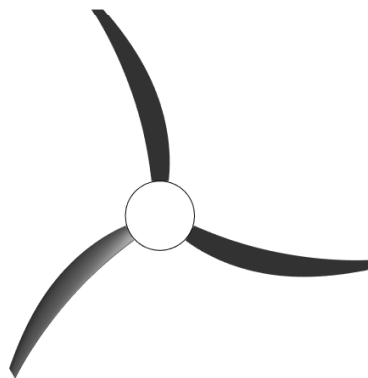


Esercitazioni in **Aerodinamica** **dell'Ala Rotante**



Giosuè Longobardo

Emanuele Sticchi

Università degli Studi di Napoli Federico II

a.a. 2021/2022

I	L'elica	11
1	Introduzione	13
2	Le prestazioni dell'elica	15
2.1	La geometria	15
2.2	L'aerodinamica	17
2.3	Le prestazioni (modello aerodinamico semplificato)	19
2.3.1	Le distribuzioni sulla pala	19
2.3.2	Le curve caratteristiche	22
2.4	Le prestazioni (modello aerodinamico completo)	24
2.4.1	Le distribuzioni sulla pala	24
2.4.2	Le curve caratteristiche	27
3	Il progetto dell'elica	29
3.1	Il progetto dell'elica per il Tecnam P92	29
3.1.1	Analisi delle prestazioni	31
II	McDonnell-Douglas AH-64	37
4	Introduzione	39
5	Le caratteristiche tecniche	41
5.1	L'aerodinamica	42
5.2	Il sistema propulsivo	45
6	L'aerodinamica del rotore	47
6.1	Prestazioni in salita assiale e <i>hover</i>	47
6.1.1	Le polari e altre curve caratteristiche	47
6.1.2	Le distribuzioni dei coefficienti spinta e coppia sulla pala	49
6.2	Volo traslato	50
6.2.1	Prestazioni	50
6.2.2	I coefficienti di flappeggio	54
6.2.3	Le distribuzioni di angoli di attacco sul disco	55
6.2.4	Le distribuzioni del numero di Mach sul disco	57
6.2.5	Il sentiero di stallo	61

7	Le prestazioni	63
7.1	Le curve di potenza	63
7.2	Le prestazioni dell'elicottero	68
7.2.1	Le velocità caratteristiche	68
III	Gli aeromotori	75
8	Il progetto della turbina eolica	77
8.1	Introduzione & stime iniziali	77
8.2	L'aerodinamica	78
8.3	La geometria	79
8.4	Le prestazioni	81
8.5	Una procedura di <i>design</i> alternativa	82
8.5.1	Il confronto delle prestazioni	83
A	Alcuni codici	85
A.1	L'elica	85
A.2	Il rotore	97
A.3	L'elicottero	119
	Bibliografia	129

ELENCO DELLE FIGURE

2.1	Rappresentazione dei principali parametri geometrici dell'elica.	16
2.2	Rappresentazione del profilo BOEING-VERTOL VR-7.	17
2.3	Curve di portanza (a) e curve $Cd - \alpha$ (b) al variare del numero di Reynolds, per il profilo alare BOEING-VERTOL-VR7, ottenute mediante il software X-foil.	18
2.4	Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.	19
2.5	Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di <i>inflow</i> , φ , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.	19
2.6	Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.	20
2.7	Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.	20
2.8	Distribuzioni del coefficiente di spinta, $\frac{dC_T}{d\bar{r}}$, (a), del coefficiente di coppia, $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) lungo la pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.	21
2.9	Prestazioni dell'elica calcolate tramite modello aerodinamico semplificato e confrontate con i risultati sperimentali riportati in [5]. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c). $\theta = 15.5^\circ$ fissato al 75% della pala.	22
2.10	Prestazioni dell'elica riportata all'interno di [5], calcolate tramite modello aerodinamico semplificato al variare dell'angolo di calettamento. Coefficiente di spinta, C_T , (a), coefficiente di potenza, C_P , (b) e efficienza, η , (c).	23
2.11	Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. $\theta = 15.5^\circ$ fissato al 75% della pala.	24
2.12	Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di <i>inflow</i> , φ , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala.	24
2.13	Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala.	25

2.14	Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della a pala.	25
2.15	Distribuzioni del coefficiente di spinta, $\frac{dC_T}{d\bar{r}}$, (a), del coefficiente di coppia, $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) lungo la pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. $\theta = 15.5^\circ$ fissato al 75% della pala.	26
2.16	Prestazioni dell'elica confrontate con i risultati sperimentali riportati in [5]. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c). $\theta = 15.5^\circ$ fissato al 75% della pala.	27
2.17	Prestazioni dell'elica riportata all'interno di [5], al variare dell'angolo di calettamento. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c).	28
3.1	Trittico del velivolo Tecnam P92, tratto dal [3].	29
3.2	Rappresentazione dei principali parametri geometrici dell'elica progettata per il Tecnam P92.	33
3.3	Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$	34
3.4	Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di <i>inflow</i> , φ , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$	34
3.5	Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$	34
3.6	Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$	35
3.7	Distribuzioni del coefficiente di spinta $\frac{dC_T}{d\bar{r}}$, (a), del coefficiente di coppia $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) sulla pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$	35
3.8	Prestazioni dell'elica progettata per il Tecnam P92; Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c) al variare del rapporto di funzionamento	36
4.1	L'elicottero McDonnell-Douglas AH-64 <i>Apache</i> fotografato in azione. . .	39
5.1	Trittico dell'elicottero McDonnell-Douglas AH-64 <i>Apache</i> , tratto dal [3]. .	41
5.2	Rappresentazione del profilo alare montato sul rotore principale dell'elicottero in analisi.	42
5.3	Rappresentazione del profilo alare montato sul rotore anti-coppia di coda dell'elicottero McDonnell-Douglas AH-64.	43
5.4	Rappresentazione della caratteristica di portanza del profilo HH-02 per diversi numeri di Reynolds. Curve ottenute attraverso il <i>software</i> X-foil.	43
5.5	Rappresentazione della curva $Cd - \alpha$ per profilo HH-02 per diversi numeri di Reynolds. Curve ottenute attraverso il <i>software</i> X-foil.	44
5.6	Rappresentazioni in sezione di uno dei motori con cui è equipaggiato l'elicottero in analisi.	45
6.1	Rappresentazione del coefficiente di spinta del rotore (a), T_c , e del coefficiente di potenza (b), Q_c , al variare del rapporto di funzionamento, μ . Le curve sono parametrizzate al variare del comando di collettivo, θ_0 . Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità.	48

6.2	Rappresentazione della polare in <i>hover</i> (a) e della cifra di merito (b) per il rotore. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità.	48
6.3	Confronto dell'effetto del numero di pale finito calcolato mediante la funzione di correzione di Prandtl e il metodo di limitazione dell'integrale.	48
6.4	Fattore di correzione k per la non uniformità dell'induzione assiale sul rotore al variare della spinta in <i>hover</i>	49
6.5	Distribuzione del coefficiente di spinta (a), T_c , e del coefficiente di potenza (b), Q_c , sulla pala, in <i>hover</i> ($\mu = 0$), per diversi valori di input del collettivo, nominalmente: $\theta_0 = 10^\circ$, $\theta_0 = 13^\circ$, $\theta_0 = 16^\circ$, $\theta_0 = 19^\circ$. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità. Le caratteristiche aerodinamiche 2D del profilo son calcolate per: $Re_\infty = 1.25 \times 10^6$ e per: $M_\infty = 0$	50
6.6	Distribuzione del coefficiente di spinta, T_c , e del coefficiente di potenza (b), Q_c , sulla pala, per diversi valori del rapporto di funzionamento. Il valore del collettivo si è assunto costante e pari a: $\theta_0 = 16^\circ$. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità. Le caratteristiche aerodinamiche 2D del profilo son calcolate per: $Re_\infty = 1.25 \times 10^6$ e per: $M_\infty = 0$	51
6.7	Andamento dell'angolo d'attacco del rotore al variare del rapporto di funzionamento $\mu = V_\infty/\Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso).	51
6.8	Andamento del rapporto di ingresso λ e dell'induzione assiale λ_i al variare del rapporto di funzionamento $\mu = V_\infty/\Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso).	52
6.9	Andamento del coefficiente di potenza P_c (a), di resistenza H_c (b) e di forza laterale Y_c (c) del rotore al variare del rapporto di funzionamento $\mu = V_\infty/\Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso). Inoltre viene anche riportato l'andamento comando collettivo, ossia il passo di radice θ_0 (d).	53
6.10	Andamento dei coefficienti di flappeggio al variare del rapporto di avanzamento in volo traslato $\mu = \frac{V_\infty}{\Omega R}$	54
6.11	Evoluzione degli angoli di attacco effettivi sul disco rotore al variare del rapporto di avanzamento μ	56
6.12	Evoluzione del numero di Mach effettivo sul disco rotore al variare del rapporto di avanzamento μ	58
6.13	Forma assunta per pianta del rotore principale	59
6.14	Forma in pianta assunta per rotore principale	59
6.15	Evoluzione del numero di Mach effettivo sul disco rotore con freccia al tip al variare del rapporto di avanzamento μ	60
6.16	Evoluzione del sentiero di stallo per valori di μ che coincidono con la prima sezione di stallo fino ad una maggiorazione del 20% di tale rapporto di funzionamento.	61
6.17	Distribuzione dell'angolo di attacco sul rotore in corrispondenza del valore del rapporto di funzionamento per cui si verifica il primo stallo di un profilo.	62
7.1	Potenza totale richiesta P_{req} dal rotore principale divisa nei contributi di: Potenza indotta P_i , parassita P_0 e dovuta alla resistenza della fusoliera P_{fus}	64
7.2	Potenza totale richiesta P_{req} dal rotore anti-coppia di coda divisa nei contributi di: Potenza indotta P_i e parassita P_0	65

7.3	Potenza totale richiesta P_{tot} dalla macchina elicottero divisa nei contributi di: Potenza richiesta dal rotore principale P_{reqMR} , dal rotore anticoppa di coda P_{reqTR} , potenze ausiliarie P_{AUX} P_0 . Inoltre viene riportata anche la potenza massima continua disponibile (AEO) P_{av}	66
7.4	Potenza totale assorbita dalla macchina elicottero parametrizzata rispetto a 4 possibili pesi al decollo.	67
7.5	Variazione delle velocità caratteristiche (V_{max} , V_{BR} e V_{BE}) in funzione del peso per l'elicottero.	68
7.6	Variazione del rateo di salita in funzione della velocità di traslazione parametrizzata con il peso dell'elicottero.	69
7.7	Variazione del massimo rateo di salita in funzione del peso al decollo dell'elicottero.	70
7.8	Variazione della massima autonomia oraria al variare del peso dell'elicottero.	71
7.9	Variazione della massima autonomia chilometrica al variare del peso dell'elicottero.	72
7.10	Variazione del massimo angolo di salita con il peso per due particolari configurazioni di potenza disponibile: $P = P(OEI)$ e $P = 75\%P(OEI)$	73
7.11	Variazione della velocità di autorotazione con il peso.	73
8.1	Rappresentazione del profilo alare DU84-132V3.	78
8.2	Curva di portanza (a) e curva $C_d - \alpha$ del profilo DU84-132V3. $Re_\infty = 1 \times 10^6$, $M_\infty = 0$. Ottenuti mediante l'utilizzo del <i>software</i> Xfoil, e prolungate in post-stallo mediante l'utilizzo del <i>software</i> Q-blade.	78
8.3	Rappresentazione della curva di portanza per valori dell'angolo di attacco compresi nell'intervallo: $\alpha \in [-20^\circ, 20^\circ]$, del profilo alare DU84-132V3. $Re_\infty = 1 \times 10^6$, $M_\infty = 0$. Ottenuti mediante l'utilizzo del <i>software</i> Xfoil.	79
8.4	Distribuzione della corda (a) e del calettamento (b) in funzione del raggio.	79
8.5	Vista laterale (a), vista dall'alto (b), vista in assonometria (c) della pala. Immagini realizzate mediante il <i>software</i> Qblade.	80
8.6	Andamento del coefficiente di potenza, C_P , in funzione del rapporto di funzionamento, λ . Risultati ottenuti mediante il <i>software</i> Qblade.	81
8.7	Andamento del coefficiente di resistenza della turbina, C_T , in funzione del rapporto di funzionamento, λ . Risultati ottenuti mediante il <i>software</i> Qblade.	81
8.8	Vista laterale (a), vista dall'alto (b), vista in assonometria (c) della pala. Immagini realizzate mediante il <i>software</i> Qblade.	82
8.9	Distribuzione della corda (a) e del calettamento (b) in funzione del raggio.	83
8.10	Andamento del coefficiente di potenza, C_P , in funzione del rapporto di funzionamento, λ . Confronto tra le prestazioni della turbina progettata con il <i>software</i> Q-blade (curva continua) e quelle della turbina progettata con il metodo proposto in [1] (curva tratteggiata).	84
8.11	Andamento del coefficiente di resistenza della turbina, C_T , in funzione del rapporto di funzionamento, λ . Confronto tra le prestazioni della turbina progettata con il <i>software</i> Q-blade (curva continua) e quelle della turbina progettata con il metodo proposto in [1] (curva tratteggiata).	84

INTRODUZIONE

In questo elaborato saranno riportate alcune esercitazioni riguardanti i principali temi trattati durante il corso Aerodinamica dell'Ala Rotante durante l'a.a. 2021/2021 tenuto dal Prof. R. Tognaccini e dal Dott. Ing. G. Di Giorgio. L'elaborato è diviso in parti, in ognuna delle quali si è deciso di approfondire un particolare tema trattato durante il corso.

Parte I

L'elica

In questa parte dell'elaborato si intende presentare una trattazione sulle prestazioni e sul progetto dell'elica. Per quanto concerne l'implementazione dei codici per l'analisi delle prestazioni e per il progetto dell'elica, si faccia riferimento all'appendice A.1. Per la *routine* di risoluzione della teoria dell'elemento di pala generale si è scelto di implementare il metodo proposto in [1] a pagina 38, introducendo, però, la possibilità di tenere in conto anche di:

- Effetto del numero di pale finito (attraverso la funzione di correzione di Prandtl);
- Riduzione della spinta causata dal mozzo;
- Dipendenza delle caratteristiche aerodinamiche $C_l = C_l(\alpha, M, Re, \bar{r})$ e $C_d = C_d(\alpha, M, Re, \bar{r})$
- Effetto della freccia Λ ;

Per quanto concerne il progetto dell'elica, si valuterà l'andamento delle principali curve caratteristiche di un'elica sviluppata attraverso un codice *ad hoc*, basato sulla procedura di *design* presentata a pag. 46 di [1].

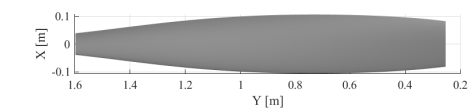
In questo capitolo si valuterà lo studio delle prestazioni di un'elica, la cui geometria è stata tratta da [5].

2.1 La geometria

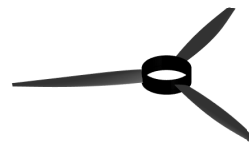
In figura 2.1 è presentata l'elica in analisi. In particolare sono riportate le viste in pianta (a e d) ed assonometriche (b e c) della pala e dell'elica tripala completa e le distribuzioni radiali di corda e calettamento. Si noti che la rappresentazione proposta trascura le sezioni di pala in prossimità del mozzo sia perché risultano di scarso interesse ai fini della prestazione stessa dell'elica e anche perché non ben caratterizzate all'interno di [5]. Sempre con riferimento alla rappresentazione dell'elica si noti che in corrispondenza dell'estremità della pala si è scelto di non rappresentare i bordi della pala raccordati, come in [5].

Parametro	Valore	
Raggio, R	1.6	[m]
Raggio mozzo, R_{hub}	0.254	[m]
Numero di pale, N	3	[/]
Giri al minuto, RPM	1200	[$giri/min$]
Solidità media, σ	0.318	[/]

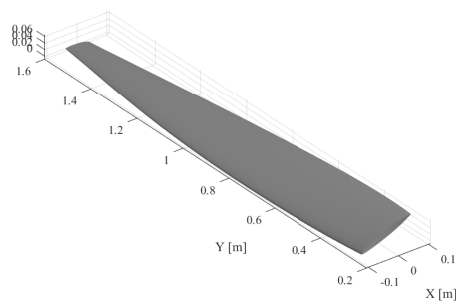
Tabella 2.1: Alcune caratteristiche geometriche dell'elica in analisi, tratta da [5].



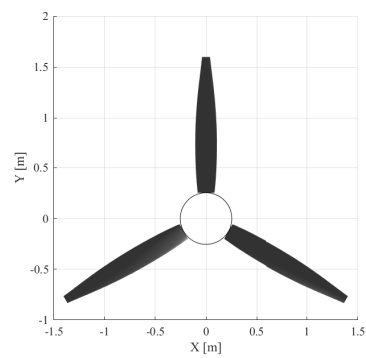
(a)



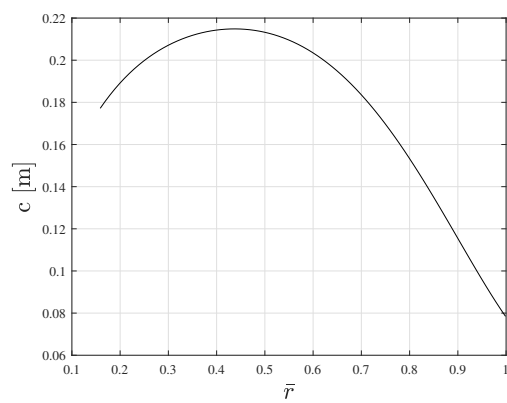
(b)



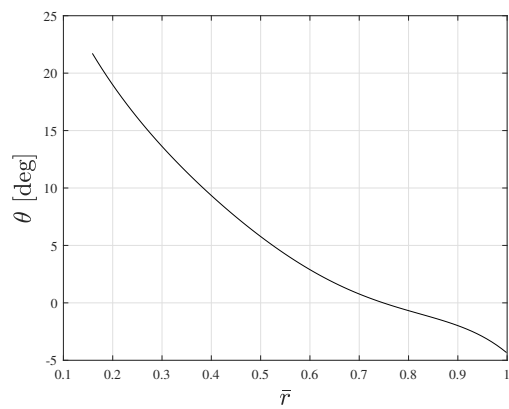
(c)



(d)



(e)



(f)

Figura 2.1: Rappresentazione dei principali parametri geometrici dell'elica.

2.2 L'aerodinamica

Una volta analizzata la geometria in *input*, ci si interessa di caratterizzare aerodinamicamente l'elica in analisi. In questo capitolo vengono presi in considerazione due possibili approcci: in un primo caso si valuterà un modello aerodinamico semplificato, in cui le funzioni che restituiscono il valore del coefficiente di portanza e resistenza in funzione dell'angolo di attacco sono le funzioni banali riportate in (2.1).

$$C_l = 2\pi\alpha \quad C_d = 0.01 \quad (2.1)$$

In seconda analisi viene implementato un modello aerodinamico più sofisticato, che tiene conto dell'utilizzo di un effettivo profilo alare e delle differenti caratteristiche aerodinamiche della corrente in cui è immerso. Il profilo alare selezionato è il BOEING-VERTOL VR-7, riportato in figura 2.2. Nelle immagini 2.3 sono presentate le curve di portanza e le curve $C_d - \alpha$ al variare del numero di Reynolds. In particolare, si è scelto di prendere in considerazione numeri di Reynolds all'interno dell'intervallo $Re_\infty \in [0.5 \times 10^6; 5 \times 10^6]$. Tale scelta è supportata dal grafico 2.4, (b), che riporta la distribuzione del numero di Reynolds lungo la pala. Tali curve caratteristiche aerodinamiche del profilo sono state utilizzate per generare un *database* aerodinamico che potesse generalizzare le funzioni $C_l = C_l(\alpha)$ e $C_d = C_d(\alpha)$ ed inserire le dipendenze dal numero di Mach e dal numero di Reynolds, come in (2.2).

$$C_l = C_l(\alpha, M, Re) \quad C_d = C_d(\alpha, M, Re) \quad (2.2)$$

Gli effetti di comprimibilità sono stati valutati in similitudine per il coefficiente di portanza, ovvero secondo la relazione riportata in (2.3), e mediante la relazione (2.4), tratta da [6], per il coefficiente di resistenza.

$$C_l(\alpha, M, Re) = \frac{C_l(\alpha, M = 0, Re)}{\sqrt{1 - M^2}} \quad (2.3)$$

$$C_d(\alpha, M, Re) = [C_d(\alpha, M = 0, Re) - k_2] + k_2 \left(1 + 0.25M^k\right) \quad , k_2 = 0.7, k = 6 \quad (2.4)$$

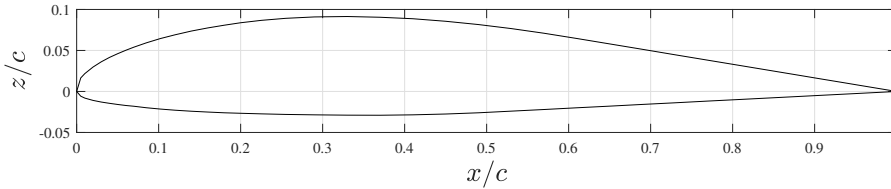
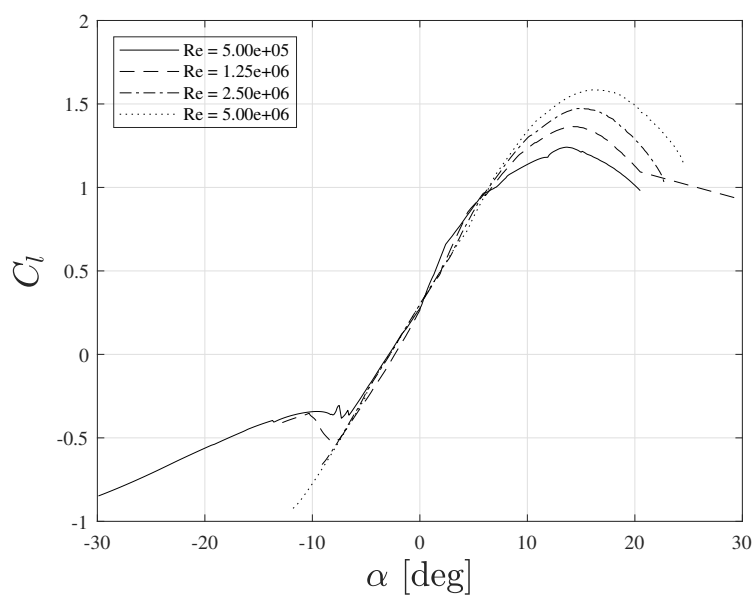
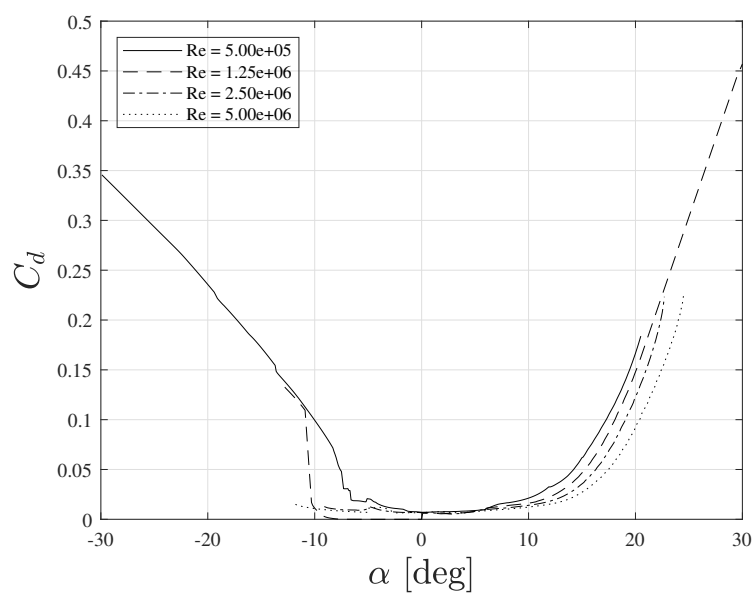


Figura 2.2: Rappresentazione del profilo BOEING-VERTOL VR-7.



(a)



(b)

Figura 2.3: Curve di portanza (a) e curve $C_d - \alpha$ (b) al variare del numero di Reynolds, per il profilo alare BOEING-VERTOL-VR7, ottenute mediante il *software* X-foil.

2.3 Le prestazioni (modello aerodinamico semplificato)

2.3.1 Le distribuzioni sulla pala

Si valutino le distribuzioni di alcune grandezze caratteristiche lungo la pala. In primo luogo si valutino le figure riportate in 2.4, che definiscono le caratteristiche della corrente che investe il profilo, stazione per stazione, in termini di numero di Mach e numero di Reynolds. In questa figura possiamo notare che il numero di Mach cresce lungo la pala con il massimo all'estremità poiché cresce con il raggio esclusivamente, viceversa il numero di Reynolds cresce fino a circa il 70% della pala, dove ha un massimo, per poi diminuire. Tale andamento è associato al fatto che i contributi al numero di Reynolds provengono sia dalla velocità che dalle corde, che invece tendono a ridursi dopo il 30% della pala.

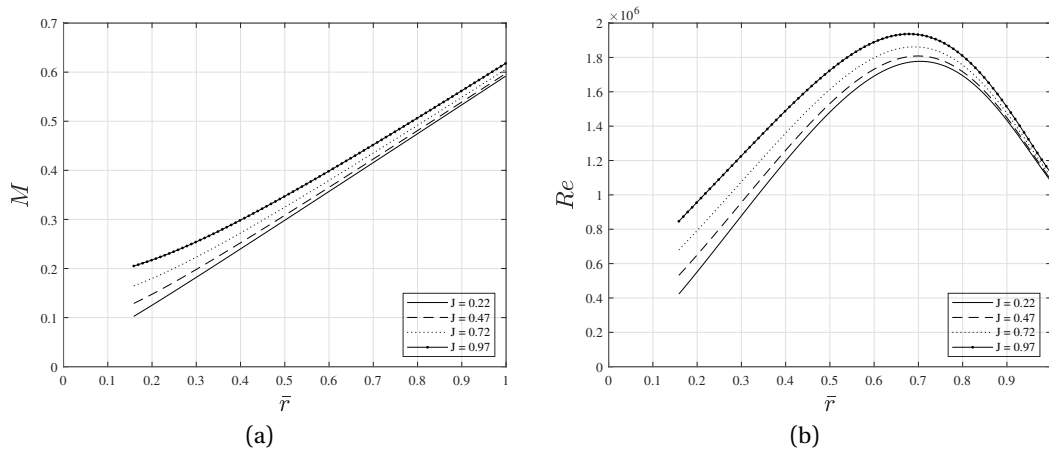


Figura 2.4: Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.

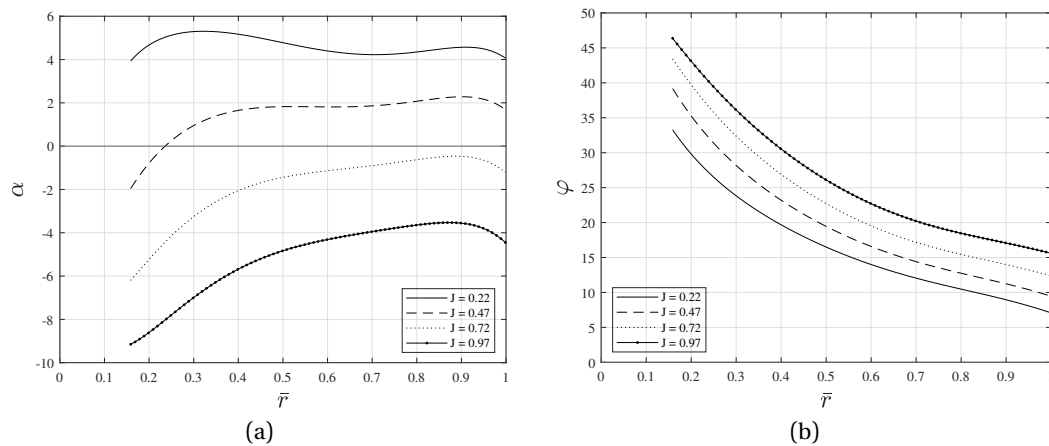


Figura 2.5: Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di inflow, φ , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.

Con riferimento alle figure 2.5 possiamo notare che l'angolo di inflow, φ , ha un andamento circa iperbolico e cresce all'aumentare del rapporto di funzionamento. L'andamento dell'angolo di attacco è circa costante lungo la pala, ciò è dovuto alla similitudine fra le distribuzioni di calettamento e angolo di inflow lungo la pala, almeno

per valori del rapporto di funzionamento medio-bassi. Viceversa per rapporti di funzionamento elevati, per i quali l'angolo di attacco è sempre negativo, insiste anche una variazione sensibile dell'angolo di attacco stesso.

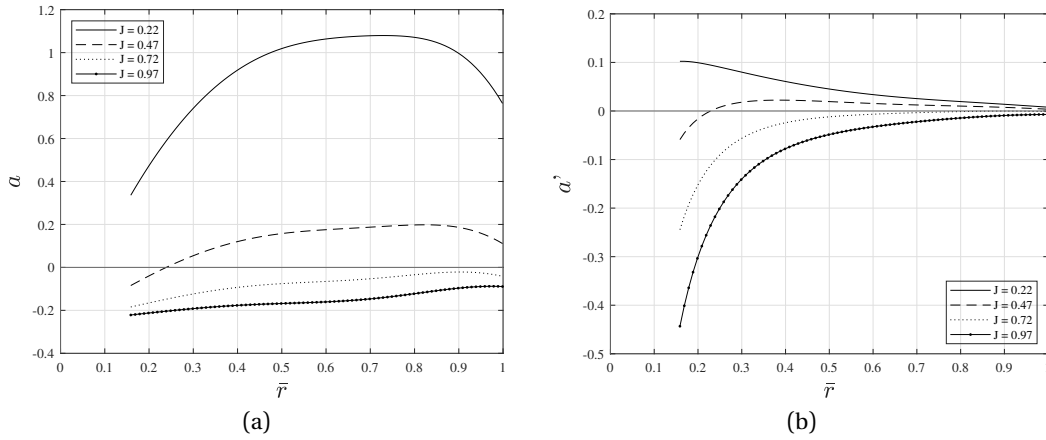


Figura 2.6: Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.

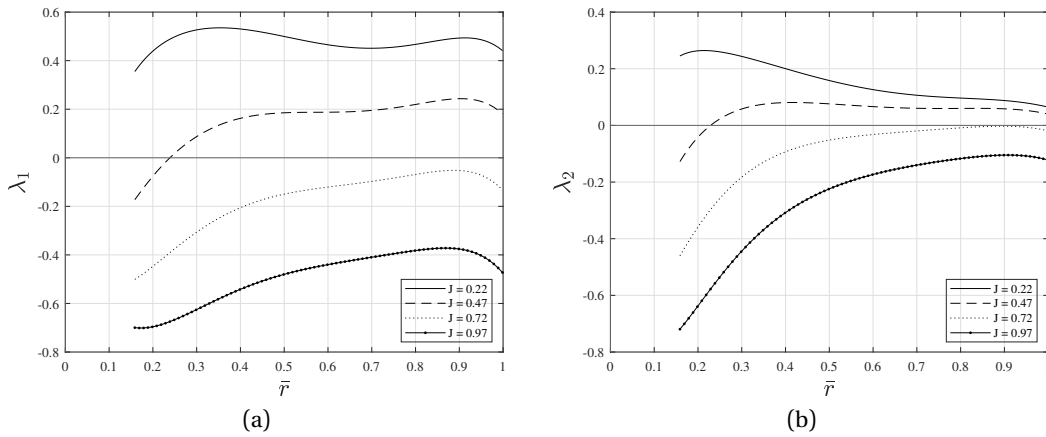


Figura 2.7: Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.

Con riferimento alle figure 2.6, possiamo notare che le induzioni sulla pala sono entrambe pressoché costanti, con eccezione della zona in prossimità del mozzo. Si noti inoltre che l'induzione assiale appare sensibilmente maggiore di quella rotazionale, come atteso. Entrambe le induzioni appaiono variare con la stessa legge dei coefficienti λ_1 e λ_2 , riportati nelle figure 2.7.

Per quanto riguarda le distribuzioni di spinta, coppia e potenza possiamo vedere (figura 2.8) che queste assumono valore molto basso in corrispondenza del mozzo, crescono lungo la pala, con andamento quasi lineare, fino a raggiungere un massimo per poi annullarsi. Per la computazione di tali distribuzioni si è tenuto in conto dell'effetto del numero di pale finito mediante l'utilizzo della funzione di correzione di Prandtl.

E' importante osservare che per ridotti rapporti di funzionamento, l'induzione assiale assume valori molto elevati, compromettendo, di fatti, la validità della teoria impulsiva.

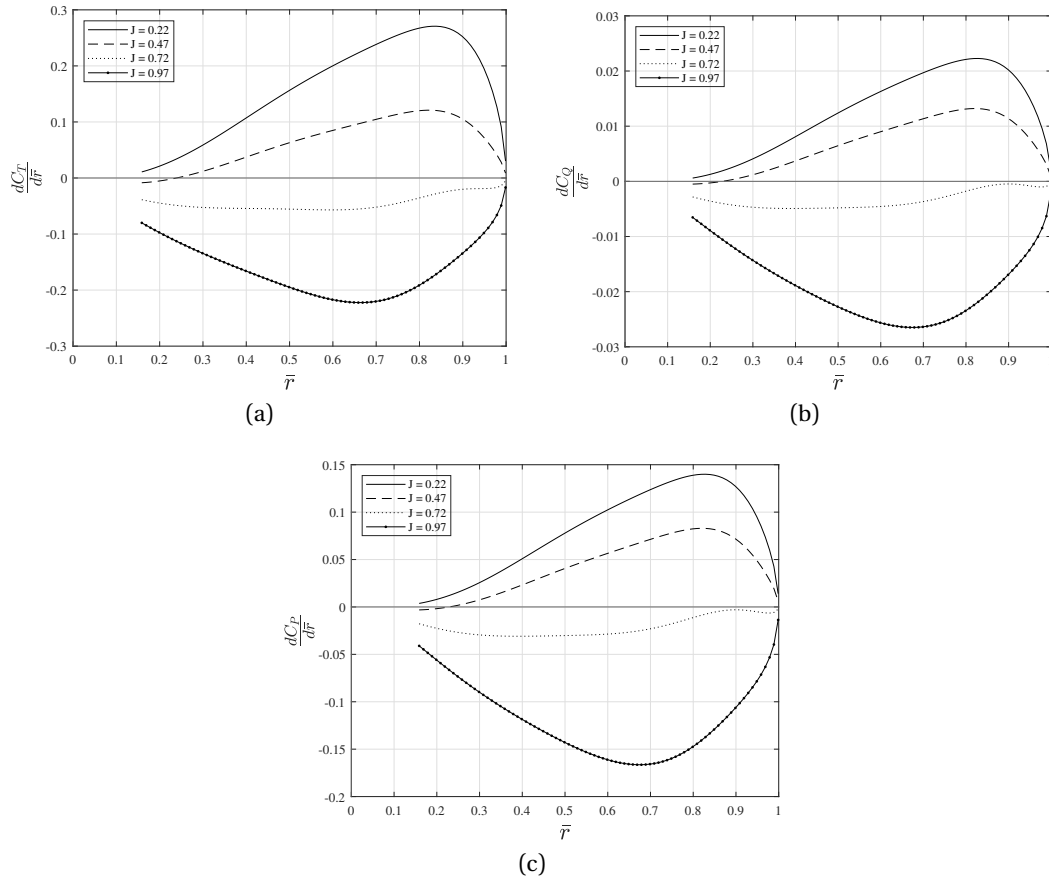


Figura 2.8: Distribuzioni del coefficiente di spinta, $\frac{dC_L}{d\bar{r}}$, (a), del coefficiente di coppia, $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) lungo la pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. $\theta = 15.5^\circ$ fissato al 75% della pala. Modello aerodinamico semplificato.

Inoltre, si osserva che per valori di $J > 0.6$ queste assumono valori sempre negativi, il che implica che in tali regimi di funzionamento la spinta e la coppia totali saranno negativi, ossia l'elica lavora nel regime a mulinello frenante. In conclusione, con questo valore di calettamento l'elica è classificabile come un elica lenta, in accordo con quanto riportato in [5].

2.3.2 Le curve caratteristiche

Di seguito sono riportate le curve caratteristiche dell'elica per diverse condizioni di funzionamento. In primo luogo siamo interessati a valutare un confronto dei risultati numerici con quelli sperimentali proposti in [5]: in questo caso si è scelto di fissare un unico calettamento: $\theta = 15.5^\circ$ fissato al 75% della pala, lo stesso adottato nelle prove sperimentali. Si può osservare che le curve si distaccano da quelle sperimentali presumibilmente perché non si tiene del conto delle effettive caratteristiche aerodinamiche dei profili montati lungo la pala.

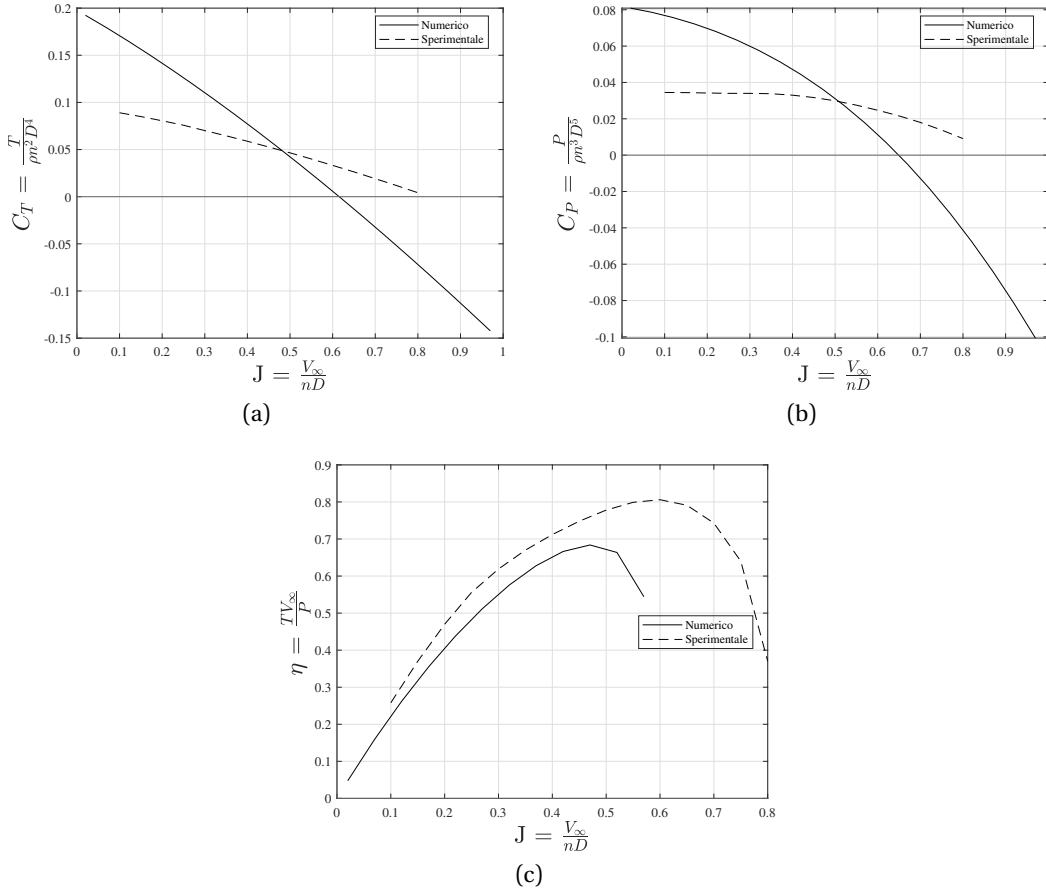


Figura 2.9: Prestazioni dell'elica calcolate tramite modello aerodinamico semplificato e confrontate con i risultati sperimentali riportati in [5]. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c). $\theta = 15.5^\circ$ fissato al 75% della pala.

In secondo luogo, una volta valutato il confronto con i dati sperimentali, viene proposta un'analisi delle prestazioni dell'elica al variare del calettamento. Le immagini di riferimento sono riportate nelle figure 2.10.

All'aumentare del calettamento cresce sia la spinta generata che la potenza richiesta. Inoltre si può osservare che i valori del rapporto di funzionamento J in corrispondenza dei quali il C_T ed il C_Q si annullano aumentano, ciò permette di utilizzare l'elica in regimi di funzionamento più veloci. Infine si osserva che all'aumentare del calettamento il massimo rendimento sale e si porta a valori di J più elevati.

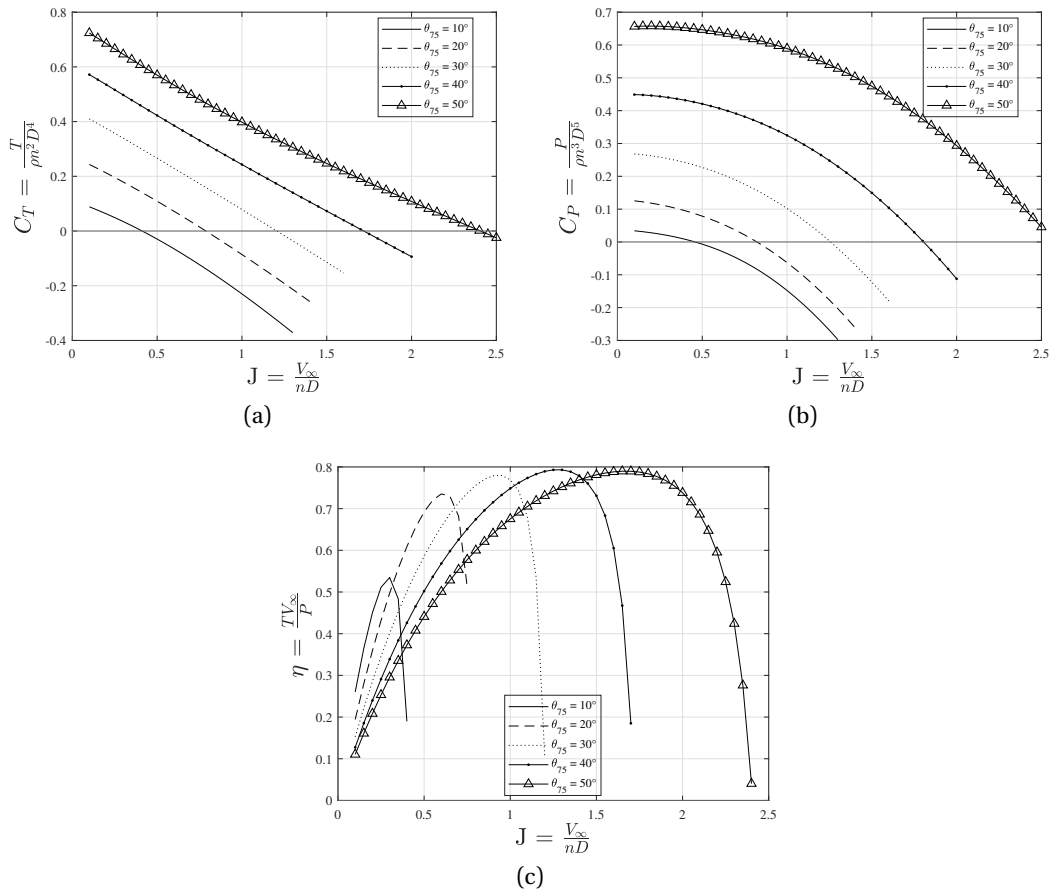


Figura 2.10: Prestazioni dell'elica riportata all'interno di [5], calcolate tramite modello aerodinamico semplificato al variare dell'angolo di calettamento. Coefficiente di spinta, C_T , (a), coefficiente di potenza, C_P , (b) e efficienza, η , (c).

2.4 Le prestazioni (modello aerodinamico completo)

In questa sezione si intende riproporre l'analisi condotta nella sezione precedente utilizzando di un modello aerodinamico più accurato, riportato nella sezione 2.2.

2.4.1 Le distribuzioni sulla pala

In prima analisi si valutino, come fatto in precedenza, le distribuzioni delle grandezze sulla pala: nelle figure 2.11, 2.12, 2.13, 2.14, 2.15. Osservando le figure di cui sopra, e confrontandole con quelle ottenute nella sezione precedente, è possibile notare che gli andamenti di tutte le grandezze in analisi sono confrontabili.

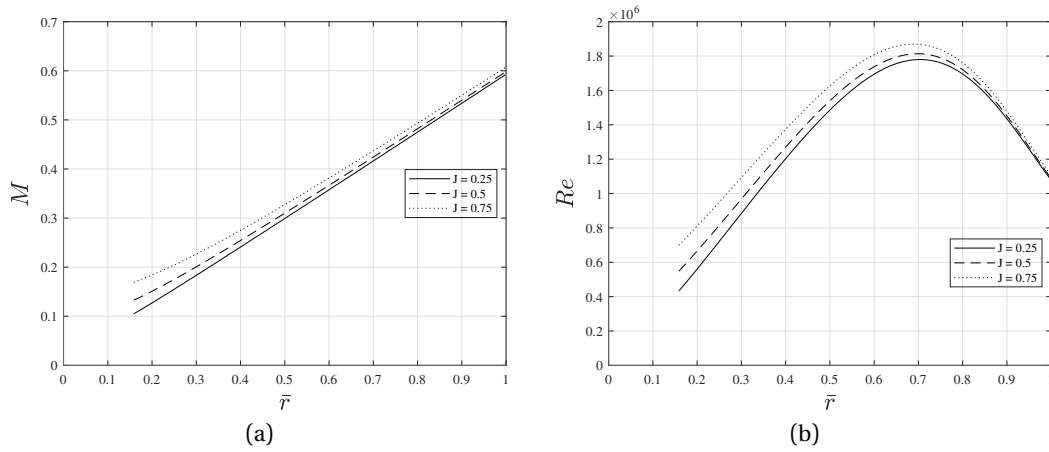


Figura 2.11: Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. $\theta = 15.5^\circ$ fissato al 75% della pala.

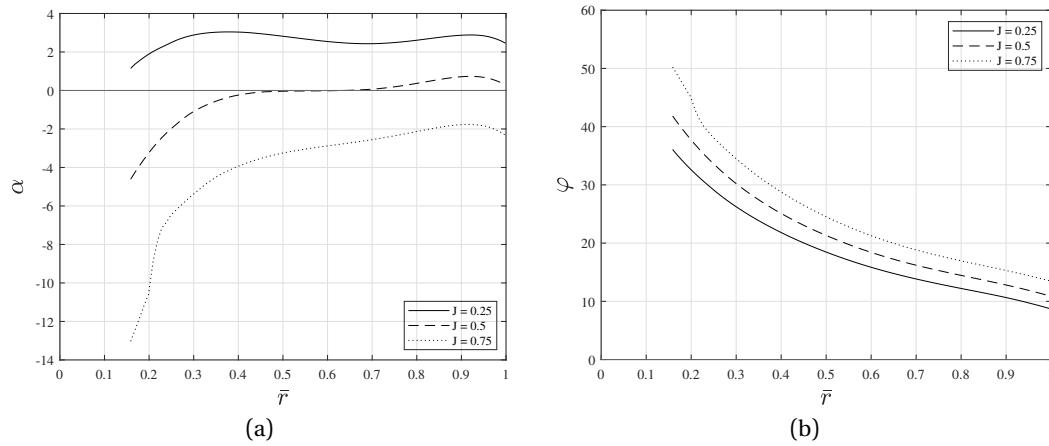


Figura 2.12: Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di inflow, φ , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala.

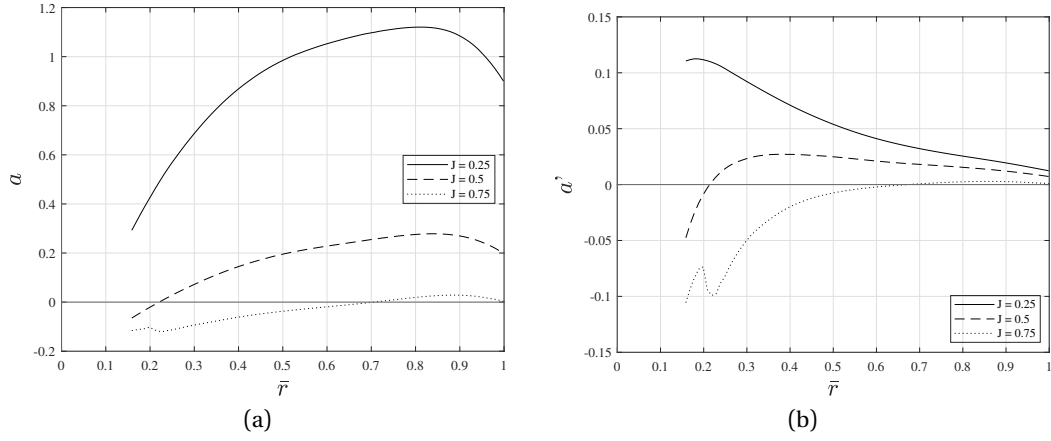


Figura 2.13: Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della pala.

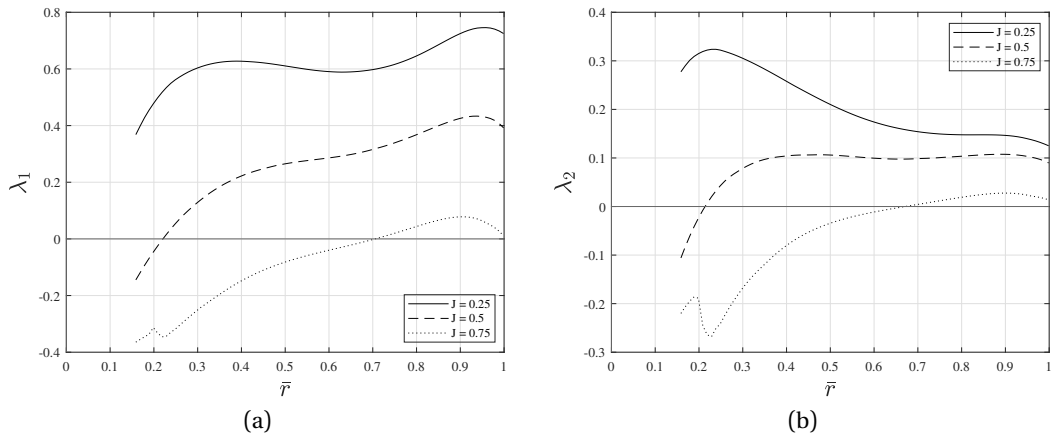


Figura 2.14: Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. $\theta = 15.5^\circ$ fissato al 75% della a pala.

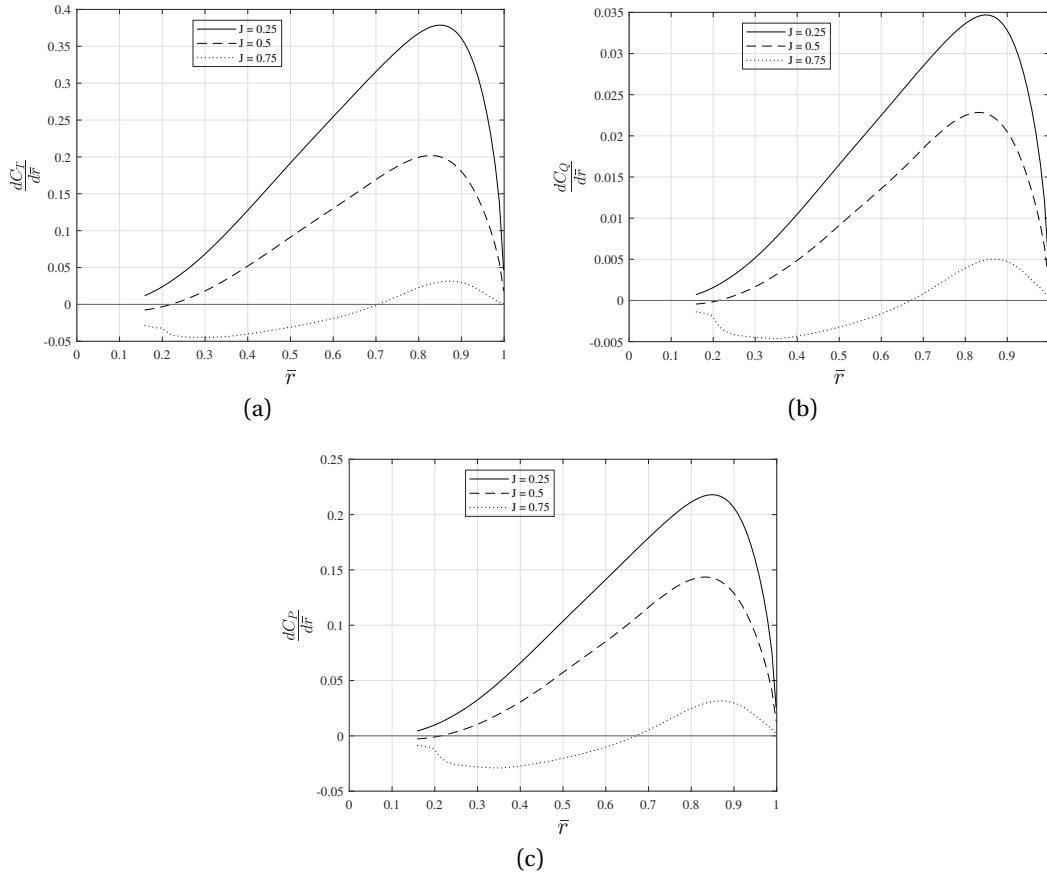


Figura 2.15: Distribuzioni del coefficiente di spinta, $\frac{dC_T}{d\bar{r}}$, (a), del coefficiente di coppia, $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) lungo la pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. $\theta = 15.5^\circ$ fissato al 75% della pala.

2.4.2 Le curve caratteristiche

Di seguito sono riportate le curve caratteristiche dell'elica per diverse condizioni di funzionamento. In primo luogo siamo interessati a valutare un confronto dei risultati numerici con quelli sperimentali proposti in [5]: in questo caso si è scelto di fissare un unico calettamento: $\theta = 15.5^\circ$ fissato al 75% della pala, lo stesso adottato nelle prove sperimentali.

Anche in questo caso le curve ottenute si distaccano da quelle sperimentali. Ciò può essere dovuto al fatto che il profilo scelto non coincide con quelli effettivamente montati lungo la pala, infatti [5] riporta ben 10 profili differenti lungo la pala per i quali non è completamente definita la geometria. Per questo motivo non è stato possibile utilizzare tali profili.

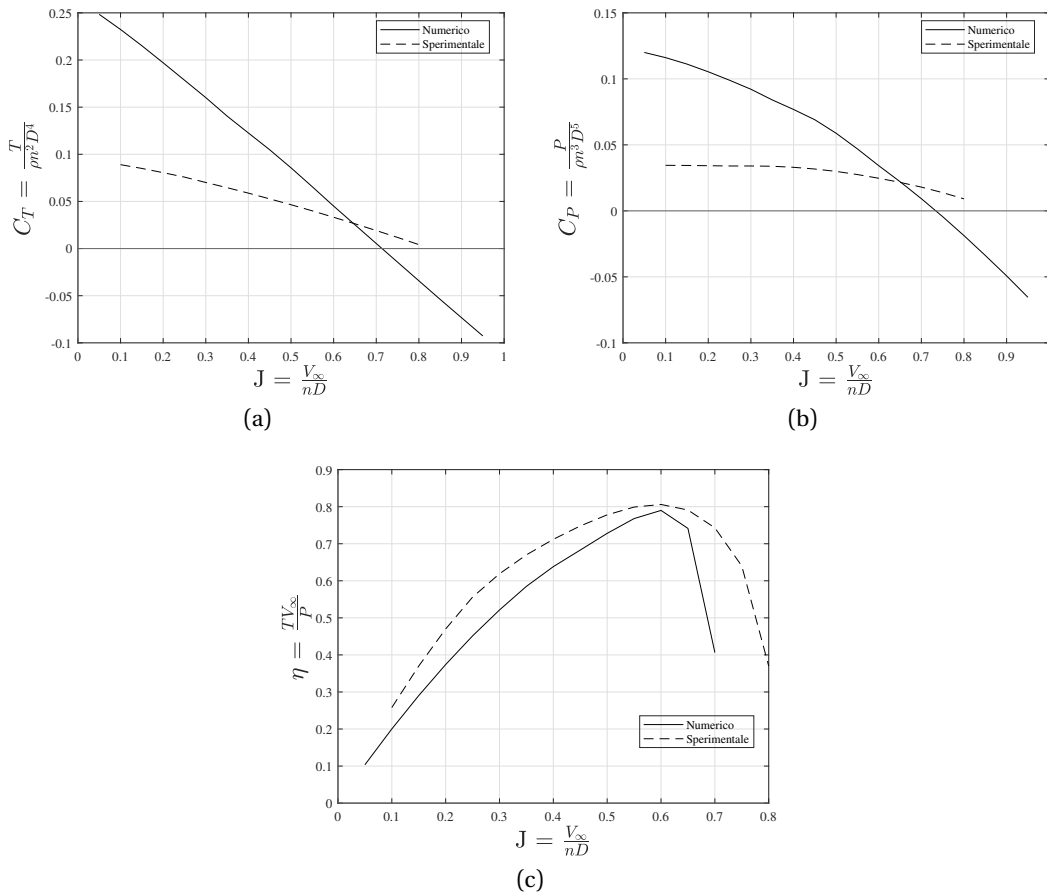


Figura 2.16: Prestazioni dell'elica confrontate con i risultati sperimentali riportati in [5]. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c). $\theta = 15.5^\circ$ fissato al 75% della pala.

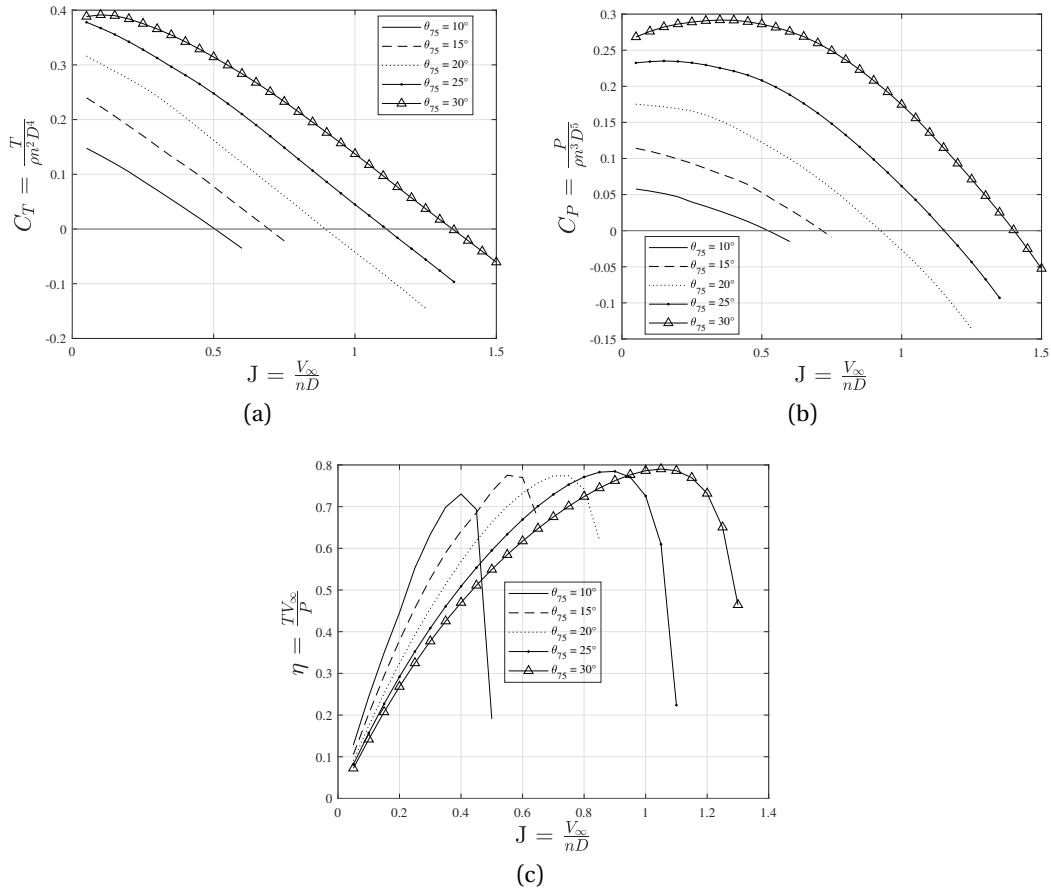


Figura 2.17: Prestazioni dell'elica riportata all'interno di [5], al variare dell'angolo di calettamento. Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c).

3.1 Il progetto dell'elica per il Tecnam P92

In questa sezione viene proposto il progetto di un'elica lenta, nello specifico si è scelto di progettare un'elica che equipaggia il velivolo di aviazione generale Tecnam P92. In tabella 3.1 vengono riportati alcuni dati geometrici, fondamentali per la stima della resistenza, ovvero della spinta richiesta all'elica, in volo traslato a velocità di crociera.

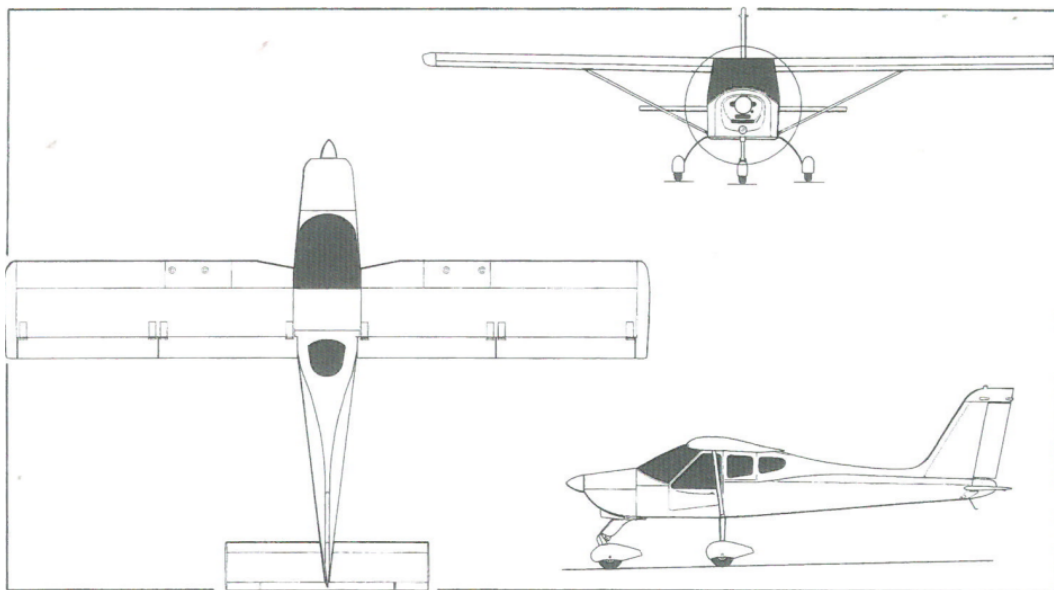


Figura 3.1: Trittico del velivolo Tecnam P92, tratto dal [3].

Riguardo le caratteristiche riportate in 3.1, è bene notare che al valore del fattore di Oswald, e , è stato assegnato un valore tipico per la classe di velivolo, non essendo disponibile in letteratura.

Una volta presentate le principali caratteristiche geometriche del velivolo, si procede ad eseguire una stima della spinta necessaria al volo traslato. In primo luogo, sfruttando l'equazione di equilibrio alla traslazione verticale, riportata in (3.1), possiamo dedurre il coefficiente di portanza, C_L , (3.2). Si noti, inoltre, che, data la ridotta quota di volo caratteristica del velivolo, si è ritenuta trascurabile la riduzione di densità con la quota. Si è assunto, quindi, che la densità non si discosti dal valore *sea level*.

$$L = 0.5\rho_{\infty}V_{\infty}^2SC_L = W \quad (3.1)$$

Parametro	Valore	
Velocità di crociera, V_{cr}	51	m/s
Superficie alare, S	13.2	m^2
Apertura alare, b	4.70	m
Allungamento alare, AR	5.73	$[/]$
Peso, W	450	kg
Fattore di Oswald, e	0.85	$[/]$

Tabella 3.1: Alcune caratteristiche del velivolo Tecnam P92 necessarie alla stima della resistenza in volo traslato.

$$C_L = \sqrt{\frac{2W}{\rho_\infty V_\infty S}} \simeq 0.205 \quad (3.2)$$

Una volta noto il coefficiente di portanza, è possibile pervenire alla scrittura del coefficiente di resistenza totale del velivolo, che verrà calcolato secondo la (3.3). Per quanto riguarda il coefficiente di resistenza parassita, in questo contesto verrà assunto pari a circa 350 dc per semplicità, tuttavia, per un'analisi più accurata si potrebbe implementare una decomposizione della resistenza parassita secondo la relazione 3.4, che stima il contributo di resistenza di ogni superficie bagnata e lo scala rispetto alla superficie di riferimento. Tale relazione tiene conto anche di altre fonti di resistenza, accomunate sotto il nome di *miscellaneous drag*, proveniente da appendici esposte. Nel nostro caso, invece, si è scelto un valore pari a circa 350 dc dal momento che i valori caratteristici di resistenza parassita sono contenuti all'interno dell'intervallo seguente: $C_{D,0} \in [250 \text{ } dc, 400 \text{ } dc]$ ed indicano il grado di sofisticatezza del progetto aerodinamico. Per velivoli con carrello fisso e ala controventata, come in questo caso, ci si avvicina di più al limite superiore dell'intervallo presentato in precedenza.

$$C_D = C_{D,0} + C_{D,i} = C_{D,0} + \frac{C_L^2}{\pi A R e} \simeq 377 \text{ } dc \quad (3.3)$$

$$C_{D,0} = \sum_k C_{D,0,k} \frac{S_k}{S_{ref}} + 10\% \quad (3.4)$$

Sommando il contributo di resistenza parassita a quello di resistenza indotta calcolato con il valore del C_L riportato in (3.2), si ottiene una stima del coefficiente di resistenza totale del velivolo in volo traslato, riportato in (3.3). Sfruttando l'equazione di equilibrio orizzontale in volo traslato, (3.5), è possibile pervenire al valore di spinta che deve erogare l'elica.

$$D = 0.5 \rho_\infty V_\infty^2 S C_D = T \simeq 801 \text{ } N \quad (3.5)$$

Tale valore della spinta rappresenta il valore di *input* all'interno della procedura di *design* dell'elica. Associato al valore della spinta necessaria per sostenere il volo traslato, è necessario fornire altri parametri per completare la procedura di *design*, i suddetti parametri sono riportati all'interno della tabella riassuntiva 3.2. Il valore della velocità angolare riportata in tabella è stato stimato fissando il numero di Mach al *tip* della pala: $M_{tip} = 0.7$.

A partire dalle informazioni riportate in 3.2 è possibile dedurre il coefficiente di spinta dell'elica, riportato in (3.6) e il rapporto di avanzamento di progetto, riportato in (3.7).

$$J = \frac{V_{cr}}{nD} = 0.67 \quad (3.6)$$

$$C_T = \frac{T}{\rho_{\infty} n^2 D^4} = 0.0416 \quad (3.7)$$

Parametro	Valore	
Numero di pale, N	2	[/]
Velocità angolare, RPM	2740	$giri/min$
Diametro, D	1.66	m

Tabella 3.2: Alcuni vincoli, geometrici e di funzionamento dell'elica del Tecnam P92, tratte dal [3].

Vengono proposti adesso i risultati della progettazione dell'elica in seno alla teoria dell'elemento di pala generalizzata. In primo luogo si valuti la geometria, riportata nelle immagini in figura 3.2. Per quanto concerne la caratterizzazione aerodinamica dell'elica, si noti che si è scelto di utilizzare il profilo BOEING-VERTOL-VR7, già presentato nelle precedenti sezioni.

Si osserva che, per la scelta del calettamento, si è imposto che le varie sezioni lavorino ad una fissata incidenza pari al valore dell'angolo di attacco per cui si ha la massima efficienza. Tale valore di angolo d'attacco, α_{id} , ricavato per una valore medio del numero di Reynolds, è pari a circa 6° , in corrispondenza del quale, si ha un coefficiente di portanza pari $C_l = 0.98$. Tale elevato valore del coefficiente di portanza implica da un lato la ridotta solidità lungo la pala, dall'altro può introdurre problemi di comprimibilità nelle sezioni situate al *tip* della pala.

Si noti esplicitamente che non si è tenuto conto delle esigenze strutturali che, all'atto della realizzazione dell'elica, prevalgono sulle esigenze aerodinamiche in prossimità del mozzo, e, spesso, gli stessi profili alari dell'elica degenerano da un profilo alare ad una sezione cilindrica.

Dall'immagine riportata in 3.2, (f) è possibile notare che il calettamento al 75% del raggio vale circa $\theta_{75} \simeq 25^\circ$.

3.1.1 Analisi delle prestazioni

Viene adesso presentata un'analisi dell'elica progettata nelle precedenti sezioni.

In figura 3.3, (a), è riportata la distribuzione del numero di Mach lungo la pala per un valore del rapporto di avanzamento pari a quello di progetto $J = \frac{V_{cr}}{nD}$. Si osserva che il numero di Mach cresce circa linearmente lungo la pala raggiungendo un valore maggiore di 0.7 all'estremità; ciò può portare a problemi di comprimibilità dati gli elevati assetti con cui è stata progettata l'elica. Sarebbe quindi opportuno prevedere per l'elica una porzione di pala a freccia verso le estremità oppure riformulare il progetto prevedendo assetto minori al *tip*.

Per quanto riguarda il numero di Reynolds, riportato in 3.3, (b), si osserva che questo cresce circa linearmente e tende a zero all'estremità.

In figura 3.4 sono riportate le distribuzioni degli angoli d'attacco e dell'angolo di *inflow* lungo la pala, per un rapporto di avanzamento pari a quello di *design*. Si può osservare che l'angolo d'attacco esibisce delle fluttuazioni intorno ad un valor medio che è maggiore di quello di progetto, ciò è ascrivibile al fatto che la procedura di *design* è una procedura approssimata, che non tiene in conto nè degli effetti viscosi nè della possibilità di riscontrare elevati angoli di *inflow*.

Passando poi all'analisi delle curve caratteristiche, possiamo notare che il coefficiente di spinta ottenuto, 3.8, (a), raggiunge, in effetti, il valore di progetto, tuttavia tale valore del coefficiente di spinta è ottenuto per un rapporto di avanzamento minore di quello di progetto.

Sempre con riferimento alla stessa figura, nell'immagine (b) possiamo notare che il valore del coefficiente di potenza conduce ad una potenza richiesta all'albero in linea con quelle erogabili dal propulsore con cui è equipaggiato il velivolo: il Tecnam P92 è equipaggiato con un propulsore *rotax* in grado di erogare circa $73,5 \text{ kW}$ a 5.800 RPM . Il massimo valore del coefficiente di potenza riportato in figura 3.8, (b), pari $C_P = 0.0342$ richiede una potenza pari a circa 50 kW , come riportato in (3.8).

$$P_{req} = C_P \rho_\infty n^3 D^5 \simeq 50 \text{ kW} \quad (3.8)$$

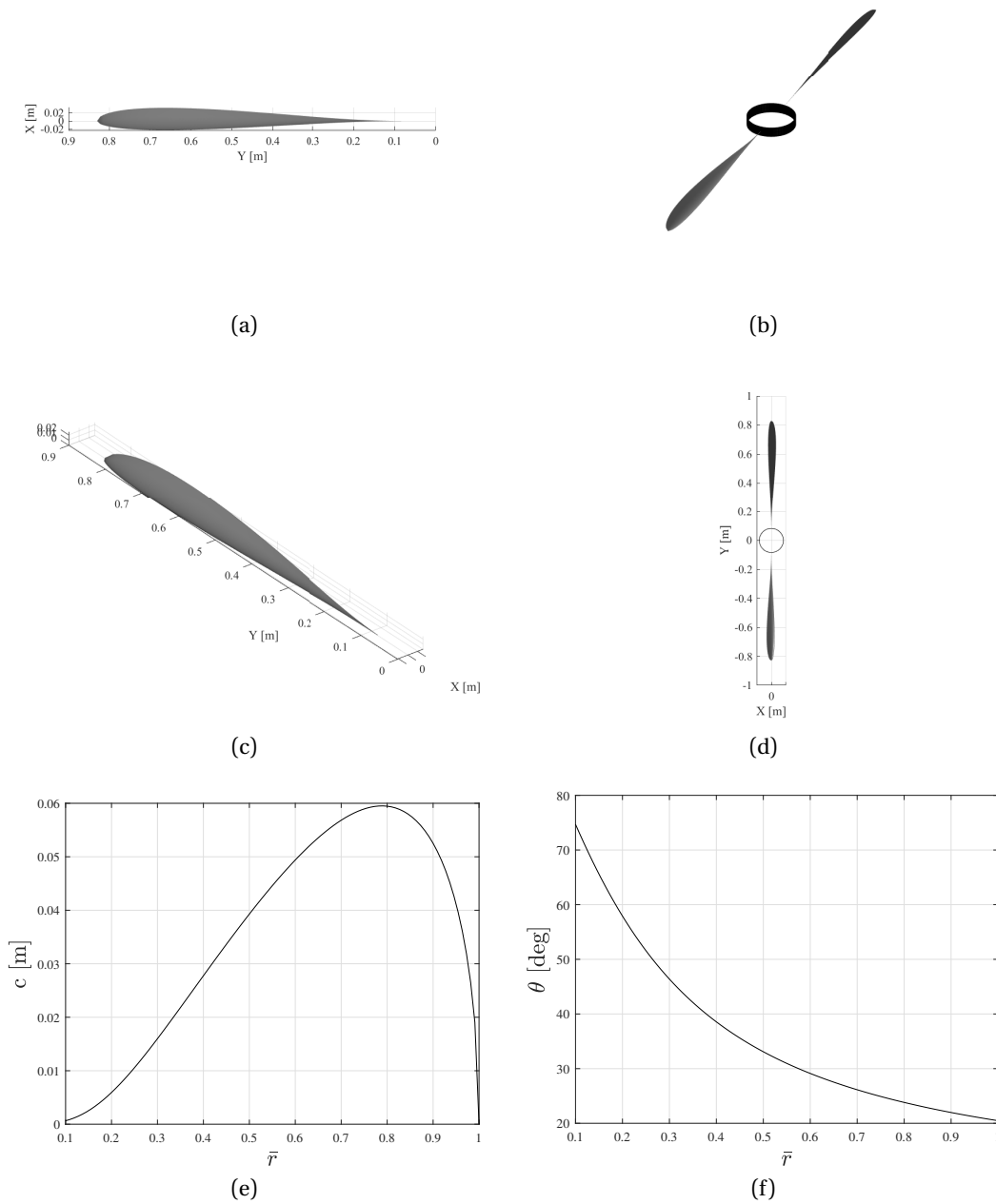


Figura 3.2: Rappresentazione dei principali parametri geometrici dell'elica progettata per il Tecnam P92.

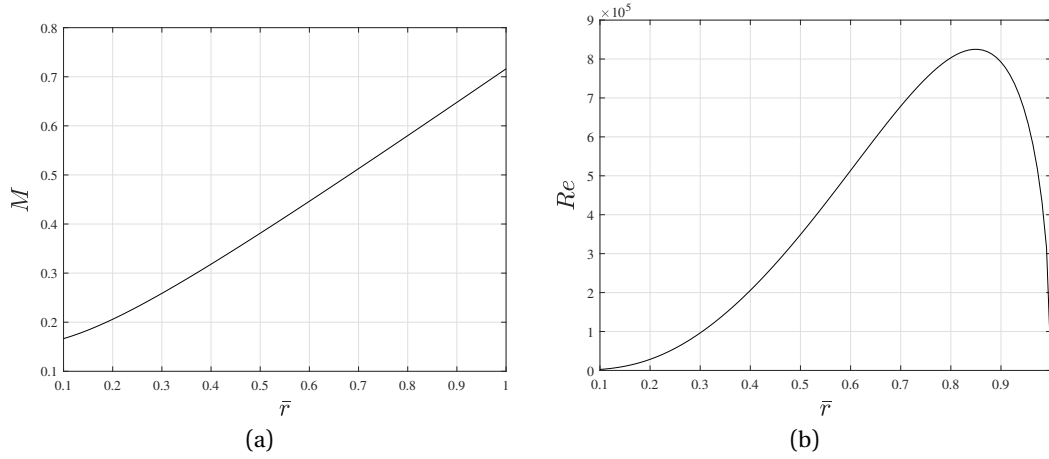


Figura 3.3: Distribuzioni del numero di Mach (a) e del numero di Reynolds (b) lungo la pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$.

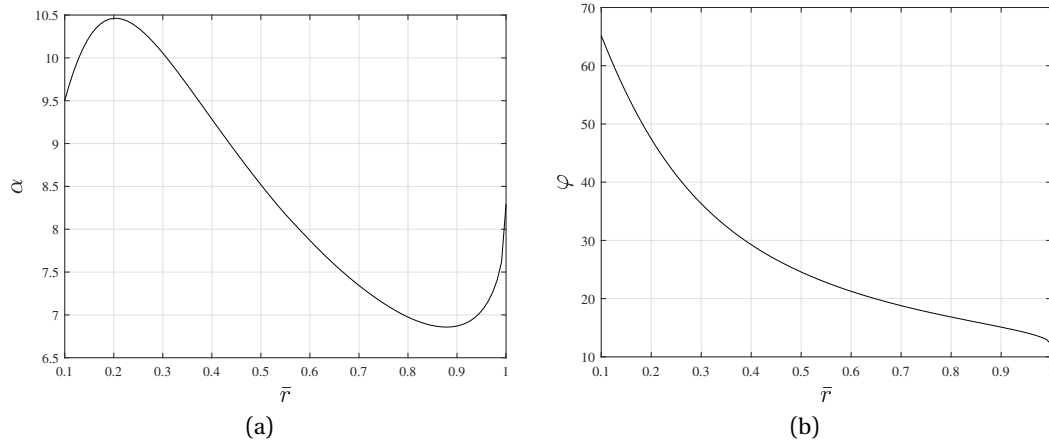


Figura 3.4: Distribuzione dell'angolo d'attacco effettivo a cui lavorano le sezioni, α , (a) e dell'angolo di *inflow*, φ , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$.

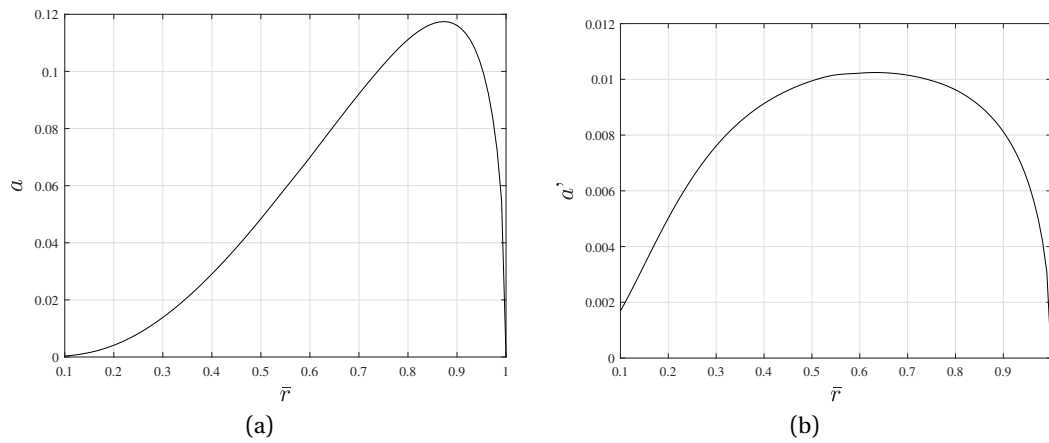


Figura 3.5: Distribuzioni dell'induzione assiale, a , (a), e dell'induzione rotazionale, a' , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$.

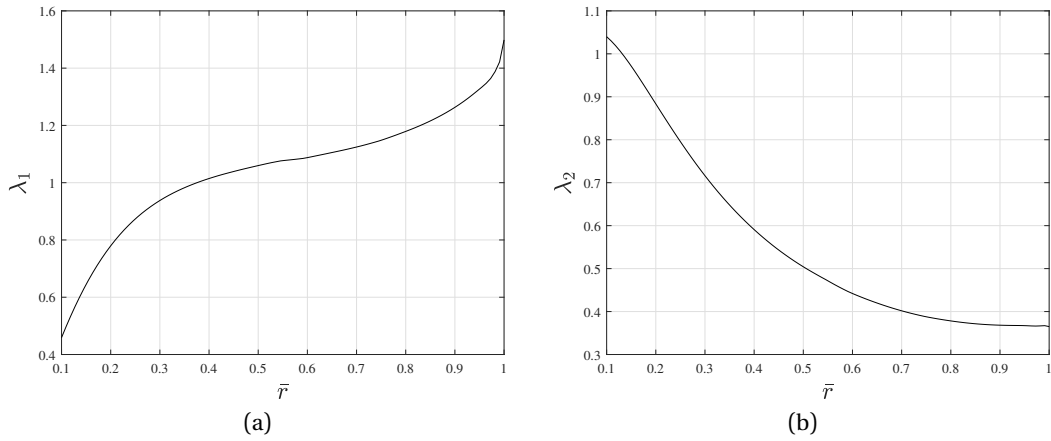


Figura 3.6: Distribuzioni dei coefficienti aerodinamici λ_1 , (a) e λ_2 , (b), sulla pala. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$.

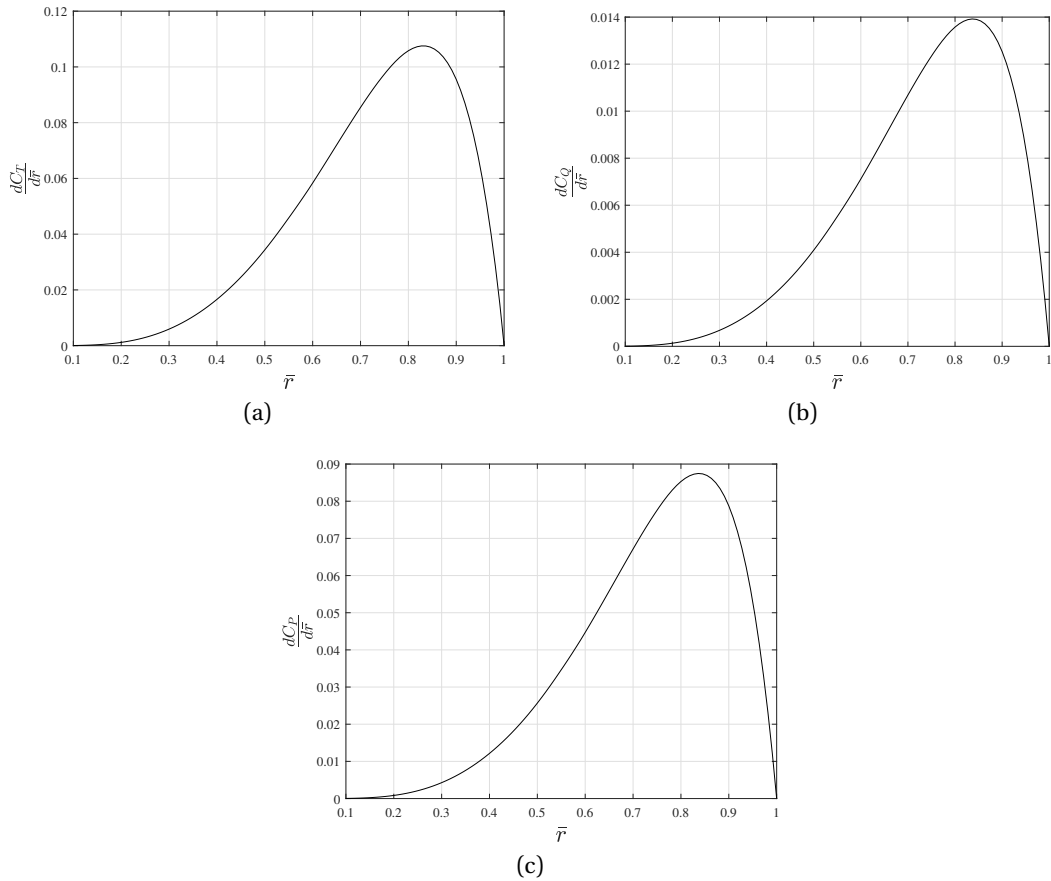


Figura 3.7: Distribuzioni del coefficiente di spinta $\frac{dC_T}{d\bar{r}}$, (a), del coefficiente di coppia $\frac{dC_Q}{d\bar{r}}$, (b), del coefficiente di potenza $\frac{dC_P}{d\bar{r}}$, (c) sulla pala. Si è tenuto conto dell'effetto del numero finito di pale attraverso la funzione di correzione di Prandtl. Rapporto di avanzamento $J = \frac{V_{cr}}{nD} = 0.67$.

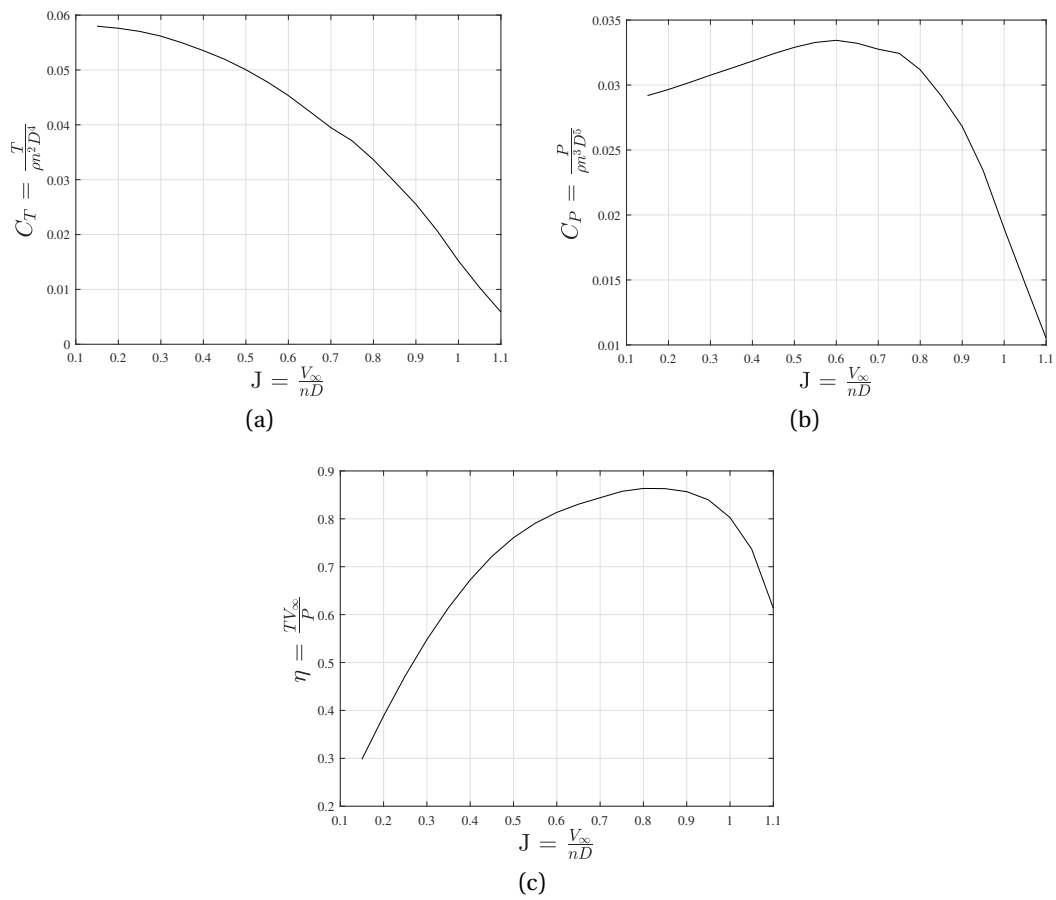


Figura 3.8: Prestazioni dell'elica progettata per il Tecnam P92; Coefficiente di spinta, C_T , (a), coefficiente di coppia, C_Q , (b) e efficienza, η , (c) al variare del rapporto di funzionamento

Parte II

McDonnell-Douglas AH-64

In questo capitolo verrà proposta l'analisi di una macchina elicottero e l'integrazione del sottoinsieme rotore all'interno della stessa. Si valuteranno poi le prestazioni della macchina per diverse condizioni di volo. La macchina in analisi è il McDonnell-Douglas(Boeing/Hughes) AH-64, anche noto come *Apache*, raffigurato in 4.1.



Figura 4.1: L'elicottero McDonnell-Douglas AH-64 *Apache* fotografato in azione.

Per quanto concerne l'implementazione dei metodi numerici impiegati che consentono di ricavare i grafici e le curve caratteristiche presentate nei prossimi paragrafi, si faccia riferimento all'appendice A.2, dove sono presenti alcuni dei codici sviluppati *ad hoc*.

CAPITOLO 5

LE CARATTERISTICHE TECNICHE

Il McDonnell-Douglas AH-64 è un elicottero d'attacco medio-pesante il cui progetto risale ai primi anni 70'. L'elicottero è stato concepito come bimotore biposto in tandem, per l'impiego in missioni anticarro, assalto, scorta e caccia anti-elicottero, da svolgere in ogni condizione di tempo e di luce. L'elicottero in analisi si avvale di una configurazione convenzionale con singolo rotore principale e rotore anti-coppia di coda, prima di presentare l'analisi delle prestazioni si riportano alcune caratteristiche tecniche salienti, nelle tabelle 5.1, 5.3 e 5.4.

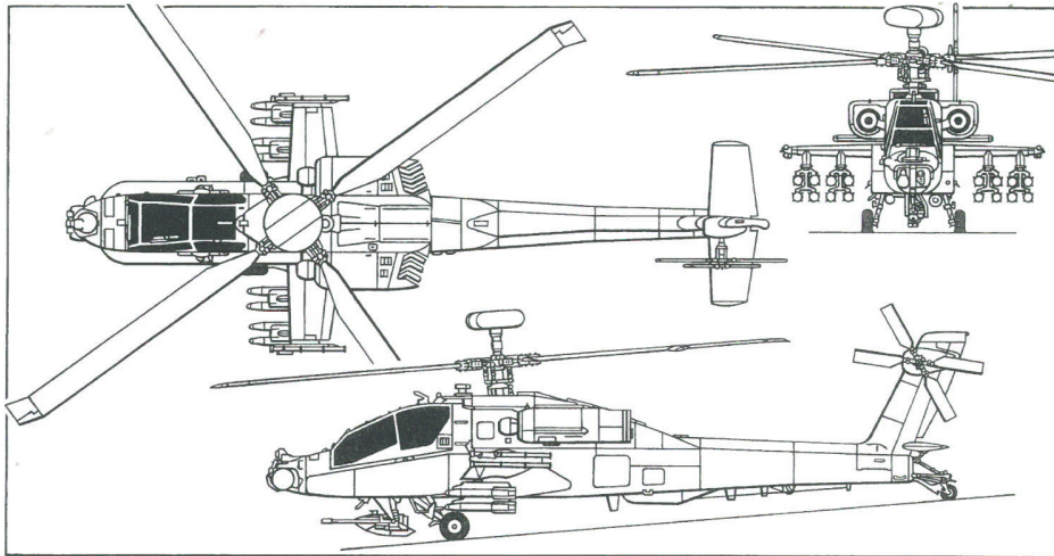


Figura 5.1: Trittico dell'elicottero McDonnell-Douglas AH-64 *Apache*, tratto dal [3].

Tutte le immagini e le tabelle presentate in questa parte dell'elaborato riguarda specificamente il velivolo di cui sopra e, per questo motivo, si è scelto di omettere tale riferimento all'interno delle didascalie di immagini e tabelle, per semplicità.

Weights	lb
Empty	10268
Maximum takeoff	17650
Fuel capacity	2422

Tabella 5.1: Alcuni pesi caratteristici del velivolo.

5.1 L'aerodinamica

Si è scelto di riportare, nelle figure 5.4 e 5.5, alcune curve caratteristiche del profilo alare che equipaggia il rotore principale, il HH-02 (Hughes-Helicopters), e il profilo che equipaggia il rotore di coda, il NACA 63-414. Come è possibile notare dalla figura 5.2, il profilo che equipaggia il rotore principale è dotato di una porzione di corda con curvatura negativa: in genere l'utilizzo di un profilo con questa caratteristica consente di ottenere un ridotto momento di cerniera tramite un *rear loading* negativo che riduce il momento aerodinamico calcolato ad $1/4$ della corda, dunque un minore sforzo di barra.

In tabella 5.2 sono riportati i dati geometrici relativi al rotore principale e di coda. Questi, ed in generale anche quelli specificati per l'elicottero, sono tratti dai manuali [3] e [4]. L'unico parametro che è stato necessario stimare, sulla base dei valori tipici per la classe di elicotteri, è stato il numero di Lock. In base al confronto con elicotteri della stessa classe di peso, è stato selezionato il valore del numero di Lock riportato in tabella 5.2.

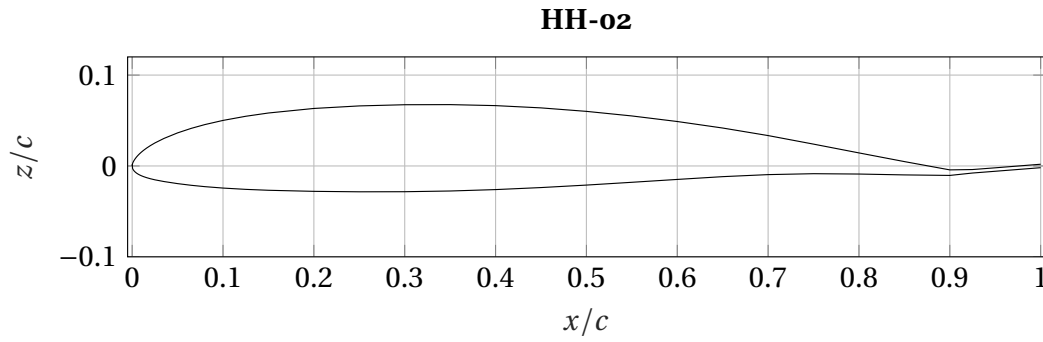


Figura 5.2: Rappresentazione del profilo alare montato sul rotore principale dell'elicottero in analisi.

		rotore principale	rotore di coda
raggio rotore, R	[ft]	24	4.6
corda, c	[ft]	1.75	0.83
solidità, σ	[/]	0.092	0.231
numero di pale, N	[/]	4	4
velocità angolare, Ω	[ft/s]	726	677
profilo alare	[/]	HH-02	NACA 63-415
washout, θ_t	[deg]	-9	-8
range di collettivo, θ_0	[deg]	+1 to +19	-15 to +27
Numero di Lock, γ	[/]	7	/

Tabella 5.2: Alcune caratteristiche geometriche dei rotori dell'elicottero in analisi.

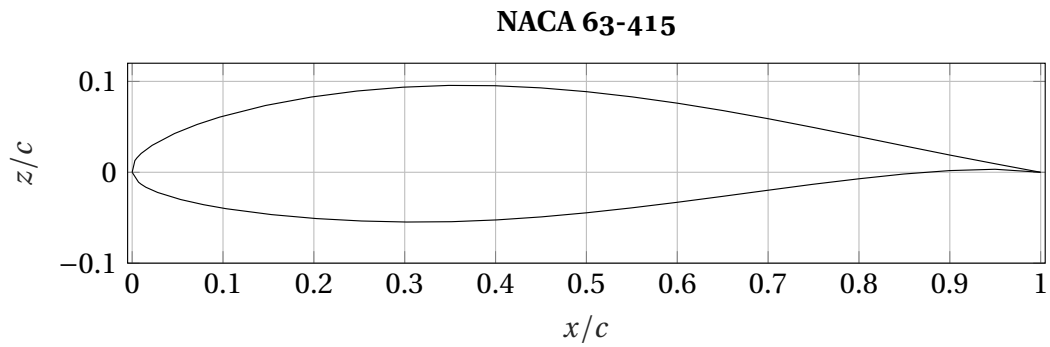


Figura 5.3: Rappresentazione del profilo alare montato sul rotore anti-coppia di coda dell'elicottero McDonnell-Douglas AH-64.

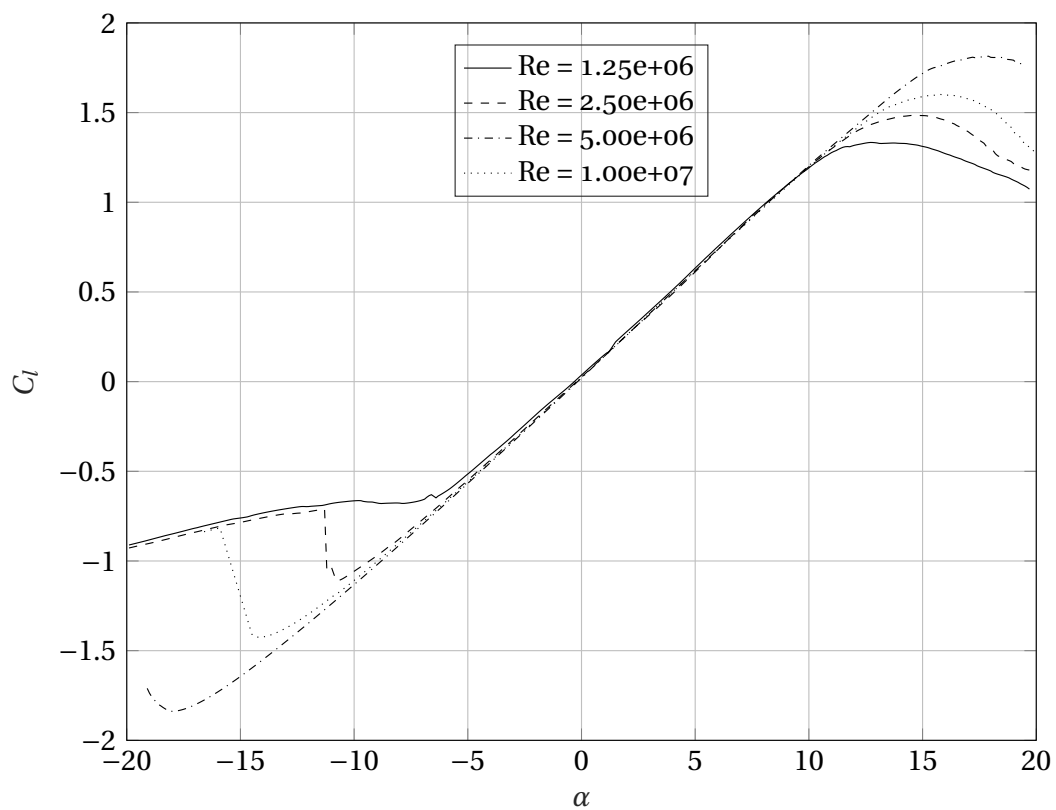


Figura 5.4: Rappresentazione della caratteristica di portanza del profilo HH-02 per diversi numeri di Reynolds. Curve ottenute attraverso il *software* X-foil.

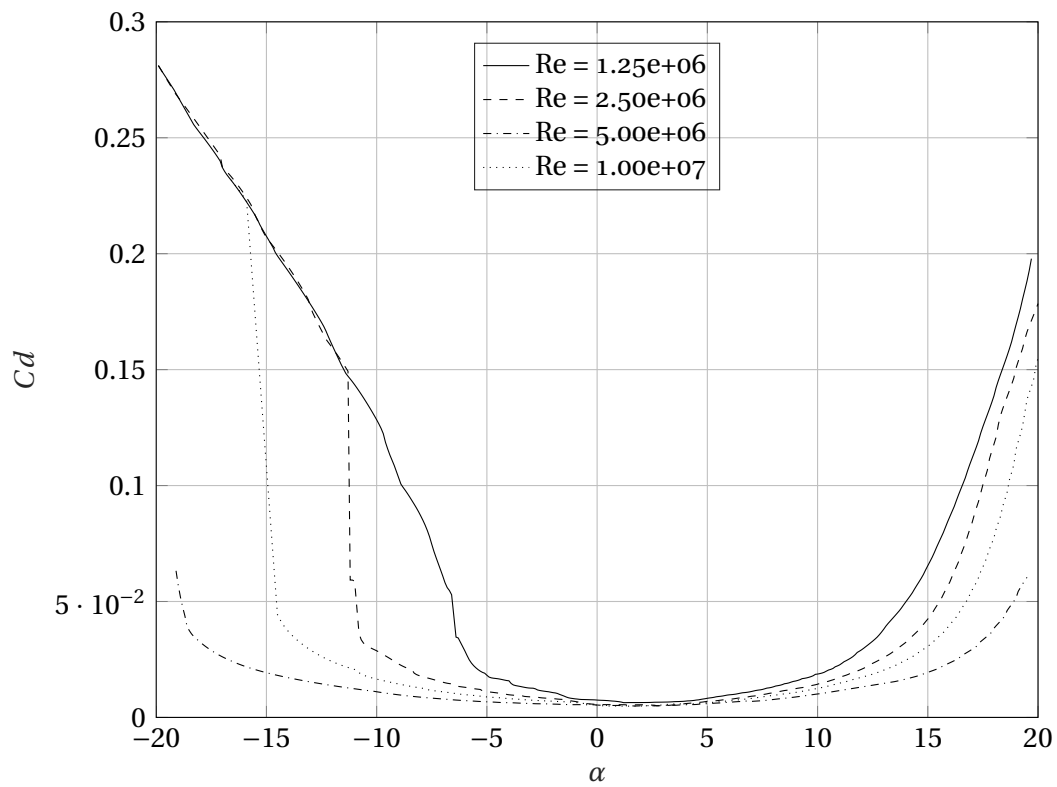


Figura 5.5: Rappresentazione della curva $C_d - \alpha$ per profilo HH-02 per diversi numeri di Reynolds. Curve ottenute attraverso il *software* X-foil.

5.2 Il sistema propulsivo

La macchina in analisi è equipaggiata con due motori T700-GE-701, forniti da General Electric, in figura 5.6 è riportata una vista in sezione. Le caratteristiche salienti del propulsore sono riportate in tabella 5.3.

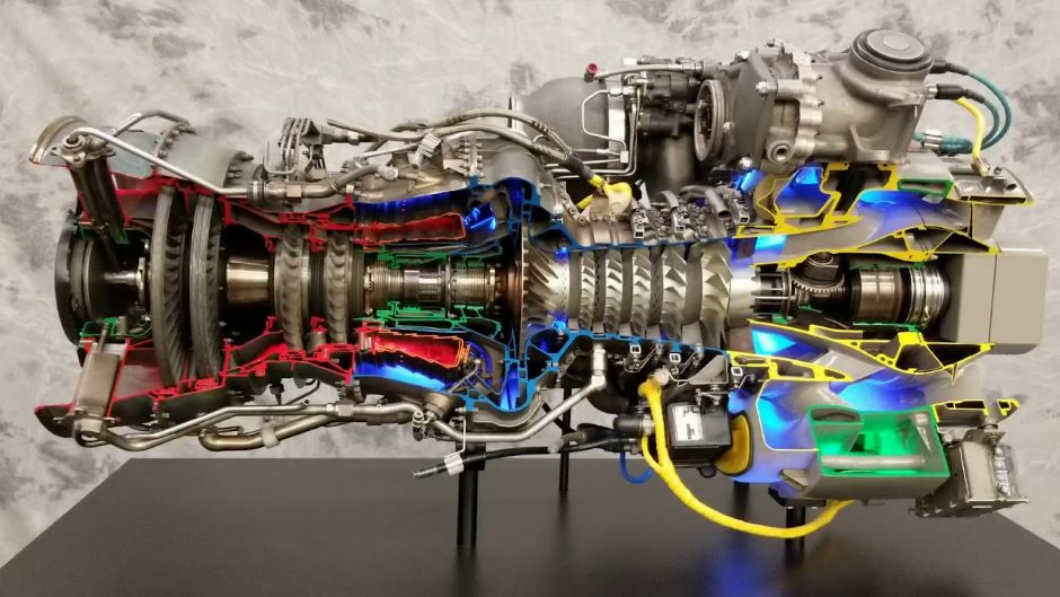


Figura 5.6: Rappresentazioni in sezione di uno dei motori con cui è equipaggiato l'elicottero in analisi.

regime	potenza all'albero [kW]	SFC [lb/(SHP × hr)]
massima continua	1279	0.462
massima	1486	0.465
contingency	1491	/

Tabella 5.3: Alcune caratteristiche del propulsore T700-GE-701 calcolate al livello del mare e in condizioni di atmosfera standard.

lunghezza	117cm
diametro	40cm
peso	207kg

Tabella 5.4: Alcune caratteristiche geometriche del propulsore T700-GE-701 calcolate al livello del mare e in condizioni di atmosfera standard.

Soffermandoci sempre sulle caratteristiche del sistema propulsivo, nelle seguenti sezioni si è assegnato un fattore di perdita della trasmissione pari a $\eta_t = 1.03$, che corrisponde quindi ad un aggravio di potenza richiesta pari al 3% a causa del sistema di trasmissione. Si ricorda, infatti, che i parametri riportati in tabella 5.3 fanno riferimento ai valori di potenza misurati al banco. Anche in questo caso, si è scelto di adottare un parametro tipico per le perdite di installazione del motore, tratto dal [2], identificato proprio come " fattore delle perdite di trasmissione ".

6.1 Prestazioni in salita assiale e *hover*

In questa sezione verrà presentata un'analisi delle prestazioni del rotore isolato in condizioni di flusso assiale e *hover*. Mediante la procedura di calcolo presentata a p.68 di [1], è stato possibile ricavare gli andamenti del coefficiente di spinta, T_c , del coefficiente di potenza Q_c e le relative distribuzioni lungo la pala, al variare del rapporto di funzionamento, μ , definito come il rapporto della velocità di salita e la *tip speed*: $\mu = \frac{V_c}{\Omega R}$. Si noti che la procedura numerica sviluppata fa riferimento al *database* aerodinamico presentato nelle figure 5.4 e 5.5. Si è scelto di adottare un unico valore del numero di Reynolds medio lungo la pala pari a $1.25 \cdot 10^6$, viceversa si è scelto di non tenere in conto degli effetti di comprimibilità, per cui si è fissato $M = 0$. Riguardo la scelta del numero di Reynolds al quale calcolare le caratteristiche aerodinamiche del profilo, si noti che, nel caso del volo assiale, per il quale la componente di velocità proveniente dalla traslazione del rotore nello spazio è trascurabile (matematicamente nulla nel caso di *hover*), il numero di Reynolds al quale lavorano i profili è esclusivamente funzione della velocità angolare del rotore e della posizione sulla pala.

6.1.1 Le polari e altre curve caratteristiche

In questo paragrafo si sono riportate le curve caratteristiche, la polare in *hover*, e l'andamento della figura di merito.

Le curve caratteristiche sono parametrizzate rispetto all'*input* di collettivo fornito dal pilota, indicato con θ_0 . Tale parametro varia all'interno dell'intervallo seguente, come indicato in [4]: $\theta_0 \in [10^\circ; 19^\circ]$. Le curve rappresentate nelle figure 6.1, 6.2 riportano, rispettivamente, gli andamenti del coefficiente di spinta e del coefficiente di coppia in funzione di μ , la polare in *hover* ed infine la cifra di merito al variare del coefficiente di spinta T_c . In figura 6.3 è riportato un confronto fra il coefficiente di spinta ottenuto tenendo in conto del numero finito di pale attraverso la funzione di correzione di Prandtl e quello ottenuto semplicemente limitando l'estremo superiore della quantità $dC_T/d\bar{r}$. Come è possibile notare dalla figura di cui sopra, lo scarto massimo fra le due teorie è limitato a circa l'8% del valore totale della spinta generata.

Il parametro ΔT_c è stato calcolato attraverso la relazione (6.1), nella quale $T_{c,Pr}$ fa riferimento al coefficiente di spinta calcolato attraverso la funzione di correzione di Prandtl, mentre $T_{c,B}$ rappresenta il coefficiente di spinta ottenuto con il metodo semplificato di cui sopra.

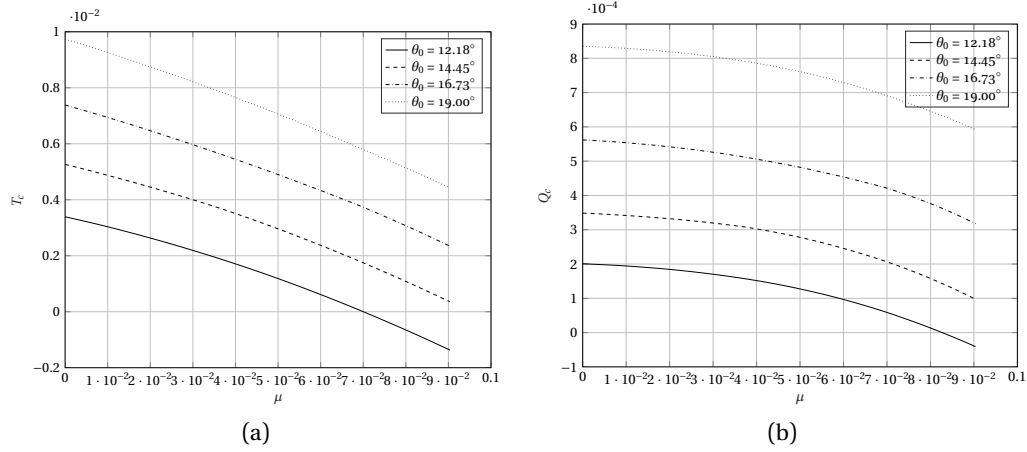


Figura 6.1: Rappresentazione del coefficiente di spinta del rotore (a), T_c , e del coefficiente di potenza (b), Q_c , al variare del rapporto di funzionamento, μ . Le curve sono parametrizzate al variare del comando di collettivo, θ_0 . Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità.

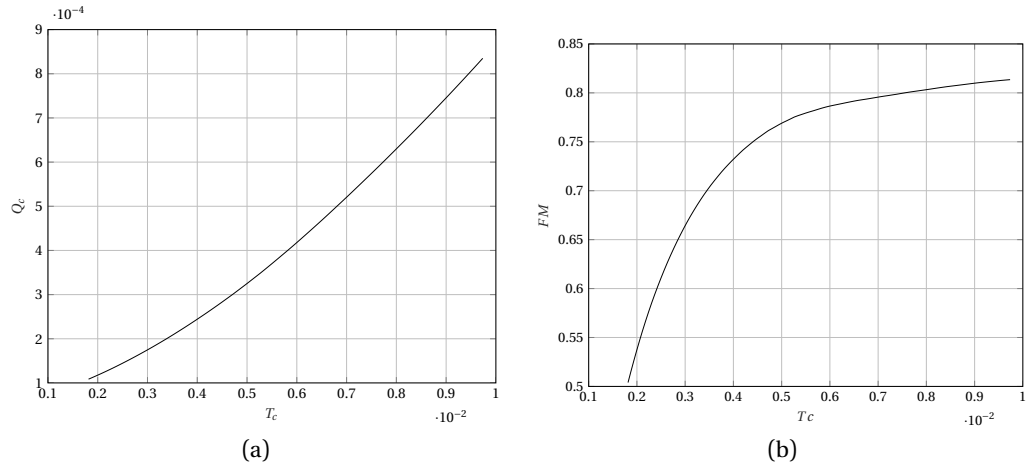


Figura 6.2: Rappresentazione della polare in *hover* (a) e della cifra di merito (b) per il rotore. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità.

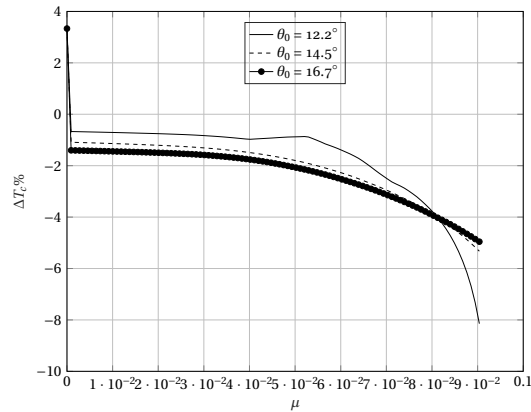


Figura 6.3: Confronto dell'effetto del numero di pale finito calcolato mediante la funzione di correzione di Prandtl e il metodo di limitazione dell'integrale.

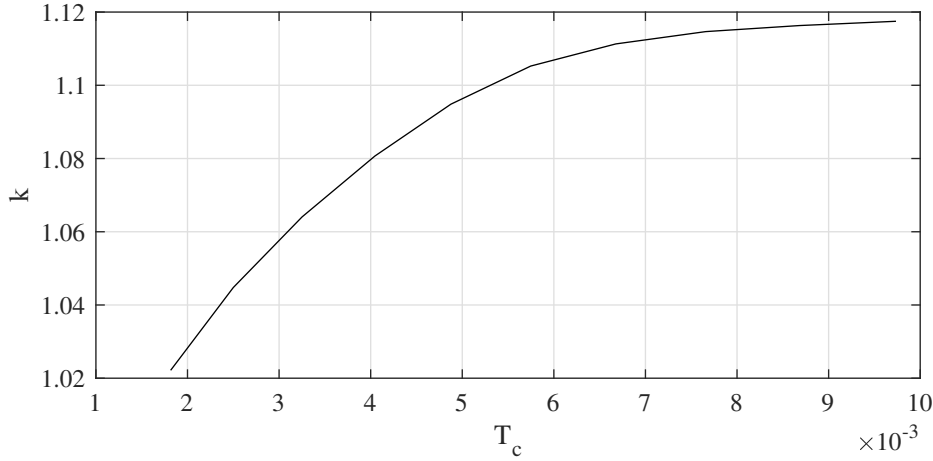


Figura 6.4: Fattore di correzione k per la non uniformità dell'induzione assiale sul rotore al variare della spinta in hover

$$\Delta T_c \% = \frac{T_{c,Pr} - T_{c,B}}{T_{c,Pr}} \times 100 \quad (6.1)$$

Il ridotto scarto fra i metodi analizzati rassicura riguardo il fatto che tenere conto delle perdite di estremità attraverso il metodo della limitazione dell'estremo superiore di integrazione possa essere considerato sufficientemente accurato.

Infine è stato possibile ricavare il valore del fattore correttivo k che permette di tenere in conto delle perdite dovute alla non uniformità dell'induzione nella valutazione della potenza indotta $P_{ci} = k T_c^{3/2} / \sqrt{2}$. Per ottenerlo è stata considerata la seguente relazione:

$$k = \frac{P_{ci}}{T_c^{3/2} / \sqrt{2}} = \frac{\int_0^1 \frac{\sigma}{2} C_l \varphi \bar{r}^3 d\bar{r}}{T_c^{3/2} / \sqrt{2}} \quad (6.2)$$

I risultati ottenuti facendo variare il calettamento di radice in una condizione di *hover* sono riportati in figura 6.4.

6.1.2 Le distribuzioni dei coefficienti spinta e coppia sulla pala

Si riportano adesso, in ipotesi di flusso assiale, le distribuzioni di coefficiente di spinta e coppia sulla pala. In un primo caso, figura 6.5, si è deciso di valutare cosa avviene, in regime di *hover* se si incrementa il comando di collettivo, θ_0 . Si noti che, variare l'angolo di collettivo, una volta fissato μ , equivale a far variare la spinta (assunta pari al peso). Da tale considerazione possiamo notare come, all'incrementarsi di θ_0 , incrementa di conseguenza la distribuzione di spinta e coppia sul disco.

In un secondo caso, invece, si è scelto di rappresentare la distribuzioni di spinta e coppia sulla pala una volta fissato il valore dell'input di collettivo, al variare del rapporto di funzionamento in salita, μ . Riportate nella figura 6.6, si può notare che all'aumentare della velocità di salita aumenta concordemente anche il coefficiente di spinta e coppia. Riguardo l'andamento delle distribuzioni di spinta sulla pala, il fatto che all'estremità della stessa sia computato un carico non nullo è associato al fatto che la correzione per gli effetti di estremità è stata condotta limitando semplicemente l'intervallo di integrazione e non applicandovi la funzione di correzione di Prandtl.

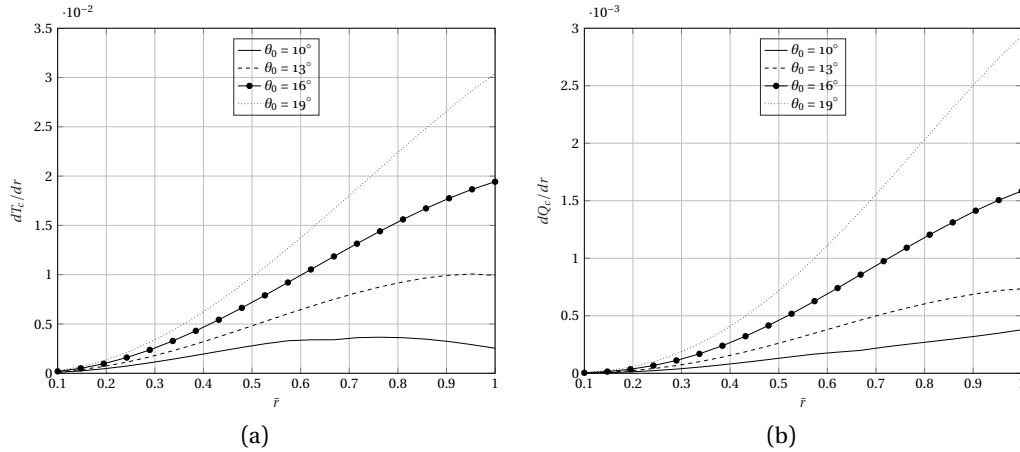


Figura 6.5: Distribuzione del coefficiente di spinta (a), T_c , e del coefficiente di potenza (b), Q_c , sulla pala, in *hover* ($\mu = 0$), per diversi valori di input del collettivo, nominalmente: $\theta_0 = 10^\circ$, $\theta_0 = 13^\circ$, $\theta_0 = 16^\circ$, $\theta_0 = 19^\circ$. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità. Le caratteristiche aerodinamiche 2D del profilo son calcolate per: $Re_\infty = 1.25 \times 10^6$ e per: $M_\infty = 0$.

6.2 Volo traslato

Per determinare le prestazioni del rotore in volo traslato, sarà necessario considerare che il rotore non è rigido bensì le pale sono incernierate alla testa della rotore mediante un certo numero di cerniere che ne permettono i moti di flappeggio, anticipo/ritardo e di variazione del passo. In questo capitolo verranno presentati i risultati di un'analisi del moto di flappeggio rispetto al piano di controllo, il quale coincide con il *piano di non variazione del passo* (si trascurano eventuali accoppiamenti tra i moti di flappeggio e di variazione del passo). In questo modo, trascurando il moto di anticipo/ritardo, la dinamica della pala articolata è completamente descritta se conosco gli angoli di flappeggio.

Oltre a queste considerazioni si faranno le seguenti ipotesi:

1. Rotazione della pala a velocità costante ($\Omega = \text{cost}$);
2. V_∞ costante e poco inclinata rispetto al piano di controllo;
3. Angolo di flappeggio massimo piccolo;
4. Eccentricità della cerniera di flappeggio nulla;
5. Moto di flappeggio di prima armonica;
6. Profilo alare costante, corda costante e calettamento lineare lungo la pala;
7. Angoli di inflow molto piccoli ($\varphi \ll 1$);
8. Gli elementi di pala lavorano in modo bidimensionale, nel tratto lineare della curva di portanza;
9. Induzione costante sul rotore;

6.2.1 Prestazioni

Nelle ipotesi citate prima, è possibile ricavare le caratteristiche del rotore mediante la procedura esposta a pag. 110 in [1].

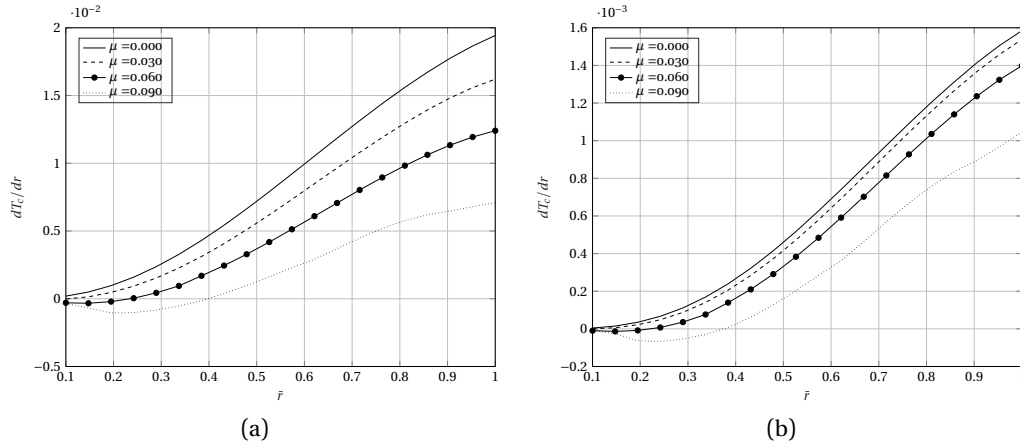


Figura 6.6: Distribuzione del coefficiente di spinta, T_c , e del coefficiente di potenza (b), Q_c , sulla pala, per diversi valori del rapporto di funzionamento. Il valore del collettivo si è assunto costante e pari a: $\theta_0 = 16^\circ$. Per l'analisi delle prestazioni del rotore si è assunto $B = 0.97$ per tenere conto delle perdite di estremità. Le caratteristiche aerodinamiche 2D del profilo son calcolate per: $Re_\infty = 1.25 \times 10^6$ e per: $M_\infty = 0$.

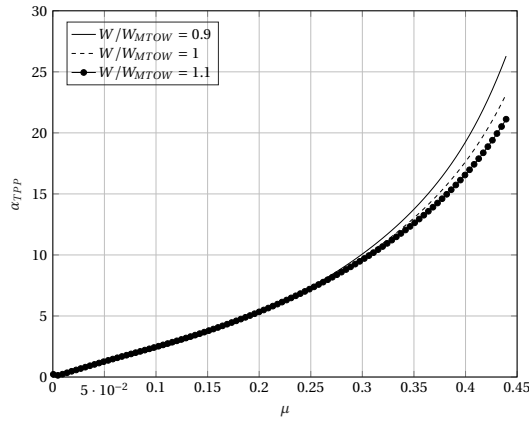


Figura 6.7: Andamento dell'angolo d'attacco del rotore al variare del rapporto di funzionamento $\mu = V_\infty / \Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso).

La spinta in questo caso viene assunta pari al massimo peso al decollo dell'elicottero. In particolare viene valutata l'influenza del peso sulle prestazioni del rotore analizzando anche le condizioni in cui il peso sia pari al 90% ed al 110% del peso massimo di decollo. I risultati inoltre vengono presentati al variare del rapporto di funzionamento μ tra zero (condizione di *hover*) ed il valore per il quale si ha la massima velocità di avanzamento dell'elicottero su cui è montato il rotore ($V_{ne} = 350 \text{ K m/h}$). In figura 6.7 viene riportato l'andamento dell'angolo d'attacco del rotore α_{TPP} . Esso aumenta all'aumentare della velocità di avanzamento, in quanto, a parità di spinta, questo dovrà essere tale da far sì che la componente di spinta nella direzione di avanzamento ($T \sin \alpha$) bilanci la somma della resistenza della fusoliera (D_{fus}) e della resistenza del rotore (H), entrambe crescenti con le perdite parassite e quindi con la velocità di avanzamento. All'aumentare del peso, α_{TPP} si riduce e questo effetto è evidente solo alle elevate velocità di avanzamento. Infine si vuole osservare che l'ipotesi di piccoli assetti del rotore (vedi ipotesi 2) viene meno alle alte velocità e quindi ciò potrebbe pregiudicare la validità di tali risultati.

In figura 6.8 vengono rappresentati il rapporto di ingresso λ e l'induzione assiale λ_i . Si osserva che l'induzione è massima in *hover* e si riduce con la velocità di avanzamen-

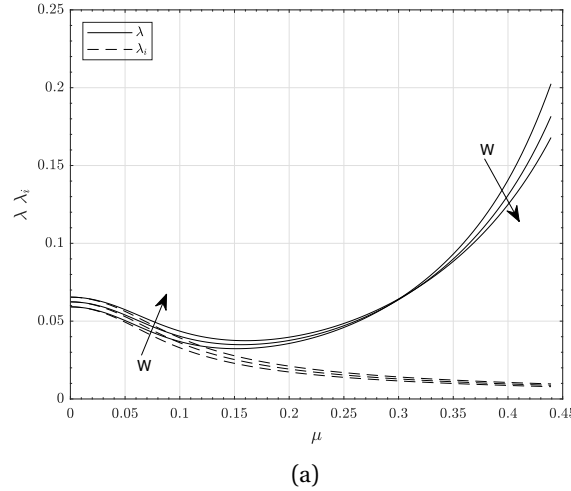


Figura 6.8: Andamento del rapporto di ingresso λ e dell'induzione assiale λ_i al variare del rapporto di funzionamento $\mu = V_\infty/\Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso).

to. La curva dell'induzione sale all'aumentare del peso in quanto essa dipende direttamente dalla spinta generata dal rotore. Per quanto riguarda il rapporto di ingresso $\lambda = \mu \tan \alpha_{TPP} + \lambda_i$, possiamo vedere che nella condizione di *hover* esso è proprio pari all'induzione assiale; all'aumentare della velocità di avanzamento esso inizialmente si riduce, in quanto $\mu \tan \alpha_{TPP}$ è piccolo e quindi il rapporto di ingresso tende a seguire l'andamento dell'induzione assiale; man mano che la velocità di avanzamento cresce il termine $\mu \tan \alpha_{TPP}$ diviene preponderante e tende a far crescere il valore di λ . L'effetto del peso è diverso a seconda del regime di volo, infatti a basse velocità si ha un aumento di λ dovuto al fatto che il termine preponderante in quel regime di volo è l'induzione, che aumenta con il peso, mentre ad alte velocità di volo l'effetto del peso è quello di far ridurre λ in quanto α_{TPP} decresce.

In figura 6.9 vengono riportati gli andamenti del coefficiente di potenza P_c , di resistenza H_c e di forza laterale Y_c del rotore. Si può osservare come in *hover* la resistenza e la forza laterale si annullano in virtù della simmetria del campo di moto rispetto all'asse del rotore. All'aumentare della velocità di avanzamento sia H_c che Y_c aumentano (per via dell'incremento delle perdite parassite). L'incremento di peso causa un leggero aumento di questi coefficienti per velocità di avanzamento intermedie. Per quanto riguarda il coefficiente di potenza, all'aumentare della velocità di avanzamento, esso subisce una iniziale decrescita, per via della riduzione dell'induzione e quindi della potenza indotta (preponderante in *hover* ed alle basse velocità), dopodiché l'incremento delle perdite parassite alle alte velocità porta ad un brusco innalzamento del P_c . L'aumento di peso porta ad un incremento della potenza richiesta, tuttavia questo effetto è rilevante alle basse velocità e diviene sempre meno importante all'aumentare della velocità di avanzamento. Nella stessa figura, è possibile osservare anche l'andamento del comando collettivo (passo di radice). A partire dalla condizione di *hover*, esso subisce una iniziale riduzione per via dell'aumento di pressione dinamica media vista dagli elementi di pala e di conseguenza, a parità di portanza, è richiesto un'assetto minore. Dopodiché, l'aumento del rapporto di ingresso e quindi dell'angolo di *inflow* $\varphi \approx u_p/t_T$ diviene preponderante e quindi, per mantenere l'assetto necessario alla produzione della spinta richiesta, l'angolo θ_0 dovrà aumentare. All'aumentare del peso è richiesto un collettivo maggiore per velocità di avanzamento medio basse, dopodiché l'effetto si inverte per via dell'effetto del peso sul rapporto di ingresso λ e sull'angolo d'attacco del rotore.

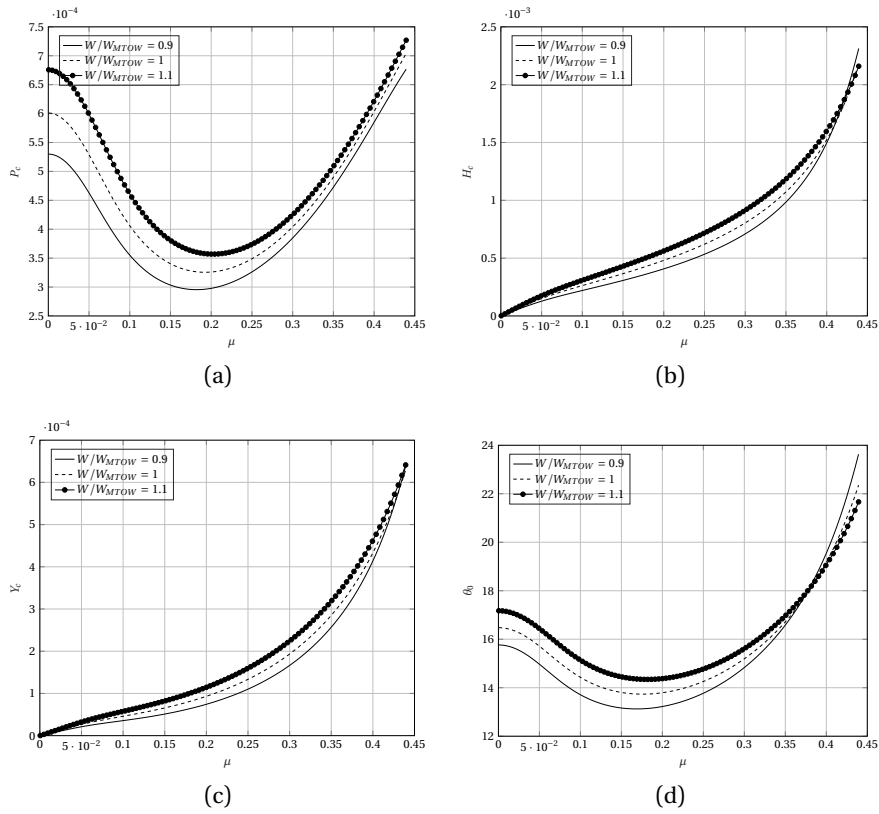


Figura 6.9: Andamento del coefficiente di potenza P_c (a), di resistenza H_c (b) e di forza laterale Y_c (c) del rotore al variare del rapporto di funzionamento $\mu = V_\infty/\Omega R$ per 3 diversi valori di spinta assegnata (supposta pari al peso). Inoltre viene anche riportato l'andamento comando collettivo, ossia il passo di radice θ_0 (d).

6.2.2 I coefficienti di flappeggio

La condizione di volo traslato richiede l'introduzione di opportune cerniere in prossimità del collegamento tra le pale ed il mozzo, che consentano di articolare il movimento della pala ed evitare che le sollecitazioni periodiche che si verificano al variare dell'angolo di azimuth si trasferiscano all'albero motore compromettendone l'integrità strutturale. Nelle nostre ipotesi il rotore articolato è dotato esclusivamente di una cerniera di flappeggio, oltre alla cerniera di variazione passo, avendo trascurato il moto di brandeggio. Per il calcolo dell'angolo di flappeggio, β , si è assunta l'ipotesi di prima armonica, in seguito alla quale vale la (6.3).

$$\beta = \beta_0 + \beta_{1,s} \sin \Psi + \beta_{1,c} \cos \Psi \quad (6.3)$$

In figure 6.10 sono riportati i coefficienti di flappeggio. Possiamo osservare che l'angolo di conicità è positivo e circa invariante con la velocità di avanzamento mentre i coefficienti di flappeggio longitudinale e laterale sono negativi ed aumentano in modulo all'aumentare della velocità. Inoltre si osserva che tutti e tre i coefficienti aumentano in modulo all'aumentare del peso.

Infine osserviamo che l'ipotesi di piccoli angoli massimi di flappeggio (ipotesi 3) è sufficientemente soddisfatta solo per $\mu < 0.3$.

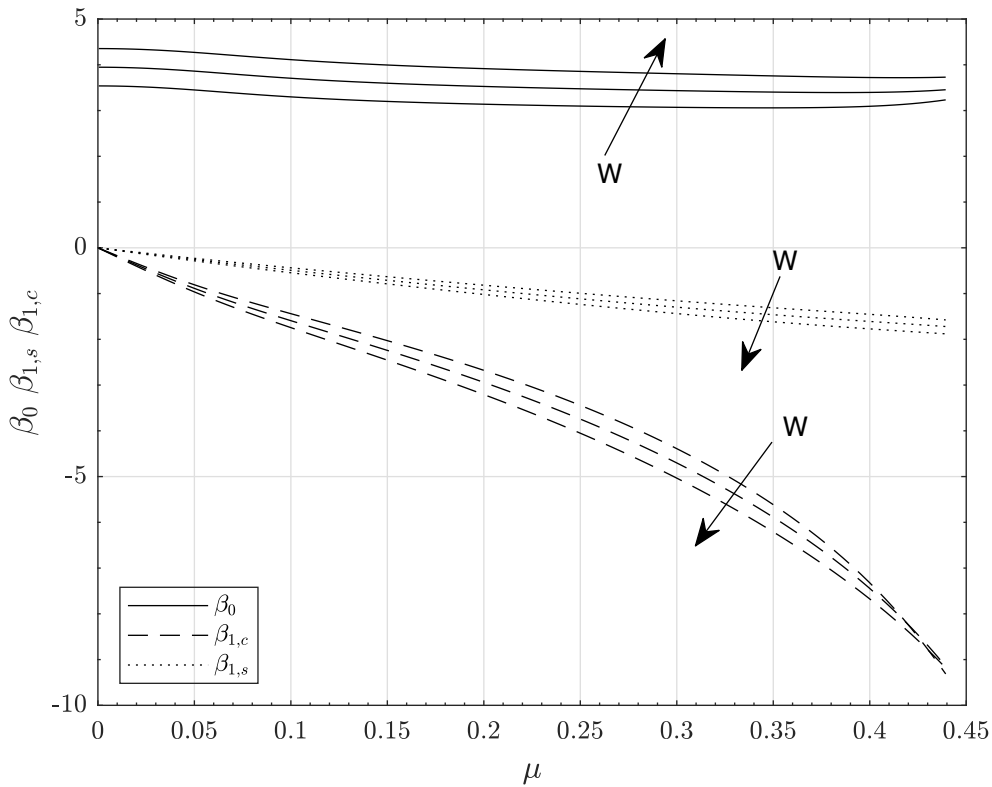


Figura 6.10: Andamento dei coefficienti di flappeggio al variare del rapporto di avanzamento in volo traslato $\mu = \frac{V_\infty}{\Omega R}$

6.2.3 Le distribuzioni di angoli di attacco sul disco

Ricavati gli angoli di flappeggio, è possibile risalire agli angoli di attacco effettivi visti dagli elementi di pala sul rotore mediante la relazione:

$$\alpha_{eff}(\bar{r}, \Psi) = \theta(\bar{r}, \Psi) - \arctan \frac{\lambda + \beta \bar{r} / \Omega + \beta \mu \cos \Psi}{\bar{r} + \mu \sin \Psi} \quad (6.4)$$

$\forall \bar{r} \in [\bar{r}_{hub}, 1]$, $\Psi \in [0, 2\pi]$. Valida nelle ipotesi in cui il moto è *quasi-stazionario*, ossia caratterizzato da una frequenza ridotta $k < 0.05$ (nel nostro caso, $k = c/2R = 0.0362$, vedi 5.2).

In figura 6.11 sono mostrati dei *contour plot* dell'angolo d'attacco effettivo sul disco rotore per valori del rapporto di avanzamento $\mu = \frac{V_\infty}{\Omega R}$ compresi tra zero (condizione di *hover*) ed il valore per il quale si ha la massima velocità di avanzamento dell'elicottero su cui è montato il rotore ($V_{ne} = 350 \text{ K m/h}$).

Si può osservare che in *hover* la distribuzione degli angoli di attacco è simmetrica.

Nel caso di rotore in volo traslato ($\mu > 0$), l'asimmetria del campo di moto si riflette anche sulla distribuzione degli angoli di attacco. In particolare, si ha la genesi di una regione di *reverse flow* per $\Psi = 270^\circ$ la quale cresce all'aumentare della velocità di avanzamento. All'interno di questa regione l'angolo d'attacco diviene fortemente negativo, per questo motivo l'ipotesi con cui sono stati ricavati gli angoli di flappeggio e quindi gli angoli d'attacco in questione, non sono ben verificate in questa zona e quindi potrebbero essere poco accurati, per questo motivo si è scelto di trascurarli nei diagrammi qui presentati.

Osserviamo inoltre che l'angolo d'attacco massimo, non supera mai il valore di fine linearità della curva di portanza, il quale è circa 12° per $Re = \frac{\rho(\Omega R)\bar{c}}{\mu} \approx 8 \cdot 10^6$. (Si noti che il numero di Reynolds utilizzato per confrontare l'angolo di attacco di fine linearità è sottosimato, dunque conservativo, dal momento che viene calcolato esclusivamente attraverso la componente ΩR).

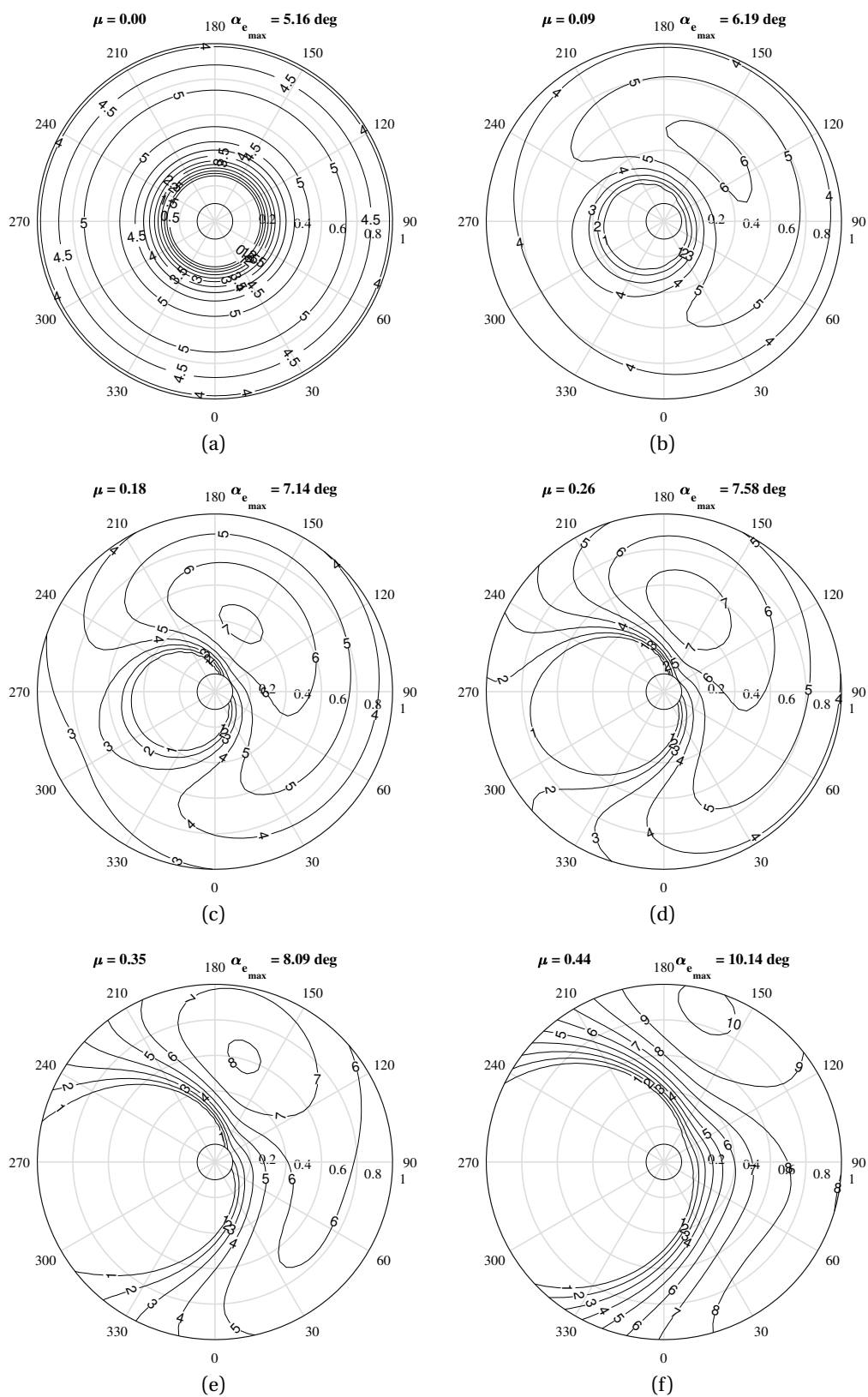


Figura 6.11: Evoluzione degli angoli di attacco effettivi sul disco rotore al variare del rapporto di avanzamento μ .

6.2.4 Le distribuzioni del numero di Mach sul disco

Come visto nella sezione precedente, è noto che durante la fase di volo traslato sul rotore agisca una pressione dinamica non omogenea, funzione sia dell'angolo di azimuth della pala che della stazione radiale che stiamo considerando. Sempre nelle ipotesi di moto *quasi-stazionario* possiamo ricavare la distribuzione del numero di Mach sul disco, una volta noti gli angoli di flappeggio sul rotore, mediante la seguente relazione:

$$M_e(\bar{r}, \Psi) = \frac{\Omega R \sqrt{(\lambda + \dot{\beta} \bar{r} / \Omega + \beta \mu \cos \Psi)^2 + (\bar{r} + \mu \sin \Psi)^2}}{a_\infty} \quad (6.5)$$

In figura 6.12 sono mostrati dei *contour plot* del numero di Mach effettivo sul disco rotore per valori del rapporto di avanzamento $\mu = \frac{V_\infty}{\Omega R}$ compresi tra zero (condizione di *hover*) ed il valore per il quale si ha la massima velocità di avanzamento dell'elicottero su cui è montato il rotore ($V_{ne} = 350 \text{ K m/h}$).

In *hover* la distribuzione del numero di Mach è ovviamente simmetrica con un valore massimo pari a 0.65 all'estremità della pala.

All'aumentare della velocità di avanzamento la distribuzione diviene asimmetrica: il numero di Mach aumenta nella zona del disco in cui la pala è in avanzamento e si riduce in quella in cui è retrocedente.

Il valore massimo del numero di Mach cresce all'aumentare della velocità di traslazione, in particolare assume valori prossimi ad 1 già per $\mu > 0.1$. Questo ci dice ancora una volta che le ipotesi alla base di questi risultati non sono ben verificate in quanto gli effetti della comprimibilità non sono trascurabili.

In realtà, in queste analisi bisognerebbe tener in conto anche dell'effetto freccia. Infatti, come è possibile notare in figura 5.1 l'effettiva geometria delle pale prevede l'utilizzo di un certo angolo di freccia in prossimità della *tip* e del quale non si è potuto tenere conto nell'analisi, dal momento che la teoria sviluppata non consente l'introduzione di questa caratteristica geometrica, ma prevede una semplice forma in pianta rettangolare.

Assumendo un angolo di freccia $\Lambda = 45^\circ \forall \bar{r} > 0.97$ (ossia per il 3% della pala, come mostrato nelle figure 6.13 e 6.14) possiamo valutare come si riduce in prima approssimazione il numero di Mach sul disco nelle ipotesi in cui l'introduzione della freccia non causi una variazione nelle distribuzioni di induzione ed angoli di flappeggio. In tal caso il numero di mach può essere valutato come:

$$M_e(\bar{r}, \Psi) = \frac{\Omega R \sqrt{(\lambda + \dot{\beta} \bar{r} / \Omega + \beta \mu \cos \Psi)^2 + (\bar{r} + \mu \sin \Psi)^2}}{a_\infty} \cdot \cos \Lambda(\bar{r}) \quad (6.6)$$

Il risultato è presentato in figura 6.15. Si può osservare una riduzione del numero di Mach massimo, fissato il rapporto di avanzamento μ . Tuttavia, nonostante questa riduzione, il numero di Mach assume ancora valori prossimi all'unità per velocità di avanzamento modeste (μ di poco superiore a 0.01). Per una migliore previsione della distribuzione del numero di Mach, si dovrebbe tenere in conto anche dell'effettiva modifica del carico, delle induzioni e degli angoli di flappeggio indotte dall'angolo di freccia. Inoltre è bene notare che le caratteristiche della freccia, ovvero la porzione di corda per cui si estende, nonché l'angolo stesso, sono stati supposti in base al trittico, riportato in 5.1 e non perché direttamente disponibili.

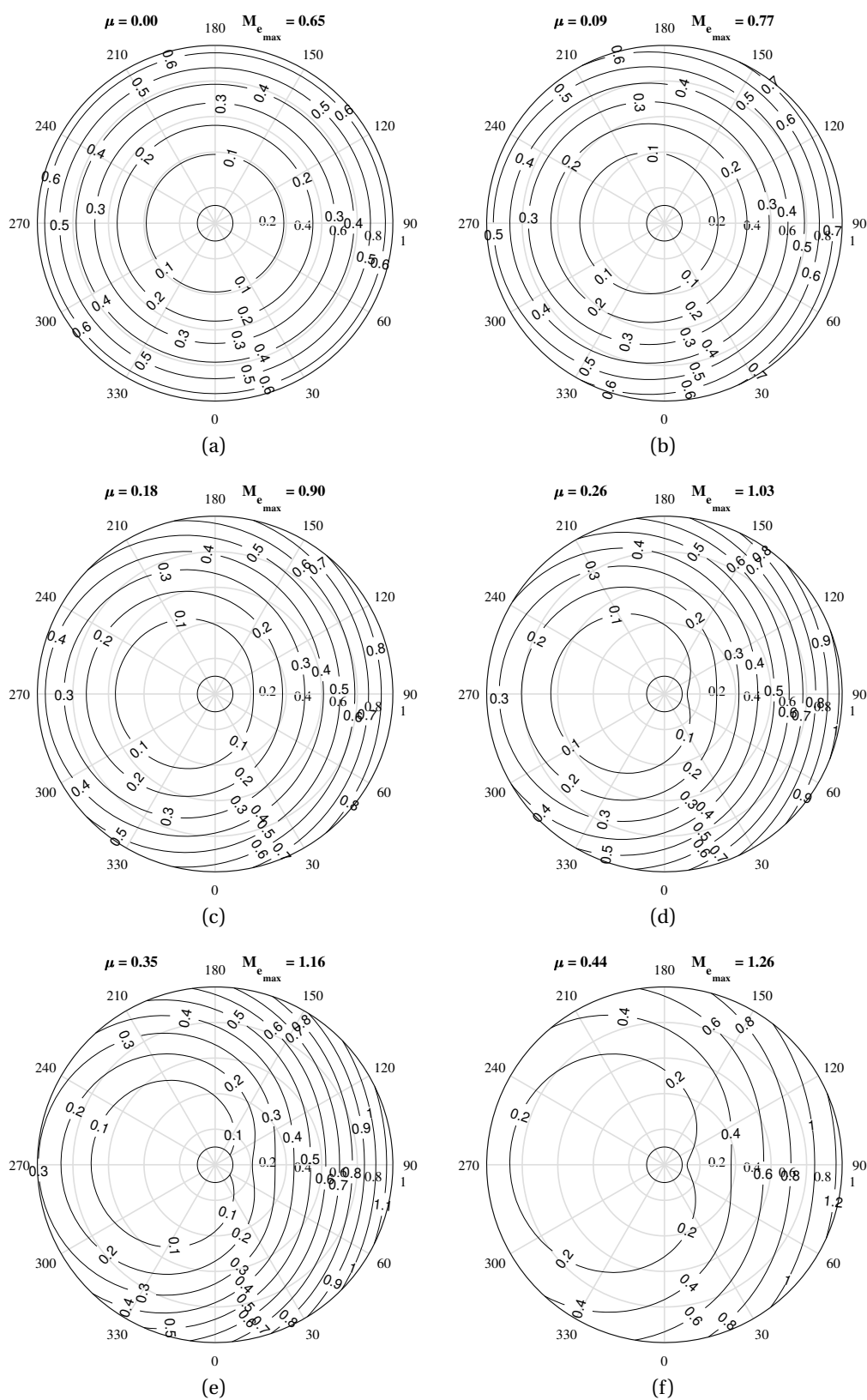


Figura 6.12: Evoluzione del numero di Mach effettivo sul disco rotore al variare del rapporto di avanzamento μ .



Figura 6.13: Forma assunta per pianta del rotore principale

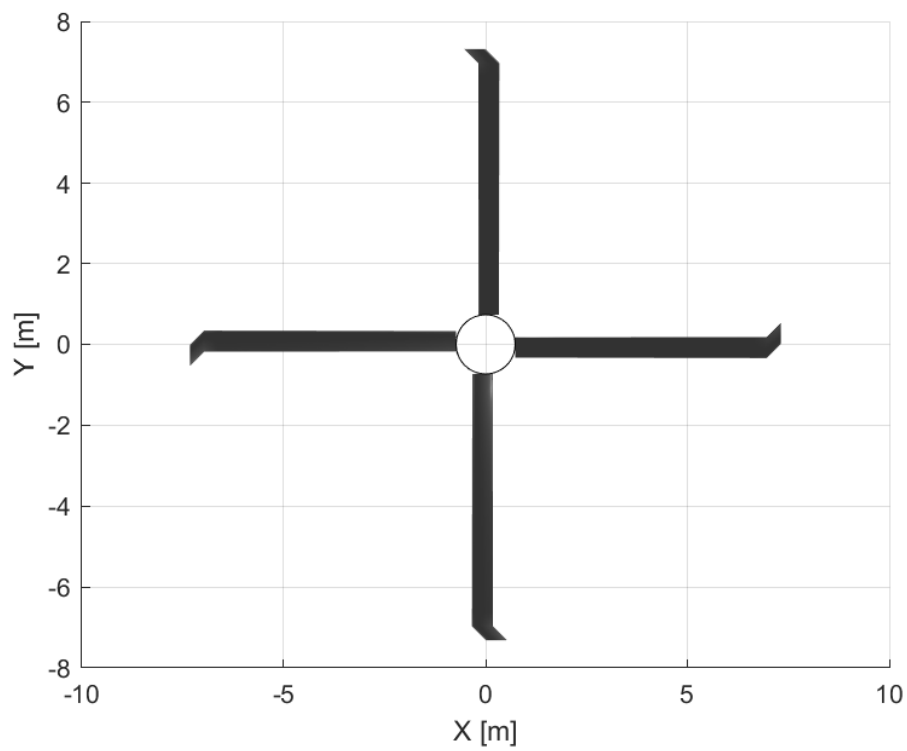


Figura 6.14: Forma in pianta assunta per rotore principale

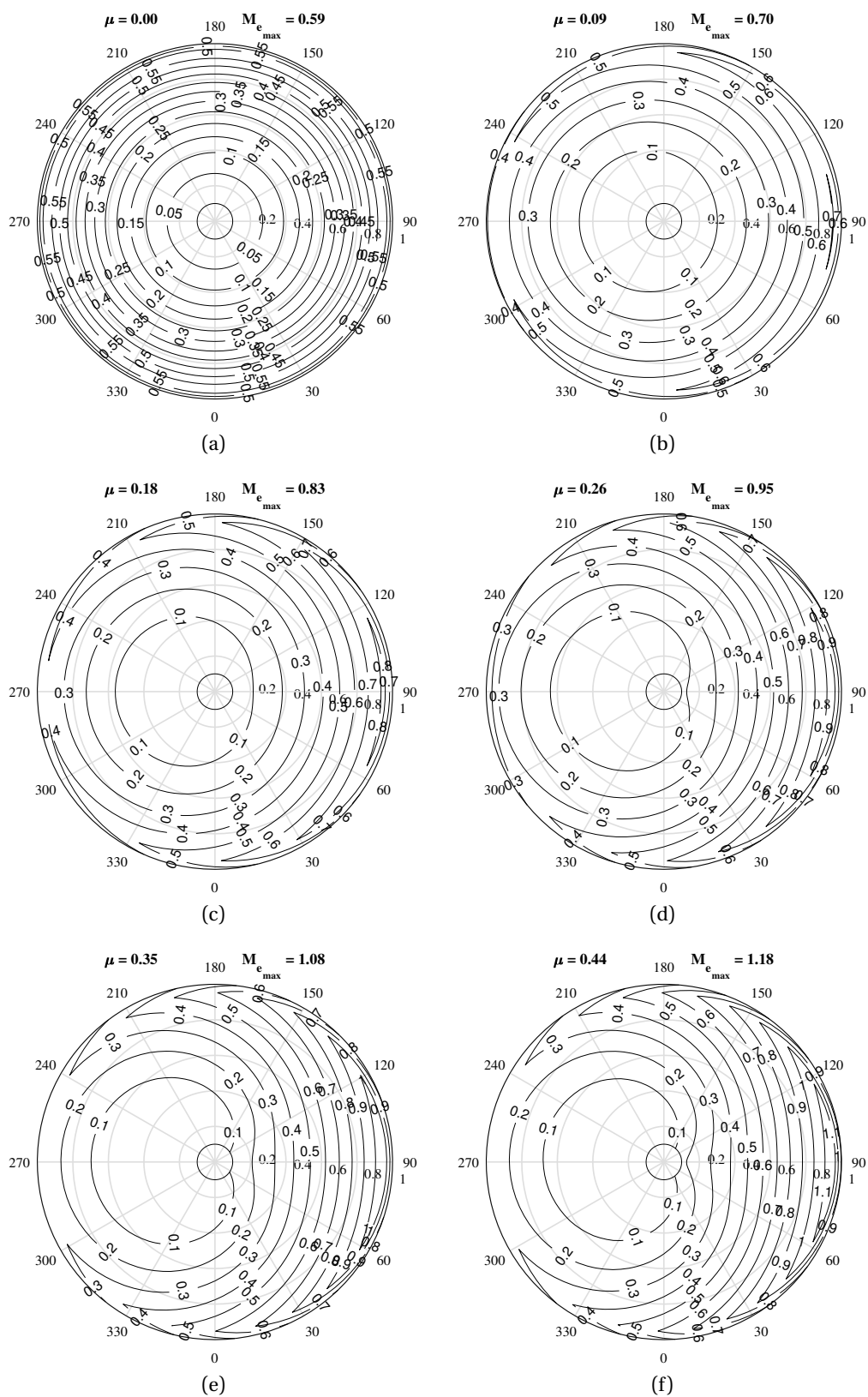


Figura 6.15: Evoluzione del numero di Mach effettivo sul disco rotore con freccia al tip al variare del rapporto di avanzamento μ

6.2.5 Il sentiero di stallo

In questo contesto si è deciso di implementare anche un'analisi del sentiero di stallo basato sull'angolo di stallo bidimensionale del profilo. Questa teoria conduce a delle mappe del sentiero di stallo come presentate in 6.16. In primo luogo viene rappresentata la distribuzione degli angoli di attacco in corrispondenza della prima sezione che stalla, riportata in 6.17. Dopodiché, una volta determinato il valore di μ in corrispondenza del quale avviene il primo stallo, è possibile aumentare progressivamente la velocità di traslazione per valutare la propagazione sul disco delle zone stallate. Si noti che il valore di primo stallo verrà indicato per comodità con la notazione μ^* ;

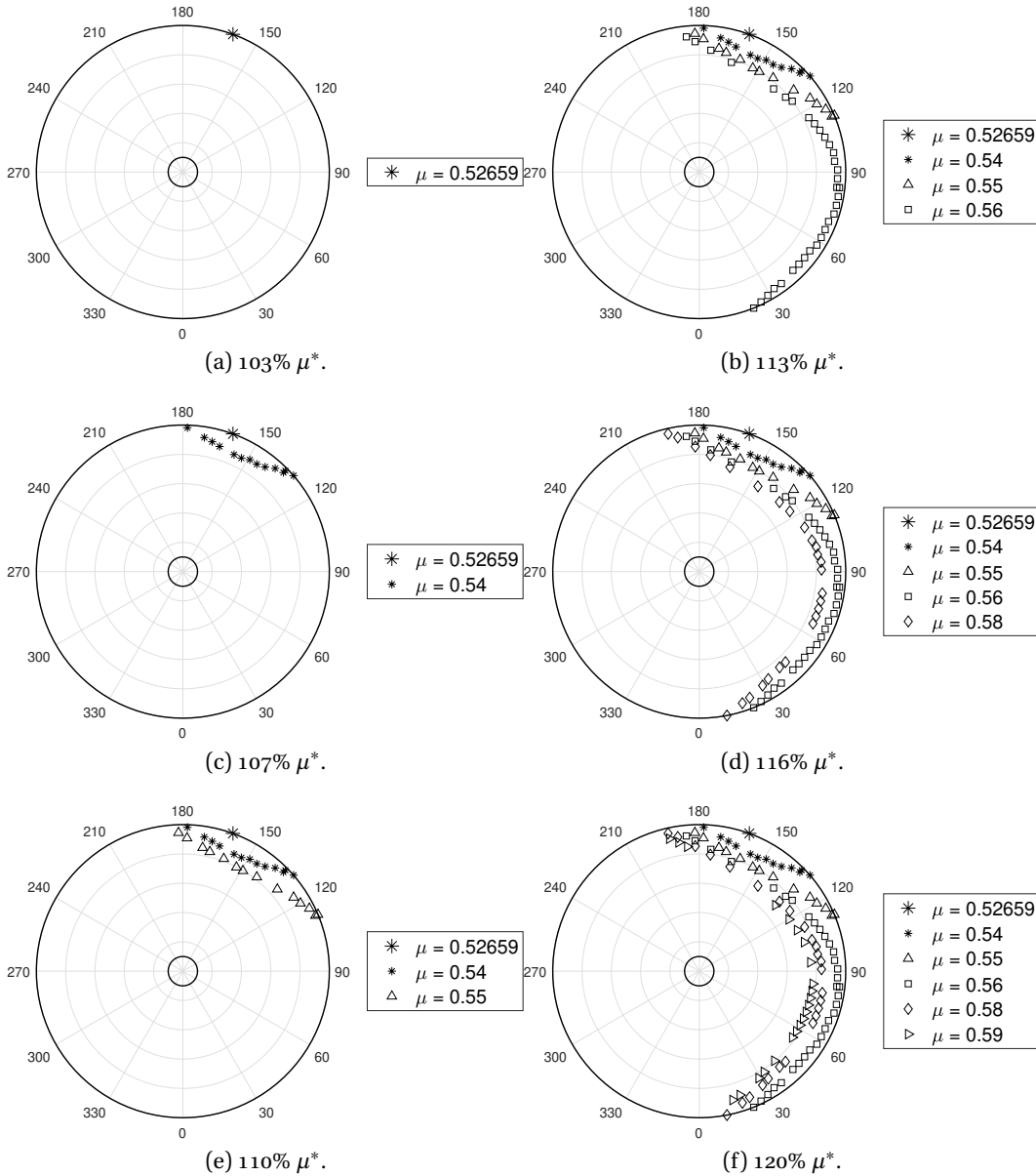


Figura 6.16: Evoluzione del sentiero di stallo per valori di μ che coincidono con la prima sezione di stallo fino ad una maggiorazione del 20% di tale rapporto di funzionamento.

Come previsto, all'aumentare della velocità traslazionale la porzione di rotore in stallo cresce progressivamente. Eventuali successivi sviluppi prevedono l'implementazione di più complessi metodi che tengano in conto del fenomeno dello stallo dinamico, non riportato in questo caso.

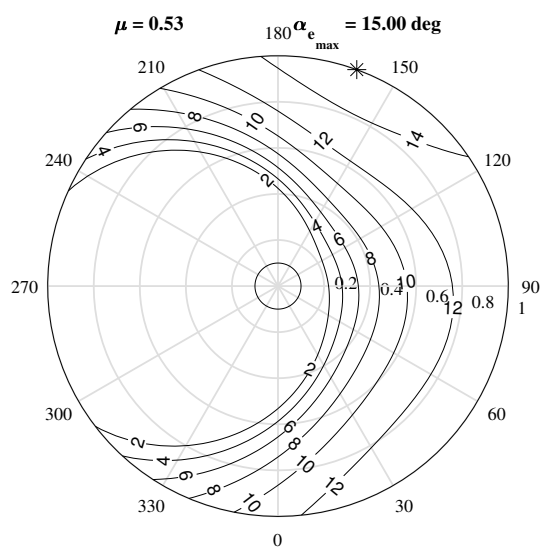


Figura 6.17: Distribuzione dell'angolo di attacco sul rotore in corrispondenza del valore del rapporto di funzionamento per cui si verifica il primo stallo di un profilo.

Una volta preso in considerazione il rotore come sotto-sistema isolato della macchina elicottero, in questo nuovo capitolo si intende valutare le prestazioni della macchina elicottero completa. In questo capitolo tutte le figure e le tabelle faranno riferimento all'elicottero AH-64, a cui è dedicata questa parte dell'elaborato, motivo per cui si è scelto di omettere tale specificazione all'interno del capitolo.

Infine si osserva che la procedure di calcolo delle prestazioni fanno riferimento alle equazioni riportate in cap. 6 paragrafo 7 di [2]. Dove sono state fatte le assunzioni circa alcuni parametri necessari per l'analisi riportate in tabella 7.1.

Osservando la tabella 7.1, si deve notare che il braccio del rotore anti-coppia di coda, l_{RC} , è stato stimato a partire dal trittico (vedi figura 5.1), dal momento che tale parametro non era riportato sui manuali di riferimento. Viceversa, il fattore di correzione della potenza che tiene conto della non uniformità dell'induzione sul rotore, k_{RP} è stato stimato nella sezione 3.1.1 per il rotore principale, mentre lo stesso parametro per il rotore anti-coppia di coda è stato stimato in base ai valori tipici adottati.

7.1 Le curve di potenza

Sempre con riferimento al capitolo 1, in cui sono state descritte le caratteristiche

tecniche della macchina in analisi, risulta interessante valutare la provenienza delle diverse aliquote di potenza assorbita. Questa tipologia di elicottero, definito convenzionale, con rotore anti-coppia di coda, presenta essenzialmente 4 principali contributi alla potenza assorbita, elencati di seguito:

- Potenza assorbita dal rotore principale;

Parametro	Valore assegnato	
Area parassita equivale, f	2	$[m^2]$
Braccio del rotore di coda, l_{RC}	9	$[m]$
Fattore di correzione, k_{RP}	1.12	$[/]$
Fattore di correzione, k_{RC}	1.4	$[/]$
Fattore delle perdite di trasmissione, η_t	1.03	$[/]$
Potenza richiesta dagli impianti ausiliari, P_{AUX} ,	20	$[kW]$
Coefficiente di resistenza di profilo, $C_{DP_{RP}}$	0.01	$[/]$
Coefficiente di resistenza di profilo, $C_{DP_{RC}}$	0.01	$[/]$

Tabella 7.1: Alcuni parametri stimati necessari per il calcolo delle prestazioni

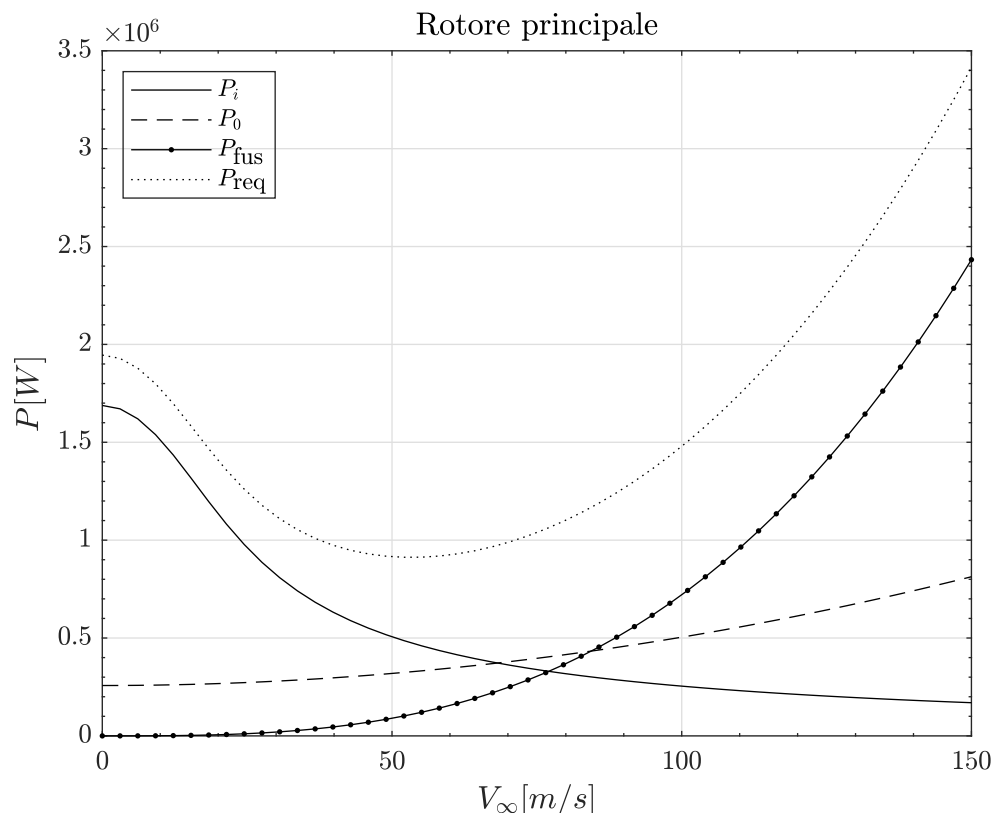


Figura 7.1: Potenza totale richiesta P_{req} dal rotore principale divisa nei contributi di: Potenza indotta P_i , parassita P_0 e dovuta alla resistenza della fusoliera P_{fus} .

- Potenza assorbita dal rotore anti-coppia di coda;
- Potenza assorbita dai sistemi ausiliari;
- Potenza parassita di fusoliera.

In questa sede verranno proposti tutti i vari contributi di cui sopra, rappresentati nelle immagini 7.1, 7.2 e 7.3. Contestualmente si noti che nelle successive figure il parametro di potenza disponibile, P_{av} farà sempre riferimento, fatta eccezione ove specificato, al regime di massima potenza continua erogata da entrambe i motori installati.

Una volta chiarita la decomposizione della potenza assorbita da un elicottero in diverse condizioni di volo, può risultare di particolare interesse valutare in che modo le suddette curve di potenza, in particolare quelle totali, si modificano al variare del peso della macchina. La tabella 5.1 riporta sinteticamente alcuni parametri di interesse riguardo le masse del velivolo in analisi. Per quello che si è detto riguardo le masse si è scelto di parametrizzare le curve di potenza totale rispetto a 4 differenti valori del peso al decollo, che sono riportati di seguito e che sono stati presi all'interno dell'intervallo $[W_{min}, W_{mtow}]$, dove W_{min} indica il peso a vuoto dell'elicottero somato al massimo carico di carburante consentito con i serbatoi interni; viceversa W_{mtow} indica il massimo peso al decollo -*maximum take off weight*-, sempre considerato con la stessa massa di carburante pari al massimo della capienza dei serbatoio interni. (Si noti contestualmente che l'architettura in analisi permette di ospitare anche ulteriori serbatoi *esterni*, non considerati in questo contesto, che permettono di estendere le prestazioni di *Endurance* e *Range*).

1. $W_1 = 6460 \text{ kg}$

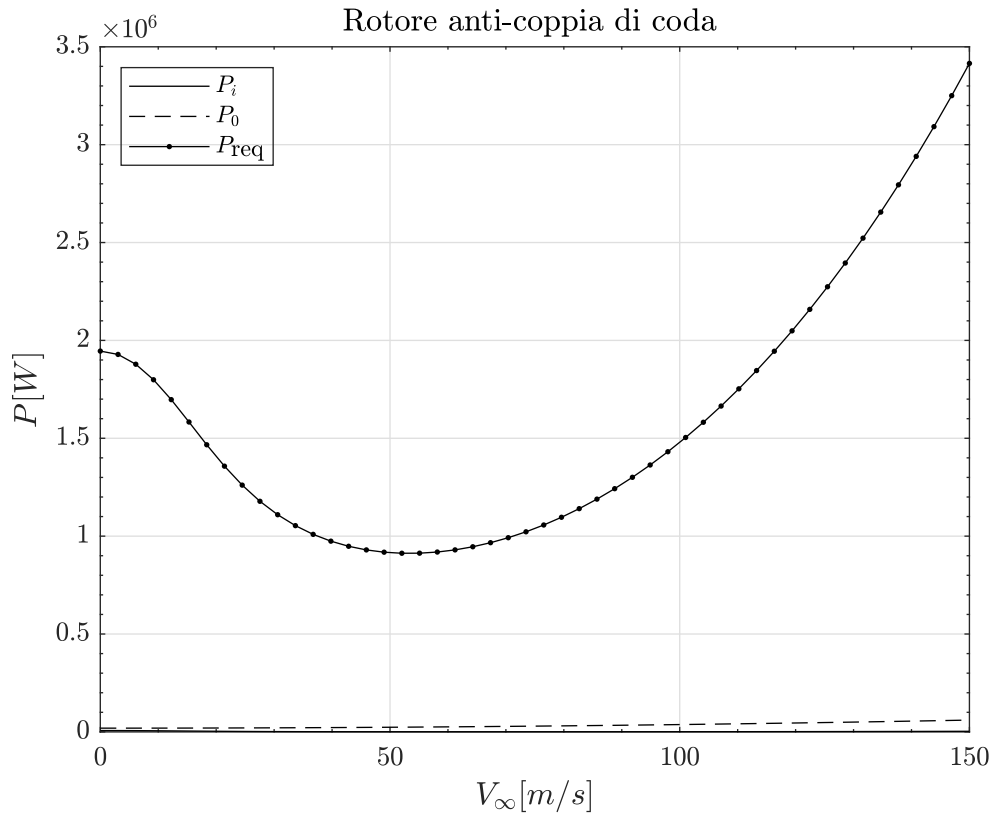


Figura 7.2: Potenza totale richiesta P_{req} dal rotore anti-coppia di coda divisa nei contributi di: Potenza indotta P_i e parassita P_0 .

2. $W_2 = 7481 kg$
3. $W_3 = 8501 kg$
4. $W_4 = 9522 kg$

Con riferimento alle curve riportate in figura 7.4, in cui si è riportato l'andamento della potenza complessiva in funzione della velocità di volo orizzontale parametrizzate rispetto al massimo peso al decollo dell'elicottero, è possibile notare che, in ogni regime di volo, un aumento di peso corrisponde con un aumento della potenza richiesta, tuttavia il maggiore scarto fra le curve è sicuramente localizzato in condizioni di *hover* o comunque per ridotte velocità di volo. Tale comportamento è ascrivibile al fatto che, per basse velocità, l'aliquota preponderante di potenza è associata alla potenza indotta, mentre il contributo associato alla velocità di traslazione tende a zero. Viceversa, per elevate velocità di traslazione, la componente preponderante di potenza deriva dal regime di funzionamento ed è indipendente dal peso, motivo per cui le curve tendono a sovrapporsi per elevati rapporti di funzionamento. I grafici come quello riportato in 7.4, che ci permettono di valutare l'evoluzione della potenza in funzione del regime di moto, risultano di particolare interesse dal momento che ci consentono di dedurre importanti informazioni circa le velocità caratteristiche. La velocità per cui si ha l'intersezione della curva di potenza richiesta con la retta della potenza disponibile rappresenta la velocità massima in volo orizzontale, V_{max} . La velocità per cui si ha un minimo della potenza viene definita V_{BE} , ovvero la velocità che ottimizza l'autonomia oraria dell'elicottero. La velocità che coincide con il minimo rapporto potenza richiesta/velocità viene battezzata V_{BR} , ovvero la velocità al quale si massimizza l'autonomia chilometrica dell'elicottero. Riguardo le velocità caratteristiche dell'elicottero e le prestazioni dello stesso si faccia riferimento alla sezione successiva per un'analisi più completa.

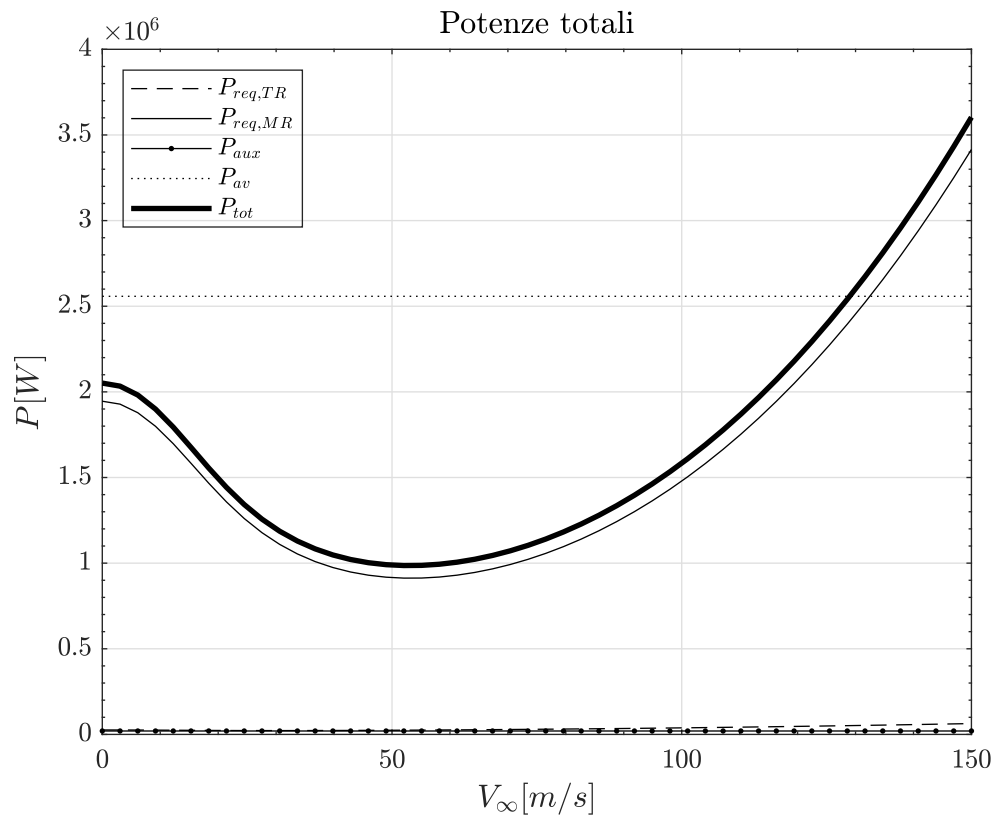


Figura 7.3: Potenza totale richiesta P_{tot} dalla macchina elicottero divisa nei contributi di: Potenza richiesta dal rotore principale $P_{req,MR}$, dal rotore anti-coppa di coda $P_{req,TR}$, potenze ausiliarie P_{AUX} P_0 . Inoltre viene riportata anche la potenza massima continua disponibile (AEO) P_{av} .

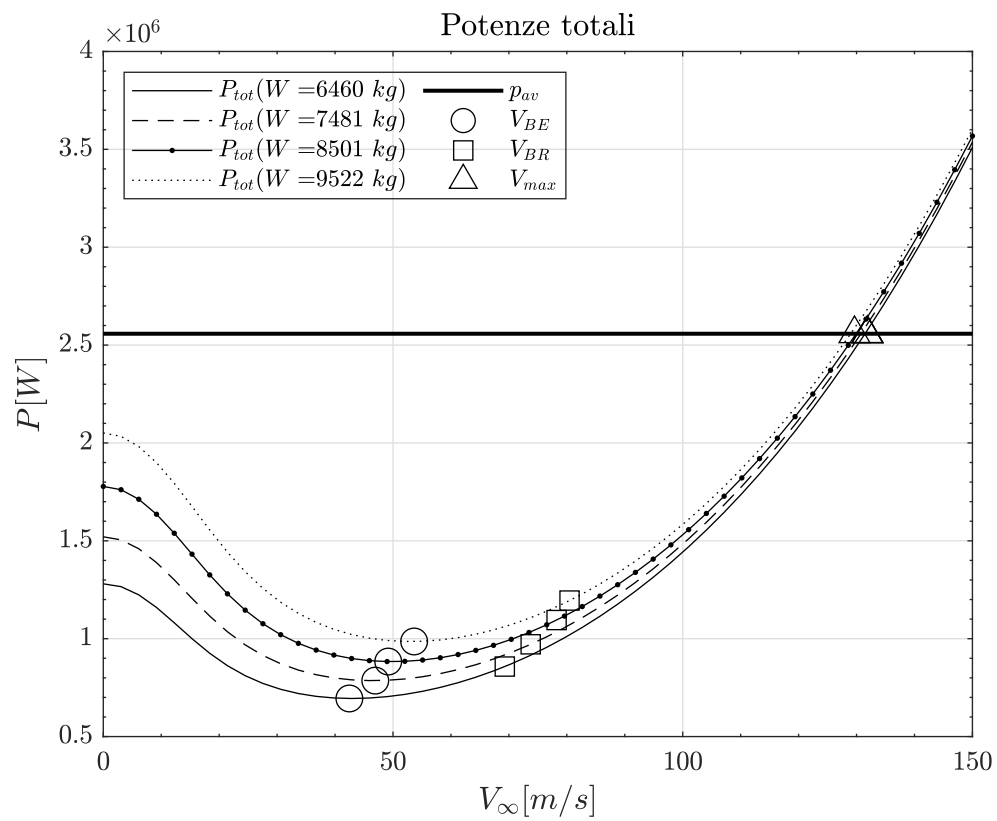


Figura 7.4: Potenza totale assorbita dalla macchina elicottero parametrizzata rispetto a 4 possibili pesi al decollo.

7.2 Le prestazioni dell'elicottero

Una volta definite le diverse aliquote di potenza assorbita è possibile valutare le prestazioni dell'elicottero in analisi, pervenendo, dunque, a parametri fondamentali come la massima velocità in volo traslato, la massima velocità di salita, l'autonomia chilometrica, l'autonomia temporale ed altri.

7.2.1 Le velocità caratteristiche

Una prima classe di prestazioni riguarda le velocità caratteristiche dell'elicottero: oltre alla velocità massima, V_{max} , sono state calcolate anche la V_{BE} e la V_{BR} , rispettivamente la velocità che massimizza l'*endurance* e la velocità che massimizza il *range*, come osservato nella precedente sezione; conoscere tali velocità per diverse condizioni di volo del velivolo consente, in fase di definizione della missione, di selezionare sempre la corretta velocità ai fini del soddisfacimento degli obiettivi di missione. Per questo motivo si è scelto di riportare la variazione di tali velocità rispetto al peso stesso del velivolo, che, come nella precedente sezione, è selezionato all'interno dell'intervallo $[W_{min}, W_{mtow}]$.

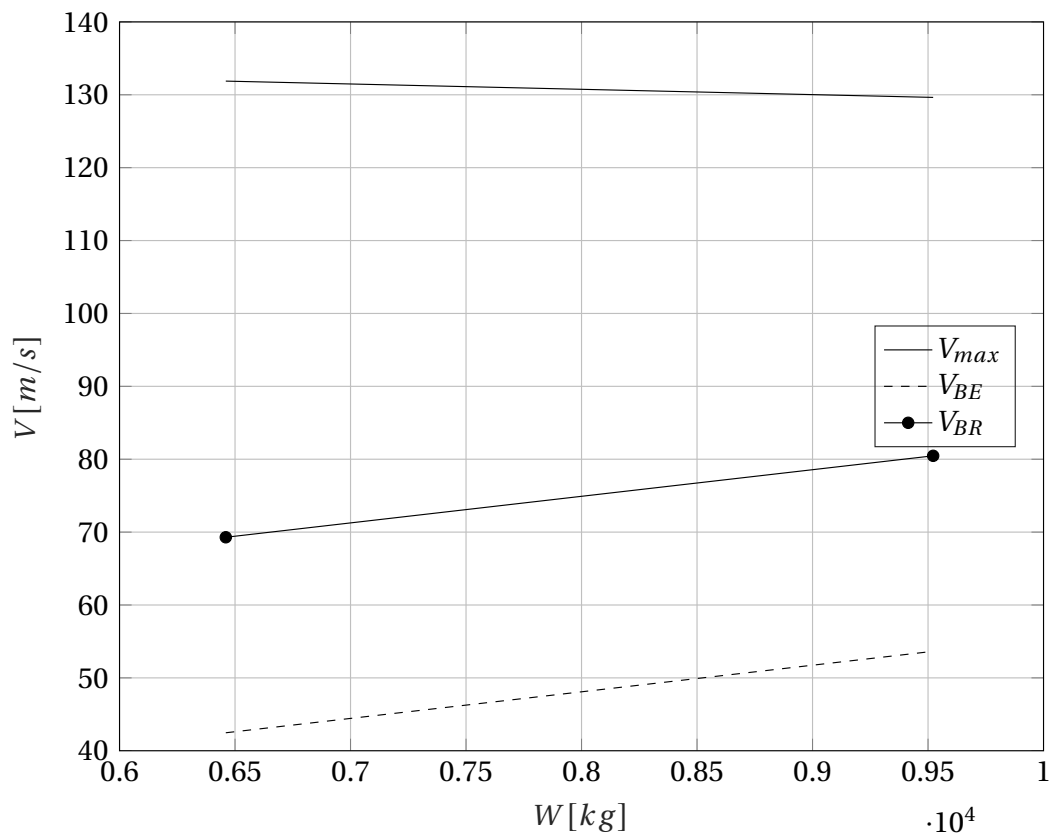


Figura 7.5: Variazione delle velocità caratteristiche (V_{max} , V_{BR} e V_{BE}) in funzione del peso per l'elicottero.

Osservando la figura 7.5, è possibile dedurre che, ad esempio, nel corso di un segmento di missione in cui è previsto un consumo rilevante del carico di combustibile a bordo è opportuno ridurre la velocità di volo qualora si sia intenzionati a rimanere nei pressi della V_{BR} o della V_{BE} dal momento che le stesse si riducono al ridursi del peso. Riguardo la velocità massima orizzontale dell'elicottero è bene specificare che, in questa sede, viene considerato esclusivamente il limite della velocità massima imposto dalla potenza installata a bordo, tuttavia, come osservato nei capitoli precedenti, in cui si è analizzata l'aerodinamica del rotore, è bene notare che la massima velocità potrebbe, in

verità, essere limitata dallo stallo del rotore più che dalle condizioni di potenza a bordo. In prima analisi si potrebbe dire che la massima velocità orizzontale che l'elicottero è in grado di mantenere è il minimo fra la velocità massima dedotta in questa sezione e quella dedotta nella sezione dell'aerodinamica del rotore. Un'altra prestazione di grande interesse per la macchina elicottero è il rateo di salita, anche noto come *ROC, rate of climb*. In primo luogo si valuti in che modo il rateo di salita si modifica in funzione del regime di volo: dalla figura 7.6 possiamo notare come all'aumentare della velocità di traslazione si riducano le prestazioni in salita dell'elicottero. Tale comportamento è chiaramente legato al fatto che all'aumentare della velocità di traslazione, la potenza richiesta cresce come mostrato nei grafici presenti nella precedente sezione (figure 7.4, 7.3), dunque la potenza residua per effettuare una salita si riduce.

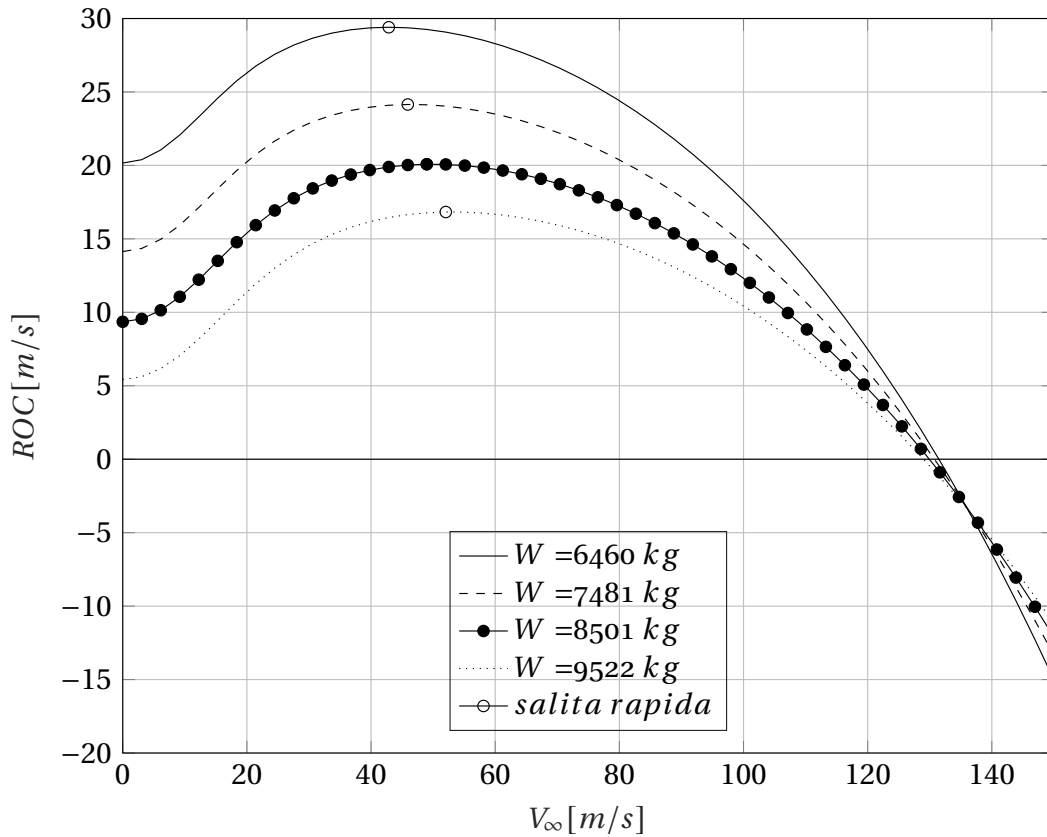


Figura 7.6: Variazione del rateo di salita in funzione della velocità di traslazione parametrizzata con il peso dell'elicottero.

Sempre con riferimento alla figura 7.6 è possibile notare che esiste una velocità in corrispondenza della quale non è più possibile effettuare volo in salita, e, oltre tale velocità, la potenza installata non consente di mantenere il volo livellato. Si è scelto di riportare sulla figura stessa anche la condizione di salita rapida, che coincide con il massimo valore del rateo di salita, indicato sulla curva tramite un pallino, come indicato in legenda.

$$ROC = \frac{P_{av} - (P_{req})_{horiz.}}{W} \quad (7.1)$$

Come atteso, all'aumentare del peso al decollo il massimo rateo di salita si riduce, tuttavia si incrementa la velocità orizzontale a cui si verifica il massimo. Si noti esplicitamente che non è stato riportato il parametro di salita ripida poiché in tali condizioni di potenza è sempre possibile, per ogni valore del peso al decollo, effettuare il volo verticale in salita, per cui, in ogni caso il massimo angolo di salita è pari a $\gamma_{max} = \frac{\pi}{2}$. Sempre

con riferimento alle prestazioni in salita dell'elicottero, è stata proposta, in figura 7.7, la degradazione delle prestazioni in salita dell'elicottero in funzione del peso a decollo. Tutte le considerazioni fatte sulla velocità in salita fanno riferimento all'equazione 7.1, tratta da [2].

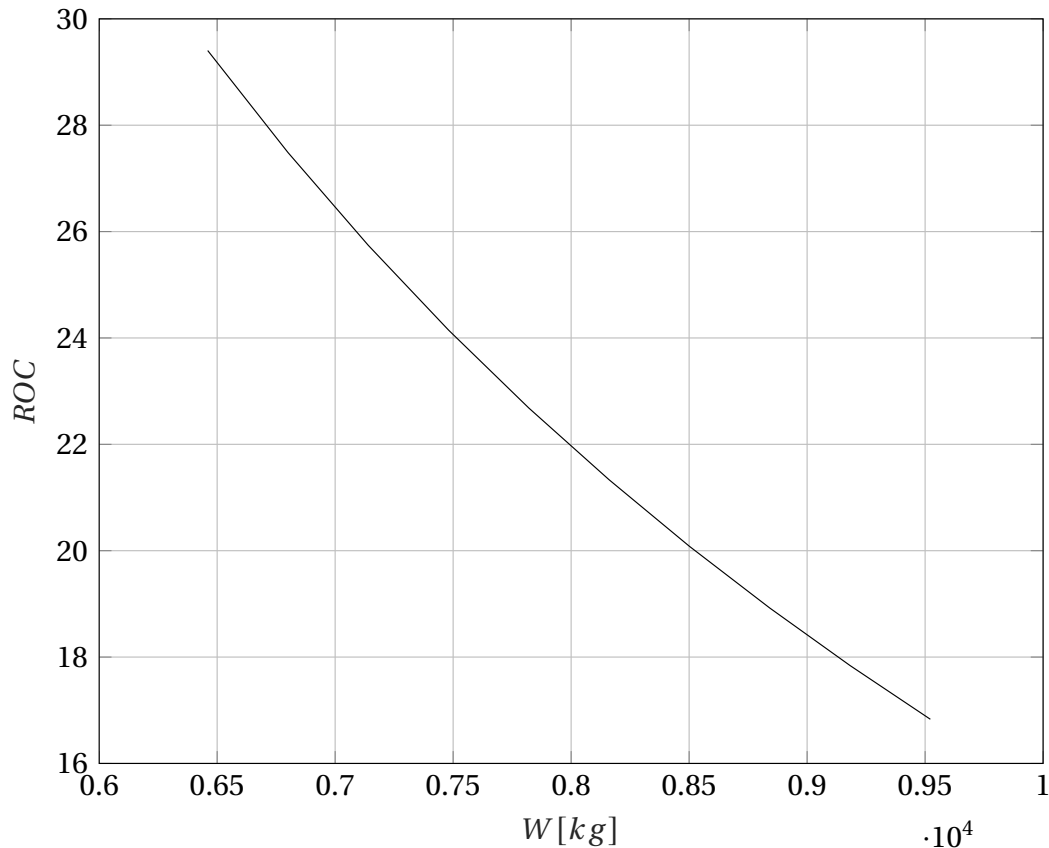


Figura 7.7: Variazione del massimo rateo di salita in funzione del peso al decollo dell'elicottero.

Altre prestazioni di rilievo per un elicottero sono le autonomie: chilometriche ed orarie, il cui andamento rispetto al peso è riportato nelle figure 7.8 e 7.9. Come atteso, un incremento del peso al decollo è associato ad una riduzione di entrambe le autonomie.

Come osservato in precedenza, quando l'elicottero dispone della piena potenza dei motori installati, per ogni condizione di peso il velivolo è sempre in grado di effettuare una salita verticale, dunque, esiste almeno una condizione di volo per cui è possibile eseguire una salita assiale, a cui compete di angolo di salita pari a 90° . In questo contesto è interessante valutare cosa accade in condizioni di piantata motore. In gergo ci si riferisce a tale condizione con l'acronimo anglosassone *OEI*, *one engine inoperative*. Dal punto di vista delle prestazioni ciò vuol dire dimezzare la potenza installata. Data la rilevanza di tale condizione di volo risulta interessante valutare l'andamento del massimo angolo di salita ottenibile per ciascuna configurazione di peso al decollo. Osservando la figura 7.10, possiamo notare che adesso il massimo angolo di salita non è più di 90° e si riduce ulteriormente all'aumentare del peso. Talvolta in condizioni di *OEI*, il sistema di controllo elettronico del motore prevede un'ulteriore riduzione della potenza erogabile al fine di preservare l'integrità strutturale del propulsore, costretto a sopperire all'assenza del secondo propulsore. Per simulare tale evenienza si è scelto di *plottare* anche un'ulteriore curva che fa riferimento a ad una riduzione pari al 25% della potenza del singolo propulsore funzionante: in questo caso esiste una condizione,

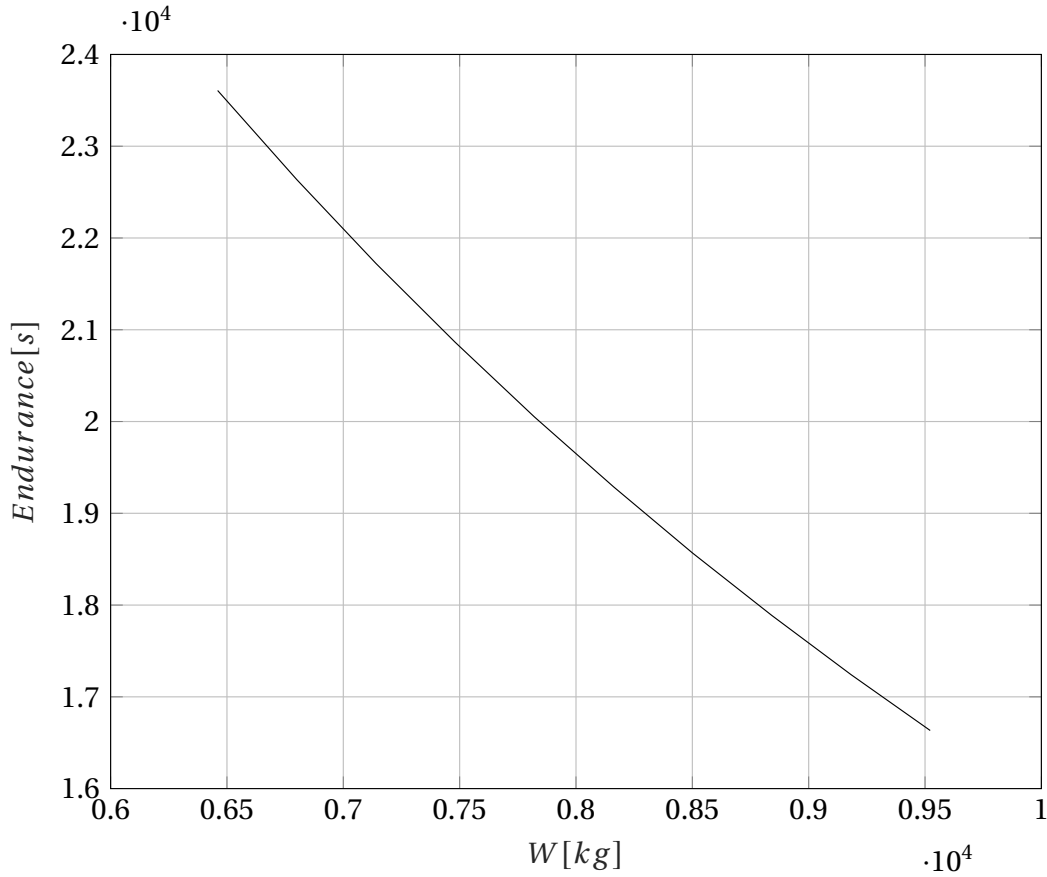


Figura 7.8: Variazione della massima autonomia oraria al variare del peso dell'elicottero.

all'interno dei regimi di funzionamento proposti, per cui non è più possibile effettuare una manovra di salita in quanto il massimo angolo di salita è negativo.

Si intende valutare, infine, le prestazioni in autorotazione. Il regime di autorotazione è una condizione di volo dimensionante in quanto la possibilità di eseguire in sicurezza una manovra di autorotazione è una caratteristica fondamentale anche ai fini della certificazione della macchina stessa. In questo contesto si valuterà la velocità in autorotazione come in [2], ovvero tramite la relazione 7.2. Nella definizione di cui prima si è caratterizzato solo il modulo, dal momento che il vettore, evidentemente, è orientato verso il basso.

$$|V_{autorot.}| = \frac{(P_{req})_{horiz.}}{W} \quad (7.2)$$

L'andamento della velocità di autorotazione appare debolmente dipendente dal peso, possibile notare in 7.11, con una variazione di pochi punti percentuali per tutto il range di pesi proposti.

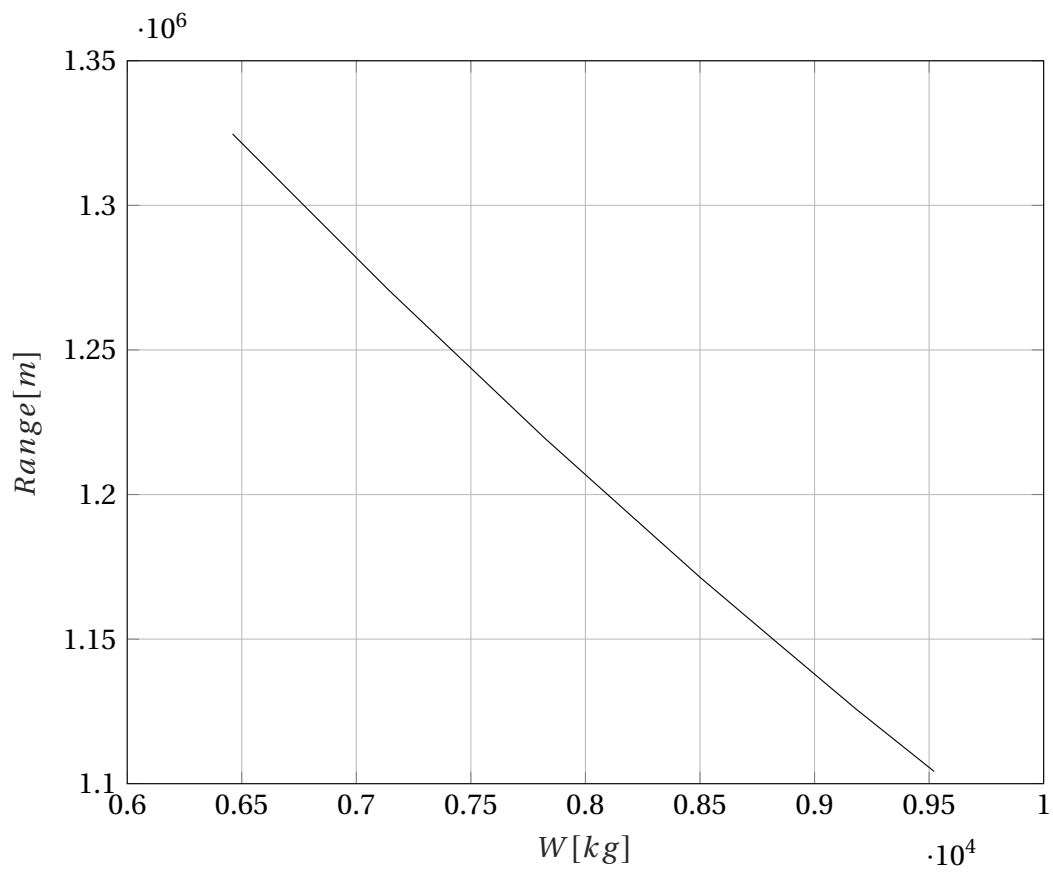


Figura 7.9: Variazione della massima autonomia chilometrica al variare del peso dell'elicottero.

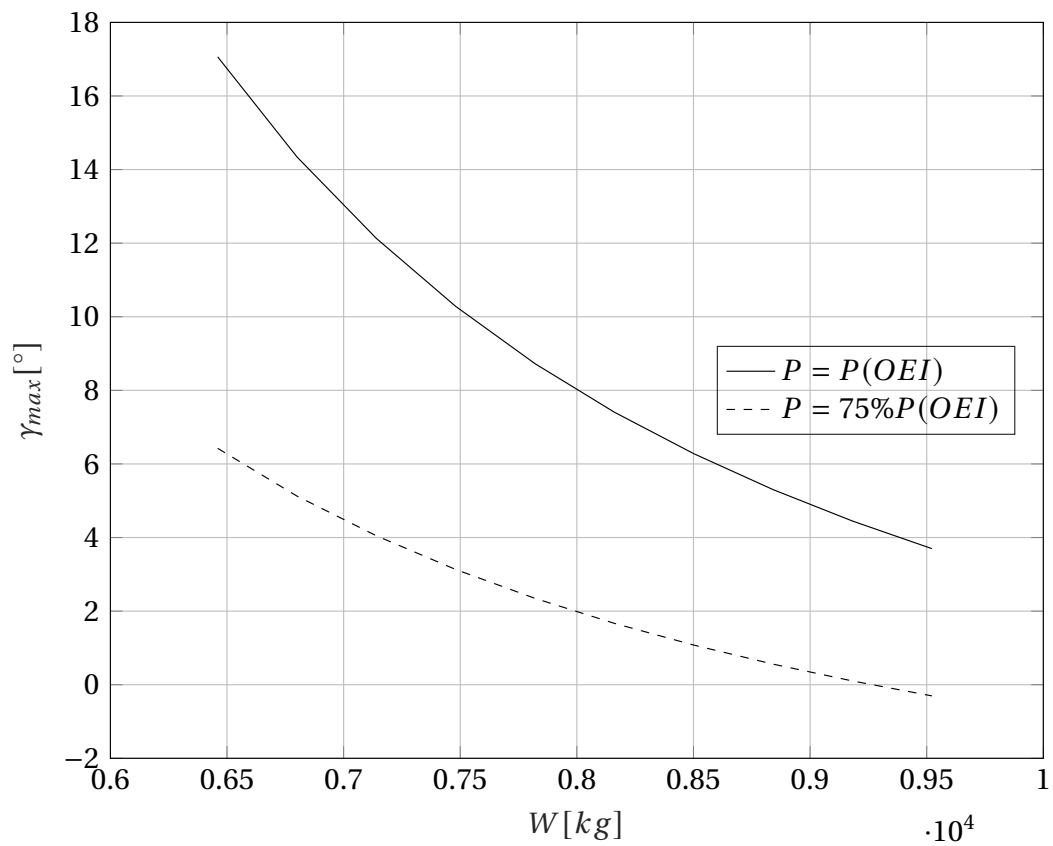


Figura 7.10: Variazione del massimo angolo di salita con il peso per due particolari configurazioni di potenza disponibile: $P = P(OEI)$ e $P = 75\%P(OEI)$.

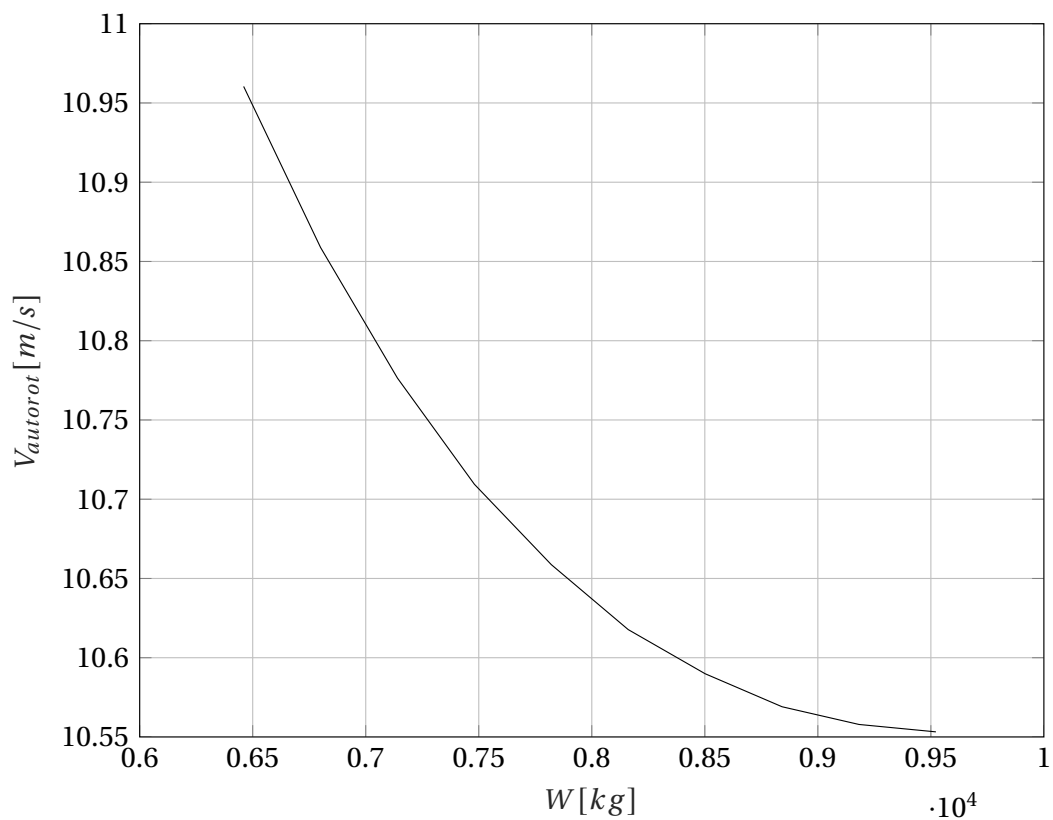


Figura 7.11: Variazione della velocità di autorotazione con il peso.

Parte III

Gli aeromotori

8.1 Introduzione & stime iniziali

In questa parte dell'elaborato si è scelto di riportare il progetto di una turbina eseguito attraverso il codice *open source* Q-blade. L'iter progettuale per la turbina parte dalla definizione della potenza erogata dalla stessa, come riportato in tabella 8.1, si è scelto di valutare la progettazione di una turbina da 1 MW. Per una stima di prima analisi delle dimensioni richieste si è ipotizzato un valore del coefficiente di potenza pari a 0.5. Con riferimento alla relazione (8.1) è possibile notare che è richiesto un raggio della pala di circa 55 metri. Tutti i parametri di riferimento riportati in relazione (8.1) sono riportati in tabella 8.1.

$$R = \sqrt{\frac{P}{0.5\rho V_{\infty}^3 C_p \pi}} = 55m \quad (8.1)$$

Densità dell'aria, ρ	1.225 kg/m^3
Velocità della corrente asintotica, V_{∞}	7 m/s
Coefficiente di potenza, C_p	0.5
Potenza generata, P	$1 \times 10^6 \text{ W}$

Tabella 8.1: Parametri necessari alla stima degli ingombri di un aeromotore ad asse orizzontale.

I parametri riportati in tabella 8.1, oltre al valore del raggio, R , sono tutti dati di *input* necessari per l'implementazione della procedura di *design*.

Infine, si è scelto un valore di progetto per il rapporto di funzionamento λ pari a 6 (turbina veloce), di conseguenza possiamo ricavare la velocità di rotazione delle pale come:

$$\Omega = \frac{\lambda_{opt} V_{\infty}}{R} = 0.7636 \quad \frac{rad}{s} \approx 7.3 \quad \frac{giri}{minuto} \quad (8.2)$$

8.2 L'aerodinamica

Un elemento fondamentale della progettazione aerodinamica di un aeromotore risiede certamente nella selezione del corretto profilo alare di cui dotare le pale. In questo contesto si è scelto il profilo alare DU84-132V3, appartenente alla classe di profili alari progettati, dalla *Delft University of Technology (DUT)*, appositamente per le turbine alari con l'obiettivo di ridurre gli effetti della contaminazione sulle prestazioni e sull'efficienza del profilo.

Esso è rappresentato in figura 8.1. In figura 8.2 sono riportate le caratteristiche di portanza e di resistenza nell'intervallo di $\alpha \in [-180^\circ; 180^\circ]$. Tali curve sono state ottenute mediante l'utilizzo del *software Xfoil* ed in seguito sono state estese in post-stallo utilizzando un *tool* integrato all'interno del *kernel* del *software* Q-blade e che si basa sul metodo di Beddoes.

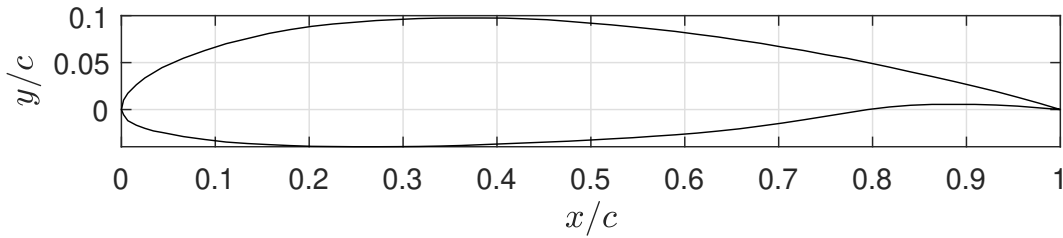


Figura 8.1: Rappresentazione del profilo alare DU84-132V3.

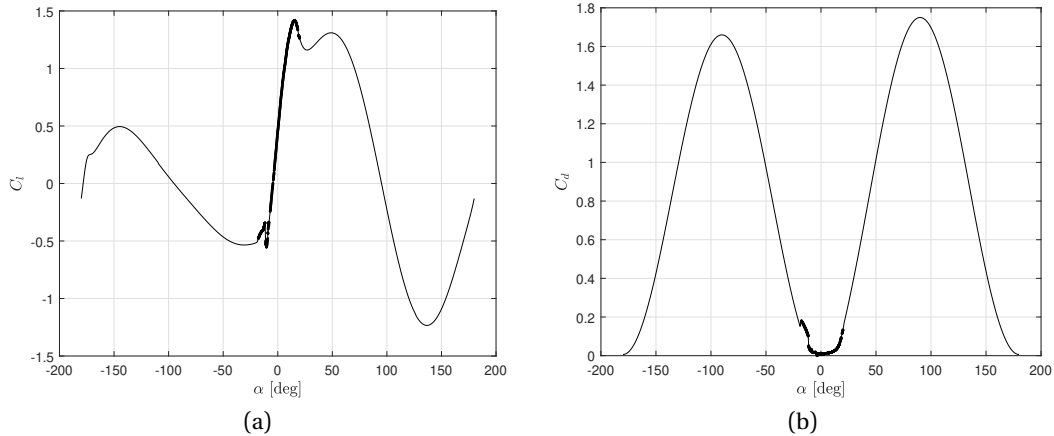


Figura 8.2: Curva di portanza (a) e curva $C_d - \alpha$ del profilo DU84-132V3. $Re_\infty = 1 \times 10^6$, $M_\infty = 0$. Ottenuti mediante l'utilizzo del *software Xfoil*, e prolungate in post-stallo mediante l'utilizzo del *software* Q-blade.

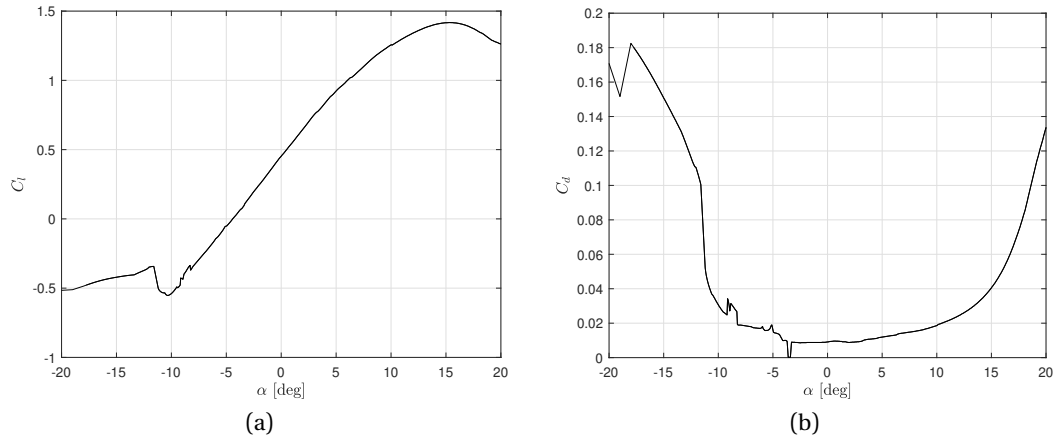


Figura 8.3: Rappresentazione della curva di portanza per valori dell'angolo di attacco compresi nell'intervallo: $\alpha \in [-20^\circ, 20^\circ]$, del profilo alare DU84-132V3. $Re_\infty = 1 \times 10^6$, $M_\infty = 0$. Ottenuti mediante l'utilizzo del software Xfoil.

8.3 La geometria

In questa sezione vengono riportati i risultati della procedura di *design* in termini di geometria della pala. In figura 8.5 vengono riportate alcune viste della pala.

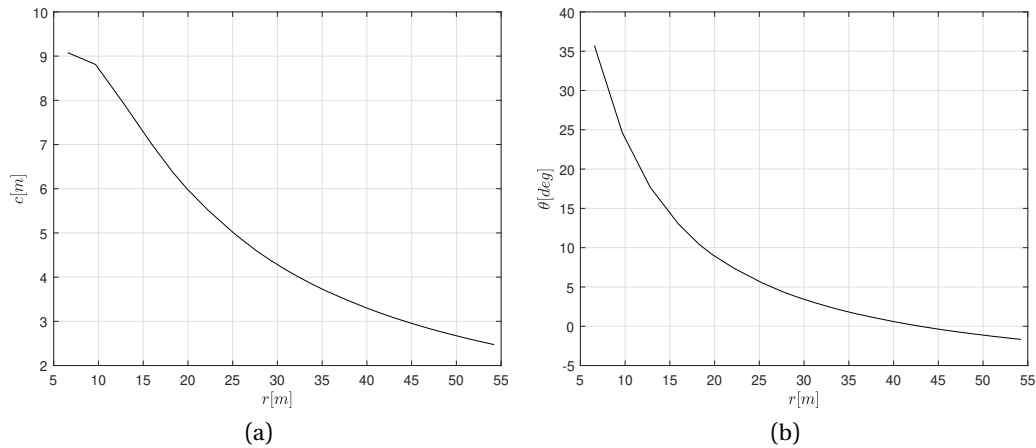
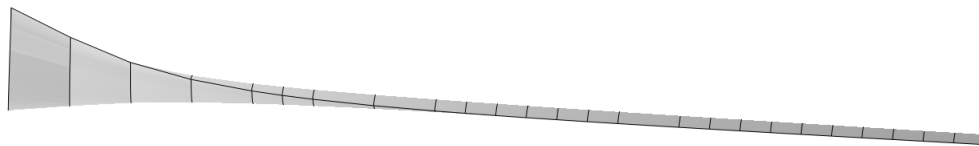
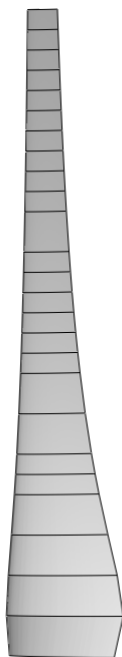


Figura 8.4: Distribuzione della corda (a) e del calettamento (b) in funzione del raggio.

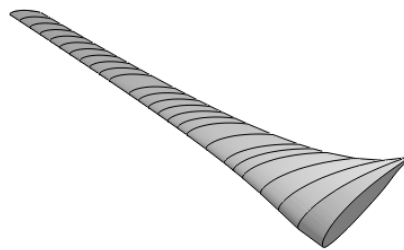
Oltre alle immagini riportate in 8.5, si è deciso di riportare anche l'andamento del calettamento e della corda in funzione della coordinata radiale. Come è possibile notare dalla figura 8.4, l'andamento del calettamento ottenuto per la turbina in analisi è molto simile rispetto a quello ricavato per la turbina ottima a pag. 128 in figura 97 del [1].



(a)



(b)



(c)

Figura 8.5: Vista laterale (a), vista dall'alto (b), vista in assonometria (c) della pala. Immagini realizzate mediante il *software* Qblade.

8.4 Le prestazioni

In questa sezione vengono riportate le principali prestazioni dell'elica, ovvero: $C_P = C_P(\lambda)$ e $C_T = C_T(\lambda)$. Tali andamenti risultano di particolare interesse per la turbina dal momento che il primo andamento restituisce in effetti le prestazioni in termini di potenza generata (una volta dimensionalizzato il coefficiente di potenza) e l'andamento della forza resistente (una volta dimensionalizzato il coefficiente di resistenza), necessaria per il dimensionamento strutturale della turbina stessa. Osservando la figura 8.6, che fa riferimento al coefficiente di potenza, è possibile notare che il valore $C_P = 0.5$, ipotizzato in prima analisi e riportato in tabella 8.1, in realtà, non viene mai realizzato e, come conseguenza, la turbina non potrà mai raggiungere la potenza obbiettivo di 1 MW. Ulteriori modifiche per il raggiungimento del valore di 1MW possono essere introdotte: come l'incremento del raggio della pala o l'introduzione di *tip vanes*.

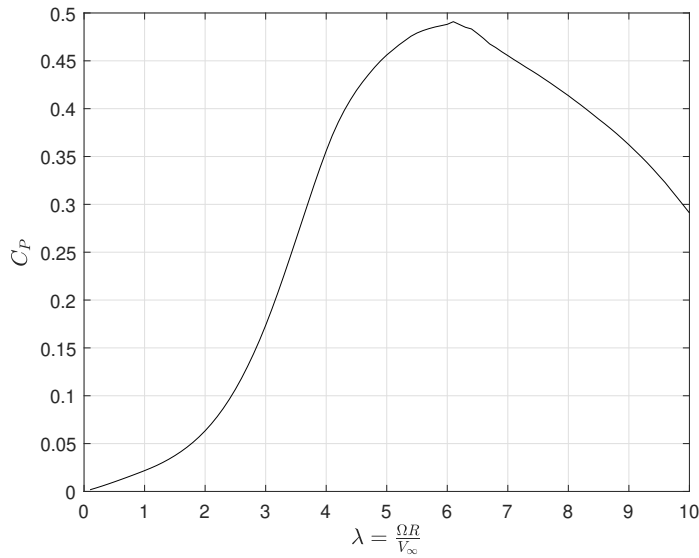


Figura 8.6: Andamento del coefficiente di potenza, C_P , in funzione del rapporto di funzionamento, λ . Risultati ottenuti mediante il *software* Qblade.

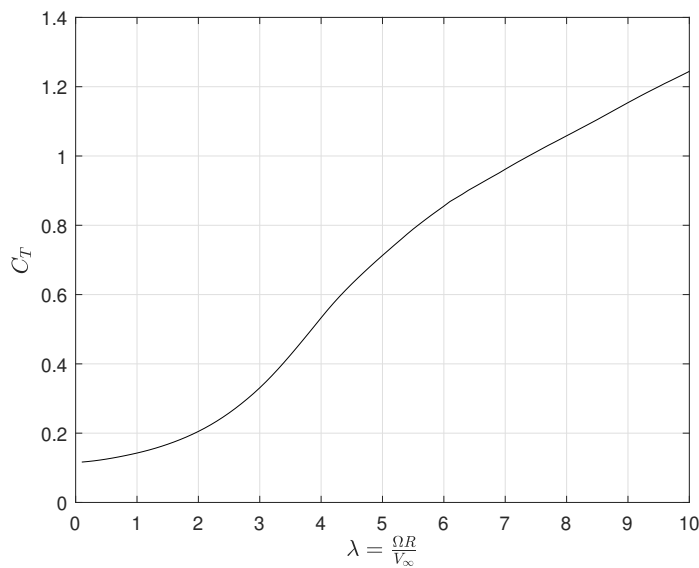


Figura 8.7: Andamento del coefficiente di resistenza della turbina, C_T , in funzione del rapporto di funzionamento, λ . Risultati ottenuti mediante il *software* Qblade.

8.5 Una procedura di *design* alternativa

In questa sezione viene proposto un metodo alternativo per il progetto della turbina eolica: è stata sviluppata una procedura iterativa basata sulle relazioni riportate alle pagine 127/128 di [1], che consente di ricavare le distribuzioni di calettamento e corda ottime lungo la pala. Anche in questo caso è stato selezionato il profilo alare DU84-132V3, per il quale sono valide le considerazioni fatte nella sezione dedicata all'aerodinamica del profilo (vedi paragrafo 8.2). Questa procedura di *design* non tiene conto degli effetti della viscosità. Contestualmente viene proposto un confronto fra le prestazioni delle turbine ottenute con le due procedure di *design*, che sfrutta lo strumento di analisi delle prestazioni di una turbina presente all'interno del *software* Q-blade. In prima analisi si valutino, in figura 8.9, le distribuzioni di calettamento e corda ottenute con questa *routine* di *design* sviluppata *ad hoc* confrontate con le distribuzioni di corda e calettamento ottenute con la procedura di *design* integrata all'interno del *software* Q-blade. In figura 8.8, invece, sono riportate alcune viste della pala isolata. Per confrontare le due procedure ci si riferisca alla figura 8.9: una differenza significativa fra le due geometrie è concentrata in corrispondenza delle sezioni in prossimità del mozzo, le quali sono dotate di una corda sensibilmente maggiore nel caso della procedura indicata nel manuale [1].

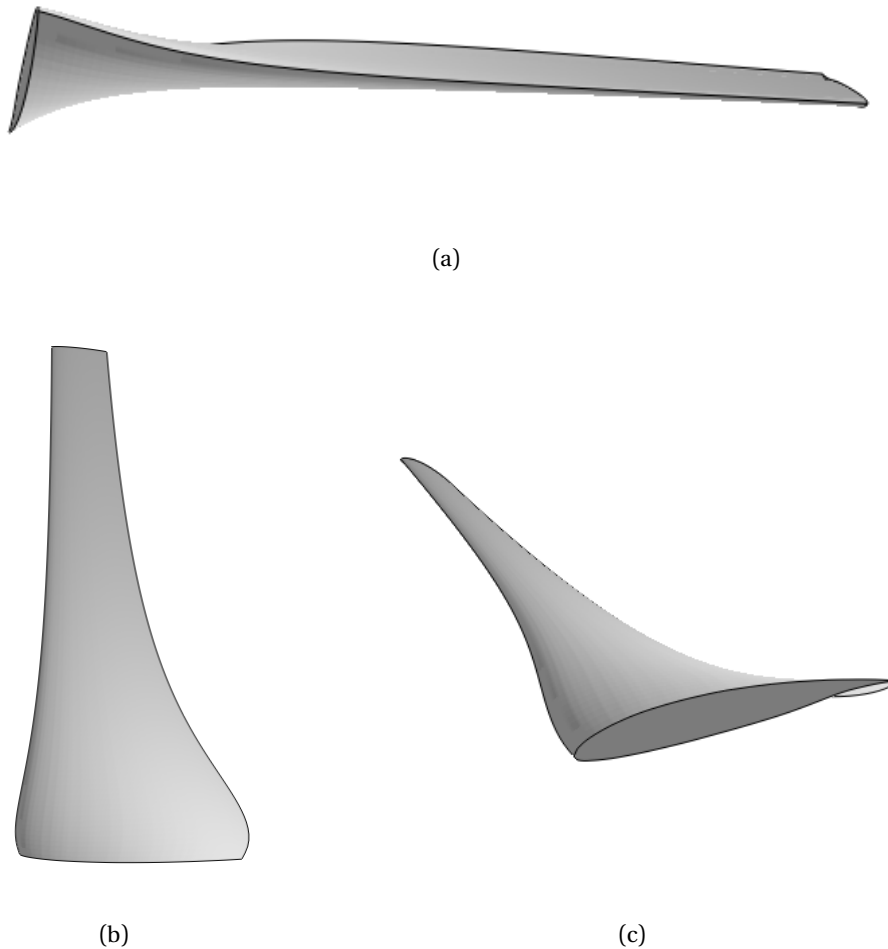


Figura 8.8: Vista laterale (a), vista dall'alto (b), vista in assonometria (c) della pala. Immagini realizzate mediante il *software* Qblade.

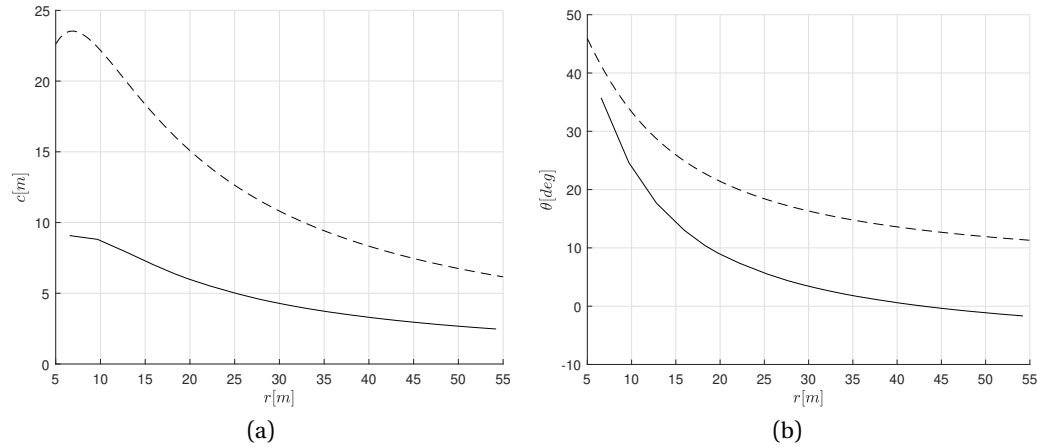


Figura 8.9: Distribuzione della corda (a) e del calettamento (b) in funzione del raggio.

8.5.1 Il confronto delle prestazioni

Una volta valutate le differenze sostanziali nella geometria, viene proposta un confronto delle prestazioni: le figure di riferimento sono le 8.10 e 8.11, nelle quali sono riportati, rispettivamente, il coefficiente di potenza e il coefficiente di resistenza al variare del rapporto di funzionamento. Dall'analisi delle figure di cui sopra è immediato verificare che la procedura di *Design* implementata in questa sezione appare meno accurata: il massimo coefficiente di potenza raggiunto è minore rispetto al caso della procedura di *design* implementato attraverso il *software* Q-blade. Tale differenza è ascrivibile, come detto precedentemente, al fatto che nella procedura di progetto proposta in [1] non si è tenuto in conto degli effetti viscosi. Sempre con riferimento alla figura 8.10, possiamo notare che, oltre alla menzionata riduzione del massimo coefficiente di potenza ottenibile, insiste anche uno *shift* del massimo, che si ottiene per valori del funzionamento minori. Tale comportamento è sempre imputabile al fatto che questa procedura di progettazione implementata non tiene conto degli effetti viscosi, sempre più significativi al crescere di λ . Si osserva che la condizione di massimo coefficiente di potenza, quindi, non corrisponde con il valore del rapporto di funzionamento di progetto impostato nella procedura di *design*.

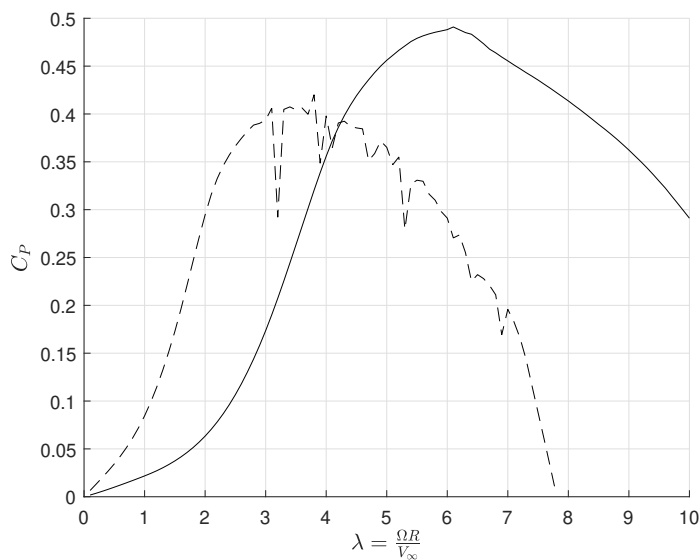


Figura 8.10: Andamento del coefficiente di potenza, C_p , in funzione del rapporto di funzionamento, λ . Confronto tra le prestazioni della turbina progettata con il *software* Q-blade (curva continua) e quelle della turbina progettata con il metodo proposto in [1] (curva tratteggiata).

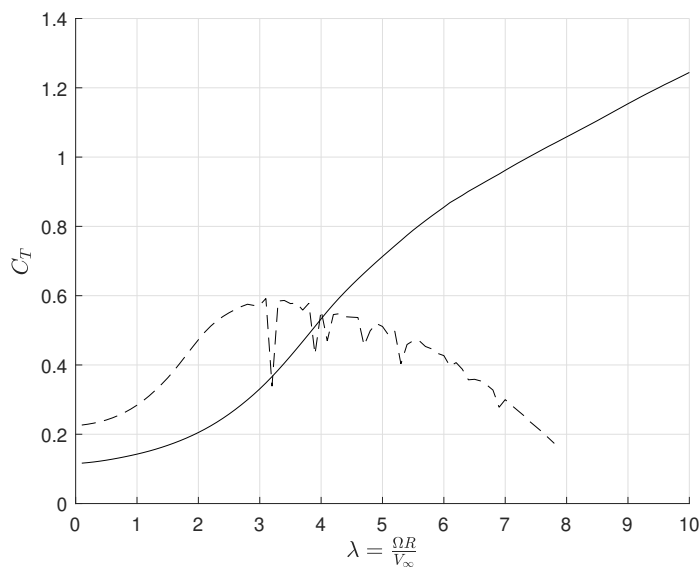


Figura 8.11: Andamento del coefficiente di resistenza della turbina, C_T , in funzione del rapporto di funzionamento, λ . Confronto tra le prestazioni della turbina progettata con il *software* Q-blade (curva continua) e quelle della turbina progettata con il metodo proposto in [1] (curva tratteggiata).

APPENDICE A

ALCUNI CODICI

In queste appendici vengono presentati alcuni codici necessari per condurre le analisi e per generare le immagini presenti all'interno di questo documento.

A.1 L'elica

Per condurre le analisi e generare le immagini in questo documento si è scelto un approccio che si basa sulla definizione di una classe dedicata all'elica, caratterizzata da metodi e proprietà specifiche per definire le caratteristiche geometriche, le condizioni ambientali della prova, le procedure numeriche per l'analisi delle prestazioni. L'utilizzo di una classe consente di creare delle simulazioni più robuste, per le quali è sempre garantita la compatibilità delle funzioni con le geometrie a cui vengono applicate. Tale approccio gode anche di elevata modularità: qualora si voglia implementare una qualsiasi altra funzione basterà aggiungere un altro metodo alla classe che risolve il problema richiesto, avendo già disponibili tutte le caratteristiche geometriche e/o ambientali. Inoltre, si è deciso di creare una proprietà della classe all'interno della quale salvare le diverse analisi eseguite sulla stessa geometria, al fine di consentire un rapido confronto al variare, ad esempio, delle condizioni ambientali o delle condizioni di peso dell'aeromobile.

Listing A.1:

```
1 classdef Elica
2     properties
3         %
4
5         % Geometria
6         %
7         N      {mustBeInteger, mustBeFinite}      % Numero di pale
8         D      {mustBePositive, mustBeFinite}      % Diametro elica, [m]
9         A_D    {mustBePositive, mustBeFinite}      % Area disco elica, [m2]
10        R      {mustBePositive, mustBeFinite}      % Raggio dell'elica, [m]
```

```

theta (:,1){mustBeReal, mustBeFinite}%Angoli di calettamento, [
    rad]
11    c    (:,1){mustBeNonnegative, mustBeFinite}% Corda delle
        sezioni, [m]
    sigma (:,1){mustBeNonnegative, mustBeFinite}      % Solidita'
13    LAMBDA (:,1){mustBeReal, mustBeFinite}      % Angolo di Freccia
        [rad]
    %
    %
15    % Funzionamento
    %
    %
17    RPM    {mustBePositive, mustBeFinite}      % Giri al minuto
    n        {mustBePositive, mustBeFinite}      % Giri al secondo
19    omega {mustBePositive, mustBeFinite}%Velocita' di rotazione, [
        rad/s]
    h {mustBeNonnegative, mustBeFinite} = 0 % Quota di
        funzionamento [m]
21    %
    %
23    % Aerodinamica
    %
    %
25    Cl= @(alpha,r_bar,M,Re) 2*pi*alpha;
    Cd= @(alpha,r_bar,M,Re) 0.01*alpha./alpha;
27    end
    properties(SetAccess = private,GetAccess=public)
29        % Vettore delle stazioni radiali
        r_bar (:,1) {mustBeInRange(r_bar,0,1,'exclude-lower')} %
            exclusive
31        n_r
        %
33        n_analisi = 0
        % ambient conditons
35        rho        {mustBePositive, mustBeFinite}
        press        {mustBePositive, mustBeFinite}
37        sound_vel {mustBePositive, mustBeFinite}
        temp        {mustBePositive, mustBeFinite}
39        mu_visc    {mustBePositive, mustBeFinite}
        % funzionamento
41        OmR
        %
43        % Analisi
        %

```

```

45     Analisi
end
47 properties(SetAccess=private,GetAccess=private)
    Des
49 end
%
51 methods
    % imposta il dominio radiale
53     function obj = r_(obj,vec_r)
        obj.r_bar=vec_r;
55         obj.n_r=length(obj.r_bar);
    end
57     % Compute some mass and geometric property. This function needs
        to
        % be called before doing any other calculations but still after
        the
59     % definition of the main properties(N,c,r_bar,R,omega). The
        function needs to be
        % called only once. The function computes any derived property.
61     function obj = derived_properties(obj)
        obj.D      = obj.R*2;
63         obj.A_D   = pi*obj.R*obj.R;
        if ~isempty(obj.c)
65             obj.sigma = obj.N/(2*pi)*obj.c.*obj.r_bar.^-1*obj.R;
        end
67     end
    % calcolo delle velocita' di rotazione
69     function obj = rot_vel(obj,valIN,val)
        switch valIN
71         case 'RPM'
            obj.RPM    = val;
73             obj.n      = obj.RPM/60;
            obj.omega  = obj.n*2*pi;
75         case 'n'
            obj.n       = val;
77             obj.RPM    = obj.n*60;
            obj.omega  = obj.n*2*pi;
79         case 'omega'
            obj.omega   = val;
81             obj.n      = obj.omega/(2*pi);
            obj.RPM     = obj.n*60;
83         otherwise
            mustBeMember(valIN,{'RPM','n','omega'})
85         end
        obj.OmR      = obj.omega*obj.r_bar*obj.R;
87     end
    % assegna le prestazioni aerodinamiche di profilo in funzione
        di
89     % dati tabulari... Da modificare
    function obj = set_aero(obj,valpha,vCl,vCd)
91         % v_alpha    vettore degli angoli di attacco in rad

```

```

93     % vCl      vettore dei Cl (stessa dimensione di v_alpha)
94     % vCd      vettore dei Cd (stessa dimensione di v_alpha)
95     obj.Cl=@(alpha) interp1(valpha,vCl,alpha);
96     obj.Cd=@(alpha) interp1(valpha,vCd,alpha);
97 end
98 %
99
100 % Ambient conditions
101 %
102
103 function obj = altitude(obj,h)
104     obj.h=h;
105     [obj.temp, obj.sound_vel, obj.press, obj.rho] = atmosisa(
106         obj.h);
107     T0 = 288.15;
108     mu0 = 1.79e-5;
109     obj.mu_visc=mu0*(obj.temp/T0)^1.5*((T0+110)/(obj.temp+110))
110     ;
111 end
112 %
113
114 %% Teoria dell'elemento di pala generale
115 %
116
117 function [f,J_,alpha,phi,a,ap,lambda1,lambda2,M,Re]=func(obj,
118     alpha,J,idx,options)
119 % Funzione da annullare per il calcolo delle prestazioni dell'
120 % elica
121 % Input:
122 % — alpha,      valore di alpha di tentativo;
123 % — J,          Rapporto di funzionamento in
124 %               corrispondenza
125 %               del quale si vogliono calcolare le
126 %               prestazioni
127 % — idx,        Indice della stazione radiale
128 % — options     Analsys options (see BEMTset)
129 %
130
131 V_inf = J *obj.n*obj.D;
132 V_eff = sqrt(V_inf^2 + (obj.omega*obj.r_bar(idx)*obj.R)^2)*
133     cos(obj.LAMBDA(idx)); % trascurato a e a' rispetto ad 1
134 M = V_eff/obj.sound_vel;
135 Re = V_eff * obj.rho * obj.c(idx)/ obj.mu_visc;
136 % calcolo dell'angolo di Inflow
137 phi=obj.theta(idx)-alpha;
138 % calcolo dei coefficienti aerodinamici

```



```

127     Cl = obj.Cl(alpha,obj.r_bar(idx),M,Re);
128     Cd = obj.Cd(alpha,obj.r_bar(idx),M,Re);
129
130     lambda1 = Cl*cos(phi) - Cd*sin(phi);
131     lambda2 = Cl*sin(phi) + Cd*cos(phi);
132     % calcolo delle induzioni
133     % ka = 0.25*obj.sigma(idx)*lambda1/(sin(phi)^2);
134     ka = 0.25*obj.sigma(idx)*lambda1/(1-cos(phi)^2);
135     kap = 0.5*obj.sigma(idx)*lambda2/sin(2*phi);
136
137     ap = kap/(1+kap);
138     a = ka/(1-ka);
139
140     % calcolo del rapporto di avanzamento
141     J_ = pi*obj.r_bar(idx)*(1-ap)*tan(phi)/(1+a);
142     % funzione da annullare
143     f = J - J_;
144 end
145 function [f,J_,alpha,phi,a,ap,lambda1,lambda2,M,Re]=BEMT_rJ_fix
    (obj,...
                                     alpha0,alpha1,J,idx,
                                     options)
146
147 % Calcola le prestazioni dell'elica per una sola stazione
    radiale
148 % e per un solo valore del rapporto di funzionamento con il
    metodo
149 % delle Secanti.
150 % Input:
151 % - alpha0 e alpha1      Valori di tentativo iniziali dell'
    angolo
152 %                        d'attacco;
153 % - J,                   Rapporto di funzionamento in
    corrispondenza del quale si
154 % vogliono
155 %                        calcolare le prestazioni;
156 % - idx,                 Indice della stazione radiale
157 % - options,             Analisis options (see BEMTset)
158 %
159
160     it = 0;
161     f0=obj.func(alpha0,J,idx,options);
162     if abs(f0)>options.toll
163         f1=obj.func(alpha1,J,idx,options);
164         while abs(f1)>options.toll
165             it = it+1;
166             qk=(f1-f0)/(alpha1-alpha0);
167             alpha0=alpha1; f0=f1;
168             alpha1=alpha1-f1/qk;
169             f1=obj.func(alpha1,J,idx,options);
170         end
171     else

```

```

171         alpha1=alpha0;
172     end
173     [f,J_,alpha,phi,a,ap,lambda1,lambda2,M,Re]=obj.func(alpha1,
        J,idx,options);

175 end
176 function obj=BEMT(obj,J,alpha0,options)
177 % Calcola le prestazioni dell'elica per tutte le stazioni
    radiali
178 % e per una serie di valori del rapporto di funzionamento, con
    il
179 % metodo delle Secanti.
    % Input:
181 % — alpha0 e alpha1      Valori di tentativo iniziali dell'
        angolo
    %
        d'attacco; [rad] (default: 0 e 10
        deg)
183 % — J,                    Valori del Rapporto di funzionamento
        in
    %
        corrispondenza dei quali si
        vogliono
185 %
        calcolare le prestazioni;
    % — options              Analisis options (see BEMTset)
187 %

    arguments
189     obj
        J      (:,1){mustBeNonnegative,mustBeFinite}
191     alpha0 (1,1){mustBeReal,mustBeFinite}=0
        options = BEMTset();
193 end
    %

195 alpha = alpha0;
    for jdx=1:length(J)
197         if jdx>1
            alpha = s.alpha(jdx-1,1);
199         end
        for idx =1:obj.n_r
201             alpha1 = 1.3*alpha;
            [~,~,alpha,phi,a,ap,lambda1,lambda2,M,Re]=
                BEMT_rJ_fix(obj,...
203                             alpha,alpha1,J(jdx),idx,
                                options);

            s.alpha(jdx,idx) = alpha;
205             s.phi(jdx,idx) = phi;
            s.a(jdx,idx) = a;
207             s.ap(jdx,idx) = ap;
            s.lambda1(jdx,idx) = lambda1;
209             s.lambda2(jdx,idx) = lambda2;

```

```

211         s.Mach(jdx,idx)    = M;
        s.Re(jdx,idx)      = Re;

213         % calcolo delle prestazioni
        s.dCt_dr_bar(jdx,idx) = pi/4*lambda1*obj.sigma(idx)
            *obj.r_bar(idx)*...
215         ((J(jdx)*(1+a))^2 + (pi*obj.r_bar(idx)*(1-ap))
            ^2);
        s.dCq_dr_bar(jdx,idx) = pi/8*lambda2*obj.sigma(idx)
            *obj.r_bar(idx)^2*...
217         ((J(jdx)*(1+a))^2 + (pi*obj.r_bar(idx)*(1-ap))
            ^2);
        %
219         % s.dCt_dr_bar(jdx,idx) = pi^3/4*obj.sigma(idx)*...
            lambda1*obj.r_bar(idx)^3*(1-ap)^2/(cos(phi)
            ^2);
        %
221         % s.dCq_dr_bar(jdx,idx) = pi^3/8*obj.sigma(idx)*...
            lambda2*obj.r_bar(idx)^4*(1-ap)^2/(cos(phi)
            ^2);

        s.dCp_dr_bar(jdx,idx) = 2*pi*s.dCq_dr_bar(jdx,idx);
223         s.eta_e(jdx,idx)    = J(jdx)/(pi*obj.r_bar(idx))*
            lambda1/lambda2;
        %
225         % s.eta_e(jdx,idx)    = (1-ap)/(1+a)*lambda1/
            lambda2*tan(phi);
        end
        if isequal(options.P_correction,'on')
227             F                = obj.F_(J(jdx)/pi);
            s.dCt_dr_bar(jdx,:) = s.dCt_dr_bar(jdx,:)'*F;
229             s.dCq_dr_bar(jdx,:) = s.dCq_dr_bar(jdx,:)'*F;
            s.dCp_dr_bar(jdx,:) = s.dCp_dr_bar(jdx,:)'*F;
231         end
        s.CT(jdx,1)=obj.simpsons(s.dCt_dr_bar(jdx,:),obj.r_bar
            (1),obj.r_bar(end));
233         if isequal(options.Hub_correction,'on')
            s.DCT(jdx,1) = -pi/8*(obj.r_bar(1))^2*...
235             J(jdx)^2*options.Cd_hub;
            s.CT(jdx,1) = s.CT(jdx,1) + s.DCT(jdx,1);
237         end
        s.CQ(jdx,1)=obj.simpsons(s.dCq_dr_bar(jdx,:),obj.r_bar
            (1),obj.r_bar(end));
239         s.CP(jdx,1)=2*pi*s.CQ(jdx,1);
            s.eta(jdx,1)=s.CT(jdx,1)/s.CP(jdx,1)*J(jdx);
241         end
        s.J=J;
243         s.options=options;
        if isequal(options.Design,'on')
245             obj.Des=s.CT;
        else
247             obj.n_analisi=obj.n_analisi+1;
            obj.Analisi{obj.n_analisi,1}=s;
249         end
        end
251         function I = simpsons(obj,f,a,b)

```

```

253     h=(b-a)/(length(f)-1);
        I= h/3*(f(1)+2*sum(f(3:2:end-2))+4*sum(f(2:2:end-1))+f(end)
            );
255     end
        %

%% Propeller Design
257     %

function obj=Design(obj,J,T,Cl,alpha,options)
259 % Design of a propeller that minimize the required power in
        certain
        % operating condition (fixed J and Thrust T).
261 % INPUT:
        % — J                Design advance ratio
263 % — T                Design Thrust [N]
        % — Cl                Design lift coefficient for each radial
265 %                        station
        % — alpha            Angle of attack corresponding to Cl
267 % — options            Analsys options (see BEMTset)
        %

arguments
    obj
271    J {mustBeNonnegative,mustBeFinite}
    T {mustBePositive,mustBeFinite}
273    Cl {mustBeReal,mustBeFinite}
    alpha {mustBeReal,mustBeFinite}
275    options=BEMTset();
end
277 options.Design='on';
options.P_correction='on';
279 if isempty(obj.LAMBDA)
    obj.LAMBDA = zeros(obj.n_r,1);
281 end
    %

if isequal(options.Freccia_opt,'on')
283     for i =1:obj.n_r
285         M_eff = sqrt((obj.OmR(i))^2+(J*obj.n*obj.D)^2)/obj.
            sound_vel;
            if M_eff > options.M_lim
287                 obj.LAMBDA(i) = acos(options.M_lim/M_eff);
            else
289                 obj.LAMBDA(i) = 0;
            end
291     end
end
end

```

```

293     V_inf=J*obj.n*obj.D;
        A=pi*obj.R^2;
295     CT= T/(obj.rho*obj.n^2*obj.D^4);
        % X=el.omega*el.R/V_inf*el.r_bar;
297     if V_inf < 1e-4
        w0=sqrt(T/(2*obj.rho*A));
299     else
        a=-0.5*(1-sqrt(1+2*T/(obj.rho*V_inf^2*A)));
301     w0=a*V_inf;
        end
303     f0=obj.funcDes(w0,V_inf,Cl,alpha,CT,options);
        if abs(f0)>options.toll
305         w01=1.5*w0;
        f1=obj.funcDes(w01,V_inf,Cl,alpha,CT,options);
307         while abs(f1)>options.toll
            qk=(f1-f0)/(w01-w0);
309             w0=w01; f0=f1;
            w01=w01-f1/qk;
311             f1=obj.funcDes(w01,V_inf,Cl,alpha,CT,options);
        end
313     else
        w01=w0;
315     end
        [~,obj]=obj.funcDes(w01,V_inf,Cl,alpha,CT,options);
317 end
function [f,obj]=funcDes(obj,w0,V_inf,Cl_id,alpha_id,CT,options
)
319     phi = atan2((V_inf+w0),(obj.omega*obj.r_bar*obj.R));
        if V_inf< 1e-4
321         OR_wo=obj.omega*obj.r_bar*obj.R/w0;
        w=w0*(1+OR_wo.^2).^-1;
323         ap=1*(1+OR_wo.^2).^-1;
        Ve =sqrt((obj.omega*obj.r_bar*obj.R).^2.*(1-ap).^2+w
            .^2);
325     else
        Chi=obj.omega*obj.r_bar*obj.R/V_inf;
327         a = w0/V_inf*Chi.^2.*((1+w0/V_inf)^2+Chi.^2).^-1;
        ap = w0/V_inf*(1+w0/V_inf)^2*((1+w0/V_inf)^2+Chi.^2)
            .^1;
329         Ve =sqrt((obj.omega*obj.r_bar*obj.R).^2.*(1-ap).^2+
            V_inf^2*(1+a).^2);
        end
331     Gamma = obj.omega*(obj.r_bar*obj.R).^2.*(4*pi*obj.r_bar
        .^2.*ap)...
        .*obj.F_(V_inf/obj.omega/obj.R);
333     obj.sigma = 1/(pi*obj.R)*(Ve.*obj.r_bar.*Cl_id).^-1.*Gamma;
        obj.c=(2*pi)*obj.sigma.*obj.r_bar*obj.R/obj.N;
335     obj.theta=phi+alpha_id;

337     obj=obj.BEMT(V_inf/(obj.n*obj.D),-2*pi/180,options);
        f=obj.Des/CT -1;
339 end

```



```

381         data_rot(j,:)=M*xz(j,:);
382     end
383     % scale airfoil
384     x_rot = data_rot(:,1)*obj.c(i);
385     z_rot = data_rot(:,2)*obj.c(i);
386     X(:,i) = x_rot - ...
387         mean(x_rot) - ...
388         (obj.r_bar(i) - r_star)*obj.R*tan(obj.LAMBDA(i)
389         )*cos(obj.theta(i));
390     Z(:,i)=z_rot - (obj.r_bar(i) - r_star)*obj.R*tan(obj.
391     LAMBDA(i))*sin(obj.theta(i));
392 end
393
394 Y=obj.r_bar'.*obj.R;
395 Y=repmat(Y,length(x),1);
396 % plot di una sola pala
397 figure
398 light('Style','local','Position',[1 -1 0]);
399 s=surf(X,Y,Z,'FaceColor',[0.65 0.65 0.65],'FaceLighting'
400     ,...
401     'gouraud','EdgeColor','none');
402 daspect([1 1 1])
403 camlight right
404 % material dull
405 xlabel('X [m]')
406 ylabel('Y [m]')
407 % plot di tutte le pale
408 figure
409 light('Style','local','Position',[1 -1 0]);
410 ang_blade=2*pi/obj.N;
411 xyz=[X(:),Y(:),Z(:)];
412 ang = 0;
413 hold on
414 grid on
415 for i =1:obj.N
416     xyz_rot=xyz*R3(ang);
417     X_rot=reshape(xyz_rot(:,1),[length(x),obj.n_r]);
418     Y_rot=reshape(xyz_rot(:,2),[length(x),obj.n_r]);
419     Z_rot=reshape(xyz_rot(:,3),[length(x),obj.n_r]);
420     s=surf(X_rot,Y_rot,Z_rot,'FaceColor',[0.65 0.65
421     0.65],...
422     'FaceLighting','gouraud','EdgeColor','none');
423     ang = ang + ang_blade;
424 end
425
426 % Hub(cilinder)
427 m=100;
428 % create Hub disc
429 r_hub = obj.r_bar(1)*obj.R;
430 xh=linspace(-r_hub,r_hub,m);
431 yh=sqrt((r_hub)^2-xh.^2);
432 xh=[xh,flip(xh)];

```

```

429     yh=[yh,-yh]';
430     Xh=repmat(xh,1,3);
431     Yh=repmat(yh,1,3);
432     % estrusion of the th Hub Disc
433     h_hub = 0.2*r_hub;
434     zh = h_hub*[-1,0, 1 + obj.c(1)*sin(obj.theta(1))/h_hub];
435     Zh = repmat(zh,length(xh),1);
436     surf(Xh,Yh,Zh,'FaceColor','k')
437     daspect([1 1 1])
438
439     xlabel('X [m]')
440     ylabel('Y [m]')
441     % Hub(parabolic)
442     m=100;
443     % create Hub disc
444     xh=linspace(-obj.r_bar(1),obj.r_bar(1),m)*obj.R;
445     yh=xh;
446     [Xh,Yh]=meshgrid(xh,yh);
447     % estrusion of the th Hub Disc
448     k=20;
449     Zh =-k*Xh.^2-k*Yh.^2+0.2;
450
451     hold on
452     surf(Xh,Yh,Zh)
453     daspect([1 1 1])
454     camlight HEADLIGHT
455
456 end
457
458 end
459
end

```


A.2 Il rotore

In questa sezione viene riportata una classe, definita in *Matlab*, per l'analisi delle prestazioni di un rotore di un elicottero. I metodi che agiscono su questa classe sono stati utilizzati per il calcolo delle prestazioni e per generare i *plot* presentati all'interno del documento. Il principale vantaggio nell'utilizzo di classi risiede nella robustezza dei dati di *input* e nell'utilizzo di funzioni specifiche per tali dati, denominate *metodi*. L'utilizzo di questa impostazione garantisce sempre la compatibilità dei dati in *input* per le funzioni.

Listing A.2:

```

1 classdef Rotor
2     properties
3         %
4
5         % Geometry
6         %
7         N      {mustBeInteger, mustBeFinite}      % Number of blades
8             , [\]
9         R      {mustBePositive, mustBeFinite}      % Rotor Radius, [m]
10            ]
11         theta_t (1,1){mustBeReal, mustBeFinite}    % pitch twist
12             angle, [rad]
13         c      (:,1){mustBeNonnegative, mustBeFinite} % Rotor chord, [m]
14         c_mean{mustBeNonnegative, mustBeFinite}    % Rotor chord, [m]
15             (?)
16         sigma {mustBeNonnegative, mustBeFinite}    % Mean solidity,
17             [\]
18         I      {mustBePositive, mustBeFinite}      % Moment of
19             inertia, [kgm^2]
20         gamma {mustBePositive, mustBeFinite}      % Lock number
21         LAMBDA {mustBeFinite}                    % Swept angle [rad] (it will be
22             used only for plotting)
23         %
24
25         % Working conditions
26         %
27
28         RPM     {mustBePositive, mustBeFinite}    % Giri al minuto
29         n       {mustBePositive, mustBeFinite}    % Giri al secondo
30         omega   {mustBePositive, mustBeFinite}    % Velocita' di
31             rotazione, [rad/s]
32         Mach_limit = 0.7                          % Maximum Mach
33             number at the tip of the blade
34         %

```

```

23      % Ambient conditions
24      %
25      h      {mustBeNonnegative, mustBeFinite}      % Quota di
           funzionamento
26      %
27      % Aerodynamics
28      %
29      Cl_alpha = 2*pi;
30      Cd_mean  = 0.01;
31      Cl = @(alpha) 2*pi*alpha;
32      Cd = @(alpha) 0.01*alpha./alpha;
33      %
34      %
35      % Analsys & Design
36      %
37      %
38      %
39      end
40      properties(SetAccess = private, GetAccess = public)
41      % radial vector
42      r_bar (:,1) {mustBeInRange(r_bar,0,1,'exclude-lower')} %
           exclusive
43      n_r
44      %
45      n_analisi_salita = 0
46      n_analisi_articulated = 0
47      n_analisi_autorot = 0
48      % ambient conditons
49      rho      {mustBePositive, mustBeFinite} = 1.23
50      press     {mustBePositive, mustBeFinite}
51      sound_vel {mustBePositive, mustBeFinite}
52      temp      {mustBePositive, mustBeFinite}
53      % geometry
54      D      {mustBePositive, mustBeFinite}      % Rotor diameter,
           [m]
55      A_D {mustBePositive, mustBeFinite}      % Rotor area, [m
           ^2]
56      k      % frequenza
           ridotta [\](c/2R)
57      % storage variables for analysis and design
58      Analisi_salita
59      Analisi_articulated;

```

```

Analisi_autorot;
Design
61 end
63
methods(Access=public)
65 % Compute ambient conditions. This function needs to be called
% first in order to set atmospheric conditions properly.
67 function obj = ambient(obj)
    [obj.temp, obj.sound_vel, obj.press, obj.rho] = atmosisa(
        obj.h);
69 end

71 % set radial domain
function obj = r(obj,vec_r)
73     obj.r_bar = vec_r;
    obj.n_r = length(obj.r_bar);
75 end

77 % mass properties
function obj = mass_prop(obj,valIN,val)
79     obj = obj.derived_properties();
    switch valIN
81         case 'G'
            obj.gamma = val;
83             obj.I = obj.Cl_alpha*obj.rho*obj.R^4*obj.c_mean
                /val;
            case 'I'
85                 obj.I= val;
                obj.gamma = obj.Cl_alpha*obj.rho*obj.R^4*obj.c_mean
                    ./obj.I;
87             otherwise
                mustBeMember(valIN,{'G','I'})
89         end
91     end

    % Compute some mass and geometric property. This function needs
    % to
93     % be called before doing any other calculations but still after
    % the
    % definition of the main properties. The function needs to be
    % called only once. The unction computes any derived property.
95     function obj = derived_properties(obj)
97         obj.D = obj.R*2;
        obj.A_D = pi*obj.R*obj.R;
99         obj.c_mean= mean(obj.c);
        obj.sigma = ( obj.c_mean*obj.N )/( pi*obj.R );
101        obj.k = obj.c_mean/2/obj.R;
103    end

    % Compute rotational velocity. This function let the user
    % decide
105    % the tip speed depending on the input parameters: sometime the

```

```

107 % rotational velocity of the rotor can be known, otherwise the
108 % upper limit for the tip speed is set by compressibility
109 % requirements: in this case the user should select the value 1
110 for
111 % the flag. While the flag is equal to 1 there's no need to
112 give
113 % the function other inputs.
114 function obj = rot_vel(obj,valIN,val,flag_external_tip_speed)
115 arguments
116     obj
117     valIN
118     val
119     flag_external_tip_speed = 0
120 end
121 if flag_external_tip_speed == 1
122     obj.omega = obj.sound_vel*obj.Mach_limit/obj.R;
123     obj.n      = obj.omega/(2*pi);
124     obj.RPM    = obj.n*60;
125 else
126     switch valIN
127     case 'RPM'
128         obj.RPM    = val;
129         obj.n      = obj.RPM/60;
130         obj.omega  = obj.n*2*pi;
131     case 'n'
132         obj.n      = val;
133         obj.RPM    = obj.n*60;
134         obj.omega  = obj.n*2*pi;
135     case 'omega'
136         obj.omega  = val;
137         obj.n      = obj.omega/(2*pi);
138         obj.RPM    = obj.n*60;
139     otherwise
140         mustBeMember(valIN,{'RPM','n','omega'})
141     end
142 end
143 end
144
145 %
146 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
147
148 %% BEMT salita assiale
149 %
150 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
151
152 function obj = BEMT_salita(obj,V_inf,theta0,options)
153 %
154 %
155 % Questa funzione consente di calcolare le prestazioni del
156 % rotore
157 % una volta fissata la velocita' di salita, ovvero il

```

```

    parametro
149 % mu. La procedura di calcolo assume valida l'ipotesi di
    % trascurabilita' dell'induzione radiale, che conduce alla
151 % definizione di una teoria esplicita per il calcolo delle
    % prestazioni del rotore. La funzione calcola le
    prestazioni per
153 % singolo valore del rapporto di funzionamento, tuttavia
    puo'
    % ricevere in input un vettore di rapporti di velocita'. In
    tal caso
155 % restituisce una matrice di gradienti di spinta e coppia
    aventi
    % per dimensioni dimesione_radiale x dimensione vettore
    velocita'
157 % Input:
    % — obj:
159 %     — omega: velocita' di rotazione del rotore.
    %     — sigma: solidita' del rotore.
161 %     — teta: calettamento del rotore.
    %     — R: raggio del rotore.
163 %     — r_bar: raggio adimensionalizzato del rotore.
    % — V_inf: velocita' di traslazione, necessaria per il
    calcolo del
165 %     rapporto di funzionamento, mu.
    % — theta0: Comando collettivo, pitch alla radice [rad]
167 % — options: Analisis options (see BEMTset_rotor)
    % Output:
169 % — dTc/dr: gradiente di spinta per stazione fissata lungo
    la pala
    % — dQc/dr: gradiente di coppia per stazione fissata lungo
    la pala
171 %

arguments
173 obj
    V_inf      (:,1){mustBeFinite}
175 theta0      (1,1){mustBeFinite}
    options = BEMTset_rotor();
177 end
    theta = theta0 + obj.theta_t*obj.r_bar;
179 for i=1:length(V_inf)
    mu(i)      = V_inf(i)/( obj.R*obj.omega );
181 for j=1:obj.n_r
    B(i,j)      = mu(i) + ( obj.Cl_alpha.*obj.sigma )
                /8;
183 B2(i,j)      = B(i,j)*B(i,j);
    C(i,j)      = obj.r_bar(j)*obj.Cl_alpha*obj.sigma
                /8*...
185 ( theta(j) — (mu(i)/obj.r_bar(j)) );
    s.lam_i(i,j) = 0.5*( sqrt( B2(i,j) + 4*C(i,j) ) — B
        (i,j) );

```

```

187         % inflow angle
        s.phi(i,j) = ( mu(i) + s.lam_i(i,j) )./obj.r_bar(
            j);
189         % angle of attack
        s.alpha(i,j) = theta(j) - s.phi(i,j);
191         s.Cl(i,j) = obj.Cl(s.alpha(i,j));
        s.Cd(i,j) = obj.Cd(s.alpha(i,j));
193         % thrust and torque distributions
        s.dTc(i,j) = 0.5*obj.sigma*s.Cl(i,j)*(obj.r_bar(j)
            ^2);
195         s.dQc_i(i,j) = 0.5*obj.sigma*(s.Cl(i,j)*s.phi(i,j))
            *(obj.r_bar(j)^3);
        s.dQc_0(i,j) = 0.5*obj.sigma*(s.Cd(i,j))*(obj.r_bar
            (j)^3);
197         s.dQc(i,j) = s.dQc_0(i,j) + s.dQc_i(i,j);
        end
199         switch options.P_correction
            case 'on'
201                 s.dTc(i,:) = s.dTc(i,:)'.*obj.F_(V_inf(i)/( obj
                    .R*obj.omega));
                    options.B = 1;
203         end
        % thrust and torque
205         s.Tc(i) = obj.simpsons(s.dTc(i,:),obj.r_bar(1),
            options.B*obj.r_bar(end));
        s.Qc_i(i) = obj.simpsons(s.dQc_i(i,:),obj.r_bar(1),obj.
            r_bar(end));
207         s.Qc_0(i) = obj.simpsons(s.dQc_0(i,:),obj.r_bar(1),obj.
            r_bar(end));
        s.Qc(i) = s.Qc_0(i)+ s.Qc_i(i);
209         if s.Tc(i)>0
            s.k(i) = s.Qc_i(i)/(s.Tc(i)^(1.5)/sqrt(2));
211         else
            s.k(i) = NaN;
213         end

215     end
    s.mu = mu;
217    s.theta0 = theta0;
    s.theta = theta;
219    obj.n_analisi_salita = obj.n_analisi_salita+1;
    obj.Analisi_salita{obj.n_analisi_salita,1} = s;
221 end

223 function I = simpsons(obj,f,a,b)
    h=(b-a)/(length(f)-1);
225    I= h/3*(f(1)+2*sum(f(3:2:end-2))+4*sum(f(2:2:end-1))+f(end)
        );
    end
227 function F=F_(obj,lambda)
    % Prandtl Correction F (computed for all r_bar)
229    % Input:

```

```

% — lambda          advance ratio (see pag. 44)
    [1,1]
231 % Output :
% — F              Prandtl correction factor
233 %              [obj.n_r,1]
F=2/pi*acos(exp(0.5*obj.N/lambda*(obj.r_bar-1)));
235 if lambda == 0
    F(isnan(F))=1;
237 end
end
239 %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% BEMT volo traslato per rotore non rigido (articolato)
241 %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function obj = BEMT_articulated(obj,valIN,ToTheta,V_inf_Vec,chi
,f,options)
243 %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Questa funzione consente di calcolare i coefficienti
    adimensionali di spinta,
245 % resistenza, forza laterale, coppia e potenza, gli angoli
    di
% flappeggio, l'angolo di attacco del rotore e l'angolo di
247 % inflow. Tutte le operazioni si basa sulle ipotesi di
    piccoli
% angoli, sono richiesti: piccoli angoli di attacco del
    rotore,
249 % piccoli angoli di flappeggio, piccoli angoli di salita.
% Vengono poi supposte costanti le velocita' di traslazione
    dell'elicottero e
251 % angolare del rotore. Si assume induzione costante sul
    rotore
% e si considera come unica cerniera presente quella di
253 % flappeggio (cfr. ipotesi per il rotore articolato).
    Questa
% funzione e' stata progettata anche per lavorare in coppia
    con
255 % la classe Helicopter.m, tuttavia qualora fornite le
    variabili
% di input e' possibile utilizzare questo metodo come
    standalone
257 % in un codice ad hoc, in cui si utilizza la sola classe
    % Rotor.m.
259 % Input:
% — valIN          : Flag per scegliere il tipo di
    risoluzione
261 %              delproblema: T → Spinta fissata;
%              Theta → collettivo fissato.

```

```

263 % — ToTheta[N o rad]: Spinta richiesta o comando collettivo
%                               (theta0) a seconda del flag valIN
265 % — V_inf[m/s]      : velocita' di avanzamento del rotore
% — Chi  [rad]       : angolo di salita del rotore
267 % — f    [visc.area]: prodotto dell'area di riferimento per
%                               la
%                               resistenza dell'intero elicottero.
269 % — options          : Analisis options (see BEMTset_rotor)
arguments
271     obj
    valIN      {mustBeMember(valIN,{'T','Theta',})}
273     ToTheta   (1,1) {mustBeFinite}
    V_inf_Vec  (:,1){mustBeNonnegative,mustBeFinite}
275     chi       {mustBeFinite}
    f         {mustBePositive, mustBeFinite}
277     options = BEMTset_rotor();
end
279
switch valIN
281     case T
        T = ToTheta;
283         Tc = T/( obj.rho*(obj.omega*obj.R)^2*obj.A_D );
    case Theta
        theta0 = ToTheta;
    otherwise
287         error('Attezione alla scelta del metodo. Scegliere
            tra T o Theta')
end
289
291
for i = 1:length(V_inf_Vec)
293     iter      = 0;          % contatore iterazioni
    iter_cond = 1;          % residuo
295     V_inf     = V_inf_Vec(i)*cos(chi);
    alpha_TPP = 0;          % valore di primo tentativo
297     lam       = 0;          % valore di primo tentativo
    if isequal(valIN,Theta)
299         mu     = ( V_inf*cos(alpha_TPP) )/( obj.omega*obj.R
            );
        %
            [lam_i,lam,Tc]=
                initialize_routin(obj,mu,alpha_TPP,theta0,
                    options);
301         Tc = 0.5*(obj.sigma*obj.Cl_alpha)*...
            (theta0/3*(1+1.5*mu^2) + obj.theta_t/4*(1+mu^2)
                ...
            — lam/2);
303     end
    D_fs      = 0.5*obj.rho*V_inf.^2*f;
    lam_c      = V_inf*sin(chi)/( obj.omega*obj.R );
307
    while iter_cond > options.toll

```



```

309      mu      = ( V_inf*cos(alpha_TPP) )/( obj.omega*obj.R
        );

311      if sqrt(mu^2 + lam^2) < 1e-2
          % nella prima iterazione lam=0 e se mu=0 allora
313      % usiamo la formula dell'induzione in hovering
          lam_i = sqrt(Tc/2);
315      else
          lam_i = Tc/( 2*sqrt(mu^2 + lam^2) );
317      end
      lam = mu*tan(alpha_TPP) + lam_i;
319      if isequal(valIN,T)
          theta0 = ( 3 /( 1 + 1.5*mu.^2 ) )*...
321      ( (2*Tc)/(obj.sigma*obj.Cl_alpha) ...
          - obj.theta_t/4 - obj.theta_t*( mu.^2 )/4 +
          0.5*lam );
323      else
          Tc = 0.5*(obj.sigma*obj.Cl_alpha)*...
325      (theta0/3*(1+1.5*mu^2) + obj.theta_t/4*(1+
          mu^2)...
          - lam/2);
327      end

329      Pc0      = obj.Cd_mean*obj.sigma*( 1 + options.k_mu
          *mu^2 )/8;
      Pci      = options.k_i*lam_i*Tc;
331      Pc_fus   = mu*( D_fs/(Tc*obj.rho*obj.omega^2*obj.R
          ^4*pi) )*Tc;
      Pc       = Pci + lam_c*Tc + Pc_fus + Pc0;
333

335      % flap coeffs.
      beta0    = obj.gamma*( theta0/8*(1 + mu^2) + ...
337      obj.theta_t/10*(1 + 5*(mu^2)/6) - lam/6 );
      betalc   = -2*mu*( (4*theta0/3 + obj.theta_t - lam)
          ...
339      /(1 - 0.5*mu^2) );
      betals   = -4*mu/3*beta0/(1 + 0.5*mu^2);
341

343      % drag and side force coeffs.
      % induced drag coeff.
      Hc_i     = obj.sigma*obj.Cl_alpha*0.5*( theta0*( -
          betalc/3 + 0.5*mu*lam ) + ...
345      obj.theta_t*( -betalc/4 + mu*lam/4 ) + 3*lam*
          betalc/4 + beta0*betals/6 + ...
          mu*( beta0^2 + betalc^2 )/4 );
347      % parasite drag coeff.
      Hc_0     = obj.sigma*obj.Cd_mean*mu/4;
349      % total drag coeff.
      Hc       = Hc_i + Hc_0;
351      % total lateral force coeff.
      Yc       = -obj.sigma*obj.Cl_alpha*0.5*...

```

```

353         ( theta0*( 3*mu*beta0/4 + betals*( 1 + 0.5*3*mu
              ^2 )/3 ) +...
              obj.theta_t*( 0.5*mu*beta0 + betals*( 1 + mu^2
              )/4 )...
355         - 3*lam*betals/4 + beta0*betalc*( 1/6 - mu^2 )
              - ...
              0.5*3*mu*lam*beta0 - betalc*betals/4);
357     % modified lam let us compute the variation of lam
              of the previous
    % iteration
359     lam_temp = lam;
    lam        = lam_i + lam_c + mu*( Hc/Tc ) + ...
361     mu*( D_fs/(Tc*obj.rho*obj.omega^2*obj.R^4*pi) )
              ;
    alpha_TPP= atan2((lam - Tc/(2*sqrt( mu^2 + lam^2 ))
              ),mu);
363     iter_cond= abs(lam - lam_temp);
    iter       = iter + 1;
365     end

367     s_art.iter_ART(i)      = iter;
    s_art.iter_cond_ART(i) = iter_cond;
369     s_art.lam_Vec(i)      = lam;
    s_art.lam_i(i)         = lam_i;
371     s_art.lam_c(i)        = lam_c;
    s_art.mu(i)            = mu;
373     s_art.Tc(i)           = Tc;
    s_art.Pc_Vec(i)        = Pc;
375     s_art.Pc_fus_Vec(i)   = Pc_fus;
    s_art.Pci_Vec(i)       = Pci;
377     s_art.Pc0_Vec(i)      = Pc0;
    s_art.Hc_i_Vec(i)      = Hc_i;
379     s_art.Hc_0_Vec(i)     = Hc_0;
    s_art.Hc_Vec(i)        = Hc;
381     s_art.Yc_Vec(i)       = Yc;
    s_art.beta0_Vec(i)     = beta0;
383     s_art.betalc_Vec(i)   = betalc;
    s_art.betals_Vec(i)    = betals;
385     s_art.alpha_TPP_Vec(i) = alpha_TPP;
    s_art.theta0(i)        = theta0;
387     s_art.theta(:,i)     = theta0 + obj.theta_t*obj.r_bar
              ;

389     end
    s_art.V_inf            = V_inf_Vec;
391     s_art.options         = options;
    [s_art.alpha_e,s_art.Mach_e] = alpha_e(obj,s_art);
393     obj.n_analisi_articulated = obj.n_analisi_articulated
              +1;
    obj.Analisi_articulated{obj.n_analisi_articulated,1} =
              s_art;
395

```

```

end
397
% compute angle of attack for each BE: alpha_e(r_bar,Psi)
399 function [alpha_e,Mach_e] = alpha_e(obj,s)
    Psi=s.options.Psi;
    alpha_e=zeros(obj.n_r,length(Psi),length(s.lam_Vec));
    Mach_e =zeros(obj.n_r,length(Psi),length(s.lam_Vec));
    401
    for idxV=1 : length(s.lam_Vec)
        b      = s.beta0_Vec(idxV) + ...
    403         s.betalc_Vec(idxV)*cos(Psi) +...
    405         s.betals_Vec(idxV)*sin(Psi);
    407     b_dot = -s.betalc_Vec(idxV)*sin(Psi) +...
        s.betals_Vec(idxV)*cos(Psi);
    409
        for i=1:obj.n_r
    411             for j=1:length(Psi)
                alpha_e(i,j,idxV) = s.theta(i,idxV) -...
    413                 atan2((s.lam_Vec(idxV) +...
                    b_dot(j)*obj.r_bar(i)/obj.omega+...
    415                     b(j)*s.mu(idxV)*cos(Psi(j))),(...
                        obj.r_bar(i) + s.mu(idxV)*sin(Psi(j))));
    417             Mach_e(i,j,idxV) = ( (s.lam_Vec(idxV) +...
                b_dot(j)*obj.r_bar(i)/obj.omega+...
    419                 b(j)*s.mu(idxV)*cos(Psi(j)))^2 + (...
                    obj.r_bar(i) + s.mu(idxV)*sin(Psi(j)))^2 )
                ...
    421             *obj.omega*obj.R/obj.sound_vel;
        end
    423     end
    end
    425 end
    %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    427 %% Sentiero di Stallo
    %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    429 function [s,ir,c] = sentiero_stallo(obj,alpha_max_2D,Vvec,valIN
        ,ToTheta,chi,f,options,color)
    %
    431
    % Questa funzione consente di calcolare il sentiero di
    % stallo
    % per assegnata spinta o callettamento di radice (comando
    433 % collettivo), angolo di salita (chi) e resistenza (f).
    % Il raggiungimento della condizione critica e'basato
    435 % sull'angolo di stallo 2D del profilo della pala in
    % condizioni
    % stazionarie (l'unico effetto instazionario di cui si
    437 % tiene in conto e' l'effetto smorzante d(beta)/dt).

```

```

439 %
440 % Input:
441 % — alpha_max_2D      : Angolo di stallo 2D
442 % — Vvec               : Vettore di incrementi rispetto alla
443 %                       V_inf
444 %                       in corrispondenza della quale si ha
445 %                       per
446 %                       la prima volta un AoA = alpha_max_2D.
447 %                       (
448 %                       ad esempio [1:0.01:1.1])
449 % — valIN              : Flag per scegliere il tipo di
450 %                       risoluzione
451 %                       del problema: T → Spinta fissata;
452 %                       Theta → collettivo fissato.
453 % — ToTheta[N o rad]: Spinta richiesta o comando collettivo
454 %                       (theta0) a seconda del flag valIN
455 % — Chi [rad]         : angolo di salita del rotore
456 % — f [visc.area]: prodotto dell'area di riferimento per
457 %                       la
458 %                       resistenza dell'intero elicottero.
459 % — options           : Analysis options (see BEMTset_rotor)
460 % Output:
461 % — s                 : Struct (output di BEMT_articulated)
462 %                       in
463 %                       corrispondenza della condizione di
464 %                       inizio stallo
465 % — ir e c            : indice di riga e colonna in s.alpha_e
466 %                       della posizioni della prima sezione
467 %                       stallata. In coordinate polari
468 %                       (obj.r_bar(r),s.options.Psi(c))
469 %
470 %
471 %
472 %
473 %
474 %
475 arguments
476 obj
477 alpha_max_2D (1,1) {mustBePositive, mustBeFinite}
478 Vvec         {mustBePositive,mustBeFinite}
479 valIN        {mustBeMember(valIN,{'T','Theta'},)}
480 ToTheta      (1,1) {mustBeFinite}
481 chi          {mustBeFinite}
482 f            {mustBePositive, mustBeFinite}
483 options = BEMTset_rotor();
484 color = 'on'
485 end
486 %
487 %
488 %
489 %
490 %
491 %
492 %
493 %
494 %
495 %
496 %
497 %
498 %
499 %
500 %
501 %
502 %
503 %
504 %
505 %
506 %
507 %
508 %
509 %
510 %
511 %
512 %
513 %
514 %
515 %
516 %
517 %
518 %
519 %
520 %
521 %
522 %
523 %
524 %
525 %
526 %
527 %
528 %
529 %
530 %
531 %
532 %
533 %
534 %
535 %
536 %
537 %
538 %
539 %
540 %
541 %
542 %
543 %
544 %
545 %
546 %
547 %
548 %
549 %
550 %
551 %
552 %
553 %
554 %
555 %
556 %
557 %
558 %
559 %
560 %
561 %
562 %
563 %
564 %
565 %
566 %
567 %
568 %
569 %
570 %
571 %
572 %
573 %
574 %
575 %
576 %
577 %
578 %
579 %
580 %
581 %
582 %
583 %
584 %
585 %
586 %
587 %
588 %
589 %
590 %
591 %
592 %
593 %
594 %
595 %
596 %
597 %
598 %
599 %
600 %
601 %
602 %
603 %
604 %
605 %
606 %
607 %
608 %
609 %
610 %
611 %
612 %
613 %
614 %
615 %
616 %
617 %
618 %
619 %
620 %
621 %
622 %
623 %
624 %
625 %
626 %
627 %
628 %
629 %
630 %
631 %
632 %
633 %
634 %
635 %
636 %
637 %
638 %
639 %
640 %
641 %
642 %
643 %
644 %
645 %
646 %
647 %
648 %
649 %
650 %
651 %
652 %
653 %
654 %
655 %
656 %
657 %
658 %
659 %
660 %
661 %
662 %
663 %
664 %
665 %
666 %
667 %
668 %
669 %
670 %
671 %
672 %
673 %
674 %
675 %
676 %
677 %
678 %
679 %
680 %
681 %
682 %
683 %
684 %
685 %
686 %
687 %
688 %
689 %
690 %
691 %
692 %
693 %
694 %
695 %
696 %
697 %
698 %
699 %
700 %
701 %
702 %
703 %
704 %
705 %
706 %
707 %
708 %
709 %
710 %
711 %
712 %
713 %
714 %
715 %
716 %
717 %
718 %
719 %
720 %
721 %
722 %
723 %
724 %
725 %
726 %
727 %
728 %
729 %
730 %
731 %
732 %
733 %
734 %
735 %
736 %
737 %
738 %
739 %
740 %
741 %
742 %
743 %
744 %
745 %
746 %
747 %
748 %
749 %
750 %
751 %
752 %
753 %
754 %
755 %
756 %
757 %
758 %
759 %
760 %
761 %
762 %
763 %
764 %
765 %
766 %
767 %
768 %
769 %
770 %
771 %
772 %
773 %
774 %
775 %
776 %
777 %
778 %
779 %
780 %
781 %
782 %
783 %
784 %
785 %
786 %
787 %
788 %
789 %
790 %
791 %
792 %
793 %
794 %
795 %
796 %
797 %
798 %
799 %
800 %
801 %
802 %
803 %
804 %
805 %
806 %
807 %
808 %
809 %
810 %
811 %
812 %
813 %
814 %
815 %
816 %
817 %
818 %
819 %
820 %
821 %
822 %
823 %
824 %
825 %
826 %
827 %
828 %
829 %
830 %
831 %
832 %
833 %
834 %
835 %
836 %
837 %
838 %
839 %
840 %
841 %
842 %
843 %
844 %
845 %
846 %
847 %
848 %
849 %
850 %
851 %
852 %
853 %
854 %
855 %
856 %
857 %
858 %
859 %
860 %
861 %
862 %
863 %
864 %
865 %
866 %
867 %
868 %
869 %
870 %
871 %
872 %
873 %
874 %
875 %
876 %
877 %
878 %
879 %
880 %
881 %
882 %
883 %
884 %
885 %
886 %
887 %
888 %
889 %
890 %
891 %
892 %
893 %
894 %
895 %
896 %
897 %
898 %
899 %
900 %
901 %
902 %
903 %
904 %
905 %
906 %
907 %
908 %
909 %
910 %
911 %
912 %
913 %
914 %
915 %
916 %
917 %
918 %
919 %
920 %
921 %
922 %
923 %
924 %
925 %
926 %
927 %
928 %
929 %
930 %
931 %
932 %
933 %
934 %
935 %
936 %
937 %
938 %
939 %
940 %
941 %
942 %
943 %
944 %
945 %
946 %
947 %
948 %
949 %
950 %
951 %
952 %
953 %
954 %
955 %
956 %
957 %
958 %
959 %
960 %
961 %
962 %
963 %
964 %
965 %
966 %
967 %
968 %
969 %
970 %
971 %
972 %
973 %
974 %
975 %
976 %
977 %
978 %
979 %
980 %
981 %
982 %
983 %
984 %
985 %
986 %
987 %
988 %
989 %
990 %
991 %
992 %
993 %
994 %
995 %
996 %
997 %
998 %
999 %
1000 %

```

```

479         f1=obj.func(V_inf1,alpha_max_2D,valIN,ToTheta,chi,f,
                    options);
481         while abs(f1)>options.toll
            qk=(f1-f0)/(V_inf1-V_inf0);
            V_inf0=V_inf1; f0=f1;
483         V_inf1=V_inf1-f1/qk;
            f1=obj.func(V_inf1,alpha_max_2D,valIN,ToTheta,chi,f,
                    ,options);
485         end
487         else
            V_inf1=V_inf0;
489         end
        [~,ir,c,s]= func(obj,V_inf1,alpha_max_2D,valIN,ToTheta,chi,
            f,options);

491         % mappa di alpha_e
            s = alphamap(obj,'Plot',{s;s.mu},color);
493         % plot di alpha_e_max
            % Create polar data
495         [r,psi] = meshgrid(obj.r_bar,s.options.Psi);
            % Convert to Cartesian
497         x = r.*cos(psi);
            y = r.*sin(psi);
499         hold on
            plot(x(c,ir),y(c,ir),'*k','MarkerSize',10)
501
503         % Dopo aver calcolato la mappa degli angoli attacco adesso
            % calcoliamo il sentiero di stallo vero e proprio
            figure
505         % ricavo alpha_e per un vettore i velocita' incrementali
            % rispetto a quella critica
507         V_inf = s.V_inf*Vvec;
            obj2 = BEMT_articulated(obj,valIN,ToTheta,V_inf,chi,f,
                options);
509         alpha_e = obj2.Analisi_articulated{obj2.
            n_analisi_articulated,1}.alpha_e;
            mu      = obj2.Analisi_articulated{obj2.
            n_analisi_articulated,1}.mu;
511         a_TPP    = obj2.Analisi_articulated{obj2.
            n_analisi_articulated,1}.alpha_TPP_Vec;
            mu = mu.* cos(a_TPP).^-1;
513         clear obj2;
            polarplot(s.options.Psi(c),obj.r_bar(ir),'*k','MarkerSize'
                ,10,...
515                 'DisplayName',['\mu = ',num2str(mu(1))]);
            hold on;
517         %           formatspec={'-',':', '—', '.-', '-s', '-d
                ', '-^'};
            formatspec={'*','^','s','d','>','<','o'}; ik=0;
519         for k = 2:length(V_inf)
            [ir,ic] = find(( alpha_e(:, :,k) - alpha_max_2D) <
                convang(0.1,'deg','rad') ) &...

```

```

521         ( (alpha_e(:, :, k) - alpha_max_2D) > 0 ) );
        %           p = polyfit(s.options.Psi(ic),obj.
        r_bar(ir),max(min(3,length(ir)-1),1));
523         %           r_new = polyval(p,s.options.Psi(ic));
        %           polarplot(s.options.Psi(ic),r_new,[
        formatspec{k-1},'k'],...
525         %           'DisplayName',['\mu = ',num2str(
            mu(k))]);
        % senza interpolare
527         ik= ik +1;
        if ik > 7
529             ik = 1;
        end
531         polarplot(s.options.Psi(ic),obj.r_bar(ir),[formatspec{
            ik},'k'],...
            'DisplayName',['\mu = ',sprintf('%0.2f',mu(k))]);
533     end

535     ax = gca;
    ax.ThetaZeroLocation = 'bottom';
537     ax.RTickLabel = '';
    legend1 = legend('AutoUpdate','off','FontSize',14);
539     % limiti rotore
    polarplot(s.options.Psi,obj.r_bar(1)*ones(length(s.options.
        Psi),1),'k',...
541         'LineWidth',1)
    hold on;
543     polarplot(s.options.Psi,obj.r_bar(end)*ones(length(s.
        options.Psi),1),'k',...
        'LineWidth',1)

545
547
549     %%%% Prova con funzione contour
    %           figure
551     %           h=polar(x,y);
    %           hold on
553     %           for k = 2:length(V_inf)-3
        %           [ir,ic] = find(( (alpha_e(:, :, k) -
        alpha_max_2D) < convang(0.1,'deg','rad') ) &...
555     %           ( (alpha_e(:, :, k) - alpha_max_2D) > 0
            ) );
        %           A = zeros(size(x));
        %           for i = 1:length(ic)
        %           A(ir(i),ic(i)) = obj2.
        Analisi_articulated{obj2.n_analisi_articulated,1}.mu(k)
        ;
559     %           end
    % %           p = polyfit(s.options.Psi(ic),obj.r_bar
        (ir),max(min(3,length(ir)-1),1));
561     % %           r_new = polyval(p,s.options.Psi(ic));

```

```

563         % %                polarplot(s.options.Psi(ic),r_new,'k');
564         %                contour(x,y,A','ShowText','on')
565         %                end
566         %                % Hide the POLAR function data and leave
567         %                annotations
568         %                set(h,'Visible','off')
569         %                % Turn off axes and set square aspect ratio
570         %                axis off
571         %                axis image
572         %                view([90 90])
573         %                title('Sentiero di stallo')
574
575     end
576
577     function [f,r,c,s]= func(obj,V_inf,alpha_max_2D,valIN,ToTheta,
578         chi,f,options)
579         obj2=obj.BEMT_articulated(valIN,ToTheta,V_inf,chi,f,options
580             );
581         s=obj2.Analisi_articulated{obj2.n_analisi_articulated,1};
582         alpha_e=max(s.alpha_e,[],'all');
583         [r,c] = find(s.alpha_e == alpha_e);
584         f = (alpha_e -alpha_max_2D)/alpha_max_2D;
585     end
586
587     %
588     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
589
590     %% Analisi dell'autorotazione
591     %
592     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
593
594     function obj = autorot_rotor(obj,V_inf_Vec,chi,f,W,theta_t)
595         %
596
597         % Questa funzione consente di calcolare le caratteristiche
598         % della manovra di autorotazione per il rotore in analisi.
599         % Come
600         % descritto a pagina 110 degli appunti, e' necessario
601         % imporre i
602         % parametri lambda_climb e mu. Come prescritto dalla
603         % manovra di
604         % autorotazione si impone che  $P = 0$  e si ricava una
605         % relazione
606         % per theta_0. Noto lambda_climb si possono calcolare
607         % lambda e
608         % lambda indotto. Noti quesri coefficienti si possono
609         % calcolare
610         % i coefficienti di forza. Noto Tc e il peso W si puo'
611         % ricavare
612         % Omega, la velocita' angolare del rotore. Infine si

```

```

        calcola
        % alfa e V_infinity.
599
    end
601    %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %% PLOTTING
603    %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function s=alphamap(obj,valIN,val,color)
605    % Plot alpha_e contour
    % INPUT:
607    % — valIN:    flag per l'input val:
        %                — 'Plot'  -> in tal caso val dovra'
        %                essere
609    %                un cell array 2x1 in cui vi e'
        %                una struct (output di BEMT_articulated)
        %                ed
611    %                un vettore di valori di mu per i quali
        %                si
613    %                desidera effettuare i plot
        %                — 'Solve' -> in tal caso val dovra'
        %                essere
615    %                una cell array 6x1 con gli input da
        %                dare
        %                alla funzione BEMT_articulated
617    % — color:    flag per il pcolor:
        %                — 'on'    -> plotta il contour a colori
619    %                — altro    -> plotta le isolinee senza
        %                colori
621    %
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

arguments
    obj
625    valIN
    val
627    color = 'on'
end
629    %

    switch valIN
631        case 'Plot'
            s=val{1,1};
633            for i =1:length(val{2,1})
                [~,idxMu(i)]=min(abs(s.mu - val{2,1}(i)));
635            end

```



```

637         case 'Solve'
        valIN      = val{1,1};
        ToTheta    = val{2};
639         V_inf_Vec = val{3};
        chi        = val{4};
641         f         = val{5};
        options    = val{6};
643         obj2=obj.BEMT_articulated(valIN,ToTheta,V_inf_Vec
            ,...
            chi,f,options);
645         s=obj2.Analisi_articulated{obj2.
            n_analisi_articulated,1};
            idxMu=1:length(s.mu);
647     otherwise
        error('Attenzione valIN puo' essere: Plot o Solve')
649     end

651
652     for i = 1:length(idxDu)
653         figure
        idxV=idxMu(i);
655         alpha_e=s.alpha_e(:, :,idxV)*180/pi;
        alpha_e(alpha_e <0) = 0;
657         % Create polar data
        [r,psi] = meshgrid(obj.r_bar,s.options.Psi);
659         % Convert to Cartesian
        x = r.*cos(psi);
        y = r.*sin(psi);
661         % define polar axes
        h = polar(x,y);
663         hold on;
        polar(s.options.Psi,obj.r_bar(1)*ones(length(s.options.
665             Psi),1),'k')
        polar(s.options.Psi,obj.r_bar(end)*ones(length(s.
            options.Psi),1),'k')
667         switch color
            case 'on'
669             % contourf(x,y,alpha_e');
            pc= pcolor(x,y,alpha_e');
671             cbar=colorbar(gca);
            cbar.Label.String = '\alpha_e';
673             cbar.Label.FontSize= 16;
            % cbar.Limits = [-10 10];
675         otherwise

677     end

679     contour(x,y,alpha_e','k','ShowText','on');
    shading interp
681
683

```

```

685         % Hide the POLAR function data and leave annotations
        set(h,'Visible','off')
687         % Turn off axes and set square aspect ratio
        axis off
        axis image
689         view([90 90])
        s_mu = sprintf('%0.2f',s.mu(idxV)/cos(s.alpha_TPP_Vec(
            idxV)));
691         s_a = sprintf('%0.2f',max(alpha_e,[],'all'));
        title(['\mu = ',s_mu,' \alpha_{e_{
            max}} = ',s_a,' deg'])
693     end
695
697
699 end
%

```

```

701 function s=MachMap(obj,valIN,val,color)
    % Plot alpha_e contour
703     % INPUT:
    % — valIN:      flag per l'input val:
705     %               — 'Plot'  → in tal caso val dovra'
    %               essere
    %               un cell array 2x1 in cui vi e'
707     %               una struct (output di BEMT_articulated)
    %               ed
    %               un vettore di valori di mu per i quali
    %               si
709     %               desidera effettuare i plot
    %               — 'Solve' → in tal caso val dovra'
711     %               essere
    %               una cell array 6x1 con gli input da
    %               dare
713     %               alla funzione BEMT_articulated
    % — color:      flag per il pcolor:
715     %               — 'on'    → plotta il contour a colori
    %               — altro    → plotta le isolinee senza
717     %               colori
    %
719     %
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    arguments
721         obj
        valIN
723         val
        color = 'on'

```

```

725     end
726     %
727
728     switch valIN
729         case 'Plot'
730             s=val{1,1};
731             for i =1:length(val{2,1})
732                 [~,idxMu(i)]=min(abs(s.mu - val{2,1}(i)));
733             end
734         case 'Solve'
735             valIN      = val{1,1};
736             ToTheta    = val{2};
737             V_inf_Vec  = val{3};
738             chi         = val{4};
739             f           = val{5};
740             options     = val{6};
741             obj2=obj.BEMT_articulated(valIN,ToTheta,V_inf_Vec
742                                     ,...
743                                     chi,f,options);
744             s=obj2.Analisi_articulated{obj2.
745                 n_analisi_articulated,1};
746             idxMu=1:length(s.mu);
747         otherwise
748             error('Attenzione valIN puo' essere: Plot o Solve')
749         end
750
751     for i = 1:length(idxMu)
752         figure
753         idxV=idxMu(i);
754         M_e=s.Mach_e(:, :,idxV);
755         % Create polar data
756         [r,psi] = meshgrid(obj.r_bar,s.options.Psi);
757         % Convert to Cartesian
758         x = r.*cos(psi);
759         y = r.*sin(psi);
760         % define polar axes
761         h = polar(x,y);
762         hold on;
763         polar(s.options.Psi,obj.r_bar(1)*ones(length(s.options.
764             Psi),1),'k')
765         polar(s.options.Psi,obj.r_bar(end)*ones(length(s.
766             options.Psi),1),'k')
767         % contourf(x,y,alpha_e');
768         switch color
769             case 'on'
770                 pc= pcolor(x,y,M_e');
771                 cbar=colorbar(gca);
772                 cbar.Label.String = 'Mach_e';
773                 cbar.Label.FontSize= 16;
774             otherwise

```

```

771         end
773
774         contour(x,y,M_e','k','ShowText','on');
775         shading interp
776
777         % Hide the POLAR function data and leave annotations
778         set(h,'Visible','off')
779         % Turn off axes and set square aspect ratio
780         axis off
781         axis image
782         view([90 90])
783         s_mu = sprintf('%0.2f',s.mu(idxV)/cos(s.alpha_TPP_Vec(
784             idxV)));
785         s_a = sprintf('%0.2f',max(M_e,[],'all'));
786         title(['\mu = ',s_mu,' M_{e_{max}}
787             = ',s_a])
788     end
789
790
791 end
792
793
794 function Model3D(obj,x,z,theta0) %center
795     % Plot 3D model of the propeller
796     % INPUT:
797     % — x e z           Airfoil coordinate (in percent of
798         chord)
799     % — center          x coordinate of the center of the
800         propeller (by default is assumed
801         the
802         center of the local chord)
803     % — theta0          Blade pitch at root station
804     %
805     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
806
807     arguments
808         obj
809         x
810         z
811         theta0 = convang(10,'deg','rad');
812     end
813     %
814
815     r_star = obj.r_bar(obj.LAMBDA == 0);
816     r_star = r_star(end); % individuo la stazione dove
817         parte la freccia
818     theta = obj.theta_t*obj.r_bar + theta0;

```

```

815     % plot the first blade
      xz=[x,z];
817     for i=1:obj.n_r
          M=R2(theta(i));
          M=M([1 3],[1 3]);
819     % set pitch
          for j=1:length(x)
821         data_rot(j,:)=M*xz(j,:);
          end
823     % scale airfoil
          x_rot = data_rot(:,1)*obj.c(i);
825          z_rot = data_rot(:,2)*obj.c(i);
          X(:,i) = x_rot - ...
827              mean(x_rot) - ...
              (obj.r_bar(i) - r_star)*obj.R*tan(obj.LAMBDA(i));
829          Z(:,i)=z_rot;
      end
831
      Y=obj.r_bar'.*obj.R;
833      Y=repmat(Y,length(x),1);
      light('Style','local','Position',[1 -1 0]);
835 % plot di una sola pala
      figure
837      s=surf(X,Y,Z,'FaceColor',[0.65 0.65 0.65],'FaceLighting'
          ,...
          'gouraud','EdgeColor','none');
839      daspect([1 1 1])
      camlight right
841 % material dull

843 % plot di tutte le pale
      figure
845      light('Style','local','Position',[1 -1 0]);
          ang_blade=2*pi/obj.N;
847          xyz=[X(:),Y(:),Z(:)];
          ang = 0;
849          hold on
          grid on
851          for i =1:obj.N
              xyz_rot=xyz*R3(ang);
853              X_rot=reshape(xyz_rot(:,1),[length(x),obj.n_r]);
              Y_rot=reshape(xyz_rot(:,2),[length(x),obj.n_r]);
855              Z_rot=reshape(xyz_rot(:,3),[length(x),obj.n_r]);
              s=surf(X_rot,Y_rot,Z_rot,'FaceColor',[0.65 0.65
                  0.65],...
                  'FaceLighting','gouraud','EdgeColor','none');
857              ang = ang + ang_blade;
859          end

861 % Hub(cilinder)
          m=100;
863 % create Hub disc

```

```

865     r_hub = obj.r_bar(1)*obj.R;
      xh=linspace(-r_hub,r_hub,m);
      yh=sqrt((r_hub)^2-xh.^2);
867     xh=[xh,flip(xh)]';
      yh=[yh,-yh]';
869     Xh=repmat(xh,1,3);
      Yh=repmat(yh,1,3);
871     % estrusion of the th Hub Disc
      h_hub = 0.2*r_hub;
873     zh = h_hub*[-1,0, 1 + obj.c(1)*sin(theta(1))/h_hub];
      Zh = repmat(zh,length(xh),1);
875     surf(Xh,Yh,Zh,'FaceColor','k')
      daspect([1 1 1])
877
      %           % Hub(parabolic)
879     %           m=100;
      %           % create Hub disc
881     %           xh=linspace(-obj.r_bar(1),obj.r_bar(1),m) '*
      %           obj.R;
      %           yh=xh;
883     %           [Xh,Yh]=meshgrid(xh,yh);
      %           % estrusion of the th Hub Disc
885     %           k=20;
      %           Zh =-k*Xh.^2-k*Yh.^2+0.2;
887     %
      %           hold on
889     %           surf(Xh,Yh,Zh)
      %           daspect([1 1 1])
891     %           camlight HEADLIGHT
893
      end
      end
895   methods(Access=private)
      function [lam_i,lam,Tc]=initialize_routin(obj,mu,alpha_TPP,
      theta0,options)
897         lam=0;
         res=1;
899         while res > options.toll
             lam_old=lam;
901             Tc = max([0.5*(obj.sigma*obj.Cl_alpha)*...
                (theta0/3*(1+1.5*mu^2) + obj.theta_t/4*(1+mu^2)...
903                 - lam/2),0]);
             lam_i = Tc/( 2*sqrt(mu^2 + lam^2) );
905             if isinf(lam_i)
                 % nella prima iterazione lam=0 e se mu=0 allora
907                 % usiamo la formula dell'induzione in hovering
                 lam_i = sqrt(Tc/2);
909             end
             lam = lam_i +mu*tan(alpha_TPP);
911             res = abs(lam -lam_old);
             end
913         end

```

```

915     end
    end

```

A.3 L'elicottero

Listing A.3:

```

classdef Helicopter
2    % The following class enables the user to computes the main
    % performances
    % of an Helicopter given its main geometrical properties. Some
4    % assumptions were made:
    % → linear geometrical twist along the span
6    % → costant chord along the span
    properties
8
        %
        % _____
10
        % Flight Conditions
        %
        % _____
12
        % cancellare se non la usiamo
        V_max    {mustBePositive, mustBeFinite}    % [m/s]
14        %
        % _____
16
        % Mass
        %
        % _____
18
        % poi vediamo
        fuel_load {mustBePositive, mustBeFinite, mustBeInRange(
            fuel_load,0,1,'exclude-lower')} % [\]
20        W_fuel    {mustBePositive, mustBeFinite}    % [N]
        W_mtow     {mustBePositive, mustBeFinite}    % [N]
22        W_empty   {mustBePositive, mustBeFinite}    % [N]
        %
        % _____
24
        % Geometry
        %
        % _____
26
        % Main Rotor
        MR = Rotor();
        % Tail rotor
28        TR = Rotor();
        % distance between rotor axes

```

```

30      lr      {mustBePositive, mustBeFinite}    % [m]
      %
      _____

32      % Propulsion
      %
      _____

34      engine_number {mustBePositive, mustBeFinite} % [\]
      SFC             {mustBePositive, mustBeFinite} % [lb/( hp*h )]
36      engine_power  {mustBePositive, mustBeFinite} % [W]
      %
      _____

38      % Power
      %
      _____

40      P_req_AUX     {mustBePositive, mustBeFinite} % [W]
      % loss trasmission coefficient
42      eta_t         {mustBePositive, mustBeFinite,...
                      mustBeGreaterThan(eta_t,1)}      % [\]
44      %
      _____

46      % Correction coeffs
      %
      _____

      % fattore di correzione dovuto alla non-uniformita' dell'
      % induzione
48      % sul rotore reale: Pc_i = k*lam_i*T_c
      k_i_MR = 1.2
50      k_i_TR = 1.4
      % Fattore di scorrimento per il calcolo della potenza parassita
52      % Pc0 = sigma*Cd_mean/8*(1+k*mu^2)
      k_mu_MR = 4.7
54      k_mu_TR = 4.7
end
56      properties(SetAccess = private, GetAccess = public)
      % Access denied: there's no way fo the user to change the value
      % of
58      % these variables.

60      % Ambient conditions
      h          {mustBeFinite}                % Altitude
62      rho       {mustBePositive, mustBeFinite}% Air density
      press      {mustBePositive, mustBeFinite}% pressione dell'aria
64      sound_vel {mustBePositive, mustBeFinite}% velocita' del suono
      dell'aria
      temp       {mustBePositive, mustBeFinite}% temperatura dell'aria
66      n_analisi = 0;

```



```

68     % Mass
        M      {mustBePositive, mustBeFinite}
        W      {mustBePositive, mustBeFinite}      % [N]
70     %


---


72     % Analisi
        %


---


74     % Power Analysis
        PA
        % Number of power analisys
76     n_PA = 0;
        % Performance Analysis
78     PerfA
        % Number of power analisys
80     n_PerfA = 0;
end
82 methods
    %% Auxiliary methods
84     % compute ambient conditions
    function obj = ambient(obj,h)
86         obj.h = h;
            [obj.temp, obj.sound_vel, obj.press, obj.rho] = atmosisa(
                obj.h);
88     %     T0 = 288.15;
89     %     mu0 = 1.79e-5;
90     %     obj.mu_visc=mu0*(obj.temp/T0)^1.5*((T0+110)/(obj.temp
+110));
    end
92     %% Solver methods
    function obj = Required_Power(obj,h,V_inf_Vec,Chi,T,f)
94         arguments
            obj
96             h      {mustBeNonnegative,mustBeFinite}
            V_inf_Vec (:,1){mustBeNonnegative,mustBeFinite}
98             Chi     {mustBeFinite}
            T         {mustBePositive,mustBeFinite}
100            f        {mustBePositive, mustBeFinite}
        end
102        V_inf_Vec = sort(V_inf_Vec); flag = sum(V_inf_Vec ~= 0);
        V_inf_Vec = V_inf_Vec(V_inf_Vec ~= 0);
104        %


---


106        % Main Rotor
        obj.MR.h      = h;          % set altitude
108        obj.MR      = obj.MR.ambient(); % compute ambient
            properties
        options      = BEMTset_rotor();

```

```

110     options.k_i = obj.k_i_MR;
111     options.k_mu = obj.k_mu_MR;
112     obj.MR = obj.MR.BEMT_articulated('T',T,V_inf_Vec,Chi,
        f,options);
        s = obj.MR.Analisi_articulated{obj.MR.
            n_analisi_articulated,1};
114     P_MR = s.Pc_vec * obj.rho*pi*obj.MR.R^5*obj.MR.
        omega^3;
        Pi_MR = s.Pci_vec * obj.rho*pi*obj.MR.R^5*obj.MR.
        omega^3;
116     P0_MR = s.Pc0_vec * obj.rho*pi*obj.MR.R^5*obj.MR.
        omega^3;
        P_fus = s.Pc_fus_Vec * obj.rho*pi*obj.MR.R^5*obj.MR.
        omega^3;
118     Q_MR = P_MR/obj.MR.omega;
        % Tail Rotor
120     T_TR = Q_MR/obj.lr;
        Tc_TR = T_TR/( obj.rho*pi*obj.TR.R^4*obj.MR.omega^2
            );
122     v_i_TR = sqrt( -0.5*(V_inf_Vec.^2 + sqrt(V_inf_Vec.^4
        + ...
            4*(T_TR.*0.5/obj.rho/(pi*obj.TR.R^2))^4)));
124     Pi_TR = obj.ki_TR*T_TR*v_i_TR;
        mu_TR = V_inf_vec/obj.TR.omega/obj.TR.R;
126     P0_TR = (obj.TR.Cd_mean*obj.TR.sigma*(1 + obj.
        k_mu_TR*mu_TR.^2)/8)...
        *(obj.rho*pi*obj.TR.R^5*obj.TR.omega^3);
128     P_TR = Pi_TR + P0_TR;
        if flag ~= 0
130         % nel caso in cui venga richiesta una condizione di
            hover
            % To do : MR in hover e controllare BEMT_Artic.. in FF
            % livellato
132         end
134     end
136
138
140
142     % required power for level flight
        function obj = Req_power_level_flight(obj,h,V_inf_Vec,T,f,valIN
            ,PoVc)
            %


---


144         % This function compute the required power curve in the
            % Power-Velocity plane and stores them inside arrays. The
            % arrays have dimensions 1 x n_vel, the dimension of the
            % velocity vector, that can be changed within the class
            % Input:
146
148

```

```

150      % - h      :
151      % - V_inf_Vec :
152      % - T      :
153      % - f      :
154      % - valIN   : a flag to chose the type of problem to
        be solved:
155      %           P -> given available power -> climb
        velocity;
156      %           Vc -> given climb velocity -> required
        power.
157      % - PoVc    : available power OR Climb velocity
        depending
158      %           on the flag ValIn
159      % Output:
160      % - P_induced_MR : induced power by the main rotor
161      % - P_parasite_MR : parasite power of the main rotor
162      % - P_fusolage_MR : parasite power of the fuselage and
        other
163      % aerodynamics exposed surfaces
164      % - P_induced_TR : induced power by the tail rotor
165      % - P_parasite_TR : parasite power of the tail rotor
166      % - P_req_hori    : required power fo forward flight
167      % - Pc_climb      : required power for a given climb
        velocity
168      % - Vc           : climb velocity for a given available
        pwer
169      %


---


170      arguments
171      obj
172      h      {mustBeNonnegative,mustBeFinite}
173      V_inf_Vec (:,1){mustBeNonnegative,mustBeFinite}
174      T      {mustBePositive,mustBeFinite}
175      f      {mustBePositive, mustBeFinite}
176      valIN   {mustBeMember(valIN,{'P','Chi'})}
177      PoVc    (1,1){mustBeFinite}
178      end
179      obj = obj.ambient(h);
180      % required thrust
181      T_TPP    = T;
182      Tc_MR    = T_TPP/( obj.rho*pi*obj.MR.R^4*obj.MR.omega^2 );
183      mu       = V_inf_Vec/( obj.MR.omega*obj.MR.R );
184      ni_i_MR  = sqrt( -0.5*(V_inf_Vec.^2 - sqrt(V_inf_Vec.^4 +
        ...
185      4*(T_TPP*0.5/obj.rho/(pi*obj.MR.R^2))^2) ));
186      lam_i_MR = ni_i_MR/(obj.MR.omega*obj.MR.R);
187      D_fs     = 0.5*obj.rho*V_inf_Vec.^2*f;
188
189      % required power (Main rotor) [W]
190      s.Pc_induced_MR = obj.k_i_MR*Tc_MR*lam_i_MR;

```

```

s.Pc_parasite_MR = obj.MR.sigma*obj.MR.Cd_mean*( 1 + obj.
    k_mu_MR*mu.^2 )/8;
192 s.Pc_fusolage_MR = mu.*( D_fs/T_TPP )*Tc_MR;
s.Pc_req_MR      = s.Pc_induced_MR + s.Pc_parasite_MR + s.
    Pc_fusolage_MR;
194
s.Pi_MR          = s.Pc_induced_MR * obj.rho*pi*obj.MR.R^5*
    obj.MR.omega^3;
196 s.P0_MR          = s.Pc_parasite_MR * obj.rho*pi*obj.MR.R^5*
    obj.MR.omega^3;
s.P_fus_MR       = s.Pc_fusolage_MR * obj.rho*pi*obj.MR.R^5*
    obj.MR.omega^3;
198 s.P_req_MR       = s.Pc_req_MR      * obj.rho*pi*obj.MR.R^5*
    obj.MR.omega^3;

200 Q_MR            = s.P_req_MR/obj.MR.omega;
T_TR             = Q_MR/obj.lr;
202 Tc_TR           = T_TR/( obj.rho*pi*obj.MR.R^4*obj.MR.omega^2 );
ni_i_TR          = sqrt( -0.5*V_inf_Vec.^2 + ...
204             0.5*sqrt( V_inf_Vec.^4 + ( T_TR/(obj.rho*
                pi*obj.TR.R^2) ).^2 ) );
lam_i_TR         = ni_i_TR/(obj.TR.omega*obj.TR.R);
206

% required power (Tail rotor) [W]
208 s.Pc_induced_TR = obj.k_i_TR*Tc_TR.*lam_i_TR;
s.Pc_parasite_TR = obj.TR.sigma*obj.TR.Cd_mean*( 1 + obj.
    k_mu_TR*mu.^2 )/8;
210 s.Pc_req_TR      = s.Pc_induced_TR + s.Pc_parasite_TR;

212 % total required power [W]

214 s.Pi_TR          = s.Pc_induced_TR * obj.rho*pi*obj.TR.R^5*obj
    .TR.omega^3;
s.P0_TR           = s.Pc_parasite_TR * obj.rho*pi*obj.TR.R^5*obj
    .TR.omega^3;
216 s.P_req_TR       = s.Pc_req_TR      * obj.rho*pi*obj.TR.R^5*obj
    .TR.omega^3;

218 s.P_req_hori = (s.P_req_MR + s.P_req_TR + obj.P_req_AUX)*
    obj.eta_t;

220 switch valIN
    case 'P' % available power
222         s.Vc      = (PoVc - s.P_req_hori)/T_TPP;
        s.P_av      = PoVc;
224         s.P_climb = s.Vc*T_TPP;
        s.P_req      = s.P_req_hori;
226     case 'Vc'% climb velocity
        s.Vc          = PoVc;
228         s.P_av      = PoVc*T_TPP + s.P_req_hori;
        s.P_climb     = s.Vc*T_TPP;

```

```

230         s.P_req = s.P_req_hori + max(0,s.P_climb*obj.
            eta_t);
231     end
232
233     s.Pc_climb = s.P_climb/obj.rho*pi*obj.MR.R^5*obj.MR.omega
        ^3;
234     s.lam_i_MR = lam_i_MR;
235     s.lam_i_TR = lam_i_TR;
236     s.V_inf_vec = V_inf_Vec;
237
238     obj.n_PA = obj.n_PA+1;
239     obj.PA{obj.n_PA,1} = s;
240
241 end
242
243 % required power for level flight
244 function obj = Performance_Heli(obj,h,P_av,T,f,fuel_load)
    %
    % -----
245
246     % This function computes the most relevant helicopter
247     % performances for a given available power.
248     % Input:
249     % - h      : flight altitude
250     % - P_av   : available power independent from the
251     %           altitude variation
252     % Output:
253     % - V_max  : maximum velocity (forward level flight)
254     % - V_BE   : best endurance velocity
255     % - V_BR   : best range velocity
256     % - ROC_max: maximum rate of climb
257     % - Endu   : endurance [s]
258     % - Range  : range [m]
259     % - gamma  : climb angle [rad]
260     % The output matrix has the following dimensions:
261     % [OUTPUT] = ih x ip x iw
262     %
    % -----
263
264     arguments
265         obj
266         h      {mustBeNonnegative,mustBeFinite}
267         P_av   {mustBeNonnegative,mustBeFinite}
268         T      {mustBePositive,mustBeFinite}
269         f      {mustBePositive,mustBeFinite}
270         fuel_load {mustBeInRange(fuel_load,0,1)} = 1;
271     end
272     SFC = convforce(obj.SFC,'lbf','N')/( 745.6*3600 );
273
274     % h_init = 500;
275     % Delta_h = 50;

```

```

276 %         for ip = 1:length(P_av)
277 %             for iw = 1:length(T)
278 %                 obj2 = Req_power_level_flight(obj,h_init,
V_inf_Vec,T(iw),f,'P',P_av(ip));
%                 s = obj2.PA{obj2.n_PA,1};
280 %                 while s.Vc > con,vvel(100,'ft/min','m/s')
%                     h = h + Delta_h;
282 %                     V_inf_Vec = linspace(0,(obj.MR.omega*obj.MR.R
) + Delta_V,N);
%                     obj2 = Req_power_level_flight(obj,h(ih),
V_inf_Vec,T(iw),f,'P',P_av(ip));
284 %                     s = obj2.PA{obj2.n_PA,1};
%                     end
286 %                     % Ceiling
%                     s.Ceiling = h;
288 %                 end
%             end
290
292 N = 70;
V_inf_Vec = linspace(0,obj.MR.omega*obj.MR.R,N);
294 for ih = 1:length(h)
295     for ip = 1:length(P_av)
296         for iw = 1:length(T)
297             obj2 = Req_power_level_flight(obj,h(ih),
V_inf_Vec,T(iw),f,'P',P_av(ip));
298             s = obj2.PA{obj2.n_PA,1};
300
301             while (max(s.P_req) < P_av(ip))
302                 Delta_V = 0.05*(obj.MR.omega*obj.MR.R);
303                 V_inf_Vec = linspace(0,(obj.MR.omega*obj.MR
.R) + Delta_V,N);
304                 obj2 = Req_power_level_flight(obj,h(ih),
V_inf_Vec,T(iw),f,'P',P_av(ip));
305                 s = obj2.PA{obj2.n_PA,1};
306             end
307             % Performances
308             sp.P_min(ih,ip,iw) = min(s.P_req);
309             sp.P_BR(ih,ip,iw) = s.P_req(s.P_req'./
V_inf_Vec == min(s.P_req'./V_inf_Vec));
310             if sum(abs( (s.P_req - P_av(ip))/max(s.P_req) )
<1e-2) == 0
311                 % P_av sempre minore di P_req
312                 sp.V_max(ih,ip,iw) = 0;
313             else
314                 sp.V_max(ih,ip,iw) = max(V_inf_Vec(abs(
(s.P_req - P_av(ip))/max(s.P_req) ) <1
e-4));
315             end
316             if sum(s.P_req_hori > P_av(ip)) ==0
317                 sp.ROC_max(ih,ip,iw) = 0;
318                 sp.gamma_max(ih,ip,iw) = 0;

```

```

318         sp.ROD_min(ih,ip,iw) = max(s.Vc);
        sp.gamma_min(ih,ip,iw) = max(atan(s.Vc'./
        V_inf_Vec));
320     else
        sp.ROC_max(ih,ip,iw) = max(s.Vc);
322         sp.gamma_max(ih,ip,iw) = max(atan(s.Vc'./
        V_inf_Vec));
        sp.ROD_min(ih,ip,iw) = 0;
324         sp.gamma_min(ih,ip,iw) = 0;
    end
326     sp.Vd_autorot_min(ih,ip,iw) = min(s.P_req_hori
        /T(iw));
        sp.V_BE(ih,ip,iw) = V_inf_Vec(s.
        P_req == sp.P_min(ih,ip,iw));
328         sp.Endu(ih,ip,iw) = obj.W_fuel*
        fuel_load/( SFC*sp.P_min(ih,ip,iw) );
        sp.V_BR(ih,ip,iw) = V_inf_Vec(s.
        P_req == sp.P_BR(ih,ip,iw));
330         sp.Range(ih,ip,iw) = sp.V_BR(ih,ip,iw)
        *obj.W_fuel*fuel_load/( SFC*sp.P_BR(ih,ip,
        iw) );
    end
332     end
    end
334     obj.n_PerfA = obj.n_PerfA + 1;
        obj.PerfA{obj.n_PerfA,1} = sp;
336     end
338     end
340 end

```


BIBLIOGRAFIA

- [1] Tognaccini, R. (2020), *Lezioni di Aerodinamica dell'Ala Rotante*, -, Napoli.
- [2] Di Giorgio, G. (2009), *Teoria del volo dell'elicottero*, -, Roma.
- [3] Janes, L. A. (2020), *Jane's World Aircrafts*, -, .
- [4] Prouty, R. W. (1985), *Military Helicopter Design Technology* , -,
- [5] Weick, F. E. (1932), *Full scale wind tunnel tests with a series of propellers of different diameters on a single fuselage*, -, Langley Aeronautical Laboratory, NACA.
- [6] Hoerner, S. F. (1965), *Aerodynamic Drag*, -, HOERNER FLUID DYNAMICS, Ca.