

UNIVERSITA' POLITECNICA DELLE MARCHE
FACOLTA' DI INGEGNERIA



Corso di Laurea Magistrale in
Ingegneria Informatica e dell'Automazione

***PROGETTO F2- Software-In-The-Loop PX4 –
Gazebo***

Docente:
Alessandro Freddi

Studenti:
Luca Liberatore
Ciro Maccarone

Anno Accademico 2022-2023

Sommario

Capitolo 1 - Obiettivo del progetto	3
Capitolo 2- Installazione e Configurazione delle applicazioni.....	5
Gazebo e Px4.....	6
QGroundControl	8
Capitolo 3 – Test Effettuati	9
Volo senza guasti	10
Volo con guasti: Consumo eccessivo della batteria	20
Volo con guasti: CA_ROTOR_CT.....	25
Volo con guasti: MC_ROLL_P	27
Conclusioni.....	30

Capitolo 1 - Obiettivo del progetto

Il progetto si pone come obiettivo la realizzazione di una connessione Software In The Loop (SITL) tra PX4 e Gazebo. Ci si concentrerà poi sullo studio dei log di segnale e sulla simulazione di alcuni guasti. In termini più approfonditi: si partirà dall'installazione dei software necessari alla simulazione del drone e, successivamente, si inietteranno dei guasti al fine di studiarne le conseguenze sul drone.

Al fine di raggiungere gli obiettivi di progetto verranno utilizzati tre applicativi: PX4, Gazebo e QGroundControl. Partendo dal primo, PX4 è un sistema open source autopilota in grado di controllare il comportamento di una varietà di droni. La sua natura open source ha permesso una diffusione globale del software nel mondo degli aerei senza equipaggio (UAV). PX4 fornisce una serie di caratteristiche per il controllo del volo, quali: il controllo di altitudine, il posizionamento GPS, il controllo degli attuatori e l'interfacciamento con i vari sensori.



Gazebo è un simulatore tridimensionale, anch'esso open source. Per la robotica, in particolare, viene utilizzato per la simulazione del comportamento di robot mobili e droni. Le funzionalità di Gazebo permettono di emulare dinamiche reali come la fisica, le forze e le collisioni. L'utilità di Gazebo risiede anche nella capacità di poter effettuare dei test in un ambiente sicuro, prima di effettuare un approccio con hardware reale. Date le sue caratteristiche, Gazebo è spesso utilizzato in combinazione con PX4 per la simulazione dei droni.



In ultima analisi, si farà uso anche dell'applicativo QGroundControl. QGroundControl è un'applicazione open source progettata per il controllo e la configurazione di UAV e droni. L'utilizzo principale che se ne farà del software sarà da supporto grafico (visuale dall'alto) e come strumento per impostare una missione di rotta del drone; infatti, l'applicazione permette agli utenti di pianificare missioni di volo in anticipo con relativo percorso preimpostato. In aggiunta, lo strumento verrà utilizzato per andare a modificare alcuni parametri del drone al fine di simulare i guasti. Una volta effettuate le simulazioni QGroundControl permette di scaricare i file di log, da cui è possibile effettuare delle analisi dai grafici forniti dallo strumento.



In definitiva, si utilizzerà PX4 da terminale per avviare la simulazione e, di conseguenza, Gazebo per la visualizzazione 3D del drone; in aggiunta, ci si avvarrà del supporto del software QGroundControl per l'impostazione della rotta del drone e per l'analisi dei log. L'installazione e la configurazione degli applicativi verrà analizzata nel prossimo paragrafo.

Capitolo 2- Installazione e Configurazione delle applicazioni

Il progetto è stato interamente sviluppato su OS Ubuntu 20.04.6 LTS, il motivo è dovuto alla incompatibilità del simulatore Gazebo con Windows.

Se dovesse essere necessaria una guida per l'installazione dello OS, l'argomento esula dallo scopo del presente documento, ma esistono diverse guide su internet su come scaricare il software (completamente gratuito) e installarlo tramite USB, e come trasformare quest'ultima in eseguibile per consentire l'installazione sulla propria macchina. Il lettore può partire da qui se necessario.

Nel capitolo corrente indicheremo le istruzioni da impartire da terminale per installare le applicazioni usate per il progetto. Le istruzioni sono disponibili sulla pagina della documentazione di PX4, nella sezione Development/GettingStarted/Toolchain Installation/Ubuntu Setup.

Lasciamo il link per comodità:

https://docs.px4.io/main/en/dev_setup/dev_env_linux_ubuntu.html#ros-gazebo-classic

Gazebo e Px4

Nel capitolo ROS Noetic/Ubuntu 20.04 le due applicazioni vengono installate insieme in pochi semplici comandi che riportiamo di seguito:

git clone https://github.com/PX4/PX4-Autopilot.git --recursive

il comando si limita a scaricare da Github la repository di Px4, l'opzione recursive serve per scaricare anche i repository git incorporati all'interno della repository principale. Se per qualche motivo doveste avere problemi ad eseguire il download, riprovate senza l'opzione appena descritta, nel nostro contesto non sarà infatti necessaria.

Per l'installazione eseguire il comando per avviare l'eseguibile, accettando ogni richiesta di conferma:

bash ./PX4-Autopilot/Tools/setup/ubuntu.sh --no-sim-tools --no-nuttx

Dopodiché riavviate il computer.

Installate dunque le dipendenze:

sudo apt-get install protobuf-compiler libeigen3-dev libopencv-dev -y

Dopodiché seguire il link per l'installazione di ROS Noetic Full che includerà Gazebo Classic al suo interno

(<http://wiki.ros.org/noetic/Installation/Ubuntu#Installation>)

Consenti al dispositivo di scaricare Ros dal sito:

sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu \$(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

Imposta le chiavi

sudo apt install curl # if you haven't already installed curl

curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc |

sudo apt-key add -

Prima dell'installazione vera e propria, assicuratevi di aggiornare il sistema operativo:

sudo apt update

Eseguite dunque il comando per scaricare e installare la versione Ros Noetic Desktop Full, che installerà anche Gazebo:

sudo apt install ros-noetic-desktop-full

Ros era inizialmente stato pensato nel progetto come il software con il quale impartire i comandi di movimento al drone, ma date le difficoltà tecniche di installazione e d'uso si è preferito optare per il programma più intuitivo, QGroundControl.

Una volta installato posizionarsi con il terminale sulla cartella di Px4, con il comando `cd` nella cartella "PX4-Autopilot" ed impartire il comando: `make px4_sitl gazebo`.

Quest'ultimo monterà la simulazione aprendo una connessione SITL e il prompt dei comandi può essere usato come interfaccia commander, dal quale monitorare i parametri del drone ed impartire manualmente dei comandi al drone QUADROTOR.

QGroundControl

Dalla pagina della documentazione di Px4, in alto al sito vedrete un link per il sito di QGroundControl. Da lì, potrete accedere al sito dell'app, procedete verso la sezione download e selezionate la versione per Ubuntu.

Riportiamo per comodità i comandi di installazione anche qui: posizionatevi sul prompt dei comandi, e se installate l'app per la prima volta, immettete questi comandi:

```
sudo usermod -a -G dialout $USER
```

```
sudo apt-get remove modemmanager -y
```

```
sudo apt install gstreamer1.0-plugins-bad gstreamer1.0-libav gstreamer1.0-gl -y
```

```
sudo apt install libqt5gui5 -y
```

```
sudo apt install libfuse2 -y
```

Procedete a fare un logout dal PC per consentire il cambio di permessi utente e rifate login.

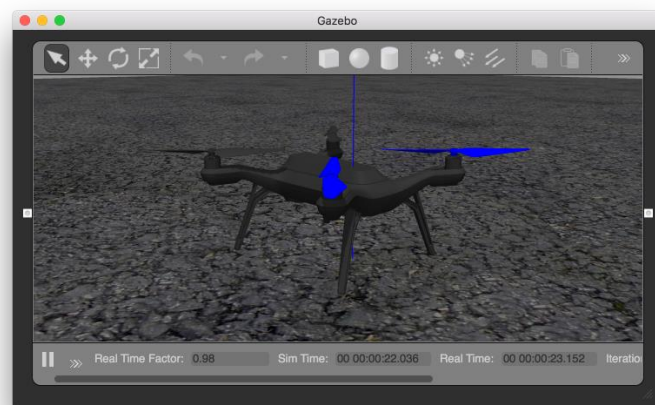
Scaricate l'immagine dell'app, per aprire l'applicazione eseguite infine i comandi:

```
chmod +x ./QGroundControl.AppImage
```

Come alternativa fate doppio clic sull'immagine scaricata.

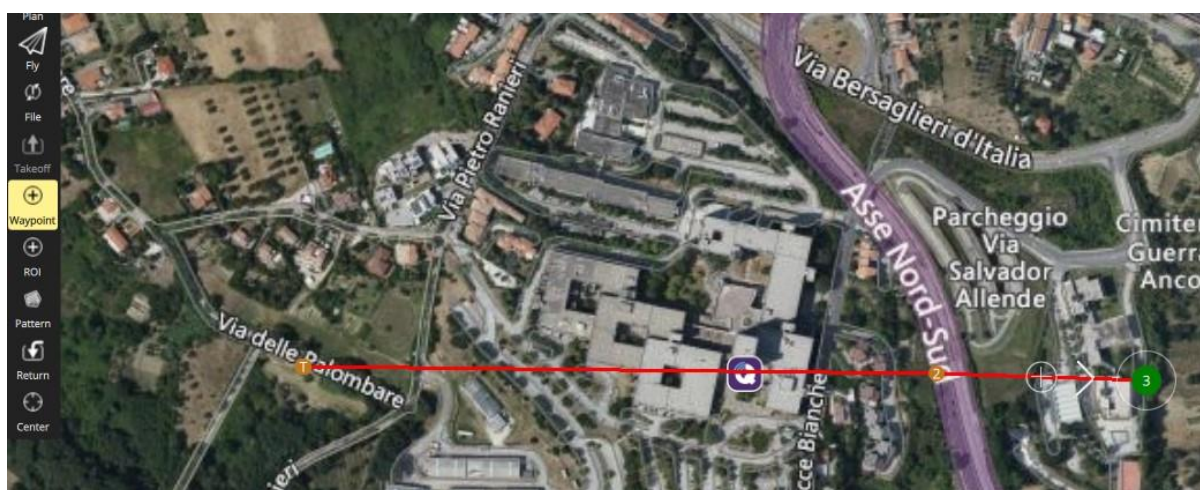
Capitolo 3 – Test Effettuati

Questo capitolo esporrà tutte le simulazioni effettuate tramite i software sopracitati e si farà uso del drone *Quadrotor* messo a disposizione da Gazebo. La simulazione del volo verrà effettuata tramite l'impostazione della missione su QGroundControl. La missione è un file che registra il viaggio che deve fare il drone, salvando parametri ad esempio come la velocità da mantenere, l'altezza e le tappe che deve percorrere.



Volo senza guasti

La prima simulazione che verrà effettuata è di tipo senza guasto; l'idea è quella di impostare un volo abbastanza lungo e rettilineo in modo da poter evidenziare eventuali comportamenti anomali da parte del drone quando, nelle fasi successive, verranno iniettati dei guasti. Innanzitutto, si imposta una rotta su QGroundControl cliccando sul pulsante "plan", presente nel riquadro in alto a sinistra. Una volta fatto ciò si può scegliere lo scenario "blank", il quale fornirà come output un menù a tendina come nella figura sottostante. Da qui si effettua il decollo del drone (*takeoff*) e successivamente si potranno impostare i punti di interesse che il drone dovrà seguire durante il proprio volo (*waypoint*). Per aggiungere i punti basta cliccare sulla mappa. È possibile salvare la missione impostata cliccando sull'opzione "file", che consente di caricare le missioni precedentemente salvate (file *.plan) e di salvarne di nuovi. Una volta caricato il file, premere il tasto azzurro in alto "Upload Required". Una volta caricato, cliccare di nuovo su "Fly" e far scorrere il pulsante in basso per avviare la simulazione.



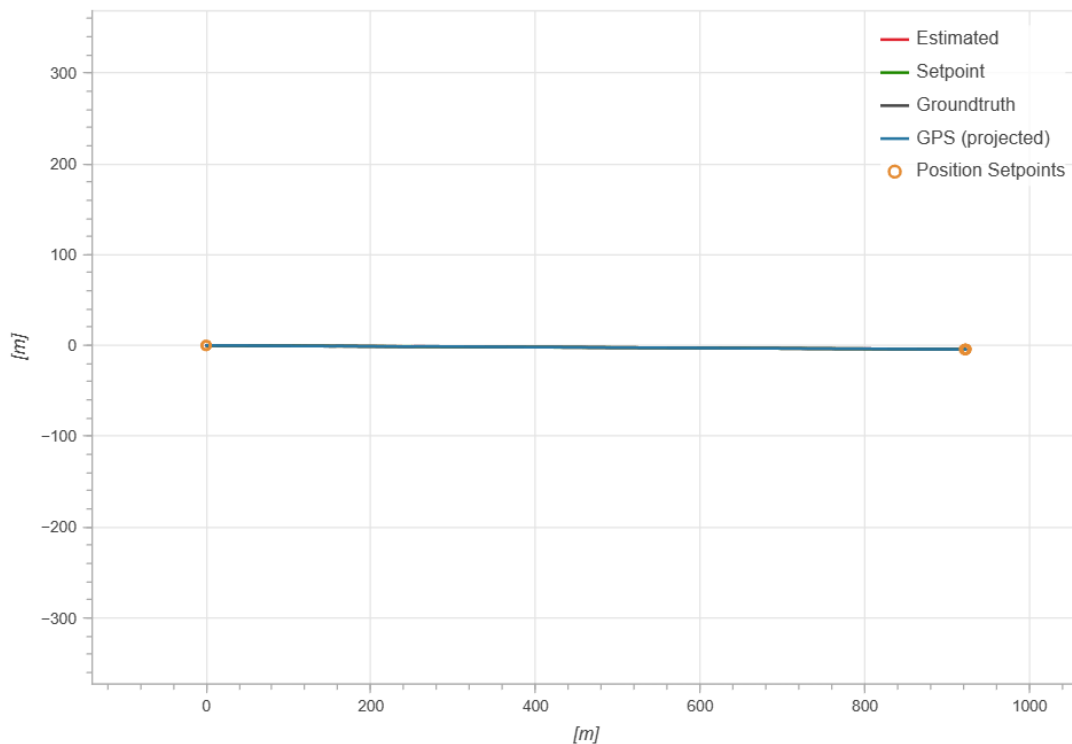
Una volta effettuato tutto il volo, vengono registrati e scaricati i log. La missione impostata per il progetto prevede le seguenti caratteristiche:

- Lunghezza: 923 metri
- tempo di percorrenza: 3 minuti e 5 secondi
- velocità: 5 m/s

Al termine della missione il drone verrà fatto atterrare correttamente.

L'analisi dei log e quindi dei relativi grafici è permessa dal sito ufficiale di PX4 all'indirizzo: <https://logs.px4.io/> Una volta effettuato il caricamento del file di

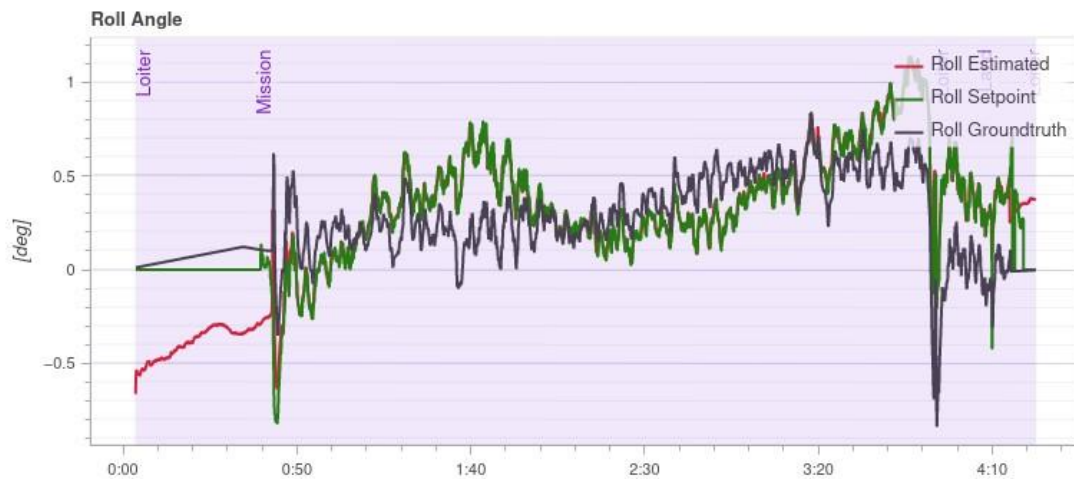
interesse è possibile andare a studiare il comportamento di alcune delle caratteristiche del drone. I risultati che si ottengono sono visualizzabili nelle figure riportate sotto.



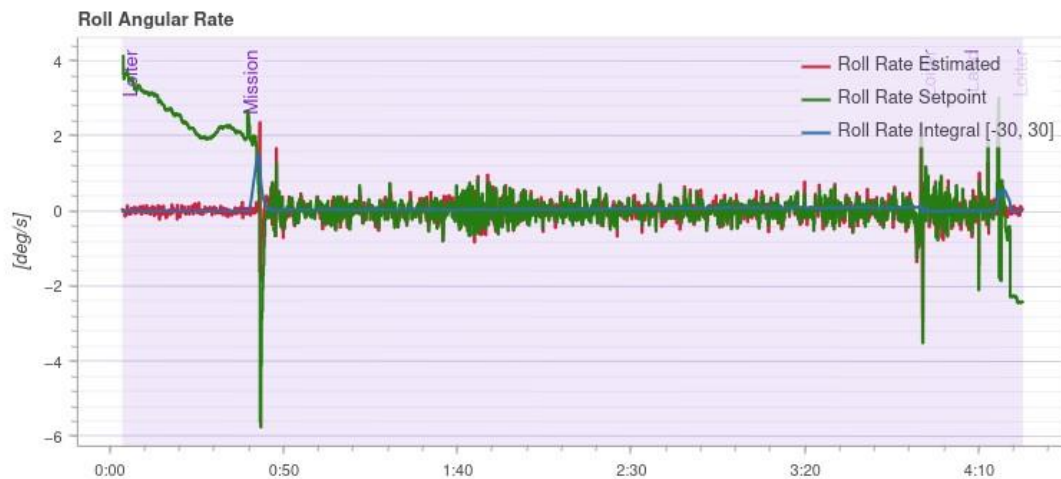
In questa prima figura si può osservare l'andamento rettilineo del drone nello spazio. Nella figura successiva si ha la stima dell'altitudine del drone che, in missione, è di circa 500 metri.



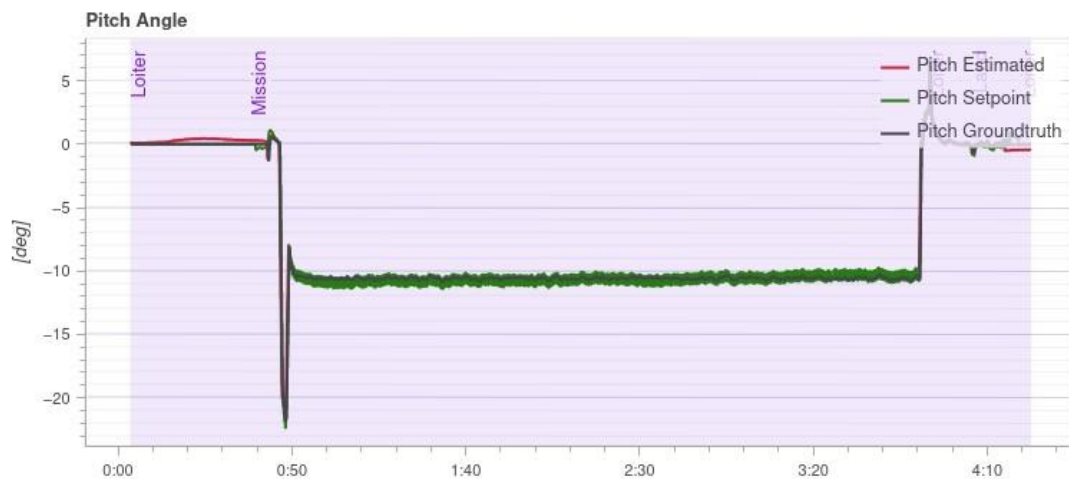
Nella figura sottostante si valuta l'angolo di rollio, il quale rappresenta l'inclinazione laterale dell'asse longitudinale del drone rispetto all'orizzonte; in altre parole, indica quanto il drone si inclina lateralmente durante il volo.



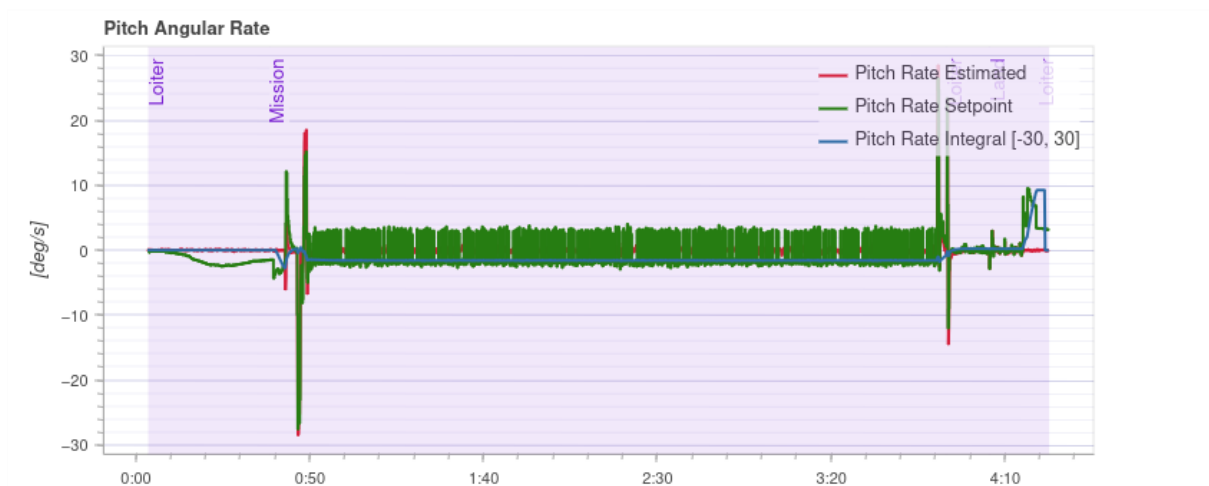
Il Roll Angular Rate rappresenta la velocità angolare di rollio del drone nel corso della missione, ovvero quanto velocemente il drone sta ruotando intorno al suo asse longitudinale.



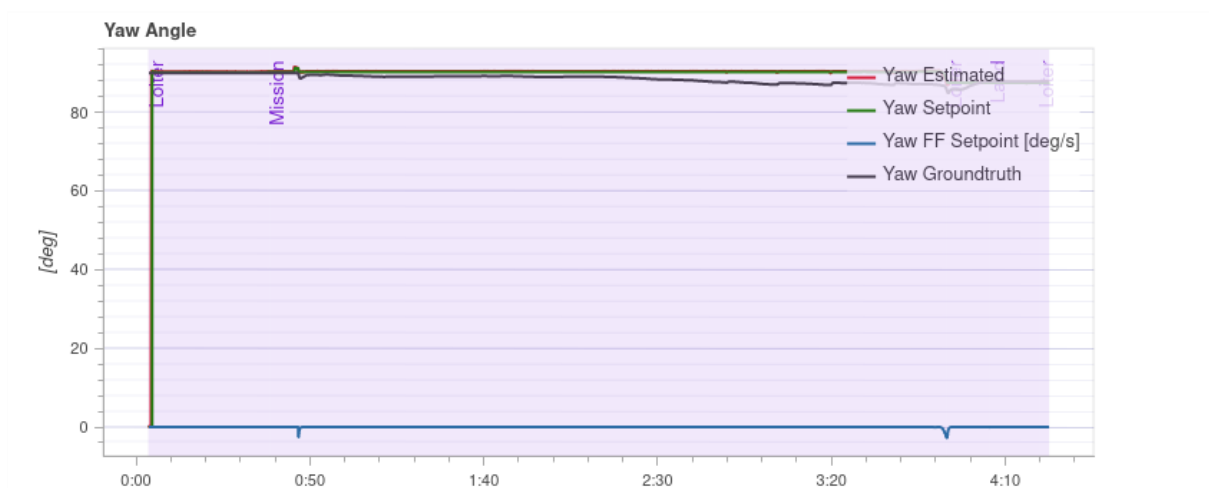
Il Pitch Angle rappresenta l'angolo di beccheggio del drone, cioè misura l'inclinazione anteriore o posteriore dell'asse longitudinale del drone rispetto all'asse y. In altri termini, indica quanto il drone si inclina in avanti o all'indietro.



Il Pitch Angular Rate indica la velocità angolare di beccheggio nel corso del tempo. Il tasso angolare di beccheggio misura quanto rapidamente il drone sta ruotando intorno all'asse trasversale, ovvero quanto velocemente si va ad inclinare in avanti o all'indietro.



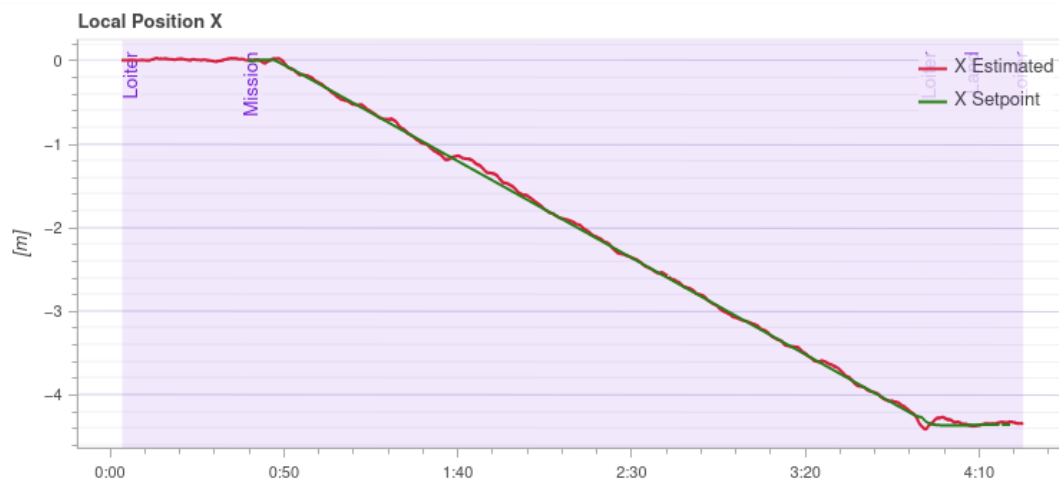
Lo Yaw Angle rappresenta l'angolo di imbardata di un drone. L'angolo di imbardata misura la rotazione intorno all'asse verticale del drone, indicando in che direzione il drone è orientato rispetto al nord magnetico o a un altro punto di riferimento.

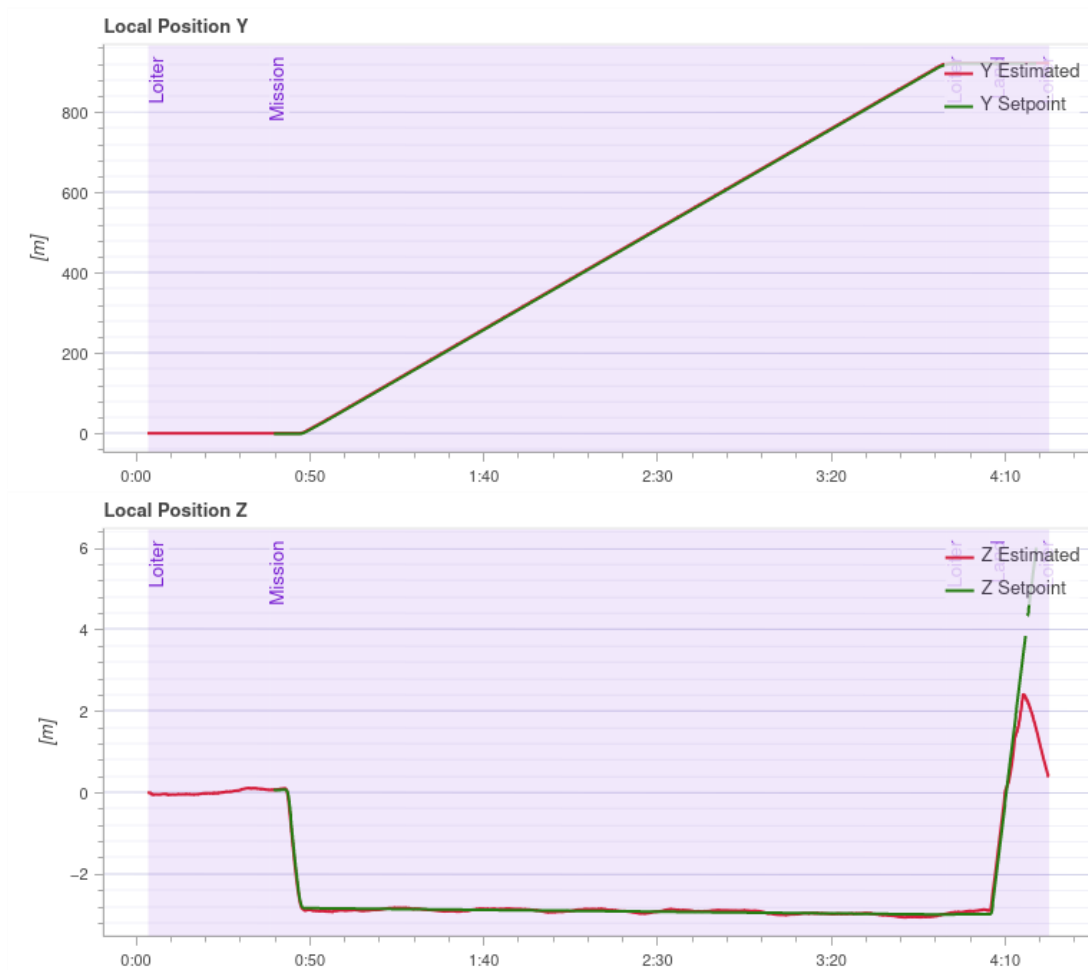


Lo yaw angular rate rappresenta la velocità angolare di imbardata del quadrotore durante la missione. Il tasso angolare di imbardata misura quanto rapidamente il drone sta ruotando intorno all'asse verticale, o meglio quanto velocemente sta girando su se stesso.

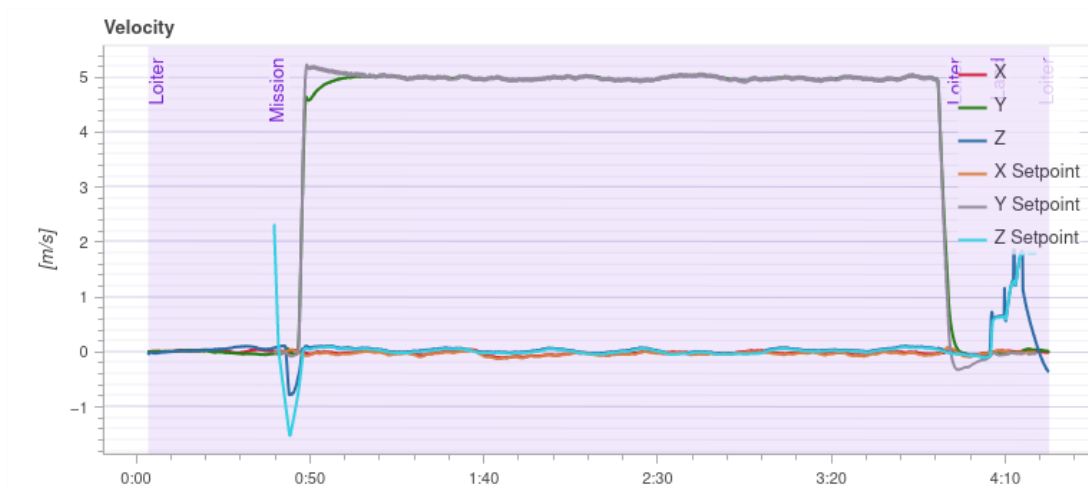


Successivamente si riportano i grafici che rappresentano la posizione del drone rispetto i tre assi cartesiani durante l'evoluzione della missione.

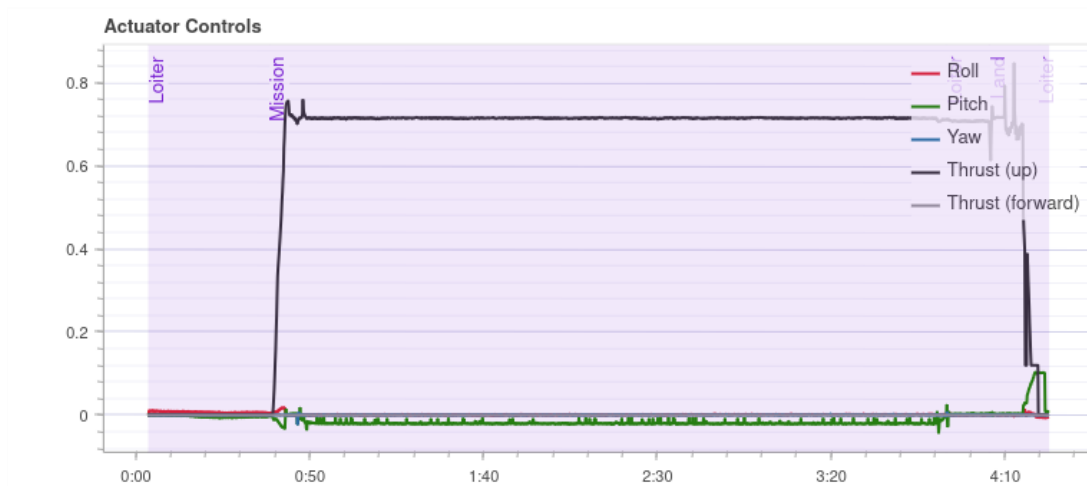




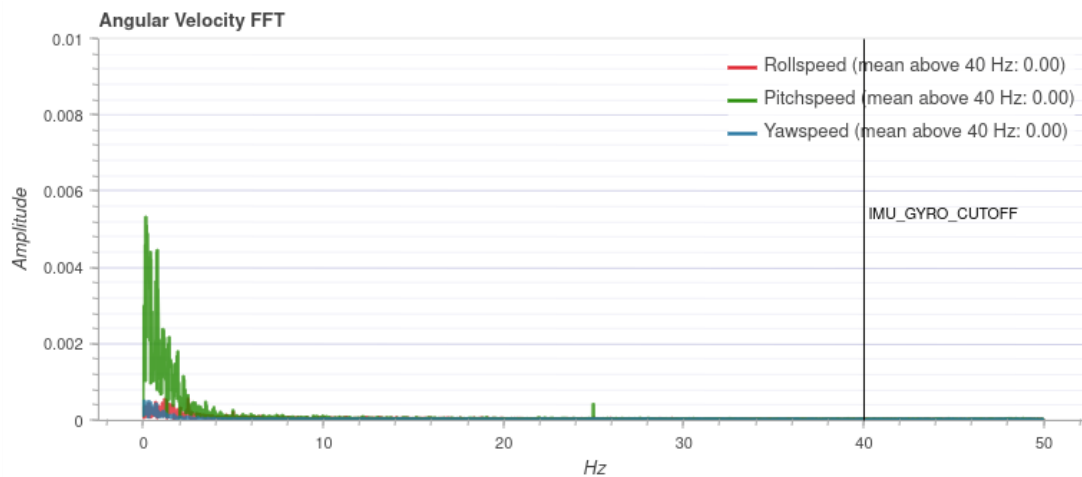
Il grafico della velocità rappresenta appunto la velocità del drone durante il tempo di osservazione. La caratteristica si può osservare nella figura successiva.



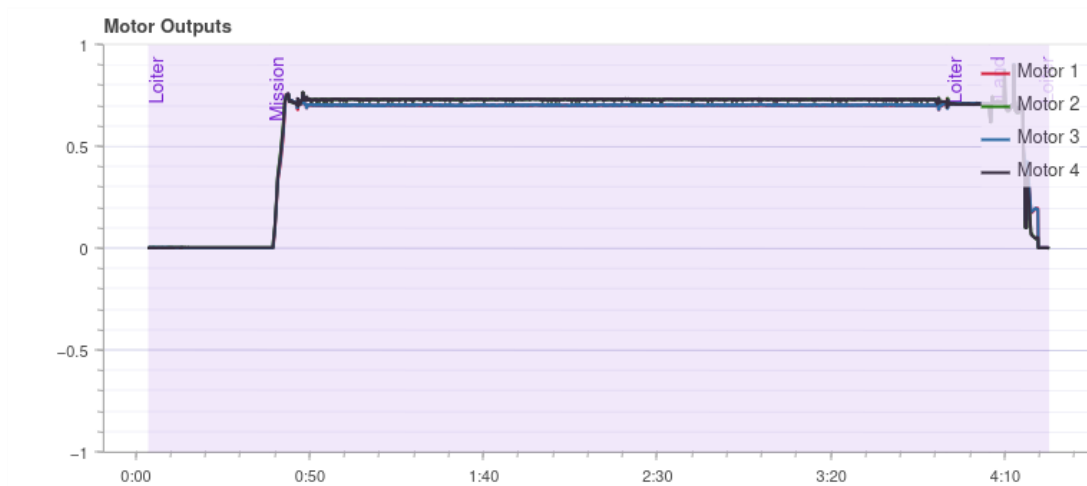
Il grafico sui controlli dell'attuatore rappresenta i comandi inviati ai vari attuatori o motori del drone nel corso del tempo.



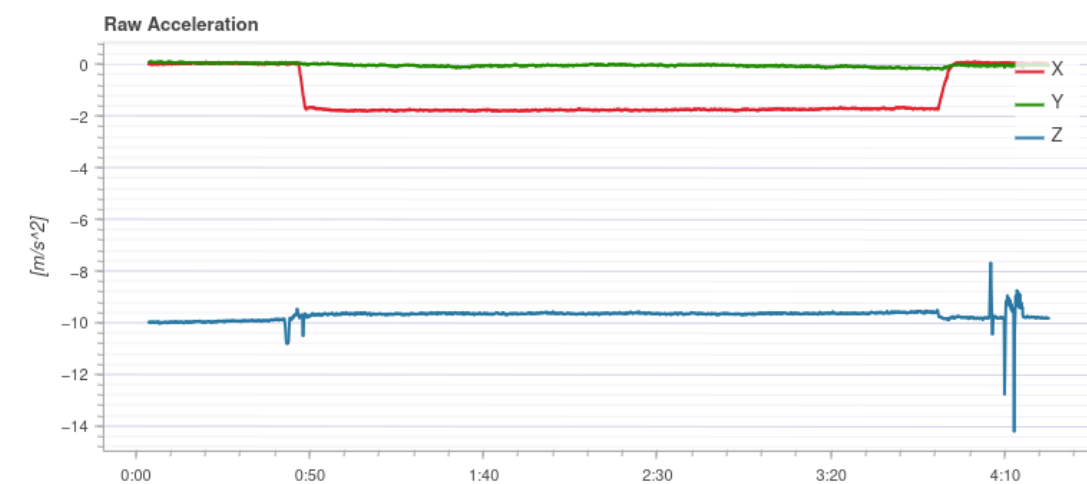
Nella figura sotto si può osservare il grafico della trasformata di Fourier della velocità angolare. In sintesi, il "grafico angular velocity FFT" misura le componenti delle velocità angolari di un veicolo in base alla frequenza, offrendo una visualizzazione delle diverse frequenze presenti nel movimento di rotazione del veicolo. Questo tipo di analisi può essere utile per diagnosticare problemi di stabilità, oscillazioni o altre anomalie nei sistemi di controllo.



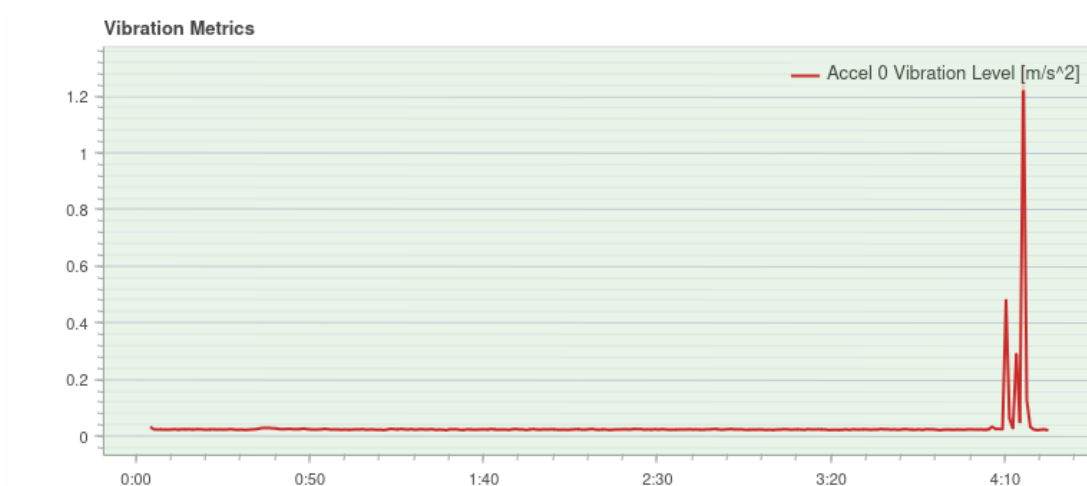
Nella figura successiva invece si può osservare la spinta erogata dai quattro motori del quadrotore, distribuita lungo l'asse temporale delle ascisse.



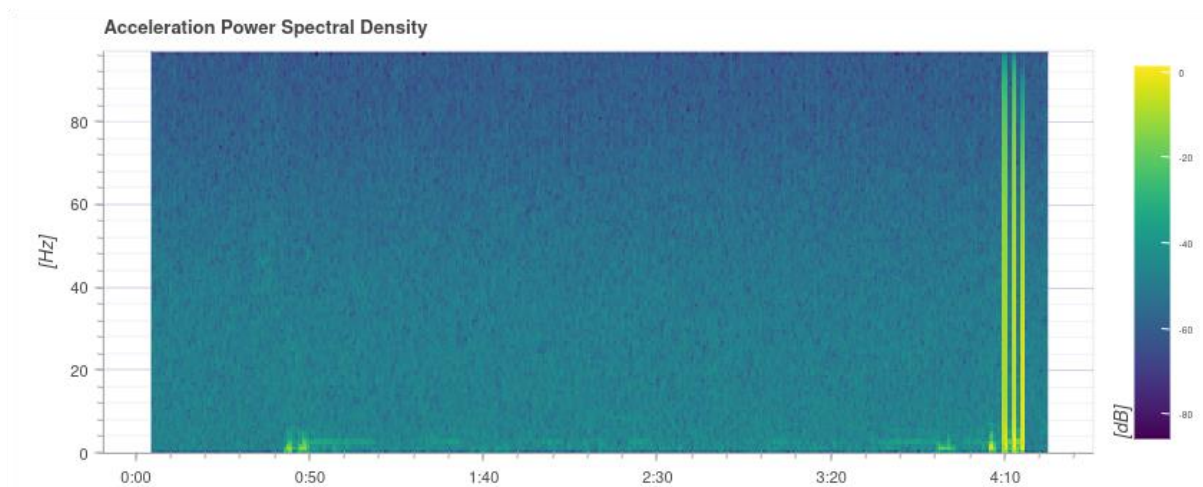
Il grafico delle "Raw Accelerations" (Accelerazioni Grezze) rappresenta le misurazioni grezze delle accelerazioni lungo i diversi assi del drone.



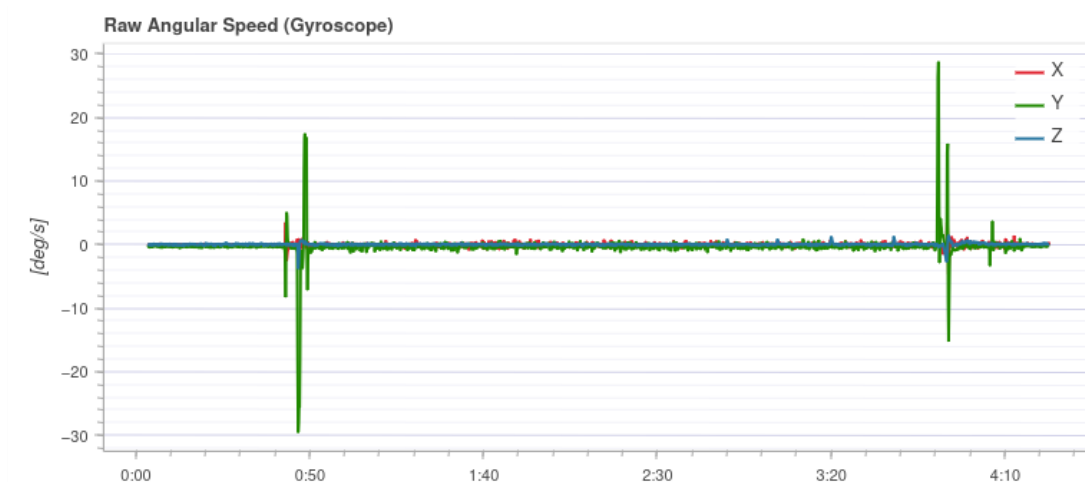
Di seguito si riporta il grafico sulle vibrazioni rilevate a bordo del drone durante il volo.



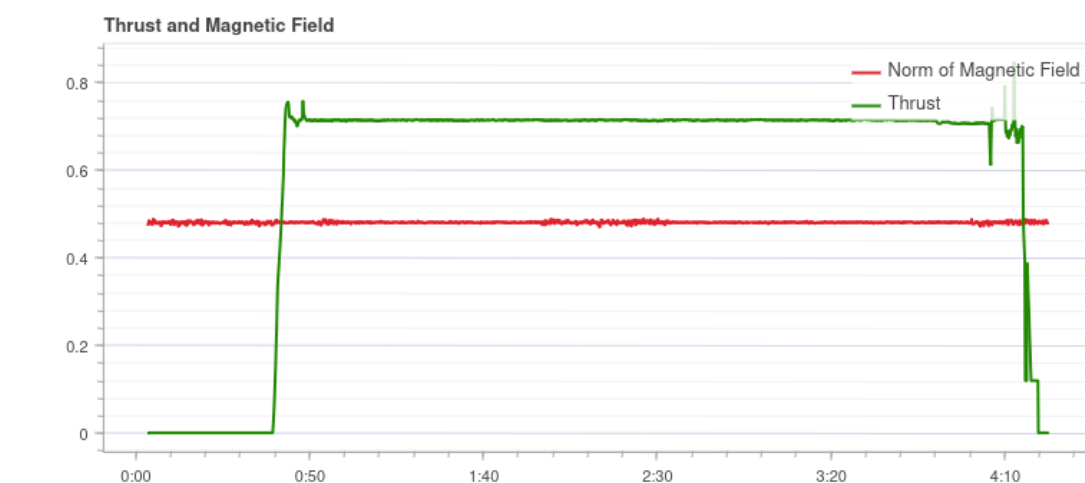
Il grafico dell'Acceleration Power Spectral Density (Densità Spettrale di Potenza dell'Accelerazione) è una rappresentazione delle componenti di frequenza delle accelerazioni misurate a bordo del drone.



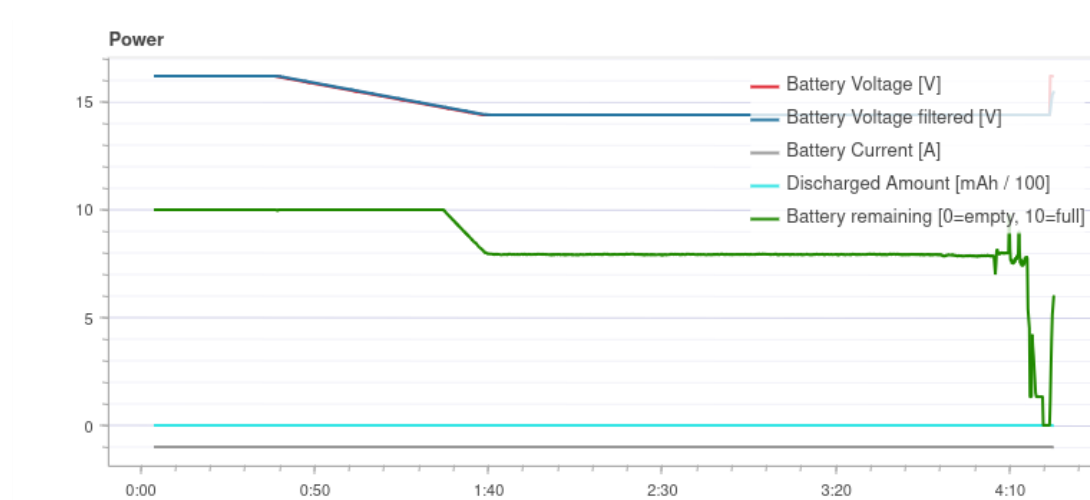
Dal grafico successivo invece si può osservare la velocità angolare del drone misurata dal giroscopio incorporato.



Nella figura successiva si osserva la spinta ed il campo magnetico.



Un altro fattore da tenere in considerazione è la carica della batteria ed il suo voltaggio, come si può osservare nella figura successiva.



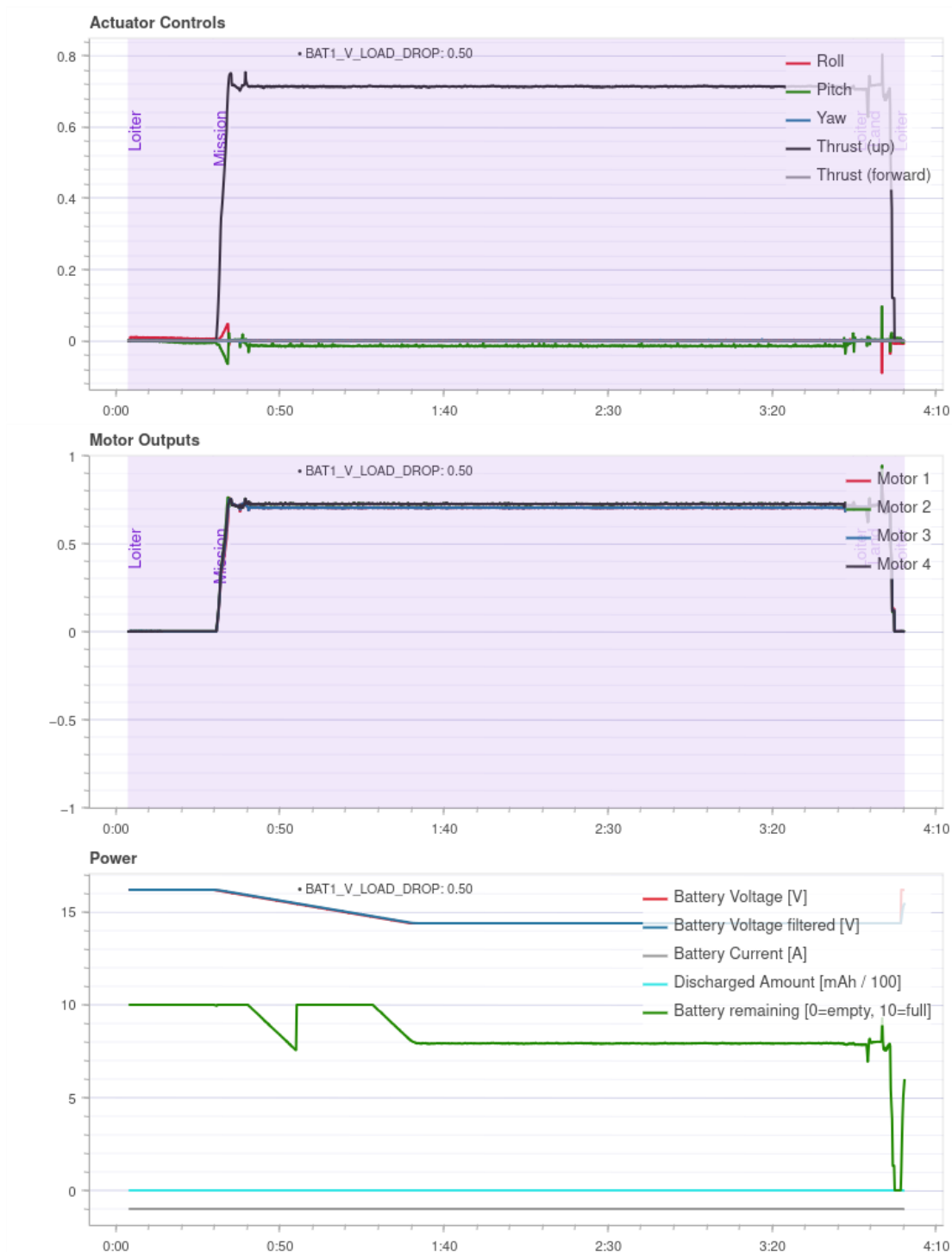
Volo con guasti: Consumo eccessivo della batteria

Il primo tentativo di iniezione di un guasto riguarda il consumo di batteria. Le modalità con cui è possibile agire sui parametri riguardano sia l'interfaccia grafica messa a disposizione di QGroundControl, per mezzo della sezione "parameters", sia PX4, tramite l'utilizzo del terminale tramite il comando: *"param set nome_parametro valore"*, il quale permette una modifica in volo del parametro di interesse.

Al fine di testare il comportamento del drone sono state effettuate due prove di modifica dei parametri riguardanti il consumo della batteria. Nel primo caso ci si è concentrati esclusivamente sulla modifica del parametro BAT1_V_LOAD_DROP. Tale parametro riguarda la velocità con cui la batteria si scarica. Il valore di default è 0.10, il valore minimo applicabile è 0.07 mentre il massimo 0.50; nella simulazione in esame si è assegnato il valore più alto possibile, in modo da avere un rapido consumo della carica. Affinché questo parametro sia correttamente interpretato dal sistema bisogna assegnare valori negativi al parametro BAT1_R_INTERNAL, che altrimenti entrerebbe in conflitto con il parametro di interesse.

I test sono stati effettuati sia andando a modificare il valore all'interno di QGroundControl, con successivo riavvio necessario del veicolo, sia in tempo reale tramite il terminale messo a disposizione di PX4.

Tuttavia, anche se dai log è evidente il calo della tensione di batteria, ad esso non corrisponde un aumento del thrust richiesto. Di conseguenza l'interpretazione che se ne trae è che, a livello di simulazione, questa caratteristica non è correttamente implementata in SITL PX4-Gazebo. I grafici riportati sotto fanno riferimento alla simulazione in cui il parametro viene modificato in live tramite la console PX4.



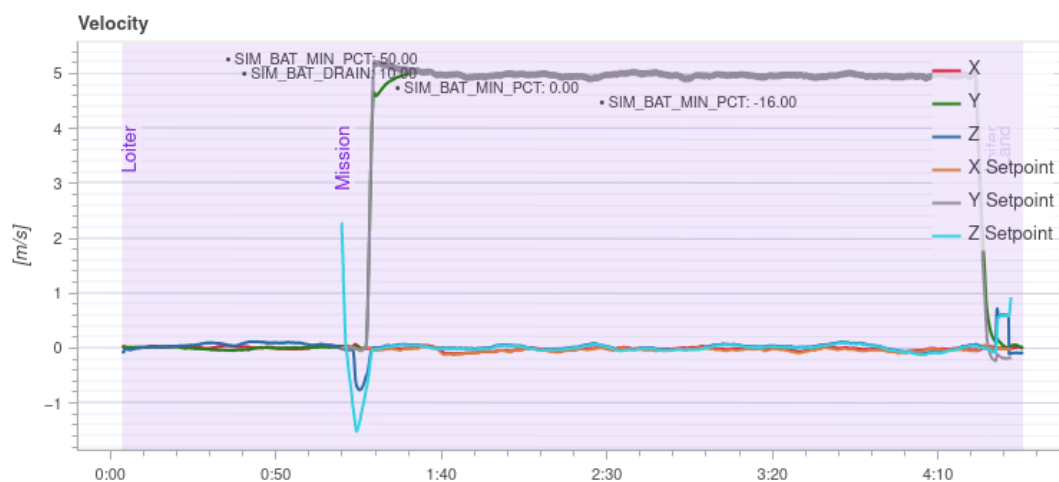
Come è possibile osservare, non si notano particolari differenze rispetto al caso senza guasti.

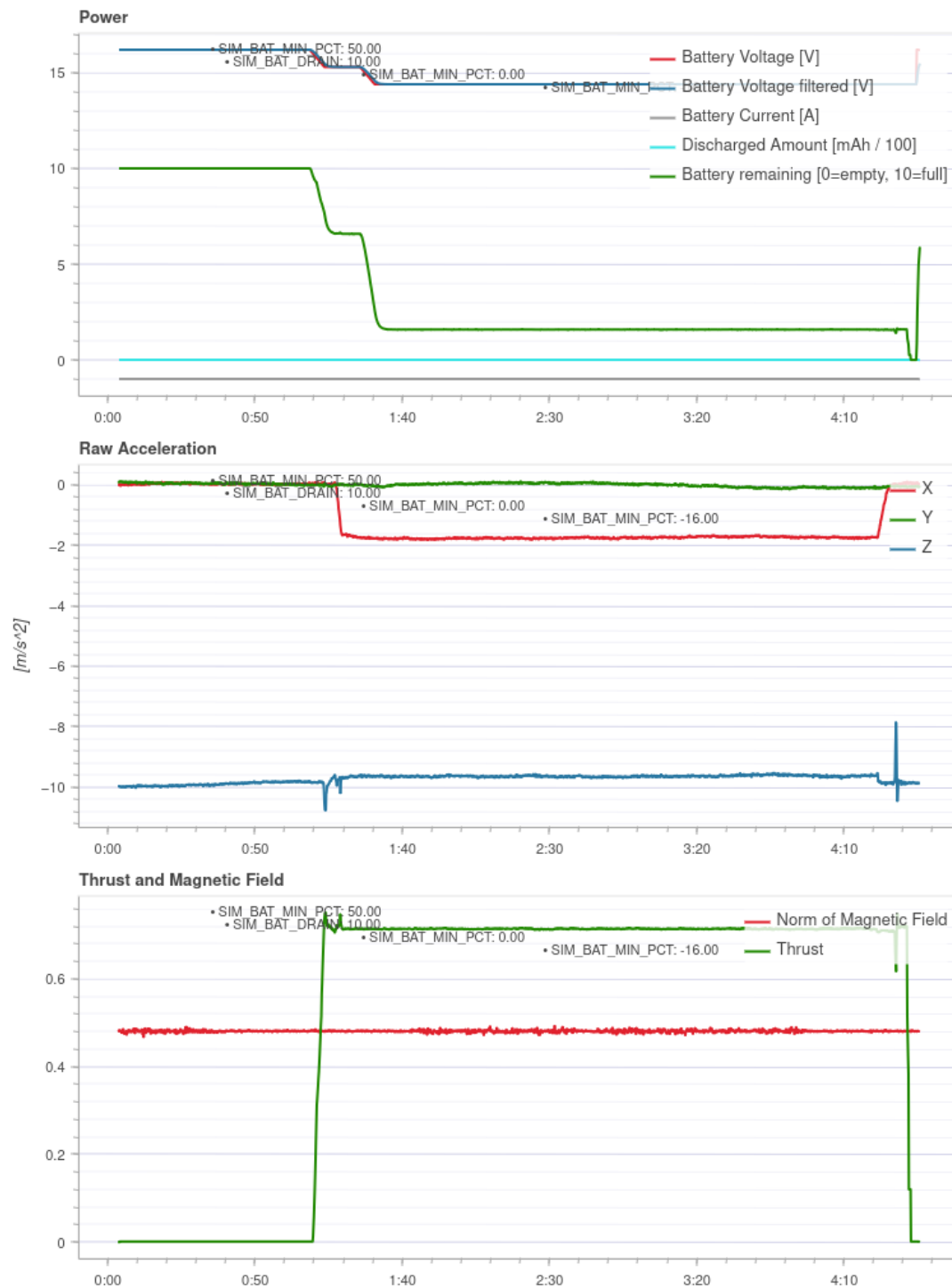
Un'ulteriore prova è stata dunque effettuata andando a modificare un altro parametro riguardante la batteria: SIM_BAT_MIN_PCT. Tale parametro di interesse è utile per andare a modificare la percentuale del livello di batteria presente nel drone durante il volo. Per i test sono stati utilizzati vari valori di entità molto bassa, sino a raggiungere il minimo consentito dal simulatore,

ovvero zero, opposto al massimo pari a 100. Tuttavia, si è notato che il sistema non è correttamente implementato, dato che somma un valore assoluto di 16 punti percentuali a qualsiasi valore venga immesso nel parametro.

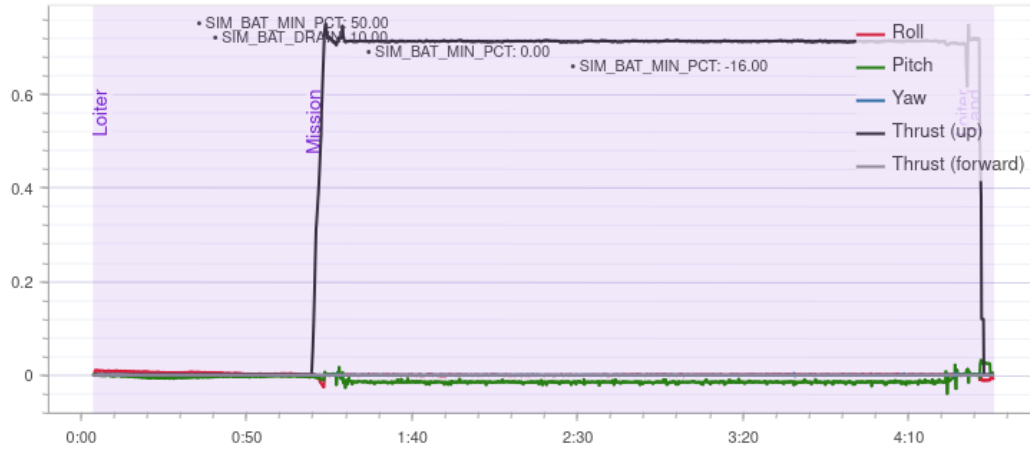
L'altro parametro che viene toccato è il SIM_BAT_DRAIN. Quest'ultimo riguarda il tempo di scaricamento della batteria in secondi. Il valore minimo settabile è di 1 secondo, il massimo 86400; il caso di default prevede un minuto. L'idea alla base di questa simulazione vuole verificare che ad un abbassamento della carica disponibile ha come conseguenza un aumento del thrust nel drone. Nelle varie simulazioni si è diminuito il valore di tale parametro, portandolo sino al caso più estremo con valore minimo, in modo da ottenere una repentina diminuzione della carica della batteria. Nell'esempio riportato sotto il valore è stato impostato a 10.

Tuttavia, in entrambe le simulazioni si è osservato come il drone effettivamente, sotto una certa soglia variabile, non riesce più a scaricarsi, se non fin quando atterra e vengono generati i failsafe e, di conseguenza, la batteria si scarica completamente. Dai risultati ottenuti per via sperimentale, si è dunque giunti alla conclusione che la gestione della batteria e le sue ripercussioni sul modello ideale non sono correttamente implementati nel simulatore. I grafici di questa simulazione sono riportati sotto. Anche in questo caso si può osservare un andamento del tutto comparabile con le esperienze precedenti senza guasti.

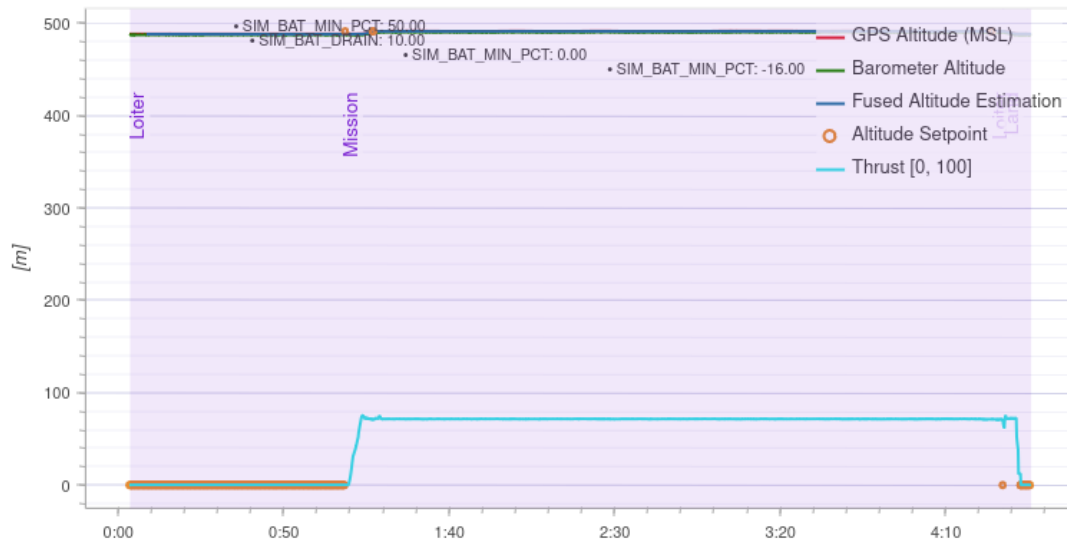




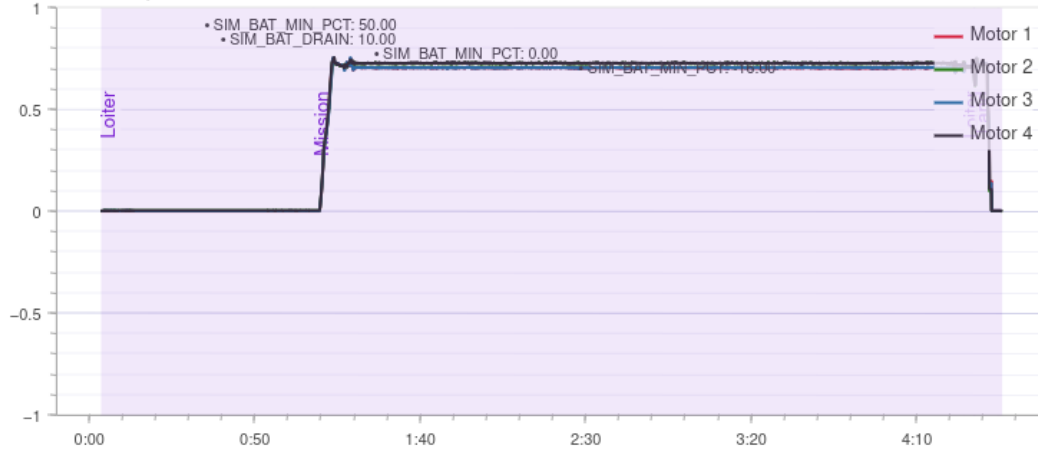
Actuator Controls



Altitude Estimate



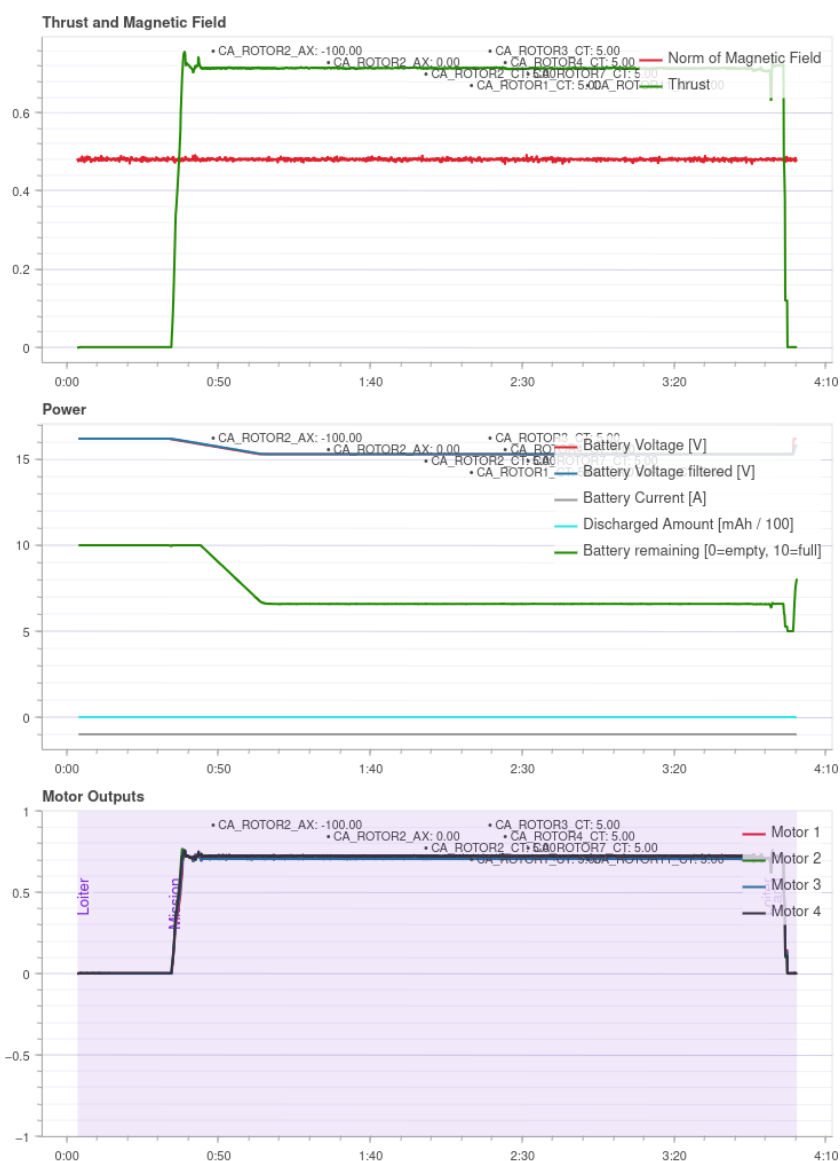
Motor Outputs

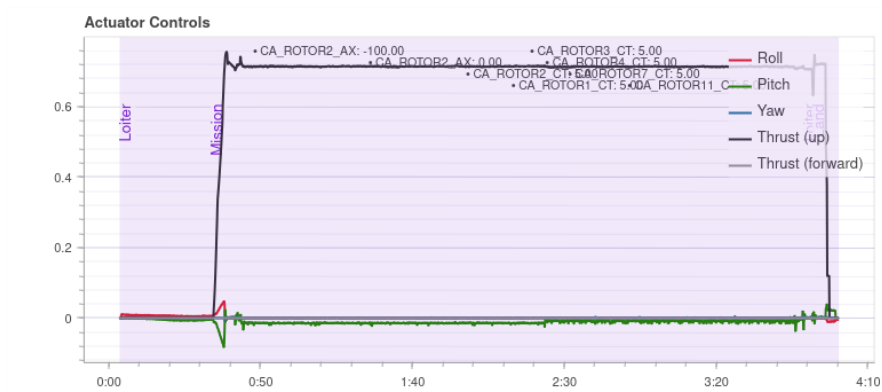


Volo con guasti: CA_ROTOR_CT

La modifica del parametro CA_ROTOR_CT gestisce il coefficiente di thrust, cioè di spinta del motore interessato, ogni motore del drone dispone infatti di questo parametro. La simulazione consisteva inizialmente di abbassare di circa il 10% il valore del parametro passando da un valore di default di 6,5 a 5 per un solo motore; per osservare se vi fossero eventuali cambiamenti di comportamento.

La simulazione, tuttavia, non ha riportato nessun cambiamento; si è dunque provato a valutare se il comportamento del parametro fosse stato implementato, abbassando drasticamente i valori di più motori a zero. Riportiamo di seguito i grafici della simulazione.

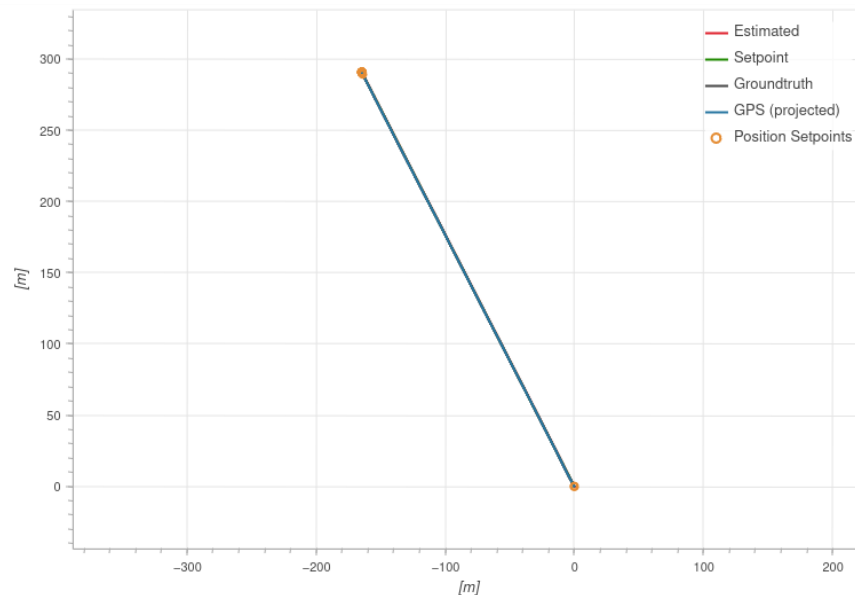




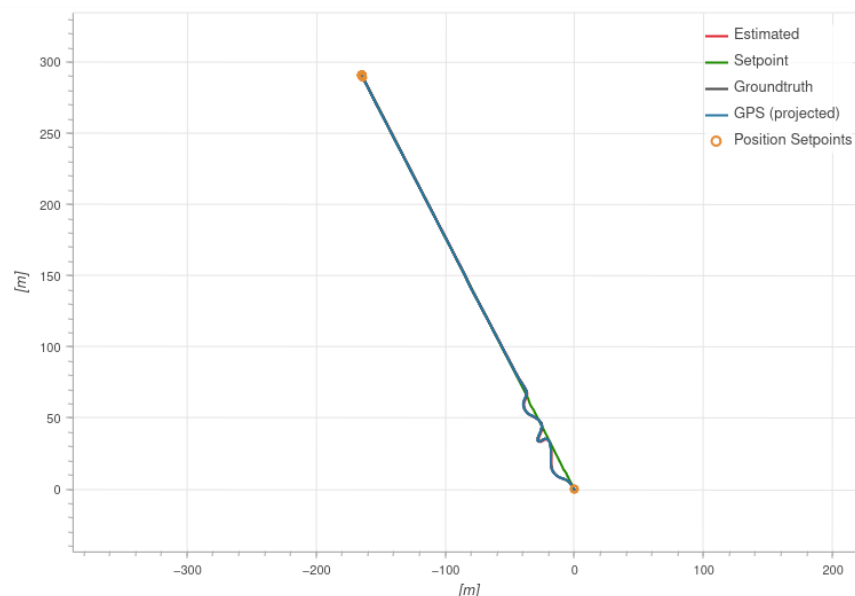
Si noti come non vi siano differenze di valori per thrust, valori di batteria e campo magnetico nonostante le numerose modifiche di parametro. Anche gli output dei motori non variano nonostante le modifiche, e non si discostano dalla situazione nominale, andando a provare nuovamente che il parametro, almeno modificato durante la simulazione, non abbia effetti. CA_ROTOR_AX modifica il thrust del motore interessato solo nell'asse direzionale orizzontale, non sortendo nessun effetto.

Volo con guasti: MC_ROLL_P

Il guasto in questione richiede una traiettoria per il quale il drone viri, e solo per questa occasione possiamo dunque la missione impostata ad hoc:



Il drone parte dalle coordinate (0,0) del grafico riportato (la posizione Setpoint in basso) rivolto verso Est, dovrà dunque ruotare per posizionarsi sulla traiettoria. Durante il suo posizionamento abbiamo dunque fatto partire il guasto: modificando il parametro MC_ROLL_P, che influisce sulla guida del veicolo controllando la quantità di correzione applicata al rollio per mantenere la traiettoria impostata, non riuscirà più a seguire il percorso predefinito. Il risultato che si ottiene è mostrato nella figura seguente, dove possiamo osservare la deviazione subita dal drone a causa del guasto rispetto alla missione impostata.



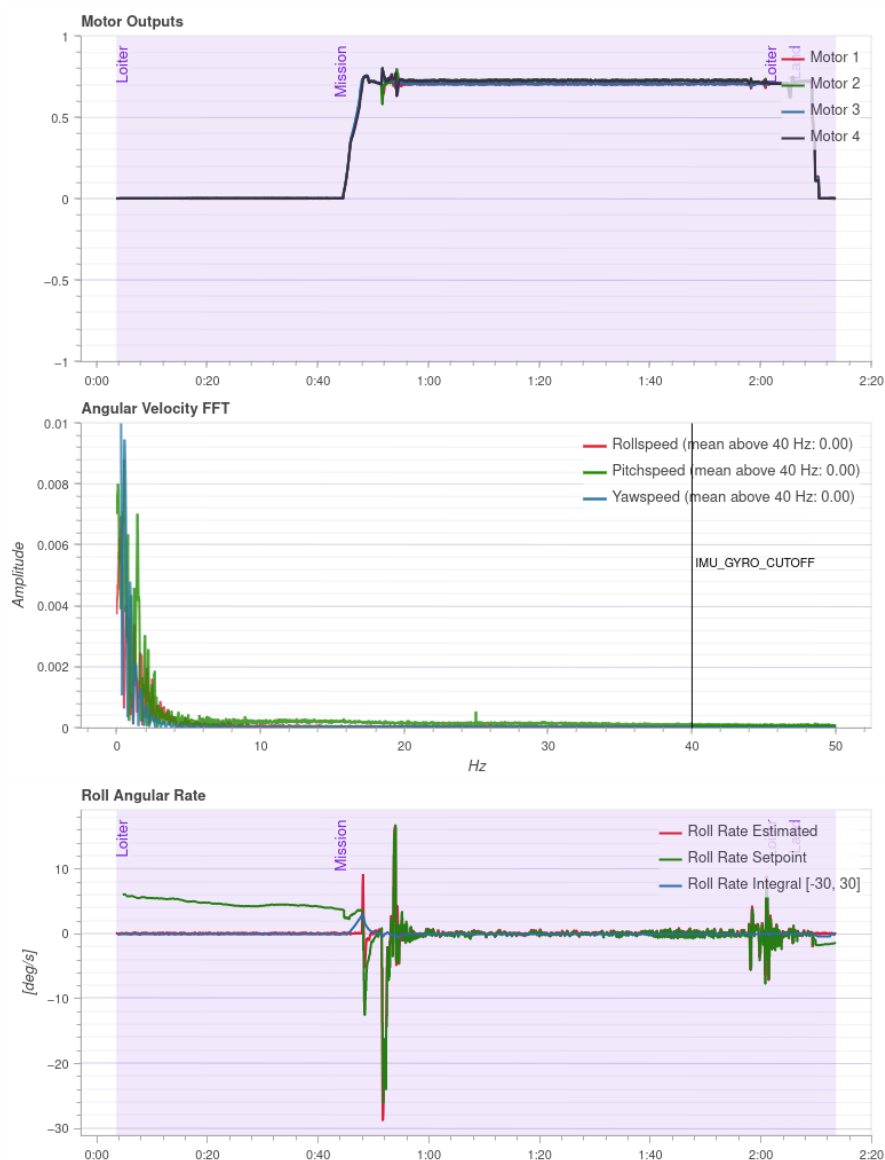
Il comando impostato all'inizio della simulazione sul prompt dei comandi (commander è stato):

param set MC_ROLL_P 0

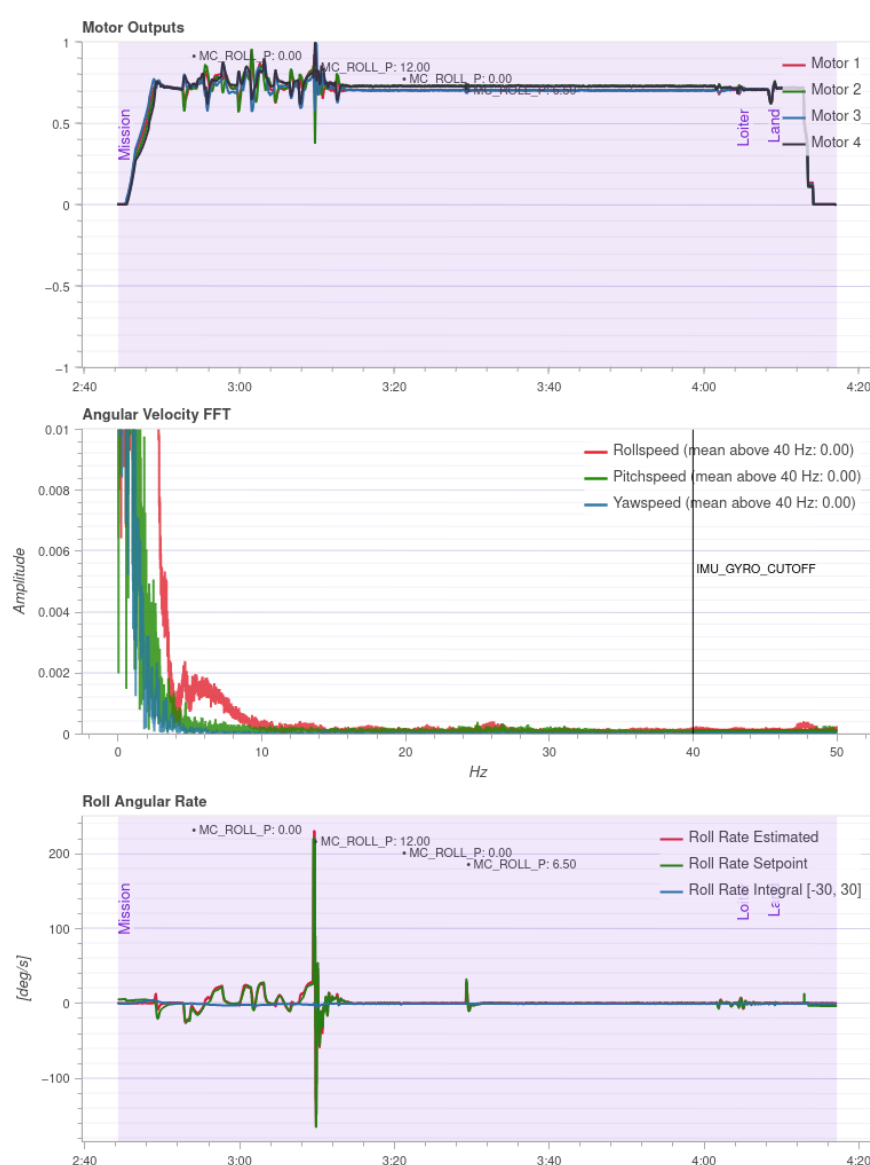
Il cui valore di default è 6,5.

Il comando causa, dunque, l'incapacità del drone di virare in maniera opportuna, riportando il parametro al valore di default il drone si riassetta immediatamente, continuando il percorso senza problemi ulteriori. Il problema avviene solo se il guasto avviene in fase di virata. Tutto questo per simulare l'incapacità dell'attuatore a fornire la giusta correzione di virata alla traiettoria.

Riportiamo un confronto dei grafici salienti confrontati con una simulazione senza guasti della stessa missione:



Riportiamo di seguito i grafici della simulazione con guasto:



Possiamo notare come i motori cerchino prima dell'assestamento, di riportarsi in una situazione di equilibrio, evidenziato dall'oscillazione degli output dei motori, e dell'analisi della velocità angolare, come possiamo notare un aumento dell'entropia almeno per le fasi iniziali del grafico, e di come ci si riconduce all'andamento della simulazione senza guasto, una volta riassetato il parametro.

Conclusioni

Px4, Gazebo e QGroundControl sono delle applicazioni che forniscono controllo su qualsiasi modello si desideri per simulare il loro comportamento, permettendo anche agli utenti di creare il proprio modello e di simularne il funzionamento sia tramite simulazioni SITL che HITL. Fornendo funzioni di monitoraggio e di log per valutare i parametri del modello simulato. Tuttavia le funzionalità di iniezione dei guasti forniti da Px4, non sono ancora stati implementati, restituendo come risultato ogni qualvolta si inietta un guasto il valore 1, che suggerisce che il comando sia stato rifiutato. Secondo i forum questo problema è stato discusso, con la risposta che semplicemente le iniezioni di guasti non sono ancora state implementate. Per simulare dunque comportamenti insoliti si è scelto di modificare parametri durante il volo, ottenendo risultati per lo più negativi, dove la maggior parte delle simulazioni le funzionalità del drone non sono mai risultate compromesse. Riteniamo possa essere in futuro un buon strumento intuitivo per simulare il funzionamento dei droni, e per testare la loro efficacia. Al momento può comunque essere utile per valutare il comportamento ideale.