimento da cui si è partiti viene proposta nella Figura 3.1.

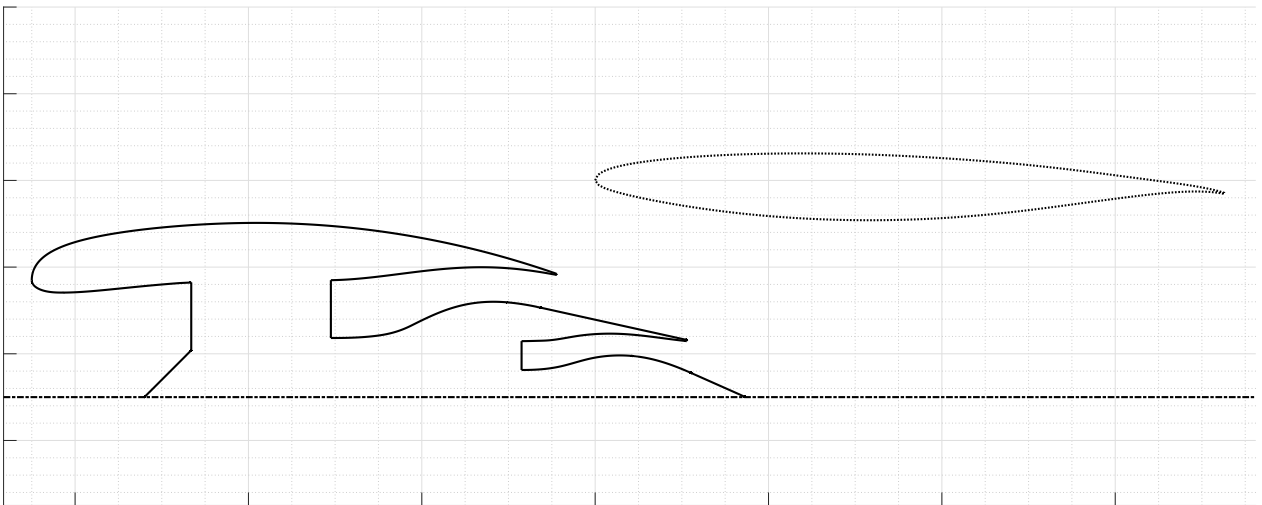


Figure 3.1: *Profilo 2D di riferimento*

Vista la flessibilità che una curva NURBS è in grado di garantire potrebbe sorgere l'idea di parametrizzare l'intera geometria da ottimizzare con un'unica curva. Il problema principale di questa soluzione sarebbe legato al vincolo di adottare un unico grado per la curva. Questo impedirebbe quindi di realizzare in modo semplice i tratti rettilinei ove necessari, o alternativamente non si riuscirebbe ad avere tratti curvi. Essendo inoltre necessario controllare anche dimensioni interne alla geometria presentata, si è reso necessario suddividere in diverse porzioni il profilo 2D rappresentato in Figura 3.1 e ad ogni parte

assegnare una curva parametrica differente.

La suddivisione scelta e la corrispondente nomenclatura sono riportate nella Figura 3.2.

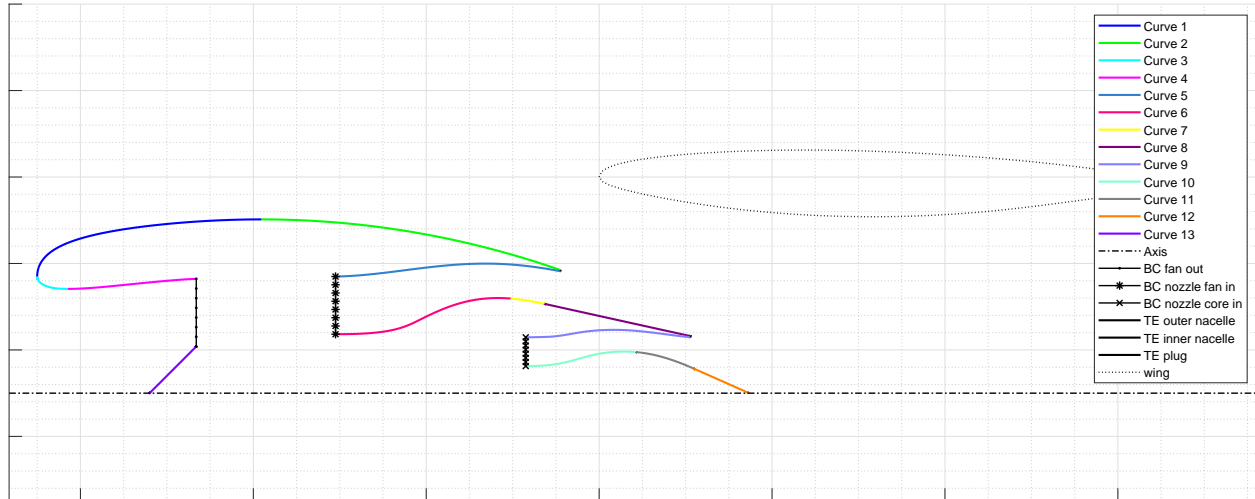


Figure 3.2: *Profilo 2D di riferimento suddiviso nelle varie parti*

Le porzioni sono state suddivise nella modalità riportata qui sotto:

- *Curve 1* : A partire dal bordo d'attacco della nacelle fino al punto di diametro esterno massimo;
- *Curve 2* : A partire dal punto di diametro esterno massimo al Trailing Edge della nacelle esterna;
- *Curve 3* : A partire dal bordo d'attacco della nacelle fino al punto di gola della nacelle esterna;
- *Curve 4* : A partire dal punto di gola della nacelle esterna fino all'AIP (Piano di Interfaccia Aerodinamica con il propulsore);
- *Curve 5* : A partire dall'uscita (lato corona) del condotto freddo al Trailing Edge della nacelle esterna;
- *Curve 6* : A partire dall'uscita (lato mozzo) del condotto freddo fino al punto di gola dell'ugello del condotto freddo;
- *Curve 7* : A partire dal punto di gola dell'ugello del condotto freddo fino al punto di uscita dell'ugello supersonico del condotto freddo;
- *Curve 8* : A partire dal punto di uscita dell'ugello supersonico del condotto freddo fino al Trailing Edge della nacelle interna;
- *Curve 9* : A partire dall'uscita (lato corona) del condotto caldo fino al Trailing Edge della nacelle interna;

- *Curve 10* : A partire dall'uscita (lato mozzo) del condotto caldo fino al punto di gola dell'ugello del condotto caldo;
- *Curve 11* : A partire dal punto di gola dell'ugello del condotto caldo fino al punto di uscita dell'ugello supersonico del condotto caldo;
- *Curve 12* : A partire dal punto di uscita dell'ugello supersonico del condotto caldo fino al punto conclusivo della spina;
- *Curve 13* : A partire dal punto sull'asse dello spinner fino al punto di diametro massimo dello spinner (situato nell'AIP).

Essendo le parti 8, 12 e 13 dei tratti lineari, si è deciso di rendere disponibile, all'utilizzatore del codice, anche la retta passante per i punti estremi di una porzione per la parametrizzazione delle curve. Le linee riportate in nero nella Figura 3.2 sono parti del profilo 2D che non sono direttamente gestibili dall'utente, bensì vengono generate a posteriori sulla base delle scelte effettuate per le 13 curve elencate.

3.1 QUOTE GEOMETRICHE

Per controllare in modo più preciso le dimensioni della nacelle si è deciso di determinare i punti estremali, di ognuna delle 13 curve, attraverso una serie di quote geometriche (che verranno descritte nei prossimi paragrafi). In questo modo è stato possibile scindere il compito di modificare la forma e le dimensioni. Infatti, la singola quota geometrica permette di variare le dimensioni di riferimento della nacelle, mentre la singola curva ha l'esclusivo compito della modifica della forma del profilo 2D avendo i punti estremali fissati. Essendo necessarie ben 36 grandezze, per garantire un controllo accurato dei principali aspetti della nacelle (inclusa la sua posizione), si è deciso di suddividerle in due categorie:

- *Dimensioni globali* : Quote geometriche influenzanti direttamente tutte le curve o quasi;
- *Dimensioni locali* : Quote geometriche influenzanti direttamente una sola o al limite poche curve.

3.1.1 DIMENSIONI GLOBALI

Le dimensioni globali utilizzate nella parametrizzazione sono cinque e sono riportate nella Figura 3.3 e nell'elenco sottostante.

1. *Axial position* = X_{LE} : Distanza assiale tra il bordo d'attacco della nacelle e il bordo d'attacco dell'ala;

2. *Vertical position* = Y_{LE} : Distanza verticale tra il bordo d'attacco della nacelle e il bordo d'attacco dell'ala;
3. *Highlight radius* = R_{Hi} : Raggio di Highlight della nacelle;
4. *Fan width* = W_{Fan} : Larghezza occupata dal solo fan nella nacelle;
5. *Core width* = W_{Core} : Larghezza occupata dal solo core nella nacelle;

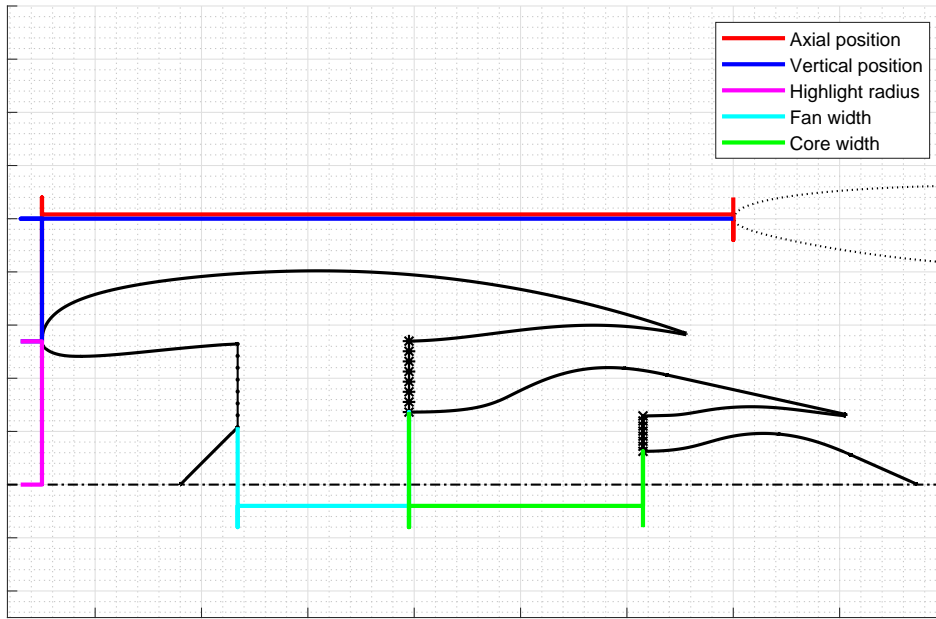


Figure 3.3: *Dimensioni globali utilizzate nella parametrizzazione*

3.1.2 DIMENSIONI LOCALI

Le 31 dimensioni locali verranno di seguito riportate per ogni curva in modo esaustivo, ma è necessario innanzitutto capire la logica con cui sono state scelte queste grandezze. L'obiettivo, che ci si è preposti, è quello di riuscire ad ottenere una parametrizzazione "progressiva". L'idea, alla base delle scelte che verranno successivamente descritte, è che ogni curva abbia il punto iniziale fissato grazie alle curve precedenti e il punto finale determinato dalle proprie dimensioni locali.

Questo approccio è particolarmente riuscito in alcune curve, mentre in altre è stato necessario introdurre informazioni riguardanti anche il punto iniziale, in quanto le curve precedenti non fornivano sufficienti informazioni riguardo allo stesso.

Nonostante ognuna delle tredici curve abbia le proprie specifiche dimensioni locali, tutte le 31 quote geometriche possono essere raggruppate in diverse 7 tipologie di grandezze, riportate qui di seguito. Ovviamente ogni curva avrà solo quelle a lei necessarie.

1. *Axial dimension* : Distanza assiale tra il punto iniziale e finale della curva;

2. *Radial dimension* : Distanza radiale tra il punto finale (o iniziale in alcune curve) della curva e l'asse della nacelle;
3. *Trailing Edge dimension* : Dimensione del bordo d'uscita in prossimità di una curva;
4. *Trailing Edge angle* : Angolo del bordo d'uscita in prossimità di una curva;
5. *Throat dimension* : Dimensione della gola del condotto freddo/caldo;
6. *Throat position* : Posizione della gola del condotto freddo/caldo;
7. *Exhaust dimension* : Dimensione dell'uscita del condotto freddo/caldo;

3.2 ASPETTI SPECIFICI DI OGNI CURVA

Sono di seguito riportate tutte curve con le relative particolarità e dimensioni locali. Per semplificare la parametrizzazione tutte le curve vengono generate in un sistema di riferimento con l'asse x coincidente con l'asse della nacelle e l'asse y sovrapposto con il raggio di Highlight, il sistema di coordinate viene riportato in Figura 3.4. In questo modo tutte le curve sono generate in un sistema di coordinate assiale / radiale e successivamente vengono traslate in blocco per posizionare la nacelle correttamente rispetto al bordo d'ingresso dell'ala.

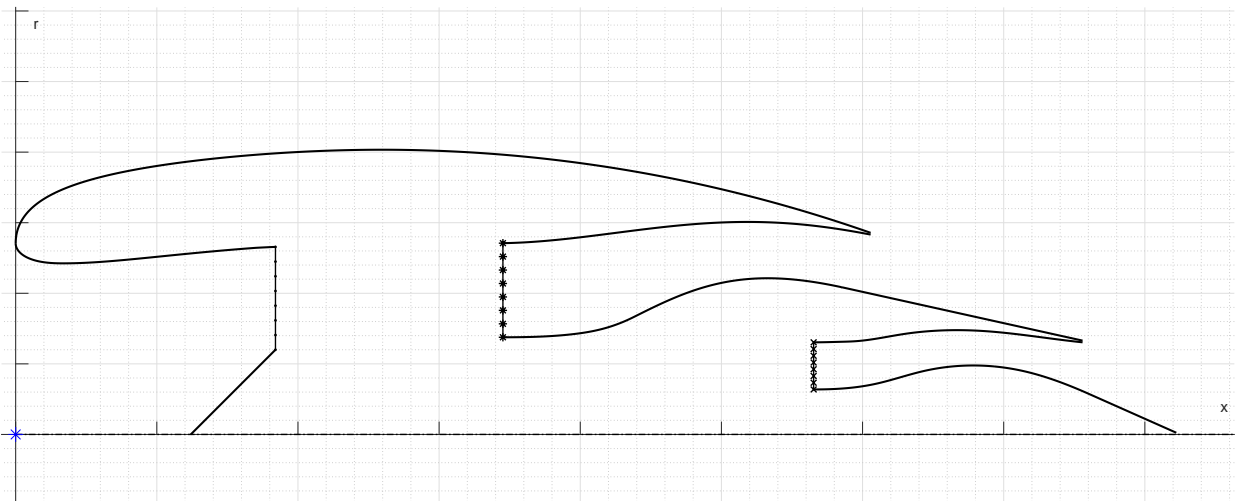


Figure 3.4: Sistema di coordinate utilizzato nella generazione del profilo 2D

3.2.1 CURVA 1

In Figura 3.5 viene riportata la curva 1 e le relative quote. Il punto iniziale della curva ha coordinate $(x,y) = (0,R_{Hi})$, determinato a partire dalle dimensioni globali, mentre il punto finale viene determinato con le quote locali:

1. *Axial dimension* = L_{pre} : Distanza assiale tra il punto iniziale e il punto a raggio massimo della nacelle;
2. *Radial dimension* = R_{max} : Raggio massimo della nacelle.

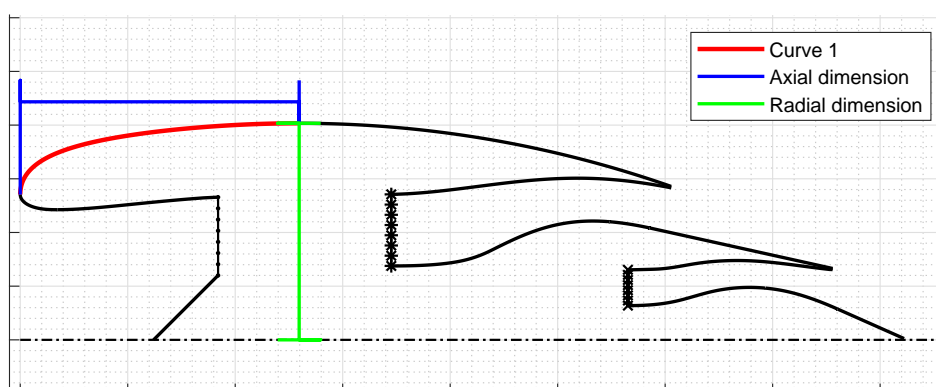


Figure 3.5: *Curva 1 e relative dimensioni locali*

Per quanto ricavato nel Paragrafo 2.3.3, è possibile imporre le condizioni di tangenza negli estremi di questa curva. Si nota in particolare che, per le scelte di parametrizzazione effettuate, la tangente iniziale y'_0 è verticale mentre la tangente finale y'_1 è orizzontale. Queste due condizioni, che poi verranno applicate anche nelle curve adiacenti, permettono di garantire la continuità di tangenza anche nel punto di connessione tra le curve 1 - 2 e 1 - 3. Le due condizioni si traducono, quindi, nelle seguenti relazioni grazie alle Equazioni 2.11 e 2.12:

- *Tangente iniziale* : $P_{0,x} = P_{1,x}$;
- *Tangente finale* : $P_{n,y} = P_{n-1,y}$.

Infine, si è deciso di non imporre alcuna condizione sulla curvatura nei punti estremi di default. Sebbene consigliato per controllare i fenomeni fluidodinamici nello strato limite, non è necessario avere continuità di curvatura tra due curve adiacenti. Questo modo operandi viene propagato anche tutte le curve seguenti per la medesima motivazione.

3.2.2 CURVA 2

In Figura 3.6 viene riportata la curva 2 e le relative quote. Il punto iniziale della curva, per quanto espresso nel Paragrafo 3.1.2, coincide con il punto finale della curva 1 mentre il punto finale viene determinato con le relative dimensioni locali:

1. *Axial dimension* = L_{post} : Distanza assiale tra il punto a raggio massimo e il punto finale della nacelle esterna;
2. *Radial dimension* = $R_{exit, fan}$: Raggio d'uscita esterno del condotto freddo.

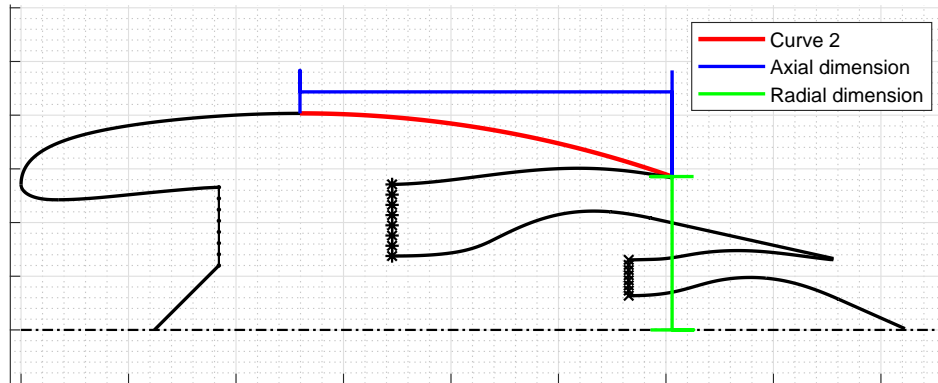


Figure 3.6: *Curva 2 e relative dimensioni locali*

Per quanto esposto nella curva precedente, per garantire la continuità della derivata prima è necessario imporre che la tangente iniziale di questa curva sia orizzontale. Per quanto riguarda la tangente finale invece, essendoci il Trailing Edge al termine della curva, non è possibile imporre a priori alcuna condizione su y'_1 . Per l'Equazione 2.11 si ottiene:

- *Tangente iniziale* : $P_{0,y} = P_{1,y}$;
- *Tangente finale* : nessuna condizione.

3.2.3 CURVA 3

In Figura 3.7 viene riportata la curva 3 e le relative quote. Il punto iniziale della curva coincide con il punto iniziale della curva 1, mentre il punto finale viene determinato con le seguenti informazioni:

1. *Axial dimension* = L_{throat} : Distanza assiale tra il punto iniziale e il punto corrispondente al raggio di gola della nacelle esterna;
2. *Radial dimension* = R_{throat} : Raggio di gola della nacelle esterna.

Le condizioni di tangenza, per questa curva, replicano in toto quelle espresse per la curva 1 e, attraverso le Equazioni 2.11 e 2.12, si ottengono le seguenti relazioni tra i punti di controllo della curva:

- *Tangente iniziale* : $P_{0,x} = P_{1,x}$;
- *Tangente finale* : $P_{n,y} = P_{n-1,y}$.

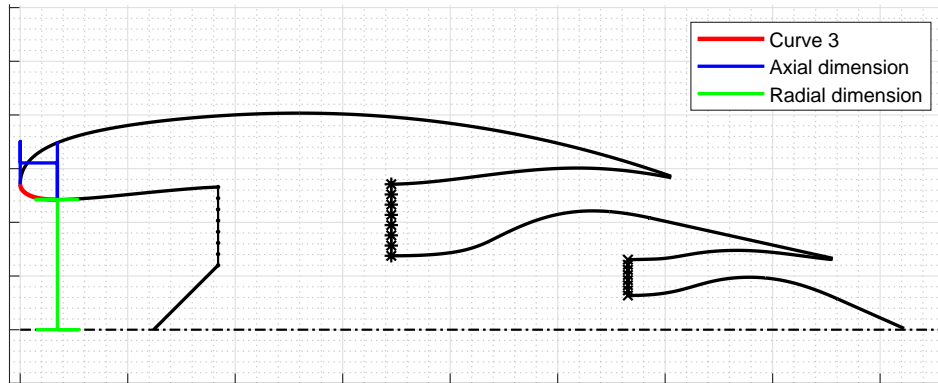


Figure 3.7: *Curva 3 e relative dimensioni locali*

3.2.4 CURVA 4

In Figura 3.8 viene riportata la curva 4 e le relative quote. Il punto iniziale della curva coincide con il punto conclusivo della curva 3, mentre il punto finale viene determinato attraverso le grandezze:

1. *Axial dimension* = L_{AIP} : Distanza assiale tra il punto di gola della nacelle esterna e la posizione assiale dell'AIP (Piano di Interfaccia Aerodinamico);
2. *Radial dimension* = R_{AIP} : Raggio di AIP.

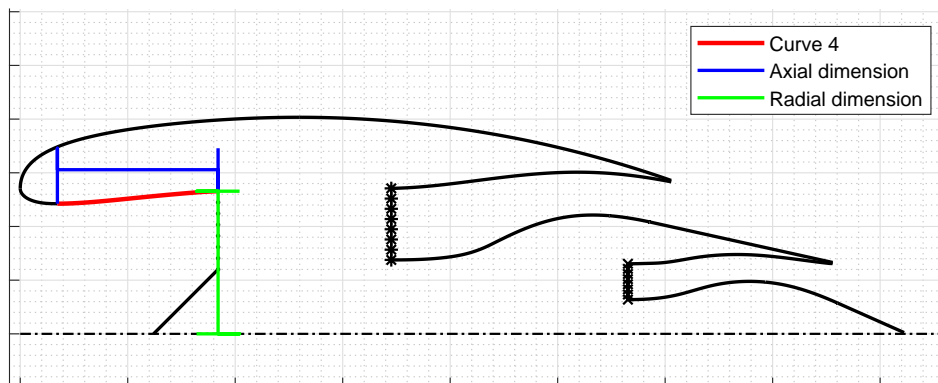


Figure 3.8: *Curva 4 e relative dimensioni locali*

Le condizioni di tangenza, per questa curva, replicano le condizioni già espresse per la curva 2 e per le Equazioni 2.11 e 2.12 quindi si traducono in:

- *Tangente iniziale* : $P_{0,y} = P_{1,y}$;
- *Tangente finale* : nessuna condizione.

3.2.5 CURVA 5

Nelle Figure 3.9 e 3.10 viene riportata la curva 5 e le relative quote locali. A differenza delle precedenti curve, il punto iniziale non può essere determinato solamente grazie alle curve precedenti. Nemmeno considerando le dimensioni globali, come per la curva 1, è possibile determinare univocamente la posizione del punto iniziale, ma solamente la sua coordinata assiale. Quest'ultima viene determinata considerando il punto finale della curva 4 a cui viene sommata la grandezza globale *fan width* W_{fan} . Per determinare, invece, la posizione radiale è necessario introdurre un'ulteriore dimensione (Figura 3.9). Inoltre, per garantire che l'impermeabilità della nacelle esterna, il punto finale della curva non viene determinato, come nelle precedenti curve, con una quota assiale e una radiale. È invece necessario, e possibile, specificare le caratteristiche del bordo d'uscita della nacelle esterna (Figura 3.10). Le dimensioni locali, per il controllo di questa curva, sono quindi:

1. *Radial dimension* = $R_{fan,shroud}$: Raggio all'uscita della BC dopo il fan, lato corona;
2. *Trailing Edge dimension* = TE_1 : Dimensione del bordo d'uscita per la nacelle esterna;
3. *Trailing Edge angle* = $TE_{1,angle}$: Angolo del bordo d'uscita per la nacelle esterna.

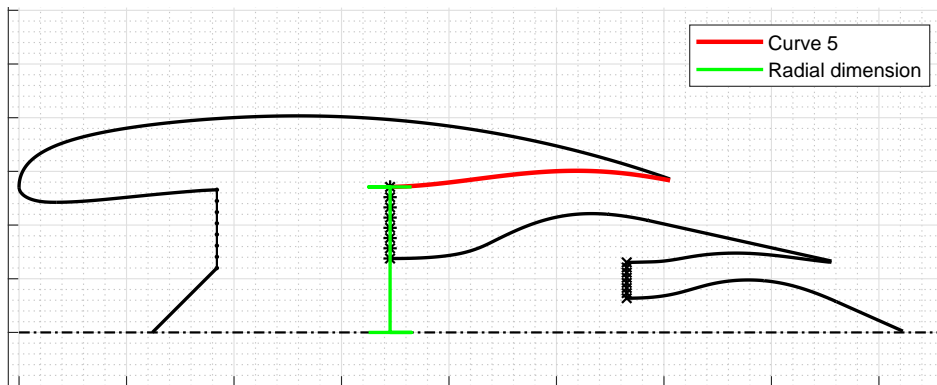


Figure 3.9: *Curva 5 e dimensioni locali relative al punto iniziale*

La convenzione adottata per la variabile $TE_{1,angle}$ è che l'angolo sia nullo se il TE è radiale, mentre un angolo antiorario, come rappresentato in Figura 3.10, è assunto positivo.

Non essendo adiacente a nessun'altra curva, per la curva 5, non è significativo e utile esprimere alcuna condizione di tangenza / curvatura per i punti estremali.

3.2.6 CURVA 6

In Figura 3.11 viene riportata la curva 6 e le relative quote. Il punto iniziale di questa curva è allineato verticalmente con il punto della curva 5, mentre per quanto riguarda la sua



Figure 3.10: *Curva 5 e dimensioni locali relative al punto finale*

quota radiale è necessario, come per la precedente, introdurre un dimensione specifica. Il punto finale della curva viene determinato in modo tale che coincida con la gola dell'ugello. Per essere sicuri che ciò avvenga non è possibile determinare la sua posizione in modo assiale e radiale. Al contrario, si fissa la posizione della gola dell'ugello nella curva 5 e perpendicolarmente alla stessa, conoscendo la dimensione della gola, si ottiene la posizione del punto finale della curva 6. Le dimensioni locali specifiche sono quindi:

1. *Radial dimension* = $R_{fan,hub}$: Raggio all'uscita della BC dopo il fan, lato mozzo;
2. *Throat dimension* = $H_{th,fan}$: Dimensione della gola del condotto freddo;
3. *Throat position* = $x_{th,fan}$: Posizione della gola del condotto freddo.

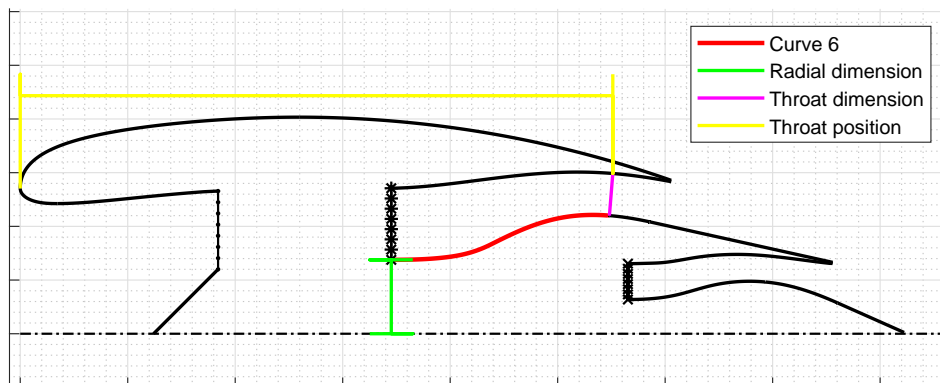


Figure 3.11: *Curva 6 e relative dimensioni locali*

Per la curva 6 non è necessario esprimere alcuna condizione riguardo la tangenza iniziale, mentre per la derivata prima nel punto finale è necessario imporre che la curva 6 sia

perpendicolare alla sezione di gola dell'ugello. In questo modo, si garantisce che la sezione considerata è perpendicolare ad entrambe le curve che la delimitano e quindi si ottiene la sezione di minima area del condotto. Per imporre il valore desiderato di y'_1 , sfruttando l'Equazione 2.12, si procede in questa modalità:

- *Tangente iniziale* : nessuna condizione;
- *Tangente finale* : $y'_1 = \frac{P_{n,y} - P_{n-1,y}}{P_{n,x} - P_{n-1,x}}$ where $y'_1 = -\frac{1}{m_{th, fan}}$.

Nella precedente relazione $m_{th, fan}$ è il coefficiente angolare della retta a cui appartiene il segmento che individua la sezione di gola nel profilo 2D della nacelle. Il coefficiente angolare in questione viene computato conoscendo i punti estremali di questo stesso segmento.

Per la curvatura non è stata imposta alcuna condizione, sebbene potrebbe essere opportuno garantire che nel punto finale della curva 6 essa sia negativa ($y''_1 < 0$), e maggiore in valore assoluto a quella della curva 5 nel punto di gola ($|y''_{1, curve_5}| < |y''_{1, curve_6}|$). L'imposizione di questo vincolo garantirebbe che non ci possano essere altri punti possibili di gola, nel condotto, nelle immediate vicinanze di quello considerato. Essendo questa una condizione complessa e che rallenterebbe notevolmente l'algoritmo, in quanto richiederebbe vincoli non lineari (Paragrafo 4.2.2) a causa delle complesse Equazioni 2.13 e 2.14, si è preferito non imporla in modo automatico.

3.2.7 CURVA 7

In Figura 3.12 viene riportata la curva 7 e le relative quote. Il punto iniziale di questa curva coincide con il punto finale della curva 6. Il punto finale, invece, viene determinato in modo tale che coincida con la sezione di uscita del condotto freddo. Per essere sicuri che ciò avvenga, a partire dal punto iniziale, si considera l'estensione assiale dell'ugello supersonico. Per determinare, poi, la posizione radiale del punto si sfrutta la dimensione della sezione di scarico che inizia nel punto finale della curva 5. Per quanto detto le quote locali necessarie sono:

1. *Axial dimension* = $L_{nozzle, fan}$: Estensione assiale dell'ugello supersonico nel condotto freddo;
2. *Exhaust dimension* = $H_{by, fan}$: Dimensione dell'uscita del condotto freddo.

Per la curva 7, per garantire la continuità, la tangente nel punto iniziale replica quanto espresso per quella finale della curva 6. Per quanto riguarda l'altro estremo della porzione 7 non si è impostata alcuna condizione vincolante. Questo, nonostante sia una buona pratica garantire che la curva 7 e la 8 siano tangenti, non è strettamente necessario.

- *Tangente iniziale* : $y'_0 = \frac{P_{0,y} - P_{1,y}}{P_{0,x} - P_{1,x}}$ where $y'_0 = -\frac{1}{m_{th, fan}}$;
- *Tangente finale* : nessuna condizione.

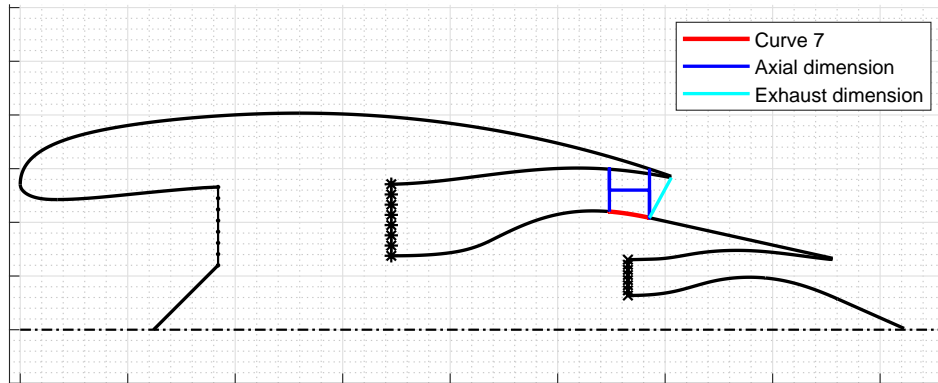


Figure 3.12: *Curva 7 e relative dimensioni locali*

Per la curvatura iniziale della curva 7 sarebbe opportuno imporla uguale a quella finale della curva 6. Essendo stato scelto però, come descritto nel paragrafo 3.2.6, di non imporre alcuna condizione sulla curvatura finale della curva 6, si replica questo approccio anche per la curva in questione.

3.2.8 CURVA 8

In Figura 3.13 viene riportata la curva 8 e le relative quote. Il punto iniziale di questa curva coincide con il punto finale della curva 7. Il punto finale della curva viene determinato, come per le prime curve, considerando la dimensione assiale della curva e il suo raggio finale.

1. *Axial dimension* = L_{cc} : Distanza assiale tra il punto di uscita del condotto freddo e il termine della nacelle interna;
2. *Radial dimension* = $R_{exit,core}$: Raggio esterno d'uscita del condotto caldo.

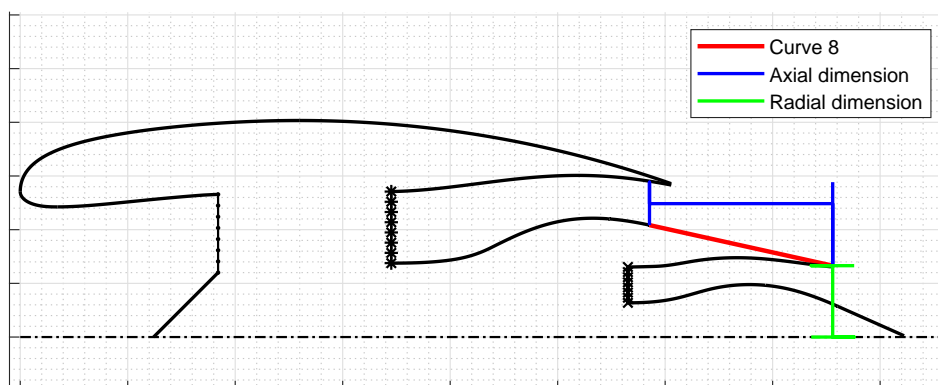


Figure 3.13: *Curva 8 e relative dimensioni locali*

Per la curva 8 non è impostata a priori nessuna condizione sulla tangenza / curvatura negli estremi della curva.

3.2.9 CURVA 9

In Figura 3.14 viene riportata la curva 9 e le relative quote. Il punto iniziale, come per la curva 5, non può essere determinato grazie alle sole curve precedenti. Per determinare la posizione assiale è necessario, infatti, considerare il punto iniziale della curva 6 a cui viene sommata la grandezza globale *core width* W_{core} . Per determinare, invece, la posizione radiale è necessario introdurre una dimensione specifica. Per garantire che l'impermeabilità, il punto finale della curva viene determinato specificando le caratteristiche del bordo d'uscita della nacelle interna. Le grandezze locali che si rendono necessarie sono quindi:

1. *Radial dimension* = $R_{core,shroud}$: Raggio all'uscita della BC dopo la turbina, lato corona;
2. *Trailing Edge dimension* = TE_2 : Dimensione del bordo d'uscita per la nacelle interna;
3. *Trailing Edge angle* = $TE_{2,angle}$: Angolo del bordo d'uscita per la nacelle interna.

In Figura 3.14 non sono riportate graficamente, per la limitata visibilità, le ultime due dimensioni. Quest'ultime, tuttavia, ripetono esattamente quanto mostrato in Figura 3.9. Inoltre sono adottate le stesse convenzioni per il valore della variabile $TE_{2,angle}$ riportate nel paragrafo 3.2.5.

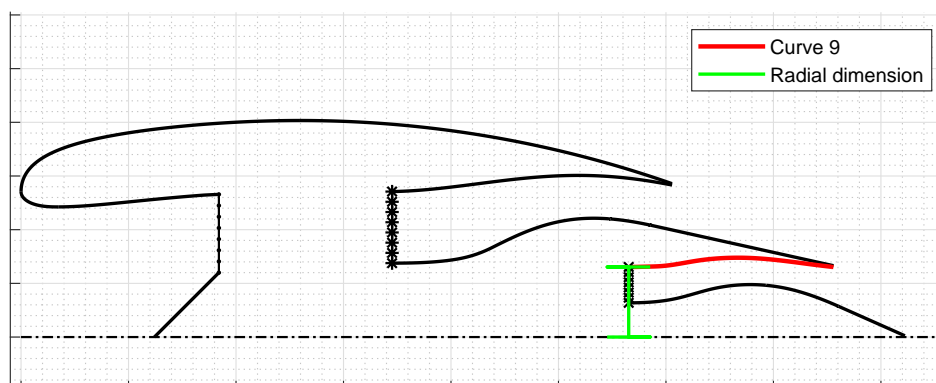


Figure 3.14: *Curva 9 e relative dimensioni locali*

Per la curva 9, non avendo curve adiacenti, non è imposta a priori nessuna condizione sulla tangenza / curvatura negli estremi della curva.

3.2.10 CURVA 10

In Figura 3.15 viene riportata la curva 10 e le relative quote. Il punto iniziale di questa curva è allineato verticalmente con il punto della curva 9, mentre per quanto riguarda la sua quota radiale è necessario introdurre la corrispondente dimensione locale. Il punto finale della curva, come per la curva 6, viene determinato in modo tale che coincida con la gola dell'ugello caldo. Per essere sicuri che ciò avvenga si determina la posizione assiale della gola nella curva 9 e perpendicolarmente ad essa, conoscendo la dimensione della gola, si ottiene la posizione del punto finale della curva 10. Le quote geometriche che vengono introdotte sono:

1. *Radial dimension* = $R_{core,hub}$: Raggio all'uscita della BC dopo la turbina, lato mozzo;
2. *Throat dimension* = $H_{th,core}$: Dimensione della gola del condotto caldo;
3. *Throat position* = $x_{th,core}$: Posizione della gola del condotto caldo.

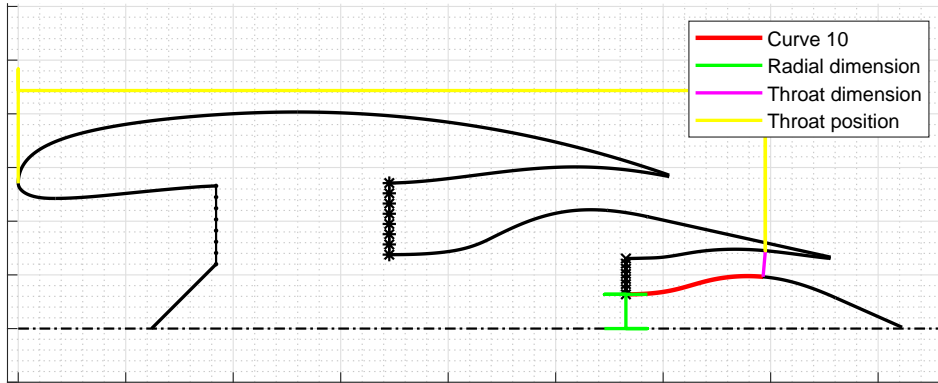


Figure 3.15: *Curva 10 e relative dimensioni locali*

Per la curva 10 ripetono quanto già espresso per la curva 6 e, sfruttando l'Equazione 2.12, si ottiene:

- *Tangente iniziale* : nessuna condizione;

- *Tangente finale* : $y'_1 = \frac{P_{n,y} - P_{n-1,y}}{P_{n,x} - P_{n-1,x}}$ where $y'_1 = -\frac{1}{m_{th,core}}$.

Nella precedente relazione $m_{th,core}$ è il coefficiente angolare della retta che comprende il segmento corrispondente alla sezione di gola. Il coefficiente viene computato conoscendo i punti estremali di questo stesso segmento.

Per la curvatura nei punti estremi della curva 10 si è deciso di adottare il medesimo approccio che per la curva 6 espresso nel Paragrafo 3.2.6.

3.2.11 CURVA 11

In Figura 3.16 viene riportata la curva 11 e le relative quote. Il punto iniziale di questa curva coincide con il punto finale della curva 10. Il punto finale della curva viene determinato in modo analogo alla curva 7 con lo scopo di garantire che sia il punto finale dell'ugello caldo. Per essere sicuri che ciò avvenga, a partire dal punto iniziale, si considera l'estensione assiale della curva e, per determinare la posizione radiale del punto, si considera la dimensione della sezione di scarico misurata a partire dal punto finale della curva 9. Per quanto detto le quote locali della curva sono:

1. *Axial dimension* = $L_{nozzle,core}$: Distanza assiale tra il punto iniziale e finale della curva;
2. *Exhaust dimension* = $H_{by,core}$: Dimensione dell'uscita del condotto caldo.

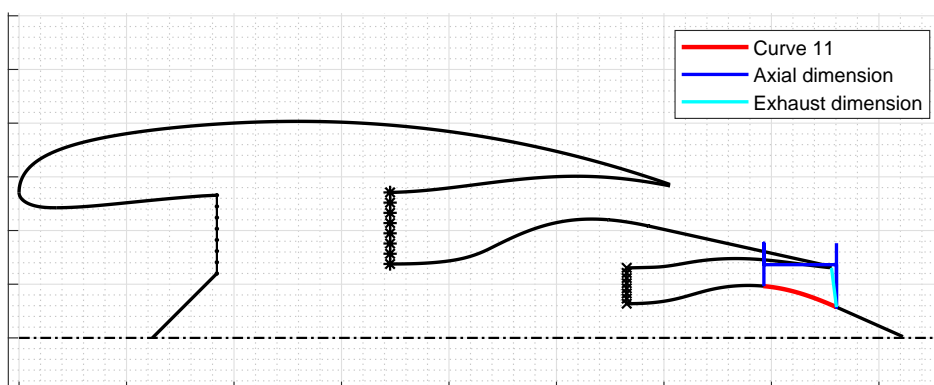


Figure 3.16: Curva 11 e relative dimensioni locali

Per le tangenti della curva 11 si adotta una metodologia analoga alla curva 7 e, sfruttando l'Equazione 2.12, si ottiene:

- *Tangente iniziale* : $y'_0 = \frac{P_{0,y} - P_{1,y}}{P_{0,x} - P_{1,x}}$ where $y'_0 = -\frac{1}{m_{th, fan}}$;

- *Tangente finale* : nessuna condizione.

Per la curvatura iniziale della curva 11, essendo adiacente alla curva 10, viene ripetuto il ragionamento effettuato nel Paragrafo 3.2.7.

3.2.12 CURVA 12

In Figura 3.17 viene riportata la curva 12 e le relative quote. Il punto iniziale di questa curva coincide con il punto finale della curva 11. Il punto finale della curva viene determinato grazie all'estensione assiale della curva ed alle caratteristiche del bordo d'uscita della spina. Pertanto le dimensioni locali necessarie sono:

1. *Axial dimension* = L_{plug} : Lunghezza assiale della spina;
2. *Trailing Edge dimension* = TE_3 : Dimensione del bordo d'uscita per la spina;
3. *Trailing Edge angle* = $TE_{3,angle}$: Angolo del bordo d'uscita per la spina.

In Figura 3.14 non sono riportate graficamente le ultime due dimensioni, per la visibilità limitata, e ripetono esattamente quanto mostrato in Figura 3.9. Inoltre sono adottate le stesse convenzioni per il valore della variabile $TE_{3,angle}$ riportate nel paragrafo 3.2.5.

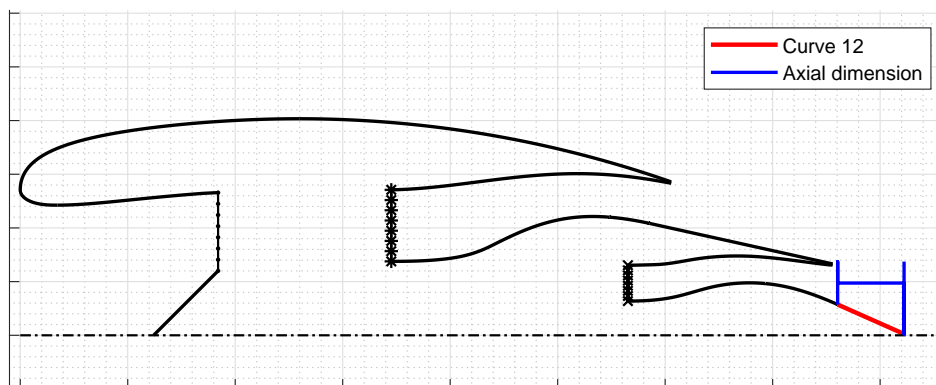


Figure 3.17: *Curva 12 e relative dimensioni locali*

Per le tangenti e la curvatura negli estremi della curva 12 non si adotta alcuna particolare condizione, nonostante potrebbe essere opportuno (ma non necessario) garantire che nel punto iniziale della curva coincidano con quelle finali della curva 11.

3.2.13 CURVA 13

In Figura 3.17 viene infine riportata la curva 13 e le relative quote. Questa curva è l'unica di tutte le curve esaminate che è completamente svincolata dalle altre, pertanto non è possibile determinate i suoi punti in modo univoco sfruttando le altre curve. I punti di questa curva sono determinati con un approccio "ibrido". Infatti per il punto iniziale della curva si conosce già, dato che è posizionato sull'asse della nacelle, che $P_{0,y} = 0$. Mentre per il punto finale si conosce la posizione assiale, infatti coincide con il punto finale della curva 4.

Sfruttando quindi le dimensioni locali della curva, che fa le veci dello spinner, è possibile ricavare le informazioni mancanti per i punti estremali:

1. *Axial dimension* = $L_{spinner}$: Lunghezza assiale dello spinner;
2. *Radial dimension* = $R_{spinner}$: Raggio massimo dello spinner.

Essendo completamente isolata la curva 13 dalle altre curve non è significativo imporre delle condizioni di tangenza / curvatura nei punti estremi.

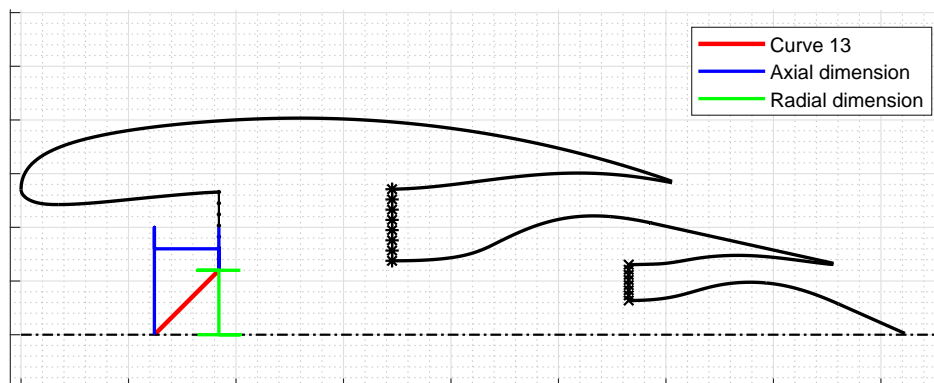


Figure 3.18: *Curva 13 e relative dimensioni locali*

3.3 COSTRUZIONE DELLA BASELINE

Visto la complessità del problema è stato necessario creare un modello di nacelle di partenza per aiutare la convergenza dell'algoritmo di ottimizzazione. Viene qui descritta la procedura con la quale è stata costruita la geometria di riferimento e le relative scelte. Si è quindi deciso di strutturare la nacelle di partenza per un turbofan UHBR con i seguenti parametri caratteristici:

- $BPR = 14$;
- $MFCR = 0.75$;
- $Ma = 0.85$.

Essendo la nacelle in questione pensata per il CRM, e dovendo successivamente interfacciarsi con il pilone dell'aereo, si è deciso di adottare alcune dimensioni della nacelle del CRM [23], riportate nella Figura 3.19. Sulla base di quanto riportato in Figura 3.19, si è deciso di adottare il diametro massimo della nacelle pari a 157.5 in o 4000 mm.

Per la scelta effettuata, si può notare la forte somiglianza tra il diametro del CRM e del Boeing 777-200ER (aereo anch'esso pensato per circa 300 posti a sedere) equipaggiato con il motore GE90-94 [24]. È emerso, però, che il turbofan GE90-94 ha un $BP = 7.8$, troppo lontano da quanto richiesto. Si è anche però notato come il motore GE58-F2 B5 fosse progettato per avere lo stesso diametro esterno e del fan, ma un rapporto di bypass molto più alto $BP = 13.1$. Si è deciso di adottare alcune dimensioni ricavate da questo modello per la nacelle in questione:

- $\frac{L_{nacelle}}{D_{max}} = 1.42 \rightarrow L_{nacelle} = 223.62 \text{ in} = 5680 \text{ mm}$;
- $D_{fan} = 123.5 \text{ in} = 3137 \text{ mm}$.
- $D_{Highlight} = 131 \text{ in} = 3327 \text{ mm}$.

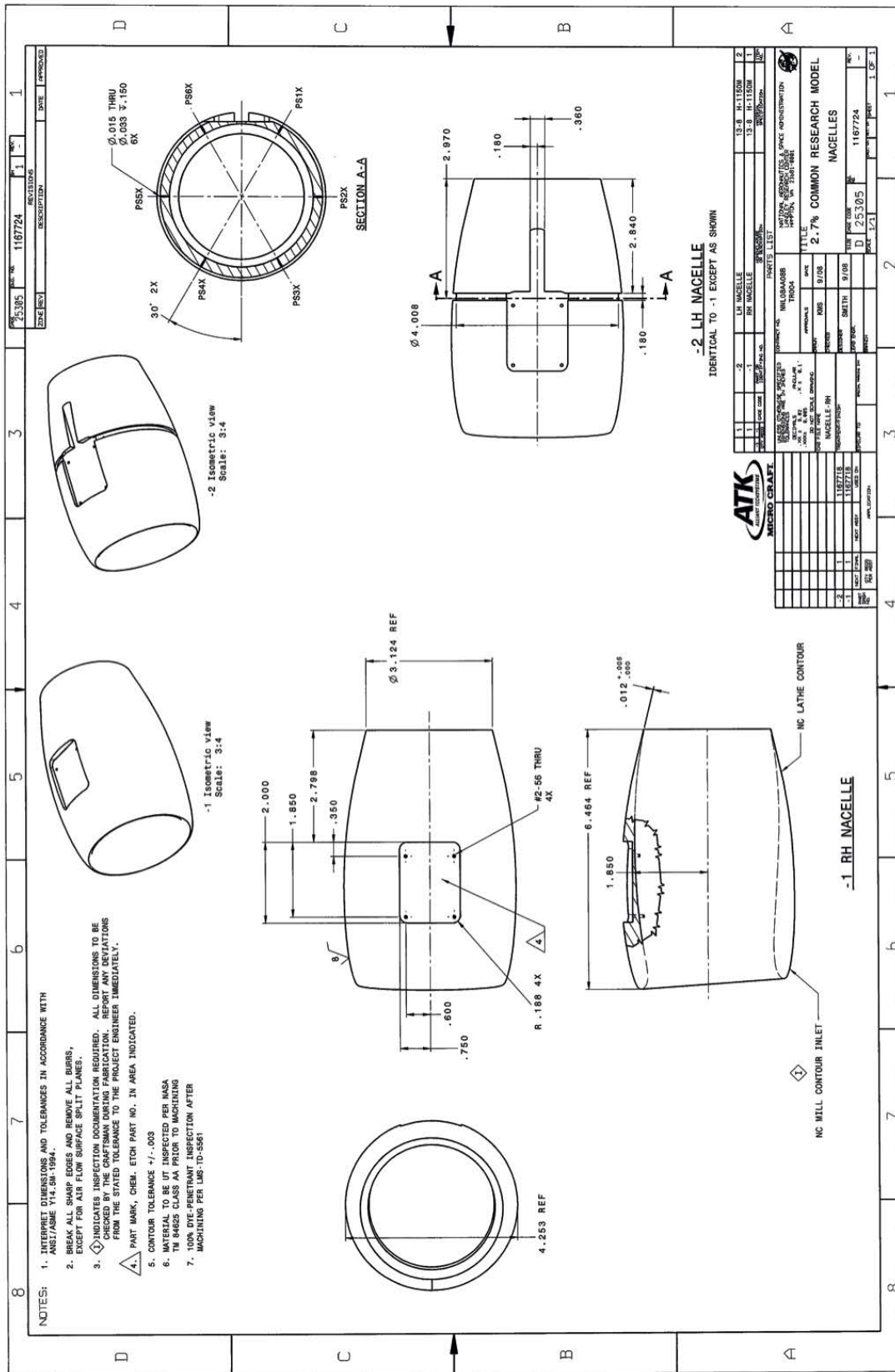


Figure 3.19: Dimensioni della TFN del CRM [23]

Inoltre è stato possibile ricavare due ulteriori informazioni dall'articolo in questione [24] riguardanti il ciclo del motore:

- $FPR = 1.45$;
- $OPR = 57$.

Considerando invece dei valori tipici per un turbofan con nacelle corta [25], è stato possibile risalire anche alle dimensioni delle parti interne, in particolare:

- $W_{fan} = 0.3 \cdot L_{nacelle} = 67.1 \text{ in} = 1704 \text{ mm}$;

Notando come le dimensioni adottate, con altri criteri, siano molto prossime a quelle di partenza della nacelle del CRM, si è deciso di ricavare anche le seguenti misure dal modello [26]:

- $D_{exit, fan} = 115.7 \text{ in} = 2939 \text{ mm}$.
- $\frac{L_{pre}}{L_{nacelle}} = 0.379 \rightarrow L_{pre} = 84.75 \text{ in} = 2153 \text{ mm}$.
- $\frac{L_{post}}{L_{nacelle}} = 0.621 \rightarrow L_{pre} = 138.87 \text{ in} = 3527 \text{ mm}$.

Grazie alle dimensioni principali della nacelle, è stato possibile risalire [13] quindi ad un suo posizionamento consigliato per il CRM:

- $X_{LE} : \sim 1.06 \cdot L_{nacelle} = 237 \text{ in} = 6020 \text{ mm}$;
- $Y_{LE} : \sim 0.23 \cdot L_{nacelle} = 51.4 \text{ in} = 1306 \text{ mm}$;

Tuttavia, nel modello proposto dalla NASA [4] è presente una TFN, quindi non è stato possibile ricavare alcuna informazione per quanto riguarda le dimensioni interne del motore. Per queste grandezze geometriche si è fatto riferimento ad una serie ulteriori informazioni [27] [28] [29] [30] [31], da cui sono state ricavate le dimensioni o i rapporti dimensionali consigliati per un turbofan con i parametri di interesse.

Avendo quindi a disposizione le dimensioni della nacelle di riferimento si è proceduto alla costruzione delle curve della stessa. A differenza di quasi tutti gli studi citati fino a questo punto, i quali hanno usato curve CST e quindi la costruzione della geometria è stata fatta in modo automatico, non è stato possibile costruire in modo univoco le curve con le sole dimensioni scelte. Si è deciso, invece di utilizzare approcci convenzionali [32], di adottare curve di nacelle già testate. Non è stato possibile però utilizzare le curve della nacelle del CRM [4], in quanto la nacelle in questione è TFN e mancherebbero le parti interne. Si è quindi deciso di utilizzare, per ricavare le curve, di sfruttare la nacelle riportata in Figura 3.20 del modello JSM [33].

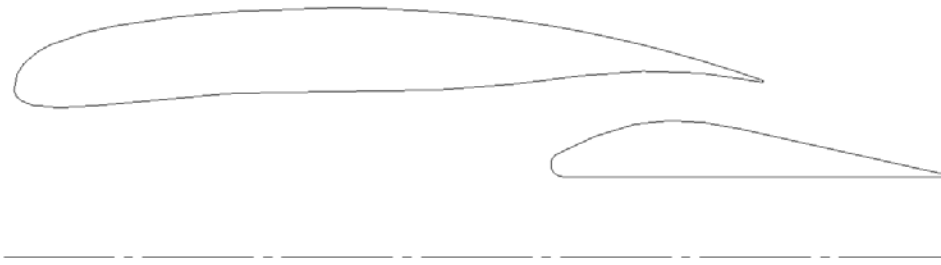


Figure 3.20: *Profilo nacelle JSM*

3.4 VALUTAZIONE SENSIBILITÀ DI OGNI CURVA

Per l'algoritmo di ottimizzazione, oltre alle 36 variabili dimensionali, verranno messe a disposizione tutte le variabili di ogni curva. Riprendendo quanto espresso in Tabella 2.2 è possibile ottenere una stima del numero ulteriore di parametri necessari. A scopo di esempio, considerando per tutte le curve i valori di $n = 5$ e $p = 3$ ($p = 4$ per Bézier), si ottengono, per la Tabella 2.3, le seguenti stime:

- *Bézier/UBS* : 78 variabili + 36 variabili dimensionali;
- *BSpline* : 91 variabili + 36 variabili dimensionali;
- *NURBS* : 156 variabili + 36 variabili dimensionali.

È evidente come il numero di variabili necessarie per riprodurre la nacelle è, al giorno d'oggi, improponibile per sostenere un'ottimizzazione efficace ed efficiente. Pertanto si è resa necessaria un'analisi di sensibilità, per ogni curva, per poter valutare il numero di variabili minimo necessario per poter ottenere comunque un risultato soddisfacente. L'analisi di sensibilità è stata condotta in successione per ogni curva con le stesse modalità, viene qui proposto lo schema e i risultati ottenuti per la curva 1.

È stato costruito un algoritmo che svolgesse le seguenti fasi automaticamente:

1. Scelta alternativa del tipo di curva: BSpline o NURBS;
2. Scelta del numero di punti di controllo n da adottare con $3 \leq n \leq 6$;
3. Scelta del grado p da adottare con $2 \leq p \leq n - 1$;
4. Calcolo delle tangenti iniziali e finali della curva di partenza;
5. Imposizione, attraverso le Equazioni 2.11 e 2.12, della tangenza alla curva parametrica;

6. Imposizione del numero di generazioni ed individui necessari all’algoritmo con le Equazioni 3.1;

$$\begin{cases} N_{individui} = 6 \cdot N_{variabili}^{1.2} \\ N_{generazioni} = 40 \end{cases} \quad 3.1$$

7. Algoritmo genetico di ottimizzazione per minimizzare l’errore di fitting.

L’errore di fitting viene valutato con l’Equazione 3.2, dove ε è l’errore locale della curva parametrica calcolato come distanza dalla curva di partenza:

$$\varepsilon_{Fitting} = \frac{\varepsilon_{mean} + \varepsilon_{max}}{2} \quad 3.2$$

Per evitare che la scelta casuale della popolazione iniziale influenzasse eccessivamente il fitting della curva si è scelto di ripetere le operazioni descritte per oltre 10 volte, la Figura 3.21 riporta il valore dell’errore minimo trovato tra tutti i tentativi per la curva 1. Le curve di Bézier, che sono rappresentate nella Figura 3.21, sono state valutate come caso particolare delle BSpline, ossia nel caso in cui fosse verificata l’Equazione 2.4.

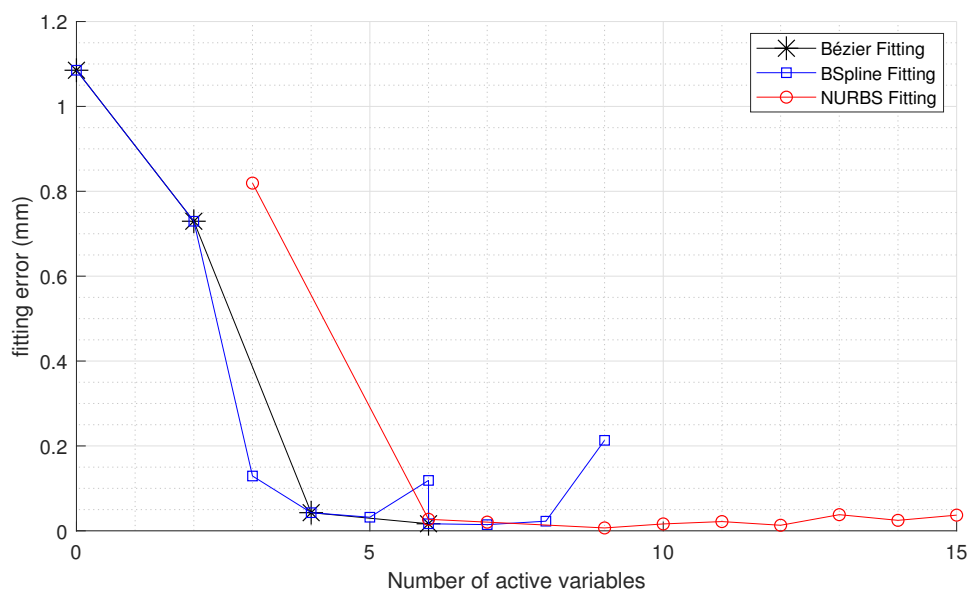


Figure 3.21: Errore di fitting per la curva 1 al variare del numero di variabili

Diventa interessante valutare i rapporti, riportati in Figura 3.22, tra gli errori di fitting e l’errore minimo. Si nota come la migliore curva assoluta per il fitting sia una NURBS (in particolare con $n = 4$ e $p = 3$) che permette di avere un errore di fitting pari alla metà delle altre curve.

Emerge molto nitidamente da questo confronto, ed i risultati ottenuti si sono ripetuti in modo molto simile anche per tutte le altre curve, che eccetto il primo tentativo (con

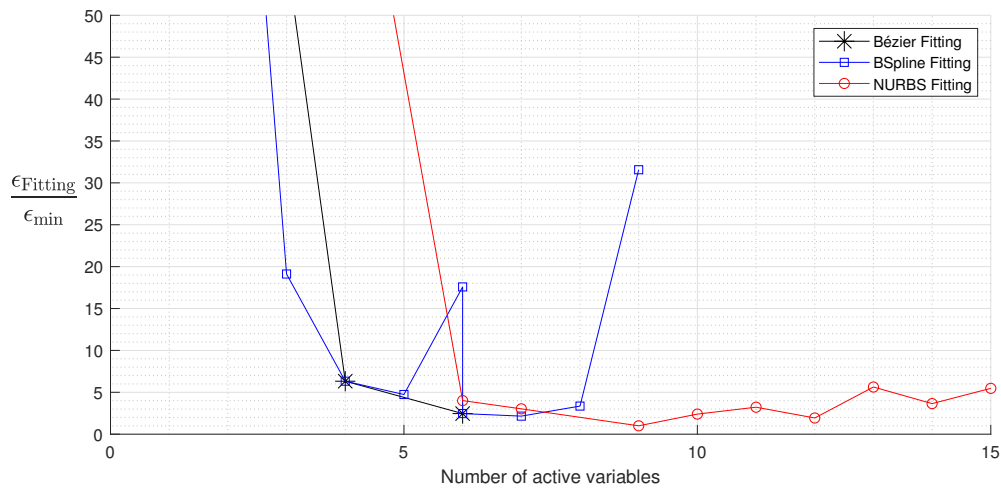


Figure 3.22: *Errore di fitting, rispetto all'errore minimo, per la curva 1 al variare del numero di variabili*

$n = 3$) quasi tutte le curve si collocano in un limitato intervallo dell'errore di fitting. È inoltre evidente come le curve BSpline e Bézier riescano ad ottenere risultati, leggermente peggiori, ma con un numero di variabili molto minore rispetto alle curve NURBS. Infine si nota molto bene che, per entrambe le tipologie di curve testate, all'aumentare del numero di parametri peggiora la qualità di fitting, nonostante l'algoritmo di ottimizzazione abbia un numero sempre maggiore di individui da testare.

Si sconsiglia quindi, ad un futuro utilizzatore del codice, di utilizzare un numero eccessivo di punti di controllo e possibilmente di adottare una tipologia di curva più semplice per facilitare l'algoritmo nella convergenza. In particolare si consiglia di adottare curve BSpline (o al limite Bézier) con $4 \leq n \leq 6$ e $3 \leq p \leq n - 1$. Infine, si sconsiglia di adottare $p = 3$ per quanto evidenziato riguardo la derivata seconda in Figura B.1.

STRUTTURA DEL CODICE

Il codice è stato pensato in modo tale da essere suddiviso in due blocchi principali, riportati in Figura 4.1, in quanto le due parti del codice svolgono funzioni complementari ma completamente diverse. Si è quindi mantenuta una suddivisione del codice in base allo scopo richiesto al codice stesso.

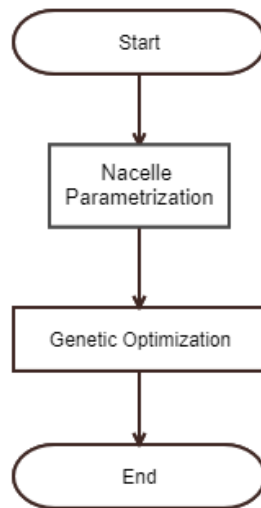


Figure 4.1: *Flowchart principale del codice*

Il codice è stato pensato con un approccio Top-down [34], ossia una strategia mirata a scomporre il problema in sotto-problemi progressivamente più semplici. In questo modo è stato possibile aver chiaro quale fosse lo scopo di ogni sotto-funzione e permettere una più agile scrittura del codice.

4.1 BLOCCO DI PARAMETRIZZAZIONE

Lo scopo principale del blocco di parametrizzazione è la generazione, sulla base della selezione effettuata dall'utente attraverso le interfacce grafiche, delle informazioni necessarie alla successiva ottimizzazione. Viene qui di seguito proposto, Figura 4.2, come sono state suddivise e affrontate le varie sotto-fasi di questo blocco di codice.

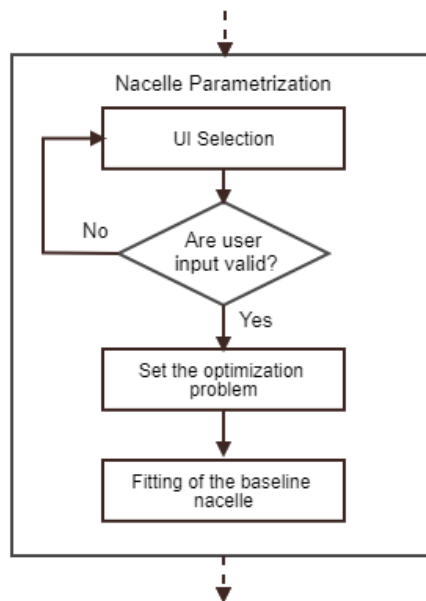


Figure 4.2: *Flowchart della parametrizzazione*

4.1.1 UI SELECTION

L'obiettivo di questa sezione del codice è quello di creare un'unica interfaccia grafica che fosse estremamente facile, completa ed intuitiva da utilizzare per un futuro utilizzatore del codice. Sebbene l'interfaccia grafica non sia strettamente necessaria per l'utilizzo di un software di calcolo, si è ritenuto necessaria la sua introduzione in questo ambito. Il motivo principale, per cui è stata adottata, è per permettere di controllare nel modo più semplice possibile tutte le possibili varianti nella parametrizzazione della nacelle. Essendo molti i controlli a disposizione si è inoltre, nonostante si preferisse avere un'unica interfaccia, ritenuto opportuno suddividere in due interfacce differenti le diverse impostazioni.

L'interfaccia grafica principale, riportata in Figura 4.3, ha il fondamentale scopo di impostare i parametri chiave della futura ottimizzazione, ossia i parametri che influenzeranno il maggior numero di variabili. In particolare, è possibile specificare le seguenti impostazioni:

- L'unità di misura usata per le dimensioni;
- Le variabili globali della nacelle;
- Lo stato nell'ottimizzazione delle variabili globali e i rispettivi limiti di ricerca;
- Lo stato di una curva nell'ottimizzazione, dopo averla selezionata dal grafico;
- La tipologia della curva selezionata (Lineare, Bézier, UBS, BSpline o NURBS);
- Il numero di punti di controllo della curva selezionata, se necessario;

- Il grado della curva selezionata, se necessario.

Inoltre, si è deciso di implementare nell'interfaccia grafica la possibilità di importare/e-sportare le impostazioni stesse di entrambe le interfacce, per permettere una maggiore velocità di utilizzo del software.

L'interfaccia grafica secondaria ha invece lo scopo di impostare i parametri specifici di una curva selezionata. In Figura 4.4 l'interfaccia viene mostrata per la curva 1 a scopo di esempio. Nell'interfaccia principale, dopo aver selezionato la curva, averla attivata nell'ottimizzazione e specificato il tipo, il numero di punti di controllo e il grado della stessa, è stato possibile accedere all'interfaccia successiva tramite il pulsante "*Other curve settings...*". Nella nuova schermata che appare è possibile inserire:

- Le variabili locali specifiche per la curva selezionata;
- Lo stato nell'ottimizzazione delle variabili locali e i rispettivi limiti di ricerca;
- Le condizioni di tangenza iniziali e finali, se libere da vincoli discussi nel Paragrafo 3.2;
- Le condizioni di curvatura iniziali e finali;
- I limiti di ricerca percentuali delle coordinate assiali per i punti di controllo;
- I limiti di ricerca assoluti delle coordinate radiali per i punti di controllo;
- I limiti di ricerca dei pesi;
- L'adozione di particolari "accorgimenti" per velocizzare l'ottimizzazione. Gli "accorgimenti" a cui si fa riferimento sono in particolare:
 - L'attivazione delle coordinate x dei punti di controllo nell'ottimizzazione;
 - L'utilizzo delle coordinate x ricavate nel fitting della baseline imposte e fissate anche nell'ottimizzazione;
 - L'utilizzo dei nodi, se disponibili, ricavati nel fitting della baseline imposte e fissate;
 - L'utilizzo dei pesi, se disponibili, ricavati nel fitting della baseline imposte e fissate.

Per evitare, in entrambe le interfacce, che l'utente possa immettere valori non consentiti sono stati implementati una serie di controlli, specie per il segno delle dimensioni e per i limiti di ricerca. Inoltre, per evitare che l'utente possa uscire dalle interfacce grafiche, attraverso il pulsante *OK*, senza aver completato l'inserimento si è reso vincolante l'inserimento dei parametri dimensionali. Infine, per evitare l'apertura multipla di medesime interfacce, rischiando quindi conflitti nell'inserimento dei dati, la seconda interfaccia è stata resa "*modal*" [35], ossia una volta che appare la schermata, e fino a che resterà aperta, non è possibile interagire con altre schermate del software.

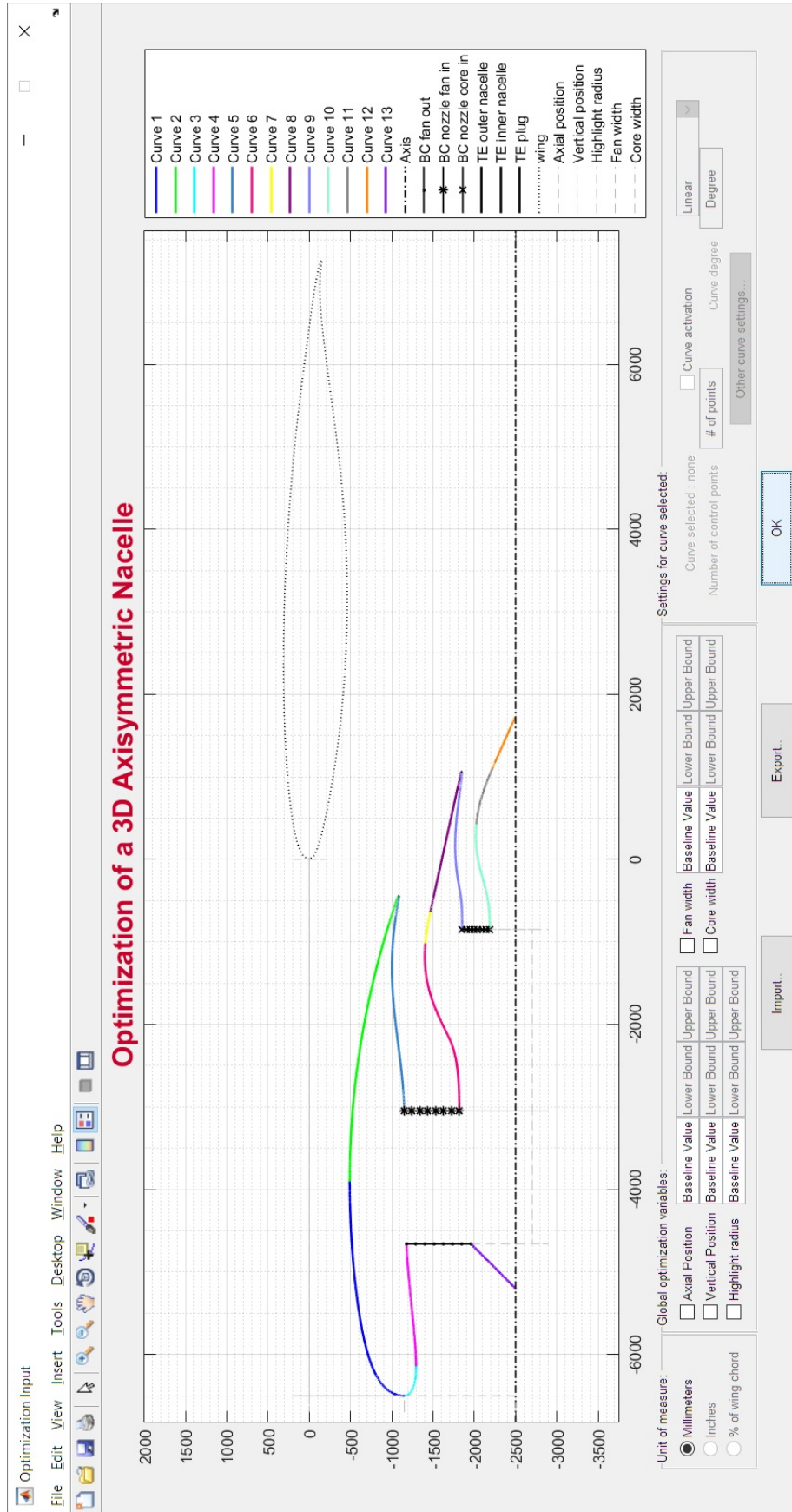


Figure 4.3: Interfaccia grafica principale

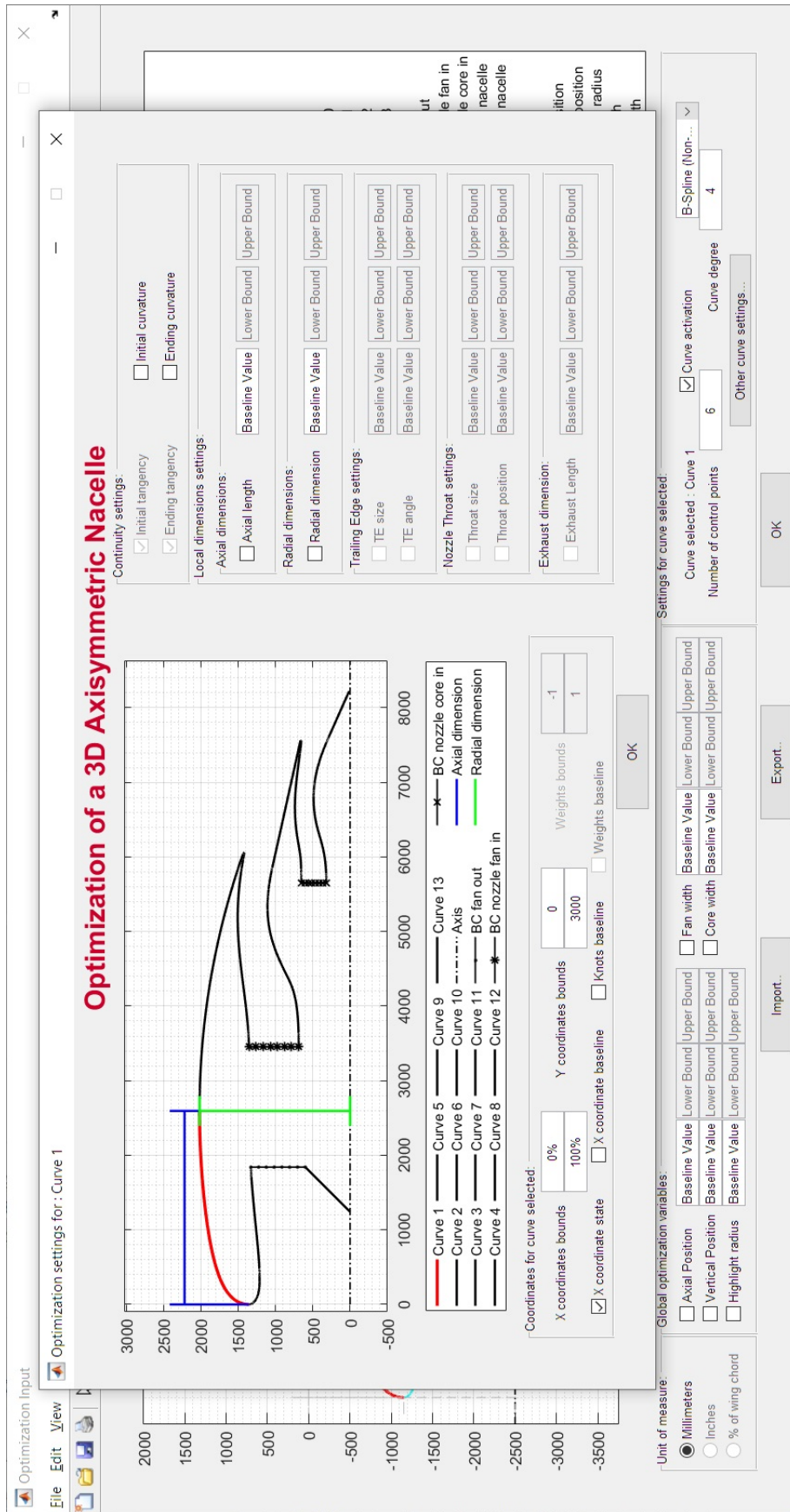


Figure 4.4: Interfaccia grafica secondaria

Nella Figura 4.5 viene riportato lo schema a blocchi qualitativo delle operazioni che vengono compiute dalla prima interfaccia grafica. Ogni parte dello schema rappresenta, secondo la logica Top-down, una o più funzioni eseguite dal codice. Il codice dell'interfaccia è stato strutturato in modo tale che, dopo la creazione dell'interfaccia grafica, essa resti in attesa di una qualunque azione dell'utente. Viene di seguito riportato un estratto del codice che mostra come è stato implementato il loop per garantire l'utilizzo dell'interfaccia grafica e l'esportazione dei suoi parametri.

```
global Variables On_go
%Variables = Struct with the selections
%On_go = check variable to close the interface

%Create the first UI figure
Figure_creation(Working_directory , Function_directory ,0)

%Is the figure still open?
while isempty(On_go)
    %yes, pause the function to wait user input
    pause(1)
end

%Has something been selected by the user?
if On_go==0.5
    %No, set empty output
    input_var = [];
else
    %Yes, set output and close figure
    input_var=Variables;
    close all
end
```

Prima della creazione dell'interfaccia vengono inizializzate le due variabili globali fondamentali per il suo funzionamento. Lo scopo di "Variables" è quello di salvare le selezioni effettuate dell'utente, dopo aver verificato la loro fattibilità, ed esportarle nello script principale. Lo scopo della variabile "On_go" è, invece, quello di garantire di uscire dal loop. In base al valore assunto da questa variabile si procede in due vie:

1. Esportando le selezioni dell'utente, in quanto hanno superato i controlli necessari;
2. Senza esportare nulla, in quanto la finestra è stata chiusa prima di completare le selezioni.

Nello schema a blocchi in Figura 4.5 viene evidenziato come in caso di selezione dell'utente di una qualunque delle impostazioni essa venga memorizzata in una variabile dedicata. Viene di seguito riportata la parte del codice necessaria per il salvataggio delle selezioni dell'utente, nel caso in cui tutti i controlli effettuati siano superati. Inoltre, si nota molto bene in queste poche righe, come vengano esportate in un'unica struttura dati le selezioni dell'utente. Progettando il codice con questa struttura è possibile avere i dati in una forma molto comoda per la successiva elaborazione e garantirne una rapida per una esportazione/importazione da file esterno.

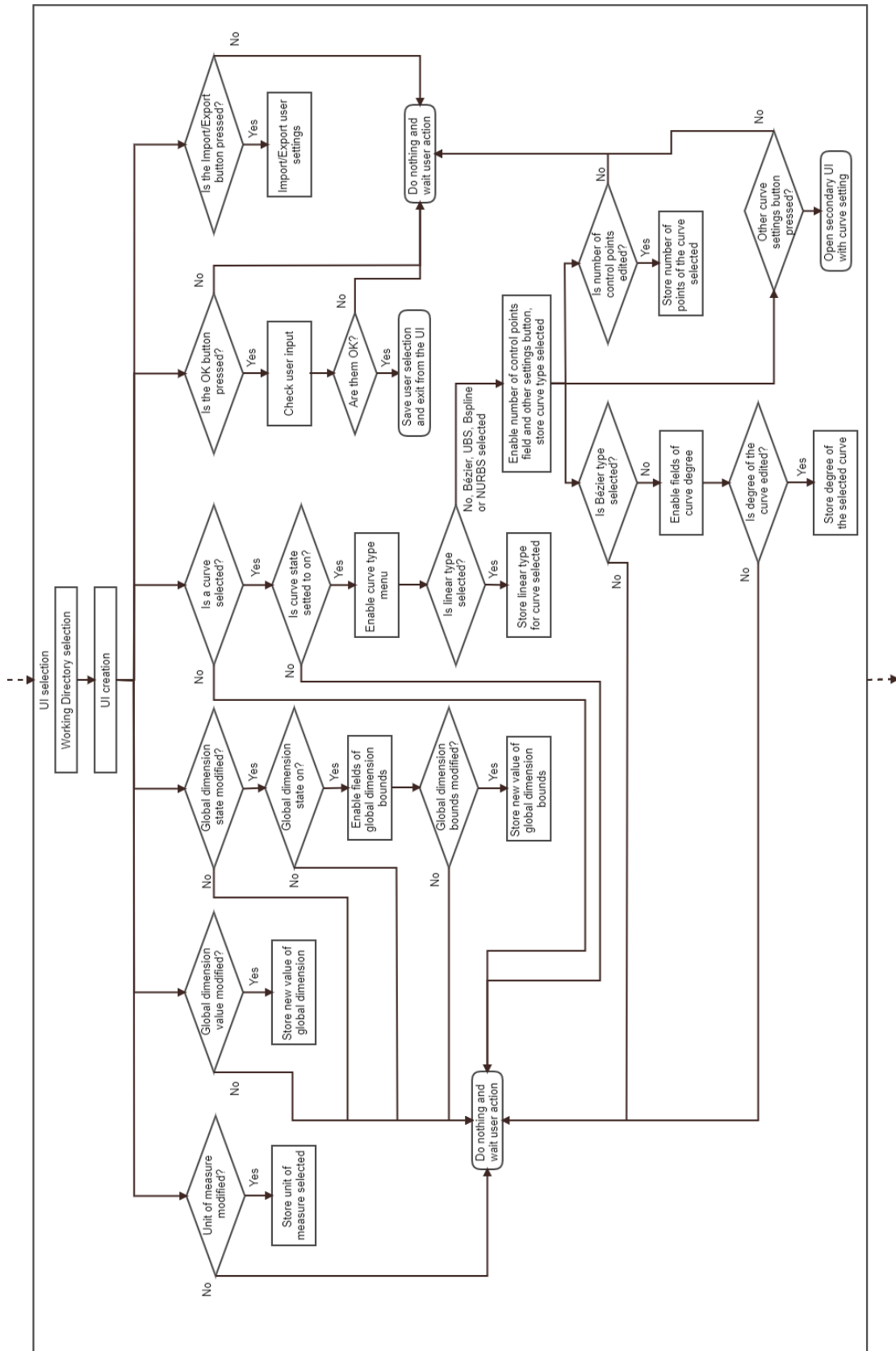


Figure 4.5: Schema a blocchi dell'interfaccia grafica principale

```

if isempty(ERROR)
    %no, so store selections and exit UI

    %write them in one variable
    Variables.unit_measure=unit_measure;
    Variables.chord_value=chord_value;
    Variables.global_var=global_value;
    Variables.local_var=local_value;
    Variables.curve=curve;

    %set global variable to exit the interface
    On_go=1;
end
    
```

In particolare i campi che vengono esportati sono:

- *unit_of_measure* = Stringa contenente l'unità di misura selezionata;
- *chord_value* = Double con il valore della corda dell'ala nel punto di posizionamento della nacelle;
- *global_var* = Matrice contenente lo stato, i valori di partenza e i limiti delle dimensioni globali;
- *local_var* = Matrice contenente lo stato, i valori di partenza e i limiti delle dimensioni locali;
- *curve* = Struttura dati contenente le informazioni necessarie per ogni curva.

Nello schema a blocchi in Figura 4.6 viene riportato il processo logico che governa l'interfaccia grafica secondaria.

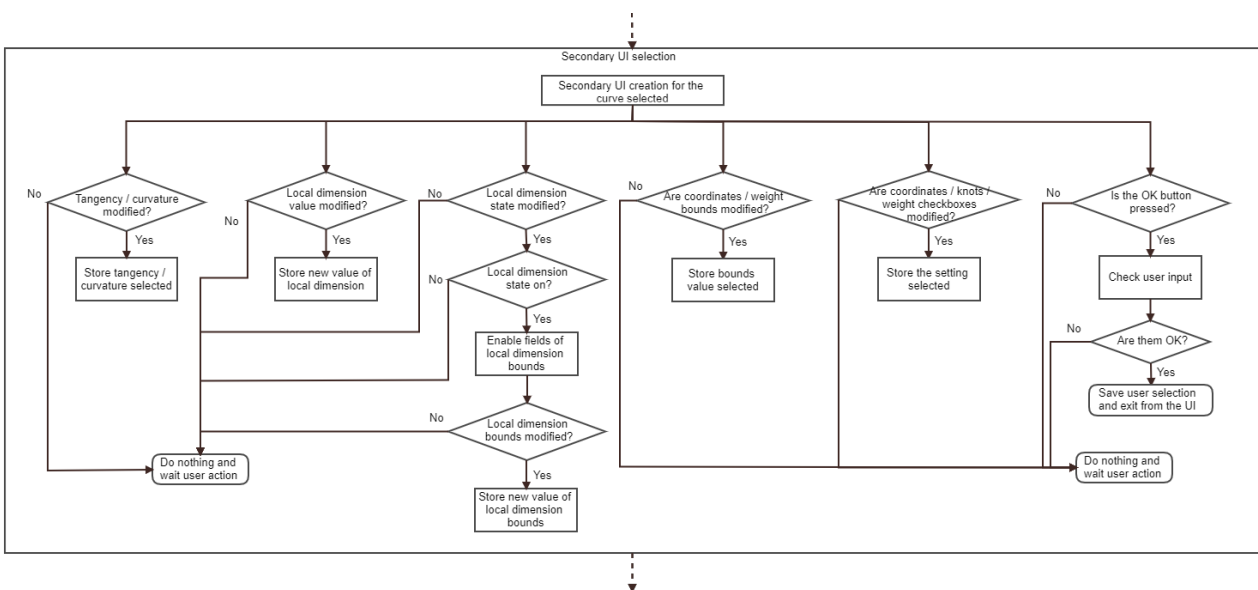


Figure 4.6: Schema a blocchi dell'interfaccia grafica secondaria

Nello schema a blocchi in Figura 4.6 è evidente come la parte riguardante le variabili locali si ripeta in modo uguale alla parte riguardante le variabili globali in Figura 4.5. Infatti questa seconda interfaccia grafica è stata costruita sfruttando le funzioni della prima, ove possibile, per permettere una più veloce scrittura del codice.

4.1.2 OPTIMIZATION PROBLEM SETTING

Nel paragrafo 4.1.1 è stato descritto come vengano selezionati i dati da utente e come poi questi vengano esportati nel codice principale. Diventa, a questo punto, fondamentale capire come questi dati "grezzi" vengano lavorati. L'obiettivo è quello di ottenere i dati in una forma adeguata per il fitting e per l'ottimizzazione della nacelle. Nello schema a blocchi in Figura 4.7 viene riportata la logica e le fasi seguite per riuscire a costruire le informazioni necessarie al problema di ottimizzazione.

A partire dalle dimensioni di riferimento fornite nel Paragrafo 3.3, le quali vengono combinate con le dimensioni specificate da utente nell'interfaccia grafica, viene creata la corrispondente nacelle, scelta come baseline, di cui viene anche salvata un'immagine. Invece di impostare una nacelle baseline fissa, si è optato per questa modalità di creazione della geometria di partenza in modo tale da rendere più semplice un'eventuale modifica alle dimensioni iniziali.

A questo punto del codice diventa fondamentale introdurre l'algoritmo di ottimizzazione che si userà. Questo è necessario per capire come strutturare i dati per fornirli all'algoritmo stesso. In particolare l'algoritmo usato, appartenente al Optimization Toolbox di Matlab, è il *gamultiobj* [36], il quale verrà implementato con la seguente Equazione 4.1:

$$x = \text{gamultiobj}(\text{fun}, \text{nvars}, A, b, Aeq, beq, lb, ub, \text{options}). \quad 4.1$$

L'aspetto fondamentale da garantire, per poter utilizzare efficacemente questo algoritmo genetico, è che la funzione da ottimizzare *fun* operi su un vettore di dati corrispondente all'individuo da valutare. Questo vettore pertanto deve contenere tutte le informazioni necessarie per costruire una nacelle. Per quanto visto nel capitolo 3, non tutte le variabili geometriche sono effettivamente a disposizione dell'algoritmo, pertanto si è resa necessaria la creazione di più vettori di dati, strutturati nel seguente modo:

- *devar_value* = Vettore con i valori di tutte le variabili attive nel processo di ottimizzazione;
- *variable_value* = Vettore con i valori di tutte le variabili coinvolte nel processo di ottimizzazione;
- *variable_state* = Vettore con lo stato di tutte le variabili coinvolte nel processo di ottimizzazione.

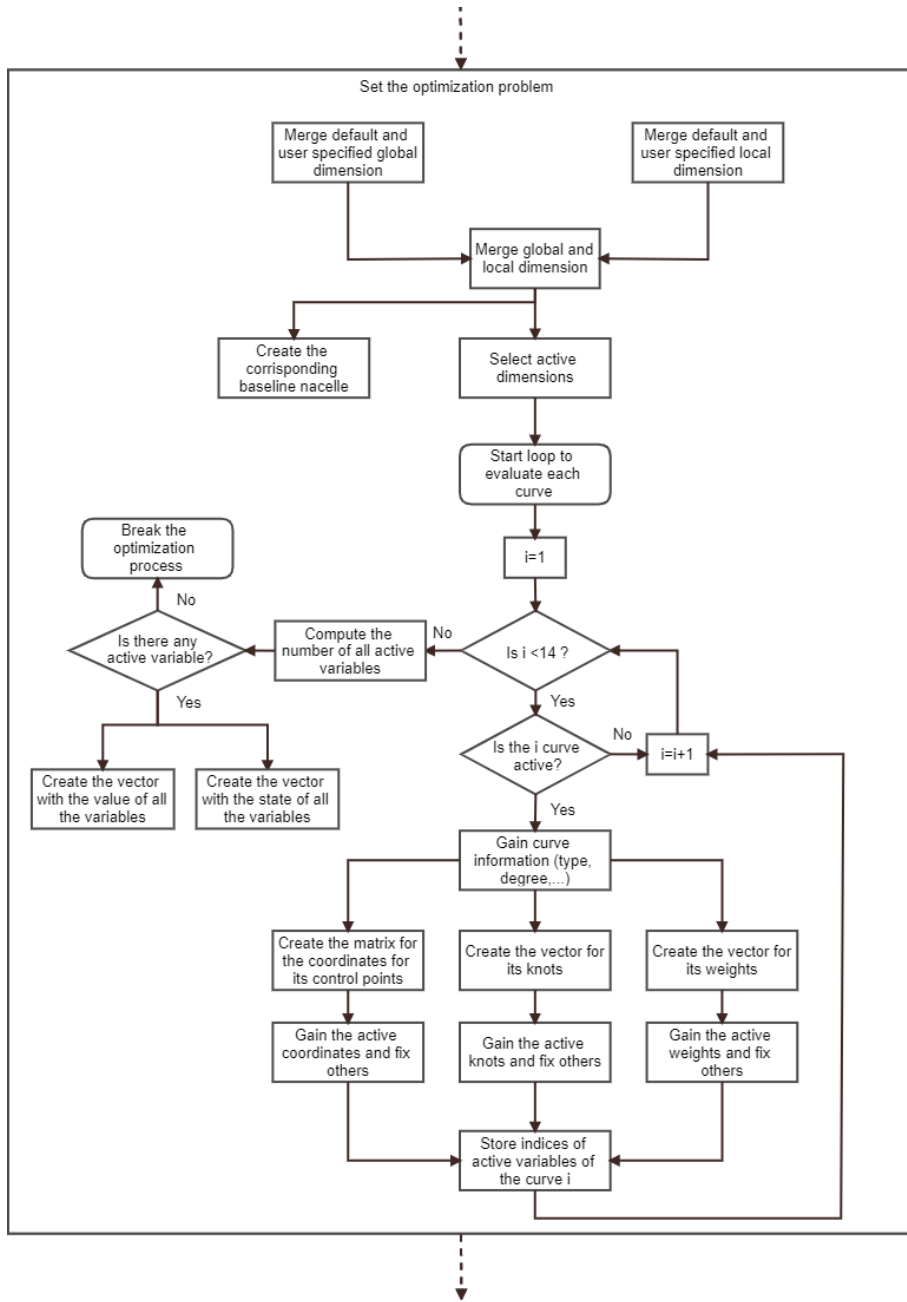


Figure 4.7: Schema a blocchi per l'impostazione del problema di ottimizzazione

La modalità con cui i tre vettori interagiscono tra loro viene affrontata nei successivi paragrafi. Nonostante ciò, il primo vettore verrà generato direttamente dall'algoritmo, mentre il secondo fornirà le informazioni necessarie per costruire il terzo vettore, con le dimensioni specifiche, per l'individuo valutato. Per compiere questo scopo, gli ultimi due vettori sono stati costruiti seguendo lo schema riportato nell'Equazione 4.2.

$$\begin{aligned}
 \vec{V}_{value} = & (Global_variables_{value} \quad Local_variables_{value} \quad Curve1_variables_{value} \quad \dots \\
 & \quad Curve2_variables_{value} \quad \dots \quad Curve13_variables_{value}) \\
 \vec{V}_{state} = & (Global_variables_{state} \quad Local_variables_{state} \quad Curve1_variables_{state} \quad \dots \\
 & \quad Curve2_variables_{state} \quad \dots \quad Curve13_variables_{state})
 \end{aligned}
 \tag{4.2}$$

4.1.3 NACELLE FITTING

Al termine del fase precedente si ottengono i vettori riportati nell'Equazione 4.2. Mentre il vettore con gli stati delle variabili è già pronto per l'uso nell'ottimizzazione, per quanto riguarda il vettore con i valori si ha un vettore "semilavorato". Infatti, la nacelle di partenza è stata costruita con le curve del JSM [33] e non con le curve parametriche richieste da utente. Si tratta quindi di eseguire un fitting della nacelle con le curve, o meglio con i parametri liberi ottenuti di ogni curva, della nacelle di partenza e successivamente "completare" il vettore \vec{V}_{value} per il caso di partenza. Per ottenere tutte le curve necessarie si segue lo schema riportato in Figura 4.8.

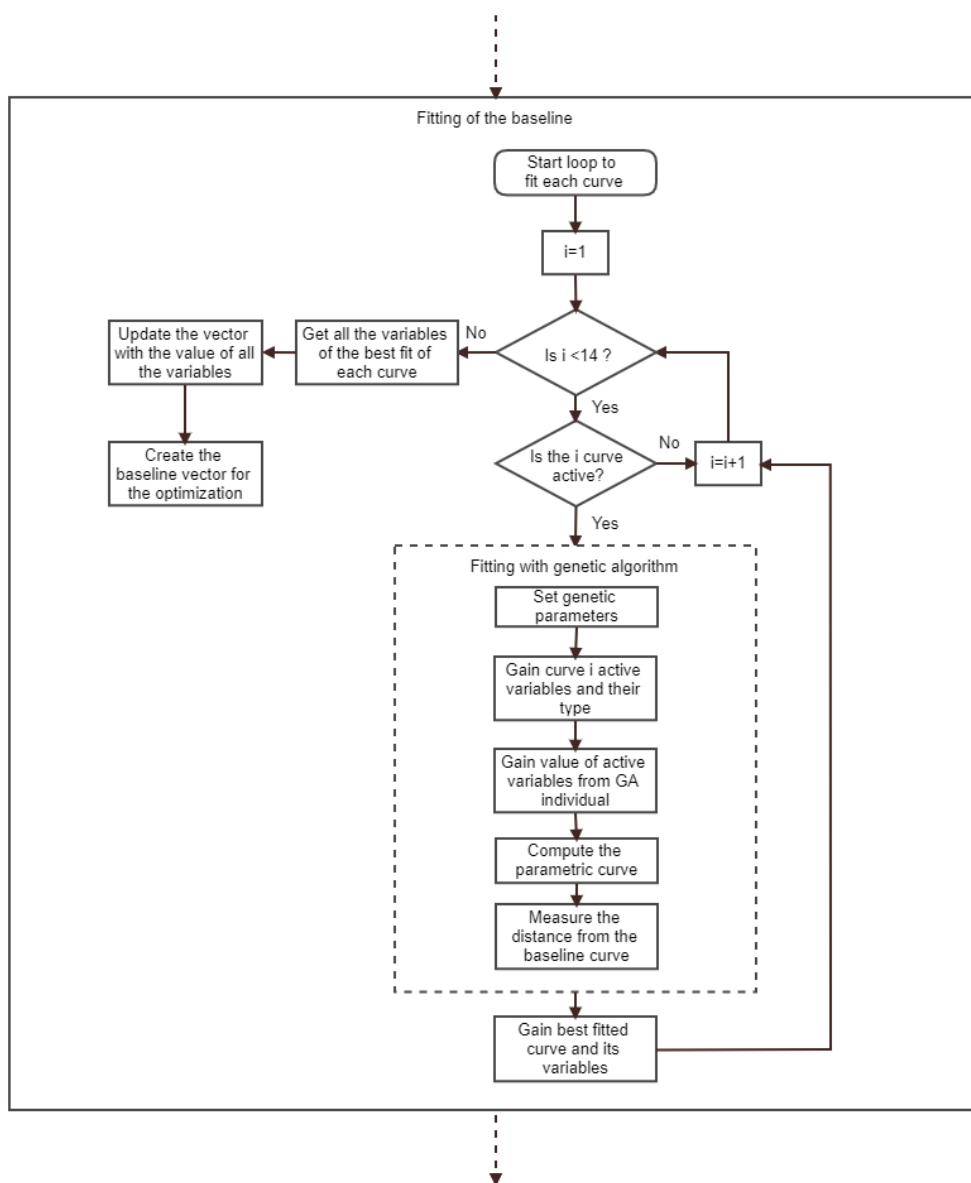


Figure 4.8: Schema a blocchi per il fitting della nacelle di partenza

Per unificare il codice di calcolo, ogni curva viene generata come se fosse una NURBS, essendo le altre tipologie di curve semplicemente casi particolari di questa. Considerando il gran numero di volte, decine di milioni, che la funzione per generare una NURBS è stata richiamata (per le analisi di sensibilità, per ogni fitting, per le ottimizzazioni) si è reso necessario una particolare cura nel codice. I principali accorgimenti introdotti riguardano la preallocazione dei vettori necessari e la riduzione al minimo dei cicli necessari. Le modifiche in questione, rispetto alla prima bozza di codice, hanno permesso di ridurre di oltre il 90% il tempo di esecuzione. Viene riportato in Appendice C il codice utilizzato.

4.2 BLOCCO DI OTTIMIZZAZIONE

Lo scopo principale del blocco di ottimizzazione è la preparazione e l'ottimizzazione della nacelle precedentemente parametrizzata. Viene qui di seguito proposto, Figura 4.9, come sono state suddivise e affrontate le varie sotto-fasi di questo blocco di codice.

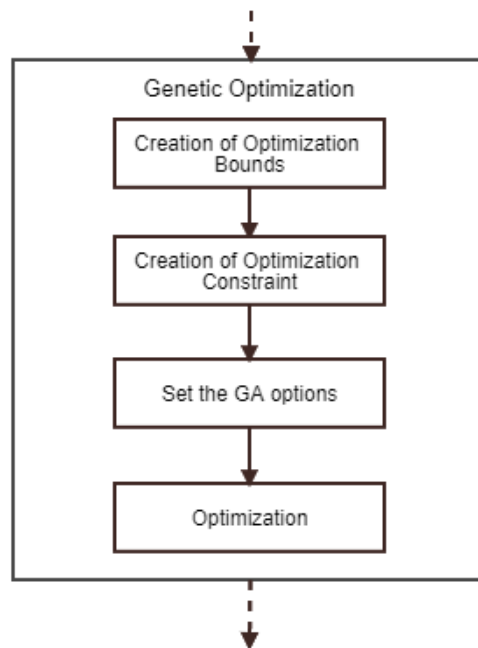


Figure 4.9: *Flowchart per il blocco di ottimizzazione*

4.2.1 OPTIMIZATION BOUNDS

Per valutare i limiti di ricerca, lb e ub , da fornire all'ottimizzatore riportato in Equazione 4.1 si è suddiviso il problema in due parti che verranno ora affrontate.

Limiti per le variabili dimensionali

Inizialmente, sfruttando le variabili dimensionali attive nella parametrizzazione, si ottengono le forme della nacelle corrispondente alle dimensioni minime e a quelle massime. Viene riportato un esempio di ciò in Figura 4.10 in cui sono state attivate solo alcune variabili dimensionali.

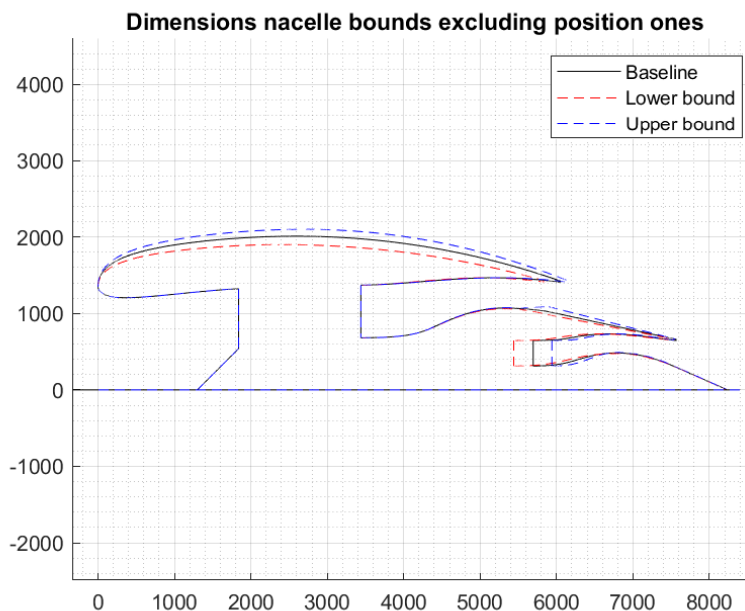


Figure 4.10: Esempio di limiti per le variabili dimensionali

Sfruttando questo modo di operare è stato possibile identificare, per ognuna delle tredici curve, il punto iniziale e finale di ognuna. Considerando in particolare le coordinate assiali, è stato possibile ricavare il punto $P_{0,x}$ minimo e il punto $P_{n,x}$ massimo di ogni curva, ottenendo quindi il range per le coordinate assiali dei punti di controllo di ogni curva, attraverso l'Equazione:

$$\Delta P_x = P_{n,x} - P_{0,x} \quad 4.3$$

Limiti per le variabili delle curve

Successivamente, grazie alle informazioni inserite nella seconda interfaccia in Figura 4.4, questo range viene modificato in base ai parametri inseriti da utente tra il 0% e il 100% della sua estensione.

Per quanto riguarda invece i limiti per le coordinate radiali dei punti di controllo e per gli eventuali pesi si adottano dei valori di default, $[0 ; 3000]$ e $[-1 ; 1]$, che possono essere modificati da utente, per ogni curva, nella medesima UI. I limiti di ricerca sui nodi di una curva invece vengono automaticamente imposti tra $[0 ; 1]$ e non sono modificabili da utente, in quanto, essendo valori normalizzati, si vuole mantenere un controllo più stretto su questi parametri.

Quindi, avendo a disposizione i limiti di ricerca per ogni curva e variabile dimensionale, vengono costruiti i due vettori lb e ub aventi dimensione pari al numero di variabili attive. Viene riportato, in Figura 4.11, il diagramma a blocchi corrispondente a quanto descritto.

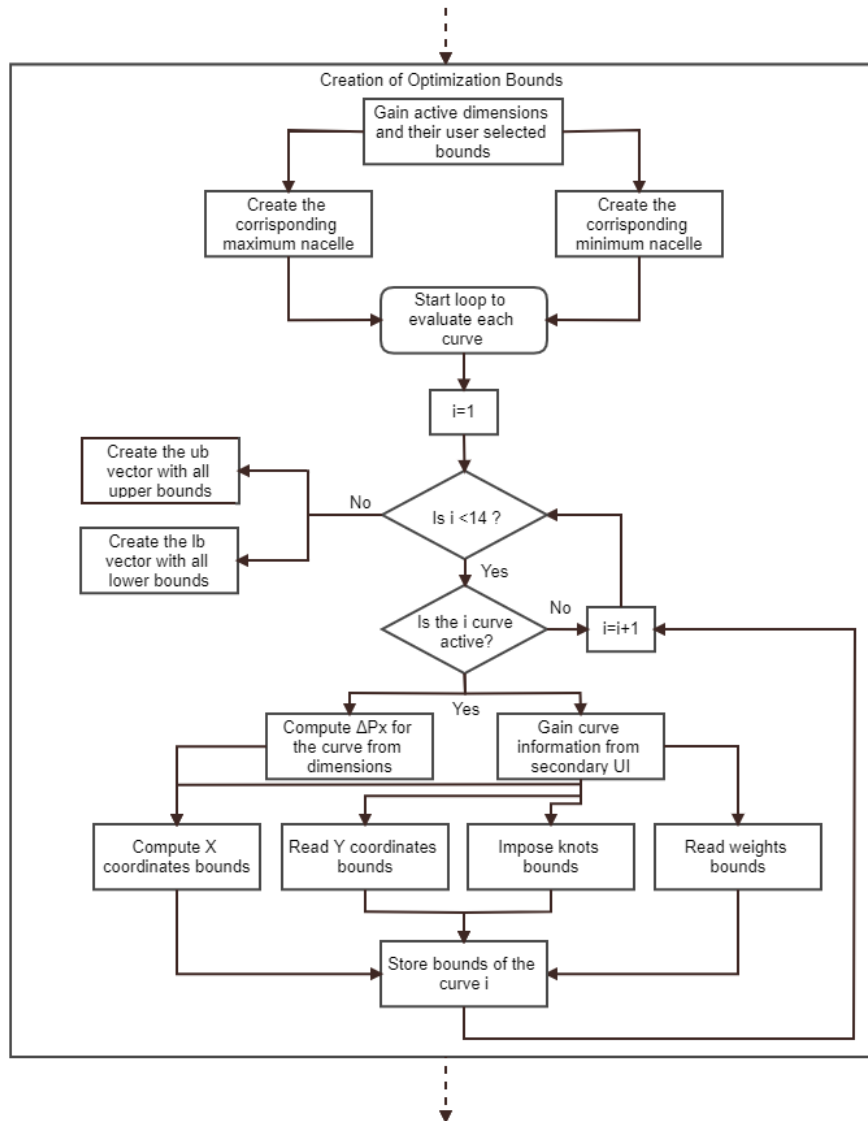


Figure 4.11: Flowchart per ottenere i limiti di ricerca dell'ottimizzatore

4.2.2 OPTIMIZATION CONSTRAINT

Per valutare i vincoli di ricerca necessari da fornire all'ottimizzatore riportato in Equazione 4.1 è fondamentale capire prima che tipologie di vincoli possono essere forniti:

- A = Matrice di dimensione $N_{vincoli} \times N_{variabili}$;
- b = Vettore colonna di $N_{vincoli}$;
- A_{eq} = Matrice di dimensione $N_{vincoli} \times N_{variabili}$;

- beq = Vettore colonna di $N_{vincoli}$.

I vincoli sono lineari e sono del tipo $A \leq b$ e $Aeq = beq$. Inoltre, anche se non è stata adottata questa soluzione, potrebbero essere forniti all'ottimizzatore vincoli non lineari di disuguaglianza e uguaglianza. I vincoli non lineari potrebbero essere adottati per specificare le condizioni di tangenza espresse dalle Equazioni 2.11, 2.12, 2.13 e 2.14, in realtà si è preferito evitare e adottare una strategia alternativa. Questo è stato fatto in quanto l'imposizione dei vincoli non lineari in primo luogo non è esatta, in secondo luogo rallenta l'ottimizzazione. Inoltre, i vincoli adottati sono stati tutti in forma di disequazioni, pertanto:

- $Aeq = []$;

- $beq = []$.

Per valutare i vincoli A e b , da fornire all'ottimizzatore si è suddiviso il problema in due parti che verranno ora affrontate.

Vincoli per le variabili dimensionali

I vincoli lineari che possono essere facilmente introdotti per le variabili dimensionali sono:

1. $R_{Hi} \leq R_{max}$;
2. $R_{exit,fan} \leq R_{max}$;
3. $R_{throat} \leq R_{Hi}$;
4. $R_{AIP} \leq R_{max}$;
5. $R_{fan,shroud} \leq R_{max}$;
6. $R_{fan,hub} \leq R_{fan,shroud}$;
7. $x_{th,fan} \leq L_{pre} + L_{post}$;
8. $L_{throat} + L_{AIP} + W_{fan} \leq x_{th,fan}$;
9. $R_{exit,core} \leq R_{exit,fan}$;
10. $R_{core,shroud} \leq R_{fan,shroud}$;
11. $R_{core,hub} \leq R_{core,shroud}$;
12. $L_{throat} + L_{AIP} + W_{fan} + W_{core} \leq x_{th,core}$;
13. $R_{spinner} \leq R_{AIP}$.

Risulta immediato notare come non tutte le curve hanno un vincolo sulle dimensioni, ne tanto meno tutte le variabili sono controllate. Il problema principale nell'imposizione di questi vincoli è legato al fatto che le variabili coinvolte possono essere variabili attive o meno nell'ottimizzazione. Diventa quindi fondamentale riuscire a gestire i 13 vincoli

elencati in base allo stato di attivazione di ogni variabile.

Vengono inizializzati, con valore pari a 0, le variabili A con dimensioni 13×36 e b con dimensione 13, dove ad ogni riga corrisponde uno dei precedenti vincoli, mentre ad ogni colonna corrisponde una delle variabili dimensionali, nell'ordine in cui sono state riportate nel Capitolo 3. Si procede progressivamente ad imporre i 13 vincoli con l'approccio mostrato di seguito. Infine vengono rimossi i vincoli non necessari. Il flowchart delle operazioni descritte viene riportato nella prima parte di Figura 4.14.

Per capire come è stato risolto il problema in questione consideriamo, a titolo di esempio, la prima disequazione e si riporta in Figura 4.12 lo schema per costruire i vincoli. È evidente come, attraverso due operatori di selezione, si proceda per capire quali siano le variabili dimensionali effettivamente attive nell'ottimizzazione. Successivamente, in base alla disequazione necessaria, si aggiornano i valori di A e b .

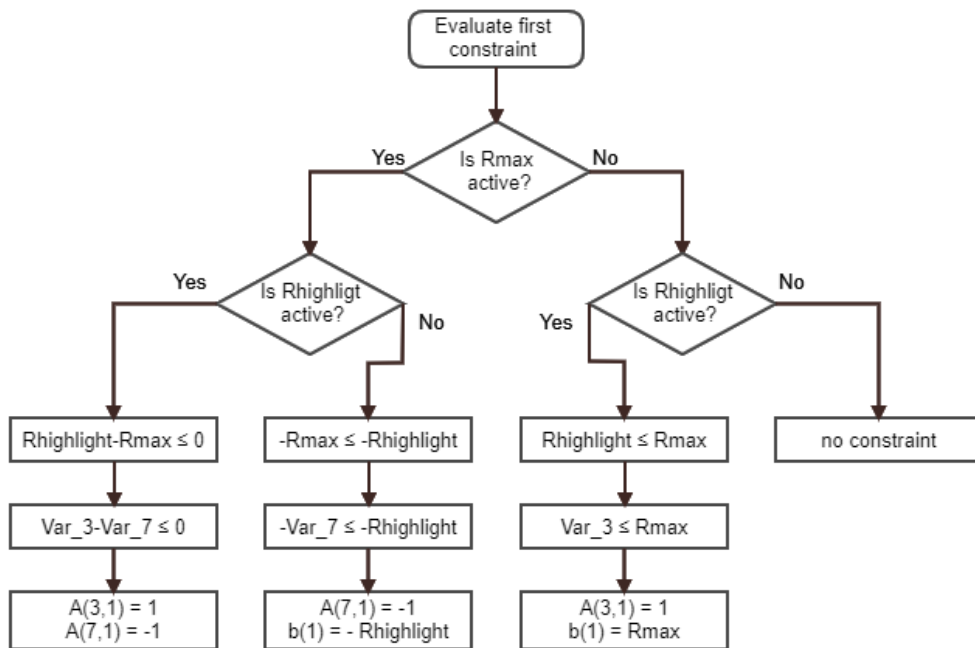


Figure 4.12: Schema logico per la costruzione del primo vincolo dimensionale

Osservando il precedente schema si nota che il caso a sinistra può essere replicato dai due casi centrali per quanto riguarda la matrice A . È quindi nata l'idea di utilizzare un unico "costruttore" per tutte le tipologie di vincolo. Inoltre, considerando questo modo di procedere si ha per le disequazioni con più variabili un aumento considerevole dei casi da considerare. Si tratta quindi di modificare leggermente l'idea esposta per renderla molto più efficiente.

Lo schema a blocchi migliorato, in modo tale da garantire il superamento dei limiti evidenziati, viene riportato in Figura 4.13 per un vincolo generico.

La funzione corrispondente a questo schema a blocchi viene riportata in Appendice D.

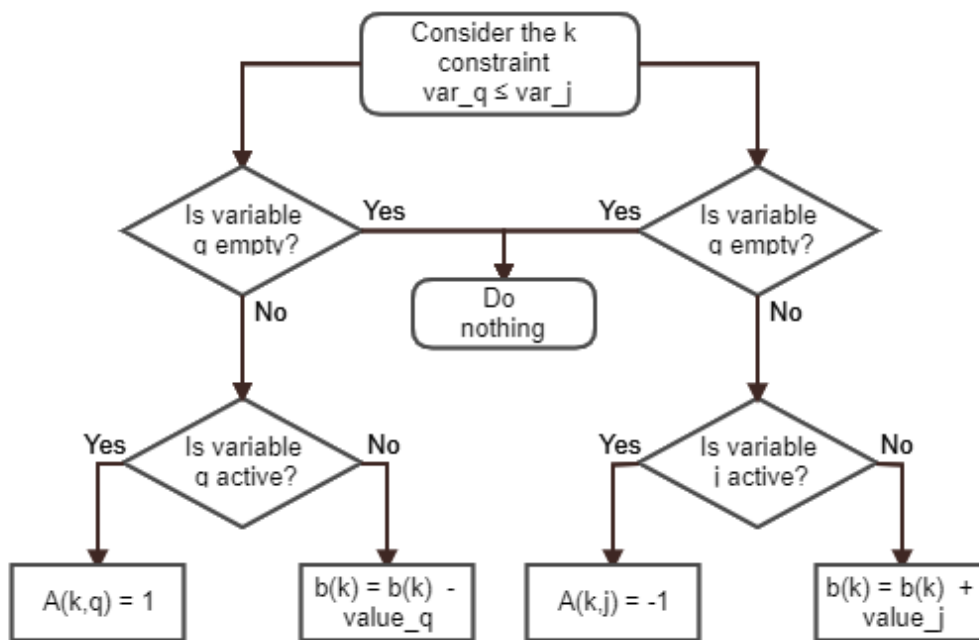


Figure 4.13: Schema logico per la costruzione di un vincolo generico

Per impostare ora, ad esempio, il vincolo 7 sarà sufficiente richiamare due volte la funzione sulla stessa riga:

- $[A,b] = \text{constraint_builder}(A,b,7,\text{state},\text{value},6,19)$
- $[A,b] = \text{constraint_builder}(A,b,7,\text{state},\text{value},8,[])$

In questo modo il vincolo 7 viene aggiornato con due variabili nella parte di destra dell'equazione ed una a sinistra. Nel caso in cui più di una variabile non sia attiva, il vettore b , per come è stato scritto il codice, viene aggiornato ad ogni chiamata della funzione e non sovrascritto.

Vincoli per le variabili delle curve

Dopo aver creato i vincoli per le variabili dimensionali si procede alla creazione dei vincoli per le singole curve. In particolare le tipologie di vincolo necessarie sono:

- Vincolo nelle coordinate x del primo punto di controllo libero;
- Vincoli nelle coordinate x dei punti di controllo centrali;
- Vincolo nelle coordinate x dell'ultimo punto di controllo libero;
- Vincoli nei nodi centrali.

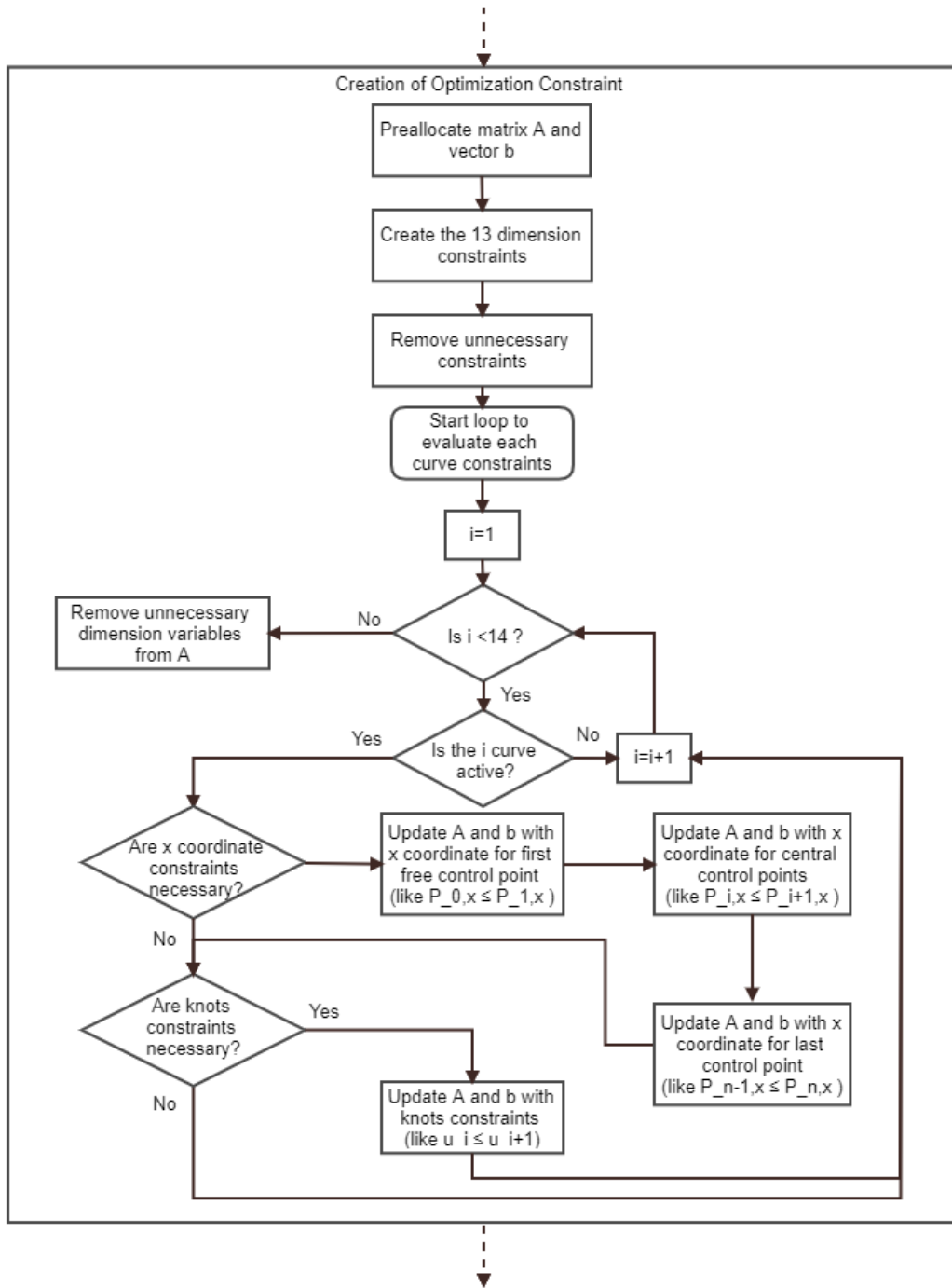


Figure 4.14: Flowchart per la costruzione dei vincoli

La prima e la terza tipologia di vincoli, che non per tutte le curve possono essere imposti in modo lineare, sono necessari a garantire che la coordinata assiale del primo (ultimo) punto di controllo libero non sia prima (dopo) del punto di controllo iniziale (finale) della curva. Infatti, in questo caso, si avrebbe che la curva parametrica non è contenuta tra i punti di controllo estremali, ottenendo di conseguenza una forma di nacelle non voluta. Queste due tipologie di vincolo sono state applicate con la medesima funzione esposta precedentemente, ad esempio per la curva 4:

- 1^a tipologia $[A,b] = \text{constraint_builder}(A,b,k,state,value,index_{P_{1,x}},10)$
- 3^a tipologia $[A,b] = \text{constraint_builder}(A,b,k,state,value,10,index_{P_{n-1,x}})$
- 3^a tipologia $[A,b] = \text{constraint_builder}(A,b,k,state,value,12,[])$

Dove con $index_{P_{1,x}}$ ($index_{P_{n-1,x}}$) si intende l'indice della variabile $P_{1,x}$ ($P_{n-1,x}$) per la curva 4 nel vettore $decvar_value$ del Paragrafo 4.1.2.

Per le altre due tipologie di vincoli si ha l'obiettivo di imporre che una coordinata x (o un nodo) di curva sia successiva alla antecedente e che preceda la conseguente, ossia che si verifichi $P_{i-1,x} \leq P_{i,x} \leq P_{i+1,x}$ (o $u_{i-1,x} \leq u_{i,x} \leq u_{i+1,x}$).

Al termine dell'inserimento di tutti i vincoli delle curve è possibile rimuovere le variabili dimensionali non utilizzate tra le 36 iniziali. Si è deciso di procedere in questa fase con questa eliminazione, nonostante si potesse già prima dell'imposizione dei vincoli delle curve, per fare in modo che la 1^a e la 3^a tipologia potessero operare su variabili dimensionali con indici "fissati".

4.2.3 IMPOSTAZIONE DEL GA

In questa fase del codice vengono impostati i parametri necessari all'algoritmo di ottimizzazione che generalmente sono fornite nella variabile *options* dell'Equazione 4.1. Le opzioni vengono fornite all'algoritmo attraverso la seguente istruzione:

```
%create options for the optimization
options=optimoptions('gamultiobj','InitialPopulation',initPop,...
    'Generations',number_Gen,'PopulationSize',number_Ind,
    'TolFun',1e-20,'PlotFcn',{@gaplotpareto,...
    @gaplotspread,@gaplotparetodistance});
```

In particolare i parametri che sono forniti sono:

- *initPop* : Popolazione iniziale dell'ottimizzazione;
- *number_Gen* : Numero di generazioni, $N_{generazioni}$, a disposizione dell'ottimizzatore;
- *number_Ind* : Numero di individui, $N_{individui}$, per ogni generazione.

La popolazione iniziale è fornita come un unico individuo formato dalle variabili della nacelle di partenza ottenute nel fitting. Gli altri due parametri vengono forniti dall'utente attraverso una semplice interfaccia grafica, nativa di MATLAB, attraverso il codice:

```
%ask the user the GA settings
prompt = {'Proportionality factor K for the number of
    individuals:', 'Number of generations:'};
Title = 'GA parameters';
```

```
dims = [1 42];  
definput = { '5 ', '40 ' };  
%get user answer  
answer = inputdlg(prompt, Title, dims, definput);
```

Da evidenziare è il parametro K , in quanto, non essendo noto a priori il numero esatto di variabili coinvolte nell'ottimizzazione, si preferisce calcolare il numero di individui $N_{individui}$ successivamente attraverso l'Equazione 4.4:

$$N_{individui} = K \cdot N_{variabili} \quad 4.4$$

Dove $N_{variabili}$ è il numero di variabili coinvolte nell'ottimizzazione, dimensione del vettore *devar_value*.

4.2.4 OTTIMIZZAZIONE

L'ottimizzazione effettiva, attraverso l'algoritmo genetico di Equazione 4.1, avviene ottimizzando la funzione *fun*. In particolare la funzione ha lo scopo, a partire da un individuo, di creare la geometria della nacelle corrispondente, procedere alla creazione della mesh di esso e successivamente alla sua simulazione fluidodinamica, infine vengono computate le cifre di prestazioni dell'individuo e si procede con il successivo. Lo schema a blocchi della funzione viene riportato nella Figura 4.15.

Nello schema a blocchi di Figura 4.15 si sottolineano in particolare le due fasi evidenziate in rosso. Infatti questi due particolari punti del codice di calcolo sono gestiti con software commerciali esterni a MATLAB. In particolare sono stati utilizzati, a scopo di prova per la scrittura del codice, Pointwise Inc per la creazione della mesh e Ansys Fluent per la simulazione fluidodinamica 3D.

Vengono riportate nelle Figure 4.16 e Figure 4.17 le immagini della mesh nei pressi della nacelle. La griglia di calcolo è stata creata seguendo le indicazioni e i modelli forniti [37] a partire dalla mesh 2D dell'individuo generato. Per quanto riguarda invece la scrittura del journal di Pointwise è stata utilizzata la guida del software [38]. Per quanto riguarda le condizioni al contorno imposte nella simulazione CFD si è fatto riferimento ad un ciclo 0D del turbofan con i parametri forniti nel Paragrafo 3.3.

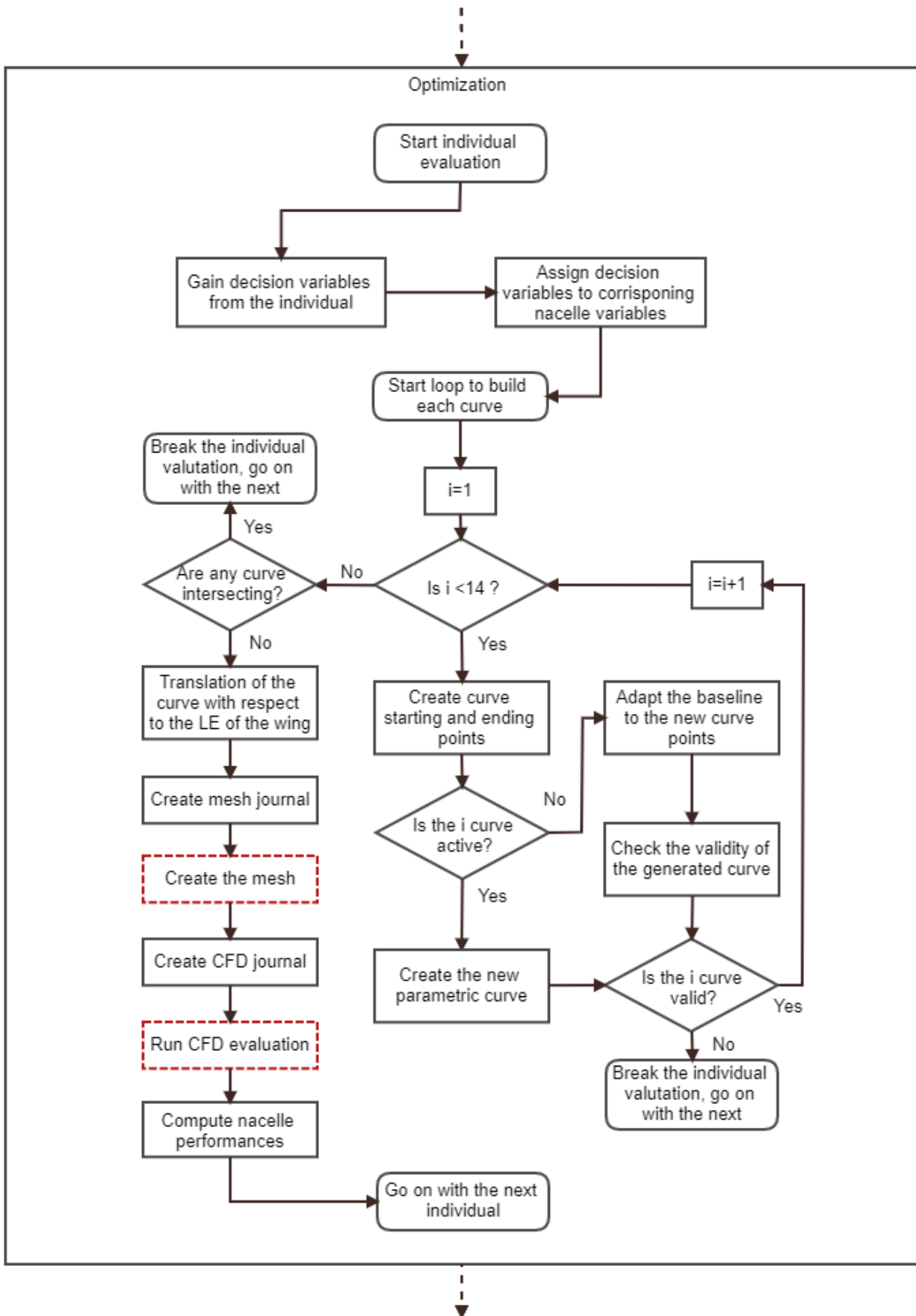


Figure 4.15: Flowchart per la valutazione di un individuo dell'ottimizzazione

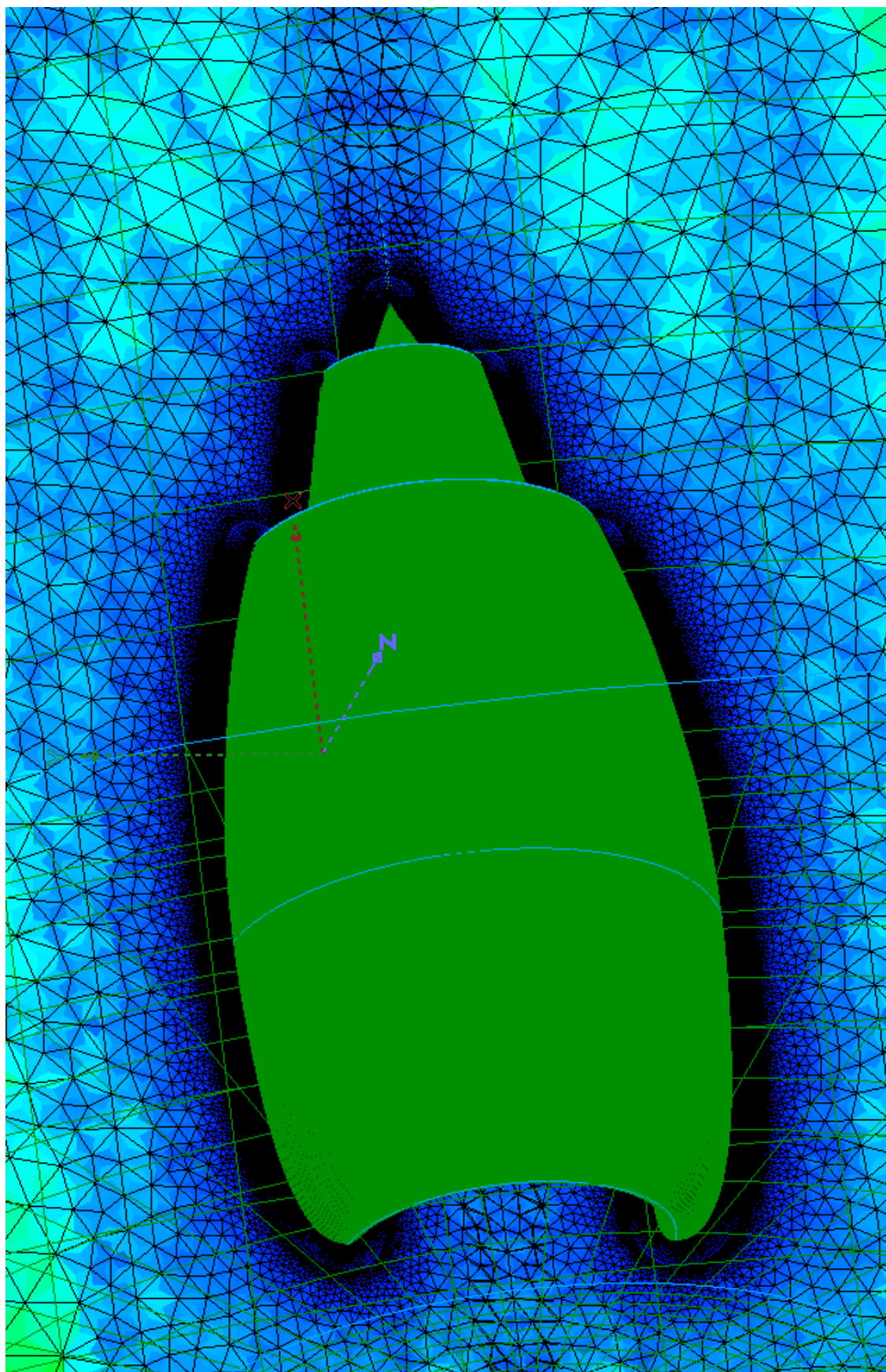


Figure 4.16: *Esempio di mesh generata attorno ad una nacelle*

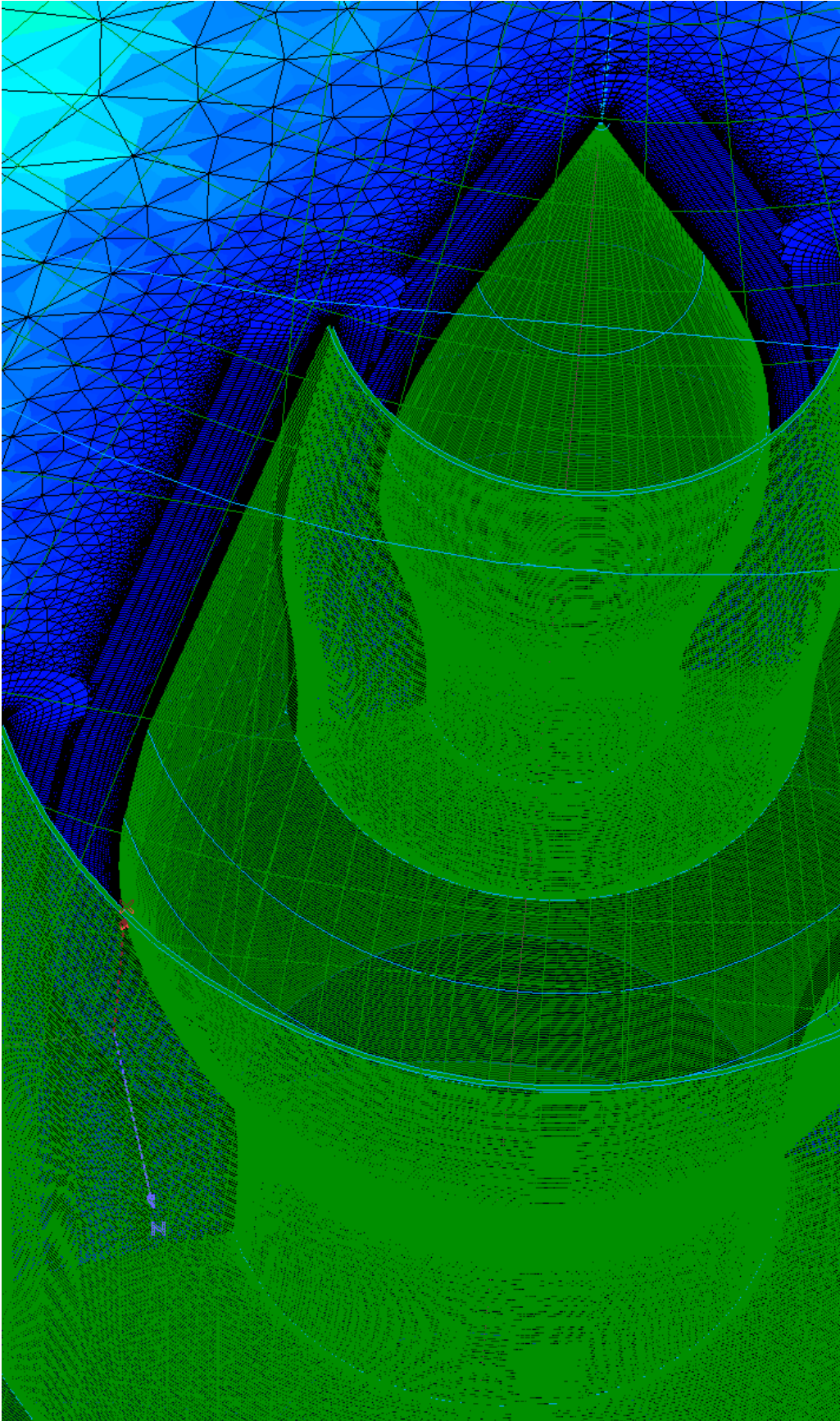


Figure 4.17: *Esempio del boundary layer della mesh generata*

CONCLUSIONS

L'obiettivo di questa tesi è lo sviluppo di un codice di calcolo per la gestione di una nacelle assialsimmetrica a partire dal suo profilo 2D. Lo scopo del codice in questione è la parametrizzazione, il fitting e la sua successiva ottimizzazione della nacelle baseline per il CRM. A differenza della TFN di partenza, si è adottato un approccio basato sulle BC per simulare la presenza del turbofan. L'ottimizzazione ha il fine di minimizzare gli effetti di interferenza e il drag aggiuntivo all'aeroplano causato da essa. L'ottimizzazione prevista è multi-punto per poter testare la nacelle al variare delle condizioni di volo.

Il codice sviluppato è in grado di gestire agilmente le più importanti dimensioni di una nacelle e la sua posizione rispetto al bordo d'attacco dell'ala. Inoltre, permette di controllare la forma di ogni parte "bagnata" dal fluido garantendo la possibilità di controllare lo stato, la tipologia di curva e la complessità della stessa per ognuna delle 13 curve introdotte.

È stata inoltre costruita una nacelle di riferimento, a partire dai dati pubblici del CRM e del ciclo tipico di un UHBR turbofan, per poter indirizzare l'algoritmo genetico e avere un test case con cui confrontare i risultati dell'ottimizzatore.

Per quanto riguarda l'ottimizzazione effettiva non è stato possibile effettuarla se non per valutare la stabilità del codice. In particolare, è stata costruita una procedura automatica per costruire la mesh (attraverso un codice commerciale) del campo fluido attorno alla nacelle e per la sua successiva valutazione con un codice fluidodinamico (anch'esso commerciale). È stata verificata la fruibilità del codice ma non è stata condotta e portata a termine una completa ottimizzazione. Visto il gran numero di variabili coinvolte si consiglia, ad un futuro utilizzatore del codice, di eseguire un DoE (Design of Experiment) per valutare le principali variabili di influenza e successivamente compiere l'ottimizzazione effettiva.

I possibili sviluppi ulteriori di questo codice, oltre all'effettivo utilizzo dello stesso, riguardano in particolare la possibilità di integrare un DoE nel codice stesso. Il principale sviluppo riguarda però la capacità di gestire nacelle dalla 3D non assialsimmetriche e la possibilità di gestire l'inclinazione della stessa. Inoltre, a lungo termine, può essere interessante introdurre la geometria del fan nel modello e non simularlo attraverso le BC.

Appendix A

TANGENTE AD UNA NURBS

Viene qui affrontata la trattazione matematica per ricavare l'equazione che permetta di legare la tangente iniziale e finale di una curva parametrica ai suoi parametri. La trattazione esposta si riferisce ad una curva NURBS ed è stata già affrontata [22], viene qui ripresa ed ampliata.

È possibile esprimere una curva parametrica NURBS $C(u)$ come:

$$C(u) = \frac{w(u) * C(u)}{w(u)} = \frac{A(u)}{w(u)} \quad \text{A.1}$$

Ricordando la formulazione delle curve NURBS nelle Equazioni 2.8 e 2.9 è possibile esprimere una NURBS anche come:

$$C(u) = \frac{\sum_{i=0}^n N_{i,p}(u)w_iP_i}{\sum_{j=0}^n N_{j,p}(u)w_j} \quad \text{A.2}$$

Risulta pertanto evidente il significato assunto dalle variabili $A(u)$ e $w(u)$ nell'Equazione A.1, ossia:

$$A(u) = \sum_{i=0}^n N_{i,p}(u)w_iP_i \quad \text{and} \quad w(u) = \sum_{j=0}^n N_{j,p}(u)w_j \quad \text{A.3}$$

Pertanto è possibile esprimere la derivata una NURBS $C'(u)$, rispetto al parametro u , attraverso la derivata del quoziente come:

$$C'(u) = \frac{w(u)A'(u) - w'(u)A(u)}{w^2(u)} = \frac{w(u)A'(u) - w'(u)w(u)C(u)}{w^2(u)} = \frac{A'(u) - w'(u)C(u)}{w(u)} \quad \text{A.4}$$

Dove $A'(u)$ e $w'(u)$ sono calcolate secondo il seguente schema:

$$A'(u) = \sum_{i=0}^{n-1} N_{i,p-1}(u)Q_i \quad Q_i = p \frac{w_{i+1}P_{i+1} - w_iP_i}{u_{i+p+1} - u_{i+1}} \quad \text{A.5}$$

$$w'(u) = \sum_{i=0}^{n-1} N_{i,p-1}(u)W_i \quad W_i = p \frac{w_{i+1} - w_i}{u_{i+p+1} - u_{i+1}} \quad \text{A.6}$$

Per valutare la derivata iniziale e finale della curva parametrica è sufficiente quindi considerare $C'(0)$ e $C'(1)$. Viene qui proposta la dimostrazione per il primo caso:

$$A'(0) = N_{0,p-1}(0)Q_0 + N_{1,p-1}(0)Q_1 + \cdots + N_{n-2,p-1}(0)Q_{n-2} + N_{n-1,p-1}(0)Q_{n-1} \quad \text{A.7}$$

$$w'(0) = N_{0,p-1}(0)W_0 + N_{1,p-1}(0)W_1 + \cdots + N_{n-2,p-1}(0)W_{n-2} + N_{n-1,p-1}(0)W_{n-1} \quad \text{A.8}$$

Sfruttando le seguenti proprietà delle funzioni base delle curve BSpline [22]:

$$\sum_{i=0}^n N_{i,p}(u) = 1 \quad \forall u \in [0; 1] \quad \text{A.9}$$

$$N_{0,p}(0) = N_{n,p}(1) = 1 \quad \forall n, p \quad \text{A.10}$$

È possibile semplificare le Equazioni A.7 e A.8 come:

$$A'(0) = N_{0,p-1}(0)Q_0 = 1 \cdot Q_0 = p \frac{w_1 P_1 - w_0 P_0}{u_{p+1} - u_1} = p \frac{w_1 P_1 - w_0 P_0}{u_{p+1}} \quad \text{with } u_1 = 0 \quad \text{A.11}$$

$$w'(0) = N_{0,p-1}(0)W_0 = 1 \cdot W_0 = p \frac{w_1 - w_0}{u_{p+1} - u_1} = p \frac{w_1 - w_0}{u_{p+1}} \quad \text{with } u_1 = 0 \quad \text{A.12}$$

Valutando anche il denominatore dell'Equazione A.4 e sfruttando le proprietà precedenti si ottiene:

$$w(0) = N_{0,p-1}(0)w_0 = 1 \cdot w_0 = w_0 \quad \text{A.13}$$

Da cui si ottiene sostituendo in A.4:

$$C'(0) = \frac{p \frac{w_1 P_1 - w_0 P_0}{u_{p+1}} - p \frac{w_1 - w_0}{u_{p+1}} C(0)}{w_0} = p \frac{w_1 P_1 - w_0 P_0 - w_1 C(0) + w_0 C(0)}{u_{p+1} w_0} \quad \text{A.14}$$

Supponendo ora che l'inizio della curva parametrica coincida con il primo punto di controllo si ha $C(0) = P_0$ pertanto si ottiene la formulazione finale:

$$C'(0) = p \frac{w_1 P_1 - w_0 P_0 - w_1 P_0 + w_0 P_0}{u_{p+1} w_0} = p \frac{w_1 P_1 - w_1 P_0}{u_{p+1} w_0} = \frac{p}{u_{p+1}} \frac{w_1}{w_0} (P_1 - P_0) \quad \text{A.15}$$

In modo analogo si ricava per la derivata prima nel punto finale la relazione:

$$C'(1) = \frac{p}{1 - u_{m-p-1}} \frac{w_{n-1}}{w_n} (P_n - P_{n-1}) \quad \text{A.16}$$

La tangente riferita agli assi cartesiani di una funzione bidimensionale parametrica può essere calcolata nel seguente modo:

$$y' = \frac{dy}{dx} = \frac{dy}{du} \frac{du}{dx} = \frac{C_y(u)}{C_x(u)} \quad \text{where } C'(u) = [C_x(u); C_y(u)] \quad \text{A.17}$$

Da cui si ricava, attraverso le Equazioni A.15 e A.16, che la tangente nei punti estremali può essere calcolata come:

$$y'_0 = \frac{C_y(0)}{C_x(0)} = \frac{\frac{p}{u_{p+1}} \frac{w_1}{w_0} (P_{1,y} - P_{0,y})}{\frac{p}{u_{p+1}} \frac{w_1}{w_0} (P_{1,x} - P_{0,x})} = \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \quad \text{A.18}$$

$$y'_1 = \frac{P_{n,y} - P_{n-1,y}}{P_{n,x} - P_{n-1,x}} \quad \text{A.19}$$

Viene di seguito riportato un esempio di curva NURBS e relativa derivata prima in modo da provare graficamente la correttezza della precedente relazione. La curva parametrica, di grado 2, è stata generata con i seguenti parametri.

Parametri	Valori
Coordinate x	6 5 2 -2
Coordinate y	1 3 4 1
Nodi	0 0 0 0.5 1 1 1
Pesi	1 0.7 0.5 1

Table A.1: Parametri usati per generazione di NURBS per esempio di calcolo di derivata prima

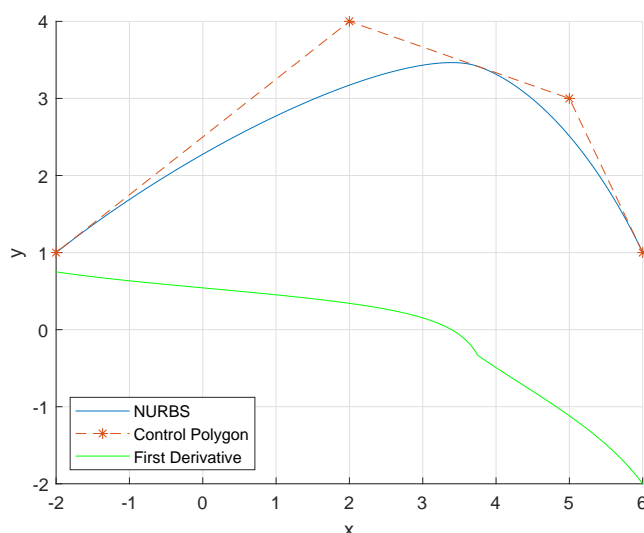


Figure A.1: NURBS di esempio con relativa derivata prima

Dalle Equazioni A.18 e A.19 si può calcolare quindi:

$$y'_0 = \frac{3 - 1}{5 - 6} = \frac{2}{-1} = -2 \quad \text{A.20}$$

$$y'_1 = \frac{4 - 1}{2 - (-2)} = \frac{3}{4} = 0.75 \quad \text{A.21}$$

A.1 CASI PARTICOLARI

Viene di seguito riportato l'unico caso particolare in cui l'Equazione A.18 fallisce. Nel caso in cui si abbia $P_{1,x} = P_{0,x}$, ossia si è in presenza di tangente verticale, l'Equazione

A.18 si riduce a:

$$y'_0 = \infty$$

A.22

Non è possibile determinare il segno della derivata in modo certo, il problema viene affrontato in Appendice B.

Appendix B

CURVATURA DI UNA NURBS

Viene qui affrontata la trattazione matematica per ricavare l'equazione che permette di legare la curvatura iniziale e finale di una curva parametrica ai suoi parametri. La trattazione esposta si riferisce ad una curva NURBS ed è stata già affrontata [22], viene qui ripresa ed ampliata.

Estendendo l'approccio utilizzato in Appendice A è possibile ricavare la formulazione matematica per calcolare la k-esima derivata di una NURBS come:

$$C^{(k)}(u) = \frac{A^{(k)}(u) - \sum_{i=1}^k \binom{k}{i} w^{(k)}(u) C^{(k)-1}(u)}{w(u)} \quad \text{B.1}$$

Dove $A'(u)$ e $w'(u)$ sono calcolate secondo il seguente schema:

$$A^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k} Q_i^{(k)} \quad \text{con} \quad Q_i^{(k)} = \begin{cases} w_i \cdot P_i & \text{if } k = 0 \\ \frac{p-k+1}{u_{i+p+1} - u_{i+k}} (Q_{i+1}^{(k-1)} - Q_i^{(k-1)}) & \text{if } k > 0 \end{cases} \quad \text{B.2}$$

$$w^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k} W_i^{(k)} \quad \text{con} \quad W_i^{(k)} = \begin{cases} w_i & \text{if } k = 0 \\ \frac{p-k+1}{u_{i+p+1} - u_{i+k}} (W_{i+1}^{(k-1)} - W_i^{(k-1)}) & \text{if } k > 0 \end{cases} \quad \text{B.3}$$

Si procede ora alla valutazione della curvatura negli estremi della curva, viene di seguito riportata la dimostrazione per il punto iniziale:

$$C''(0) = \frac{A''(0) - 2w'(0)C'(0) - w''(0)C(0)}{w(0)} \quad \text{B.4}$$

Dove i termini $A''(0)$ e $w''(0)$ possono essere calcolati grazie alle Equazioni B.2 e B.3:

$$\begin{aligned} A''(0) &= \sum_{i=0}^{n-2} N_{i,p-2} Q_i^{(2)} = N_{0,p-2} Q_0^{(2)} + N_{1,p-2} Q_1^{(2)} + \cdots + N_{n-3,p-2} Q_{n-3}^{(2)} + N_{n-2,p-2} Q_{n-2}^{(2)} \\ w''(0) &= \sum_{i=0}^{n-2} N_{i,p-2} W_i^{(2)} = N_{0,p-2} W_0^{(2)} + N_{1,p-2} W_1^{(2)} + \cdots + N_{n-3,p-2} W_{n-3}^{(2)} + N_{n-2,p-2} W_{n-2}^{(2)} \end{aligned} \quad \text{B.5}$$

Grazie alle proprietà A.9 e A.10 si ottiene in B.5:

$$\begin{aligned}
A''(0) &= N_{0,p-2}Q_0^{(2)} = 1 \cdot Q_0^{(2)} = \frac{p-1}{u_{p+1}-u_2} (Q_1^{(1)} - Q_0^{(1)}) \\
&= \frac{p-1}{u_{p+1}-u_2} \left(\left(\frac{p}{u_{p+2}-u_2} (Q_2^{(0)} - Q_1^{(0)}) \right) - \left(\frac{p}{u_{p+1}-u_2} (Q_1^{(0)} - Q_0^{(0)}) \right) \right) \\
&= \frac{p-1}{u_{p+1}-u_2} \left(\left(\frac{p}{u_{p+2}-u_2} (w_2P_2 - w_1P_1) \right) - \left(\frac{p}{u_{p+1}-u_2} (w_1P_1 - w_0P_0) \right) \right) \\
w''(0) &= N_{0,p-2}W_0^{(2)} = 1 \cdot W_0^{(2)} = \frac{p-1}{u_{p+1}-u_2} (W_1^{(1)} - W_0^{(1)}) \\
&= \frac{p-1}{u_{p+1}-u_2} \left(\left(\frac{p}{u_{p+2}-u_2} (W_2^{(0)} - W_1^{(0)}) \right) - \left(\frac{p}{u_{p+1}-u_2} (W_1^{(0)} - W_0^{(0)}) \right) \right) \\
&= \frac{p-1}{u_{p+1}-u_2} \left(\left(\frac{p}{u_{p+2}-u_2} (w_2 - w_1) \right) - \left(\frac{p}{u_{p+1}-u_2} (w_1 - w_0) \right) \right)
\end{aligned} \tag{B.6}$$

Supponendo che il punto iniziale della curva parametrica coincida con il primo punto di controllo, per cui si ha $(C(\theta) = P_0)$, e assumendo che la curva abbia grado $p \geq 2$, viceversa non avrebbe senso calcolare la derivata seconda, si ha che $u_2 = 0$ pertanto le precedenti equazioni possono essere riscritte come:

$$\begin{aligned}
A''(0) &= \frac{p(p-1)}{u_{p+1}} \left(\frac{w_2P_2 - w_1P_1}{u_{p+2}} - \frac{w_1P_1 - w_0P_0}{u_{p+1}} \right) \\
w''(0) &= \frac{p(p-1)}{u_{p+1}} \left(\frac{w_2 - w_1}{u_{p+2}} - \frac{w_1 - w_0}{u_{p+1}} \right)
\end{aligned} \tag{B.7}$$

Sostituendo quanto ricavato nell'Equazione B.4 e grazie alle precedenti Equazioni A.11, A.12, A.13 e A.15 si ottiene:

$$\begin{aligned}
C'''(0) &= \frac{\frac{p(p-1)}{u_{p+1}} \left(\frac{w_2P_2 - w_1P_1}{u_{p+2}} - \frac{w_1P_1 - w_0P_0}{u_{p+1}} \right)}{w_0} - 2 \frac{\frac{w_1 - w_0}{u_{p+1}} \frac{p}{u_{p+1}} \frac{w_1}{w_0} (P_1 - P_0)}{w_0} - \\
&\quad \frac{\frac{p(p-1)}{u_{p+1}} \left(\frac{w_2 - w_1}{u_{p+2}} - \frac{w_1 - w_0}{u_{p+1}} \right) P_0}{w_0}
\end{aligned} \tag{B.8}$$

Raccogliendo a fattor parziale i punti di controllo si ottiene:

$$\begin{aligned}
C'''(0) &= \frac{p \cdot P_0}{w_0 \cdot u_{p+1}} \cdot \left(w_0 \frac{p-1}{u_{p+1}} + 2p \frac{w_1 - w_0}{u_{p+1}} \frac{w_1}{w_0} - (p-1) \left(\frac{w_2 - w_1}{u_{p+2}} - \frac{w_1 - w_0}{u_{p+1}} \right) \right) \\
&\quad - \frac{p \cdot P_1}{w_0 \cdot u_{p+1}} \cdot \left(w_1(p-1) \left(\frac{1}{u_{p+2}} - \frac{1}{u_{p+1}} \right) + 2p \frac{w_1 - w_0}{u_{p+1}} \frac{w_1}{w_0} \right) \\
&\quad + \frac{p \cdot P_2}{w_0 \cdot u_{p+1}} \cdot w_2 \frac{p-1}{u_{p+2}}
\end{aligned} \tag{B.9}$$

La precedente relazione può essere riscritta in modo compatto come:

$$C'''(0) = \frac{p \cdot P_0}{w_0 \cdot u_{p+1}} \cdot A - \frac{p \cdot P_1}{w_0 \cdot u_{p+1}} \cdot B + \frac{p \cdot P_2}{w_0 \cdot u_{p+1}} \cdot C \tag{B.10}$$

Per ricavare ora la curvatura riferita agli assi cartesiani di una funzione bidimensionale parametrica si può adottare il seguente approccio come nell'Equazione A.17:

$$\begin{aligned}
y'' &= \frac{d_2 y}{dx_2} = \frac{d'y}{dx} = \frac{d'y}{du} \frac{du}{dx} = \frac{d}{du} \left(\frac{dy}{du} \frac{du}{dx} \right) \frac{du}{dx} = \frac{d}{du} \left(C'_y \frac{1}{C'_x} \right) \frac{1}{C'_x} \\
&= \frac{dC'_y}{du} \frac{1}{(C'_x)^2} + \frac{C'_y}{C'_x} \cdot \frac{1}{du} \left(\frac{1}{C'_x} \right) = \frac{C''_y}{(C'_x)^2} + \frac{C'_y}{C'_x} \cdot \frac{0 \cdot C'_x - 1 \cdot C''_x}{(C'_x)^2} \\
&= \frac{C''_y + \frac{C'_y}{C'_x} \cdot (-C''_x)}{(C'_x)^2} = \frac{C''_y - C''_x \cdot y'}{(C'_x)^2}
\end{aligned} \tag{B.11}$$

Sostituendo nella precedente l'equazione B.10 si ottiene:

$$\begin{aligned}
y''_0 &= \frac{p \cdot (P_{0,y} - y'_0 P_{0,x})}{w_0 \cdot u_{p+1}} \cdot \frac{A}{(C'_x(0))^2} - \frac{p \cdot (P_{1,y} - y'_0 P_{1,x})}{w_0 \cdot u_{p+1}} \cdot \frac{B}{(C'_x(0))^2} + \\
&+ \frac{p \cdot (P_{2,y} - y'_0 P_{2,x})}{w_0 \cdot u_{p+1}} \cdot \frac{C}{(C'_x(0))^2}
\end{aligned} \tag{B.12}$$

Raccogliendo i termini in comune si ottiene:

$$y''_0 = \frac{p \cdot ((P_{0,y} - y'_0 P_{0,x}) \cdot A - (P_{1,y} - y'_0 P_{1,x}) \cdot B + (P_{2,y} - y'_0 P_{2,x}) \cdot C)}{w_0 \cdot u_{p+1} \cdot (C'_x(0))^2} \tag{B.13}$$

Sostituendo nella precedente l'equazione A.15 si ottiene:

$$\begin{aligned}
y''_0 &= \frac{p \cdot (AP_{0,y} - y'_0 AP_{0,x} - BP_{1,y} - y'_0 BP_{1,x} + CP_{2,y} - y'_0 CP_{2,x})}{w_0 \cdot u_{p+1} \cdot \left(\frac{p}{u_{p+1} w_0} (P_{1,x} - P_{0,x}) \right)^2} \\
&= \frac{w_0 \cdot u_{p+1} \cdot (AP_{0,y} - y'_0 AP_{0,x} - BP_{1,y} + y'_0 BP_{1,x} + CP_{2,y} - y'_0 CP_{2,x})}{p \cdot (w_1 (P_{1,x} - P_{0,x}))^2} \\
&= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{AP_{0,y} - y'_0 AP_{0,x} - BP_{1,y} + y'_0 BP_{1,x} + CP_{2,y} - y'_0 CP_{2,x}}{(P_{1,x} - P_{0,x})^2}
\end{aligned} \tag{B.14}$$

Notando come nell'Equazione A.18 la tangenza dipenda dalla differenza tra le coordinate dei punti, si procede ad un cambio di variabili. Si effettua l'operazione di traslazione riportata qui sotto:

$$\begin{cases} \Delta x_i = P_{i,x} - P_{0,x} \\ \Delta y_i = P_{i,y} - P_{0,y} \end{cases} \tag{B.15}$$

Questa traslazione consente di spostare l'origine degli assi cartesiani nel punto P_0 e la posizione dei punti successivi viene misurata come distanza da questo punto. La traslazione B.15 consente di ottenere le nuove coordinate dei punti come:

$$\begin{cases} \Delta x_0 = P_{0,x} - P_{0,x} = 0 \\ \Delta y_0 = P_{0,y} - P_{0,y} = 0 \\ \Delta x_1 = P_{1,x} - P_{0,x} \\ \Delta y_1 = P_{1,y} - P_{0,y} \\ \Delta x_2 = P_{2,x} - P_{0,x} \\ \Delta y_2 = P_{2,y} - P_{0,y} \end{cases} \tag{B.16}$$

Sostituendo le nuove coordinate nelle Equazioni A.18 e B.14 si ottiene:

$$y'_0 = \frac{\Delta y_1 - \Delta y_0}{\Delta x_1 - \Delta x_0} = \frac{\Delta y_1}{\Delta x_1} \quad \text{B.17}$$

$$\begin{aligned} y''_0 &= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{A\Delta y_0 - y'_0 A\Delta x_0 - B\Delta y_1 + y'_0 B\Delta x_1 + C\Delta y_2 - y'_0 C\Delta x_2}{(\Delta x_1 - \Delta x_0)^2} \\ &= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{-B\Delta y_1 + y'_0 B\Delta x_1 + C\Delta y_2 - y'_0 C\Delta x_2}{(\Delta x_1)^2} \end{aligned} \quad \text{B.18}$$

Sostituendo il valore della tangenza ricavato B.17 in B.19, con le opportune semplificazioni, si ottiene:

$$\begin{aligned} y''_0 &= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{-B\Delta y_1 + \frac{\Delta y_1}{\Delta x_1} B\Delta x_1 + C\Delta y_2 - \frac{\Delta y_1}{\Delta x_1} C\Delta x_2}{\Delta x_1^2} \\ &= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{-B\Delta y_1 + B\Delta y_1 + C\Delta y_2 - \frac{\Delta y_1}{\Delta x_1} C\Delta x_2}{\Delta x_1^2} \\ &= \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{C\Delta y_2 - \frac{\Delta y_1}{\Delta x_1} C\Delta x_2}{\Delta x_1^2} = C \cdot \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{\Delta y_2 - \frac{\Delta y_1}{\Delta x_1} \Delta x_2}{\Delta x_1^2} \\ &= w_2 \frac{p-1}{u_{p+2}} \cdot \frac{w_0 \cdot u_{p+1}}{p \cdot w_1^2} \cdot \frac{\Delta y_2 - \frac{\Delta y_1}{\Delta x_1} \Delta x_2}{\Delta x_1^2} \\ &= \frac{w_2 \cdot w_0}{w_1^2} \cdot \frac{p-1}{p} \cdot \frac{u_{p+1}}{u_{p+2}} \cdot \frac{\Delta x_2}{\Delta x_1^2} \cdot \left(\frac{\Delta y_2}{\Delta x_2} - \frac{\Delta y_1}{\Delta x_1} \right) \end{aligned} \quad \text{B.19}$$

Eseguito il cambio di coordinate B.15 a ritroso si ottiene quindi la forma finale:

$$y''_0 = \frac{w_2 \cdot w_0}{w_1^2} \cdot \frac{p-1}{p} \cdot \frac{u_{p+1}}{u_{p+2}} \cdot \frac{P_{2,x} - P_{0,x}}{(P_{1,x} - P_{0,x})^2} \cdot \left(\frac{P_{2,y} - P_{0,y}}{P_{2,x} - P_{0,x}} - \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \right) \quad \text{B.20}$$

In modo del tutto analogo si ottiene la seguente relazione per la derivata seconda nel punto finale:

$$y''_1 = \frac{w_{n-2} \cdot w_n}{w_{n-1}^2} \cdot \frac{p-1}{p} \cdot \frac{1-u_{m-p-1}}{1-u_{m-p-2}} \cdot \frac{P_{n-2,x} - P_{n,x}}{(P_{n-1,x} - P_{n,x})^2} \cdot \left(\frac{P_{n-2,y} - P_{n,y}}{P_{n-2,x} - P_{n,x}} - \frac{P_{n-1,y} - P_{n,y}}{P_{n-1,x} - P_{n,x}} \right) \quad \text{B.21}$$

Viene di seguito riportato un esempio di curva NURBS e relativa derivata prima e seconda in modo da provare graficamente la correttezza della precedente relazione. La curva parametrica, di grado 2, è stata generata con i parametri forniti precedentemente Tabella A.1.

Dalle Equazioni B.20 e B.21 si può calcolare quindi:

$$y''_0 = \frac{0.5 \cdot 1}{0.7^2} \cdot \frac{2-1}{2} \cdot \frac{0.5}{1} \cdot \frac{2-6}{(5-6)^2} \cdot \left(\frac{4-1}{2-6} - \frac{3-1}{5-6} \right) = -\frac{125}{98} \quad \text{B.22}$$

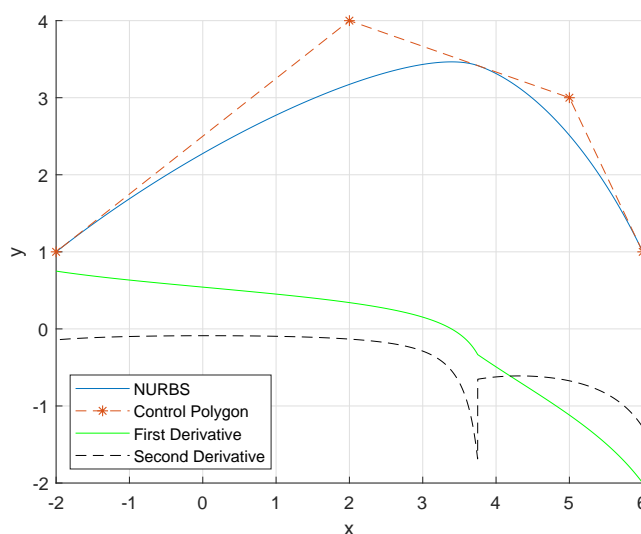


Figure B.1: *NURBS di esempio con relativa derivata prima e seconda*

$$y_1'' = \frac{0.7 \cdot 1}{0.5^2} \cdot \frac{2-1}{2} \cdot \frac{1-0.5}{1-0} \cdot \frac{5-(-2)}{(2-(-2))^2} \cdot \left(\frac{3-1}{5-(-2)} - \frac{4-1}{2-(-2)} \right) = -\frac{91}{640} \quad \text{B.23}$$

Si può notare, nella Figura B.1, che è presente un punto in cui la derivata seconda ha una discontinuità, ad $x \sim 3.75$. Il motivo di questo fatto è legato alle proprietà delle NURBS [22]. Una curva NURBS ha tutte le derivate continue nell'intervallo tra due nodi, mentre la curva è differenziabile $p - k$ volte in un nodo, dove k è la molteplicità del nodo stessa.

Essendo quindi la curva in questione di grado $p = 2$ ed essendo il nodo centrale di molteplicità $k = 1$, si ha che la NURBS rappresentata in Figura B.1 nel nodo centrale è differenziabile $p - k = 1$ volte, pertanto il punto $x \sim 3.75$ corrisponde esattamente al nodo centrale della NURBS in quanto la derivata seconda non è continua.

B.1 CASI PARTICOLARI

Vengono di seguito riportate alcune formule semplificate per i casi particolari della relazione B.20. Nel caso in cui si abbia $P_{2,x} = P_{0,x}$ l'Equazione B.20 si riduce a:

$$y_0'' = \frac{w_2 \cdot w_0}{w_1^2} \cdot \frac{p-1}{p} \cdot \frac{u_{p+1}}{u_{p+2}} \cdot \frac{P_{2,y} - P_{0,y}}{(P_{1,x} - P_{0,x})^2} \quad \text{B.24}$$

Nel caso invece in cui si abbia $P_{1,x} = P_{0,x}$, ossia si è in presenza di tangente verticale, l'Equazione B.20 si riduce a:

$$y_0'' = y_0' = \pm\infty \quad \text{B.25}$$

Per valutare il segno corretto di entrambe le derivate è necessario valutare la seguente espressione:

$$\text{sgn}(-w_2 \cdot w_0 \cdot (P_{2,x} - P_{0,x}) \cdot (P_{1,y} - P_{0,y})) \quad \text{B.26}$$

L'Equazione B.20 si riduce alla seguente per curve BSpline:

$$y_0'' = \frac{p-1}{p} \cdot \frac{u_{p+1}}{u_{p+2}} \cdot \frac{P_{2,x} - P_{0,x}}{(P_{1,x} - P_{0,x})^2} \cdot \left(\frac{P_{2,y} - P_{0,y}}{P_{2,x} - P_{0,x}} - \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \right) \quad \text{B.27}$$

La quale si riduce ulteriormente alla seguente per curve Bézier, grazie all'Equazione 2.4:

$$y_0'' = \frac{n-2}{n-1} \cdot \frac{P_{2,x} - P_{0,x}}{(P_{1,x} - P_{0,x})^2} \cdot \left(\frac{P_{2,y} - P_{0,y}}{P_{2,x} - P_{0,x}} - \frac{P_{1,y} - P_{0,y}}{P_{1,x} - P_{0,x}} \right) \quad \text{B.28}$$

Le formule per il punto finale vengono derivate da B.21 e sono speculari.

Appendix C

CODICE DI CALCOLO PER CURVE NURBS

Viene qui riportato il codice di calcolo utilizzato per ottenere le coordinate 2D di una NURBS a partire dai suoi parametri tipici di input.

```
function [C] = NURBS(NP, Points ,U, p, Weight)
%NURBS is a function to calculate the points (x,y) of a NURBS.
%The endpoint of the curve is automatically included.
%
%Input:
% -NP = number of points to construct the curve
% -Points = matrix of control points
%         every row corrispond to a point
%         first column corrispond to x coordinates
%         second column corrispond to y coordinates
% -U = column vector of the knots
% -p = degree of the curve
% -Weight = column vector of the weights
%Output:
% -C = is the matrix with the curve

%Creation of the vector containing the normalized parametric
%points to be evaluated on the NURBS
u=linspace(0,1,NP);
%Number of curve control points
n=size(Points,1);

%Creation of the bspline basic functions needed
[N] = BSpline_basis_function(u,U',p);

%Creation of a cell to inserted:
% first column = numerator of the corresponding R(i,p)
% second column = denominator of the corresponding R(i,p)
R = cell(n,2);

%Creation of the NURBS basic functions R(i,p) for each weight
for i = 1:n
```

```

    %numerator of the NURBS basic function i
    Num=N{i ,end}*Weight(i);

    %denominator of the NURBS basic function i
    Den=zeros(1,NP);
    %for all weights
    for j=1:n
        Den=Den+N{j ,end}*Weight(j);
    end

    %saving of basic function components
    R{i ,1}=Num;
    R{i ,2}=Den;
end

%Preallocation of the NURBS curve
C=zeros(NP,2);

%Creation of the NURBS curve
for i=1:n
    %values of the corresponding NURBS basic functions
    Num=R{i ,1};
    Den=R{i ,2};

    %evaluate each point of the vector u
    for k = 1:NP
        C(k,:) = Points(i,:) * Num(k) / Den(k) + C(k,:);
    end
end
end
end

```

C.1 BSPLINE BASIS FUNCTION

Viene di seguito riportato il codice di calcolo utilizzato per generare le funzioni base di una BSpline richiamato nel precedente codice. [22]

```

function [N] = BSpline_basis_function(u,U,p)
%B_SPLINE_BASIS_FUNCTION is a function to calculate the basic
%functions N(i,p) of a BSpline where i is the involved knots and p
%is the degree of the calculated basic function
%
%Input:

```

```

% -u = row vector of the parametric points on which calculate the
%       basis functions
% -U = column vector of the nodes
% -p = degree of the curve
%Output:
% -N = cell with all the B-spline basis functions

%Let NP be the number of points to be evaluated
NP = size(u,2);
%Number of curve knots
M = size(U,2);

%Preallocate a cell variable to store, in each cell, the matrix
%corresponding to the B-spline basis function N(i,p)
N = cell(M-1,p+1);

%% — Calculation of the basic functions like N(i,0)
%Basis functions are saved in the cell as N{i+1,1} since the
%first index for Matlab is 1

%for all the basis function like N(i,0)
for i=1:M-1
    %preallocation of a line vector to compute N(i,0)
    Ni0=zeros(1,NP);

    %assignment of knots involved in the evaluation
    u_starting=U(i);
    u_ending=U(i+1);

    %gain the points that are between the two selected knots
    state=(u>=u_starting & u<u_ending);

    %update the N(i,0) basis function setting the points between
    %active knots equal to 1
    Ni0(state)=1;

    %allocation of N(i,0) in the cell N
    N{i,1}=Ni0;
end

%% — Inclusion of the ending point in the basis functions
%find out which are the final nodes
last_knots=find(U==1);

%get the previous N(i,0)

```

```

Ni0_previous=N{last_knots(1)-1,1};

%overwrite the its last point
Ni0_previous(end)=1;

%save it in the cell N
N{last_knots(1)-1,1}=Ni0_previous;

%% — Calculation of the basic functions  $N(i,p)$ 
%basic functions  $N(i,p)$  are saved in the cell as  $N\{i+1,p+1\}$  since
%the first index for Matlab is 1

%for each basis function up to the max degree
for P=1:p
    %basic functions are evaluated with grades greater than 0.
    %Gradually as the grade increases, the number of functions to
    %be calculated decreases

    %for each basic function up to that permitted by the degree P
    for i=1:M-1-P

        %assignment of knots involved in the evaluation
        u_i1=U(i);
        u_i2=U(i+1);
        u_p1=U(i+P);
        u_p2=U(i+P+1);

        %compute coefficient based on knots differences
        a=u_p1-u_i1;
        b=u_p2-u_i2;

        %the necessary basic functions of a lower grade are called
        NiP_previous=N{i,P};
        NiP_next=N{i+1,P};

        %the basis function  $N(i,P)$  is calculated by subdividing it
        %into two parts to be able to check any relations 0/0
        NiP_part1=(u-u_i1)/a.*NiP_previous;
        NiP_part2=(u_p2-u)/b.*NiP_next;

        %in the case in which we have 0/0, that is NaN, replace
        %with 0 [20]
        check_part1=isnan(NiP_part1);
        NiP_part1(check_part1)=0;
        check_part2=isnan(NiP_part2);
        NiP_part2(check_part2)=0;
    end
end

```

```
%join the two parts of N(i,p)  
NiP=NiP_part1+NiP_part2;  
  
%save it in the cell N  
N{i ,P+1}=NiP;  
end  
end  
end
```


Appendix D

CODICE PER COSTRUIRE UN VINCOLO DI DISUGUAGLIANZA GENERICO

Viene di seguito riportato il codice per aggiornare la matrice dei vincoli A e il vettore b per un vincolo generico.

```
function [A,b] = constraint_builder(A,b,k,state,value,j,q)
%COSTRAINT_BUILDER is a function to build a constraint, like
% variable  $q \leq$  variable  $j$ , for the optimization of a nacelle
%
%Input:
%  $A$  = inequality constraint matrix to update
%  $b$  = inequality constraint vector to update
%  $k$  = number of the constraint
%  $state$  = vector with state of all dimension variables
%  $value$  = vector with value of all dimension variables
%  $j$  = right variable considered
%  $q$  = left variable considered
%
%Output:
%  $A$  = inequality constraint matrix updated
%  $b$  = inequality constraint vector updated

%% — UPDATE VARIABLE J

%is there a left variable?
if isempty(j)==0
    %yes, so update it

    %is variable  $j$  a curve variable or is a dimension
    % variable active?
    if j>36 || state(j)==1
        %yes, it is

        %write the value  $-1$ , in line  $k$  and in column  $j$ , of the
        %constraint matrix  $A$ 
        A(k,j)=-1;
    else
```

```
    %no, it isn't

    %add to the constraint vector b, in line k, the
    %value of dimension j
    b(k)=b(k)+value(j);
end
end

%% — UPDATE VARIABLE Q

%is there a right variable?
if isempty(q)==0
    %yes, so update it

    %is variable q a curve variable or is a dimension
    % variable active?
    if q>36 || state(q)==1
        %yes, it is

        %write the value 1 in line k and in column q of the
        %constraint matrix A
        A(k,q)=1;
    else
        %no, it isn't

        %subtract to the constraint vector b, in line k, the
        %value of dimension q
        b(k)=b(k)-value(q);
    end
end
end
end
```


BIBLIOGRAPHY

- [1] Wikipedia contributors. Heinkel he 176 — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Heinkel_He_176&oldid=925350708.
- [2] Wikipedia contributors. General electric tf39 — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=General_Electric_TF39&oldid=917538619.
- [3] Advisory Group for Aerospace Research and Development (AGARD). Special course on Subsonic/Transonic Aerodynamic Interference for Aircraft. *Report for NATO Science and Technology Organization*, January 1983.
- [4] Vassberg JC. et al. Development of a Common Research Model for Applied CFD Validation Studies. *ReportNo. AIAA.2008-6919*.
- [5] Ashley Dittberner Melissa B. Rivers. Experimental Investigation of the NASA Common Research Model (Invited). *28th AIAA Applied Aerodynamics Conference 28 June - 1 July 2010, Chicago, Illinois*.
- [6] Ashley Dittberner Melissa B. Rivers. Experimental Investigations of the NASA Common Research Model in the NASA Langley National Transonic Facility and NASA Ames 11-Ft Transonic Wind Tunnel (Invited). *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition 4 - 7 January 2011, Orlando, Florida*.
- [7] NASA, AIAA. 3rd aiaa cfd high lift prediction workshop (hilftpw-3). <https://hilftpw.larc.nasa.gov/index-workshop3.html>.
- [8] NASA, AIAA. 4th aiaa cfd drag prediction workshop. <https://aiaa-dpw.larc.nasa.gov/Workshop4/workshop4.html>.
- [9] NASA, AIAA. 5th aiaa cfd drag prediction workshop. <https://aiaa-dpw.larc.nasa.gov/Workshop5/workshop5.html>.
- [10] NASA, AIAA. 6th aiaa cfd drag prediction workshop. <https://aiaa-dpw.larc.nasa.gov/>.
- [11] Christopher TJ Sheaf Robert Christie Tomasz P. Stańkowski, David G. MacManus. Aerodynamics of aero engine installation. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering December 2016, vol.230, no.14 pp2673-2692*.
- [12] David G. MacManus Michal Slaby John J. Otter, Ioannis Goulos. Aerodynamic Analysis of Civil Aero-Engine Exhaust Systems using Computational Fluid Dynamics. *Journal of Propulsion and Power, Vol. 34, No. 5 (2018), pp. 1152-1165*.
- [13] Matthew Robinson Christopher T. Sheaf Tomasz P. Stankowski, David G. MacManus. The aerodynamic effects of VHBR engine installation to the Common Research Model. *Journal of Aircraft, Vol. 54, Issue 6, 2017, pp. 2270-2284*.
- [14] John Otter David MacManus Nicholas Grech Christopher Sheaf Ioannis Goulos, Tomasz Stankowski. Aerodynamic Design of Separate-Jet Exhausts for Future Civil Aero-engines—Part I: Parametric Geometry Definition and Computational Fluid Dynamics Approach. *Journal of Engineering for Gas Turbines and Power, August 2016, Vol. 138*.

- [15] John Otter David MacManus Nicholas Grech Christopher Sheaf Ioannis Goulos, Tomasz Stankowski. Aerodynamic Design of Separate-Jet Exhausts for Future Civil Aero-engines—Part II: Design Space Exploration, Surrogate Modeling, and Optimization. *Journal of Engineering for Gas Turbines and Power*, August 2016, Vol. 138.
- [16] Tomasz Stankowski David MacManus Nicholas Grech Christopher Sheaf Ioannis Goulos, John Otter. Design Optimisation of Separate-Jet Exhausts for the Next Generation of Civil Aero-Engines. *Proceedings of 23rd International Symposium for Air-Breathing Engines - ISABE 2017, 4-8 September 2017, Manchester, UK*.
- [17] David MacManus Philip Woodrow Christopher Sheaf Ioannis Goulos, Tomasz Stankowski. Civil Turbofan Engine Exhaust Aerodynamics: Impact of Bypass Nozzle After-body Design. *Aerospace Science and Technology, Volume 73, February 2018*.
- [18] Kelvin Richards Christopher Sheaf Matthew H. Robinson, David G. MacManus. Short and slim nacelle design for ultra high BPR engines. *55th AIAA Aerospace Sciences Meeting, AIAA SciTech Forum, (AIAA 2017-0707)*.
- [19] Alexander Heidebrecht Nicholas Grech Matthew H. Robinson, David G. MacManus. An optimization method for nacelle design. *55th AIAA Aerospace Sciences Meeting, AIAA SciTech Forum, (AIAA 2017-0708)*.
- [20] NASA. Common research model. <https://commonresearchmodel.larc.nasa.gov/>.
- [21] Stefan Keye et al. Edward N. Tinoco, Olaf P. Brodersen. Summary of Data from the Sixth AIAA CFD Drag Prediction Workshop: CRM Cases 2 to 5. *American Institute of Aeronautics and Astronautics*.
- [22] Wayne Tiller Les Piegl. The NURBS Book. *Second Edition*, 1997.
- [23] NASA, AIAA. 6th aiaa cfd drag prediction workshop. <https://aiaa-dpw.larc.nasa.gov/Workshop6/forms/DataForm.html>.
- [24] David L. Daggett, Stephen T. Brown, and Ron T. Kawai. Ultra-efficient engine diameter study. *NASA/CR—2003-212309*. <https://ntrs.nasa.gov/search.jsp?R=20030061085>.
- [25] Mark H. Waters and Edward T. Schairer. Analysis of turbofan propulsion system weight and dimensing. *NASA TM X- 73,199*, January 1977.
- [26] Robert Christie, Alexander Heidebrecht, David MacManus. An automated approach to nacelle parameterization using intuitive class shape transformation curves. *Journal of Engineering for Gas Turbines and Power, Volume 139, Issue 6, 062601-1-9*.
- [27] Andreas Petrusson. Aerodynamic evaluation of nacelles for engines with ultra high bypass ratio. *Chalmers University Of Technology, Göteborg, Sweden 2017*.
- [28] Andrey Savelyev, Nikolay Zlenko, Evgeniy Matyash, Sergey Mikhaylov and Andrey Shenkin. Optimal design and installation of ultra high bypass ratio turbofan nacelle. *Central Aerohydrodynamic Institute (TsAGI), Moscow*.
- [29] Odis C. Pendergraft Jr, Anthony M. Ingraldi, Richard J. Re, Timmy T. Kariya. Installation effects od wing-mounted turbofan nacelle-pylons on a 1/17-scale, twin-engine, low-wing transport model. *NASA Technical Paper 3168*, March 1992.

-
- [30] Christopher M. Heath, Justin S. Gray, Michael A. Park, Eric J. Nielsen, and Jan-Renee Carlson. Aerodynamic shape optimization of a dual-stream supersonic plug nozzle. January 2015.
- [31] Jingjing Chen, Yadong Wu, Zhonglin Wang, Anjenq Wang. Nacelle: Air intake aerodynamic design and inlet compatibility. *Proceedings of the ASME 2014 Gas Turbine India Conference*, December 15-17, 2014, New Delhi, India.
- [32] M. J. Langley. The design of axisymmetric cowls for podded nacelles for high by-pass ratio turbofan engines. *Aircraft Research Association Ltd, Bedford*, May 1979.
- [33] NASA, AIAA. 3rd aiaa cfd high lift prediction workshop (hiliftpw-3) geometries. <https://hiliftpw.larc.nasa.gov/Workshop3/geometries.html>.
- [34] Wikipedia contributors. Top-down and bottom-up design — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Top-down_and_bottom-up_design&oldid=923927690.
- [35] MATLAB. Figure properties. <https://it.mathworks.com/help/matlab/ref/matlab.ui.figure-properties.html>.
- [36] MATLAB. Gamultiobj. <https://it.mathworks.com/help/gads/gamultiobj.html>.
- [37] NASA, AIAA. 6th aiaa cfd drag prediction workshop - grids. <https://aiaa-dpw.larc.nasa.gov/Workshop6/grids.html>.
- [38] Pointwise, Inc. Pointwise, inc. - glyph. <https://www.pointwise.com/glyph>.

RINGRAZIAMENTI

Voglio ringraziare il Professore Ernesto Benini per la possibilità che mi ha dato con questa tesi e il supporto che mi ha fornito nel corso di questi mesi. Ringrazio, inoltre, l'azienda Hit09 e l'Ingegnere Rita Ponza per la collaborazione che mi hanno dato. Un grazie in particolare va anche ad Andrea Dal Monte ed ad Andrea Magrini per il loro aiuto e la loro pazienza continua nelle varie fasi del lavoro.

Ringrazio anche tutte quelle persone, amici e familiari, che mi hanno accompagnato in questi anni di università. In particolare, voglio ringraziare i miei genitori Ivano e Manuela, i quali mi hanno sempre spronato a dare il meglio di me ed a non accontentarmi mai dei risultati raggiunti. Infine, ringrazio Ivy, La donna che negli ultimi anni, giorno dopo giorno, ha riempito la mia vita di emozioni e che, passo dopo passo assieme, mi sta trasformando in uomo.