



Politecnico di Milano

FACOLTÀ DI INGEGNERIA INDUSTRIALE
Corso di Laurea Magistrale in Ingegneria Aeronautica

TESI DI LAUREA MAGISTRALE

**Modellazione e implementazione numerica
di un motore a turbina per elicottero**

In collaborazione con AgustaWestland

Candidato:

Michele Cappuccilli

Matricola 761325

Relatore:

Prof. Lorenzo Trainelli

Correlatori:

Ing. Riccardo Bianco Mengotti

Dott. Domenico Leonello

Indice

1	Inquadramento e finalità	13
1.1	Propulsione negli elicotteri	13
1.1.1	Componenti tipici del motore per elicotteri	13
1.1.2	Ciclo termodinamico	17
1.1.3	Componenti connessi al propulsore	18
1.2	Finalità del lavoro di tesi	19
2	Stato dell'arte	21
2.1	Modellazione di motori a turbina	21
2.2	Software esistenti	23
3	Fisica del problema e modellazione	27
3.1	Modellazione delle turbomacchine	27
3.2	Proprietà dei fluidi	29
3.3	Componenti del modello	33
3.3.1	Presa d'aria	35
3.3.2	Compressore	37
3.3.3	Plenum	38
3.3.4	Combustore	41
3.3.5	Turbina generatrice di gas	43
3.3.6	Turbina di potenza	44
3.3.7	Modello di carico semplificato	45
3.3.8	Dinamica degli alberi di trasmissione	48
3.4	Dinamica del sistema e metodo di soluzione	49
3.5	Parametrizzazione	51
3.5.1	Esempio di adattamento delle mappe per un generico motore	54
4	Implementazione	59
4.1	Requisiti del codice	59
4.2	Genesi e struttura del codice Matlab	60

4.2.1	Modello γ - Matlab	61
4.2.2	Modello ISO - Matlab	65
4.2.3	Funzioni di supporto	68
4.3	Struttura del codice C	69
4.3.1	Funzione Principale: EngineModel	70
4.3.2	Funzioni di supporto	72
4.4	Metodo di soluzione e di integrazione	73
4.5	Interfaccia grafica	74
4.6	DLL - Dynamic Link Library	75
4.7	GSL - Gnu Scientific Libraries	76
5	Validazione	77
5.1	Confronto con dati sperimentali a regime	77
5.2	Validazione dinamica: risposta ad un gradino di portata di combustibile	79
6	Controllore automatico del propulsore	87
6.1	Architettura e logica di controllo	87
6.1.1	Limiti di impiego del motore	90
6.2	Implementazione	91
6.2.1	Implementazione in linguaggio Matlab	92
6.3	Simulazioni del motore in anello chiuso	94
6.3.1	Simulazione 1 - controllore proporzionale	94
6.3.2	Simulazione 1 - controllore PI	97
6.3.3	Simulazione 1 - controllore PID	97
6.3.4	Simulazione 2 - controllore proporzionale	101
6.3.5	Simulazione 2 - controllore PI	103
6.3.6	Simulazione 2 - controllore PID	104
7	Simulazione di una manovra	107
7.1	Illustrazione della manovra e software impiegati	107
7.2	Confronto utilizzando alcuni parametri del motore GE t700	109
7.3	Confronto utilizzando parametri identificati	112
8	Conclusioni e sviluppi futuri	117
8.1	Conclusioni	117
8.2	Sviluppi futuri	118

Elenco delle figure

1.1	Schema semplificato di motore turbo-albero connesso a trasmissione e rotore.	14
1.2	Schema di ciclo joule Brayton ideale nel piano T-S.	17
1.3	Ciclo Joule-Brayton reale nel piano T-S.	18
2.1	Il motore General Electric t700.	22
3.1	Variazione della costante dei gas R in funzione della temperatura per i gas combustibili di kerosene.	30
3.2	Calore specifico dei gas ottenuti dalla combustione di kerosene in funzione della temperatura per diversi rapporti aria/combustibile.	31
3.3	Rapporto fra calori specifici del kerosene in funzione di temperatura e rapporto aria/combustibile.	32
3.4	Rapporto fra calori specifici k dell'aria e dei gas combustibili di kerosene con rapporto aria/combustibile di $\frac{1}{35}$ ottenuto con il modello di Guha.	34
3.5	Calore specifico c_p dell'aria e dei gas combustibili con rapporto aria/combustibile di $\frac{1}{35}$, ottenuto con il modello di Guha per un ampio intervallo di temperature.	34
3.6	Calore specifico di aria e gas combustibili ottenuto con modello di Walsh, con FAR=1/35.	35
3.7	Schema del motore e flusso di informazioni fra i vari componenti.	36
3.8	Pala di rotore semplificata.	46
3.9	Potenza richiesta dal carico modellato in funzione della velocità di rotazione n_2 confrontata con i valori di potenza e velocità di rotazione nominali per il motore GE t700.	48
3.10	Diagramma di flusso per il processo di generazione e adattamento delle mappe delle turbomacchine.	51
3.11	Rapporti di compressione/espansione normalizzati.	54
3.12	Mappa normalizzata della portata d'aria elaborata dal compressore (g_2) in funzione del regime di rotazione n_1	55

3.13	Mappa normalizzata del rapporto di espansione della turbina di potenza in funzione della temperatura in uscita dalla turbina HPT.	55
3.14	Mappa dei $\beta_{compressore}$ e β_{HPT} adattata ad un generico motore.	56
3.15	Mappa (adattata ad un ipotetico motore) della portata d'aria elaborata dal compressore (g_2) in funzione del regime di rotazione n_1	57
3.16	Mappa (adattata ad un ipotetico motore) del rapporto di espansione della turbina di potenza in funzione della temperatura in uscita dalla turbina HPT.	57
4.1	Schema del codice γ implementato in Matlab.	62
4.2	Mappa della temperatura in uscita dal compressore in funzione del regime di rotazione n_1 nel modello γ per il motore GE t700.	63
4.3	Schema delle funzioni implementate nel <i>codice ISO</i>	67
4.4	Funzione EngineModel.c : ingressi e uscite.	72
5.1	Andamento dell'ingresso di portata di combustibile a gradino.	83
5.2	Risposta ad un ingresso a gradino di portata di combustibile: confronto del regime di rotazione n_1 con i dati del report Ballin - 1988.	84
5.3	Risposta ad un ingresso a gradino di portata di combustibile: confronto della pressione all'uscita dal compressore p_{21} con i dati del report Ballin - 1988.	84
5.4	Risposta ad un ingresso a gradino di portata di combustibile: confronto della temperatura T_3 in uscita dalla camera di combustione con i dati del report Ballin - 1988.	85
5.5	Risposta ad un ingresso a gradino di portata di combustibile: confronto della temperatura T_4 in uscita dalla turbina di alta pressione con i dati del report Ballin - 1988.	85
6.1	Architettura di un sistema di controllo di un motore a turbina per elicotteri.	88
6.2	Logica di controllo del regolatore PID implementato all'interno del FADEC.	90
6.3	Sistema motore-controllore in anello chiuso.	92
6.4	Struttura del controllore FADEC nell'implementazione in linguaggio Matlab.	93
6.5	Simulazione 1 con controllore P. Velocità di rotazione dei due alberi e variabile di controllo g_b	96

6.6	Simulazione 1 con controllore Proporzionale. Velocità angolare dell'albero della turbina di potenza. Confronto con il riferimento da inseguire.	96
6.7	Inseguimento del riferimento di velocità di rotazione costante con controllore PI.	98
6.8	Simulazione 1 con controllore PI. Andamento dei due stati meccanici e della variabile di controllo durante il primo secondo di simulazione.	98
6.9	Velocità di rotazione n_1, n_2 e portata di combustibile g_b durante l'inseguimento del riferimento costante $n_{2_{ref}}$ con controllore di tipo PI.	99
6.10	Simulazione 1 con controllore PID. inseguimento del riferimento sulla velocità di rotazione n_2	99
6.11	Simulazione 1 con controllore PID. Velocità di rotazione e portata di combustibile durante la simulazione.	100
6.12	Simulazione 1 con controllore PID. Particolare del primo secondo di simulazione.	100
6.13	Inseguimento del riferimento da parte del sistema motore - controllore in anello chiuso. Simulazione 2. Controllore solo proporzionale.	102
6.14	Velocità di rotazione dei due alberi e flusso di carburante in camera di combustione durante la simulazione 2 di inseguimento di un riferimento variabile di n_2 con motore controllato da un regolatore proporzionale.	102
6.15	Simulazione 2 con controllore PI. Inseguimento del riferimento variabile $n_{2_{REF}}$	103
6.16	Simulazione 2 con controllore PI. Andamento dei due stati meccanici n_1, n_2 e della variabile di controllo g_b	104
6.17	Simulazione 2 con controllore PID. Inseguimento del riferimento variabile di velocità di rotazione n_2	105
6.18	Simulazione 2 con controllore PID. Storia temporale dei due stati meccanici n_1 ed n_2 e della variabile di controllo g_b durante la simulazione.	106
7.1	Storia temporale della portata di combustibile g_b durante la simulazione effettuata in Flighlab e poi riprodotta con il modello ISO.	109
7.2	Confronto fra la velocità di rotazione n_1 calcolata dal modello ISO e quella ottenuta dalla simulazione in Flightlab. Si riporta anche il valore nominale per il motore considerato. Parametri incogniti derivati dal motore GE t700.	110

7.3	Confronto della temperatura T_4 dei gas fra le due turbine calcolata con il modello ISO e il software Flightlab. Parametri incogniti derivati dal motore GE t700.	111
7.4	Potenza espressa dal modello ISO confrontata con quella espressa dal modello FlightLab. Per i parametri incogniti si è fatto ricorso a quelli del motore GE t700.	112
7.5	Confronto della temperatura T_4 . Modello ISO con parametri modificati e simulazione FlightLab.	114
7.6	Confronto sulla velocità n_1 . Simulazione effettuata con software FlightLab e con modello ISO con parametri modificati.	114
7.7	Confronto sulla potenza erogata all'interfaccia con il carico. Simulazione condotta con software FlightLab e con modello ISO con parametri modificati.	115

Elenco delle tabelle

3.1	Coefficienti dei polinomi per il calcolo di k e c_p dei gas combusti nel modello di Walsh [1].	31
3.2	Dati ricavati da test NASA su motore GE t700 (prima parte).	51
3.3	Dati ricavati da test NASA su motore GE t700 (seconda parte).	51
5.1	Errori percentuali dei due modelli rispetto al caso 2 di trim sperimentale.	78
5.2	Errori percentuali dei due modelli rispetto al caso 3 di trim sperimentale.	79
5.3	Errori percentuali dei due modelli rispetto al caso 4 di trim sperimentale.	79
5.4	Errori percentuali dei due modelli rispetto al caso 5 di trim sperimentale.	80
5.5	Errori percentuali dei due modelli rispetto al caso 6 di trim sperimentale.	80
5.6	Condizioni iniziali di temperatura nelle varie sezioni del motore prima dell'ingresso a scalino di portata di combustibile.	81
5.7	Condizioni iniziali delle portate prima dell'ingresso a scalino di portata di combustibile.	82
5.8	Condizioni iniziali delle pressioni prima dell'ingresso a scalino di portata di combustibile.	82
6.1	Taratura dei guadagni con il metodo di Ziegler e Nichols.	89
6.2	Condizioni iniziali della simulazione 1 di motore controllato in anello chiuso da controllore proporzionale.	95
7.1	Dati del motore P&W PT6C-67C.	108

Sommario

In questo lavoro viene proposto un modello numerico di motore per elicottero semplificato, flessibile e adattabile a più applicazioni attraverso il minore numero di parametri possibile. Tale modello è stato implementato in diverse versioni prima in linguaggio Matlab e poi in linguaggio C attraverso lo sviluppo di una DLL, allo scopo di essere poi incluso in codici di simulazione e di meccanica del volo. La modellazione termodinamica è stata validata confrontando i risultati ottenuti da alcune simulazioni con dei dati sperimentali disponibili in letteratura. È stato poi affiancato al modello di motore anche un controllore che ne regoli il funzionamento, seguendo una logica simile a quella di un FADEC. Infine si è validato il codice simulando una manovra tipica e confrontando i risultati con quelli di riferimento.

The object of this thesis work is the development of a simplified, flexible, customizable helicopter engine numerical model, adaptable to several existing engines by tuning a small set of parameters. The model has been implemented both in Matlab and C computing languages, resulting in a DLL that can be easily linked with Flight Mechanics simulation codes. The thermodynamic model has been validated comparing the simulation results with available experimental data. An automatic controller for the engine has been developed and integrated to reproduce typical FADEC logics. Finally, the model has been tested using real inputs from a typical manoeuvre and comparing the results with reference data.

Capitolo 1

Inquadramento e finalità

1.1 Propulsione negli elicotteri

La propulsione tramite motori a turbina a gas è quella preponderante negli elicotteri di dimensioni medie e grandi. Il motore a turbina per elicotteri è di tipo *turboalbero* (turboshaft): tutta la potenza che si riesce a estrarre dal fluido di lavoro viene trasferita attraverso ad un'albero di trasmissione che muoverà i carichi ad esso connessi; i carichi sono costituiti principalmente dal rotore principale, dal rotore di coda e da alcune utenze secondarie degli impianti del velivolo. I motori turboalbero sono simili ai propulsori turboelica utilizzati in velivoli ad ala fissa, ma a differenza di questi ultimi non è necessario progettare il motore in modo che il flusso di gas di scarico conservi all'uscita dal motore un'energia cinetica importante perchè non se ne sfrutta il contributo di reazione nella spinta, bensì tutta la potenza che fornisce il propulsore viene trasferita tramite l'albero di potenza alla trasmissione e quindi alle utenze.

1.1.1 Componenti tipici del motore per elicotteri

I componenti di un tipico motore a turbina per elicotteri sono riportati in figura 1.1, essi sono:

- la presa d'aria,
- il compressore,
- la camera di combustione,
- la turbina generatrice di gas,
- la turbina di potenza.

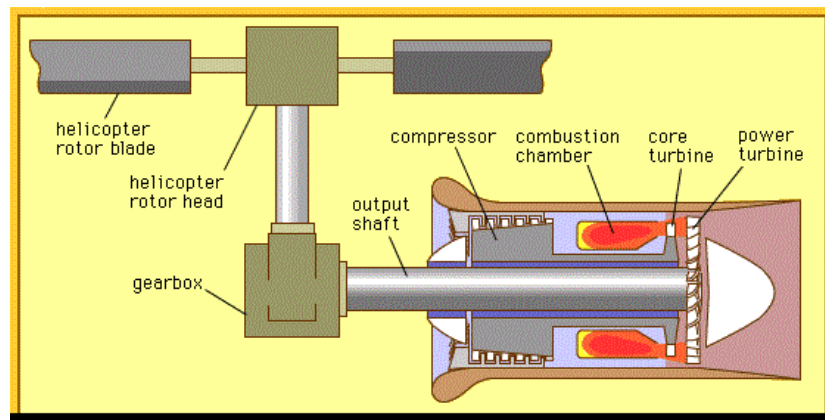


Figura 1.1: Schema semplificato di motore turbo-albero connesso a trasmissione e rotore.

La *presa d'aria* è il componente che cattura l'aria esterna per convogliarla all'interno del motore. Essa può essere di due tipi:

- *statica*: se l'aria fluisce all'interno del motore solo per l'azione di aspirazione effettuata dal propulsore.
- *dinamica*: se l'aria che fluisce nel propulsore lo fa anche sotto l'effetto della velocità dovuta al moto dell'elicottero.

Gli obiettivi di una buona presa d'aria dinamica sono di:

- aumentare la pressione, per evitare parte del lavoro di compressione al compressore;
- realizzare un flusso uniforme a monte del compressore;
- ottenere minime perdite di pressione totale, e quindi minimizzare gli attriti;
- provocare la minima resistenza aerodinamica esterna;
- avere minimo peso e lunghezza.

Dalla presa d'aria, l'aria è convogliata verso il *compressore*, dove l'aria subisce una compressione che la porta ad una pressione e temperatura più elevate. Il compressore è una macchina operatrice pneumofora, ovvero una macchina che innalza la pressione di un gas mediante l'impiego di energia meccanica. I compressori utilizzati nei motori per elicotteri sono principalmente di due tipi:

- *assiale*, dove l'aria fluisce parallelamente rispetto all'asse longitudinale del motore. I compressori assiali sono composti di più stadi, ognuno formato da un elemento solidale al supporto del motore, lo statore, e da un elemento rotante, il rotore.
- *centrifugo*, in cui l'aria, dopo essere stata raccolta dalle palette di una girante, è compressa per azione della forza centrifuga associata al moto di rotazione delle palette del compressore.

In molti motori elicotteristici si realizzano soluzioni ibride in cui alcuni stadi del compressore sono realizzati tramite moduli assiali, mentre altri con moduli centrifughi.

L'aria in uscita dal compressore viene poi diretta verso la *camera di combustione* attraverso un *diffusore* atto a rallentare il flusso e creare le condizioni migliori per l'accensione e la stabilizzazione della fiamma. In camera di combustione l'aria è miscelata con il combustibile (generalmente kerosene), e la miscela risultante viene bruciata liberando l'energia necessaria all'alimentazione del propulsore. La conformazione delle camere di combustione per elicotteri è generalmente di tipo anulare con flusso inverso, ovvero essa corre per tutta la circonferenza del propulsore e la direzione del flusso d'aria è ruotata di 180° , consentendo un minore ingombro del propulsore. L'aria in arrivo dal diffusore del compressore viene divisa in due flussi:

- un flusso primario dove l'aria, miscelata con il combustibile all'interno del tubo di fiamma, raggiunge temperature dell'ordine dei 2500°C ;
- un flusso secondario, in cui l'aria è più fredda e che consente di non danneggiare i materiali con cui è realizzata la camera di combustione.

Il combustibile viene iniettato in camera di combustione attraverso degli iniettori che si dividono in primari, i quali consentono la combustione nelle normali situazioni operative, e secondari, che provvedono attraverso degli igniter anche all'accensione del motore. La combustione avviene con grande eccesso di aria, generalmente in un propulsore per elicotteri il rapporto aria/combustibile è dell'ordine di 35:1. All'uscita dalla camera di combustione i gas combusti prodotti hanno una temperatura dell'ordine di 1500 K.

Successivamente, i gas combusti provenienti dalla camera di combustione vengono inviati nella prima turbina, la *turbina di alta pressione* (o turbina generatrice di gas [2]), che ha il compito di estrarre dal fluido l'energia necessaria ad alimentare il compressore. La turbina generatrice di gas e il compressore sono collegati rigidamente da un albero, che generalmente viene indicato come albero della turbina generatrice di gas. La turbina è composta

da più stadi a loro volta formati da uno statore e un rotore. Lo statore ha i compiti di deflettere il flusso così che l'aria investa ogni paletta del rotore con l'incidenza adeguata, e di convertire parte dell'energia del flusso ad alta pressione ed alta temperatura in energia cinetica. Ogni rotore è composto da un albero e di alcune ruote sulle quali sono fissate le palette, le quali sfruttano l'energia cinetica del flusso per mantenere in rotazione l'albero.

A valle della turbina generatrice di gas si trova la *turbina di potenza* (o turbina libera), anch'essa a uno o più stadi, che sfrutta l'energia residua dei gas di scarico per fornire la potenza necessaria a mantenere in rotazione l'albero di potenza. Essendo la turbina di potenza collegata alla trasmissione e quindi al rotore, che opera in regimi di rotazione pressoché costanti, essa deve mantenere una velocità di rotazione assegnata, nonostante la potenza del carico ad essa collegata sia variabile in funzione delle condizioni di volo e dei comandi. Il mantenimento della velocità di rotazione della turbina di potenza è possibile grazie alla variazione del numero di giri della turbina generatrice di gas. Il collegamento fra le due turbine è solo di tipo pneumatico, in quanto non vi è nessun organo meccanico che le colleghi. Le turbine lavorano a regimi di rotazione compresi fra i 20000 e i 60000 giri/minuto. La velocità di rotazione della turbina generatrice di gas è indicata generalmente con N_1 (o N_g), mentre quella della turbina di potenza con N_2 (o N_f o N_{PT}). A valle delle turbine i gas combusti sono scaricati esternamente tramite un ugello.

Il propulsore aeronautico è l'unica fonte di potenza nella maggior parte dei velivoli, in special modo negli elicotteri, dove il requisito di leggerezza è molto stringente: per questo motivo oltre a sottrarre potenza all'albero tramite generatori elettrici/alternatori che alimentino l'impianto elettrico, è necessario anche spillare una quantità della portata d'aria elaborata dal compressore per alimentare l'impianto pneumatico e il sistema di controllo ambientale (ECS) per la pressurizzazione, se è prevista, e il condizionamento. Generalmente tale spillamento viene effettuato nei primi stadi del compressore, quelli a più bassa pressione.

La maggior parte dei motori per elicotteri è dotato di *due alberi*: il primo (*albero della turbina generatrice di gas*) collega il compressore alla turbina di alta pressione (HPT: High Pressure Turbine o turbina generatrice di gas), il secondo (*albero della turbina di potenza*) è quello di potenza che collega meccanicamente la seconda turbina (PT: Power turbine o turbina di potenza o turbina libera) al carico tramite la trasmissione. Il collegamento fra la prima e la seconda turbina, come già detto, è solo di tipo pneumatico: ciò comporta la possibilità per il motore che i due alberi ruotino a velocità diverse. Il secondo albero infatti, essendo collegato al rotore tramite la trasmissione, deve girare a una velocità pressoché costante (variabile al massimo di pochi punti

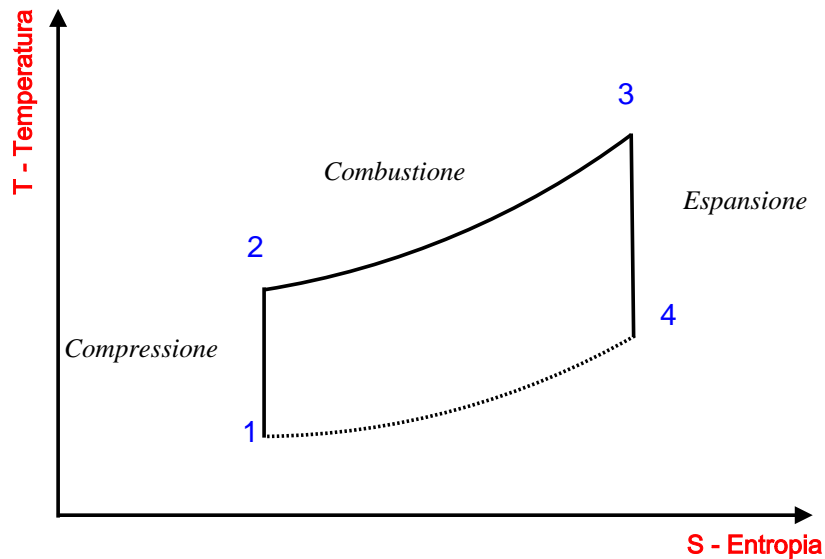


Figura 1.2: Schema di ciclo joule Brayton ideale nel piano T-S.

percentuali in alcune particolari situazioni di volo), mentre l'albero del compressore e della turbina generatrice di gas, oltre a girare a velocità più elevate (nell'ordine dei 60 000 RPM), deve essere in grado di variare la sua velocità per mantenere costante quella del secondo albero a fronte di variazioni della potenza richiesta dal carico.

1.1.2 Ciclo termodinamico

Analizzando nel dettaglio il ciclo termodinamico del motore a turbina per applicazioni aeronautiche, che è il ciclo Joule-Brayton, esso si compone di una compressione, una combustione, un'espansione, che avvengono rispettivamente nel compressore, in camera di combustione e in turbina. Essendo il ciclo Joule-Brayton un ciclo aperto, poichè il fluido di lavoro è l'aria dell'atmosfera, esso si chiude idealmente al di fuori del motore attraverso il ripristino delle condizioni dell'aria da parte dell'atmosfera. Nel caso ideale (figura 1.2) le compressioni/ espansioni sono considerate isoentropiche, ovvero senza aumento di entropia durante la trasformazione, mentre si assume che la combustione avvenga senza perdite energetiche. Nella realtà, e dunque nell'implementazione di un modello fedele ad essa, si considerano delle trasformazioni reali nelle turbomacchine, che non sono più isoentropiche, e un processo di combustione affetto da perdite di pressione (figura 1.3).

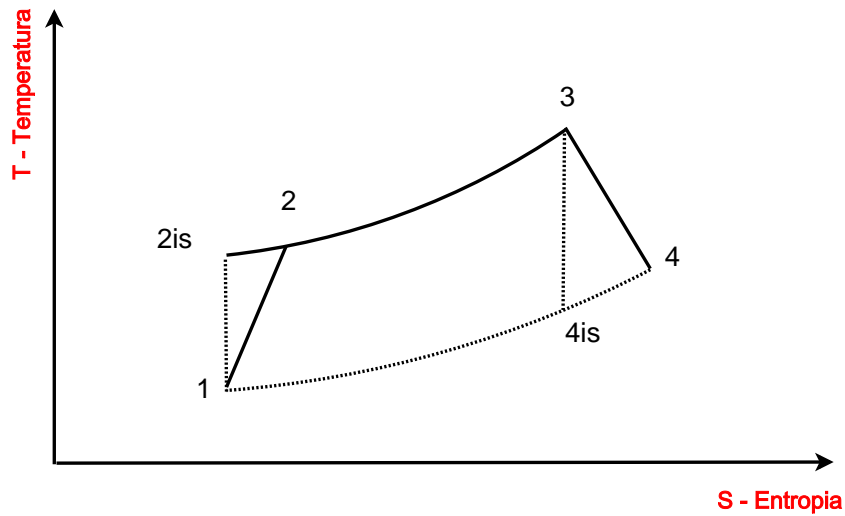


Figura 1.3: Ciclo Joule-Brayton reale nel piano T-S.

1.1.3 Componenti connessi al propulsore

L'importanza di garantire la potenza necessaria al rotore e la necessità di mantenere la velocità del rotore costante in ogni situazione impone ai propulsori per elicotteri l'utilizzo di un sistema di controllo che ne governi il funzionamento; tale sistema di controllo, dapprima realizzato con tecnologie analogiche o miste analogico-digitali, oggi è costituito dal FADEC (Full Authority Digital Engine Control), sistema di controllo digitale che governa tutte le funzionalità del propulsore e che ha autorità assoluta sul suo funzionamento. L'implementazione di un FADEC reale è un'attività complessa e costosa che viene effettuata molto spesso dalle stesse ditte produttrici dei propulsori, che per ragioni di segreto industriale non divulgano le logiche che sono alla base dell'implementazione del controllore.

Nelle applicazioni elicotteristiche di elicotteri medio/grandi i motori installati sono usualmente più di uno, ed è quindi fondamentale l'accoppiamento fra più propulsori e la loro gestione. Alcuni degli studi più tipici di dinamica del volo di elicotteri sono le condizioni OEI (One Engine Inoperative), o la cosiddetta Categoria A (si veda a riguardo il riferimento [3]).

Numerosi sono i sistemi e gli impianti che del motore regolano il funzionamento (gestione delle temperature, afflusso del carburante, lubrificazione, anti-ghiaccio, sistemi di accensione), per un'analisi accurata di questi aspetti si è fatto riferimento all'ottimo [4], che tratta con rigore i sistemi connessi ai moderni motori turbofan commerciali, ma nell'implementazione del modello non è stato ritenuto necessario entrare nel dettaglio di questi componenti.

1.2 Finalità del lavoro di tesi

La simulazione della dinamica del volo di un aeromobile, specialmente se esso è ad ala rotante, è profondamente influenzata dal sistema propulsivo e dal suo funzionamento, in particolar modo durante i transitori e nel volo manovrato. La disponibilità di un modello di propulsore che sappia riprodurre fedelmente la risposta di un motore reale all'interno di un simulatore per lo studio della meccanica del volo è un aspetto fondamentale.

Questa tesi si pone l'obiettivo di mettere a punto un modello di motore *semplificato, generale e adattabile* con pochi aggiustamenti a vari propulsori esistenti. Il codice che ne risulta sarà impiegato in simulatori di meccanica del volo, sia all'interno del Dipartimento di Ingegneria Aerospaziale del *Politecnico di Milano* che presso la divisione *Flight Mechanics* della ditta *AgustaWestland*. Dovrà essere possibile adattare il codice al maggior numero di propulsori a turbina per elicottero, così da rendere agevole il confronto tra l'installazione di diversi motori su di una stessa macchina. Il codice dovrà anche essere eseguito all'interno di simulatori di meccanica del volo e quindi sarà necessario che soddisfi un requisito *real-time* (non stretto) e sarà necessario mettere a punto un'interfaccia grafica per rendere agevole la prova del codice in modalità stand-alone.

Riassumendo gli obiettivi del codice saranno:

- adattabilità a molti motori tramite pochi parametri fondamentali, facilmente reperibili;
- requisito *real-time* (non stretto);
- scrittura in linguaggio C per integrazione in sistemi più estesi (Simulatori di MdV (Meccanica del volo), ecc.);
- interfaccia grafica per prove di motore stand-alone.

Capitolo 2

Stato dell'arte

2.1 Modellazione di motori a turbina

La disponibilità di grande potenza di calcolo ha reso possibile in tempi recenti il proliferare di studi sulla simulazione del funzionamento di motori e generatori a turbina a gas, sia per il progetto del propulsore/generatore, sia per l'integrazione dello stesso all'interno di un sistema più complesso. I due bisogni, quello dello studio del propulsore in sé e quello dell'integrazione all'interno di un sistema più vasto (ad esempio un velivolo o un impianto industriale), hanno per forza di cose una diversa esigenza di dettaglio nella simulazione: nel primo caso si deve essere in grado di riprodurre tutti i fenomeni inerenti il motore con grande precisione così da poter vagliare le differenti scelte progettuali, nel secondo invece la richiesta è quella di riprodurre il funzionamento del componente propulsore/generatore agli effetti esterni, ovvero nell'interazione che esso ha con il sistema principale, sia esso aeroplano, elicottero o impianto industriale.

Il nostro lavoro si riconduce alla seconda tipologia di simulazione, in particolare la modellazione del motore a turbina che abbiamo messo a punto ha lo scopo di fornire ad un qualunque programma di simulazione di meccanica del volo i parametri del motore di interesse per la simulazione della dinamica del velivolo.

La grande maggioranza dei lavori riguardanti simulazioni di motori a turbina per elicotteri, citando il lavoro di Ballin [5] e quello di Duyar e Gu [6], basano i loro modelli sulle prove effettuate dai laboratori NASA Lewis negli anni '80 sul motore General Electric t700 (figura 2.1), largamente diffuso e utilizzato, tra gli altri sull'elicottero Sikorsky UH-60 (due unità per elicottero), sull'AgustaWestland AW149 (due unità per elicottero) e sull'AgustaWestland AW101 (tre unità per elicottero).

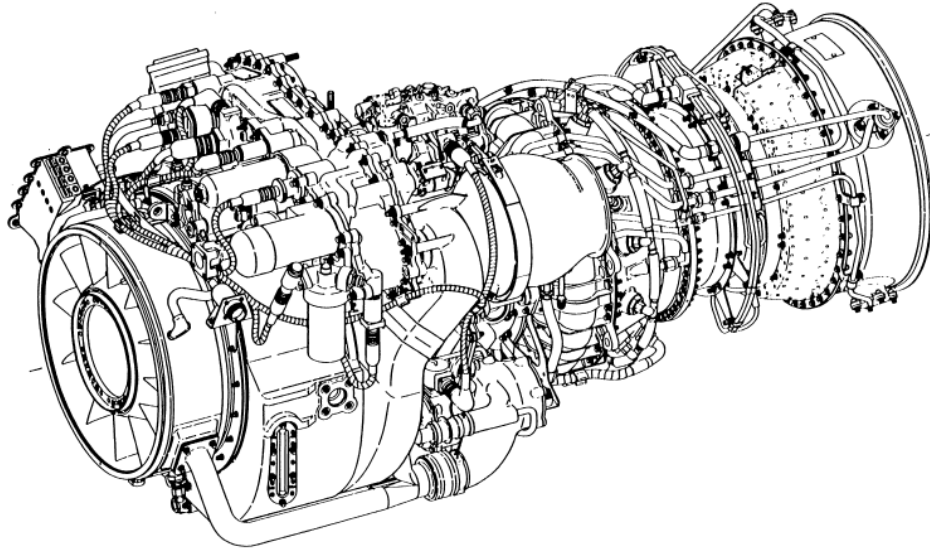


Figura 2.1: Il motore General Electric t700.

Un lavoro di riferimento per ogni simulazione di motore per elicotteri è il memorandum tecnico [5], dove sulla base di esperimenti compiuti dalla NASA su un motore General Electric t700, si mette a punto un modello semplificato *real-time* e lo si confronta con i dati sperimentali rilevati al banco. Il modello messo a punto è di tipo non lineare: tramite la conoscenza dei valori degli stati termodinamici del motore per vari casi di trim sperimentali, vengono dapprima calcolati off-line dei punti di funzionamento che poi il modello utilizzerà durante la simulazione attraverso una logica di adattamento non lineare. A questo memorandum si farà riferimento in fase di validazione perché esso contiene la gran parte dei dati sperimentali a cui si riferisce la maggioranza dei lavori successivi: tali dati sperimentali saranno quelli con cui ci si confronterà nelle simulazioni di validazione statica e dinamica della parte termodinamica del modello. In [5] è anche presente un implementazione del controllore del motore, così da poterne simulare il comportamento anche in anello chiuso insieme ad un modello dinamico semplificato dell'elicottero UH-60.

Prosecuzione del lavoro di Ballin ([5]) è il report [6], dove, a partire dal modello di Ballin, viene messo a punto un simulatore semplificato, linearizzato e in anello aperto del solito motore GE t700. Collegando in una formulazione agli stati diversi punti di funzionamento del modello trattato in [5], viene quindi generato un sistema lineare che rende più agevole l'implementazione e la sua risoluzione in *real-time*.

Nell'articolo di Camporeale e Fortunato [7] si trova un approccio più re-

cente al problema: il lavoro si pone lo scopo di mettere a punto in ambiente *Simulink*® un codice modulare che consenta la simulazione di impianti industriali di turbine a gas. Il modello che viene messo a punto dagli autori è del tipo “aerotermico”, similmente a quanto fatto successivamente anche nella tesi di Kocer [8] e nel recente articolo di Kocer [9], con una rappresentazione concentrata delle turbomacchine. La modellazione del componente *combustore* da noi implementata è concorde con quella utilizzata in tale lavoro, ovvero quella di un puro accumulatore di energia, delegando il bilancio di massa ad un componente “plenum” che non trova riscontro in un componente fisico.

In [9] e nella tesi [8] si ritrova un approccio moderno molto simile a quello di Camporeale e al.

2.2 Software esistenti

I software disponibili per la simulazione di motori a turbina a gas si possono raggruppare in due tipologie:

- software di simulazione del motore *real-time*;
- banchi prova virtuali non *real-time* (*engine deck*);

I **software non real-time** consentono una maggiore accuratezza della modellazione e nella discretizzazione degli stati termodinamici del motore. Questi software vengono spesso realizzati dal produttore del propulsore sulla base di prove sperimentali ottenute al banco, e la loro utilità è quella di avere a disposizione un modello molto accurato che può sostituire il vero motore in molte applicazioni dove l'utilizzo del propulsore reale sarebbe impossibile per questioni di sicurezza o di costo:

- prove con propulsore “in the loop”;
- prove pericolose che potrebbero danneggiare il propulsore;
- verifica dei sistemi di controllo e di diagnostica;

Fra i software esistenti in questo campo, uno dei più conosciuti è *GSP: Gas turbine Simulation Program* ([10]), messo a punto dall’NLR (Nationaal Lucht- en Ruimtevaartlaboratorium: centro di ricerca aeronautico olandese) e scaricabile liberamente nella sua versione dimostrativa. Esso consente la simulazione di una vasta gamma di motori/generatori a turbina a gas, essendo un codice modulare con un’interfaccia grafica abbastanza intuitiva. Il codice sorgente è scritto in Delphi, linguaggio orientato agli oggetti, e non

è reso disponibile nella versione liberamente scaricabile. GSP è un codice completo che consente di simulare anche l'usura dei componenti e le perdite di installazione del motore. Durante l'uso di questo programma abbiamo registrato una difficoltà a convergere in situazioni lontane dal funzionamento nominale, ed è quindi stato impossibile utilizzare tale programma come confronto nelle simulazioni eseguite, se non in quelle in condizioni nominali, dove erano disponibili anche dei dati sperimentali, che sono stati preferiti a quelli generati da GSP. Va riconosciuto a questo programma una intuitiva interfaccia grafica e una buona documentazione disponibile on line. Esso è stato infatti impiegato in numerosi progetti europei di ricerca.

I **software real-time** sono utili invece per un numero di situazioni dove è richiesto che il funzionamento avvenga nei tempi caratteristici del fenomeno reale, rinunciando magari a parte dell'accuratezza garantita dagli engine deck:

- osservatore del propulsore per i sistemi di bordo del velivolo;
- simulatore di motore da inserire in simulatori di volo;

Un software di questa tipologia è *TERTS: Turbine Engine Real-Time Simulator* ([11]), messo a punto dallo stesso team dell'NLR di Amsterdam che ha ideato anche GSP. TERTS è in effetti un software derivato da GSP, che rispetta però un requisito *real-time*. Esso è implementato in ambiente Matlab-Simulink.

Gran parte dei software *real-time* per la simulazione di motori implementano un modello lineare ottenuto dall'identificazione di un sistema. Molto spesso vengono implementati modelli "lineari a tratti" dove una serie di relazioni lineari viene utilizzata per coprire lo spazio di stati altamente non lineare [6]. Modelli lineari separati vengono determinati per differenti condizioni operative (per esempio diverse velocità di rotazione). Questo tipo di approccio è largamente diffuso per l'utilizzo in simulatori di volo o per il progetto di sistemi di controllo. Per analisi più accurate di ogni nuovo effetto, è comunque preferibile un approccio termodinamico che si attenga maggiormente alla fisica del problema, così da avere un modello di base che possa essere impiegato in una maggiore varietà di prove. Il set di equazioni non lineari che traduce generalmente i principi fisici di conservazione della massa, bilancio di energia e conservazione del momento angolare, viene poi risolto in diversi modi, fra i metodi di soluzione più utilizzati in letteratura ricordiamo: il metodo di Newton-Raphson, la generazione di Jacobiani in prossimità di alcuni punti di funzionamento noti, e il metodo di Broyden per l'aggiornamento dello Jacobiano. Il calcolo dello Jacobiano, se da una parte consente una rapida convergenza e un'efficace implementazione del codice *real-time*, in

quanto può essere effettuato off-line prima della simulazione, dall'altra parte può portare a un insieme di problemi quali dei punti di stazionarietà, mal condizionamento numerico della matrice Jacobiana o problemi di oscillazione intorno a un punto di equilibrio. Il requisito *real-time* di avere un tempo massimo di esecuzione del codice porta nella maggior parte di questi software a un numero massimo di iterazioni concesse (troncamento delle iterazioni): dopo un limitato numero di passi temporali (massimo tempo di esecuzione all'interno di un passo temporale), l'iterazione viene fermata e l'accuratezza del calcolo viene accettata per quella che è. All'iterazione successiva si assume che il modello sia giunto a convergenza. Questa assunzione è ragionevole se la simulazione del motore coinvolge transitori con grandi gradienti che durano brevi periodi. Negli intervalli fra uno di questi transitori veloci e l'altro si assume che il propulsore lavori quasi stazionariamente, cosicché ogni errore residuo nelle equazioni venga eliminato.

Capitolo 3

Fisica del problema e modellazione

La modellazione del problema è avvenuta tenendo sempre presente l'obiettivo di un modello semplice, che riproduca i fenomeni fisici di interesse in modo completo, senza però entrare troppo nel dettaglio del comportamento dinamico sia dal punto di vista della gasdinamica che della chimica, cosa che avrebbe richiesto un impegno eccessivo nella modellazione e nell'esecuzione del codice, che nelle intenzioni del lavoro deve poter avvenire in *real-time*.

La determinazione del modello è stata pensata considerando come base di partenza il motore General Electric t700 al quale si riferisce gran parte della letteratura in materia (es. [5, 6]), così da potersi confrontare su casi test il più possibile vicini a quelli disponibili. Si è posta attenzione all'estendibilità del modello a propulsori diversi enfatizzando il requisito di portabilità del modello, rendendolo dipendente da alcuni parametri fisici caratterizzanti il motore possibilmente reperibili in letteratura per nuovi e differenti motori.

3.1 Modellazione delle turbomacchine

In accordo con la letteratura più recente [7, 8, 9], si è scelto di modellare le turbomacchine (compressore e turbine) come dei componenti istantanei, considerando la loro dinamica interna talmente veloce da non influire sulle dinamiche più lente del sistema complessivo. Le trasformazioni che avvengono all'interno delle turbomacchine sono state modellate considerando la corrispettiva espansione o compressione isoentropica e correggendola poi con un rendimento per determinare il punto finale della trasformazione reale. La determinazione del *rapporto di compressione o di espansione* per ogni turbomacchina è stato valutato a partire da *mappe* che parametrizzano queste

grandezze in funzione di un parametro caratteristico che identifica il regime di funzionamento del motore nella condizione considerata. Tale parametro, per quanto riguarda il compressore e la turbina generatrice di gas, è stato identificato nel regime di rotazione dell'albero della turbina generatrice di gas (n_1), e invece nella temperatura di uscita dalla turbina generatrice di gas (T_4) per la turbina di potenza.

La determinazione delle portate richiamata ed elaborata dal compressore viene fatta in modo analogo al calcolo del rapporto di compressione, parametrizzando la portata in funzione del regime di rotazione dell'albero della turbina generatrice di gas (n_1).

Le scelte delle grandezze sulle quali parametrizzare le mappe sono state dettate dal ragionamento sulla fisica del problema: la scelta del parametro n_1 per la determinazione dei rapporti di compressione e della portata elaborata dalle turbomacchine calettate sull'albero della turbina generatrice di gas (compressore e turbina di alta pressione) sono in accordo con il fatto che una rotazione a maggiore velocità del compressore comporta un richiamo maggiore di aria, e quindi una maggiore portata, e che la stessa rotazione a maggiore velocità comporta un maggiore lavoro di compressione e quindi un salto maggiore della pressione fra l'imbocco e l'uscita del componente. Lo stesso ragionamento si può trasferire alla turbina. Si pensi ad esempio a un compressore che lavori al doppio della velocità di un componente analogo in due identici propulsori: il componente alla velocità doppia, a meno di regolazioni dell'ingresso dell'aria de parte dell'IGV (Inlet Guide Vane) e di limiti dovuti al regime di moto del fluido nella turbomacchine, elaborerà una portata maggiore del suo omologo che gira a più bassa velocità e farà eseguire al fluido una maggiore variazione di pressione fra l'ingresso e l'uscita del componente.

Per quanto riguarda la scelta del parametro su cui valutare il regime di funzionamento della turbina di potenza, la scelta è caduta sulla temperatura di uscita dalla prima turbina, in quanto questo è uno dei parametri variabili con i vari regimi del motore ed era disponibile un confronto con dati sperimentali in letteratura ([5]). Sarebbe stato possibile parametrizzare le mappe su altre grandezze (ad esempio la pressione p_4 a valle della turbina HPT), ma non si avrebbe avuto modo di confrontarsi con quello che è il funzionamento di un motore vero al banco. Infatti in letteratura non è disponibile una campagna di prove che fornisca maggiori variabili di interesse per un motore elicotteristico in prove al banco.

3.2 Proprietà dei fluidi

Determinazione delle proprietà dell'aria. A causa delle grandi variazioni di temperatura e per mantenere un'elevata accuratezza della simulazione, le proprietà dei fluidi di lavoro, aria e gas combusti, devono essere calcolate per ogni condizione in ogni stadio del propulsore. L'aria è considerata una miscela di gas semi-ideali, similmente a quanto fatto in [7], e le proprietà termodinamiche della miscela (calori specifici c_p , c_v e loro rapporto k) vengono determinate in base alla sola temperatura tramite una tabella. La pressione della miscela aria è la somma delle pressioni parziali dei gas che la compongono. Sotto queste ipotesi le proprietà termodinamiche del fluido sono funzioni della temperatura e della composizione della miscela. Il fluido aria è quello elaborato all'interno dei componenti presa d'aria, compressore e plenum.

Determinazione delle proprietà dei gas combusti. Sotto l'ipotesi, generalmente verificata nei moderni motori, che la combustione avvenga con grande eccesso d'aria e che il processo di combustione possa essere considerato completo all'uscita della camera di combustione, il fluido di lavoro considerato durante tutte le trasformazioni all'interno delle turbine e della camera di combustione è una miscela di gas combusti, con uno specifico rapporto aria/combustibile (FAR: Fuel to Air Ratio), noto dall'apporto di combustibile e di aria nel motore all'istante considerato. Per la definizione delle proprietà termodinamiche dei fluidi di lavoro si è fatto riferimento al capitolo 3 del testo [1]. La costante dei gas R (dimensionalmente $[J/KgK]$) per la miscela di gas combusti è calcolata in funzione del rapporto aria/combustibile (FAR) come:

$$R_{kerosene} = 287.05 - (0.00990) (FAR) + (10^{-7}) (FAR)^2$$

dove FAR è definito come:

$$FAR = \frac{m_{fuel}}{m_{air}} = \frac{\dot{m}_{fuel}}{\dot{m}_{air}}$$

Si noti (figura 3.1) che per i gas ottenuti dalla combustione di kerosene con aria, la costante dei gas R non cambia in modo considerevole per valori di FAR quali quelli utilizzati nei motori aeronautici (0.01:0.05).

Il calore specifico c_p dei gas combusti provenienti dalla combustione del kerosene è funzione della temperatura e della composizione della miscela aria/combustibile. Il kerosene impiegato nella propulsione aeronautica ha una composizione chimica abbastanza omogenea, e il suo calore specifico per

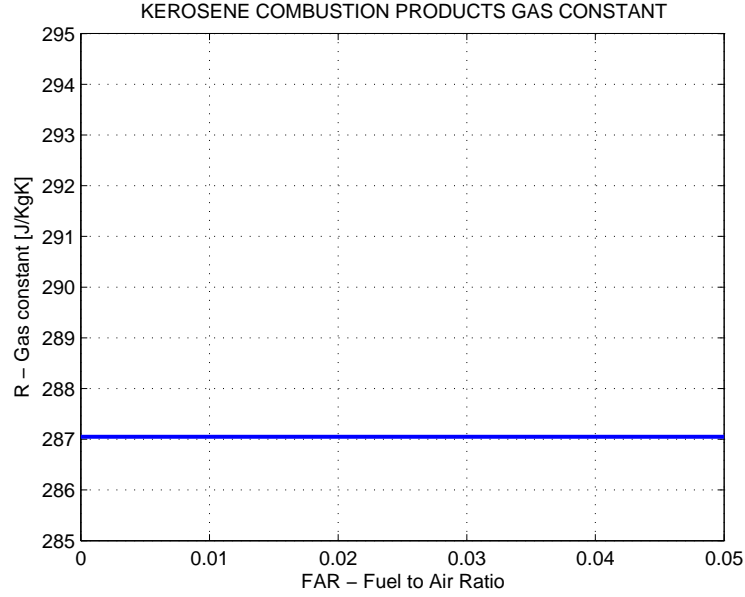


Figura 3.1: Variazione della costante dei gas R in funzione della temperatura per i gas combusti di kerosene.

temperature comprese fra 200 K e 2000 K è ben approssimatao dalla equazione 3.1, formula empirica che tiene conto della temperatura e del rapporto aria/combustibile:

$$c_p = A_0 + A_1 T_Z + A_2 T_Z^2 + A_3 T_Z^3 + A_4 T_Z^4 + A_5 T_Z^5 + A_6 T_Z^6 + A_7 T_Z^7 + A_8 T_Z^8 + \frac{FAR}{1 + FAR} (B_0 + B_1 T_Z + B_2 T_Z^2 + B_3 T_Z^3 + B_4 T_Z^4 + B_5 T_Z^5 + B_6 T_Z^6 + B_7 T_Z^7) \quad (3.1)$$

dove $T_Z = \frac{T}{1000} [K]$ e $A_0 \dots A_8$ e $B_0 \dots B_7$ sono i coefficienti del polinomio riportati in tabella 3.2.

Per l'andamento del calore specifico dei gas combusti del kerosene si faccia riferimento alla figura 3.2

Per il calcolo del rapporto fra calori specifici dei gas combusti del kerosene, denominato con k si è fatto riferimento alla relazione:

$$k = \frac{c_p}{c_v} = \frac{c_p}{c_p - R}$$

L'andamento del rapporto dei calori specifici è riportato nella figura 3.3.

A_0	0.992313	B_0	-0.718874
A_1	0.236688	B_1	8.74781
A_2	-1.852148	B_2	-15.863157
A_3	6.083152	B_3	17254096
A_4	-8.893933	B_4	-10.233795
A_5	7.097112	B_5	3.081778
A_6	-3.234725	B_6	-0.361112
A_7	0.794571	B_7	-0.003919
A_8	-0.081873	B_8	0.0555930

Tabella 3.1: Coefficienti dei polinomi per il calcolo di k e c_p dei gas combusti nel modello di Walsh [1].

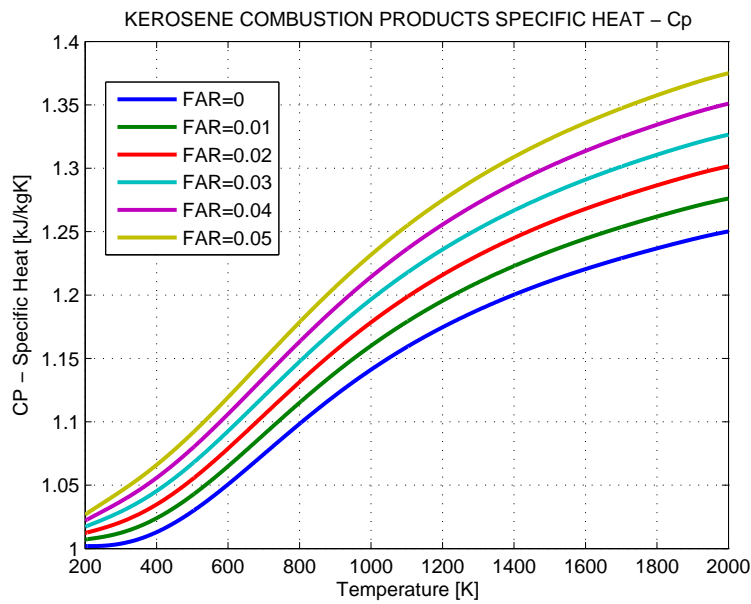


Figura 3.2: Calore specifico dei gas ottenuti dalla combustione di kerosene in funzione della temperatura per diversi rapporti aria/combustibile.

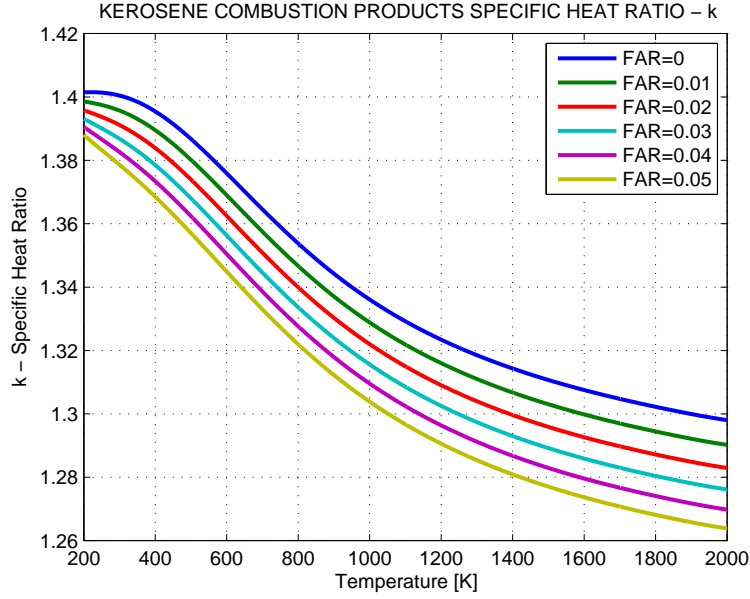


Figura 3.3: Rapporto fra calori specifici del kerosene in funzione di temperatura e rapporto aria/combustibile.

Le relazioni presentate sono valide per temperature comprese fra 200 e 2000 K, ma dal momento che in camera di combustione si possono raggiungere per brevi momenti anche temperature dell'ordine dei 2500 K, si rende necessario un modello più complesso, quale quello messo a punto in [12]: in esso vengono implementate le seguenti relazioni, valide fra 300 e 3500 K e anche oltre con errori modesti:

$$\frac{c_{p_{combustibile}}}{c_{paria}} = \frac{1 + FAR(I_0 + I_1 T + \frac{I_2}{T^2})}{1 + FAR}$$

dove i coefficienti I_0 , I_1 e I_2 si determinano come:

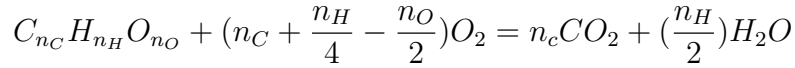
$$I_0 = \frac{18.0566 n_C + 8.3485 n_H + 15.1616 n_O}{M_{combustibile}} \quad (3.2)$$

$$I_1 = \frac{0.00223 n_H}{M_{combustibile}} \quad (3.3)$$

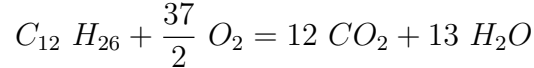
$$I_2 = \frac{1077768.4 n_C}{M_{combustibile}} \quad (3.4)$$

in cui $M_{combustibile}$ è la massa molecolare di una molecola di combustibile (nel nostro caso kerosene, $M_{kerosene} = 170$), mentre n_C , n_O , n_H sono il numero

di moli di carbonio, ossigeno e idrogeno necessarie alla generica reazione di combustione:



Nel caso di combustione di kerosene in aria secca in rapporto stechiometrico avremo:



e quindi i numeri di moli assumono i valori:

$$n_C = 12 \quad (3.5)$$

$$n_H = 26 \quad (3.6)$$

$$n_O = 0 \quad (3.7)$$

L'utilizzo di questo schema di soluzione permette un'ottima rappresentazione delle proprietà dei gas combusti di kerosene in un intervallo di temperature molto vasto (200-5000 K), con errori quantificati da [12] in qualche decimo di punto percentuale.

Nelle figure 3.5 e 3.4 si è confrontato il comportamento dei gas di scarico del modello di Guha e quello dell'aria per temperature fino ai 3000 K. Il modello polinomiale di [1], invece, per temperature superiori ai 2200 K non è più utilizzabile perchè fornisce valori di c_p e k palesemente impossibili: si veda a riguardo la figura 3.6.

3.3 Componenti del modello

In figura 3.7 sono visibili tutti i componenti del modello di motore. Si noti come non ci sia un'effettiva corrispondenza fra i componenti fisici e quelli del modello numerico.

Il modello numerico è composto dei seguenti componenti:

- presa d'aria;
- compressore;
- plenum;
- combustore;
- turbina di alta pressione;
- turbina di potenza;

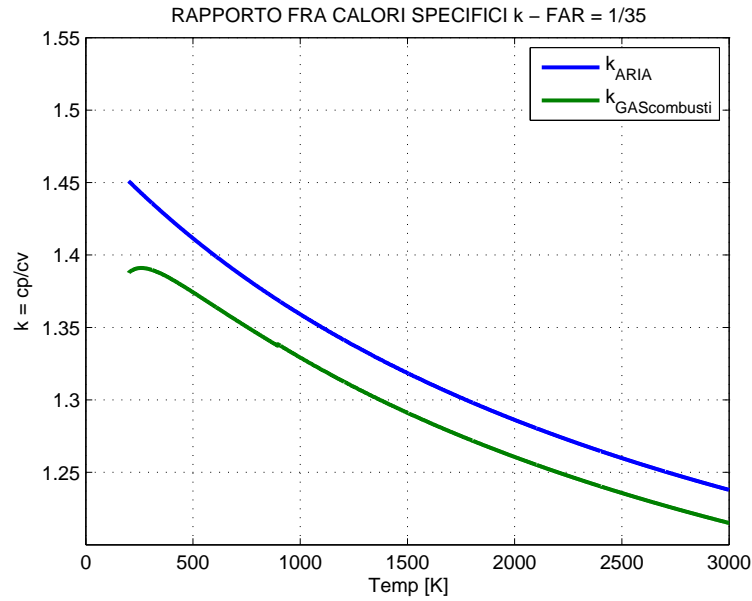


Figura 3.4: Rapporto fra calori specifici k dell'aria e dei gas combusti di kerosene con rapporto aria/combustibile di $\frac{1}{35}$ ottenuto con il modello di Guha.

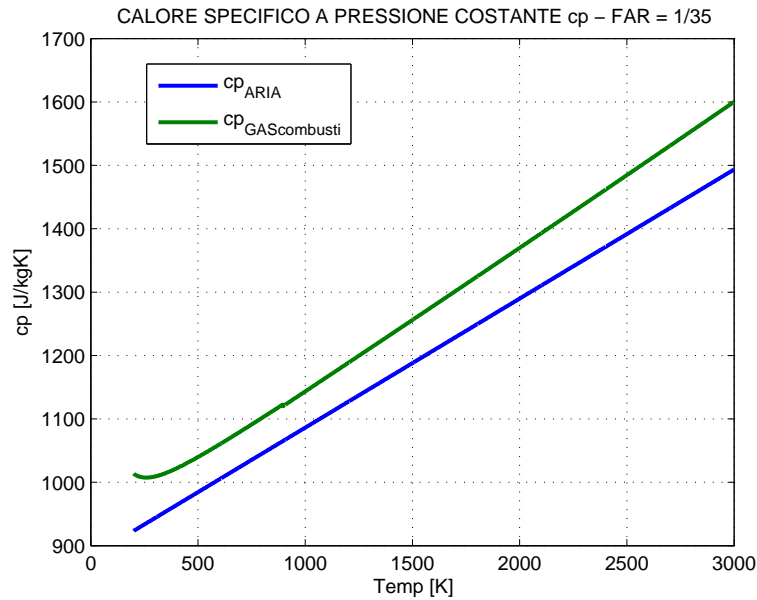


Figura 3.5: Calore specifico c_p dell'aria e dei gas combusti con rapporto aria/combustibile di $\frac{1}{35}$, ottenuto con il modello di Guha per un ampio intervallo di temperature.

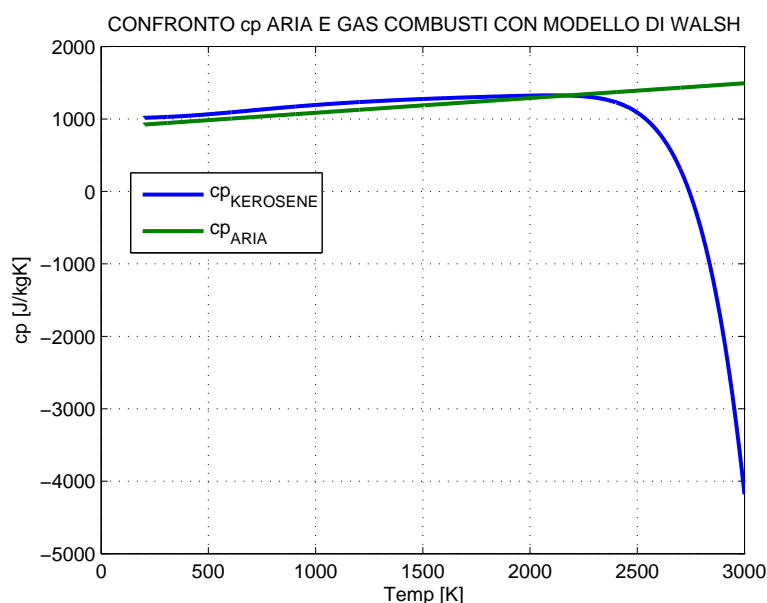


Figura 3.6: Calore specifico di aria e gas combusti ottenuto con modello di Walsh, con FAR=1/35.

- carico.

Si noti come i componenti del modello non corrispondano esattamente a quelli di un motore fisico. E' evidente che il plenum non è presente in un motore fisico; questo componente, come verrà spiegato più estesamente nel seguito, serve a implementare un principio di conservazione della massa non stazionario all'interno di un volume. Allo stesso modo nel modello sono presenti un generico compressore e due generiche turbine non suddivise in stadi, e non è specificato se questi componenti siano di tipo assiale o centrifugo. Questa generalità dei componenti del modello è tesa a riprodurre il maggior numero di motori possibile, senza entrare nel dettaglio della specifica configurazione. Nell'implementazione del modello si è scelto di riprodurre i componenti agli effetti esterni e non entrare nel dettaglio delle loro peculiarità proprio per mantenere maggiore generalità e portabilità rispetto a diversi sistemi.

3.3.1 Presa d'aria

La presa d'aria è del tipo dinamico, ideale perché si ipotizza che non vi siano perdite per attrito. In presenza di una componente di velocità v non nulla in direzione perpendicolare al piano dell'ingresso dell'aria nel motore, la

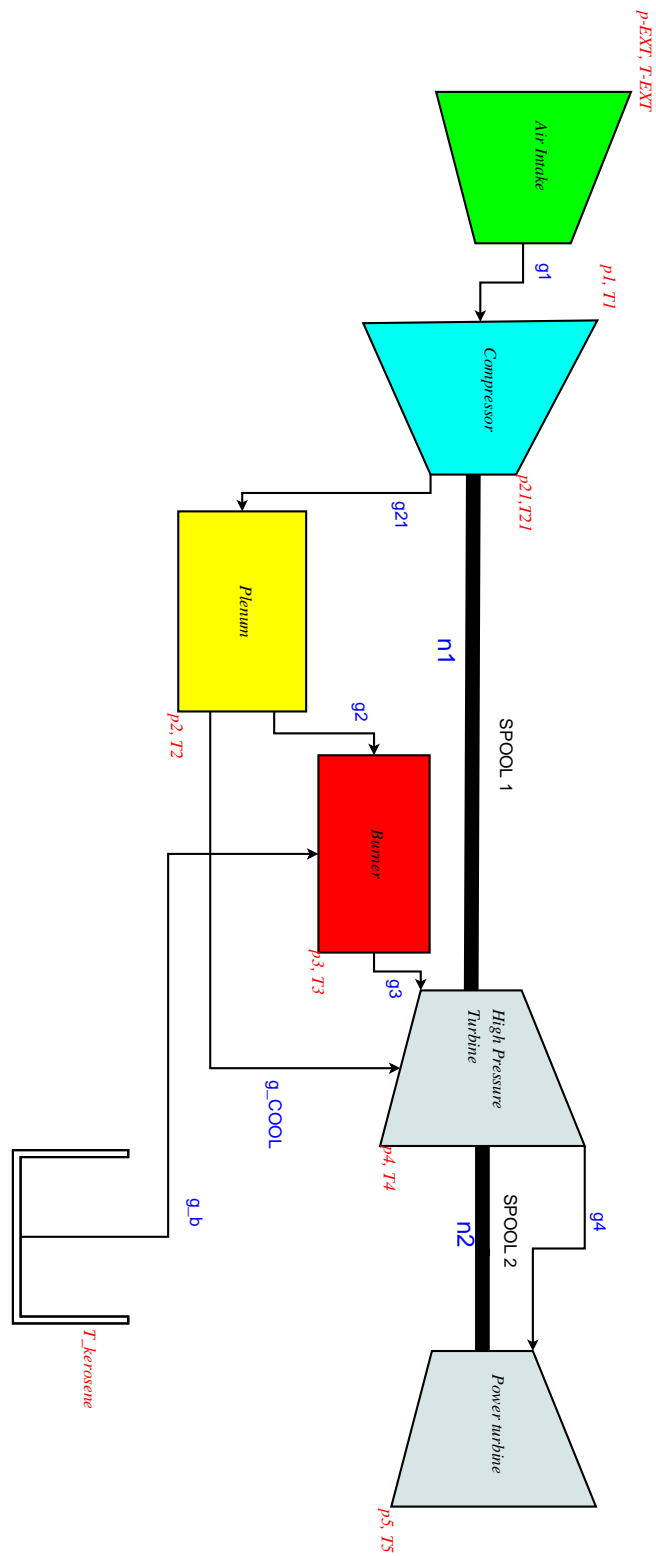


Figura 3.7: Schema del motore e flusso di informazioni fra i vari componenti.

pressione totale e la temperatura totale all'ingresso del compressore vengono incrementate di una quantità fornita dalle relazioni:

$$\Delta p = \frac{1}{2} \rho v^2 \quad (3.8)$$

$$\Delta T = \frac{v^2}{2c_p} \quad (3.9)$$

La 3.8 traduce il teorema di Bernoulli per flussi incomprimibili, la 3.9 deriva dal primo principio della termodinamica. in questo modo la pressione totale si aggiorna come:

$$p_{tot} = p_s + \Delta p \quad (3.10)$$

e la temperatura totale:

$$T_{tot} = T_s + \Delta T \quad (3.11)$$

dove p_s e T_s sono la pressione statica e la temperatura statica, assunte omogenee su tutta la sezione all'imbocco della presa d'aria.

3.3.2 Compressore

Il compressore è un componente adiabatico, istantaneo e privo di dimensione (0-D). Durante i transitori, in accordo con [7], la risposta del compressore è assunta essere quella di un componente quasi stazionario, ovvero la sua dinamica è considerata molto più veloce di quella dei componenti più lenti, e sono quindi utilizzate mappe ricavate dal funzionamento del componente a regime. Le mappe forniscono la portata elaborata dal compressore g_{21} e il rapporto di compressione β_C in funzione della velocità di rotazione n_1 , passando attraverso una normalizzazione e una successiva scalatura come verrà illustrato in dettaglio nel paragrafo relativo alle mappe.

$$\beta_c = f(n_1) \quad (3.12)$$

$$g_{21} = f(n_1) \quad (3.13)$$

La temperatura al termine della compressione viene calcolata attraverso una trasformazione isoentropica corretta con un rendimento. Chiamando T_1 e p_1 la pressione e la temperatura all'imbocco del compressore e quindi all'uscita della presa d'aria avremo:

$$T_{21is} = T_1 \left(\frac{p_{21}}{p_1} \right)^{\frac{k-1}{k}} \quad (3.14)$$

La 3.14 è la trasformazione che l'aria subirebbe passando dalla pressione P_1 alla pressione P_{21} idealmente senza perdite e senza aumento di entropia (trasformazione isoentropica). Naturalmente, noto il rapporto di compressione β_c definito come

$$\beta_c = \frac{p_{21}}{p_1} \quad (3.15)$$

è possibile riscrivere la 3.14 come:

$$T_{21is} = T_1(\beta_c)^{\frac{k-1}{k}} \quad (3.16)$$

Nota la T_{21is} , che è la temperatura di uscita dalla compressione isoentropica, è possibile calcolare il lavoro compiuto dal compressore correggendo il lavoro isoentropico con un rendimento:

$$L_{compressore} = c_p(T_1 - T_{21is})\eta_c \quad (3.17)$$

Il rendimento η_c , definito rendimento isoentropico del compressore, è assunto essere costante in tutto il campo di funzionamento della turbomacchina, ma potrebbe in futuro essere parametrizzato in funzione di una grandezza quale il regime di rotazione dell'albero. E' infine possibile calcolare la temperatura reale al termine della compressione attraverso la 3.18:

$$T_{21} = T_1 - \frac{L_{compressore}}{c_p} \quad (3.18)$$

La temperatura T_{21} all'uscita dal compressore è quindi dipendente dalle condizioni ambientali di temperatura e pressione esterne, dalla velocità di volo del velivolo e dal rapporto di compressione che il compressore esprime al momento.

3.3.3 Plenum

Come accennato in precedenza, utilizzando un plenum si riesce a riprodurre un effetto di comprimibilità dei gas al passaggio nella camera di combustione e quindi anche un ritardo nell'adattamento della pressione. L'utilizzo del plenum, comune a molta letteratura [7, 9], permette di attribuire alla camera di combustione o combustore la sola funzione di accumulatore di energia. Il plenum si rende necessario per tenere conto del volume dei vari componenti che nell'implementazione del modello sono trattati come elementi 0-D ovvero privi di volume. Nella nostra implementazione l'unico plenum impiegato è posto fra l'uscita del compressore e l'imbocco del combustore, e serve a considerare il bilancio di massa dovuto all'aria proveniente dal compressore,

l'aria spillata per le varie utenze, l'aria diretta al raffreddamento della turbina, l'aria elaborata dal combustore. Il bilancio di massa non stazionario è dato dall'equazione:

$$\frac{V_{plenum}}{m R T_{out}} \frac{dp_{out}}{dt} = g_{in} - g_{out} \quad (3.19)$$

dove V_{plenum} è il volume attribuito al componente plenum, T_{out} e p_{out} la temperatura e la pressione all'uscita dal plenum, g_{in} e g_{out} le portate massiche in ingresso e in uscita dal componente plenum. La portata d'aria in ingresso al plenum è quella elaborata dal compressore:

$$g_{in} = g_{compressore} = g_{21}$$

La portata g_{out} che abbandona il plenum è divisa in due parti:

$$g_{out} = g_2 = g_{cool} + g_3$$

- g_{cool} : portata d'aria spillata e inviata alle utenze e al raffreddamento della turbina HPT.
- g_3 : portata d'aria che viene inviata nel combustore per partecipare alla combustione.

Possiamo quindi determinare la portata massica in uscita dal plenum g_2 in funzione della variazione della pressione in uscita dal plenum $\frac{\partial p_2}{\partial t}$ e delle portate massiche di aria del compressore e spillate:

$$g_2 = g_{21} - g_{cool} - \frac{V_{plenum}}{k_2 R T_2} \tau_{plenum} \frac{dp_2}{dt}$$

dove la costante di tempo τ_{plenum} è definita come:

$$\tau_{plenum} = \frac{M_{plenum}}{k_2 g_2}$$

Poichè la pressione p_2 è, per l'ipotesi di nessuna perdita nel plenum, la stessa pressione all'uscita dal compressore, determinata tramite una mappa in funzione della velocità di rotazione n_1 e della temperatura p_1 , ovvero:

$$p_2 = p_{21} = \beta_C(n_1)p_1 = f(n_1, p_1)$$

Allora è possibile esprimere la derivata $\frac{\partial p_2}{\partial t}$ in funzione dell'accelerazione rotativa del primo albero \dot{n}_1 , che è uno dei residui del sistema, secondo il seguente ragionamento:

$$\frac{\partial p_2}{\partial t} = \dot{\beta}_C p_1 + \beta_C \dot{p}_1$$

Poichè si può assumere che p_1 vari molto lentamente rispetto a β_C , ovvero:

$$\dot{p}_1 \approx 0$$

allora possiamo scrivere:

$$\frac{\partial p_2}{\partial t} = \dot{\beta}_C p_1$$

Dove la derivata temporale $\dot{\beta}_C$ è in realtà funzione dell'accelerazione di rotazione, e si può così esplicitare:

$$\dot{\beta}_C = \frac{\partial \beta_C}{\partial t} = \frac{\partial \beta_C}{\partial n_1} \frac{\partial n_1}{\partial t} = \frac{\partial \beta_C}{\partial n_1} \dot{n}_1$$

Per valutare la derivata del rapporto di compressione rispetto alla velocità di rotazione, $\frac{\partial \beta_C}{\partial n_1}$ si può alternativamente:

- valutare la derivata del polinomio con cui si è costituita la mappa di β_C .
- approssimare tale derivata nel campo di funzionamento.

Nel caso $\frac{\partial \beta_C}{\partial n_1}$ si possa valutare dal polinomio che esprime la mappa di β_C

$$\beta_C = a_0 + a_1 n_1 + a_2 n_1^2 + a_3 n_1^3 + \dots + a_n n_1^n$$

allora

$$\frac{\partial \beta_C}{\partial n_1} = a_1 + a_2 n_1 + a_3 n_1^2 + \dots + a_n n_1^{n-1}$$

Nel caso invece si voglia approssimare la derivata, si può ad esempio ipotizzare un andamento lineare del rapporto di compressione con la velocità di rotazione dell'albero e quindi:

$$\beta_C = \beta_C(n_1 = 0) + \frac{\beta_{Cnom}(n_1 = n_{1nom}) - \beta_C(n_1 = 0)}{n_{1nom}} n_1$$

Poichè in condizioni di compressore fermo si può assumere che la pressione a valle del componente sia la stessa a valle dello stesso (nessuna compressione):

$$\beta_C(n_1 = 0) = 1$$

allora la derivata parziale assume il valore costante:

$$\frac{\partial \beta_C}{\partial n_1} = \frac{\beta_{Cnom} - 1}{n_{1nom}}$$

Otteniamo quindi la portata in uscita dal plenum ad ogni istante come:

$$g_2 = \dot{m}_{out} = g_{21} - g_{cool} - \frac{p_2 V_{plenum}}{k_2 R T_2} \frac{\partial \beta_C}{\partial n_1} \dot{n}_1 \quad (3.20)$$

3.3.4 Combustore

La camera di combustione è, nel nostro modello, un puro accumulatore di energia. Essa viene implementata con il nome di combustore (burner), per risaltare la funzione di generazione di energia attraverso la reazione di combustione. Alla camera di combustione è associata la sola dinamica termica del sistema: quella della temperatura in camera di combustione T_3 . Essendo la camera di combustione l'elemento subito a valle del plenum, si assume che in essa le condizioni all'imbocco e allo sbocco siano omogenee sulla sezione ed eguali a quelle in uscita dal plenum, che sono le stesse in uscita dal compressore:

$$p_{incombustore} = p_2 = p_{21}$$

$$T_{incombustore} = T_2 = T_{21}$$

All'interno del componente combustore avviene la miscelazione dell'aria con il combustibile. La portata di combustibile g_b è l'ingresso principale del sistema, mentre la portata d'aria in ingresso è quella proveniente dal plenum, come già visto nel paragrafo relativo:

$$g_3 = g_2 - g_{cool}$$

Il componente combustore non è privo di perdite come il plenum, ma fra l'imbocco e l'uscita è prevista una perdita di carico per il fluido quantificabile con un rendimento y_{CC} :

$$p_3 = p_{outcombustore} = p_2 y_{CC}$$

Il rendimento y_{CC} è assunto essere costante.

L'equazione che descrive la dinamica del combustore è ottenuta dal bilancio di energia instazionario:

$$\frac{d(M_{CC} u_{CC})}{dt} = \frac{g_{in} h_{in} + g_b (h_b + \eta_b LHV) - g_{out} h_{out}}{g_{out} c_{p_{out}}} \quad (3.21)$$

dove M_{CC} e u_{CC} sono rispettivamente la massa di gas presenti all'interno del componente combustore e l'energia interna specifica dei gas; LHV è il potere calorifico inferiore del combustibile (Lower Heating Value).

Trascurando le variazioni della massa e del calore specifico l'equazione 3.21 può essere riscritta come

$$\dot{T}_{out} = \frac{g_{in}h_{in} + g_b(h_b + \eta_b LHV) - g_{out}h_{out}}{g_{out}c_{p_{out}}\tau_{CC}} \quad (3.22)$$

La costante di tempo τ_{CC} può essere valutata come:

$$\tau_{CC} = \frac{M_{CC}}{k g_{out}}$$

Usando la nomenclatura del modello e della figura 3.7, la dinamica termica, relativa alla temperatura in camera di combustione può essere espressa dalla:

$$\dot{T}_3 = \frac{g_2 h_2 + g_b(h_b + \eta_b LHV) - g_3 h_3}{g_3 c_{p_3} \tau_{cc}} \quad (3.23)$$

dove le entalpie specifiche h_3, h_2, h_b sono calcolate come:

$$h_2 = c_{p_2} T_2$$

$$h_3 = c_{p_3} T_3$$

$$h_b = c_{v_{kerosene}} T_{kerosene}$$

Per le proprietà del combustibile si è fatto riferimento a [13].

Riscriviamo l'equazione 3.23 può essere riscritta esplicitando le relazioni delle entalpie specifiche:

$$\dot{T}_3 = \frac{R T_3 k_3}{p_2 y_{CC} V_{CC} c_{p_3}} [g_2 c_{p_2} T_2 + g_b(c_{v_b} h_b + \eta_b LHV) - (g_2 + g_b)c_{p_3} T_3] \quad (3.24)$$

Nell'equazione 3.24, le grandezze dipendenti da T_3 , e quindi dalla stessa derivata \dot{T}_3 sono:

$$k_3, c_{p_3}, T_3$$

e quindi l'equazione risulta implicita.

I calori specifici e i loro rapporti sono calcolati attraverso il modello di Guha ([12]) per i gas combustibili di kerosene, come spiegato nella sezione relativa ai fluidi e alle loro proprietà. I metodi di integrazione adottati dovranno tenere in conto l'implicitezza del sistema per la risoluzione delle equazioni differenziali.

3.3.5 Turbina generatrice di gas

La turbina generatrice di gas (HPT: High Pressure Turbine) è modellata come un componente istantaneo e privo di volume, a singolo stadio. L'ipotesi che sta alla base di una tale semplificazione, similmente a quanto fatto per il compressore, è che la dinamica del componente turbina sia molto più veloce delle due dinamiche meccaniche e della dinamica termica del sistema, cosicchè il componente possa essere approssimato in modo quasi-statico. Similmente al compressore, il rapporto di espansione della turbina β_{HPT} è parametrizzato, attraverso una mappa, su un parametro caratteristico che individui il regime di funzionamento del componente: la velocità di rotazione n_1 , normalizzata rispetto al suo valore nominale.

$$\beta_{HPT} = f(\bar{n}_1) \quad (3.25)$$

Il rapporto di espansione così ottenuto è un β globale, che tiene conto di tutti gli eventuali stadi di una turbina reale.

La pressione a valle della turbina di alta pressione è quindi determinata a partire dal regime di rotazione normalizzato e dalla pressione p_3 in camera di combustione:

$$p_4 = p_{outHPT} = \beta_{HPT} p_3 = f(p_3, \bar{n}_1) \quad (3.26)$$

Alla turbina HPT giunge parte dell'aria spillata dal compressore, per il raffreddamento delle palette, di quest'afflusso di aria si tiene conto con il bilancio di portata:

$$g_4 = g_{outHPT} = g_3 + g_{cool} \quad (3.27)$$

Per il calcolo della temperatura in uscita dalla turbina di alta pressione, necessaria poi al calcolo della potenza generata, si è fatto ricorso a una espansione isoentropica corretta con un rendimento: si trova prima la temperatura relativa alla trasformazione isoentropica che porti il fluido alla stessa condizione di pressione prevista, si calcola il lavoro che produce tale espansione e infine questo valore di lavoro viene corretto con un rendimento isoentropico. Il valore del lavoro calcolato con il rendimento isoentropico viene quindi diviso per il calore specifico, determinato con la temperatura della trasformazione isoentropica corrispondente, per ottenere la temperatura finale al termine dell'espansione reale.

$$T_{4is} = T_3 \left(\frac{p_4}{p_3} \right)^{\frac{k3-1}{k3}} = T_3 (\beta_{HPT})^{\frac{k3-1}{k3}} \quad (3.28)$$

$$c_{p4is} = f(T_{4is}, FAR) \quad (3.29)$$

$$L_{HPT} = c_{p_{4is}}(T_3 - T_{4is})\eta_{HPT} \quad (3.30)$$

$$c_{p_4} = f(T_4) \quad (3.31)$$

$$T_4 = T_3 - \frac{L_{HPT}}{c_{p_4}} \quad (3.32)$$

La potenza generata dalla turbina è immediatamente calcolabile, noto il salto entalpico fra l'imbocco e lo sbocco del componente:

$$P_{HPT} = g_3 h_3 + g_{cool} h_{cool} - g_4 h_4 \quad (3.33)$$

dove le entalpie sono calcolate nel modo seguente:

$$h_3 = c_{p_3} T_3 \quad (3.34)$$

$$h_{cool} = c_{p_2} T_2 = c_{p_{21}} T_{21} \quad (3.35)$$

$$h_4 = c_{p_4} T_4 \quad (3.36)$$

L'entalpia viene calcolata utilizzando il modello di Guha per le proprietà dei gas combustibili di kerosene. Si noti che la portata g_{cool} è composta di sola aria alla temperatura di uscita dal plenum, in quanto è la portata d'aria utilizzata per raffreddare le palette della turbina. Questo comporta un nuovo calcolo del rapporto aria/combustibile FAR, che modifica quindi le proprietà dei gas di scarico.

3.3.6 Turbina di potenza

La turbina di potenza (PT: power turbine) è stata modellata similmente a alla turbina generatrice di gas eccetto che per due differenze:

1. la turbina PT a differenza della turbina HPT non riceve una portata d'aria più fresca per il raffreddamento delle palette;
2. la pressione a valle della turbina PT non può assumere valori inferiori a quelli della pressione esterna (condizione di flusso adattato).

La portata di gas elaborata dalla turbina PT, non essendoci ulteriori immissioni di fluido, è la stessa che proviene dall'uscita della turbina HPT:

$$g_5 = g_4 \quad (3.37)$$

La determinazione della pressione all'uscita della turbina avviene sempre attraverso una mappa, ma poichè in questo caso il componente deve lavorare in un regime di giri pressochè costante in quanto dalla sua velocità di rotazione dipende la velocità di rotazione del carico (rotore), non è possibile parametrizzare il β di espansione sul regime di rotazione n_2 . Il parametro scelto per la parametrizzazione del rapporto di espansione β_{PT} è la temperatura in uscita dalla turbina generatrice di gas, normalizzata con il suo valore nominale.

$$\beta_{PT} = f(\overline{T}_4) \quad (3.38)$$

$$p_5 = \beta_{PT} p_4 = f(p_4, \overline{T}_4) \quad (3.39)$$

in cui la pressione p_5 non può mai essere inferiore a quella atmosferica. Noto il valore di p_5 è possibile determinare il valore di temperatura dei gas all'uscita della turbina tramite un'espansione isoentropica corretta con un rendimento, esattamente come fatto nel componente HPT.

$$T_{5is} = T_4 \left(\frac{p_5}{p_4} \right)^{\frac{k_4-1}{k_4}} \quad (3.40)$$

$$c_{p5is} = f(T_{5is}, FAR5) \quad (3.41)$$

$$L_{PT} = c_{p5is} (T_4 - T_{5is}) \eta_{PT} \quad (3.42)$$

$$c_{p5} = f(T_5, FAR) \quad (3.43)$$

$$T_5 = T_4 - \frac{L_{PT}}{c_{p5}} \quad (3.44)$$

La potenza, espressa in Watt, della turbina di potenza è proporzionale al salto entalpico a cavallo del componente ed alla portata d'aria elaborata:

$$P_{PT} = g_5 (c_{p4} T_4 - c_{p5} T_5) \quad (3.45)$$

3.3.7 Modello di carico semplificato

Dovendo provare il funzionamento del codice senza avere a disposizione fin da subito un modello sofisticato di rotore da connettere al propulsore, è stato messo a punto un modello semplificato di carico, che riproducesse in maniera approssimata il rotore dell'elicottero Sikorsky UH-60.

I dati del rotore considerato sono:

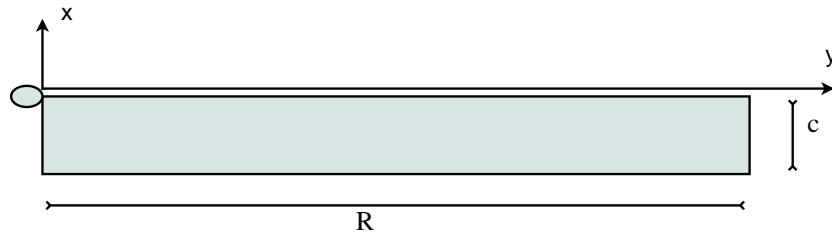


Figura 3.8: Pala di rotore semplificata.

- diametro rotore: $D = 16.36 \text{ m}$;
- numero pale: 4;
- superficie del disco rotore: $S = 210 \text{ m}^2$;
- corda media della pala del rotore principale : $c = 0.53 \text{ m}$;
- profilo utilizzato per le pale del rotore principale: SC1095;
- velocità angolare nominale del rotore $\omega_{ROT} = 240 \text{ RPM}$;

Sotto le ipotesi di:

- volo in hovering (nessuna componente di velocità longitudinale);
- coppia resistente dovuta solo alla resistenza delle pale del rotore;
- pala rettangolare non svergolata e rigida;
- movimenti di flappeggio e di comando trascurati;
- rapporto di trasmissione costante in ogni condizione di funzionamento;
- presenza di un solo tipo di profilo in tutta l'apertura;
- coefficiente di resistenza C_D costante e pari a quello di massima efficienza del profilo;
- densità dell'aria costante e pari al valore standard ISA;

si è calcolata la resistenza per unità di lunghezza del profilo:

$$d = \frac{D}{l} = \frac{1}{2} \rho v^2 c C_D \quad (3.46)$$

Integrando la resistenza per unità di lunghezza sul Raggio del rotore $R = \frac{D}{2}$ otteniamo la resistenza di una pala:

$$D_{pala} = \int_0^R \frac{1}{2} \rho v^2 c C_D dy \quad (3.47)$$

Si noti che la velocità della sezione infinitesima del rotore dipende in generale dalla coordinata y e dalla velocità di volo dell'elicottero.

$$v = f(y, v_{volo})$$

. Ad esempio per la pala che avanza parallela alla velocità di volo:

$$v(y, v_{volo}) = v_{volo} + \omega_{ROT} y \quad (3.48)$$

ma considerando come da ipotesi l'elicottero in hover ($v_{volo} = 0$):

$$v(y) = \omega_{ROT} y \quad (3.49)$$

Moltiplicando la resistenza di un tratto di pala infinitesimo per la sua distanza dal centro (coordinata y in figura 3.8), integrando il contributo infinitesimo sul raggio del rotore e moltiplicando per il numero di pale, otteniamo la coppia esercitata dal rotore :

$$C = n_{pale} \int D_{pala} dy = 4 \int_0^R \frac{1}{2} \rho v^2 c C_D y dy \quad (3.50)$$

sostituendo alla velocità $v(y)$ la sua espressione 3.49, otteniamo:

$$C = 4 \int_0^R \frac{1}{2} \rho (\omega_{ROT} y)^2 c C_D y dy \quad (3.51)$$

dove, per le ipotesi prima descritte, è possibile estrarre dall'integrale tutti i termini non dipendenti da y e ottenere:

$$C = 2 \rho c C_D \omega_{ROT}^2 \left[\frac{y^4}{4} \right]_0^R = \frac{1}{2} \rho c C_D R^4 \omega_{ROT}^2 \quad (3.52)$$

Sappiamo che la velocità angolare del rotore è legata alla velocità dell'albero di potenza n_2 attraverso il rapporto di riduzione della trasmissione:

$$\omega_{ROT} = n_2 R_{trasmissione} \quad (3.53)$$

Essendo i regimi di rotazione nominali noti $n_{2_{nom}}$ e ω_{ROT} , allora è noto anche il rapporto di trasmissione:

$$R_{trasmissione} = \frac{n_{2_{nominale}}}{\omega_{ROT_{nominale}}} \quad (3.54)$$

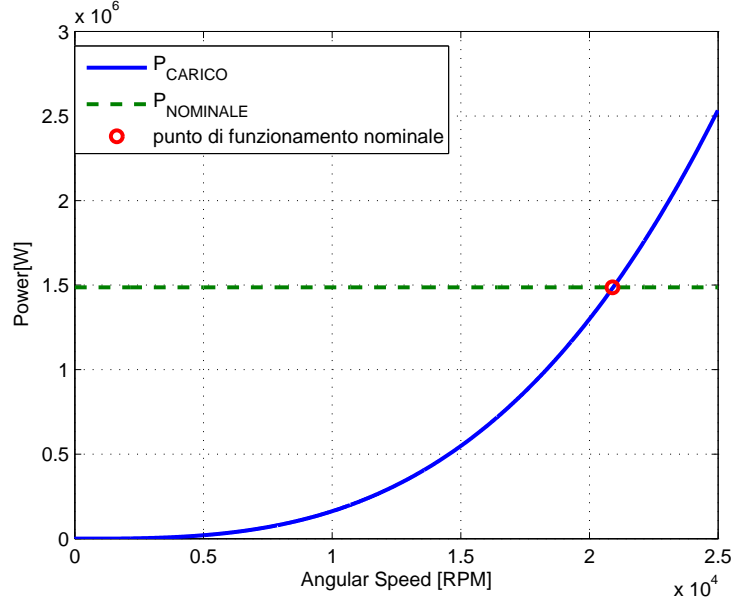


Figura 3.9: Potenza richiesta dal carico modellato in funzione della velocità di rotazione n_2 confrontata con i valori di potenza e velocità di rotazione nominali per il motore GE t700.

Sostituendo la 3.53 nella 3.52 è quindi possibile scrivere il valore di coppia richiesta dal carico:

$$C_{LOAD}(n_2) = \frac{1}{2} \rho c C_D R^4 (n_2 R_{trasmissione})^2 = K_R n_2^2 \quad (3.55)$$

Nella 3.55 è possibile notare che *la coppia richiesta dal carico è una funzione quadratica della velocità di rotazione n_2* . La costante K_R è pari a:

$$K_R = \frac{1}{2} \rho c C_D R^4 R_{trasmissione}^2 \quad (3.56)$$

La potenza dissipata dal carico è quindi ottenibile moltiplicando la coppia C_{LOAD} per la velocità di rotazione dell'albero:

$$P_{LOAD} = C_{LOAD} n_2 = K_R n_2^3 \quad (3.57)$$

Si riporta in figura 3.9 l'andamento della potenza richiesta dal carico così modellato in funzione della velocità di rotazione n_2 .

3.3.8 Dinamica degli alberi di trasmissione

Note le potenze espresse dalle turbine P_{PT} e P_{HPT} , la potenza richiesta dal compressore P_C , la potenza richiesta dal carico P_{LOAD} e le inerzie dei due

alberi di rotazione J_{PT} e J_{HPT} (intese come inerzie complessive dell'albero e degli elementi a lui connessi: compressori, turbine, carichi, ecc.), è possibile scrivere le dinamiche meccaniche del motore, che traduce il principio di conservazione del momento angolare. Nella loro modellazione si è tenuto conto di alcune perdite per attrito, modellate come frazioni delle potenze delle turbine: $P_{f_{HPT}}$ e $P_{f_{PT}}$ sono le potenze perse rispettivamente sull'albero della turbina generatrice di gas e sull'albero della turbina di potenza:

$$P_{f_{HPT}} = K_{f1} P_{HPT} \quad (3.58)$$

$$P_{f_{PT}} = K_{f2} P_{PT} \quad (3.59)$$

dove K_{f1} e K_{f2} sono moltiplicatori che esprimono la frazione di potenza persa in attriti (attriti meccanici, resistenza sul disco dei rotori, e potenza perduta per muovere eventuali utenze ausiliarie). Nel modello si è implementato un valore di qualche centesimo, in linea con [7].

L'equilibrio dei momenti è esprimibile come:

$$\frac{d\omega}{dt} = \frac{P}{J\omega} \quad (3.60)$$

dove J è il momento di inerzia del corpo rotante, P la potenza delle forze agenti sul sistema, ω la velocità angolare.

Le espressioni delle dinamiche meccaniche sui due alberi si possono scrivere esplicitando le accelerazioni angolari come:

$$\dot{n}_1 = \frac{P_{HPT} - P_C - P_{f_{HPT}}}{J_{HPT}n_1} \quad (3.61)$$

$$\dot{n}_2 = \frac{P_{PT} - P_{load} - P_{f_{PT}}}{J_{PT}n_2} \quad (3.62)$$

3.4 Dinamica del sistema e metodo di soluzione

La dinamica del sistema è governata da equazioni algebriche ed equazioni differenziali. Le grandezze che sono determinate da equazioni differenziali, tutte del primo ordine, sono tre:

- la dinamica meccanica determina le accelerazioni angolari dei due alberi \dot{n}_1 e \dot{n}_2 ;
- la dinamica termica determina la variazione temporale di temperatura in camera di combustione \dot{T}_3 .

Il sistema di 3 equazioni differenziali 3.63 accoppiate e equazioni algebriche fornisce quindi i tre residui \dot{T}_3 , \dot{n}_1 e \dot{n}_2 :

$$\begin{cases} \dot{n}_1 = \frac{P_{HPT} - P_C - P_{f_{HPT}}}{J_{HPT} n_1} \\ \dot{n}_2 = \frac{P_{PT} - P_{load} - P_{f_{PT}}}{J_{PT} n_2} \\ \dot{T}_3 = \frac{R T_3 k_3}{p_2 y_{CC} V_{CC} c_{p3}} [g_2 c_{p2} T_2 + g_b (c_{vb} h_b + \eta_b LHV) - (g_2 + g_b) c_{p3} T_3] \end{cases} \quad (3.63)$$

Il sistema di 3 equazioni differenziali 3.63 accoppiate e equazioni algebriche fornisce quindi i tre residui \dot{T}_3 , \dot{n}_1 e \dot{n}_2 . L'integrazione dei tre residui fornisce il vettore di stato del sistema.

I residui sono ordinabili in un vettore y' :

$$y' = \begin{pmatrix} \dot{n}_1 \\ \dot{n}_2 \\ \dot{T}_3 \end{pmatrix}$$

Definiamo y il vettore degli stati:

$$y = \begin{pmatrix} n_1 \\ n_2 \\ T_3 \end{pmatrix}$$

Alcune variabili del motore (rapporti di compressione ed espansione, portate), come già visto, dipendono attraverso delle mappe dagli stati del sistema, poiché alcune di esse sono parametrizzate su n_1 , mentre altre su T_4 , che è uno stretto parente di T_3 . La valutazione di queste grandezze richiede quindi la soluzione del sistema, ma a meno di utilizzare gli stati calcolati ad un'istante precedente, essa non è ancora disponibile all'inizio dell'iterazione: il sistema è quindi implicito.

Generalizzando quindi il problema si presenta nella forma:

$$y = f(y, y', t) \quad (3.64)$$

Per risolvere e integrare ad ogni passo un'equazione tipo la 3.64 è necessario un solutore per equazioni differenziali implicite in quanto il sistema è implicito, poiché lo stato dipende dal valore di se stesso allo stesso istante (ad esempio la portata in uscita dal plenum è funzione della accelerazione di rotazione \dot{n}_1) ed è quindi necessario procedere numericamente risolvendo il sistema fino a convergenza.

Per quanto riguarda i solutori impiegati, si rimanda al capitolo sull'implementazione.



Figura 3.10: Diagramma di flusso per il processo di generazione e adattamento delle mappe delle turbomacchine.

<i>Test Case</i>	<i>FUEL FLOW</i> [lb/h]	<i>P1</i> [PSI]	<i>T1</i> [°R]	<i>P5</i> [PSI]
1	140.1	14.37	516.7	14.37
2	297.2	14.17	515.6	14.43
3	372.0	14.16	508.3	14.46
4	458.4	14.09	508.0	14.60
5	560.4	14.02	507.2	14.63
6	694.4	13.92	507.2	14.72

Tabella 3.2: Dati ricavati da test NASA su motore GE t700 (prima parte).

3.5 Parametrizzazione

Per la determinazione di alcune grandezze di interesse per la simulazione delle turbomacchine sono state impiegate delle mappe, messe a punto partendo da alcuni dati sperimentali (riassunti nella tabella 3.2 e nella tabella 3.3) disponibili dal report NASA [5] e relativi a prove effettuate al banco sul propulsore General Electric t700 nella configurazione utilizzata sull'elicottero Sikorsky UH-60.

Ogni mappa è ottenuta interpolando i punti sperimentali attraverso una funzione lineare a tratti (usando la funzione *interp1* in linguaggio Matlab o *gsl interp linear* in linguaggio C usando le librerie GSL). Si è scelto di utilizzare

<i>Test Case</i>	n_1 [%]	n_2 [%]	g_{21} [lb/s]	p_{21} [PSI]	T_{21} [PSI]	T_4 [R]
1	65.9	52.6	3.2	58.0	832.0	1413
2	84.7	95.7	5.16	113.1	1026.0	1577
3	87.7	95.7	6.16	139.0	1081	1626
4	90.4	95.7	6.92	161.1	1127	1731
5	92.6	95.7	7.66	184.8	1173	1838
6	95.9	95.7	8.5	211.9	1228	1974

Tabella 3.3: Dati ricavati da test NASA su motore GE t700 (seconda parte).

una spezzata interpolante passante per i punti considerati per semplicità, ma l'utilizzo di una funzione interpolante con derivate continue è possibile e di agevole implementazione.

Tali mappe sono state utilizzate per la determinazione di:

- portata elaborata dal compressore: g_{21} ,
- rapporto di compressione del compressore: β_C ,
- rapporto di espansione della turbina di alta pressione: β_{HPT} ,
- rapporto di espansione della turbina di potenza: β_{PT} .

I parametri sui quali sono parametrizzati la portata e i rapporti di espansione appena elencati sono stati scelti in base ai dati sperimentali disponibili e in base al legame che intercorre fra i componenti del motore.

Per le mappe che decidono il funzionamento della prima parte del motore, ovvero per i componenti calettati sull'albero della turbina generatrice di gas, si è scelto di far variare le mappe con la velocità di rotazione n_1 . Tale scelta è pressoché obbligata in quanto il parametro che caratterizza il punto di funzionamento dei componenti compressore e turbina HPT è proprio il loro regime di rotazione. Si ha quindi:

$$\beta_{HPT} = f(n_1)$$

$$\beta_C = f(n_1)$$

$$g_{21} = f(n_1)$$

Per la turbina di potenza la scelta della variabile di parametrizzazione è stata più complessa, in quanto il regime di rotazione dell'albero della turbina di potenza si deve mantenere costante in ogni condizione di funzionamento del motore. Tale peculiarità, e la disponibilità di pochi parametri sperimentali (riportati tutti nelle tabelle 3.2 e 3.3), hanno portato a scegliere la temperatura T_4 all'uscita dalla turbina generatrice di gas come parametro caratterizzante il regime di funzionamento della turbina di potenza.

$$\beta_{PT} = f(T_4)$$

Le mappe ricavate dai dati NASA sono state adimensionalizzate rispetto ai valori nominali a regime, così da renderle estendibili a ogni possibile motore, noti i suoi parametri caratteristici (si veda a riguardo il diagramma in figura 3.10). Le dipendenze dei vari parametri sono quindi riferite ai valori dei parametri normalizzati:

$$\bar{\beta}_{HPT} = f(\bar{n}_1)$$

$$\bar{\beta}_C = f(\bar{n}_1)$$

$$\bar{g}_{21} = f(\bar{n}_1)$$

$$\bar{\beta}_{PT} = f(\bar{T}_4)$$

dove i parametri soprastegnati sono normalizzati rispetto al loro valore nominale:

$$\bar{a} = \frac{a}{a_{nominal}}$$

Per l'estensione delle mappe ad un qualsiasi propulsore devono essere noti dei parametri che si riferiscano al suo funzionamento nominale, mentre la forma della mappa viene generata a partire dal funzionamento del motore GE t700 in 6 situazioni di trim riprodotte al banco sperimentalmente.

L'elenco dei parametri da reperire per la scalatura della mappa ad un particolare propulsore sono:

- n_{1nom} : valore nominale della velocità di rotazione dell'albero della turbina generatrice di gas,
- n_{2nom} : valore nominale della velocità di rotazione dell'albero della turbina di potenza,
- β_{Cnom} : valore nominale del rapporto di compressione totale (tutti gli stadi) del compressore,
- β_{HPTnom} : valore nominale del rapporto di espansione complessivo della turbina generatrice di gas,
- β_{PTnom} : valore nominale del rapporto di espansione complessivo della turbina di potenza,
- $g_{compr_{nom}}$: valore nominale della portata elaborata dal compressore al regime nominale,
- gb_{nom} : valore nominale della portata di carburante,
- T_{4nom} : valore nominale della temperatura in uscita dalla turbina generatrice di gas.

Nella figura 3.13 è riportata la mappa normalizzata del rapporto di espansione della turbina di potenza: in ascissa è riportata la frazione di temperatura in uscita dalla turbina di potenza.

Nella figura 3.11 sono riportate le mappe del rapporto di compressione normalizzato del compressore e del rapporto di espansione normalizzato della

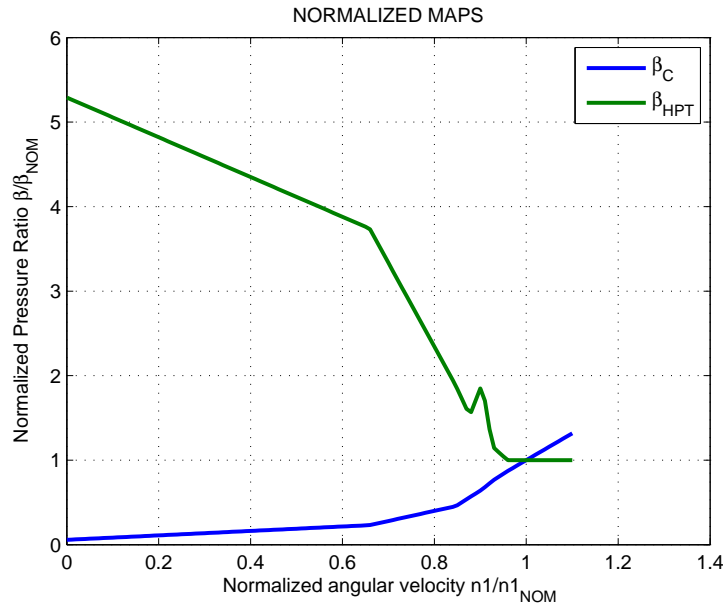


Figura 3.11: Rapporti di compressione/espansione normalizzati.

turbina HPT, entrambi parametrizzati sul regime di rotazione normalizzato \bar{n}_1 . Si noti come entrambe le mappe passino per il punto (1, 1), che è il punto di funzionamento nominale.

Nella figura 3.12 è riportata la mappa della portata di aria elaborata dal compressore in funzione di \bar{n}_1 .

3.5.1 Esempio di adattamento delle mappe per un generico motore

Nelle figure 3.15, 3.16, 3.11 si riporta un esempio di adattamento delle mappe normalizzate a un ipotetico motore, avente i seguenti parametri:

- $n_{1nom} = 75000 \text{ RPM}$,
- $n_{2nom} = 31000 \text{ RPM}$,
- $\beta_{Cnom} = 25.5$,
- $\beta_{HPTnom} = 0.25$,
- $\beta_{PTnom} = 0.58$,
- $g_{compressore_{nom}} = 7 \text{ kg/s}$,

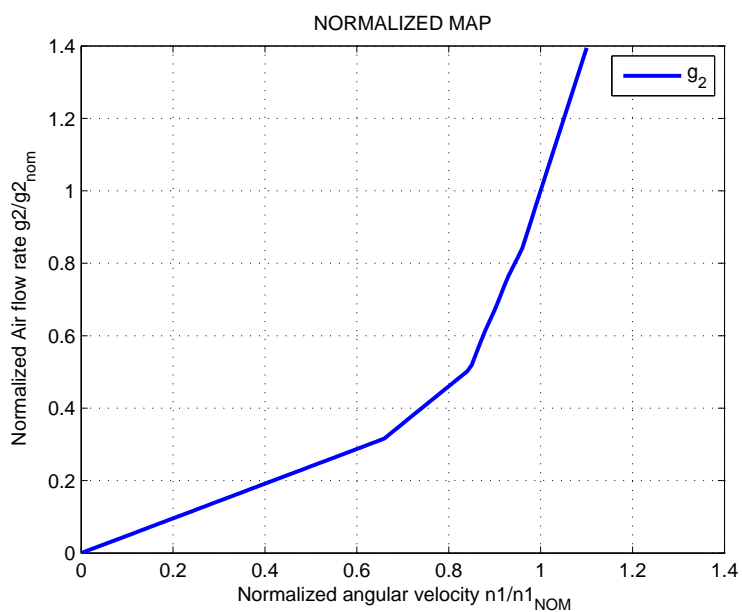


Figura 3.12: Mappa normalizzata della portata d'aria elaborata dal compressore (g_2) in funzione del regime di rotazione n_1 .

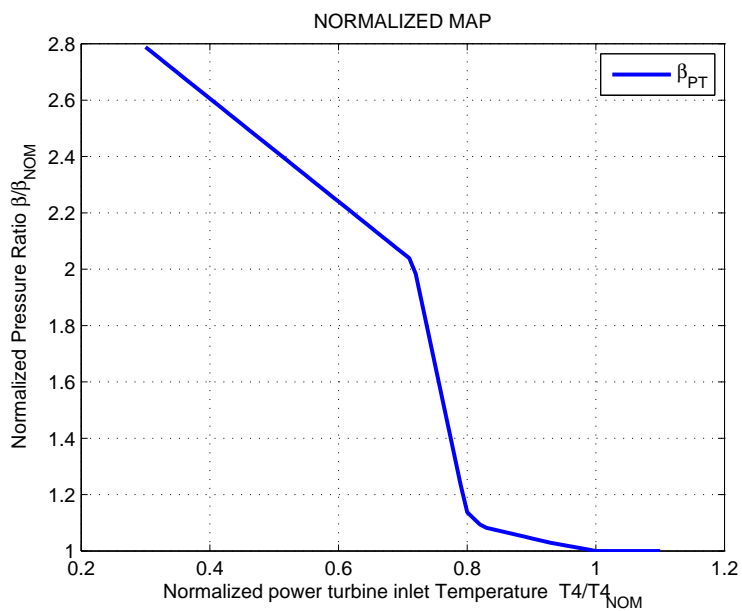


Figura 3.13: Mappa normalizzata del rapporto di espansione della turbina di potenza in funzione della temperatura in uscita dalla turbina HPT.

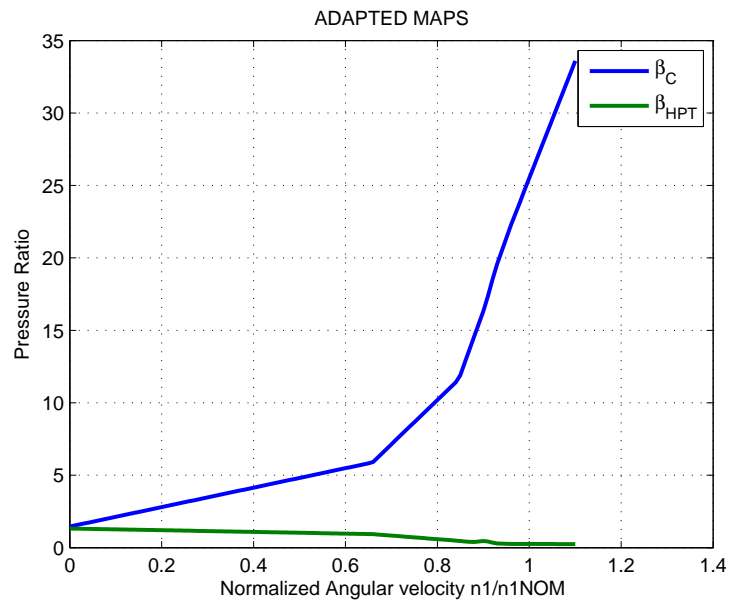


Figura 3.14: Mappa dei $\beta_{compressore}$ e β_{HPT} adattata ad un generico motore.

- $g_{b_{nom}} = 0.25 \text{ kg/s}$,
- $T_{4_{nom}} = 900 \text{ K}$.

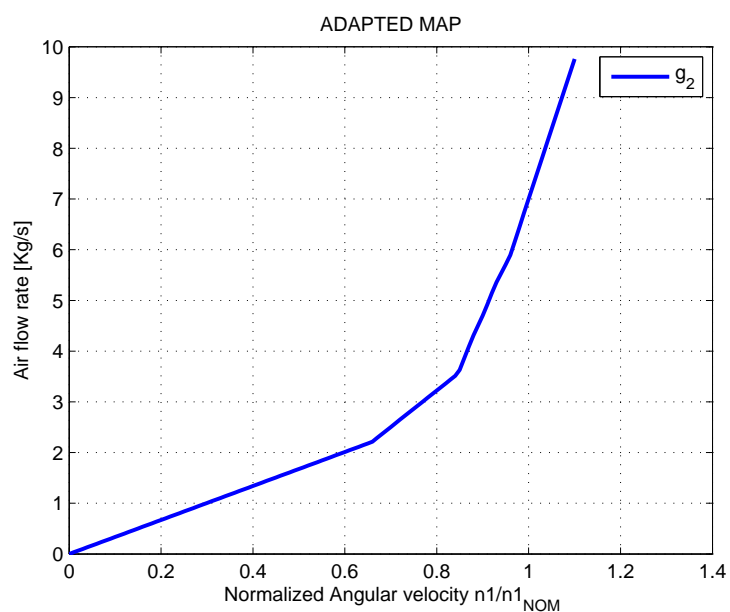


Figura 3.15: Mappa (adattata ad un ipotetico motore) della portata d'aria elaborata dal compressore (g_2) in funzione del regime di rotazione n_1 .

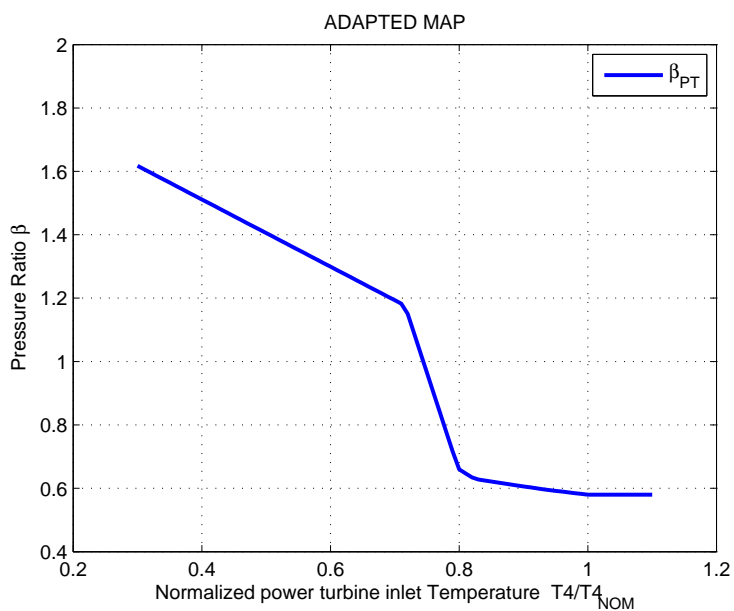


Figura 3.16: Mappa (adattata ad un ipotetico motore) del rapporto di espansione della turbina di potenza in funzione della temperatura in uscita dalla turbina HPT.

Capitolo 4

Implementazione

4.1 Requisiti del codice

L'implementazione del codice è avvenuta tenendo conto dei requisiti previsti per la sua esecuzione all'interno di un simulatore o comunque di un'interfaccia di simulazione :

- esecuzione *real-time*;
- facilità di interfacciarsi con diversi programmi (simulatori, interfacce grafiche stand-alone);
- possibilità di adattare rapidamente la simulazione all'applicazione motoristica desiderata.

Si è scelto di implementare il primo modello in *Matlab*, per i notevoli vantaggi che un linguaggio di alto livello orientato al calcolo numerico comporta nella prima parte di stesura di un codice di questo genere:

- velocità di scrittura;
- facilità di elaborazione dei dati;
- possibilità di sfruttare funzioni complesse già implementate (integratori, ecc.);
- documentazione vasta e chiara.

L'implementazione in Matlab presenta diversi limiti:

- difficoltà di interfacciare il codice Matlab con altri software;
- scarsa efficienza computazionale del linguaggio.

Per questi motivi il codice Matlab, già validato, è stato successivamente tradotto in *linguaggio C* attraverso la creazione di una **DLL** (Dynamic Link Library: libreria a collegamento dinamico). L'implementazione sotto forma di DLL in C permette di:

- utilizzare uno dei linguaggi di programmazione più efficienti;
- eseguire il codice chiamando la DLL da un qualsiasi altro programma;
- eseguire facilmente il codice su qualsiasi piattaforma;
- caricare selettivamente le varie DLL e da più programmi contemporaneamente;
- applicare modifiche al modello modificando soltanto la libreria DLL senza modificare tutti i codici chiamante.

La traduzione del codice dal linguaggio Matlab al linguaggio C sotto forma di DLL ha necessariamente comportato diversi adattamenti, ad esempio richiedendo la sostituzione delle funzioni native di Matlab (interpolazione attraverso INTERP1, POLYFIT, ecc.), con funzioni implementate in C. Dove possibile si sono utilizzate funzioni facente parte delle GSL (Gnu Scientific Libraries), librerie di funzioni per il calcolo scientifico ben documentate e molto vaste, sebbene non intuitive nell'uso come le funzioni native di Matlab. Dove non è stato possibile utilizzare funzioni o procedure già disponibili, si è creata la funzione relativa, inclusa poi nella DLL come funzioni di supporto.

4.2 Genesi e struttura del codice Matlab

Il codice implementato in Matlab ha subito diverse modifiche nel tempo, sia a livello di struttura che a livello di relazioni fisiche implementate. Si è dapprima implementato un modello fortemente parametrizzato, in cui la determinazione degli stati termodinamici delle varie sezioni del motore è determinato da alcune mappe ricavate dal funzionamento a regime del solito motore GE t700. Le varie versioni di questo tipo di modello molto parametrizzato sono state identificate con le lettere dell'alfabeto greco: α, β, γ . Il codice Matlab nella sua versione γ (dipendente da mappe parametriche per la determinazione di ogni stato termodinamico) è stata validata confrontando i risultati che si ottenevano dalle simulazioni con dati sperimentali ricavati dalla letteratura.

Successivamente, validato il *codice* γ , si è deciso di rendere il modello meno dipendente da parametri e quindi di modellare le trasformazioni che il fluido

subisce all'interno del ciclo termodinamico come espansioni e compressioni reali. Il codice a cui si è giunti finalmente al termine delle fasi di modellazione, implementazione, taratura e validazione è stato denominato *codice ISO*, a rimarcare il fatto che le trasformazioni che avvengono nelle turbomacchine sono compressioni o espansioni isoentropiche corrette con un rendimento. Tale modello è meno vincolato al sistema di partenza da cui sono stati ricavati i parametri, il motore GE t700, e quindi ci si aspetta che riproduca meno fedelmente del *modello* γ il comportamento di tale propulsore, ma sia meglio adattabile ad altri sistemi.

Nel capitolo riguardante la validazione si confronteranno i risultati ottenuti dalle simulazioni effettuate con il codice ISO e con il codice γ e i dati sperimentali, mettendo quindi in relazione gli errori dei due modelli.

4.2.1 Modello γ - Matlab

Il *modello* γ presenta una struttura modulare, dove ogni componente del sistema è rappresentato da una funzione. Le variabili in uscita dalla funzione rappresentano gli ingressi per un'altra funzione: ad esempio la portata d'aria in uscita dalla funzione compressore sarà un ingresso per la funzione plenum, che viene eseguita dopo la funzione compressore. Ordinando in questo modo il codice è stato possibile isolare i banchi del codice provando separatamente le varie funzioni. I vari moduli che compongono il modello sono:

- compressore,
- plenum1,
- combustore,
- HPT,
- PT,
- spool 1,
- spool 2.

I moduli che al loro interno implementano una dinamica (*Combustore* - T3, *Spool 1* - n_1 , *Spool 2* - n_2), restituiscono in uscita un residuo, ovvero la derivata dello stato del sistema, e ad ogni iterazione i residui verranno integrati dall'integratore per fornire i nuovi stati e quindi la nuova soluzione.

Gli stati termodinamici (pressioni, temperature e portate) relativi ai blocchi che implementano le turbomacchine in questo modello vengono generati

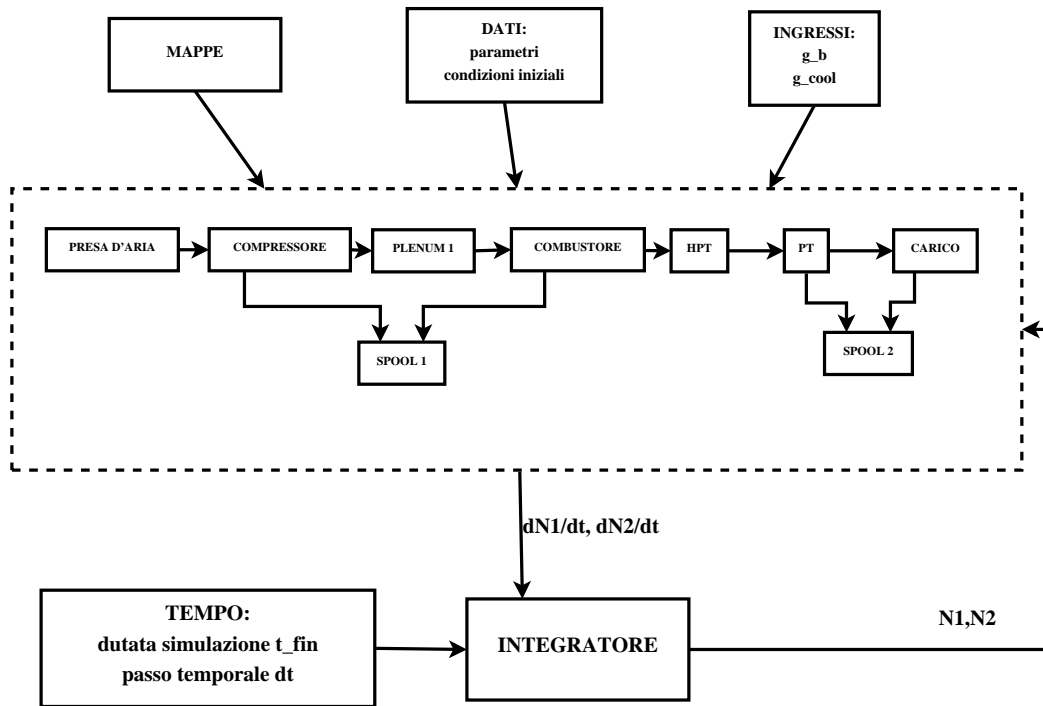


Figura 4.1: Schema del codice γ implementato in Matlab.

tramite delle mappe, che forniscano la grandezza desiderata in funzione del parametro utilizzato nella modellazione. Così la temperatura, la pressione e la portata in uscita dal *compressore* sono funzioni della velocità di rotazione dell'albero della turbina generatrice di gas attraverso una mappa:

$$T_{21} = f(n_1)$$

$$p_{21} = f(n_1)$$

$$g_{21} = f(n_1)$$

Anche la temperatura e la pressione (o il rapporto di espansione) in uscita dalla *turbina HPT* sono funzioni della velocità di rotazione dell'albero della turbina generatrice di gas:

$$T_4 = f(n_1)$$

$$\beta_{HPT} = f(n_1)$$

Mentre per quanto riguarda la temperatura in uscita dalla *turbina PT*, e il rapporto di espansione della turbina *PT*, essi sono funzioni della temperatura T_4 in uscita dalla turbina *HPT*:

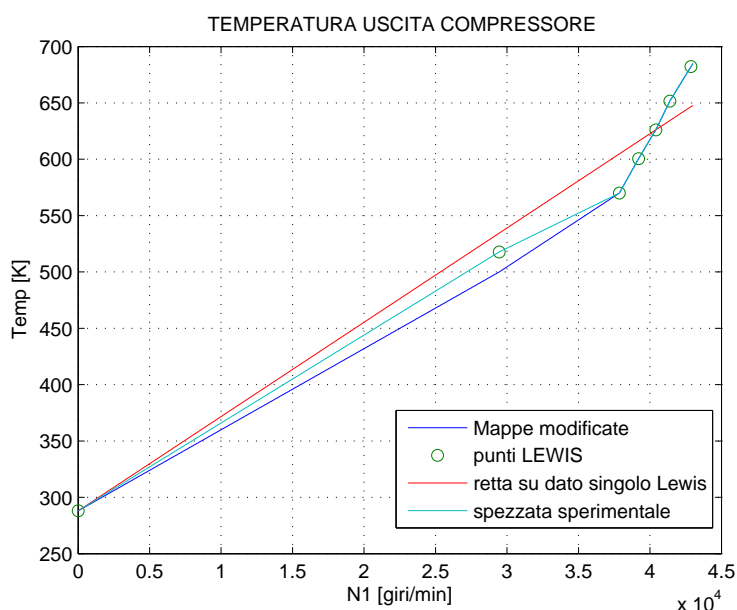


Figura 4.2: Mappa della temperatura in uscita dal compressore in funzione del regime di rotazione n_1 nel modello γ per il motore GE t700.

$$T_5 = f(T_4)$$

$$\beta_{PT} = f(T_4)$$

Le mappe sono funzioni continue della variabile su cui si è parametrizzato il dato di interesse. Esse interpolano i dati sperimentali dei 6 casi di trim del report [5]. Come funzione interpolante è stata scelta una spezzata (funzione lineare a tratti): si sono confrontati i risultati ottenuti attraverso l'impiego di altre funzioni (spline cubiche, polinomi di ordine 2,3,4), ma dal momento che le differenze di funzionamento del codice erano minime si è scelto di utilizzare semplici spezzate lineari per semplicità della formulazione. Si è anche pensato di utilizzare come funzioni interpolanti dei polinomi che minimizzassero l'errore dello stato termodinamico in relazione al dato sperimentale (mappe modificate), opzione che è stata poi scartata perché introduceva nel modello un grado di aleatorietà che difficilmente avrebbe trovato riscontro in un'applicazione diversa da quella del motore GE t700 utilizzato per la generazione delle mappe e poi la valutazione degli errori. Un esempio di mappa per il modello γ è visibile in figura 4.2, dove è riportata la temperatura in uscita dal compressore in funzione del regime di rotazione del primo albero.

La modularità del codice gamma è apprezzabile nel listato che ne richiama in successione i componenti ad ogni iterazione:

```

% PRESA D'ARIA
p1(i)=p_tot(i);
T1(i)=T_tot(i);
rho(i)=rho_0;
%COMPRESSORE
[p21(i),T21(i),g21(i),P_c(i)]=compressore(p1(i),T1(i),T21(i-1),...
      n1(i-1),pg2s,pbetac,pT21,P_pt(i-1),P_f_hpt(i-1));
%PLENUM 1
[p2(i),T2(i),g2(i),g_cool(i),tau_p(i)]=plenum1(p21(i),T21(i),g21(i),...
      g_b(i),dt,p2(i-1),V_p1);
%CAMERA DI COMBUSTIONE
[g_gas(i),p3(i),T3(i),dT3(i),iter_comb(i),err_comb(i)]=combustore(p2(i),T2(i)
g_b(i),T3(i-1),dt,LHV,T3max,dT3(i-1));
%TURBINA GENERATRICE DI GAS
[p4(i),T4(i),P_hpt(i),g4(i)] = hpt(p3(i),T3(i),T4(i-1),g_gas(i),...
      g_cool(i),T21(i),pT4, n1(i-1),g_b(i),pbetahpt);
%TURBINA DI POTENZA
[p5(i),T5(i),P_pt(i)]=powerturbine(p4(i),T4(i),T5(i-1),g4(i),p_atm,...
n1(i),g_b(i),pbetapt);
%CARICO
[C_ass(i),vel_rotore(i)]=caricoLEWIS(n2(i-1),pcl,g_b(i));
P_load(i)=C_ass(i)*n2(i-1)*0.104719755; %[W]potenza carico
%DINAMICA ALBERI DI TRASMISSIONE
%modellazione perdite per attrito:
P_f_hpt(i)=abs(P_hpt(i)*0.01); %[W]pot perdita in attr.mecc=1/100 Pot hpt
P_f_pt(i)=abs(P_pt(i)*0.01); %[W]pot perdita in attr.mecc=1/100 Pot pt
%calcolo accelerazioni angolari:
[n1_p(i)]=spool1(P_hpt(i),P_c(i),P_f_hpt(i),J_hpt,n1(i-1));
[n2_p(i)]=spool2(P_pt(i),P_load(i),P_f_pt(i),J_pt,n2(i-1));
%INTEGRAZIONE
%Crank Nicholson
n1(i)=n1(i-1)+0.5*dt*(n1_p(i-1)+n1_p(i));
n2(i)=n2(i-1)+0.5*dt*(n2_p(i-1)+n2_p(i));

```

Si noti come essendo la formulazione del modello gamma esplicita, sia possibile integrare le 2 equazioni dinamiche della meccanica del modello esplicitamente con il metodo dei trapezi (o di Crank-Nicholson). La dinamica termica infatti, ritenuta molto più veloce delle dinamiche meccaniche, viene risolta all'interno del componente Bruciatore, con un limite di iterazioni massime per non sfiorare oltre il tempo massimo di calcolo per il passo temporale

considerato. Nel caso il componente non converga sulla temperatura T_3 nel numero di iterazioni massimo con la tolleranza assegnata, l'utente ne viene informato con un segnale di attenzione.

Il modello così implementato ha fornito buoni risultati se paragonato ai dati sperimentali disponibili. La convergenza si è sempre rivelata soddisfacente con passi temporali dell'ordine del decimo di secondo, sebbene sia stata rilevata una certa difficoltà quando venivano forniti ingressi con variazioni repentine della portata di carburante.

Si raccomanda quindi, qualora interessi implementare il codice del modello gamma, di tarare il FADEC in modo che la derivata temporale della portata di carburante immessa in camera di combustione sia sempre in modulo minore del valore che impedisce la convergenza del sistema.

4.2.2 Modello ISO - Matlab

Il *modello ISO* implementa espansioni e compressioni reali per determinare gli stati termodinamici del fluido dopo il passaggio nelle turbomacchine ed utilizza delle mappe parametriche solo per la determinazione di portata d'aria elaborata dal compressore e rapporti di espansione o compressione delle tre turbomacchine. Il modello numerico è quindi quello del capitolo relativo alla modellazione, con le dinamiche riassunte nel sistema 3.63.

Le funzioni implementate nel modello ISO sono:

- *ENGINEnew2.m*: dinamica del motore;
- *ENGINEparameters.m*: personalizzazione dei parametri caratteristici del motore;
- *NEWMAPS.m*: generazione delle mappe normalizzate utilizzate nelle turbomacchine;
- *INITIALconditions.m*: assegnamento delle condizioni iniziali per la simulazione;
- *FUELflow.m*: assegnamento degli ingressi del motore (portata di combustibile g_b);
- *ENGINEload.m*: modello di carico che si interfaccia col motore.

L'implementazione della dinamica del *modello ISO* è avvenuta, a differenza del modello γ , in un'unica funzione principale denominata *ENGINEnew2.m* la cui sintassi è del tipo:

```
function [ep] = ENGINEnew2(t,y,yp)
```

Il vettore $[ep]$ raccoglie la differenza fra le derivate degli stati calcolate dalla funzione $(\dot{T}_3, \dot{n}_1, \dot{n}_2)$ e i residui (yp) calcolati al passo precedente e forniti in ingresso alla funzione insieme al valore degli stati (y). Il vettore $[ep]$ è quindi composto da:

$$e_{\dot{n}_1} = \dot{n}_1 - yp(1) \quad (4.1)$$

$$e_{\dot{n}_2} = \dot{n}_2 - yp(2) \quad (4.2)$$

$$e_{\dot{T}_3} = \dot{T}_3 - yp(3) \quad (4.3)$$

e risulta quindi

$$[ep] = [e_{\dot{n}_1}; e_{\dot{n}_2}; e_{\dot{T}_3}]$$

Il vettore $[ep]$ è quello che servirà alla funzione integratrice *ode15i* per iterare e integrare i residui del sistema.

Le condizioni iniziali di tutte le variabili del motore vengono assegnate all'interno della funzione *INITIALconditions.m*; l'assegnamento delle condizioni iniziali deve essere consistente, ovvero deve essere:

$$[ep] = \mathbf{0}$$

o in altri termini, facendo riferimento al sistema 3.63:

$$f(t_0, y_0, yp_0) = 0$$

Qualora gli stati termodinamici iniziali siano lontani da un punto consistente per il modello, durante la fase di integrazione si andrà incontro a problemi di convergenza. Per ovviare a questo problema si può calcolare un punto di partenza consistente perturbando il modello fino a fare raggiungere al sistema una condizione iniziale consistente. In Matlab la funzione *decic.m* permette di calcolare la condizione iniziale consistente da cui far partire l'integrazione con le funzioni *ode*; è possibile specificare alla funzione *decic* quali stati è possibile variare e quali invece devono essere mantenuti fissi.

Le mappe relative al funzionamento delle turbomacchine vengono generate attraverso la funzione *NEWMAPS.m*: essa genera delle mappe normalizzate a partire dai dati sperimentali disponibili dal report [5]. Tali mappe andranno poi scalate attraverso i parametri caratteristici del motore considerato nella singola simulazione. Le mappe necessarie al funzionamento del modello ISO sono:

- mappa per il rapporto di compressione normalizzato del compressore:

$$\bar{\beta}_{Compressore} = f(\bar{n}_1)$$

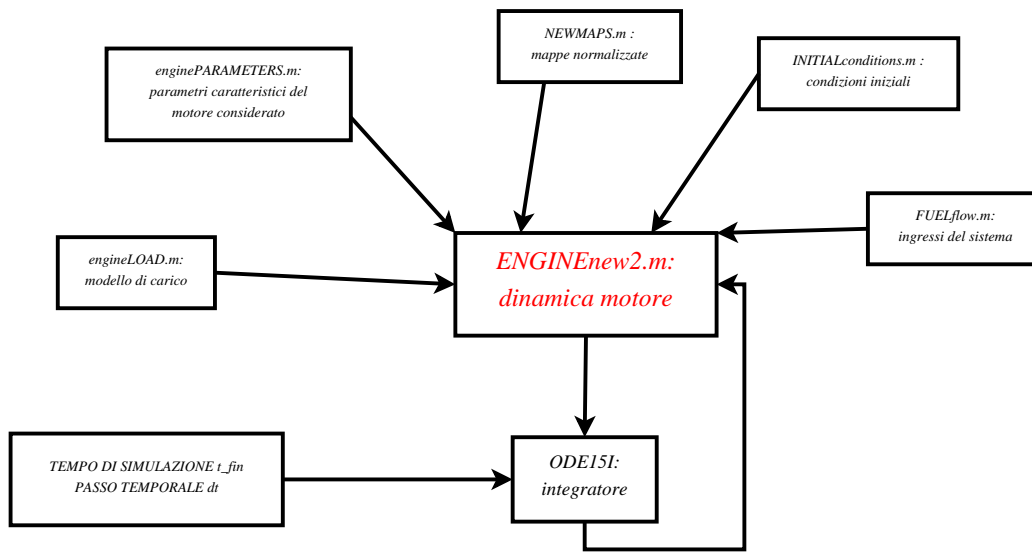


Figura 4.3: Schema delle funzioni implementate nel codice ISO.

- mappa per il rapporto di espansione normalizzato della turbina HPT:

$$\bar{\beta}_{HPT} = f(\bar{n}_1)$$

- mappa per il rapporto di espansione normalizzato della turbina PT:

$$\bar{\beta}_{PT} = f(\bar{T}_4)$$

- mappa per la portata d'aria normalizzata elaborata dal compressore:

$$\bar{g}_{Compressore} = f(\bar{n}_1)$$

Per essere utilizzate le mappe devono essere scalate con i parametri caratteristici del motore considerato. Tali parametri devono essere assegnati all'interno della funzione *ENGINEparameters.m*:

- n_{1nom} : valore nominale della velocità di rotazione dell'albero della turbina generatrice di gas;
- n_{2nom} : valore nominale della velocità di rotazione dell'albero della turbina di potenza;
- J_{HPT} : momento d'inerzia dell'albero della turbina generatrice di gas, comprensivo di tutti gli elementi ad esso collegati;

- J_{PT} : momento d'inerzia dell'albero della turbina di potenza, comprensivo di tutti gli elementi ad esso collegati;
- V_{CC} : volume della camera di combustione;
- β_{Cnom} : valore nominale del rapporto di compressione totale (tutti gli stadi) del compressore;
- β_{HPTnom} : valore nominale del rapporto di espansione complessivo della turbina generatrice di gas;
- β_{PTnom} : valore nominale del rapporto di espansione complessivo della turbina di potenza;
- $g_{comprnom}$: valore nominale della portata elaborata dal compressore al regime nominale;
- gb_{nom} : valore nominale della portata di carburante;
- $T4_{nom}$: valore nominale della temperatura in uscita dalla turbina generatrice di gas.

La funzione *FUELflow.m* assegna la storia della portata di carburante per la durata della simulazione. Nelle prove di validazione della termodinamica del motore è stata assegnata direttamente una storia temporale alla portata del carburante, mentre nell'utilizzo del modello all'interno di un codice di meccanica del volo o di simulazione, tale funzione verrà sostituita da un regolatore automatico, che in base ad alcuni ingressi regolerà la portata di carburante con una specifica logica di controllo.

Il carico che si interfaccia al motore è stato implementato nella funzione *ENGINEload.m*. Il modello di carico è quello presentato nel capitolo relativo alla modellazione, dove la potenza richiesta dal carico è proporzionale alla velocità di rotazione n_2 tramite la 3.57.

4.2.3 Funzioni di supporto

Per il funzionamento del modello ISO sono necessarie alcune funzioni di supporto, che implementano nel modello le proprietà del gas già descritte nel capitolo relativo alla modellazione.

La funzione *aria2.m* permette di determinare le proprietà termodinamiche dell'aria secca. La funzione ha una sintassi del tipo:

```
[k,cp,cv]=aria2(T)
```

L'*help* scritto per la funzione ne spiega il funzionamento:

```
ARIA2 [k,Cp,Cv]=aria2(T) funzione che fornisce le proprietà
    termodinamiche dell'aria, assunta gas perfetto,
    ricevendo in ingresso la temperatura dell'aria alla
    quale si vogliono valutare.
    Cp = [J/KgK] calore specifico a pressione costante
    Cv = [J/KgK] calore specifico a volume costante
    k = [adim] rapporto calori specifici
    T = [K] temperatura dell'aria
```

ARIA2 è ottenuta interpolando dati sperimentali con un polinomio interpolante di ordine 2.

Similmente a quanto fatto per l'aria, si è impiegata una funzione che calcolasse le proprietà termodinamiche dei gas di combustione. Il modello impiegato per i gas combustibili è quello messo a punto da Guha nel 2000 (riferimento [12]). La funzione di riferimento per i gas combustibili è la funzione *kerosene.m*. Essa ha la sintassi:

```
[k,cp,R] = kerosene(T,FAR)
```

Gli ingressi della funzione sono la temperatura del gas T e il rapporto aria/combustibile FAR (Fuel to Air Ratio). Le uscite della funzione sono: k il rapporto fra calori specifici, c_p il calore specifico a pressione costante (unità di misura $[\frac{J}{kgK}]$), c_v il calore specifico a volume costante, R la costante dei gas combustibili.

4.3 Struttura del codice C

Il codice C è stato scritto sotto forma di DLL, ovvero tutte le funzioni necessarie al modello sono raccolte all'interno di una libreria a collegamento dinamico che andrà poi caricata ed eseguita in fase di compilazione ed esecuzione della simulazione da un codice chiamante.

Il file a cui si farà riferimento nel seguito è appunto il file sorgente della libreria DLL *EngineModel.c*, che al suo interno contiene le funzioni implementate nel modello e alcuni file *header* (con estensione .h) che richiamati nella libreria mettono a disposizione delle funzioni e delle definizioni definite al loro interno.

Le funzioni contenute nella libreria *EngineModel.dll* sono:

- **EngineModel**: funzione principale che implementa la dinamica del motore;

- **kAIR**: riceve in ingresso la temperatura e fornisce in uscita il rapporto fra i calori specifici dell'aria;
- **cpAIR**: riceve in ingresso la temperatura e fornisce in uscita il calore specifico dell'aria a pressione costante;
- **cvAIR**: riceve in ingresso la temperatura e fornisce in uscita il calore specifico dell'aria a volume costante;
- **kGAS**: riceve in ingresso la temperatura e il rapporto aria/combustibile e fornisce in uscita il rapporto fra i calori specifici dei gas combustibili di kerosene e aria con il modello di Guha;
- **cpGAS**: riceve in ingresso la temperatura e il rapporto aria/combustibile e fornisce in uscita il calore specifico a pressione costante dei gas combustibili di kerosene e aria con il modello di Guha;
- **MAPgeneration**: genera degli oggetti di tipo `gsl_interp` contenenti le mappe normalizzate per il funzionamento delle turbomacchine;
- **GetEngineStates**: memorizza il valore degli stati del sistema;
- **GetEngineOutput**: memorizza in un vettore il valore delle uscite desiderate;
- **EngineModelInit**: inizializza i parametri utilizzati dalla funzione *EngineModel*;
- **EngineModelStep**: fa compiere al sistema un passo di integrazione;

4.3.1 Funzione Principale: EngineModel

La funzione principale, che implementa le relazioni fisiche del modello di motore è *EngineModel*. La funzione viene dichiarata nel modo seguente:

```

ENG_DLL_EXPORT int EngineModel(
    double dXdt[ENG_STATES_NUM],
    double Y[ENG_OUT],
    double X[ENG_STATES_NUM],
    double U[ENG_CONTROLS],
    gsl_interp * pg2n,
    gsl_interp * pbetacn,
    gsl_interp * pbetahptn,
    gsl_interp * pbetaptn,

```

```
engine_struct *engine_par,  
double Pload)
```

La funzione `EngineModel` riceve in ingresso:

- il valore dei residui $dXdt$,
- il valore degli stati X ,
- una struttura *engine par* contenente i parametri caratteristici del motore,
- il valore degli ingressi U (portata di carburante e portata d'aria spillata),
- il vettore Y che raccoglie le uscite del sistema,
- Quattro oggetti di tipo *gsl interp* che contengono le mappe necessarie alle turbomacchine,
- il valore di potenza $Pload$ richiesta dal carico.

La funzione `EngineModel` restituisce in uscita una variabile di tipo *int* che rappresenta un codice per la buona esecuzione o meno della funzione. In realtà le uscite della funzione, coerentemente con la sintassi del linguaggio C, vengono memorizzate nei vettori in ingresso e sono:

- Il vettore dei residui $dXdt$, contenente le derivate dei tre stati del sistema,
- Il vettore delle uscite Y , contenente i valori delle variabili interessanti da esportare a ogni iterazione,
- Il vettore degli stati X .

I residui andranno poi forniti all'integratore insieme al valore degli stati per determinare i valori degli stati al passo successivo. Le uscite saranno invece utili nel caso sia necessario valutarne l'andamento temporale, ad esempio elaborandole in forma grafica.

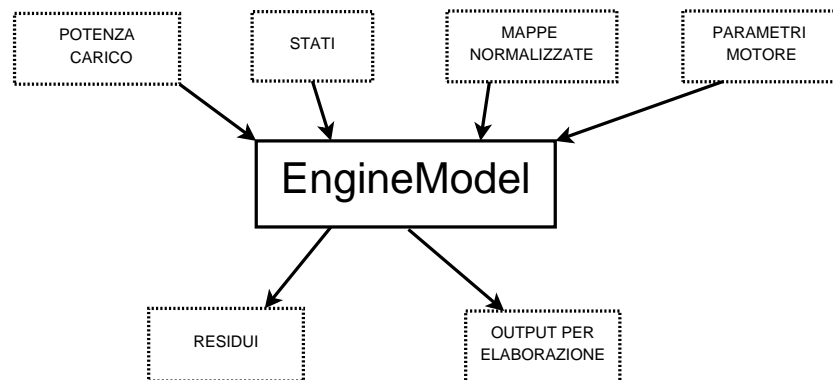


Figura 4.4: Funzione EngineModel.c : ingressi e uscite.

4.3.2 Funzioni di supporto

All'interno della DLL sono implementate diverse funzioni di supporto:

- kAIR
- cpAIR
- cvAIR
- kGAS
- cpGAS
- MAPgeneration
- GetEngineStates
- GetEngineOutput
- EngineModelInit
- EngineModelStep

Tali funzioni consentono di: determinare le proprietà termodinamiche dei fluidi di lavoro (funzioni *kAIR*, *cpAIR*, *cvAIR*, *kGAS*, *cpGAS*), generare le mappe necessarie alla simulazione delle turbomacchine (funzione *MAPgeneration*), interfacciarsi con il programma di simulazione o l'eventuale interfaccia grafica (funzioni *GetEngineStates* e *GetEngineOutput*), avviare la simulazione (funzione *EngineModelInit*) o farla procedere nel tempo (funzione *EngineModelStep*).

4.4 Metodo di soluzione e di integrazione

Le equazioni che governano il modello sono un sistema di equazioni miste algebrico differenziali implicite (eq: 3.63), ed è quindi necessario tenere conto dell'implicità del problema nella scelta del solutore e dell'integratore temporale.

Linguaggio Matlab Per quanto riguarda l'implementazione in Matlab, per la scelta dell'integratore si è fatto riferimento alla documentazione del programma relativa alle equazioni differenziali ordinarie. Matlab implementa una famiglia di solutori per equazioni differenziali ordinarie: le funzioni ODE (Ordinary Differential Equations). Fra tutte le funzioni ODE, l'unica adatta ai problemi impliciti è la funzione ODE15i; essa ha una struttura del tipo :

```
[TOUT, YOUT] = ODE15I(ODEFUN, TSPAN, Y0, YP0)
```

Fra le uscite della funzione: TOUT è un vettore temporale che raccoglie gli istanti a cui sono state calcolate le soluzioni YOUT. Analizzando gli ingressi invece: ODEFUN è la funzione che implementa l'equazione differenziale implicita da integrare, TSPAN l'intervallo temporale in cui integrare, Y0 un vettore che raccoglie le condizioni iniziali, e YP0 un vettore che raccoglie le condizioni iniziali delle derivate.

Le condizioni iniziali devono essere consistenti, ovvero

$$f(T0, Y0, YP0) = 0$$

La funzione utilizza un metodo di integrazione di ordine variabile a seconda della funzione integranda. È comunque possibile impostare delle opzioni per la tolleranza o il tipo di integratore con il comando OPTIONS.

Linguaggio C Nell'implementazione del modello in linguaggio C, data la flessibilità del codice, si è lasciata libertà di scelta dell'integratore da utilizzare. Nella scelta si dovrà tenere conto dell'implicità del sistema e dei tempi caratteristici di ognuna delle dinamiche considerate. La funzione Engine-Model fornisce in uscita i tre residui del sistema: i primi due sono relativi alle dinamiche meccaniche (n_1 ed n_2) ed il terzo è relativo alla dinamica termica (T_3). A seconda della specifica applicazione considerata i tempi caratteristici associati ad una o all'altra dinamica varieranno, ad esempio a seconda delle inerzie associate agli alberi per quanto riguarda le dinamiche meccaniche, o a seconda del volume della camera di combustione per quanto riguarda le dinamiche termiche. Si può comunque affermare che, per applicazioni su propulsori simili al GE t700, ci si aspetta che la dinamica più veloce

sia quella termica associata alla camera di combustione. La scelta del tempo di integrazione e del metodo di soluzione andrà quindi effettuata basandosi sul tempo caratteristico di tale dinamica ed eventualmente sui tempi caratteristici delle altre dinamiche che l'integratore dovrà gestire. Ad esempio in un programma di simulazione della dinamica di un elicottero i tempi caratteristici delle dinamiche del motore andranno messi in relazione con i tempi caratteristici del rotore principale, che a seconda del tipo di modello impiegato richiederà un passo di integrazione più o meno breve, che comunque è in relazione con la frequenza di rotazione del rotore.

$$\Delta t_{integratore} < \min(\tau_{motore}, \tau_{rotore})$$

La scelta del passo di integrazione è in questo caso una decisione che verrà presa durante l'integrazione del modello di motore all'interno del programma di simulazione.

4.5 Interfaccia grafica

Data la flessibilità dell'implementazione del codice in linguaggio C sotto forma di DLL, si è cercato di mettere a punto un'interfaccia grafica che permettesse di simulare i vari motori off-line senza interfacciarsi con nessun codice di meccanica del volo. Tale interfaccia grafica deve essere abbastanza intuitiva da permettere all'utente di ottenere in breve tempo le informazioni sul funzionamento del motore, avendo inizialmente personalizzato la simulazione tramite i parametri caratteristici del propulsore che si desidera simulare.

L'interfaccia grafica sarà composta da:

- un pannello principale con i componenti del motore,
- una serie di pannelli che riportino gli andamenti temporali delle grandezze di interesse durante la simulazione,
- un pannello di selezione dei parametri caratteristici del motore da simulare.

Per l'implementazione dell'interfaccia grafica si è utilizzato il software LAZARUS. Lazarus è un ambiente ad interfaccia integrata rilasciato e sviluppato dal compilatore a codice sorgente aperto Free Pascal. Il linguaggio con cui viene sviluppata l'interfaccia grafica è quindi il Pascal orientato agli oggetti.

4.6 DLL - Dinamic Link Library

In informatica, una *dynamic-link library* (libreria a collegamento dinamico) è una libreria software che viene caricata dinamicamente in fase di esecuzione, invece di essere collegata staticamente a un eseguibile in fase di compilazione. Queste librerie sono note con l'acronimo *DLL*, che è l'estensione del file che hanno nel sistema operativo Microsoft Windows, o anche con il termine librerie condivise (da *shared library*, usato nella letteratura dei sistemi Unix). Nei sistemi che usano ELF come formato dei file eseguibili, come ad esempio Solaris o Linux, sono anche note come *.so*, abbreviazione di *Shared Object*.

La separazione del codice in librerie a collegamento dinamico permette di suddividere il codice eseguibile in parti concettualmente separate, che verranno caricate solo se effettivamente necessarie. Inoltre, una singola libreria, caricata in memoria, può essere utilizzata da più programmi, senza la necessità di essere nuovamente caricata, il che permette di risparmiare le risorse del sistema. Questo metodo di *loading on demand* consente, inoltre, installazioni parziali di un sistema software, in cui sono effettivamente presenti sulla memoria di massa solo le librerie associate alle funzioni che l'utente desidera utilizzare, come selezionate in fase di installazione: in questo modo ad esempio viene caricato il sistema operativo Microsoft Windows. Un altro vantaggio è la possibilità di aggiornare un programma modificando solo le DLL: inserendo una versione diversa della DLL, che contiene ad esempio dei bug-fix, tutti i programmi che la usano saranno automaticamente aggiornati senza bisogno di essere ricompilati. Il principale svantaggio è legato al fatto che una nuova versione di una DLL potrebbe effettuare dei cosiddetti *breaking changes*, in modo volontario o inconsapevolmente, a causa di bug nella nuova versione. Un *breaking change* è un cambiamento critico nel comportamento del codice della funzione che la rende non più compatibile con le convenzioni in uso (ad esempio, una funzione che prima restituiva NULL in caso di errore nei parametri e che ora restituisce un valore non nullo). Ancora più critico il caso in cui un programma di installazione sovrascrive una DLL con una versione più vecchia.

Le librerie dinamiche vengono caricate dal sistema operativo all'interno dello spazio di memoria del processo che le ha richieste. In questo modo l'accesso al codice della DLL avrà prestazioni quasi equivalenti a quelle del codice dell'applicazione stessa o del codice delle librerie statiche (in seguito si vedrà perché quasi equivalenti). Per evitare che il codice dell'applicazione e quello della DLL occupino la stessa posizione in memoria, il linker dovrà predisporre la DLL per la rilocazione. In pratica, il sistema operativo determina un'area di memoria disponibile e rimappa ogni riferimento alla memoria contenuto nel codice della DLL. Siccome quest'operazione richiede tempo, ogni

DLL dispone di un proprio indirizzo di base ideale: la rilocazione sarà necessaria solo se a questo indirizzo predeterminato è già stata mappata una precedente DLL.

4.7 GSL - Gnu Scientific Libraries

Durante l'implementazione in linguaggio C si è fatto ricorso alle librerie open-source GSL (Gnu Scientific Libraries), che permettono di utilizzare funzioni di calcolo numerico già implementate e validate. Si è fatto riferimento al manuale ([14]) per la documentazione relativa ad ognuna delle librerie utilizzate. Le librerie GSL sono librerie software per calcolo numerico molto usate nella matematica applicata e nelle scienze; sono scritte in linguaggio C, ma sono disponibili wrappers per altri linguaggi di programmazione. Le GSL sono parte del progetto GNU Project e sono distribuite sotto la licenza GNU *General Public License*.

Le librerie utilizzate in questo lavoro sono:

- Interpolazione: *gsl-interp*;
- Soluzione di equazioni implicite:

L'inclusione delle librerie considerate viene operata attraverso un file header nel nostro file DLL, includendo in modo selettivo le singole librerie necessarie, ad esempio con il comando:

```
#include <gsl/gsl_interp.h>
```

Per l'utilizzo delle funzioni relative si rimanda alla documentazione già citata.

Capitolo 5

Validazione

La validazione del modello termodinamico è stata effettuata confrontando i dati ottenuti dalle simulazioni con i dati sperimentali disponibili dal report [5], con cui si è confrontata gran parte della letteratura successiva per la validazione dei propri lavori. Per la validazione della termodinamica del modello, in anello aperto, si è proceduto in due modi:

- confronto degli stati termodinamici in diverse condizioni di regime;
- confronto della risposta del motore a un ingresso di carburante a gradino;

I dati sperimentali disponibili non coprono tutte le grandezze caratteristiche del motore e pertanto la validazione è stata possibile solo per le grandezze di cui si è trovato un riferimento.

5.1 Confronto con dati sperimentali a regime

Per potere confrontarsi con i dati sperimentali del motore GE t700, relativi ad alcune prove al banco, è necessario simulare le medesime condizioni in cui si trovava il motore sperimentale. Il confronto è stato possibile solo sui dati termodinamici disponibili, e quindi il riscontro della buona aderenza o meno del modello al comportamento del vero propulsore non è completa, ma viene effettuata solo nel sottospazio degli stati termodinamici disponibili.

Per ottenere una risposta a regime del propulsore si è scelto di simulare un funzionamento di 100 secondi, con un passo temporale di $\Delta t = 0.1$ s, così da essere sicuri che tutti i transitori associati alle dinamiche fossero terminati e che gli stati avessero ormai raggiunto stabilmente il loro valore a regime.

Si è scelto di confrontare i dati sperimentali disponibili e quelli forniti dai due modelli messi a punto in Matlab: il *modello γ* e il *modello ISO*.

L'errore percentuale del generico stato termodinamico x ottenuto dalle simulazioni, è definito come:

$$err\% = \left| \frac{x - \bar{x}}{\bar{x}} \right| \quad (5.1)$$

dove \bar{x} è il valore sperimentale dello stato termodinamico considerato.

Di seguito si riportano le tabelle relative agli errori a regime dei due modelli per i casi di trim provati nel report [5].

<i>Grandezza</i>	<i>Valore sperimentale</i>	<i>err% γ</i>	<i>err% ISO</i>
T_1	286 [K]	0.00	0.00
T_{21}	570 [K]	3.91	9.48
T_4	876 [K]	2.28	13.77
g_{21}	2.34 [kg/s]	4.33	23.17
p_1	97702 [Pa]	0.00	0.00
p_{21}	779824 [Pa]	3.78	30.14
p_5	99494 [Pa]	4.15	1.80
n_1	37861 [RPM]	2.58	13.54

Tabella 5.1: Errori percentuali dei due modelli rispetto al caso 2 di trim sperimentale.

Si noti come gli errori relativi agli stati termodinamici T_1 e p_1 siano sempre nulli, in quanto in condizioni di velocità di volo nulla, coincidono con le condizioni dell'aria esterna e quindi con le condizioni al contorno che si sono imposte per il confronto con i dati sperimentali.

L'errore più marcato è quello riscontrato nella pressione di scarico dei gas combusti, all'uscita della turbina di potenza. Tale errore, in percentuale anche molto elevato (ad esempio nel caso del test case 6 raggiunge anche il 50%), è paragonabile in modulo con gli errori commessi nel valutare le altre pressioni, ma poichè la pressione p_5 è prossima alla pressione atmosferica, l'errore percentuale risulta molto più marcato rispetto a quello che si ha ad esempio nella valutazione di p_{21} , che è di un ordine di grandezza più elevato. È possibile ridurre l'errore su p_5 modificando il valore dei rendimenti isoentropici delle turbomacchine. Si è scelto comunque di mantenere i rendimenti ai seguenti valori:

- $\eta_C = 0.96$
- $\eta_{HPT} = 0.96$
- $\eta_{PT} = 0.86$

5.2. VALIDAZIONE DINAMICA: RISPOSTA AD UN GRADINO DI PORTATA DI COMBUSTIBILI

- $\eta_{CC} = 0.98$

perchè questi sono i valori riportati più spesso in letteratura per il motore GE t700.

Gli errori commessi nelle simulazioni sono riportate nelle tabelle 5.1, 5.1, 5.1, 5.1, 5.1.

<i>Grandezza</i>	<i>Valore sperimentale</i>	<i>err% γ</i>	<i>err% ISO</i>
T_1	282 [K]	0.00	0.00
T_{21}	600 [K]	1.09	2.42
T_4	903 [K]	1.56	3.59
g_{21}	2.79 [kg/s]	2.94	0.59
p_1	97633 [Pa]	0.00	0.00
p_{21}	958405 [Pa]	0.28	0.79
p_5	99494 [Pa]	1.78	27.69
n_1	39201 [RPM]	0.73	0.15

Tabella 5.2: Errori percentuali dei due modelli rispetto al caso 3 di trim sperimentale.

<i>Grandezza</i>	<i>Valore sperimentale</i>	<i>err% γ</i>	<i>err% ISO</i>
T_1	282 [K]	0.00	0.00
T_{21}	626 [K]	0.70	1.04
T_4	961 [K]	1.11	1.56
g_{21}	3.13 [kg/s]	2.12	4.29
p_1	97150 [Pa]	0.00	0.00
p_{21}	1110785 [Pa]	0.98	5.53
p_5	1100667 [Pa]	3.01	27.07
n_1	40408 [RPM]	0.41	1.17

Tabella 5.3: Errori percentuali dei due modelli rispetto al caso 4 di trim sperimentale.

5.2 Validazione dinamica: risposta ad un gradino di portata di combustibile

La grandezza che rappresenta l'ingresso principale del sistema, attraverso cui si può controllare il regime di rotazione del motore, è l'afflusso di carburante g_b : una validazione dinamica può essere effettuata rilevando la risposta

<i>Grandezza</i>	<i>Valore sperimentale</i>	<i>err% γ</i>	<i>err% ISO</i>
T_1	282 [K]	0.00	0.00
T_{21}	652 [K]	1.46	0.13
T_4	1021 [K]	0.28	0.35
g_{21}	3.47 [kg/s]	3.61	7.46
p_1	96667 [Pa]	0.00	0.00
p_{21}	1274196 [Pa]	4.53	10.24
p_5	100873 [Pa]	1.38	37.43
n_1	41392 [RPM]	1.15	1.83

Tabella 5.4: Errori percentuali dei due modelli rispetto al caso 5 di trim sperimentale.

<i>Grandezza</i>	<i>Valore sperimentale</i>	<i>err% γ</i>	<i>err% ISO</i>
T_1	282 [K]	0.00	0.00
T_{21}	682 [K]	6.38	1.15
T_4	1096 [K]	4.86	1.41
g_{21}	3.86 [kg/s]	13.47	9.84
p_1	95978 [Pa]	0.00	0.00
p_{21}	1461051 [Pa]	19.4	13.43
p_5	101494 [Pa]	21.13	50.52
n_1	42867 [RPM]	4.88	3.42

Tabella 5.5: Errori percentuali dei due modelli rispetto al caso 6 di trim sperimentale.

5.2. VALIDAZIONE DINAMICA: RISPOSTA AD UN GRADINO DI PORTATA DI COMBUSTIBILI

del sistema a una perturbazione di questo ingresso. Nel report NASA [5], la validazione dinamica del modello di motore in anello aperto viene effettuata confrontando i dati ottenuti dalla risposta ad un gradino di portata di combustibile del modello semplificato con quella di un simulatore ad alta fedeltà proprietario del produttore del propulsore. Essendo i dati del report NASA disponibili solo in formato grafico, si è proceduto dapprima digitalizzando i grafici e poi elaborandoli per ottenere dei vettori che sono poi stati confrontati con i risultati ottenuti dalla simulazione.

La simulazione è stata effettuata a parità di condizioni iniziali, di tempi e di passo di integrazione. Si riportano di seguito le condizioni di simulazione:

- durata temporale $t = 5s$;
- passo di integrazione $\Delta t = 14 \text{ ms}$;
- afflusso di carburante iniziale $g_{b_{iniziale}} = 400 \text{ lb /h} = 0.050\,399\,16 \text{ kg s}^{-1}$;
- afflusso di carburante finale $g_{b_{step}} = 775 \text{ lb /h} = 0.097\,648\,37 \text{ kg s}^{-1}$;
- momento di applicazione del gradino di afflusso di carburante $t = 0.5s$;

Le condizioni iniziali da cui si è fatta partire la simulazione sono state:

<i>Temperatura</i>	<i>Valore [k]</i>
T_1	288.15
T_{21}	633
T_3	1171
T_2	633
T_4	900
T_5	730

Tabella 5.6: Condizioni iniziali di temperatura nelle varie sezioni del motore prima dell'ingresso a scalino di portata di combustibile.

L'andamento dell'ingresso a gradino è visibile in figura 5.1, mentre il confronto della risposta è riportato nelle figure 5.2, 5.4, 5.5, 5.3.

Si noti la buona riproduzione del comportamento della temperatura di uscita dalla camera di combustione (figura 5.4), che è uno dei tre stati del sistema. La buona riproduzione della dinamica del componente indica una corretta modellazione, che riesce a catturare, nonostante la relativa semplicità, le dinamiche tipiche con un errore modesto.

Nelle altre figure è possibile notare come lo scostamento dei valori sia più marcato, sebbene le forme delle risposte siano riprodotte abbastanza

<i>Portata</i>	<i>Valore</i> [$\frac{kg}{s}$]
g_{21}	2.882
g_2	2.882
g_3	2.882
g_4	2.932
g_5	2.932
g_{cool}	0
g_b	0.05039916

Tabella 5.7: Condizioni iniziali delle portate prima dell'ingresso a scalino di portata di combustibile.

<i>Pressione</i>	<i>Valore</i> [Pa]
p_1	101325
p_{21}	1036000
p_2	1036000
p_3	1021000
p_4	331800
p_5	126900

Tabella 5.8: Condizioni iniziali delle pressioni prima dell'ingresso a scalino di portata di combustibile.

5.2. VALIDAZIONE DINAMICA: RISPOSTA AD UN GRADINO DI PORTATA DI COMBUSTIBILI

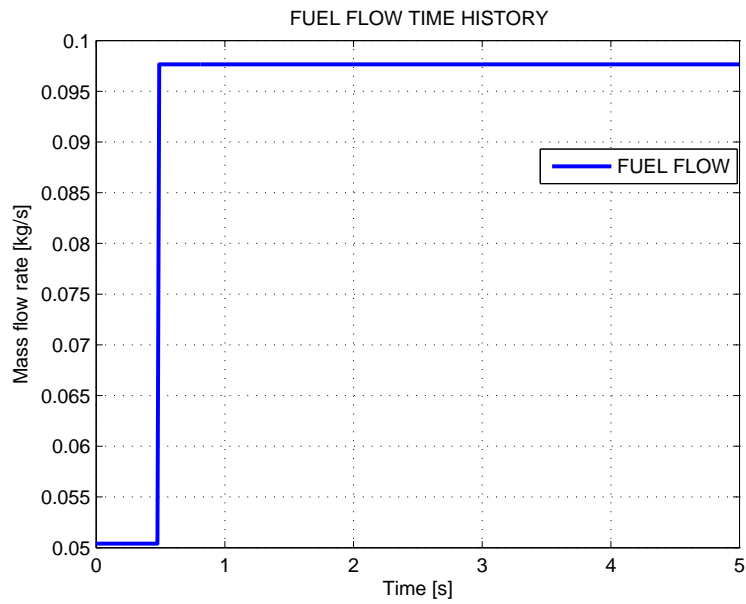


Figura 5.1: Andamento dell'ingresso di portata di combustibile a gradino.

fedelmente. Per ottenere delle risposte ancora più vicine ai dati sperimentali sarebbe possibile ottimizzare i rendimenti delle turbomacchine, ma in questa fase del lavoro si è preferito mantenere i valori costanti e pari a quelli riportati in letteratura per il motore GE t700.

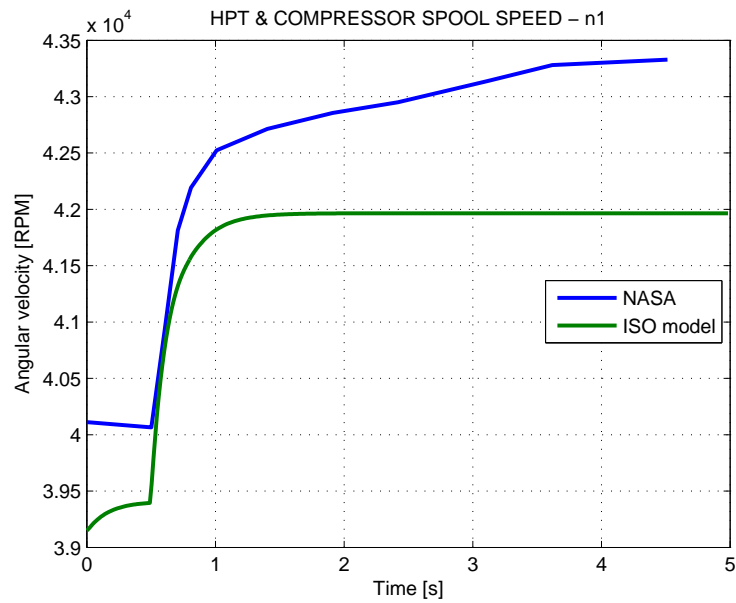


Figura 5.2: Risposta ad un ingresso a gradino di portata di combustibile: confronto del regime di rotazione n_1 con i dati del report Ballin - 1988.

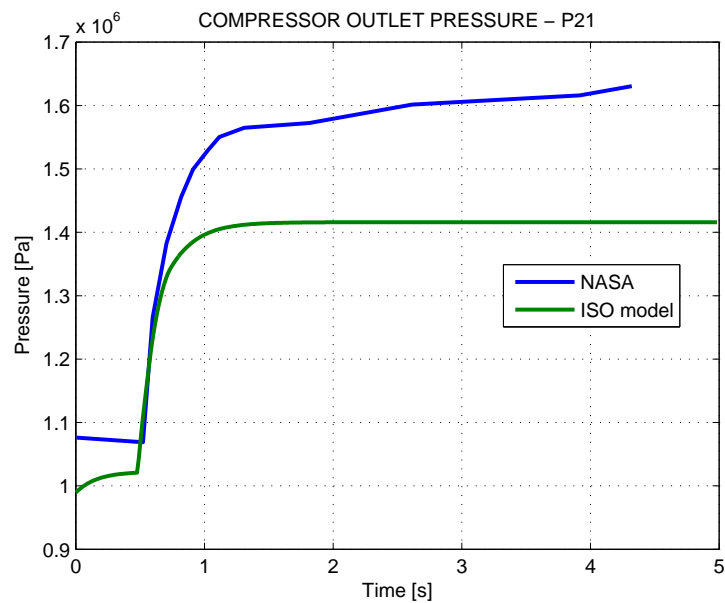


Figura 5.3: Risposta ad un ingresso a gradino di portata di combustibile: confronto della pressione all'uscita dal compressore p_{21} con i dati del report Ballin - 1988.

5.2. VALIDAZIONE DINAMICA: RISPOSTA AD UN GRADINO DI PORTATA DI COMBUSTIBILE

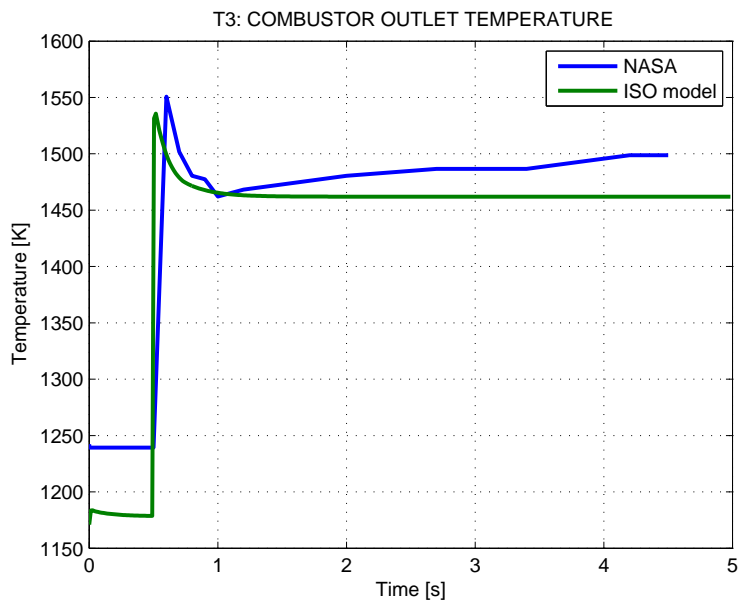


Figura 5.4: Risposta ad un ingresso a gradino di portata di combustibile: confronto della temperatura T_3 in uscita dalla camera di combustione con i dati del report Ballin - 1988.

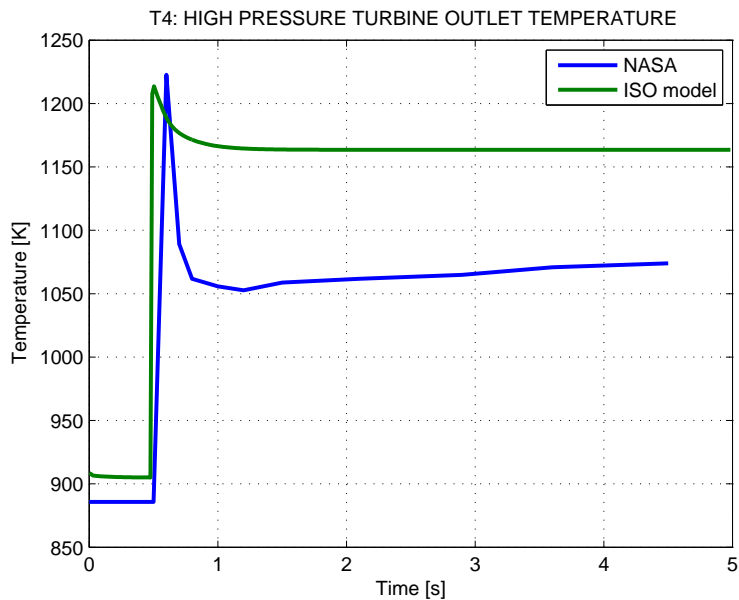


Figura 5.5: Risposta ad un ingresso a gradino di portata di combustibile: confronto della temperatura T_4 in uscita dalla turbina di alta pressione con i dati del report Ballin - 1988.

Capitolo 6

Controllore automatico del propulsore

6.1 Architettura e logica di controllo

Il motore a turbina per elicotteri è generalmente regolato da un controllore automatico, che ne garantisca la funzionalità entro dei limiti di sicurezza. I sistemi di controllo moderni sono di tipo digitale, senza possibilità di passare in modalità manuale: essi prendono il nome di **FADEC** (Full Authority Digital Engine Control) proprio perchè il controllo del motore è totalmente automatico. Il FADEC a sua volta si compone di due componenti principali: ECS e FCU.

Il sistema di controllo del motore ECS (Engine Control System) ha il compito di regolare la potenza del motore in funzione delle necessità operative dell'elicottero. Al variare delle condizioni operative della macchina esso deve:

- regolare la potenza erogata;
- gestire la variazione di flusso di combustibile;
- limitare i regimi di rotazione;
- consentire avviamento e spegnimento;

L' FCU (Fuel Control Unit) gestisce l'afflusso di carburante al motore e su comando dell'ECS trasforma il segnale elettrico del controllore in flusso di carburante attraverso una pompa elettrica.

La figura 6.1 mostra l'architettura del sistema di controllo FADEC nella sua struttura più generale. Il sistema elabora diversi tipi di informazioni:

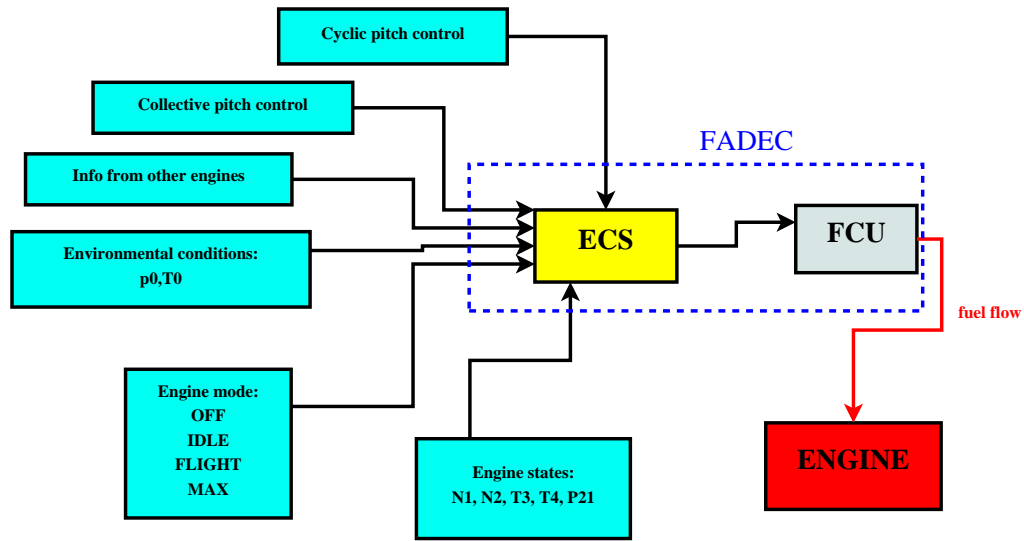


Figura 6.1: Architettura di un sistema di controllo di un motore a turbina per elicotteri.

- grandezze misurate da sensori posti sul motore: velocità di rotazione n_1 e n_2 , pressione p_{21} a valle del compressore, temperatura T_3 a monte della turbina HPT e temperatura T_4 tra le due turbine;
- comandi imposti dal pilota: passo ciclico e passo collettivo;
- condizioni ambientali esterne: p_0, T_0 .

Per ogni situazione operativa il FADEC calcola la potenza richiesta dal carico, la velocità di rotazione necessaria all'albero della turbina generatrice di gas ($n_{1desiderata}$) per generare tale potenza al secondo albero e il flusso di combustibile da inviare in camera di combustione per variare della quantità Δn_1 la velocità di rotazione del primo albero.

Il controllore impiegato nella maggior parte dei FADEC è di tipo PID, ovvero con azione proporzionale, integrativa e derivativa.

Il controllore acquisisce in ingresso un valore dal processo, e lo confronta con un valore di riferimento. La differenza, il cosiddetto segnale di errore $e(t)$, viene quindi usata per determinare il valore della variabile di uscita del controllore, che è la variabile manipolabile del processo. Il PID regola l'uscita in base a:

- il valore del segnale di errore (azione proporzionale);
- i valori passati del segnale di errore (azione integrale);

TIPO	K_P	K_I	K_D
P	$0.5K_U$		
PI	$0.45K_U$	$0.54\frac{K_U}{T_U}$	
PID	$0.6K_U$	$0.9\frac{K_U}{T_U}$	$0.075K_U T_U$

Tabella 6.1: Taratura dei guadagni con il metodo di Ziegler e Nichols.

- quanto velocemente il segnale di errore varia (azione derivativa).

Chiamando u il generico vettore di controllo, esso viene fornito in uscita dal regolatore PID come:

$$u(t) = \frac{k_i}{T_i} \int u(t)dt + k_P u(t) + k_d T_d \frac{du(t)}{dt} \quad (6.1)$$

Oppure, chiamando $K_I = \frac{k_i}{T_i}$, $K_D = k_d T_d$, nella più comune forma:

$$u(t) = K_I \int u(t)dt + K_P u(t) + K_D \frac{du(t)}{dt} \quad (6.2)$$

dove le costanti K_I , K_D , K_P sono i guadagni dell'azione integrale, derivativa e proporzionale; l'equazione 6.2 può contenere tutti i tre termini a destra o solo alcuni di essi a seconda del tipo di controllore che si vuole implementare (si veda a riguardo il riferimento [15]).

La taratura dei guadagni può avvenire attraverso il secondo metodo di Ziegler e Nichols (spiegata nei riferimenti [15, 16]):

1. Si utilizza in primo luogo un controllo solo proporzionale, ovvero con $K_P \neq 0$, $K_I = K_D = 0$.
2. Si aumenta il guadagno K_P finché nel sistema non si instaurano *oscillazioni sostenute*, che cioè non spariscono dopo un transitorio: questa è una misura dell'effetto dei ritardi e della dinamica del processo.
3. Il valore minimo di K_P che provoca le oscillazioni sostenute del sistema viene denominato *guadagno critico* K_U .
4. Il periodo delle oscillazioni sostenute del sistema è il *periodo critico* T_U .
5. Noti il periodo critico e il guadagno critico è possibile determinare il valore dei guadagni attraverso la tabella 6.1.

Per avere un'idea dell'ordine di grandezza del guadagno dell'azione proporzionale si è messo in relazione il massimo valore dell'errore, che si avrebbe

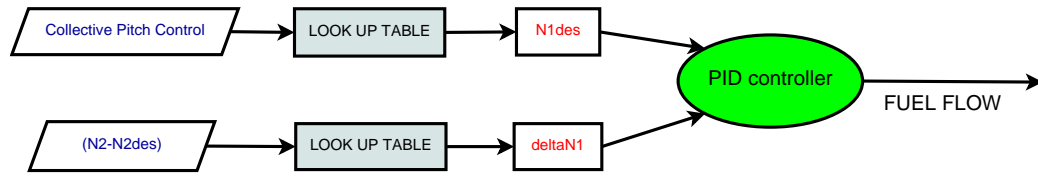


Figura 6.2: Logica di controllo del regolatore PID implementato all'interno del FADEC.

quando il secondo albero fosse fermo mentre la velocità desiderata fosse quella nominale, e il massimo valore della portata di combustibile. L'errore massimo sul regime n_2 risulta:

$$err_{MAX} = n_{2des} - n_2 = n_{2nom} - 0 = n_{2nom} \quad (6.3)$$

La determinazione del guadagno K_P può avvenire come:

$$K_{P_{iniziale}} = \frac{g_{bMAX}}{n_{2MAX}} \quad (6.4)$$

Nel caso del motore per aeromobili, la variabile manipolabile è la portata di combustibile diretta in camera di combustione (g_b o portata di combustibile), mentre l'errore viene calcolato sul regime di rotazione dell'albero di potenza, la 6.2 va quindi valutata assegnando:

$$u(t) = g_b(t) \quad (6.5)$$

$$e(t) = n_2(t) - n_{2ref} \quad (6.6)$$

Dove n_{2ref} è il valore del regime di rotazione da mantenere per la determinata condizione di volo. Ovviamente per raggiungere il valore di n_{2ref} agendo sul flusso di combustibile il motore reagirà accelerando n_1 e poi trasferendo il moto al secondo albero, l'azione è quindi indiretta. É possibile implementare delle tabelle (look up table) che dato il valore di Δn_2 e di comando del passo ciclico, forniscano il valore di n_1 da far raggiungere al motore.

La logica di controllo del PID implementato all'interno di un FADEC in realtà tiene anche conto dei comandi di passo ciclico e passo collettivo che il pilota applica al velivolo, aggiornando il valore di n_{1ref} da inseguire in base a questi dati. La logica di controllo qui descritta, implementata in un generico FADEC ha lo schema riportato in figura 6.2.

6.1.1 Limiti di impiego del motore

Come detto il FADEC deve regolare la portata di combustibile g_b così da mantenere il funzionamento del motore all'interno dei suoi limiti operativi.

Le prestazioni che vengono richieste al motore dal pilota o dai sistemi di gestione del volo devono essere compatibili con numerosi vincoli, sia del velivolo (e delle parti che lo compongono: sistema di trasmissione, rotore, ecc..) che del motore stesso. Il sistema di controllo deve assicurare che l'utilizzo del motore avvenga in ogni condizione nel rispetto di questi limiti, il FADEC deve infatti:

- garantire che il regime di rotazione n_2 si mantenga in prossimità del valore nominale in tutte le condizioni operative;
- limitare il regime di rotazione n_1 sia inferiormente (per non avere troppo ritardo nella successiva accelerazione) che superiormente (*overspeed*: regimi troppo elevati potrebbero portare i componenti soggetti alla forza centrifuga a superare i propri limiti meccanici);
- garantire la composizione ottimale della miscela aria/combustibile sia in accelerazione (rischio di *surge*: fiammate che sottopongono i componenti meccanici a temperature troppo elevate), che in decelerazione (rischio di *flame out*: spegnimento della fiamma);
- limitare la potenza P_{PT} erogata dal propulsore, per non sottoporre ad eccessive sollecitazioni le componenti a valle del motore;
- limitare la coppia del motore per non portare a rottura la trasmissione;
- limitare la temperatura di ingresso in turbina per non sottoporre le palette della turbina a un eccessivo stress termico.

6.2 Implementazione

Come effettuato per la parte termodinamica, l'implementazione del controllore è avvenuta prima in linguaggio Matlab, e successivamente in linguaggio C, sfruttando la facilità di scrittura ed elaborazione dei dati di Matlab.

Il controllore del propulsore è, dal punto di vista dell'implementazione, assimilabile ad un sistema che, noti i parametri del sistema-motore (i suoi stati termodinamici) e noti gli ingressi (ad esempio un target di potenza da fornire al carico o un comando di passo ciclico del pilota), genera l'ingresso di carburante al motore (si veda a riguardo la figura 6.3). L'unica differenza fra l'implementazione del motore senza controllore e l'implementazione del motore controllato è quindi la presenza di un sistema (implementato attraverso una funzione) che varia il flusso di carburante invece di assegnargli una storia temporale come veniva fatto ad esempio con la funzione *FUELflow.m*

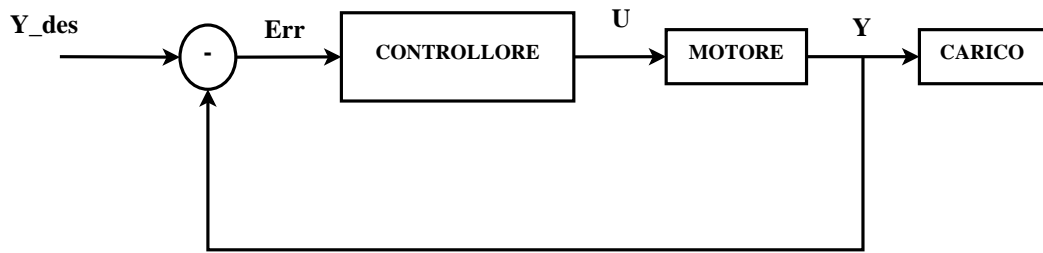


Figura 6.3: Sistema motore-controllore in anello chiuso.

nel modello ISO. Ovviamente la presenza di un sistema dinamico come il controllore, modificherà la risposta del sistema complessivo, composto da motore e controllore.

6.2.1 Implementazione in linguaggio Matlab

Non dovendo il codice scritto in linguaggio Matlab interfacciarsi con alcun simulatore di meccanica del volo, il controllore sviluppato per il modello ISO non ha a disposizione le grandezze tipiche con cui un FADEC regola tipicamente il suo funzionamento: i comandi di passo ciclico e passo collettivo forniti dal pilota. Si è quindi implementato un controllore di tipo PID, dove l'obiettivo che viene richiesto è quello di mantenere il regime di rotazione dell'albero di potenza in prossimità del regime nominale. In una implementazione all'interno di un simulatore sarebbe naturalmente necessario elaborare i comandi di passo ciclico e passo collettivo per poi gestire l'afflusso di carburante in modo che la potenza dei propulsori sia tale da mantenere ad ogni istante il rotore in rotazione al regime nominale.

Il controllore in Matlab è una funzione del tipo

```
[gb]=FADEC(Parametri)
```

che regola il flusso di combustibile da inviare al motore in base al valore che assumono alcune variabili interessanti per il controllo. Queste variabili consentono al controllore di valutare lo stato in cui si trova ad operare il motore, e permettono di decidere la portata di carburante da inviare in camera di combustione. È richiesto quindi all'utente interessato all'implementazione di un motore in anello chiuso, di fornire non soltanto quei parametri utili alla simulazione della parte termodinamica del motore, ma anche alcuni limiti meccanici o termici del sistema da simulare, che saranno necessari a impostare la logica di controllo del FADEC. Ad esempio per l'applicazione in un simulatore di meccanica del volo di un elicottero, bisognerà specificare al controllore:

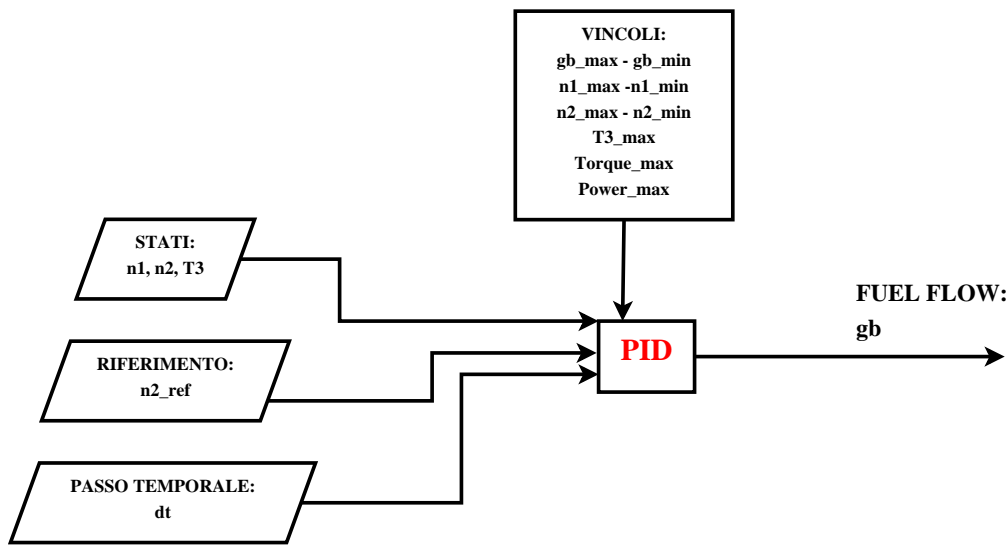


Figura 6.4: Struttura del controllore FADEC nell'implementazione in linguaggio Matlab.

- la massima potenza ($P_{PT_{MAX}}$) (o coppia (C_{MAX})) trasferibile alla trasmissione,
- il minimo ($n_{2_{MIN}}$) e il massimo regime ($n_{2_{MAX}}$) di rotazione sopportato dal carico,
- il minimo ($n_{1_{MIN}}$) e il massimo ($n_{1_{MAX}}$) regime di rotazione ammesso per l'albero della turbina generatrice di gas: per avere un ritardo accettabile nella successiva accelerazione del motore il primo, e per non sforare i limiti strutturali con le forze centrifughe il secondo,
- il minimo ($g_{b_{MIN}}$) e il massimo ($g_{b_{MAX}}$) afflusso di combustibile ammesso in camera di combustione,
- la massima temperatura T_3 accettabile in camera di combustione.

Nell'implementazione la taratura dei guadagni del regolatore PID è stata effettuata attraverso il secondo metodo di Ziegler e Nichols, andando poi a modificare leggermente i valori forniti dalla tabella 6.1 per privilegiare la prontezza della risposta o l'annullamento più rapido possibile dell'errore di n_2 .

6.3 Simulazioni del motore in anello chiuso

Si riportano gli andamenti degli stati termodinamici di alcune simulazioni effettuate con il modello ISO di motore controllato da un PID in ambiente Matlab. Al propulsore è collegato un carico, la cui potenza è descritta dalla 3.57. L'obiettivo del controllore è quello di mantenere il carico in rotazione a una velocità prossima a quella del riferimento $n_{2_{REF}}$.

6.3.1 Simulazione 1 - controllore proporzionale

I parametri caratteristici della simulazione 1 sono:

- tempo di simulazione: 5 secondi;
- passo di integrazione: 0.01 secondi;
- tipo di controllore: Proporzionale;
- guadagno: $K_P = 4.65 \cdot 10^{-3}$;
- target di velocità di rotazione: $n_2 = n_{2_{nominale}} = 20900[RPM]$;

La simulazione è stata effettuata accoppiando il modello di carico che simula in modo semplificato il rotore dell'elicottero Sikorsky UH-60 ad un solo motore. Tale tipo di utilizzo del motore è simile a quello che si avrebbe durante un guasto di uno dei due propulsori, situazione OEI (One Engine Inoperative), in cui la potenza necessaria alla macchina deve essere fornita interamente da una sola delle due unità propulsive. Tale condizione è molto gravosa per il motore e ci si aspetta che il consumo sia superiore a quello che si ha durante il funzionamento normale.

Le condizioni iniziali della simulazione 1 sono riportati nella tabella 6.3.1.

Osservazioni L'aggiunta del controllore e della sua dinamica ha richiesto nell'implementazione in Matlab con la funzione integratrice ode15i di ridurre il passo di integrazione a $\Delta t = 0.01$ s per non avere problemi di convergenza. Tale passo di integrazione è dell'ordine di grandezza di quello richiesto dalla dinamica del rotore in un simulatore che implementi un modello di rotore con grandezze mediate sul giro, ma è inferiore di un ordine di grandezza di quello richiesto dalla sola dinamica termica-meccanica del motore in anello aperto.

Variabile	Valore	Unità di misura
n_1	$44700 \frac{\pi}{30}$	rad/s
n_2	$20000 \frac{\pi}{30}$	rad/s
p_1	95978	Pa
T_1	281.778	K
p_{21}	95978	Pa
T_{21}	281.778	K
p_2	95978	Pa
T_2	281.778	K
p_3	95978	Pa
T_3	281.778	K
\dot{T}_3	0	K/s
p_4	98978	Pa
T_4	281.778	K
p_5	95978	Pa
T_5	281.778	K
P_c	0	W
P_{hpt}	0	W
P_{pt}	0	W
P_{fpt}	0	W
P_{fhpt}	0	W
g_{21}	0	Kg/s
g_2	0	Kg/s
g_3	0	Kg/s
g_4	0	Kg/s
τ_{CC}	0	s
\dot{n}_1	0	$rad\ s^{-2}$
\dot{n}_2	0	$rad\ s^{-2}$
\dot{T}_3	0	K/s

Tabella 6.2: Condizioni iniziali della simulazione 1 di motore controllato in anello chiuso da controllore proporzionale.

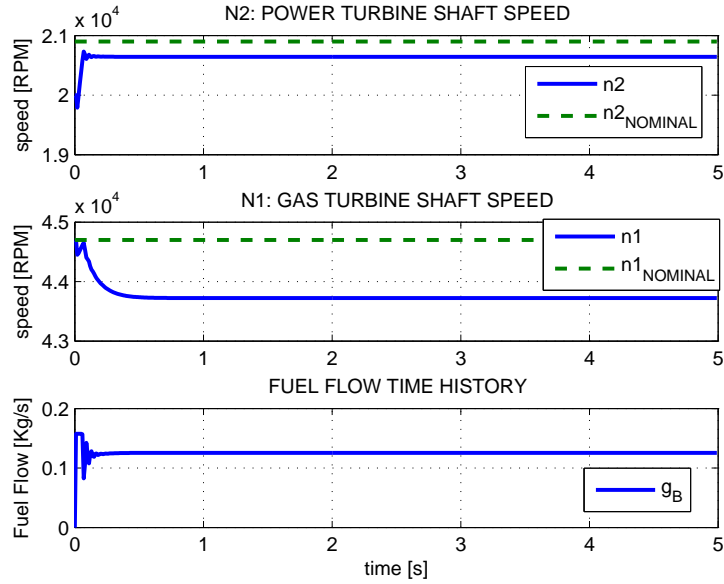


Figura 6.5: Simulazione 1 con controllore P. Velocità di rotazione dei due alberi e variabile di controllo g_b .

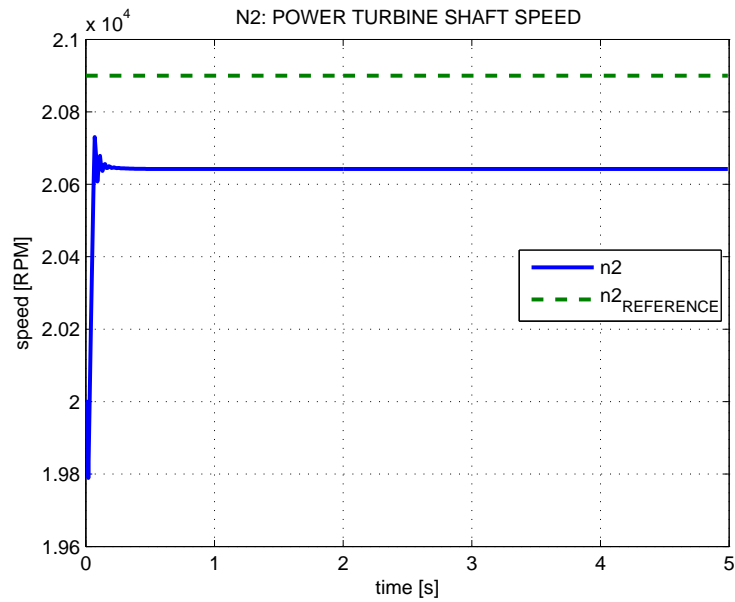


Figura 6.6: Simulazione 1 con controllore Proporzionale. Velocità angolare dell'albero della turbina di potenza. Confronto con il riferimento da inseguire.

Utilizzo di un controllore proporzionale L'utilizzo di un controllore solo proporzionale (di tipo P) con un guadagno adeguato fornisce prestazioni piuttosto soddisfacenti. L'andamento della variabile di controllo nella simulazione 1 mostra una sovraelongazione di lieve entità e si attesta sul valore di regime dopo circa 0.18 s. La variabile da controllare raggiunge dopo circa 0.15 s il valore del riferimento, mentre la velocità dell'albero della turbina generatrice di gas (n_1), si attesta su un valore di poco inferiore al valore nominale. Il non annullarsi dell'errore a regime è coerente con il tipo di controllore scelto: un controllore solo proporzionale non è capace di annullare l'errore a regime, ma è pronto nell'inseguire il riferimento. Il consumo di carburante per mantenere in rotazione il rotore con un solo propulsore è, come ci si aspetta, è superiore a quello che si avrebbe in condizioni nominali. Il surplus di portata di carburante richiesto rispetto alle condizioni nominali è di circa il 33%.

6.3.2 Simulazione 1 - controllore PI

La simulazione 3 riprende esattamente i dati iniziali della simulazione 1, ma con l'impiego di un controllore di tipo PI: proporzionale e integrativo. Tale controllore permette l'annullamento dell'errore a regime. Per la taratura dei guadagni si è partito dai valori ottenuti dalla regola di Ziegler e Nichols, variando poi il guadagno della componente integrativa perchè il valore calcolato con la regola della tabella 6.1 provocava oscillazioni sostenute.

Si noti (figura 6.7) come il controllore con azione proporzionale e integrativa riesca ad annullare l'errore a regime, a differenza del controllore solo proporzionale. L'azione integrativa provoca un ritardo nell'inseguimento del riferimento: la velocità di rotazione n_2 raggiunge il suo valore target dopo circa 0.4 s (figura 6.8).

In figura 6.9 è riportato l'andamento delle due velocità di rotazione e della variabile di controllo g_b , che rappresenta l'uscita del controllore.

6.3.3 Simulazione 1 - controllore PID

La simulazione 1 condotta utilizzando un controllore PID, che impieghi tutti i tre contributi proporzionale, integrativo e derivativo, fornisce come risultati le storie temporali visibili nelle figure 6.10, 6.11 e 6.12.

È possibile notare (figura 6.11) come già dopo 0.25 s la velocità n_2 sia giunta già in prossimità del riferimento.

In figura 6.12 si apprezza come nei primi istanti della simulazione la variabile di controllo cambi repentinamente, a causa dell'azione derivativa. Tali

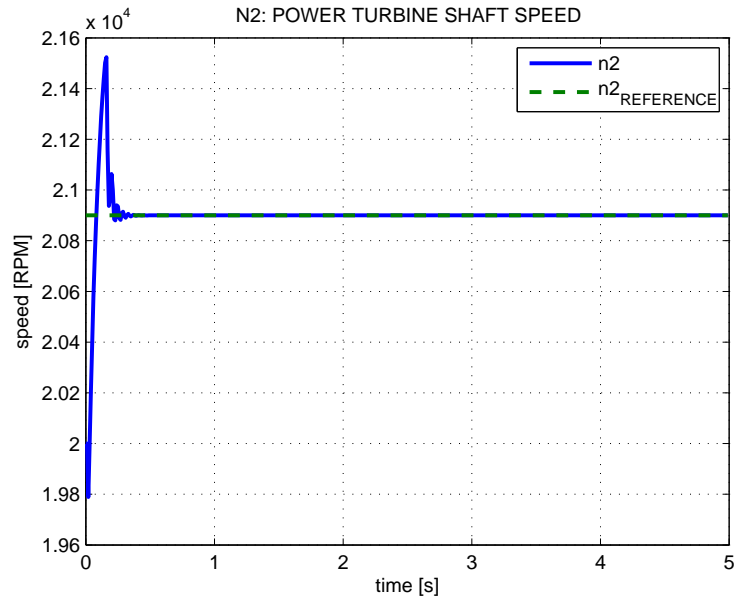


Figura 6.7: Inseguimento del riferimento di velocità di rotazione costante con controllore PI.

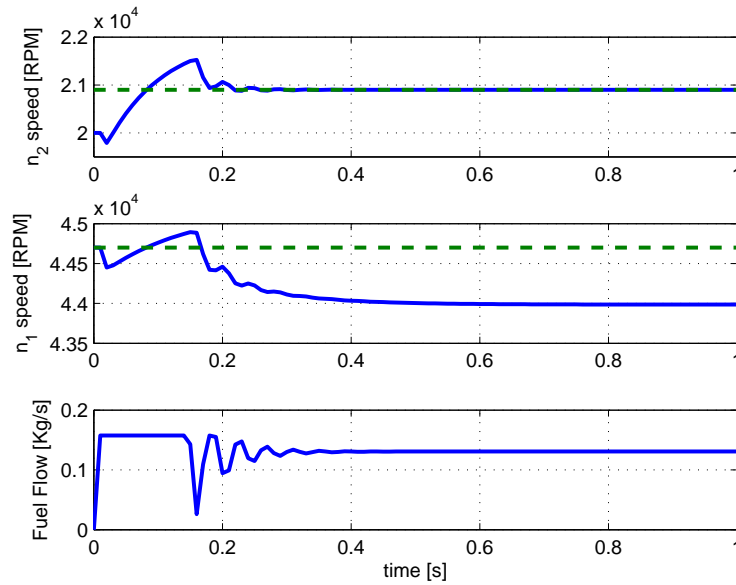


Figura 6.8: Simulazione 1 con controllore PI. Andamento dei due stadi meccanici e della variabile di controllo durante il primo secondo di simulazione.

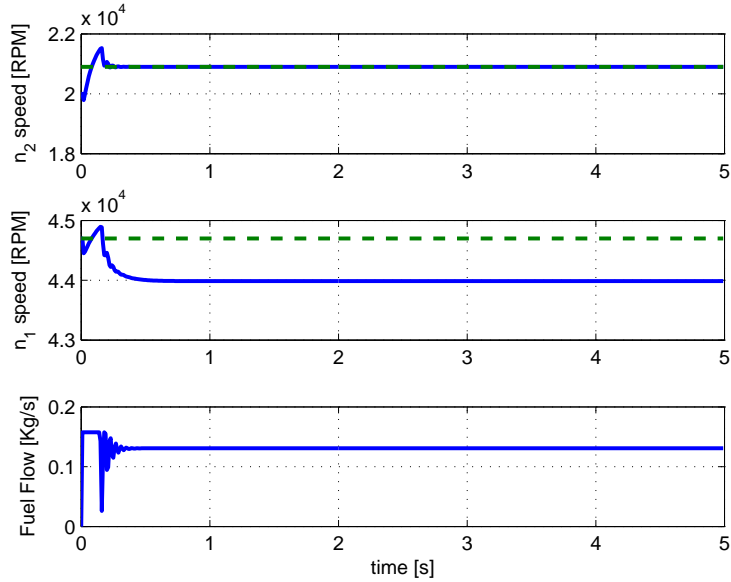


Figura 6.9: Velocità di rotazione n_1, n_2 e portata di combustibile g_b durante l'inseguimento del riferimento costante $n_{2_{ref}}$ con controllore di tipo PI.

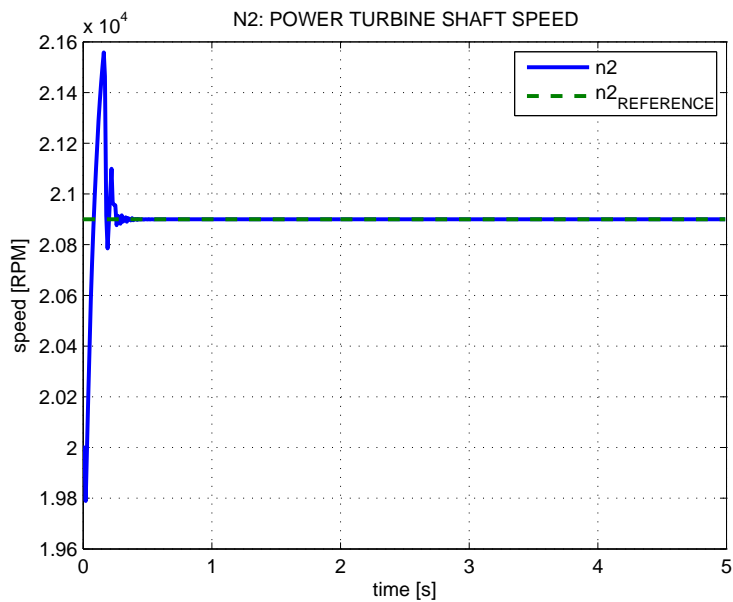


Figura 6.10: Simulazione 1 con controllore PID. inseguimento del riferimento sulla velocità di rotazione n_2 .

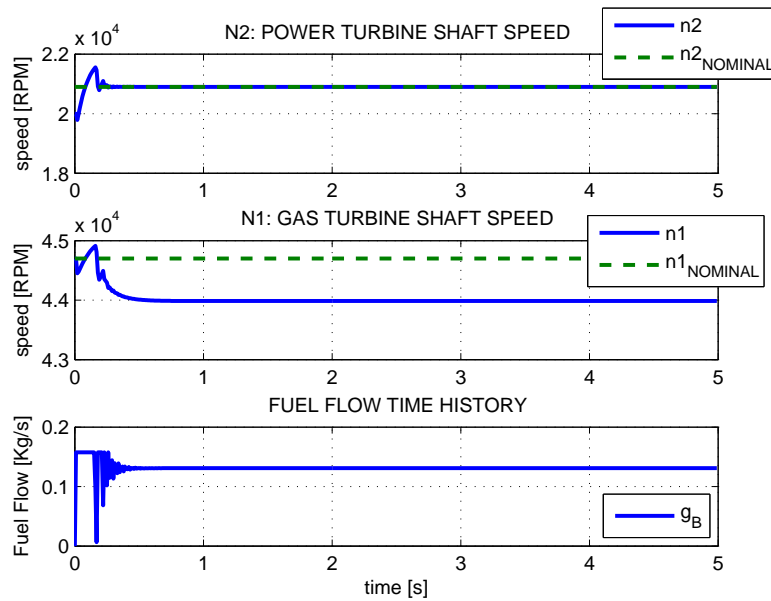


Figura 6.11: Simulazione 1 con controllore PID. Velocità di rotazione e portata di combustibile durante la simulazione.

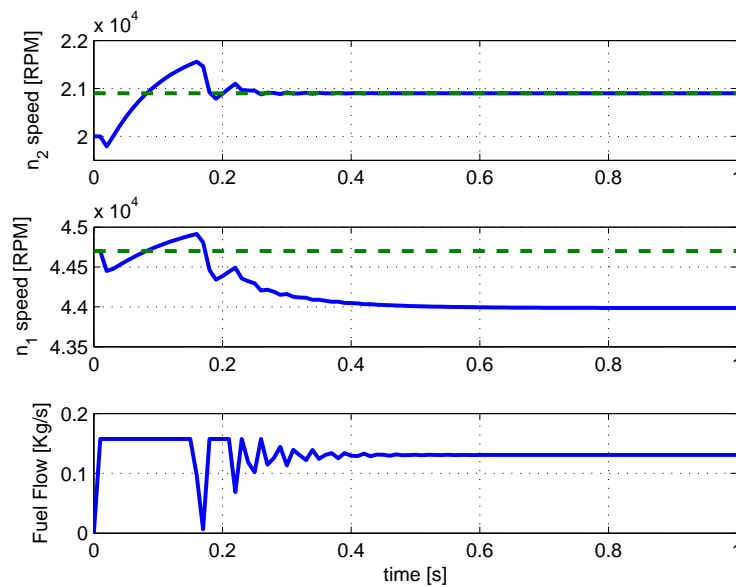


Figura 6.12: Simulazione 1 con controllore PID. Particolare del primo secondo di simulazione.

cambiamenti repentini sono possibili solo in via teorica, in quanto un attuatore reale (in questo caso il motore elettrico che aziona la pompa del carburante), ha una sua banda passante e quindi non riuscirebbe a fornire variazioni di g_b così repentine se queste fossero a una frequenza superiore a quella massima della banda dell'attuatore. Nel nostro modello gli attuatori non sono stati modellati, ma non sarebbe difficile implementare un modello equivalente della dinamica dell'attuatore, così da tenere conto dei ritardi e delle dinamiche associate.

6.3.4 Simulazione 2 - controllore proporzionale

La simulazione 2 prevede un riferimento di velocità di rotazione n_{2ref} che si attesti su tre valori diversi durante la durata del calcolo. Sia t l'istante considerato:

1. per $t \leq \frac{t_{fin}}{3}$ il target è $n_{2ref} = 1.1 n_{2nominal}$
2. per $\frac{t_{fin}}{3} < t \leq \frac{2t_{fin}}{3}$ il target è $n_{2ref} = 0.95 n_{2nominal}$
3. per $\frac{2t_{fin}}{3} < t \leq t_{fin}$ il target è $n_{2ref} = 1.1 n_{2nominal}$

I parametri caratteristici della simulazione 2 sono:

- tempo di simulazione: 5 secondi;
- passo di integrazione: 0.01 secondi;
- guadagno: $K_P = 4.65 \cdot 10^{-3}$;
- target: n_2 variabile durante la simulazione.

Tale simulazione è in effetti una prova della reattività del sistema in quanto il riferimento varia repentinamente e si vuole verificare la capacità di inseguimento del sistema controllato in anello chiuso. Le condizioni iniziali e i parametri sono gli stessi della simulazione 1, riportati nella tabella 6.3.1.

In figura 6.13 è mostrato l'inseguimento del riferimento da parte del sistema. Nonostante la repentina variazione del riferimento (a gradino), il controllore proporzionale è rapido nell'inseguirlo: il transitorio dura circa 0.2s. Il limite del controllore di tipo proporzionale è che l'errore a regime non tende ad annullarsi ma si attesta su un valore costante, in questo caso l'errore esaurito il transitorio è sempre minore di 1000 *RPM* ed è proporzionale al modulo del valore di riferimento. Per avere l'annullamento dell'errore

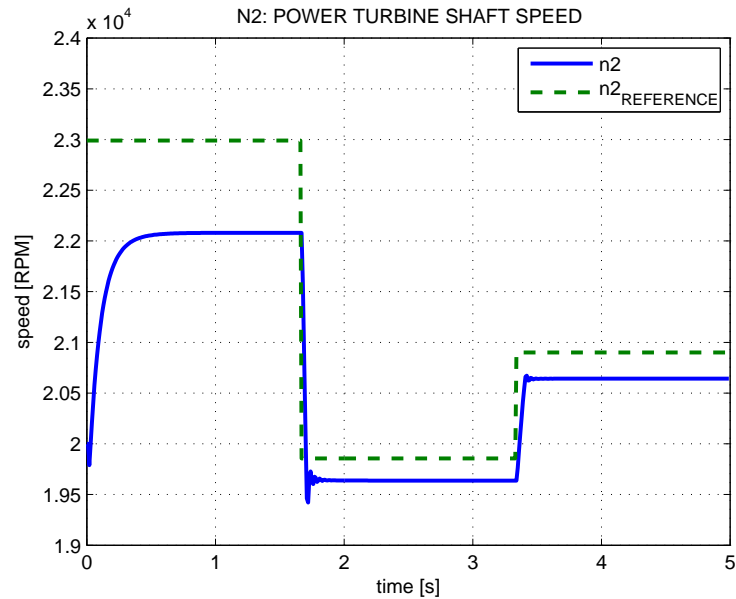


Figura 6.13: Inseguimento del riferimento da parte del sistema motore - controllore in anello chiuso. Simulazione 2. Controllore solo proporzionale.

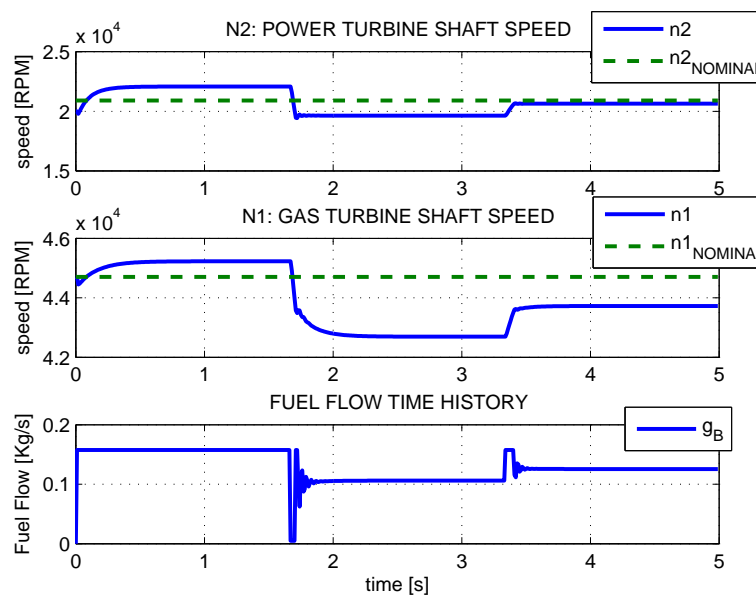


Figura 6.14: Velocità di rotazione dei due alberi e flusso di carburante in camera di combustione durante la simulazione 2 di inseguimento di un riferimento variabile di n_2 con motore controllato da un regolatore proporzionale.

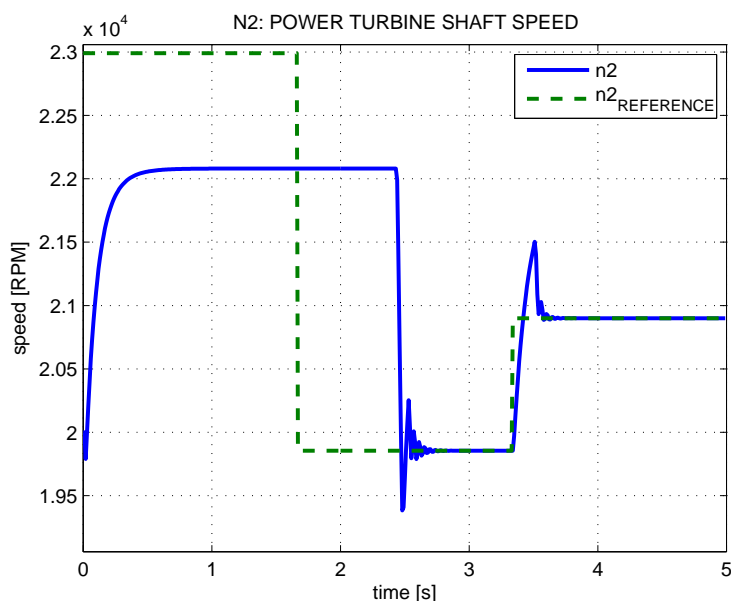


Figura 6.15: Simulazione 2 con controllore PI. Inseguimento del riferimento variabile $n_{2_{REF}}$.

a regime si dovrà impiegare un controllore che implementi anche l'azione integrale (di tipo PI).

In figura 6.14 è possibile vedere l'andamento dei due stati meccanici del sistema (n_1 ed n_2) e la storia della variabile di controllo g_b . Si noti come il controllore proporzionale sconti la sua prontezza con una certa oscillazione della variabile di controllo negli istanti subito successivi alla variazione del riferimento. Tale comportamento è dovuto all'utilizzo di una costante K_P piuttosto elevata calcolata con il secondo metodo di Ziegler e Nichols. Riducendo tale costante è possibile ottenere un controllore meno reattivo che però eviti al sistema tali oscillazioni. A seconda dell'applicazione sarà opportuno privilegiare la prontezza o la stabilità del controllo, ma in linea generale un semplice controllore proporzionale ben tarato sembra poter garantire buone prestazioni.

6.3.5 Simulazione 2 - controllore PI

Similmente a quanto fatto per la simulazione 1, si riportano i risultati della simulazione 2 condotta con un controllore di tipo PI con azione proporzionale e integrativa. Le condizioni della simulazione sono invariate rispetto a quelle utilizzate per la simulazione 2 con controllore solo proporzionale.

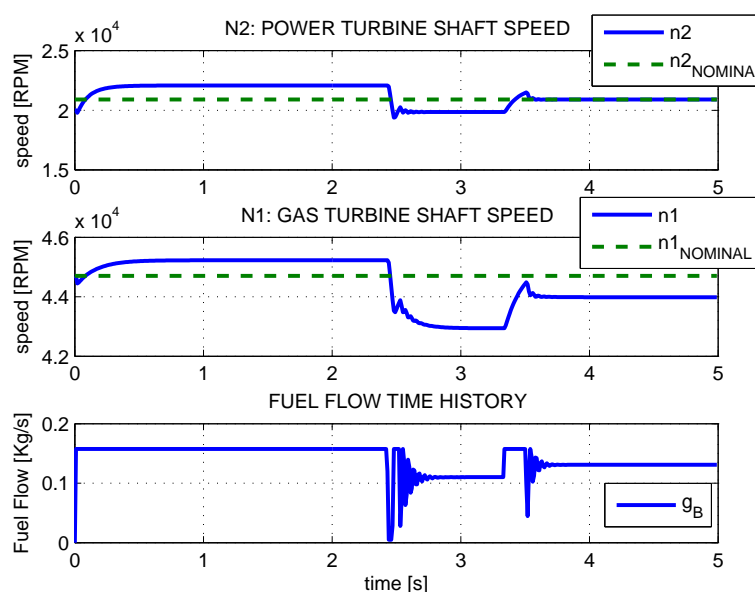


Figura 6.16: Simulazione 2 con controllore PI. Andamento dei due stati meccanici n_1, n_2 e della variabile di controllo g_b .

In figura 6.15 è possibile notare come il controllore con azione proporzionale e integrativa sia meno pronto nell'inseguire il riferimento, ma garantisca l'annullamento dell'errore dopo il transitorio.

In figura 6.16 si vede come le oscillazioni della variabile di controllo g_b provochino un andamento oscillatorio degli stati meccanici del sistema n_1 e n_2 . Tali oscillazioni possono essere alleviate variando il guadagno K_I della componente derivativa, ma a scapito della prontezza del controllore. In questo caso si è preferito mantenere le costanti al valore suggerito dalla tabella 6.1.

Si noti come il riferimento di n_2 più elevato (quello del primo terzo del tempo di simulazione) non sia raggiungibile con un solo motore per limiti di potenza. Esso è in effetti una condizione non fisica, in quanto si chiederebbe al rotore di girare ad un regime troppo elevato. Tale condizione è stata inventata solo per provare la risposta dinamica del sistema motore-controllore.

6.3.6 Simulazione 2 - controllore PID

Utilizzando un controllore PID, quindi con azione proporzionale, integrativa e derivativa, i risultati della simulazione 2 sono quelli mostrati nelle figure

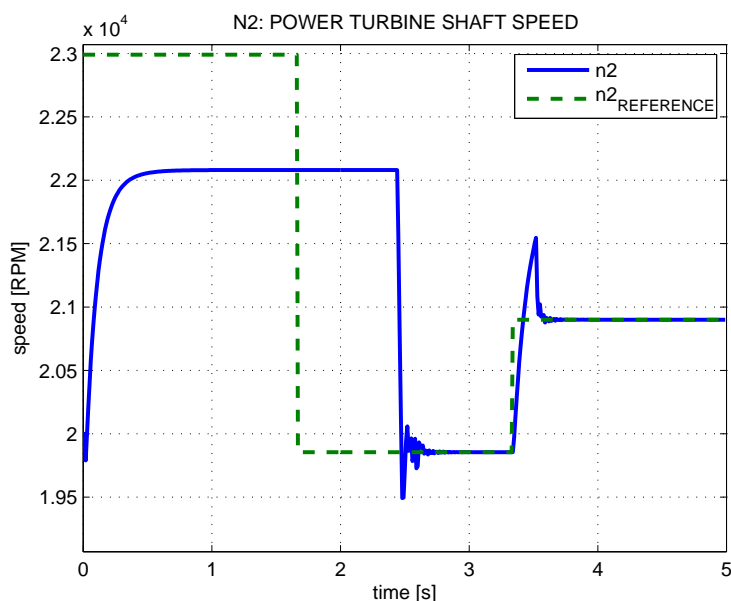


Figura 6.17: Simulazione 2 con controllore PID. Inseguimento del riferimento variabile di velocità di rotazione n_2 .

6.17 e 6.18. I guadagni sono determinati come nell'ultima riga della tabella 6.1.

La figura 6.17 mostra una risposta del sistema controllato dal controllore PID molto simile a quella del sistema controllato dal controllore PI, a meno delle sovraelongazioni che nel caso del PID sono meno marcate e di intensità più breve. Permangono comunque delle oscillazioni al mutare del valore del riferimento n_{2_REF} , e un certo ritardo di annullamento dell'errore.

Tali comportamenti sono sicuramente da imputare ai valori dei guadagni impiegati, che sono sì in linea con il secondo metodo di Ziegler e Nichols, ma potrebbero non essere i più adatti per il problema in esame. In questo lavoro di tesi si è voluto solo dimostrare come l'architettura PID con guadagni determinati dal secondo metodo di Ziegler e Nichols sia flessibile e consenta di ottenere buone prestazioni di controllo, senza entrare troppo nello specifico della taratura dei guadagni per una specifica applicazione, in quanto il modello è nelle sue intenzioni estendibile al maggior numero di applicazioni. Delle opportune variazioni dei guadagni sicuramente possono consentire il raggiungimento di ottime prestazioni con un controllore PID e si rimanda quindi l'utente ai riferimenti [15, 16] per tarare opportunamente questi guadagni nelle condizioni in cui si trovasse ad operare.

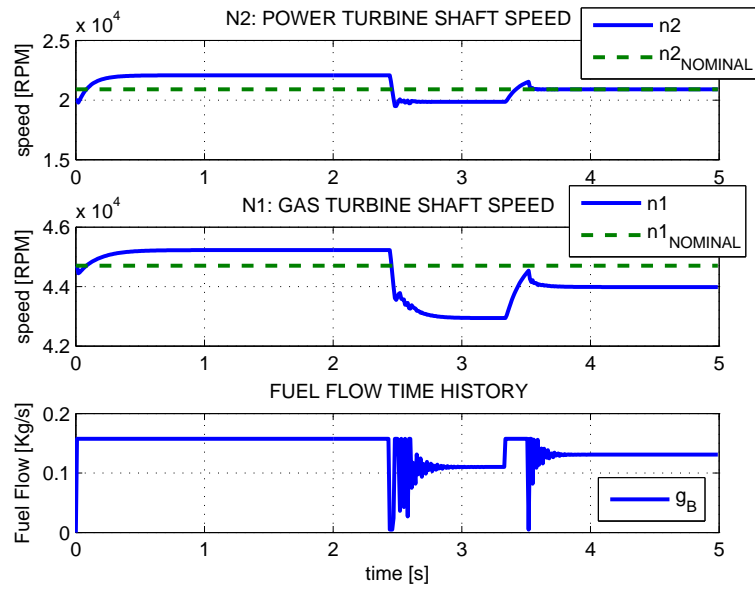


Figura 6.18: Simulazione 2 con controllore PID. Storia temporale dei due stati meccanici n_1 ed n_2 e della variabile di controllo g_b durante la simulazione.

Capitolo 7

Simulazione di una manovra

7.1 Illustrazione della manovra e software impiegati

Al fine di riprodurre l'utilizzo del modello in una condizione operativa in un simulatore, non avendo ancora a disposizione il codice di meccanica del volo in cui includere il modello scritto in C trattato in questo lavoro, si è proceduto sostituendo agli ingressi della simulazione degli andamenti reali provenienti da un caso studiato nell'ufficio *flight mechanics* della ditta *AgustaWestland*.

La simulazione riproduce una steep descent, ovvero una discesa ripida e un atterraggio, effettuato con un modello di elicottero reale implementato nel software di simulazione *Flightlab®* sviluppato da Advanced Rotorcraft Technology. L'elicottero in questione monta il propulsore *Pratt & Whitney PT6C-67C*, i cui parametri caratteristici rintracciabili sul testo [17] sono riportati in tabella 7.1. Tale simulazione è realistica e riproduce con grande fedeltà i parametri registrati durante un'analogia prova di volo effettuata dal vero. Si è scelto di confrontarsi con un software di simulazione e non con dei dati sperimentali provenienti da prove di volo per la maggiore disponibilità di parametri prodotti dal simulatore rispetto a quelli misurati durante la prova di volo. In particolare la portata di combustibile g_b , non viene misurata durante una tipica prova di volo, e poiché essa rappresenta l'ingresso (o la variabile di controllo) del sistema, era fondamentale avere a disposizione la sua storia temporale. Per questo motivo si è preferito confrontare i parametri calcolati dal *modello ISO* con quelli derivanti dalla simulazione effettuata con il software FlightLab.

Il modello ISO del motore è stato quindi personalizzato andando a modificare il file *enginePARAMETERS.m*, e specificando i parametri caratteristici disponibili riportati in tabella 7.1. Per alcuni parametri non disponibili si

Configurazione	4A+C
Portata d'aria [kg/s]	4.5
Albero	Frontale
Lunghezza [mm]	1506
Larghezza [mm]	571.5
Potenza al decollo [kW]	1142
$n_{1nominal}$ [RPM]	38200
$n_{2nominal}$ [RPM]	21000

Tabella 7.1: Dati del motore P&W PT6C-67C.

è in prima battuta fatto ricorso ai dati del già citato motore *GE t700*, in quanto è un propulsore della stessa categoria con una configurazione poco dissimile dal propulsore *P&W PT6c-67C*, ma è molto più documentato di quest'ultimo. Successivamente si è cercato di identificare tali parametri dai dati disponibili e di tarare meglio il modello.

Il software *FlightLab* calcola nel corso della simulazione alcune delle variabili di interesse del motore. Poiché la simulazione in esame viene condotta con entrambi i motori attivi, il software in questo caso ha registrato le storie temporali dei parametri di entrambi i propulsori. Si è poi scelto di considerare i parametri del solo motore 1 e di confrontarli con quelli generati nelle medesime condizioni dal nostro modello.

La simulazione è stata effettuata impiegando lo stesso passo temporale e la stessa durata di quelli della simulazione con software Flightlab. Essi sono:

- Durata simulazione: $t_{fin} = 429 \text{ s}$
- Passo temporale: $dt = 0.0081 \text{ s}$

In tal modo è stato possibile fornire ad ogni istante al modello ISO gli stessi ingressi forniti al modello Flightlab, così da confrontare istante per istante le grandezze calcolate dai due modelli di motore.

Le variabili su cui è stato possibile effettuare un confronto, perchè calcolate dal software Flightlab, sono:

- velocità di rotazione della turbina generatrice di gas: n_1 ;
- velocità di rotazione della turbina di potenza: n_2 ;
- temperatura dei gas all'uscita dalla turbina generatrice di gas: T_4 ;
- coppia fornita dal motore all'albero della turbina di potenza;
- pressione dei gas in camera di combustione: p_3 ;

7.2. CONFRONTO UTILIZZANDO ALCUNI PARAMETRI DEL MOTORE GE T700109

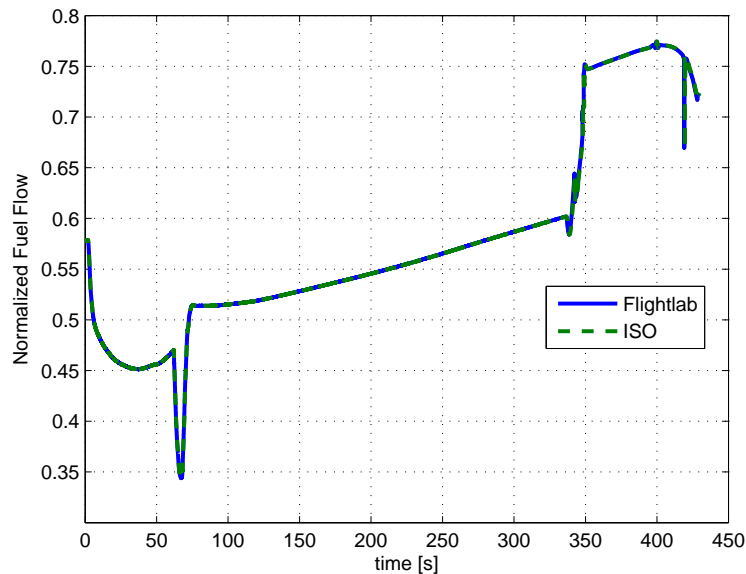


Figura 7.1: Storia temporale della portata di combustibile g_b durante la simulazione effettuata in Flightlab e poi riprodotta con il modello ISO.

La time history dell'ingresso, ovvero la storia temporale dell'afflusso di combustibile è riportata in figura 7.1, confrontando il vettore proveniente da Flightlab e quello riprodotto nel modello ISO. Essi coincidono. Si noti come questa manovra preveda delle variazioni repentine e di grande entità dell'afflusso di combustibile, risultando un buon banco di prova per valutare la risposta dinamica del nostro modello.

7.2 Confronto utilizzando alcuni parametri del motore GE t700

Velocità n_1 Osservando la figura 7.2 si noti come la dinamica dell'albero della turbina generatrice di gas sia ben colta dal modello ISO. Gli errori rispetto alla simulazione effettuata in Flightlab sono di piccola entità. Tale comportamento evidenzia un'ottima approssimazione della dinamica dell'albero della turbina generatrice di gas del motore Pratt & Whitney con i parametri derivati dal motore GE t700. Si noti anche come il motore si trovi ad operare in tutto il tempo della simulazione al di sotto del regime nominale, in quanto esso deve fornire circa la metà della potenza richiesta dal carico, essendo affiancato dalla seconda unità propulsiva.

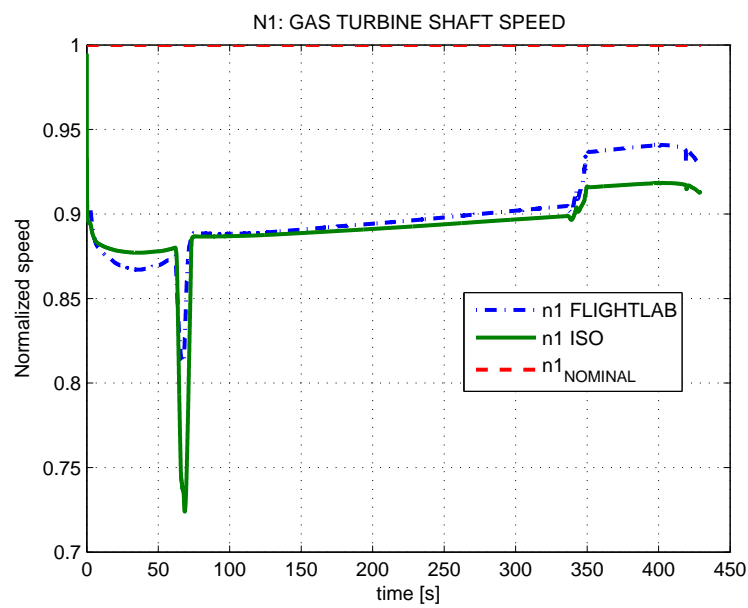


Figura 7.2: Confronto fra la velocità di rotazione n_1 calcolata dal modello ISO e quella ottenuta dalla simulazione in Flightlab. Si riporta anche il valore nominale per il motore considerato. Parametri incogniti derivati dal motore GE t700.

7.2. CONFRONTO UTILIZZANDO ALCUNI PARAMETRI DEL MOTORE GE T700111

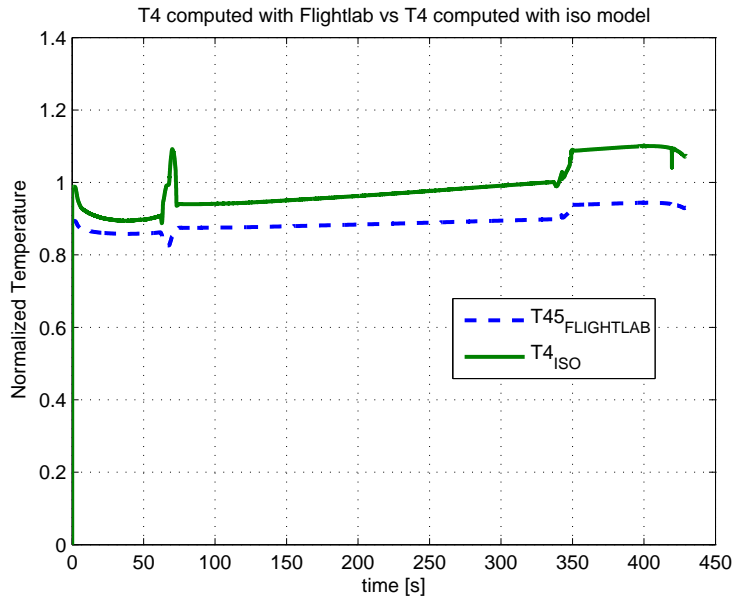


Figura 7.3: Confronto della temperatura T_4 dei gas fra le due turbine calcolata con il modello ISO e il software Flightlab. Parametri incogniti derivati dal motore GE t700.

Temperatura T_4 In figura 7.3 è riportato il confronto relativo alla temperatura dei gas fra le due turbine, ottenuto con i parametri di primo tentativo per il motore Pratt & Whitney, derivati dal motore GE t700. L'errore sulla temperatura fra le due turbine è modesto, anche se il modello ISO sovrastima la temperatura in tutta la durata della simulazione

Potenza erogata Si noti come la potenza espressa dal modello, impostando i parametri incogniti uguali a quelli del motore GE t700 sia molto superiore a quella calcolata dal software Flightlab (figura 7.4), nonostante la forma del suo andamento sia analoga. Questo è segno di una buona modellazione, ma di una cattiva scelta dei parametri di taratura.

È necessaria quindi una messa a punto di questi parametri se si vogliono ridurre gli errori.

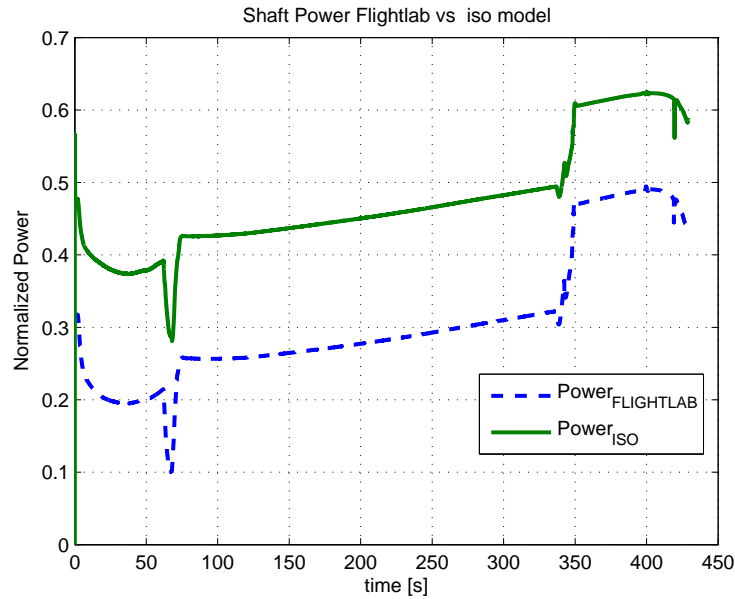


Figura 7.4: Potenza espressa dal modello ISO confrontata con quella espressa dal modello FlightLab. Per i parametri incogniti si è fatto ricorso a quelli del motore GE t700.

7.3 Confronto utilizzando parametri identificati

Per garantire una riproduzione più accurata del motore Pratt & Whitney PT6C-67C si è reso necessario tarare i parametri di personalizzazione del modello per tale motore. Non riuscendo a rintracciare in rete e in letteratura un maggior numero di parametri di quelli riportati in tabella 7.1, si è resa necessaria un'identificazione a partire dai dati disponibili. Tale problema è comune alla gran parte dei motori per elicotteri, che raramente sono bene documentati e i cui parametri caratteristici sono spesso resi disponibili dal produttore solo ai clienti, con accordi di non divulgazione.

Per quanto riguarda la scelta dei momenti di inerzia dei due alberi di rotazione J_{HPT} e J_{PT} si è deciso di scalare i rispettivi valori in funzione del diametro del motore. Essendo il momento di inerzia dipendente dal quadrato della distanza dall'asse di rotazione:

$$J = \int \rho d^2(x, y, z) dV$$

si è pensato di scalarlo in dipendenza della dimensione radiale del motore elevata alla seconda potenza:

$$J_{HPT_{pt6c}} = J_{HPT_{t700}} \frac{R_{pt6c}^2}{R_{t700}^2} \quad (7.1)$$

$$J_{PT_{pt6c}} = J_{PT_{t700}} \frac{R_{pt6c}^2}{R_{t700}^2} \quad (7.2)$$

in cui R_{pt6c} , R_{t700} equivalgono a metà del diametro nominale del motore, grandezza facilmente reperibile in letteratura.

Analogamente, non conoscendo il volume della camera di combustione, ipotizzando che esso sia direttamente proporzionale al volume del motore, si è proceduto scalando il dato proveniente dal solito propulsore GE t700 con il rapporto fra i volumi dei propulsori:

$$V_{CC_{pt6c}} = V_{CC_{t700}} \frac{V_{pt6c}}{V_{t700}} \quad (7.3)$$

In cui V_{CC} è il volume della camera di combustione e V_{pt6c} , V_{t700} sono i volumi dei due motori, calcolati dai dati disponibili in letteratura come:

$$V_{motore} = L_{motore} D_{motore}^2 \frac{\pi}{4}$$

È stato impossibile rintracciare in letterature il rapporto di compressione del compressore e i rapporti di espansione delle turbine. In mancanza di dati si sono ipotizzati dei valori simili a quelli del motore GE t700.

Temperatura T_4 In figura 7.5 è riportato il confronto fra la simulazione condotta in FlightLab e quella condotta con il modello ISO utilizzando i parametri modificati. È possibile notare come l'inseguimento sia buono: la temperatura calcolata dal modello ISO si trova nell'intorno dei valori restituiti dal software Flightlab durante tutta la durata della simulazione.

Velocità n_1 In figura 7.6 è riportato il confronto sulla velocità di rotazione dell'albero della turbina generatrice di gas. Si può notare come il modello ISO con i parametri modificati riproduca in modo peggiore la dinamica di n_1 rispetto al modello ISO con i parametri del motore GE t700. Questo comportamento è da imputarsi in larga misura alla scelta dei parametri effettuati per il modello, che evidentemente non è in linea con i valori reali del motore, che risultano impossibili da reperire in letteratura. Per avere un match migliore in questa dinamica è quindi necessaria un'ulteriore messa a punto dei parametri. Sebbene gli errori siano maggiori, la forma della risposta è riprodotta comunque fedelmente, come nella simulazione effettuata con i parametri derivati dal motore GE t700.

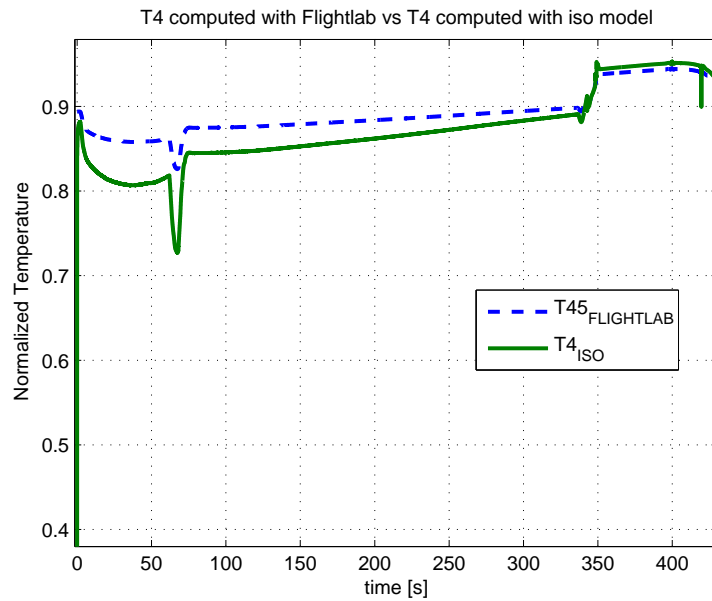


Figura 7.5: Confronto della temperatura T_4 . Modello ISO con parametri modificati e simulazione FlightLab.

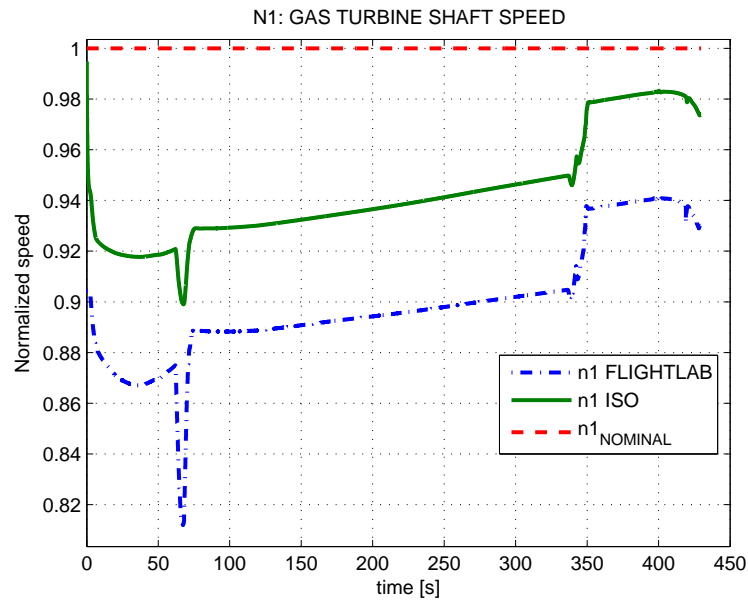


Figura 7.6: Confronto sulla velocità n_1 . Simulazione effettuata con software FlightLab e con modello ISO con parametri modificati.

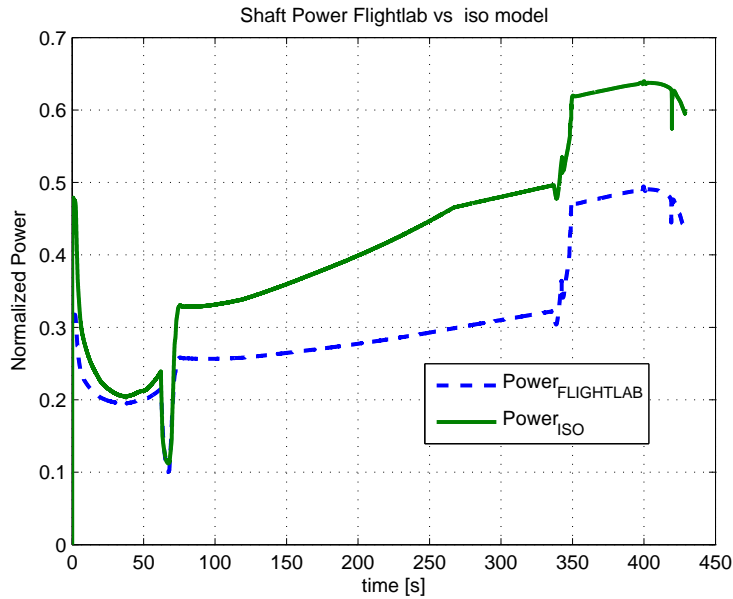


Figura 7.7: Confronto sulla potenza erogata all'interfaccia con il carico. Simulazione condotta con software FlightLab e con modello ISO con parametri modificati.

Potenza erogata In figura 7.7 è riportato l'andamento della potenza erogata al carico. Si noti come con il tuning dei parametri si riesce ad inseguire la prima parte della simulazione in modo soddisfacente, mentre nell'ultima parte il modello ISO continua a sovrastimare la potenza erogata, sebbene in modo minore rispetto alla simulazione effettuata con i parametri ricavati dal motore GE t700. La forma della risposta è comunque riprodotta in modo fedele. L'incapacità del modello di riprodurre in maniera rigorosa l'erogazione della potenza al carico è da imputarsi all'indisponibilità dei parametri necessari alla sua messa a punto.

Capitolo 8

Conclusioni e sviluppi futuri

8.1 Conclusioni

In questo lavoro di tesi è stato sviluppato un modello termodinamico semplificato per la simulazione di motori turboalbero di impiego elicotteristico.

Il modello è concepito per essere integrato in codici di simulazione di Meccanica del Volo (anche real-time), in modo da aumentarne la capacità di descrivere fedelmente il comportamento dell'elicottero al di fuori delle condizioni di regime stazionarie, estendendo l'analisi a condizioni di volo manovrato dove risultano significativi i transitori nell'erogazione della potenza.

Il modello è stato implementato sia in linguaggio Matlab, sia in linguaggio C. È stata posta particolare attenzione nella possibilità di adattamento al maggior numero possibile di propulsori per elicotteri e sono stati forniti gli strumenti per personalizzare il codice in base alle esigenze dell'utente.

Inizialmente, il modello è stato sviluppato in una versione fortemente dipendente da funzioni parametriche, ricavate dal comportamento di un propulsore esistente e ben documentato. Questa versione, capace di riprodurre con errori minimi il funzionamento di tale propulsore, è servita da base di lavoro per lo sviluppo di versioni successive caratterizzate da una maggiore indipendenza da andamenti parametrici ricavati dalla letteratura, capaci di riprodurre efficacemente il comportamento di una classe di propulsori a turbina per elicotteri a partire da un numero ridotto di dati di modellazione.

Ogni versione del modello è stata validata sia considerando la risposta a regime in diverse situazioni di funzionamento, sia la sua risposta dinamica ad una portata di combustibile con profilo temporale a scalino, comparando i risultati con dati sperimentali disponibili in letteratura.

Il modello termomeccanico del propulsore è stato quindi integrato dallo sviluppo di un controllore di tipo PID che ne governa la dinamica secondo una

logica del tipo correntemente utilizzato nei FADEC (Full Authority Digital Engine Control). Tale controllore agisce direttamente sull'entità del flusso di carburante immesso in camera di combustione per regolare la potenza raccolta sul secondo albero del propulsore.

Il modello sviluppato è stato infine adattato al caso di un propulsore correntemente utilizzato da AgustaWestland Spa, ricevendo in ingresso la storia della portata di combustibile corrispondente ad una traiettoria manovrata di discesa rapida. I dati di ingresso, così come quelli di confronto per i risultati, sono stati forniti da un modello di Meccanica del Volo validato in uso presso l'azienda.

I risultati ottenuti dalle simulazioni mostrano come il modello sia facilmente estendibile ad applicazioni diverse attraverso una dozzina di parametri caratteristici del motore da simulare. Durante il processo di validazione, confrontandosi con dati sperimentali, si è dimostrato come il comportamento di un motore reale venga ben approssimato dal modello. Confrontandosi poi con software di simulazione più sofisticati per due diversi motori, si è dimostrato come i risultati ottenuti riproducano in modo accurato la dinamica del propulsore, a patto di calibrare accuratamente i parametri di personalizzazione.

I valori degli errori commessi sono giustificati dall'esiguità del numero di parametri utilizzati per caratterizzare ogni motore: l'accuratezza del modello è in linea con l'intento di ottenere uno strumento di simulazione veloce da mettere a punto attraverso poche grandezze di personalizzazione.

La principale limitazione di tale modello è quella di simulare il funzionamento del propulsore in condizioni non troppo lontane dai regimi di funzionamento tipici. Nonostante il comportamento del modello si sia dimostrato sempre rispondente alla fisica del problema, non è possibile fornire un riscontro di validità della simulazione per regimi di funzionamento del motore molto lontani da quelli nominali perché parte della modellazione è avvenuta a partire da dati sperimentali ricavati in condizioni vicine a quelle di progetto.

8.2 Sviluppi futuri

La continuazione più naturale di questo lavoro di tesi consiste nell'effettiva integrazione del modello termomeccanico del propulsore all'interno di un codice di simulazione di Meccanica del Volo dell'elicottero, da validare attraverso il confronto con strumenti analoghi, di tipo proprietario e quindi non interamente sotto il controllo dell'utente, ovvero di dati provenienti da prove di volo.

Quest'attività sarebbe in grado di mettere in luce i punti di forza e le lacune del modello, fornendo indicazioni per ulteriori correzioni e sviluppi che lo mettano in grado di adattarsi con fedeltà a maggior numero possibile di applicazioni.

Il codice potrebbe essere poi reso più complesso, modellando dei fenomeni che nel presente lavoro sono stati trascurati per motivi di semplicità e di tempo. In particolare, potrebbe essere sviluppato un modello di trasmissione e dispersione del calore, un modello di usura dei componenti meccanici e di stima della variazione dei rendimenti dei vari componenti in base alle condizioni di impiego del motore.

Si potrebbe infine considerare la possibilità di implementare una logica di co-simulazione per separare le dinamiche più veloci (nella gran parte delle applicazioni costituite dalle quelle termiche), dalle dinamiche più lente (generalmente quelle meccaniche): in tal modo sarebbe possibile rilassare il passo di integrazione temporale del codice e ottenere un codice ancora più veloce nell'esecuzione.

Ringraziamenti

Il primo ringraziamento va al Professor Trainelli, per la gentilezza e la disponibilità dimostrata durante la preparazione di questo lavoro: la sua guida è stata preziosa e fondamentale durante tutti questi mesi.

Un ringraziamento particolare va all'Ingegnere Leonello e all'Ingegnere Bianco Mengotti che hanno messo a mia disposizione il loro tempo e la loro professionalità ogni volta che necessitavo di un loro aiuto, con pazienza e gentilezza.

Alla mia famiglia, che ha reso possibile tutto quanto, vanno il mio affetto e la mia gratitudine. In particolare ai miei nonni, che avrei voluto abbracciare più spesso.

Grazie ad Irene, che ha avuto pazienza durante tutti questi esami.

Un grazie di cuore a tutti i miei amici, che hanno condiviso con me degli anni belli e intensi: Nicolò, Cristy, Paolo, Bombo, Maria Chiara, Edo, Fede, Luca, Carlos, Daniele, Francesco, Hombre, Giorgio, Juri, Simo, Barbara, Vittorio, Zizzius, Matteo, Javi, Angelo, Lucia, Balza, Geppo, Giovanni, Romeo, Arrigo, Srdjan, Riccardo, Fede, Ale.

Bibliografia

- [1] P.P. Walsh and P. Fletcher. *Gas Turbine Performance, Second Edition*. Blackwell Science, Limited, 2004.
- [2] U. Ghezzi. *Motori per aereomobili*. Politecnico. Manuali. Libreria Clup, 2005.
- [3] M. Arra. *Elicottero (l')*. Biblioteca tecnica Hoepli. Hoepli, 2001.
- [4] A. Linke-Diesinger. *Systems of Commercial Turbofan Engines: An Introduction to Systems Functions*. Springer-Verlag, 2008.
- [5] M. Ballin. *A High Fidelity Real Time Simulation of a small Turboshaft Engine*. NASA technical memorandum 100991, 1988.
- [6] A. Duyar, Z. Gu, and J. Litt. A simplified dynamic model of the t700 turboshaft engine. *Journal of the American Helicopter Society*, 40:62 – 70, 1995.
- [7] S. Camporeale, B. Fortunato, and M. Mastrovito. A modular code for real time dynamic simulation of gas turbines in simulink. *Transactions of the ASME*, 128:506 – 517, 2006.
- [8] G. Kocer. Aerothermodynamic modeling and simulation of gas turbines for transient operating conditions, 2008.
- [9] O. Uzol. A new high-fidelity transient aerothermal model for real-time simulations of the t700 helicopter turboshaft engine. *Isi Bilimi ve Teknigi Dergisi*, 31:37–44, 2010.
- [10] W. Visser and M.J. Broomhead. *GSP A generic object-oriented gas turbine simulation environment*, 2000.
- [11] W.P.J. Visser, M.J. Broomhead, and J. van der Vorst. Terts, a generic real-time gas turbine simulation environment. *NLR-TP-2002-069*, 1:1 – 20, 2001.

- [12] A. Guha. An efficient generic method for calculating the properties of combustion products. *IMechE*, 215:375–387, 2000.
- [13] Coordinating Research Council. *Handbook of Aviation Fuel Properties*, 1983.
- [14] M. Galassi, J. Davies, J. Theiler, et al. *GNU Scientific Library*. 2011.
- [15] G. Magnani, G. Ferretti, and P. Rocco. *Tecnologie dei sistemi di controllo*. Istruzione scientifica. McGraw-Hill Companies, 2007.
- [16] K.J. Åström and T. Hägglund. *PID Controllers: Theory, Design and Tuning*. Setting the standard for automation. Isa, 1995.
- [17] P.A. Jackson, K. Munson, L.T. Peacock, S. Bushell, and D. Willis. *Jane's All the World's Aircraft 2012-2013*. IHS Jane's All the World's Aircraft. Jane's Information Group, 2012.
- [18] T. Schobeiri. Getran: A generic, modularly structured computer code for simulation of dynamic behavior of aero- and power generation gas turbine engines. *ASME, Turbomachinery Committee*, 116:483–494, 1994.
- [19] D. Pashayev, D. Askerov, C. Ardil, and R. Sadiqov. Numerical modeling of gas turbine engines. *International Journal of Computational and Mathematical Sciences*, 1:53–60, 2007.
- [20] P.A. Jackson, K. Munson, L.T. Peacock, S. Bushell, and D. Willis. *Jane's All the World's Aircraft 2011-2012*. IHS Jane's All the World's Aircraft. Jane's Information Group, 2011.