POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Civile

Tesi di Laurea Magistrale

Veicoli autonomi terrestri per il rilievo automatico speditivo in aree critiche



T		
$\mathbf{R} \mathbf{\Phi}$	$\operatorname{lator}\epsilon$	•

Prof. Andrea Maria Lingua

Correlatori:

Ing. Vincenzo Di Pietra

Ing. Irene Aicardi

_____Luglio 2020

Laureando

Federico Faedda

"La geolocalizzazione è al centro dell'attuale rivoluzione digitale, con nuovi servizi che trasformano $la\ nostra\ vita\ quotidiana.$ Galileo migliorerà del 1000% la precisione della geolocalizzazione e sarà alla base della prossima generazione di tecnologie basate sulla localizzazione, come le automobili autonome, $i\ dispositivi\ connessi$ o i servizi urbani intelligenti. Oggi mi rivolgo agli imprenditori europei con questo invito: immaginate cosa potete fare con $Galileo.\ Non\ aspettate,\ innovate!"$

Maroš Šefčovič Vicepresidente della Commissione Europea per l'Unione dell'Energia

Ringraziamenti

Questo lavoro di tesi rappresenta la conclusione di un percorso complesso, giunto al termine dopo tanti sacrifici e con l'aiuto e il supporto di tante persone, a cui desidero esprimere una sincera gratitudine.

Un ringraziamento speciale va al Professor Andrea Maria Lingua, per avermi dato la possibilità di lavorare a un progetto stimolante e affascinante, e all'Ingegner Vincenzo Di Pietra, per avermi assistito e guidato in questo percorso. Entrambi sono stati di grande supporto nel periodo di sospensione della ricerca sperimentale in laboratorio, permettendomi di proseguire efficientemente il lavoro fuori sede.

Ringrazio l'Ingegner Irene Aicardi, per avermi guidato nell'impostazione globale del lavoro, e Paolo Maschio, per il supporto tecnico che ha richiesto questo progetto.

Ringrazio Gianluca Dara, per avermi accolto nel PIC4SeR e avermi aiutato quando si sono manifestate alcune problematiche di natura elettronica.

Ringrazio tutte le persone che lavorano nel laboratorio di Fotogrammetria, Geomatica e GIS, per avermi accolto in un ambiente armonioso e stimolante.

Un ringraziamento speciale va al Professor Pierluigi Claps, per avermi guidato durante il periodo di coordinamento del team studentesco Mi Lego al Territorio, una delle esperienze più caratterizzanti del mio percorso di laurea magistrale.

Ringrazio la mia famiglia e i miei amici, i miei maggiori sostenitori, con particolare gratitudine a mia madre: a lei dedico questo lavoro di tesi.

Indice

\mathbf{So}	mma	ario		7
Al	bstra	$\operatorname{\mathbf{ct}}$		8
In	$\operatorname{trod}_{\mathfrak{l}}$	ızione		9
1	Stat	o dell'	Arte	13
	1.1	Naviga	azione	14
	1.2	Uncre	wed Ground Veichle	15
		1.2.1	Nascita e sviluppo	16
		1.2.2	Utilizzo degli UGV	19
		1.2.3	Architettura del sistema	21
	1.3	Global	l Navigation Satellite System	25
	1.4	Costel	lazioni GNSS	26
	1.5	Archit	ettura dei sistemi GNSS	27
		1.5.1	Segmento spaziale	28
		1.5.2	Segmento di controllo	28
		1.5.3	Segmento utente	29
	1.6	Strutt	ura del segnale	29
	1.7	Errori	che interessano le misure GNSS	31
		1.7.1	Errori dell'orologio	32
		1.7.2	Errori dell'orbita	32
		1.7.3	Rifrazione atmosferica	34
		1.7.4	Centro di fase dell'antenna: offset e variazione	36
		1.7.5	Multipath	36
	1.8	Tecnic	che di posizionamento basate su segnali GNSS	37
		1.8.1	Posizionamento assoluto di codice	38
		1.8.2	Posizionamento assoluto di fase	40
		1.8.3	Posizionamento differenziale	42
	1.9	Correz	zioni differenziali da rete di stazioni permanenti	45
		1.9.1	Trasmissione delle correzioni via NTRIP	46

2	Ma	teriali e Metodi	48
	2.1	Scelta dei componenti e disegno sperimentale	48
		2.1.1 Veicolo - Traxxas X Maxx	49
		2.1.2 Autopilot Hardware	50
		2.1.3 Modulo GNSS a elevate prestazioni - Piksi Multi	53
		2.1.4 Ground Control Station - Mission Planner	55
	2.2	Interconnessione e protocolli di comunicazione	57
	2.3	Implementazione	62
		2.3.1 Schema di realizzazione	62
	2.4	Configurazione Pixhawk	65
		2.4.1 Collegamenti	65
		2.4.2 Configurazione tramite Mission Planner	69
	2.5	Configurazione Piksi Multi	73
		2.5.1 Cavo artigianale	73
		2.5.2 Configurazione UGV Piksi	74
		2.5.3 Configurazione Piksi stazione base	77
	2.6	Costo e caratteristiche del prototipo realizzato	80
3	Val	idazione del Prototipo	84
	3.1	Scenario di intervento	85
	3.2	Pianificazione della missione tramite Mission Planner	87
		3.2.1 Parametri di guida globali	87
		3.2.2 Disegno della traiettoria	90
	3.3	Ottimizzazione dei parametri di guida tramite simulazione	92
		3.3.1 Simulazione missione	93
		3.3.2 Trattamento dei dati per la comparazione	94
		3.3.3 Affinamento della velocità di sterzata	96
		3.3.4 Affinamento della traiettoria in curva	98
		3.3.5 Estensione della svolta	99
		3.3.6 Riduzione della velocità in curva	101
		3.3.7 Parametri definitivi	107
	3.4	Analisi delle pendenze	107
	3.5	Percorso per la chiesa di Sant'Agostino	109
	3.6	Collaudo con missione autonoma	112
4	Ris	ultati	116
	4.1	Risultati delle simulazioni	116
		4.1.1 Discussione	125
	4.2	Risultati delle missioni autonome	128
		4.2.1 Discussione	138

5	Conclusioni		
	5.1 Limiti e sviluppi futuri	. 142	
A	cronimi	144	
\mathbf{El}	lenco delle figure	148	
Εl	lenco delle tabelle	151	

Sommario

La tecnologia supporta sempre più intensamente l'operatività umana, sia nella vita quotidiana sia nelle attività produttive e professionali, portando grandi rivoluzioni e innovazioni. L'invenzione degli strumenti, la prima fase dell'evoluzione tecnologica, ha permesso di aumentare l'efficienza e la produttività del lavoro. La successiva nascita delle macchine ha sostituito lo sforzo umano, consentendo di superarne i limiti e incrementando notevolmente il rendimento del lavoro. L'automazione, l'ultimo stadio dell'evoluzione tecnologica, elimina il controllo continuo da parte dell'operatore umano. Gli algoritmi di esecuzione autonoma conferiscono alle macchine il massimo livello di produttività e la maggior precisione e affidabilità possibile.

Lo sviluppo della robotica mobile ha promosso la nascita e l'impiego dei sistemi a guida autonoma in molteplici ambiti di applicazione, estesi anche all'ingegneria civile. In questo settore, spesso si rende necessario svolgere campagne di rilievo speditivo, in aree critiche e in contesti pericolosi per l'incolumità degli operatori. L'implementazione della tecnologia di guida autonoma in un veicolo terrestre, in grado di trasportare la strumentazione di rilievo, navigando su un suolo avverso e pericoloso, costituisce un prodotto pratico, preciso, affidabile e altamente versatile. In questo lavoro di tesi è stato realizzato un prototipo di veicolo terrestre a guida autonoma (UGV), con navigazione basata sulle tecniche di posizionamento satellitare GNSS, capace di trasportare la strumentazione di rilievo in aree critiche.

L'elaborato illustra inizialmente lo Stato dell'Arte degli UGV e della tecnica del posizionamento satellitare, esponendo le basi teoriche propedeutiche alla realizzazione del progetto. Successivamente, si focalizza sulla procedura di implementazione, la quale ha richiesto la configurazione dei componenti hardware, la disposizione dei collegamenti elettrici, la creazione di un adeguato supporto meccanico e la simulazione del comportamento tramite apposito software, fase preliminare al collaudo.

 $\mathrm{L}'UGV$ realizzato è in grado di navigare autonomamente, in ambiente esterno, con elevata precisione rispetto alla traiettoria pianificata.

Abstract

Technology increasingly supports human operations, both in daily life and in production and professional activities, bringing great revolutions and innovations. Invention of tools, the first technological stage, has made it possible to increase work efficiency and productivity. The consecutive creation of the machines replaced human physical effort, allowing to overcome the limits and significantly increasing the work performance. Automation, the final stage of technological evolution, removes the continuous control by the human operator. The autonomous execution algorithms give to machines the highest level of productivity and the highest precision and reliability possible.

The development of mobile robotics has promoted the birth and use of autonomously guided systems in multiple areas of application, also extended to civil engineering. In this sector, often is needed to do expeditious survey campaigns in critical areas and dangerous contexts for relief workers. Implementation of autonomous driving technology in a ground vehicle, capable of carrying survey instruments, driving on adverse and dangerous soil, represent a practical, precise, reliable and highly versatile product. In this thesis, a prototype of a self-driving ground vehicle (UGV) was created, with navigation based on satellite positioning techniques GNSS, capable of transporting survey instruments in critical areas.

The document initially illustrates the *UGVs* and satellite positioning technique State of the Art, exposing the theoretical bases preparatory for project realization. Next, it focuses on the implementation procedure, which required the configuration of the hardware components, the arrangement of the electrical connections, the creation of an adequate mechanical support and the simulation of the behavior through specific software, preliminary to the testing.

The realized UGV can navigate autonomously, in an external environment, with high precision respect to the planned trajectory.

Introduzione

I droni¹ rappresentano uno degli strumenti più innovativi ed efficienti da poter utilizzare per le operazioni di rilievo. La rappresentazione del territorio è fondamentale per ogni intervento di progettazione, pianificazione e gestione del territorio, e negli interventi di soccorso. In emergenza c'è la necessità di conoscere l'ambiente circostante, rendendo necessario lo svolgimento di campagne di rilievo speditivo, in aree critiche e in contesti pericolosi per l'incolumità degli operatori. Conoscendo la geometria degli edifici, lo sviluppo delle infrastrutture e gli ostacoli presenti sul percorso, si possono sviluppare missioni e piani di intervento sicuri, affidabili e altamente accurati.

In questo contesto, gli *UGV*, *droni* terrestri caratterizzati dall'assenza del pilota umano a bordo, possono essere impiegati per svolgere rilievi, per misurare il livello di contaminanti nell'intorno di palazzi e sul terreno, possono fornire immagini e, in generale, intervenire in aree non accessibili dalle altre tipologie di *droni*, assistendo i soccorritori nelle procedure di evacuazione del personale[10]. Oltre a poter essere comandati da remoto, sono anche in grado di autopilotarsi, seguendo le istruzioni del software che li governa. La precisione della loro navigazione autonoma dipende da molti fattori, in funzione di essa possono essere impiegati per diverse finalità.

L'implementazione di una tecnologia di guida autonoma su un UGV di grandi dimensioni ed elevata potenza, in grado di trasportare strumentazioni di rilievo pesanti e ingombranti, procedendo e inoltrandosi su un suolo avverso e pericoloso, costituisce uno strumento nuovo, pratico e altamente versatile. Il principale obiettivo di questo progetto consiste nella realizzazione di un veicolo autonomo terrestre, avente tali caratteristiche, da utilizzare per missioni in ambiente esterno, i cui costi di realizzazione e manutenzione siano contenuti.

Per raggiungere l'obiettivo, sono stati ricercati e utilizzati hardware e software liberi (*open source*), i quali hanno permesso di contenere significativamente i costi. I principali componenti da individuare sono:

¹Si definisce *drone* un qualsiasi veicolo pilotabile da remoto.

- Veicolo
- AutoPilota
- Tecnica di navigazione autonoma

Il prototipo è stato sviluppato sulla base di un veicolo radiocomandato da modellismo, il Traxxas X Maxx, opportunamente privato della propria carenatura ed equipaggiato con un supporto appositamente progettato. L'X Maxx è un veicolo fuoristrada dotato un motore brushless molto potente, per la categoria, il cui controllore elettronico della velocità (ESC) gestisce batterie agli ioni di litio (LiPo) da 4 a 8 celle. Queste caratteristiche lo rendono adatto al trasporto di grossi pesi su percorsi difficili, anche di grande estensione. Il moto viene ripartito sulle quattro ruote con un sistema di trazione integrale, ideale per affrontare suoli irregolari, utilizzando il singolo motore brushless, e la sterzata è gestita da un servomotore: la guida si realizza gestendo input e output tramite due semplici connettori a 3 pin.

Il controllo dei motori è stato affidato all'AutoPilota *Pixhawk*, sul quale è stato installato il software *ArduPilot*, mentre per la configurazione del sistema e la pianificazione delle missioni di guida autonoma è stato utilizzato il software *Mission Planner*. Sono tutti prodotti realizzati dalla comunità *open source* e sono capaci di controllare più tipologie di veicoli. Al controllore *Pixhawk* sono poi connesse diverse periferiche necessarie per il controllo del veicolo, sia manuale sia autonomo.

Tra le possibili tecnologie con le quali poter realizzare la guida autonoma, la tecnica del posizionamento satellitare GNSS, adottata in questo progetto, costituisce la soluzione di minor onere, sia economico sia di integrazione nel sistema, per l'utilizzo in ambiente esterno. Attualmente è possibile acquistare ricevitori GNSS ad alte prestazioni, predisposti per la comunicazione con altri sistemi open, a prezzi accessibili. Il ricevitore ad alte prestazioni individuato per questo progetto è il Piksi Multi, prodotto dall'azienda Swift Navigation: è un modulo a basso costo, consente di raggiungere la precisione centimetrica tramite la tecnica di posizionamento RTK ed è stato concepito per una rapida integrazione nei sistemi di guida autonoma. Il veicolo realizzato non è in grado di navigare in ambienti chiusi, più in generale, dove non è disponibile in segnale GNSS. La scelta di una tecnica di guida autonoma funzionante solo in ambiente esterno è strettamente correlata alla natura del progetto. Per realizzare la navigazione autonoma in ambienti chiusi è necessario utilizzare tecniche più sofisticate, le quali comportano un considerevole aumento dei costi e della difficoltà di implementazione, senza apportare miglioramenti tali, alla guida esterna, da giustificarne la scelta. Pertanto, la guida indoor può essere solo manuale, costituendo un limite stabilito dal principio.

Il presente documento è organizzato in 5 capitoli, segue una breve rassegna sul loro contenuto.

- **Stato dell'Arte** Sono esposte le conoscenze odierne sul settore degli *UGV*, includendo l'analisi della loro nascita e evoluzione, e sulla tecnica del posizionamento satellitare.
- Materiali e metodi Sono riportati i materiali e i metodi utilizzati per la realizzazione del prototipo, illustrando: la filosofia con cui sono stati scelti i singoli componenti, le operazioni necessarie per consentirne la corretta comunicazione con il sistema e l'intera procedura di implementazione.
- Validazione del prototipo Sono descritti i procedimenti seguiti per la simulazione del comportamento del veicolo e la successiva fase di collaudo. Sono stati analizzati i dati relativi alle traiettorie percorse dall'UGV, durante lo stadio di simulazione per comprendere come completare la configurazione del sistema, e durante la fase di collaudo per studiarne il comportamento reale.
- Risultati Sono riportate le evidenze delle analisi svolte nel capitolo precedente e la loro interpretazione. In particolare: sono analizzati gli scostamenti fra le traiettorie percorse, sia simulate sia reali, e la traiettoria di riferimento pianificata.
- Conclusioni Insieme alle conclusioni, si discutono i limiti del progetto, con uno sguardo ai possibili sviluppi futuri.

Ridefinizione degli obiettivi in corso d'opera

Il presente lavoro di tesi è stato iniziato nell'ottobre 2019 e concluso nel luglio 2020. Il 23 febbraio 2020, a causa dell'emergenza *COVID-19*, il Rettore del Politecnico di Torino ha stabilito la sospensione delle attività di ricerca sperimentale, sino a data da definire. Questo improvviso arresto della sperimentazione in laboratorio ha comportato una riorganizzazione del lavoro di tesi, in relazione alla quale:

- il capitolo dello Stato dell'Arte è stato approfondito, arricchendo l'analisi storica degli *UGV* e le tecniche di posizionamento GNSS;
- il capitolo di materiali e metodi è stato ampliato, raggiungendo un grado di elevato dettaglio nell'illustrazione delle procedure di configurazione;
- nel capitolo di validazione del prototipo è stata inclusa una corposa parte di simulazione, svolta fuori sede, inizialmente prevista in misura più ridotta.

Nel mese di giugno, seguendo un piano di riammissione graduale, è stato concesso l'ingresso di un numero limitato di tesisti nei locali del Politecnico,. Nel giorno 17 giugno 2020 è stato possibile riprendere la sperimentazione in laboratorio. Lo studio dettagliato affrontato nei mesi di lavoro fuori sede ha permesso di assemblare i componenti con rapidità e precisione tali da consentire di ottenere il controllo del veicolo tramite *Pixhawk* nella stessa giornata. Nella giornata successiva è stato completato il supporto meccanico, provvisoriamente realizzato con mattoncini Lego², e sono stati svolti i primi test di guida autonoma. L'approfondita fase di simulazione ha permesso di ottenere un ottimo rispetto della traiettoria pianificata dalla prima missione, pertanto non è stato necessario modificare i parametri di guida per migliorarne il comportamento. Nella settimana successiva sono stati svolti gli ultimi test di guida autonoma, riportati nel capitolo di validazione del prototipo, e fondamentali per stabilire le conclusioni del progetto.

² "Nella mia esperienza universitaria sono stato coordinatore del team studentesco Mi Lego al Territorio. Non avendo accesso e non essendo autorizzato all'utilizzo e la lavorazione dei materiali necessari alla fabbricazione di un adeguato supporto, e non potendo attendere altro tempo, vista la vicinanza della data di discussione (23 luglio) e la determinazione nel voler collaudare il prototipo, ho pensato di utilizzare i mattoncini Lego, gentilmente concessi in prestito dal team. In questo modo, ho potuto includere nella mia tesi una delle esperienze più significative e caratterizzanti del mio percorso formativo al Politecnico di Torino, e una delle passioni che mi accompagnano da sempre, sicuramente tra i responsabili della mia ambizione a voler diventare Ingegnere. E ho potuto fare anche un particolare omaggio al progetto ArduPilot, il quale nacque con la creazione di un drone realizzato con mattoncini e controllore (Mindstorm) Lego[2][3][38]."

Capitolo 1

Stato dell'Arte

A seguito di un evento naturale distruttivo, al fine di poter pianificare le attività di immediato intervento, è necessario svolgere delle operazioni di rilievo, volte a determinare la dimensione, l'estensione e la localizzazione dei danneggiamenti. La regione interessata dal fenomeno, caratterizzata da ambienti inaccessibili e ostili per l'uomo, è definita area critica.

Le tecniche di rilievo speditivo¹ adottate in queste situazioni sono molteplici e dipendono dal tipo di dato da acquisire.

Le applicazioni più comuni prevedono l'acquisizione di immagini, sia bidimensionali sia tridimensionali, e il rilievo di punti con ricevitori GNSS. In generale, le immagini possono essere acquisite impiegando diversi strumenti, a partire dalle classiche macchine fotografiche o videocamere, per arrivare ai laser scanner e agli algoritmi di $SLAM^2$ nelle applicazioni più avanzate.

Quando si opera in aree critiche è importante disporre di mezzi che consentano l'esecuzione dei rilievi salvaguardando l'incolumità degli operatori.

A tal fine, nella pratica odierna sono di largo impiego alcuni particolari veicoli, denominati UGV - Uncrewed Ground Veichle e UAV - Uncrewed Aerial Veichle: sono robot mobili (noti anche come droni), terrestri e aerei, capaci di autopilotarsi. Possono trasportare gli strumenti di misura ed eseguire i rilievi, arrivando in posti non raggiungibili dall'essere umano, evitando l'esposizione al rischio e fornendo anche diverse prospettive di osservazione, rispetto alle tradizionali metodologie.

¹Con il termine *speditivo* si individua una metodologia per cui il rilievo è eseguito in un breve lasso di tempo. Se in origine la rapidità delle operazioni comprometteva la precisione delle misure, l'odierna tecnologia permette di svolgere le operazioni speditive con elevata precisione.

²Per SLAM (*Simultaneous Localization and Mapping*) si intende il processo per cui un robot si muove in un ambiente sconosciuto, costruisce una mappa tridimensionale di tale ambiente e la impiega per localizzarsi e navigare all'interno di detta mappa.

La capacità dei robot mobili di muoversi e svolgere le operazioni di rilievo in completa autonomia rende il loro impiego sempre più diffuso nelle aree critiche.

L'interesse maggiore, infatti, riguarda la capacità di navigare autonomamente. Per essere autonomo e completare con successo un compito, il robot deve essere in grado di comprendere l'ambiente che lo circonda, facendo fronte ai cambiamenti delle condizioni ambientali nelle quali deve operare, quindi adattandosi agli ambienti dinamici. E per riuscirci, deve avere la capacità di autogoverno: il corso delle sue azioni deve essere determinato mediante un processo di ragionamento [24].

1.1 Navigazione

Per un agente mobile la capacità di navigare è una delle abilità più importanti [24]. E nelle applicazioni di rilievo in ambito emergenziale questa abilità assume un ruolo centrale, costituendo il fattore critico essenziale per l'esito della missione. La navigazione autonoma è governata da tre processi fondamentali:

- localizzazione: è necessario individuare e definire la propria posizione all'interno di un determinato sistema di riferimento;
- pianificazione del percorso: nota la posizione del robot, deve essere definito un percorso per raggiungere i punti desiderati;
- costruzione, interpretazione e utilizzo della mappa: il robot può pianificare il percorso sulla base di una mappa preesistente o utilizzare una mappa generata durante la navigazione, in ogni caso è richiesta la capacità di interpretazione della mappa.

L'elemento base di ogni sistema di navigazione robotica è il sistema di riferimento. Immaginando di basare la navigazione sul piano cartesiano, il robot potrebbe calcolare la propria posizione tramite le misurazioni interne ricavate dal proprio movimento (odometria): è il principio della propriocezione. Ma non è possibile basare la navigazione sulla sola odometria in quanto la struttura di riferimento sarebbe direttamente dipendente dalla navigazione e subirebbe continue alterazioni.

Durante la guida, il mezzo è soggetto a perturbazioni, le quali si traducono in errori di percorso per un sistema così concepito: si immagini il solo fenomeno di slittamento delle ruote per cui i sensori traducono la rotazione delle ruote in una distanza percorsa, ignorando che il robot invece è rimasto nella stessa posizione.

Una soluzione al problema consiste nella determinazione del percorso mediante l'identificazione di univoci *punti di riferimento*, i quali possono essere usati per pianificare un percorso o per determinare l'orientamento del robot rispetto alla mappa.

In questo caso, la navigazione è basata sulla *esterocezione*, ossia la percezione dell'ambiente da parte dell'agente[24].

Per identificare i punti di riferimento esistono molteplici tecniche. In generale, si distinguono due metodologie principali sulle quali poi si basa la navigazione stessa:

Ricostruzione dell'ambiente in 3 dimensioni Questa tecnica è basata sulle caratteristiche fisiche dell'ambiente circostante, le quali vengono rilevate mediante sensori LIDAR, tecniche di Range Imaging e Vision Recognition, sensori fotoelettrici per il riconoscimento dei colori e algoritmi di SLAM[28]. I punti di riferimento (fisici) sono costituiti dalle forme rilevate, e la tecnica si adatta alla navigazione sia all'interno di edifici sia esterna. Gli strumenti richiedono molta potenza di calcolo, hanno un costo elevato e la percezione dell'ambiente circostante è comunque affetta da ambiguità percettive[24].

Utilizzo di punti geolocalizzati Con questa tecnica, la navigazione è basata sull'acquisizione di segnali GNSS, di conseguenza può essere utilizzata solo in
ambiente esterno. I punti di riferimento (virtuali) sono identificati in un sistema di riferimento indipendente dal moto, e i ricevitori GNSS possono arrivare all'accuratezza del centimetro, permettendo una navigazione molto precisa
con una spesa contenuta: anche i più performanti sono attualmente disponibili a costi contenuti, non richiedono grandi risorse hardware e possono essere
facilmente integrati in altri sistemi.

Nella navigazione esterna, la tecnica basata su segnali GNSS è attualmente di grande interesse e impiego. La versatilità, l'elevata precisione, il basso costo, la rapidità nell'implementazione e la praticità nell'analisi dei dati sono caratteristiche che fanno spesso preferire questa soluzione per la navigazione, trovando grande applicazione nel campo non solo degli UGV, ma dei droni in generale.

1.2 Uncrewed Ground Veichle

Con il termine *UGV* - *Uncrewed Ground Veichle* (noto anche come *Rover*) si definisce un veicolo terrestre caratterizzato dall'assenza del pilota umano a bordo. In questa descrizione non viene specificato il tipo di controllo da parte dell'operatore.

In origine era possibile il solo controllo manuale, mentre attualmente possono navigare in totale autonomia, il raggiungimento di questo ultimo stadio è stato ottenuto con un processo di evoluzione graduale, in cui questi robot mobili hanno acquisito sempre più autonomia.

Livello di automazione

Sheridan[34] classificò le modalità di controllo in:

- manuale: le azioni del veicolo vengono decise interamente dall'uomo;
- controllo di supervisione: l'operatore umano programma il robot con intermittenza, ricevendo continuamente informazioni da un hardware interconnesso con il veicolo, capace di prendere decisioni autonome;
- *automatico*: l'apparato è in grado di prendere autonomamente qualsiasi tipo di decisione, sostituendo il controllo di supervisione umano.

In senso più ampio, si può definire UGV una qualsiasi apparecchiatura meccanizzata in grado di muoversi sulla superficie del terreno e trasportare qualcosa, non necessariamente un essere umano [13].

Architettura del sistema

Il funzionamento dei *Rover* viene governato dall'interazione di:

- *Hardware*: l'insieme di sensori, controllore e dispositivi di uscita che fanno muovere il veicolo;
- Software autopilota: il codice in esecuzione sul controllore, da cui vengono derivati i firmware appositamente progettati in funzione della tipologia di veicolo (Rover, firmware, per un UGV realizzato tramite ArduPilot, software);
- Software di controllo: chiamato Ground Control Station GCS, rappresenta l'interfaccia tramite la quale il computer comunica con il controllore, sia per la configurazione, sia per il monitoraggio della missione;
- Meccanica: l'insieme degli organi di movimento, che permettono la navigazione dell'UGV, e delle strutture di sostegno, che ospitano hardware e meccanica.

1.2.1 Nascita e sviluppo

La prima testimonianza di un UGV risale al 1921: si tratta di un'automobile corazzata controllata da remoto, concepita per proteggere la persona trasportata [46]. La nascita di questa tipologia di veicoli è avvenuta in ambito militare. Si osservi come nello stesso articolo citato [46] gli autori scrissero: Questa dimostrazione illustra la possibile applicazione di questo metodo in tempo di guerra.



Figura 1.1: Prima dimostrazione di un'automobile radiocomandata - UGV [46].

Difatti, il primo veicolo radiocomandato in assoluto comparve durante la prima guerra mondiale, nel 1916: l'Aerial Target, un UAV realizzato dalla RAF - Royal Aircraft Factory³, sviluppato sulla base dell'idea di un capitano della RFC - Royal Flying Corp, forza aerea britannica durante la prima guerra mondiale[37]. L'obiettivo era la realizzazione di un veicolo capace di mettere fuori uso i dirigibili degli avversari senza esporre i soldati.

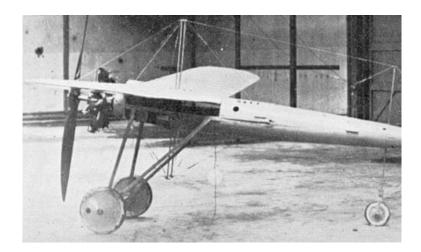


Figura 1.2: Primo veicolo radiocomandato - UAV[37].

 $^{^3}$ Meglio nota con il suo successivo nome, RAE - Royal Aircraft Establishment, si tratta dell'istituto di ricerca britannico fondato dal Ministero della difesa britannico. Da non confondersi con la RAF - Royal Air Force, istituita nel 1918.

Sin dalla loro nascita, i *droni* hanno sempre avuto lo scopo di salvaguardare l'incolumità dell'essere umano, utilizzando il mezzo come suo diretto sostituto.

La Robotica mobile

Se in origine gli UGV sono stati concepiti per l'uso militare, i successivi sviluppi nell'ambito della ricerca scientifica hanno prodotto moltissime applicazioni civili. Vi fu un particolare momento di svolta per il settore nel 1956, alla Dartmouth College Conference, quando fu coniato il temine $intelligenza \ artificiale^4$, riferito all'abilità dei robot mobili di svolgere particolari mansioni.

Fu una conferenza di grande ispirazione per la comunità scientifica, in seguito alla quale nacquero molti progetti e realizzazioni di robot mobili. Il primo esemplare capace di ragionare sulle proprie azioni fu Shakey The Robot[26], sviluppato da Nils Nilsson, nel 1969, a Stanford. Negli anni settanta furono realizzati molti altri modelli, tra cui il primo Rover NASA, progettato per le esplorazioni spaziali[24]. In questa prima fase di sperimentazione sulla robotica mobile, i ricercatori lavorarono molto sui software dei robot, tralasciando lo sviluppo hardware. Questa mancanza comportò notevoli problematiche al momento della realizzazione, che diventava la fase di maggiore difficoltà nell'intero progetto. La successiva evoluzione dell'industria tecnologica ha introdotto hardware sempre più potenti, costituendo un supporto adeguato per lo sviluppo degli UGV.

Se in passato gli UGV sono sempre stati progettati interamente come robot mobili, attualmente sono diffusi i modelli realizzati sulla base di veicoli radiocomandati esistenti pronti all'uso (Ready to Run - RTR RC Cars), ai quali si implementano un controllore dei servomotori e le periferiche necessarie alla navigazione. Questa innovazione ha reso molto più accessibile il loro impiego e sviluppo, senza

dover disporre di grandi budget e strumentazioni sofisticate. E rende molto versatile il loro utilizzo, in quanto i componenti danneggiati durante l'utilizzo possono essere reperiti velocemente e a costo contenuto, conferendo elevata riparabilità del sistema. Diversi danni possono essere riparati direttamente sul campo, permettendo di non interrompere una missione. Possono sempre essere pilotati da remoto, tramite radiocomandi, ma l'applicazione più interessante riguarda la guida autonoma: sono in grado di determinare la propria posizione e calcolare il proprio percorso con elevata precisione, utilizzando i sensori di bordo e le proprie risorse di elaborazione.

⁴Il termine *intelligenza artificiale* è relativo alla personale interpretazione delle azioni intraprese dal robot, è un giudizio soggettivo[24]: il Matematico Alan Turing sottolineò come la definizione di intelligenza dipenda direttamente dalle proprie conoscenze[43].

1.2.2 Utilizzo degli UGV

I progressi della ricerca scientifica hanno permesso un grande sviluppo degli UGV. I primi esperimenti erano fondamentalmente robot da laboratorio, molto ingombranti, poco precisi e capaci di svolgere poche e specifiche operazioni semplici. Con le tecniche odierne possono avere dimensioni molto contenute, navigare con elevata precisione e svolgere operazioni, anche complesse, in ambienti sconosciuti. Di conseguenza il loro impiego è stato esteso a molteplici ambiti e diversi scopi[14].

In Europa, la crescita del mercato UGV è stata ampiamente supportata dai Framework Program (FP) istituiti negli anni 2000. In particolare, i programmi FP6 e FP7 hanno promosso e finanziato la ricerca tecnologica, con particolare concentrazione sullo sviluppo di ricevitori GNSS Galileo, ad alta precisione e basso costo, e programmi di ricerca sulla collaborazione tra robot. Molte applicazioni innovative dei veicoli autonomi sono nate dallo sviluppo di questi progetti.

Il loro impiego in ambito militare è ovviamente cresciuto nel tempo, supportando operazioni di sorveglianza, ricognizione e acquisizione di bersagli[14]. Hanno poi rivoluzionato il settore agricolo: la necessità di risparmiare risorse e manodopera ha promosso la creazione di veicoli autonomi estremamente efficienti. Parallelamente, questi UGV acquisiscono molte informazioni aggiuntive, utilizzabili dal gestore per aumentare la produzione e gestire più efficacemente le risorse.

Con la stessa filosofia, sono utilizzati anche per le attività minerarie, come mezzi da cantiere per il trasporto di materiale, per mappare il livello di contaminanti all'interno di edifici o nel terreno e, in generale, per fornire immagini ottenibili esclusivamente con il loro impiego[10].

Lo scopo predominante degli UGV resta la sostituzione dell'uomo per le operazioni pericolose, ma non più nel solo ambito militare: il loro impiego nelle operazioni di ricerca e soccorso (SAR) sta diventando sempre più centrale, estendendosi anche ad altre applicazioni, pericolose per l'uomo, in ambiti non emergenziali.

Contesti pericolosi

Gli UGV, utilizzati nelle situazioni di pericolo, hanno permesso di salvare molte vite e, in generale, evitare tanti possibili incidenti. Il loro impiego nelle operazioni di ricerca e soccorso (SAR) non è comunque immediato: il contesto è ostile all'impiego diretto della tecnologia, richiede pertanto soluzioni robuste che possano essere impiegate rapidamente. Da ogni esperienza, le squadre di ricercatori hanno potuto osservare problematiche e punti di forza, permettendo, negli anni, l'evoluzione dei robot mobili, delle loro modalità operative e delle loro procedure di impiego.

Primi casi Un'importante testimonianza fu l'impiego dei rover lunari⁵, opportunamente adattati, nel disastro di Chernobyl. Dovevano essere rimossi dei frammenti di grafite radioattiva dal tetto della centrale. Un lavoro non eseguibile da un essere umano, invece adatto all'utilizzo di UGV, in quel caso comandati da remoto. Altrettanto rilevante fu il loro impiego nelle operazioni di soccorso dell'attentato alle torri gemelle del 11 settembre 2001: gli UGV strisciarono attraverso spazi estremamente piccoli, su cataste di macerie instabili, per raggiungere le vittime. L'utilizzo dei droni può ridurre l'esposizione dei soccorritori a pericoli evitabili [10].

In Italia La prima comparsa degli UGV a supporto delle operazioni di soccorso avvenne per il terremoto dell'Emilia Romagna (2012), nell'ambito del progetto NIFTi (2010-2013)[25]. Il programma è stato fondato per sviluppare metodologie volte a migliorare la collaborazione tra operatore umano e il robot di terra[11]. Nell'intervento italiano, UGV e UAV, comandati manualmente dai piloti a terra, sono stati utilizzati nel Comune di Mirandola per eseguire scansioni tridimensionali della Chiesa di San Francesco e del Duomo, al fine di valutare l'entità dei danni e delle lesioni[21]. Dopo questa esperienza, gli UGV vennero impiegati anche a seguito del terremoto di Amatrice (2016), nell'ambito del progetto TRADR (2013-2017)[44].

Il programma, costruito sull'esperienza del progetto NIFTi, aveva l'obiettivo di creare modelli ambientali per aiutare gli operatori a comprendere come lavorare nell'area danneggiata [11]. Anche in questo caso sono stati impiegati gli UGV, aiutati nelle operazioni da alcuni UAV, entrambi pilotati manualmente per ottenere il rilievo tridimensionale delle chiese di San Francesco e Sant'Agostino [22].

Prospettive di crescita del mercato UGV Il mercato globale degli UGV si stima debba raggiungere i 7.45 miliardi di dollari entro la fine del 2026[12], con un tasso annuo di crescita composto (CAGR - Compound Annual Grouth Rate) del 14.95%[20]. La presenza, in tutto il mondo, di numerose aziende produttrici di UGV su larga scala costituisce un enorme potenziale per la crescita del settore nel prossimo futuro. L'impiego diffuso degli UGV in molteplici settori industriali comporta un avvicinamento da parte dei governi di diversi Paesi del mondo, i quali sono sempre più interessati a commissionare UGV con requisiti personalizzati[12]. La quota di mercato maggioritaria si stima sarà rappresentata dagli Stati Uniti d'America, in cui le applicazioni nel settore della difesa continueranno ad attrarre molti investitori[20].

 $^{^5}$ Furono usati diversi UGV nel disastro di Chernobyl, i modelli lunari ebbero maggior successo rispetto agli altri in quanto progettati per resistere alle radiazioni cosmiche.

1.2.3 Architettura del sistema

Come esposto nei paragrafi precedenti, il funzionamento degli UGV è garantito dall'interazione di diversi componenti, i quali costituiscono l'architettura del sistema, che varia in funzione della complessità del veicolo.

Questi veicoli autonomi possono essere realizzati integralmente ex novo, affrontando la progettazione di ogni singolo componente: un approccio molto complesso, per il quale sono richieste competenze avanzate di robotica e meccatronica, unitamente a importanti risorse economiche e un tempo di sviluppo e realizzazione esteso.

Un approccio meno complicato prevede l'utilizzo delle automobili RTR RC, alle quali viene opportunamente implementata una componentistica hardware in grado di gestire la navigazione autonoma.

Lo sviluppo di un UGV dunque si basa inizialmente su una valutazione economica del tipo costi-benefici. Nell'ambito di questa analisi, si considera il costo totale dell'UGV, espresso dalla somma di due fattori principali[5]:

- Costo di sviluppo e realizzazione: dipende direttamente dal livello di autonomia che si vuole ottenere: più automatismi vengono sviluppati, più il costo sale, in maniera esponenziale[5]: le applicazioni più avanzate hanno un costo molto elevato, sia di realizzazione, sia di progettazione.
- Costo di utilizzo: il life-cycle operations and support O & S diminuisce quasi linearmente all'aumentare del livello di autonomia: gli automatismi aumentano l'efficienza del veicolo durante le missioni, diminuendo i costi operativi e aumentando la produttività.

Il costo ottimale, di conseguenza, corrisponde al giusto compromesso tra i due fattori, come riportato in figura 1.3. Nel caso della navigazione in esterna, non si ottiene da una progettazione integrale, ma dall'adattamento di un'automobile RTR RC.

Hardware

La creazione di un'automobile $RTR\ RC$ autonoma richiede diversi componenti hardware:

• controllore di volo: noto anche come *flight controller*, si tratta di un controllore di servomotori, generalmente realizzato sulla base di progetti liberi⁶, il quale, utilizzando gli input di tutti i componenti hardware, è in grado di decidere autonomamente le manovre da eseguire, inviando output a dispositivi come ESC (Electronic Speed Control) e servomotori;

⁶Il termine *libero* si riferisce alle tecnologie *open-source*: trattasi di prodotti la cui licenza ne consente la modifica, lo studio, l'utilizzo e la condivisione.

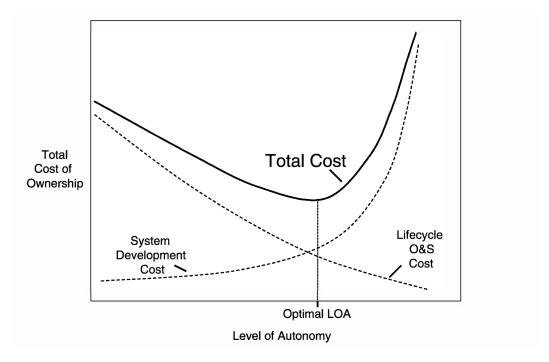


Figura 1.3: Costo totale di un UGV [5]

- ricevitore GNSS: è la periferica responsabile dei dati di navigazione, capace di calcolare la propria posizione con elevata precisione e avente un costo contenuto;
- **periferiche**: per realizzare un sistema efficiente, con un adeguato livello di autonomia, è necessario disporre anche di elementi quali segnalatori acustici (buzzer), interrutori switch per armare e disarmare i motori, due radio di telemetria per la comunicazione in tempo reale, dei sensori sonar per l'aggiramento degli ostacoli, batterie per l'alimentazione del veicolo e una coppia trasmittente e ricevente per il controllo manuale remoto⁷.

Software AutoPilota

Nel mercato odierno, sono presenti numerosi controllori di volo. Per realizzare un veicolo a guida autonoma, la prima variabile da definire è l'individuazione dell'*Autopilota*: è il software che gestisce la guida, di conseguenza è la componente più importante di qualsiasi veicolo autonomo.

⁷Il controllo manuale deve essere sempre garantito, sia per consentire la mobilità del Rover all'infuori della missione, sia per consentire il pronto intervento in caso di comportamenti anomali durante la guida autonoma, assumendo particolare importanza nella fase di collaudo.

Per gli *UGV* derivati da *RTR RC Car*, la soluzione più diffusa è *Ardupilot:* il software autopilota open source più avanzato, completo e affidabile, attualmente disponibile [4]. È stato sviluppato in oltre 5 anni da un team di ingegneri professionisti e scienziati informatici. Installato in oltre 1.000.000 di veicoli in tutto il mondo [4], la possibilità di realizzare un sistema a guida autonoma tramite Ardupilot ha permesso a una moltitudine di utenti amatoriali di sviluppare i propri progetti e condividerli con la comunità. Di conseguenza, molti fornitori esterni hanno creato prodotti compatibili col sistema, che beneficia di ampia compatibilità con sensori, computer associati e sistemi di comunicazione.

Controllore di volo

ArduPilot può essere eseguito su diverse AutoPilot Board.

Il criterio di scelta dipende dalle caratteristiche del controllore e dalle specifiche esigenze, progettuali e operative. Alcuni possiedono un'ottima sensoristica integrata e consentono la connessione di molte periferiche: in questo modo il sistema può disporre di sensori ridondanti, i quali rendono la guida più sicura. Per questa tipologia, il più impiegato è il *Pixhawk*. Altri sono stati progettati con particolare riguardo all'isolamento dalle vibrazioni, come *CUBE*, *CUAV v5* e *Holybro Kakute F7*. In caso di piccoli telai, una soluzione idonea si trova nel *Pixracer*. Infine, per veicoli basati sulla navigazione per immagini, è possibile utilizzare *Emlid NAVIO 2*, un controllore basato su codice Linux.



Figura 1.4: Controllori di volo compatibili con ArduPilot

Firmware

Il firmware contiene tutte le istruzioni necessarie per la guida autonoma di uno specifico veicolo. Il controllore pilota l'UGV inviando segnali di output calcolati sulle istruzioni del firmware: ogni categoria di veicolo ha le proprie regole di guida e deve seguire le istruzioni del rispettivo firmware.

Il firmware sviluppato per gli UGV è ArduRover, la cui ultima versione attualmente disponbile⁸ è la 4.0.0.

Per ogni tipologia vengono fornite 4 versioni di codice binario:

- *stable*: la versione stabile più recente, consigliata per un utilizzo comune e per utenti non esperti;
- beta: dedicata agli utenti che intendono collaborare allo sviluppo, contiene nuove funzionalità testate ma ancora instabili;
- *latest*: la versione più recente, contenente gli ultimi aggiornamenti sui quali sono stati svolti pochissimi test (non può essere considerata quindi una beta);
- build storiche: sono le versioni stabili precedenti, possono risultare utili per ovviare a problematiche mai insorte.

Struttura Il firmware viene reso disponibile, da ArduPilot, in 4 diversi formati file:

- *.apj: sono i firmware ArduPilot JSON, contengono il firmware caricabile dalla GSC;
- *.px4: è la precedente denominazione dei file *.apj, utilizza lo stesso formato;
- *.hexfile: sono i firmware scritti nel formato esadecimale Intel, realizzati per il caricamento *Direct Firmware Update DFU* in sistemi basati su architettura STM32, generalmente impiegati quando il bootloader è danneggiato;
- *_with_bl.hex: sono le varianti esadecimali con bootloader integrato.

Software di controllo - Ground Control Station

La GCS è un'unità di controllo operativo che permette agli utenti di configurare, testare e modificare il veicolo. Può essere eseguita su computer, tablet e smartphone e può comunicare con l'hardware via telemetria o tramite connessione usb.

La GCS è di fondamentale importanza per la visualizzazione dei dati in tempo reale, come prestazioni e posizione del rover, e può fungere da cabina di pilotaggio virtuale in quanto consente anche lo streaming video da telecamera⁹.

Vi è inoltre la possibilità di interagire durante le missioni, inviando specifici comandi

⁸Fonti aggiornate al 15 luglio 2020.

⁹È possibile monitorare la guida autonoma del veicolo, o pilotarlo, tramite visore (occhiali FPV, da non confondersi con i visori VR) in modalità FPV (First Person View).

e modificando determinati parametri di guida in tempo reale. Questa caratteristica è di fondamentale importanza per fronteggiare eventuali inconvenienti, non previsti (o imprevedibili) in fase di pianificazione della missione.

Esistono una moltitudine di *GCS* disponibili gratuitamente. La decisione dipende principalmente dalla tipologia di utente:

- gli utenti che acquistano veicoli pronti all'utilizzo generalmente preferiscono le applicazioni da eseguire sul dispositivi mobili, quali tablet e smartphone;
- gli sviluppatori autonomi hanno la necessità di accedere spesso a strumenti di analisi e configurazione, pertanto hanno la necessità di utilizzare i software più completi, come *Mission Planner*, *APM Planner* 2 e *QGroundControl*.

La GCS più utilizzata per le applicazioni ArduPilot è *Mission Planner*[39]. È stata progettata e realizzata da Michael Oborne, sviluppatore software di Ardupilot dal 2010, di conseguenza il suo utilizzo è fortemente raccomandato dalle guide ArduPilot.

1.3 Global Navigation Satellite System

Con l'acronimo GNSS (Global Navigation Satellite System) si intende il sistema di posizionamento e navigazione costituito da molteplici costellazioni di satelliti artificiali. Elaborando i segnali radio emessi dai satelliti, i ricevitori satellitari calcolano la loro posizione con una tecnica di intersezione spaziale di tipo distanziometrico¹⁰, nel sistema di riferimento geocentrico o ECEF (Earth-Centered Earth-Fixed) proprio di ogni costellazione satellitare. La propria posizione può essere stimata utilizzando diverse tecniche, dalle quali dipende l'accuratezza.

Inoltre, la posizione può essere determinata sia in tempo reale (real-time) sia a posteriori, in una successiva fase di post processamento (post-processing). Le applicazioni real-time sono di particolare interesse per le operazioni di navigazione, mentre il post-processing è di interesse per rilievi statici, nei quali interessa un'elevata precisione.

Attualmente, sono tre le costellazioni ad avere copertura globale del segnale: GPS, Glonass e Galileo. In generale, il sistema è in continua espansione:

¹⁰La distanza geometrica viene calcolata come distanza Euclidea tra satellite e centro di fase dell'antenna del ricevitore. Si veda il paragrafo 1.8 a pagina 37.

- lo sviluppo delle costellazioni regionali, orientato alla copertura globale, sta avanzando rapidamente;
- altre costellazioni sono in fase di pianificazione e realizzazione: il numero di costellazioni è in continuo aumento.

La presenza di molteplici costellazioni, unitamente all'architettura che caratterizza i sistemi di posizionamento satellitari, conferiscono molte caratteristiche positive al sistema GNSS, tra cui:

- Robustezza Intesa come la possibilità di ottenere un posizionamento affidabile e di precisione anche in condizioni sfavorevoli. Per esempio, negli ambienti urbani il segnale satellitare è ostacolato dagli edifici.
- Continuità Il posizionamento è sempre reso possibile, in qualsiasi momento e area del mondo. Questo rende possibile il rilevamento continuo di punti fissi, al fine di stimarne la posizione con elevata precisione.
- Ridondanza Ogni sistema satellitare garantisce, in ogni momento, la visibilità di almeno 4 satelliti: rappresenta il numero minimo per poter stimare la posizione di un ricevitore. La presenza di più costellazioni permette di disporre delle misure di molti più satelliti, rispetto alla configurazione minima, conferendo ridondanza alle misure e al contempo robustezza.
- Precisione e affidabilità I sistemi satellitari sono stati progettati per permettere la stima della posizione di un punto con elevatissima precisione, conferendo affidabilità alle misure ricavate.

1.4 Costellazioni GNSS

Per molti anni, l'unico sistema GNSS pienamente operativo è stato il **GPS** (Global Positioning System), sviluppato dal Dipartimento della Difesa (DoD) degli Stati Uniti d'America. La sua realizzazione fu avviata nel 1973, ed è organizzato in 3 blocchi di satelliti, di cui il primo fu lanciato nel 1978.

In origine il sistema venne concepito per il solo utilizzo militare, successivamente il Congresso degli Stati Uniti d'America decise di consentirne l'utilizzo anche per applicazioni civili. Il GPS divenne pienamente operativo nel 1994.

Il sistema russo **GLONASS** (GLObalnaya NAvigatsionnaya Sputnikovaya Sistema) nasce negli anni 70-80 per scopi militari e solo successivamente viene aperto anche all'utenza civile (decreto del 7 Marzo 1995 del governo della Federazione Russa). La costellazione è stata completata nel 1997 con 24 satelliti in orbita, ma negli anni successivi, a causa della grave situazione economica interna, la Federazione Russa non è riuscita a mantenere attiva l'intera costellazione e pertanto nel

corso degli anni è andata via via in decadenza. Nel 2002, con la ripresa economica della Russia, è stato dato il via al programma di rilancio del sistema satellitare GLONASS, con il ripristino alla piena operatività nel dicembre 2011.

Dopo il declino del GLONASS, l'unico sistema di posizionamento pienamente operativo fu l'americano GPS. Nel 2003 l'Unione Europea e la European Space Agency (ESA) stipularono l'accordo che avviò ufficialmente lo sviluppo del sistema di posizionamento satellitare europeo **ESA - Galileo**, alternativo al GPS. A differenza dei precedenti, Galileo viene progettato per essere sempre disponibile all'utenza civile, non solo militare. Il sistema è diventato operativo nel dicembre 2016 con 18 satelliti. Ulteriori sono stati lanciati negli anni successivi e la sua completa operatività, con 30 satelliti, è prevista entro l'anno 2020.

Nel 2000 anche la Cina ha iniziato lo sviluppo del proprio sistema di posizionamento **BeiDou** (BeiDou Navigation Satellite System). La costellazione completa sarà composta da 35 satelliti, 5 in Orbita Geostazionaria (GEO) e 30 in orbita quasi circolare, non GEO. Nato inizialmente per garantire la copertura solo per la nazione, è stato poi ampliato per diventare un sistema globale, e la sua realizzazione è prevista entro l'anno 2020.

Ogni costellazione è stata progettata per il posizionamento autonomo, con la congiunta possibilità di collaborazione con gli altri sistemi (compatibilità). Si definisce compatibilità la capacità di due sistemi satellitari di essere utilizzati separatamente o insieme senza che i diversi segnali interferiscano. Si definisce invece interoperabilità la capacità di utilizzare due servizi insieme per ottenere prestazioni migliori a livello utente. Un numero crescente di accordi tra gli operatori garantisce l'interoperabilità di sistemi e segnali, rendendo possibile un posizionamento più preciso e affidabile. Il maggior numero di satelliti compensa i segnali disturbati dei singoli satelliti e anche i ricevitori più economici oggi in commercio ricevono i segnali di più costellazioni.

1.5 Architettura dei sistemi GNSS

Ogni sistema GNSS è essenzialmente costituito da 3 segmenti principali:

- Segmento spaziale
- Segmento controllo
- Segmento utente

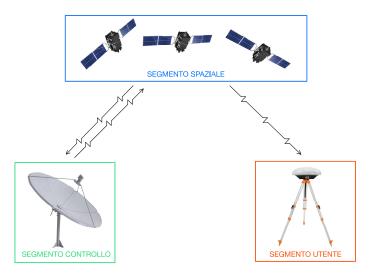


Figura 1.5: Segmenti dei sistemi GNSS

1.5.1 Segmento spaziale

Con segmento spaziale si intende il complesso della costellazione satellitare in orbita intorno alla terra e i segnali radio trasmessi dai singoli satelliti. Le sue funzioni principali sono la generazione e trasmissione di segnali di codice e fase portante, nonché la memorizzazione e trasmissione del messaggio di navigazione caricato dal segmento di controllo. Queste trasmissioni sono controllate da orologi atomici altamente stabili a bordo dei satelliti. Tutte le costellazioni GNSS sono state progettate in modo che siano visibili almeno quattro satelliti contemporaneamente da qualsiasi punto della superficie terrestre in qualsiasi momento.

1.5.2 Segmento di controllo

Il segmento di controllo (noto anche come segmento di terra) è responsabile del corretto funzionamento del sistema. È costituito dai centri operativi a terra per il controllo dello stato di funzionamento dei satelliti, per le operazioni routinarie di telemetria, per l'inseguimento dei satelliti, per l'invio di comandi di manovra, per il calcolo delle orbite dei satelliti e dei dati degli orologi e per l'uplink stesso dei dati. Le funzioni più importanti del segmento di controllo sono:

- il calcolo delle orbite satellitari, la trasmissione delle effemeridi e la determinazione degli errori degli orologi;
- l'applicazione di manovre e correzioni d'orbita, per preservare la configurazione satellitare;
- l'invio del messaggio di navigazione;

• il mantenimento del tempo GNSS (GNSS Time).

In tutte le scale temporali satellitari GNSS si fa riferimento al tempo GNSS. Il tempo GNSS è sincronizzato con il tempo di riferimento che viene indirizzato su UTC. I parametri di precisione del tempo GNSS influenzano l'accuratezza del posizionamento e della temporizzazione nonché l'accuratezza del calcolo e della previsione degli offset del tempo GNSS per l'interoperabilità dei sistemi. L'evoluzione degli orologi permette di ridurre significativamente gli errori di misura. Le effemeridi¹¹ permettono di conoscere con precisione, relativamente alta, le funzioni orbitali e le coordinate dei satelliti.

1.5.3 Segmento utente

Il segmento utente è composto dall'insieme dei ricevitori GNSS. La loro funzione principale è quella di ricevere il segnale dai satelliti, determinare le **pseudorange**¹² (e altre osservabili) e risolvere le equazioni di navigazione al fine di ottenere le coordinate del centro di fase dell'antenna. La precisione del posizionamento satellitare è direttamente dipendente dalla tipologia di ricevitore. Un generico ricevitore GNSS è costituito da una serie di blocchi fondamentali:

- una antenna con preamplificatore associato;
- una sezione di radio frequenza;
- un microprocessore per le operazioni computazionali;
- un oscillatore a precisione intermedia;
- un sistema di alimentazione;
- una memoria per l'archiviazione dei dati.

1.6 Struttura del segnale

La precisione del sistema è garantita dagli orologi atomici a bordo dei satelliti che generano un segnale a elevata stabilità. Gli oscillatori hanno una frequenza fondamentale f_0 a cui corrisponde una lunghezza d'onda:

$$\lambda = \frac{c}{f_0} = \frac{3 \times 10^8 \,\text{m/s}}{f_0}$$
 (1.1)

¹¹Le Effemeridi sono tabelle che contengono valori calcolati, nel corso di un particolare lasso di tempo, di diverse grandezze astronomiche variabili, come le orbite.

¹²Con il termine range viene indicata la distanza satellite-ricevitore.

Le componenti principali del segnale si distinguono in tre parti fondamentali:

- **componente portante**: onde sinusoidali generate moltiplicando la frequenza fondamentale per determinate quantità;
- componente impulsiva codice: sequenze di zero e uno (onde quadre) che consentono al ricevitore di determinare il tempo di viaggio del segnale radio dal satellite al ricevitore. Le onde quadre vengono generate da un algoritmo che si ripete nel tempo in maniera pseudo-casuale. Quando i ricevitori svolgono l'analisi spettrale del segnale, questo appare inizialmente come un rumore e per questo motivo il codice viene anche chiamato come PRN (Pseudo Random Noise);
- messaggio di navigazione: è un segnale binario costruito in modo tale da fornire al Segmento Utente tutte le informazioni necessarie per calcolare correttamente la posizione del ricevitore. Contiene informazioni sulle effemeridi satellitari, parametri di bias degli orologi, almanacchi (con effemeridi a precisione ridotta), stato di salute del satellite e altre informazioni complementari.

Sia le portanti sia i segnali di modulazione GPS sono generati a partire dalla frequenza fondamentale di oscillazione degli orologi a bordo $f_0 = 10.23 \,\mathrm{MHz}$, moltiplicata per valori diversi così da ottenere tutti i segnali.

Componente	Frequenza (MHz)	λ
Frequenza fondamentale	$f_0 = 10.23$	
Portante L1	$154 \cdot f_0 = 1575.42$	$19.0~\mathrm{cm}$
Portante L1	$120 \cdot f_0 = 1227.60$	$24.4~\mathrm{cm}$
Codice P	$f_0 = 10.23$	$29.3 \mathrm{m}$
Codice C/A	$f_0/10 = 1.023$	$293.0~\mathrm{m}$
Messaggio di navigazione	$f_0/204600 = 50E-6$	

Tabella 1.1: Caratteristiche dei segnali GPS [33]

Attraverso il modulatore, i codici modulano le portanti provocandone lo sfasamento di π sulla fase. Conoscendo quindi il codice, è possibile demodulare la portante e risalire al segnale orginale. Il ricevitore crea una replica del segnale in arrivo, applicando un'inversione dove il codice ha modulato l'onda, tornando di fatto a uno sfasamento nullo.

Concettualmente, le osservazioni satellitari sono distanze dedotte dalla differenza di tempo o di fase dell'onda intercorrente tra l'istante in cui il segnale viene emesso dal satellite e l'istante di ricezione in cui viene misurato, moltiplicato per la velocità di propagazione dell'onda.

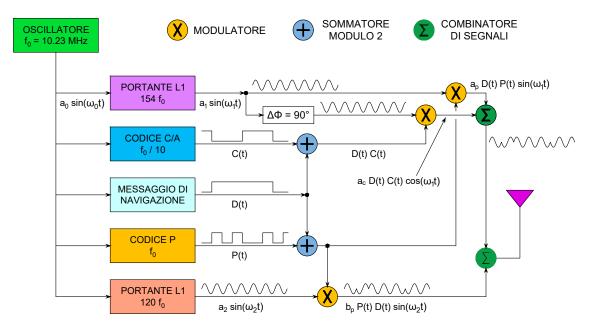


Figura 1.6: Sequenza di generazione del segnale GPS

In questo modo il tempo viene misurato da due orologi indipendenti e si genera un errore di misura dovuto alla non sincronizzazione degli orologi. Per eliminare questo errore ogni sistema satellitare calcola un proprio tempo (GNSS Time), che permette di determinare gli offset dell'orologio del satellite e dell'orologio del ricevitore rispetto al GNSS Time. La distanza che viene di conseguenza determinata prende il nome di pseudo-distanza (pseudo-range) e può essere ricavata secondo due modalità:

- misura di codice: calcolata a partire dal tempo di propagazione del segnale sui codici che modulano la portante: attraverso questa misura si può ottenere un posizionamento istantaneo con bassa precisione;
- misura di fase: la misura viene effettuata attraverso una grandezza campione (la lunghezza dell'onda λ) che, nell'intero segnale captato dal ricevitore, è contenuta un numero intero di volte (numero di cicli) più una frazione: questa tipologia di misura viene utilizzata per posizionamenti di elevata precisione.

1.7 Errori che interessano le misure GNSS

Le misure di range sono affette da errori sistematici e accidentali, che degradano la precisione nel calcolo della posizione. Le sorgenti di errore possono essere classificate in tre gruppi: errori relativi al satellite, errori relativi alla propagazione del segnale ed errori relativi al ricevitore.

Sorgente	Effetto
Satellite	Errori dell'orologio
	Errori dell'orbita
Propagazione del segnale	Rifrazione ionosferica
	Rifrazione troposferica
Ricevitore	Errori dell'orologio
	Variazione del centro di fase dell'antenna
	Multipath

Tabella 1.2: Errori nelle misure GNSS

Considerati tutti i contributi di errore, si ottiene un posizionamento con precisione dell'ordine di alcuni metri, troppo bassa per le operazioni di rilievo. Alcuni errori di tipo sistematico possono essere modellati, dando origine a termini aggiuntivi che rendono le equazioni di codice e di fase più complesse.

In particolare vengono solitamente esplicitati i termini relativi al ritardo ionosferico e troposferico, e agli errori degli orologi: gli errori sistematici possono essere eliminati o ridotti attraverso combinazioni lineari delle osservabili stesse o in una configurazione di rete di ricevitori vicini che effettuano misure contemporanee.

1.7.1 Errori dell'orologio

Come riportato nella tabella 1.2, sia il satellite sia il ricevitore sono soggetti agli errori dell'orologio, che si possono distinguere in:

- asincronismo (offset): traslazione temporale dell'origine dei tempi;
- deriva dipendente dal tempo.

È possibile modellare l'offset dell'orologio dei satelliti δ^{sat} (oscillatori atomici) con un polinomio di grado 2.

Con un polinomio di grado 8 è possibile modellare l'offset dell'orologio del ricevitore δ_{ric} (oscillatori al quarzo).

1.7.2 Errori dell'orbita

Per determinare la posizione dei satelliti in orbita vengono utilizzate le effemeridi. Esistono 3 tipologie di effemeridi:

• broadcast: sono effemeridi predette e disponibili in tempo reale: vengono trasmesse nel messaggio di navigazione e permettono il posizionamento con precisione di qualche metro;

- ultra rapide: sono anch'esse predette e disponibili in tempo reale: vengono calcolate 4 volte al giorno e aumentano molto la precisione del posizionamento, permettendo di arrivare ad accuratezze centimetriche;
- finali: sono calcolate a posteriori: a differenza delle precedenti, non si tratta più di una previsione dell'orbita, ma della misura, da parte del segmento di controllo, della traiettoria effettivamente percorsa dal satellite: sono quindi destinate a operazione di post processamento, permettendo di ottenere precisioni centimetriche.

L'orbita del satellite è sempre affetta da errori di natura geometrica. Detti errori sono particolarmente influenti per il posizionamento assoluto; verrà meglio analizzato più avanti¹³. L'errore d'orbita è meno influente nel posizionamento relativo: Nel posizionamento relativo il fenomeno è caratterizzato da correlazione spaziale, quindi l'errore relativo sulla distanza satellite-ricevitore è comune per tutti i ricevitori coinvolti nel posizionamento relativo, e la precisione finale dipende dalla lunghezza della baseline. Secondo quanto espresso, è possibile scrivere una proporzione tra le grandezze, quindi stimare l'errore in maniera empirica:

$$\frac{\delta r}{r} = \frac{\delta b}{b} \longrightarrow \delta r = \frac{r}{b} \delta b \tag{1.2}$$

dove b indica la lunghezza della baseline, r la distanza satellite ricevitore, δb e δr gli errori relativi. In tabella 1.3 sono riportati alcuni esempi di come l'errore della baseline viene influenzato dall'errore d'orbita:

Errore orbita	Lunghezza baseline	Errore baseline	Errore baseline
2.5 m	1 km	0.1 ppm	- mm
$2.5 \mathrm{m}$	10 km	$0.1~\mathrm{ppm}$	$1 \mathrm{\ mm}$
$2.5 \mathrm{m}$	100 km	$0.1~\mathrm{ppm}$	10 mm
$2.5 \mathrm{m}$	1000 km	0.1 ppm	100 mm
0.5 m	1 km	$0.002~\mathrm{ppm}$	- mm
$0.5 \mathrm{m}$	10 km	0.002 ppm	- mm
$0.5 \mathrm{m}$	100 km	$0.002~\mathrm{ppm}$	2 mm
0.5 m	1000 km	0.002 ppm	2 mm

Tabella 1.3: Errori baseline indotti dall'errore d'orbita

¹³Le tecniche di posizionamento assoluto e relativo sono illustrate al paragrafo 1.8 a pagina 37.

1.7.3 Rifrazione atmosferica

Quando il segnale attraversa l'atmosfera subisce variazioni della velocità di propagazione e deviazioni, dovute alla variazione dell'indice di rifrazione.

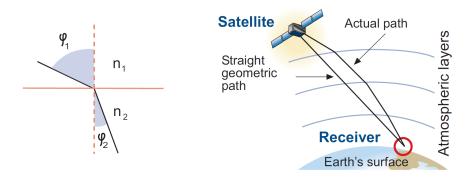


Figura 1.7: Effetti della legge di Snell sul percorso del segnale[33]

La variazione spaziale e temporale dell'indice di rifrazione è dovuta alla non omogeneità della densità del gas e del plasma atmosferici. La legge di Snell afferma che se un'onda elettromagnetica viaggia da un ambiente con indice di rifrazione n_1 a un ambiente con indice di rifrazione n_2 , considerato un angolo di incidenza φ_1 all'interfaccia tra i due mezzi, la sua traiettoria verrà deviata con angolo φ_2 . La relazione che vi intercorre è espressa come:

$$\varphi_1 \cdot n_1 = \varphi_2 \cdot n_2 \tag{1.3}$$

In figura 1.7 viene mostrato come il segnale, attraversando l'atmosfera, passa da un mezzo meno denso a uno più denso. Si noti come il range subisce una curvatura tale per cui la sua distanza percorsa è maggiore rispetto a quella geometrica.

Tra le diverse regioni in cui viene suddivisa l'atmosfera, due in particolare alterano il segnale: la **troposfera** e la **ionosfera**.

Rifrazione troposferica

La troposfera è la fascia dell'atmosfera a diretto contatto con la superficie terrestre. Il suo spessore varia in funzione della latitudine, risultando più spessa all'equatore (circa 20 km) e meno spessa ai poli (circa 8 km). In essa sono concentrati i 3/4 dell'intera massa gassosa e quasi tutto il vapore acqueo dell'atmosfera. L'errore prodotto dal ritardo troposferico è inferiore a 3 mm per satelliti con elevazione maggiore a 20°, aumenta sino 2 cm per elevazioni di circa 10° e raggiunge un massimo di 17 cm per elevazioni di circa 5°[29]. Non è possibile modellare il ritardo sfruttando le due frequenze dei satelliti, in quanto la troposfera è un mezzo non dispersivo per onde radio con frequenze sino a 15 GHz. La rifrattività della troposfera può essere suddivisa in due componenti:

- componente secca (dry): produce circa il 90% del ritardo troposferico totale, dipende principalmente dalla pressione atmosferica ed è facilmente modellabile. La sua entità è di circa 2 m;
- componente bagnata (wet): porta un ritardo massimo di 40 cm, dipende dal vapore d'acqua presente lungo il percorso perciò è difficilmente modellabile.

Esistono diversi modelli per descrivere questo ritardo: i principalmente impiegati sono Hopefield[16] e Saastamoinen[32].

Rifrazione ionosferica

La ionosfera rappresenta la parte alta dell'atmosfera. Si estende tra i 50 e i 1000 km di altitudine: è la fascia in cui le radiazioni solari e, in misura molto minore, i raggi cosmici provenienti dallo spazio provocano la ionizzazione dei gas componenti. Questo fenomeno altera la densità di elettroni e ioni liberi, influenzando sensibilmente l'indice di rifrazione dei segnali radio.

La ionosfera può essere ulteriormente divisa in strati, differenti per composizione e per l'intensità di radiazione solare ricevuta, con proprietà della ionizzazione che varia da strato a strato. Può essere ulteriormente suddivisa in strati, differenti per composizione e intensità di radiazione solare ricevuta, quindi le proprietà della ionizzazione variano da strato a strato. L'entità del ritardo ionosferico sulla L1 è valutata in circa 15 m in direzione zenitale, ma può triplicarsi a basse elevazioni. Essendo la ionosfera un mezzo dispersivo, l'indice di rifrazione varia in funzione della frequenza, perciò la migliore strategia per stimare il ritardo ionosferico è quella di utilizzare due frequenze diverse. Per questo motivo tutti i sistemi di navigazione satellitare, a partire dal vecchio TRANSIT (precursore del GPS), inviano segnali su due frequenze. A causa della sua natura dispersiva, si può porre che la rifrazione sia proporzionale al termine $1/f^2$, dove f rappresenta la frequenza della portante. Indicando con I il ritardo, con i pedici 1 e 2 le portanti, è possibile esprimere la seguente relazione:

$$I_1 \cdot f_1^2 = I_2 \cdot f_2^2 \longrightarrow I_2 = \frac{f_1^2}{f_2^2} I_1$$
 (1.4)

L'effetto della ionosfera su una frequenza può essere valutato in funzione del ritardo sull'altra. Quindi i ricevitori in doppia frequenza sono in grado di eliminare il ritardo ionosferico utilizzando una frequenza ottenuta dalla combinazione lineare delle misure di codice (R) o fase (ϕ) :

$$\phi_{\text{iono-free}} = \frac{f_1^2 \phi_1 - f_2^2 \phi_2}{f_1^2 - f_2^2} \qquad R_{\text{iono-free}} = \frac{f_1^2 R_1 - f_2^2 R_2}{f_1^2 - f_2^2}$$
(1.5)

Questa combinazione prende il nome di Iono-free.

1.7.4 Centro di fase dell'antenna: offset e variazione

La misura della fase dell'onda è riferita al centro di fase dell'antenna. Questo punto è di natura elettronica e si individua teoricamente, senza alcun riferimento meccanico e generalmente non coincide con il centro geometrico dell'antenna. Questo errore varia in funzione dell'elevazione del satellite, dell'azimut e dell'intensità del segnale, per cui è differente per le due portanti. Si distinguono due contributi: uno di offset (costante) e uno di variazione. L'offset, essendo costante, può essere determinato facilmente e di solito è compreso nelle informazioni tecniche del ricevitore. La variazione determina la non univocità della posizione del centro di fase. Può raggiungere 2 cm e, trattandosi di un errore sistematico, può essere mappato per ogni antenna. Infine, per riportare la misura a un punto geometrico, si fa riferimento all'ARP (Antenna Reference Point)¹⁴: viene definito come punto di intersezione tra l'asse verticale di simmetria dell'antenna e il fondo della stessa.

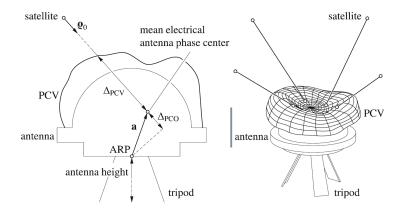


Figura 1.8: Centro di fase elettronico dell'antenna e ARP[15]

1.7.5 Multipath

Il fenomeno è ben descritto dalla sua traduzione letterale: percorso multiplo. Una parte del segnale, rimbalzando sulle superfici vicine al ricevitore, arriva all'antenna in maniera indiretta. La maggior distanza percorsa in questo modo genera un errore di misura, che nel caso delle misure di codice può arrivare a 1,5 volte la lunghezza d'onda: circa 450 m[33]. La modellazione di questo errore non è possibile: l'arbitrarietà delle superfici in prossimità dei luoghi in cui può essere installato un

¹⁴L'utente può confermare la posizione esatta del centro di fase dell'antenna consultando la documentazione disponibile presso l'IGS(International GNSS Service), che ne fornisce la definizione. Si veda ftp://igs.org/pub/station/general/antenna.gra

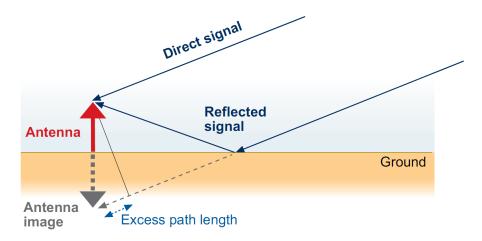


Figura 1.9: Differenze tra segnale diretto e indiretto [33]

ricevitore non permette di realizzare un modello di correzione che abbia validità generale. È comunque possibile stimarne il ritardo prodotto. Si consideri un ricevitore che staziona in un punto fisso.

Il ritardo troposferico e gli errori di orologio sono comuni a tutti i percorsi riflessi. Il ritardo ionosferico e il multipath dipendono dalla frequenza della portante. Operando la differenza tra due combinazioni *iono-free* (1.5), una di codice e una di fase, si elidono quasi tutti i contributi di errore: restano il rumore e il multipath. Il rumore può essere modellato e successivamente eliminato, permettendo di determinare l'entità del multipath. Una volta noto, questo può essere depurato dalle equazioni di misura.

Infine alcuni ricevitori sono in grado di riconoscere i segnali riflessi sfruttando la polarizzazione dell'onda. Il satellite emette un segnale polarizzato circolarmente in senso orario. La riflessione fa invertire la polarizzazione del segnale, la quale diventa antioriaria. I ricevitori in grado di riconoscerla non considerano il contributo del segnale indiretto nelle misure, risolvendo a monte il problema.

1.8 Tecniche di posizionamento basate su segnali GNSS

La distanza geometrica è calcolata come distanza Euclidea tra il satellite e il centro di fase dell'antenna (ricevitore). Per calcolarla, devono essere note le 3 coordinate spaziali, nello stesso sistema di riferimento, di entrambi. Per la generica epoca t:

$$\rho_{ric}^{sat}(t) = \|\mathbf{r}^{sat}(t) - \mathbf{r}_{ric}\|$$

$$= \sqrt{(x^{sat}(t) - x_{ric})^2 + (y^{sat}(t) - y_{ric})^2 + (z^{sat}(t) - z_{ric})^2}$$
(1.6)

Principalmente si distinguono due tipologie fondamentali di posizionamento:

- posizionamento assoluto o stand alone: viene ottenuto con l'impiego di un singolo ricevitore GNSS che, nella maggior parte dei casi, esegue solo misure di codice; nella terminologia internazionale sono sinonimi i termini point positioning, absolute point positioning, single point positioning;
- posizionamento relativo: si ottiene utilizzando le osservazioni, di codice o di fase, di due ricevitori, effettuate simultaneamente sugli stessi satelliti: questa tecnica viene utilizzata per eliminare o mitigare gli errori di tipo sistematico presenti nelle misure e comuni ai due ricevitori.

Inoltre, il posizionamento può essere:

- *statico*: si determina la posizione del punto mediando le coordinate ottenute stazionandovi per un opportuno intervallo di tempo;
- *cinematico*: si ottiene la posizione del ricevitore in continuo movimento: l'insieme delle posizioni misurate descrive la traiettoria percorsa.

Infine, si distinguono due modalità di determinazione delle misure:

- post-elaborazione: le osservazioni vengono memorizzate nella memoria del ricevitore: successivamente i dati verranno scaricati in un calcolatore dove, per mezzo di software di post-processamento dedicati, sarà possibile elaborare le osservazioni;
- tempo reale: la posizione viene determinata nell'istante in cui viene eseguita la misura, al più dopo pochi secondi.

1.8.1 Posizionamento assoluto di codice

Le osservabili di codice si basano sulla misura del tempo di volo, ovvero l'intervallo di tempo che intercorre dal momento di emissione del segnale alla sua ricezione da parte del ricevitore. Moltiplicando questo tempo per la velocità della luce si trova la distanza apparente¹⁵[33]:

$$R_{\rm ric}^{\rm sat} = c\Delta t$$

Le antenne GNSS captano indistintamente i segnali generati da tutti i satelliti, lasciando al ricevitore il compito di associare ogni segnale al rispettivo satellite. Per fare ciò il ricevitore convoglia il segnale su ogni canale a disposizione, e su ognuno confronta il segnale in arrivo con la replica generata localmente del codice di ogni

¹⁵Viene definita apparente per distinguerla dalla reale: espressa in questo modo non considera gli errori

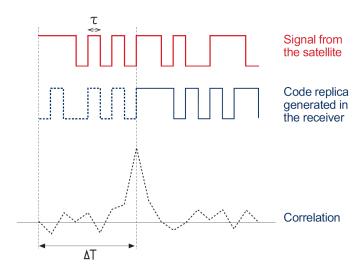


Figura 1.10: Determinazione del tempo di volo del segnale[33]

satellite, fino a individuarne la corrispondenza¹⁶. Una volta riconosciuto il codice PRN del satellite agganciato dal ricevitore, viene introdotta una traslazione (shift) alla replica del codice per allinearla secondo il tempo del proprio orologio. L'entità dello shift costituisce la misura di tempo che genera la pseudo-distanza. L'operazione appena descritta è definita come correlazione di codice (code correlation) e si basa sulla proprietà dei codici pseudo-casuali di mostrare la massima correlazione quando le due sequenze sono perfettamente allineate. Quindi il tempo di volo si determina come:

$$\Delta t = t_{\rm ric} - t^{\rm sat} = [t_{\rm ric}(GNSS) - \delta_{\rm ric}] - [t^{\rm sat}(GNSS) - \delta^{\rm sat}]$$

$$\Delta t = \Delta t(GNSS) + \Delta \delta$$
(1.7)

dove:

$$\Delta t(GNSS) = t_{\rm ric}(GNSS) - t^{\rm sat}(GNSS) \quad e \quad \Delta \delta = \delta^{\rm sat} - \delta_{\rm ric}$$
 (1.8)

Quindi la misura di pseudo-range di codice assume la seguente espressione:

$$R_{ric}^{sat} = c\Delta t = c\Delta t (GNSS) + c\Delta \delta = \rho_{ric}^{sat} + c\Delta \delta$$
 (1.9)

dove δ è la distanza Euclidea (equazione 1.6).

Considerato che le coordinate del satellite e l'offset del suo orologio sono ricavabili dal messaggio di navigazione, per risolvere l'equazione di pseudorange devono essere determinate 4 incognite:

• le 3 coordinate del ricevitore;

 $^{^{16}}$ Questa operazione, denominata $Time\ To\ First\ Fix\ (TTFF)$, rappresenta la prima fase del posizionamento e richiede un certo lasso di tempo per essere completata.

• l'offset dell'orologio del ricevitore.

Per ogni satellite viene scritta una equazione di osservazione, devono quindi essere visibili almeno quattro satelliti per impostare un sistema di equazioni determinato, o al più ridondante, per ogni epoca di osservazione. In funzione del numero di satelliti n_{sat} e di epoche n_{ep} , la condizione minima affinché il sistema sia risolvibile è espressa come:

$$n_{sat}n_{ep} \ge 3 + n_{ep} \tag{1.10}$$

Come precedentemente affermato, nel caso in cui venga considerata una singola epoca di misura sono necessari 4 satelliti. In questo tipo di posizionamento, per ottenere una buona precisione è particolarmente importante, durante la misura, la configurazione geometrica dei satelliti in vista: la loro posizione nello spazio.

Diluition Of Precision I DOP sono indici di bontà geometrica della configurazione satellitare. Hanno origine di carattere statistico in quanto vengono ricavati dalla matrice dei cofattori delle coordinate:

$$Q_{xx} = \begin{bmatrix} q_{xx} & q_{xy} & q_{xz} & q_{xt} \\ q_{yx} & q_{yy} & q_{yz} & q_{yt} \\ q_{zx} & q_{zy} & q_{zz} & q_{zt} \\ q_{tx} & q_{ty} & q_{tz} & q_{tt} \end{bmatrix}$$
Matrice dei cofattori (1.11)

$$GDOP = \sqrt{q_{xx} + q_{yy} + q_{zz} + q_{tt}}$$
 Geometric Diluition Of Precision (1.12a)
 $PDOP = \sqrt{q_{xx} + q_{yy} + q_{zz}}$ Position Diluition Of Precision (1.12b)
 $HDOP = \sqrt{q_{xx} + q_{yy}}$ Horizontal Diluition Of Precision (1.12c)
 $VDOP = \sqrt{q_{zz}}$ Vertical Diluition Of Precision (1.12d)
 $TDOP = \sqrt{q_{tt}}$ Time Diluition Of Precision (1.12e)

I DOP vengono calcolati in tempo reale e sono naturalmente riferiti alla posizione del punto rilevato. Trattandosi di varianze, per assicurare una buona precisione nel posizionamento devono avere valori prossimi all'unità.

1.8.2 Posizionamento assoluto di fase

Per effettuare un posizionamento di precisione è necessario utilizzare le osservabili di fase. Diversamente dalla misura di codice, questa tecnica presenta un'ambiguità di fase iniziale che deve essere determinata, e permette di arrivare a una precisione del posizionamento di pochi millimetri[33].

L'onda sinusoidale si propaga nello spazio alla velocità c, presentando periodicamente gli stessi valori di intensità. Questo fenomeno si verifica in un lasso di tempo definito periodoT nel quale il segnale percorre una lunghezza d'onda λ nello spazio.

Trattandosi di una sinusoide, la sua periodicità è pari a 2π , per cui le grandezze sono correlate tra loro:

$$\frac{t}{T} = \frac{d}{\lambda} = \frac{\varphi}{2\pi} \tag{1.13}$$

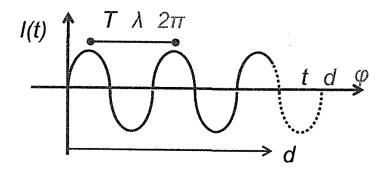


Figura 1.11: Rappresentazione della portante sinusoidale[8]

Sapendo che la fase varia a 0 a 2π ogni periodo T, è possibile definire la velocità angolare come pulsazione $\omega = 2\pi/T$, quindi esprimere la fase come:

$$\varphi = \frac{2\pi}{T}t = \omega t \tag{1.14}$$

Nell'operazione di misura, il satellite viene agganciato all'istante iniziale t_0 nel quale la distanza satellite-ricevitore può essere espressa come somma del numero intero di cicli (periodi) N_{ric}^{sat} più una frazione della fase $\phi_{ric}^{sat}(t_0)$ moltiplicati per la lunghezza d'onda λ :

$$\rho_{ric}^{sat}(t_0) = N(t_0)\lambda + \phi(t_0)\lambda \tag{1.15}$$

Il numero intero di lunghezze d'onda deve essere determinato quindi aggiunge un'incognita al problema. In un'epoca successiva t la distanza viene espressa sempre allo stesso modo, riferita però al nuovo istante temporale t, e sarà diversa in quanto il satellite ha percorso un tratto d'orbita in quel periodo. Dall'istante di ricezione del segnale t_0 , il ricevitore misura la variazione di numero intero di cicli c(t), per cui la distanza può essere espressa in riferimento al numero intero di lunghezze d'onda iniziale N_{ric}^{sat} :

$$\rho_{ric}^{sat}(t) = N(t)\lambda + \phi(t)\lambda = N(t_0)\lambda + c(t)\lambda + \phi'(t)\lambda$$
(1.16)

Considerato che la frequenza è espressa come $f = c/\lambda$, si può dividere l'equazione (1.16) per la lunghezza d'onda λ , esprimendo così la fase ϕ come somma di diversi contributi:[15]

$$\phi(t) = \frac{1}{\lambda}\rho(t) - N(t) - f\left[\delta_{ric}(t) + \delta^{sat}(t)\right]$$
 (1.17)

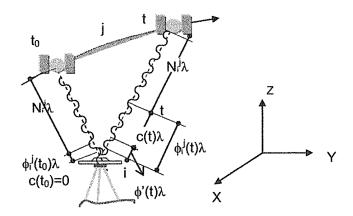


Figura 1.12: Misura di fase agli istanti \mathbf{t}_0 e t[8]

Spostando l'errore dell'orologio del satellite al primo membro, introducendo $\Delta \delta(t) = \delta^{sat} - \delta_{ric}$ l'equazione diventa:

$$\phi(t) + f\delta^{sat}(t) = \frac{1}{\lambda}\rho(t) - N(t) - f\delta_{ric}(t)$$
(1.18)

La configurazione base espressa con la equazione (1.10) diventa:

$$n_{sat}n_{ep} \ge 3 + n_{ep} + n_{sat} \tag{1.19}$$

Considerando la condizione in cui 4 satelliti siano visibili contemporaneamente, sono richieste 3 epoche di misura per arrivare alla soluzione; con 5 satelliti risultano sufficienti 2 epoche.

1.8.3 Posizionamento differenziale

Nel posizionamento differenziale con misure GNSS (abbreviato in \mathbf{DGNSS}) vengono impiegati due o più ricevitori di cui uno ($stazione\ master$) staziona fisso in un punto di coordinate note, acquisendone la posizione con elevata precisione.

Si tratta di una tecnica di misura in tempo reale in cui la stazione master calcola le correzioni, pseudorange correction (PRC) per le misure di codice e Carrier Phase Correction (CPC) per le misure di fase, e le trasmette istantaneamente al ricevitore remoto (rover), permettendogli di aumentare l'accuratezza.

Considerando gli errori correlati spazialmente¹⁷ al tempo iniziale t_0 , si riscrivono, rispettivamente, l'equazione della *pseudo-range* di codice (1.9) e l'equazione di fase

 $^{^{17}{\}rm Rispettivamente}$ indicati come: E_A^{sat} : Effemeridi, I_A^{sat} : Ionosfera, T_A^{sat} : Troposfera

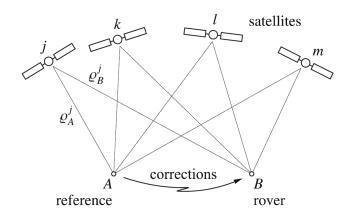


Figura 1.13: Schema del posizionamento differenziale[15]

(1.18), valutate nella stazione master **A**, come: [15]

$$R_A^{sat}(t_0) = \rho_A^{sat}(t_0) - c[\delta_A(t_0) + \delta^{sat}(t_0)] + E_A^{sat}(t_0) + I_A^{sat}(t_0) + T_A^{sat}(t_0)$$

$$\lambda \phi_A^{sat}(t_0) = \rho_A^{sat}(t_0) - c[\delta_A(t_0) + \delta^{sat}(t_0)] - \lambda N_A^{sat} + E_A^{sat}(t_0) + I_A^{sat}(t_0) + T_A^{sat}(t_0)$$

$$(1.20)$$

La correzioni PRC e CPC per il satellite all'epoca t_0 vengono espresse dalle relazioni:

$$PRC^{sat}(t_{0}) = \rho_{A}^{sat}(t_{0}) - R_{A}^{sat}(t_{0})$$

$$= c[\delta_{A}(t_{0}) + \delta^{sat}(t_{0})] - E_{A}^{sat}(t_{0}) - I_{A}^{sat}(t_{0}) - T_{A}^{sat}(t_{0})$$

$$CPC^{sat}(t_{0}) = \rho_{A}^{sat}(t_{0}) - \lambda \phi_{A}^{sat}(t_{0})$$

$$= \lambda N_{A}^{sat} + c[\delta_{A}(t_{0}) + \delta^{sat}(t_{0})] - E_{A}^{sat}(t_{0}) - I_{A}^{sat}(t_{0}) - T_{A}^{sat}(t_{0})$$
(1.21)

calcolabili dalla distanza geometrica $\rho_A^{sat}(t_0)$ espressa dalla equazione (1.6), ottenuta dalle coordinate note della stazione master e dalle effemeridi broadcast, e dalla pseudorange calcolata con la prima delle equazioni (1.20).

Per le misure di codice, in aggiunta alle PRC, vengono calcolate le correzioni di range rate (RRC), per considerare la variazione temporale tra le successive PRC. Quindi la correzione di pseudorange alla generica epoca t viene predetta con le informazioni della precedente epoca t_0 :

$$PRC^{sat}(t) = PRC^{sat}(t_0) + RRC^{sat}(t_0)(t - t_0)$$
(1.22)

dove il termine $t - t_0$ viene definito latenza: intervallo di tempo che intercorre tra il calcolo della correzione dalla stazione master e la sua applicazione dal ricevitore remoto rover. L'accuratezza delle misure è maggiore per piccole variazioni di pseudorange e latenze.

La distanza del ricevitore remoto $\bf B$ all'istante t può essere espressa adattando le (1.20):

$$R_B^{sat}(t) = \rho_B^{sat}(t) - c[\delta_B(t) + \delta^{sat}(t)] + E_B^{sat}(t) + I_B^{sat}(t) + T_B^{sat}(t)$$

$$\lambda \phi_B^{sat}(t_0) = \rho_B^{sat}(t_0) - c[\delta_B(t_0) + \delta^{sat}(t_0)] - \lambda N_B^{sat} + E_B^{sat}(t_0) + I_B^{sat}(t_0) + T_B^{sat}(t_0)$$
(1.23)

Applicando le correzioni PRC (predette) e CPC alle equazioni (1.20), si ricava la distanza corretta del ricevitore remoto \mathbf{B} al generico istante t:

$$R_B^{sat}(t)_{corr} = R_B^{sat}(t) = +PRC^{sat}(t)$$

$$= \rho_B^{sat}(t) - c\delta_{AB}(t) + \Delta E_{AB}^{sat}(t) + \Delta I_{AB}^{sat}(t) + \Delta T_{AB}^{sat}(t)$$

$$\lambda \phi_B^{sat}(t)_{corr} = \rho_B^{sat}(t) = +CPC^{sat}(t)$$

$$= \rho_B^{sat}(t) - c\delta_{AB}(t) - \lambda \Delta N_{AB}^{sat} + \Delta E_{AB}^{sat}(t) + \Delta I_{AB}^{sat}(t) + \Delta T_{AB}^{sat}(t)$$

$$(1.24)$$

dove:

- $\delta_{AB} = \delta_B \delta_A$ rappresenta il contributo degli errori di orologio dei ricevitori: l'errore di orologio dei satelliti si elide, di conseguenza non compare;
- ΔN_{AB}^{sat} rappresenta la singola differenza delle ambiguità di fase.

Nelle precedenti relazioni sono state introdotte le seguenti quantità relative:

$$\Delta E_{AB}^{sat}(t) = E_B^{sat}(t) - E_A^{sat}(t)$$

$$\Delta I_{AB}^{sat}(t) = I_B^{sat}(t) - I_A^{sat}(t)$$

$$\Delta T_{AB}^{sat}(t) = T_B^{sat}(t) - T_A^{sat}(t)$$

$$(1.25)$$

Per distanze moderate tra *master* e *rover* (la cui distanza viene denominata *base-line*), gli errori spazialmente correlati si riducono significativamente, al punto da poter essere considerati trascurabili.

In questi casi, la distanza corretta risulta:

$$R_B^{sat}(t)_{corr} = \rho_B^{sat}(t) - c\delta_{AB}(t)$$

$$\lambda \phi_B^{sat}(t)_{corr} = \rho_B^{sat}(t) - c\delta_{AB}(t) - \lambda \Delta N_{AB}^{sat}$$
(1.26)

Si osservi che nel caso in cui la latenza sia nulla, questa tecnica risulta uguale al posizionamento relativo:

- per le misure di codice l'equazione di code-range coincide con la singola differenza tra i range di codice misurati in A e B;
- per le misure di fase è definito come posizionamento cinematico in tempo reale Real Time Kinematic RTK.

1.9 Correzioni differenziali da rete di stazioni permanenti

Il posizionamento differenziale di fase, illustrato al capitolo 1.8.3, permette di determinare la posizione del ricevitore remoto *rover* con accuratezza centimetrica, nel caso in cui non disti eccessivamente dalla stazione *master*.

Sino a ora non è stata quantificata questa distanza limite. Dai risultati di prove sperimentali [7] si evince come baseline lunghe oltre 10 km sono soggette alla decorrelazione spaziale degli errori, che comporta scarti significativi nelle misure[8]. Questo significa che ogni stazione può inviare correzioni differenziali affidabili entro un determinato raggio di azione.

Con l'evoluzione dei sistemi GNSS e delle tecniche di posizionamento, nacque l'idea di realizzare una rete di stazioni permanenti (*Continuously Operating Reference Station - CORS*) in grado di calcolare e trasmettere le correzioni differenziali ai ricevitori remoti.

Questo significa dover stimare gli errori sistematici nella posizione del ricevitore. Partendo dalle osservazioni continue effettuate dalle singole stazioni permanenti, è possibile ricavare un modello della loro variazione in tutto il territorio monitorato, quindi ricavare le correzioni puntuali per interpolazione.

Concettualmente, questo equivale a creare una *stazione base virtuale* da cui il *rover* riceverà le correzioni.

Questa soluzione ha reso possibile il posizionamento differenziale con accuratezza centimetrica, affidandosi con le osservazioni di stazioni permanenti che distano sino a 80km tra loro.

Trasmissione dei dati Esistono molteplici metodologie con cui è possibile trasmettere i dati GNSS. I principali mezzi di trasmissione sono le bande *UHF* e *Internet IP*; i principali formati di dati, generalmente accettati da tutti i ricevitori, sono:

- RINEX Receiver Independent Exchange Format: è un formato indipendente dal tipo di ricevitore, per questo leggibile dalla quasi totalità dei software di elaborazione;
- RTCM Radio Technical Commision for Maritime Services: è il protocollo di comunicazione ¹⁸ utilizzato per le correzioni differenziali e le misure in tempo reale;

¹⁸La definizione di protocollo di comunicazione è data al capitolo 2.2.

• NMEA - National Marine Electronics Association: è un formato ASCII ed è il principale protocollo applicato per la trasmissione di dati tramite porta seriale.

1.9.1 Trasmissione delle correzioni via NTRIP

Di largo impiego è la trasmissione delle correzioni tramite *Internet IP*, da molti anni disponibile a costi accessibili e oggi sempre più performante. Per trasmettere le correzioni differenziali con lo standard RTCM via internet, la *BKG* (Bundesamt für Kartographie und Geodäsie)¹⁹ ha sviluppato il protocollo **NTRIP** (Networked Transport of RTCM via Internet Protocol)[45].

Si tratta di un protocollo generico di livello applicativo, basato sul protocollo $\mathrm{HTTP}/1.1^{20}$ migliorato per supportare lo streaming di dati GNSS. Permette la connessione simultanea di utenti sia stazionari sia in movimento.

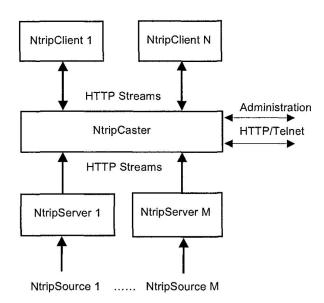


Figura 1.14: Componenti software del protocollo NTRIP[45]

L'architettura del protocollo Ntrip, illustrata in figura 1.14, è costituita da quattro componenti software:

- Ntrip Source: genera un flusso di dati a uno specifico Ntrip Server;
- Ntrip Server: trasferisce il flusso di dati al Ntrip Caster;

¹⁹Agenzia federale per la cartografia e la geodesia tedesca, commissione responsabile dei sistemi di riferimento europei. Si veda https://www.bkg.bund.de/DE/Home/home.html

²⁰supporta lo streaming di dati tramite protocollo TCP/IP e UDP.

- Ntrip Caster: è un server HTTP, duplica la sorgente dei dati per distribuire le correzioni RTCM/RTK via Internet a più client;
- Ntrip Client: è una radio che riceve il flusso di dati e lo incanala sul ricevitore.

In questo sistema, l'utente è rappresentato dal Ntrip Client.

Capitolo 2

Materiali e Metodi

Nel presente capitolo sono esposte le caratteristiche dei componenti individuati, unitamente ai criteri di scelta e alle procedure sperimentali impiegate per la loro implementazione.

2.1 Scelta dei componenti e disegno sperimentale

L'obiettivo di questo progetto è la creazione di un UGV capace di navigare autonomamente in ambiente esterno con elevata precisione, da progettare e realizzare con costi ottimali, logica riportata al paragrafo 1.2.3.

Di conseguenza, la soluzione individuata prevede l'adattamento di un'automobile RTR RC mediante l'utilizzo di hardware e software liberi, attualmente molto performanti, unitamente all'utilizzo di un ricevitore GNSS a elevate prestazioni, il quale deve fornire i dati di posizionamento per svolgere la navigazione con elevata precisione. Inoltre, questa soluzione consente di utilizzare i dati di posizionamento, acquisiti durante le missioni, per eventuali operazioni di post-processamento.

Un progetto così strutturato è basato sull'utilizzo dell'autopilota ArduPilot, pertanto si è fatto affidamento alla documentazione presente nel sito di Ardupilot[4]. Molte procedure sono descritte in maniera completa ed esaustiva. Inoltre, sulla base del veicolo che si sta realizzando, vengono indicati i componenti raccomandati da utilizzare. Il rispetto delle indicazioni e dei consigli riportati nella guida aumenta le probabilità di successo, determinando in questo caso la scelta di diversi componenti.

Infine, qualora dovessero insorgere problematiche, si può fare ricorso ai forum di sviluppatori: è possibile trovare una descrizione del proprio problema, una eventuale soluzione o comunque delle discussioni di supporto alla risoluzione degli inconvenienti. E nel caso in cui si dovessero riscontrare eventuali complicazioni non ancora discusse, è possibile esporre la propria problematica a una comunità di utenti generalmente competente e disponibile.

2.1.1 Veicolo - Traxxas X Maxx

Il veicolo adottato è un Traxxas X Maxx: un modello di fuoristrada radiocomandato (RC) molto potente e di grandi dimensioni (scala 1:6). Le sospensioni robuste e la trazione integrale permettono di affrontare suoli sconnessi e sdrucciolevoli, e il suo contenuto raggio di sterzata consente di effettuare tutte le manovre necessarie, anche in piccoli spazi.

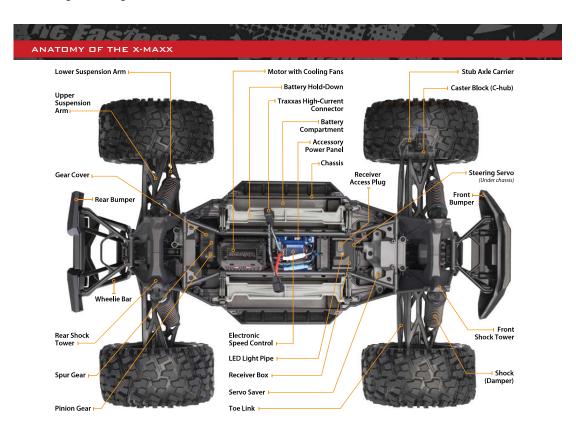


Figura 2.1: Traxxas X Maxx

Spinto da un motore brushless Velineon 1200 XL da 1200Kv^1 , può raggiungere la velocità di oltre 80 km/h. L'erogazione della potenza è regolata dal controllore elettronico della velocità (*Electronic Speed Control - ECS*) VXL-8s, progettato per operare con la potenza erogata da batterie agli ioni di litio (LiPo). Sono richieste 4, 6 o 8 celle (disposte in serie: S).

 $^{^1\}mathrm{Con}\;\mathrm{Kv}$ si indica il numero di rotazioni al minuto che il motore compie quando viene applicato 1 V. Questo dato viene utilizzato per determinare la velocità di rotazione del motore per una data tensione. Con un motore da 1200 Kv, alimentato da 2 batterie a 4 celle (8 S in totale) si raggiungono 35 520 rpm.

La possibilità di raggiungere elevate velocità non è di particolare interesse. Le generose dimensioni del veicolo permetterebbero l'eventuale futura installazione di altri strumenti di rilievo. Il *Rover* deve trasportare un maggior peso rispetto a quello di progetto: si vuole impiegare l'elevata potenza per rendere possibile la navigazione anche in queste condizioni.



Figura 2.2: Dimensioni globali del veicolo

La presenza di un singolo motore, da cui il movimento viene ripartito sulle 4 ruote, permette una più leggera configurazione dell'autopilota. La presenza di due motori separati, rispettivamente per le ruote anteriori e posteriori, comporterebbe una più articolata procedura di configurazione e un maggior onere computazionale.

Infine, è stata considerata la popolarità dei modelli prodotti da Traxxas: un marchio molto forte nella comunità RC e la stessa Ardupilot ha realizzato diversi progetti UGV basati su loro modelli.

2.1.2 Autopilot Hardware

L'autopilota individuato corrisponde al modello raccomandato dalla guida Ardu-Pilot: il Pixhawk 1. Di conseguenza, le periferiche necessarie alla navigazione sono state scelte sempre seguendo le indicazioni della guida, la quale assicura una compatibilità del tipo plug&play²: una volta connesse al Pixhawk non è necessario configurarle per poterle utilizzare.

Pixhawk L'autopilota Pixhawk 1 è un controllore di volo basato sul progetto hardware FMUv2-Pixhawk (combina le funzionalità di PX4FMU + PX4IO) ed esegue il

² ArduPilot consente comunque la modifica dei parametri per la connessione: possono insorgere eventuali problematiche da risolvere con una modifica delle impostazioni, o può essere necessario apportare delle modifiche per particolari sviluppi avanzati, da parte di utenti esperti.

codice PX4 sul sistema operativo $NuttX^3[42]$. Originariamente prodotto e venduto da 3DRobotics[1], attualmente è disponibile una versione leggermente migliorata prodotta da mRobotics[23]. Le sue specifiche tecniche performanti lo rendono idoneo al controllo di qualsiasi veicolo autonomo. Possiede una sensoristica integrata di supporto alla guida e un elevato numero di interfacce input/output (I/O). In generale, tutte le periferiche vengono automaticamente rilevate e configurate una volta connesse.

Componente	Specifiche
Processore	32bit STM32F427 Cortex-M4F core with FPU
	168 MHz
	256 KB RAM
	2 MB Flash
	32 bit STM32F103 failsafe co-processor
Sensori	ST Micro L3GD20H 16 bit gyroscope
	ST Micro LSM303D 14 bit accelerometer $/$ magnetometer
	Invensense MPU 6000 3-axis accelerometer/gyroscope
	MEAS MS5611 barometer
Interfacce	5x UART, one high-power capable, 2x with HW flow control
	2x CAN (one with internal 3.3V transceiver)
	Spektrum DSM/DSM2/DSM-X® Satellite compatible input
	Futaba S.BUS [®] compatible input and output
	PPM sum signal input
	RSSI (PWM or voltage) input
	I2C
	SPI
	3.3 and 6.6V ADC inputs
	Internal microUSB port and external extension
Caratteristiche	External safety switch
	Multicolor LED main visual indicator
	High-power, multi-tone piezo audio indicator
	microSD card for high-rate logging

Tabella 2.1: Specifiche Pixhawk

Non avendo particolari vincoli fisici legati al veicolo, la scelta di questo modello risulta la soluzione altamente raccomandata dalla guida:

 $^{^3}$ NuttX è un sistema operativo molto compatto ed efficiente, progettato per applicazioni *embedded*, ovvero sistemi governati da un microprocessore progettati per un determinato utilizzo, come il Pixhawk.



Figura 2.3: Pixhawk 1

- è direttamente compatibile con un'ampia gamma di sensori fondamentali, collegabili nelle porte dedicate;
- supporta la ridondanza di moduli GNSS e unità di misura inerziale (IMU);
- riconosce sensori complementari grazie alla presenza di porte addizionali, come la I2C e la CAN;

Periferiche Per poter raggiungere la configurazione operativa minima, sono state individuate le periferiche *essenziali* da collegare al controllore di volo:

- un modulo GNSS con chip U-Blox e bussola integrata;
- due radio per la telemetria, operanti sulla frequenza di 433 MHz;
- un modulo di alimentazione (Battery Eliminator Circuit BEC);
- una scheda di memoria micro SD da 8Gb;
- un interruttore Safety Switch+LED;
- un segnalatore audio buzzer;

- una ricevente FrSky modello X4R, con canale SBUS OUT⁴;
- una radio trasmittente modello FrSky Taranis X9D Plus;
- cablaggi con connettore JST-GH a una estremità e DF-13 all'altra;
- supporto con 4 smorzatori antivibrazione.

Quando si acquista un veicolo RTR RC, questo è sempre fornito di una ricevente originale: è un componente essenziale per il radio controllo. I moduli preinstallati però non sono idonei per le applicazioni di guida autonoma, pertanto si deve rimuovere il componente originale e sostituirlo con uno adatto, in questo caso presente nell'elenco precedente.

2.1.3 Modulo GNSS a elevate prestazioni - Piksi Multi

La difficoltà maggiore nella realizzazione di questo progetto è la realizzazione dalla navigazione autonoma con posizionamento di elevata precisione, da ottenersi con tecnica differenziale RTK (1.8.3). Il modulo Ublox, individuato nel paragrafo precedente, determina la propria posizione con accuratezza di qualche metro, dimostrandosi inadeguato allo scopo. Viene pertanto implementato come unità secondaria.

Per ottenere il posizionamento di precisione è stato necessario individuare un modulo GNSS ad alte prestazioni che fosse: piccolo, leggero, capace di determinare la posizione velocemente (requisito fondamentale per la navigazione) e compatibile con il controllore *Pixhawk*.

Il modulo *Piksi Multi*, prodotto dall'azienda americana *Swift Navigation*⁵, soddisfa tutti i requisiti, e la sua implementazione è supportata da una dettagliata guida realizzata dalla casa madre[36]. Di seguito le principali caratteristiche:

- accuratezza centimetrica;
- ricezione correzioni RTK;
- rapida convergenza alla soluzione RTK;
- bassa latenza delle soluzioni, < 30 ms;
- ricevitore multicostellazione in doppia frequenza;

⁴La presenza del canale SBUS OUT(o PPM OUT) permette di inoltrare gli input della trasmittente al controllore di volo.

⁵Swift Navigation sviluppa ricevitori GNSS ad alta precisione, concepiti secondo la filosofia dell'hardware libero, ed economicamente accessibili[35].

- robustezza delle informazioni di posizionamento;
- interfacce adattabili: UART, Ethernet, CAN, USB;
- protocolli di comunicazione: Swift Binary Protocol (SPB) e NMEA 0183;
- MEMS⁶ IMU e magnetometro integrati;
- tensione di alimentazione compresa tra 5.0 e 15.0 V.

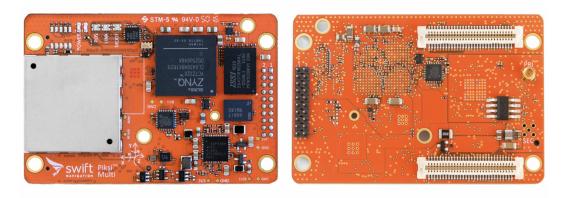


Figura 2.4: Piksi Multi, fronte (sinistra) e retro (destra)

La precisione del centimetro è un requisito fondamentale per la navigazione autonoma. Inoltre, la rapida convergenza alla soluzione RTK comporta una fase di inizializzazione molto veloce, di notevole importanza nel caso di perdita del segnale durante una navigazione avviata. Infine, la ricezione dei segnali di più costellazioni rende le soluzioni molto robuste, migliorando le prestazioni della navigazione in ambienti con visibilità del cielo limitata o ostacolata.

Come esposto nello stato dell'arte (1.8.3), per ottenere il posizionamento differenziale RTK è necessario l'impiego di due ricevitori capaci di comunicare tra loro. Swift Navigation commercializza un set, completo di tutti i componenti necessari, concepito per questo scopo: Piksi Multi Evaluation Kit. Tra tutti, in particolare:

- 2 ricevitori GNSS Piksi Multi;
- 2 evaluation board (scheda hardware che consente la connessione diretta del Piksi Multi a diverse periferiche);

⁶La tecnologia *MEMS* - *Micro Electro Mechanical Systems* consente di integrare sullo stesso substrato di silicio circuiti elettronici e dispositivi opto-meccanici. In questo modo è possibile realizzare dei sensori in grado di elaborare istantaneamente i dati registrati al fine di svolgere determinate azioni.

- 2 antenne GNSS;
- 2 radio per la trasmissione delle correzioni RTK.

L'antenna in dotazione nel *Piksi Multi Evaluation Kit* è inadeguata⁷ per l'installazione sul Traxxas X Maxx: la sua maggior dimensione, il diametro pari a 152 mm, è quasi pari alla larghezza media del supporto⁸, 170 mm su cui devono essere installati tutti i componenti. Pertanto, per il Piksi da installare nel *Rover* è stata individuata un'antenna elicoidale, modello *M1227HCT-A2-SMA* prodotto dall'azienda *Maxtena Inc.*, a elevate prestazioni: con il diametro di 30 mm può essere installata agevolmente sul *Rover*.

2.1.4 Ground Control Station - Mission Planner

La Ground Control Station selezionata in questo progetto è $Mission\ Planner$, progettata e realizzata da Michael Oborne, sviluppatore software di Ardupilot dal 2010. Si tratta della prima GCS mai realizzata: è la più completa e offre tutti gli strumenti di configurazione e analisi necessari per lo sviluppo di questo UGV.

Sarà utilizzata sia per la configurazione del Rover sia per la programmazione delle missioni autonome. Collegando il controllore di volo *Pixhawk* tramite interfaccia USB, è possibile configurare tutti i parametri necessari sia per leggere gli input delle periferiche collegate sia per gestire i segnali di output. Tra le principali caratteristiche di pianificazione delle missioni, in particolare le seguenti:

- possibilità di scelta tra molteplici mappe di background, tra cui Google Maps, ArcGIS, Bing, Open street maps e la possibilità di scegliere anche una mappa personalizzata come background;
- inserimento diretto dei waypoint tramite click con mouse sulla mappa;
- comandi di missione elencati e selezionabili tramite menù a tendina;
- download log file della missione per post elaborazioni;
- configurazione del proprio airframe;
- simulatore di missione SITL;
- lettura degli output dal Pixhawk.

⁷Nella fase di collaudo è stata comunque installata e collegata al ricevitore *Tersus BX316* (si veda capitolo 3.6 a pagina 112), trovando collocazione nella coda del veicolo: ulteriori test sono necessari per valutarne l'impiego per il *Piksi Multi*.

⁸Le caratteristiche del supporto sono descritte al paragrafo 2.3.1.

Mission Planner è organizzato in schede, tramite le quali si accede alle schermate di configurazione dei parametri, pianificazione della missione, monitoraggio in tempo reale del veicolo e simulazione della missione. Le diverse sezioni sono catalogate nella regione in alto a sinistra dell'interfaccia del programma.



Figura 2.5: Schermata di configurazione della GCS Mission Planner

Software - Simulatore SITL

Il simulatore SITL (Software In The Loop) permette di emulare il comportamento del rover in missione, senza collegare alcun hardware. Si tratta di una build⁹ del codice di guida autonoma che utilizza un normale compilatore C++, offrendo un eseguibile nativo con il quale poter testare il comportamento del proprio airframe.

Prima di svolgere una nuova missione con un qualsiasi veicolo, è necessario simularne il comportamento. La missione consiste nel passaggio per punti di riferimento, chiamati waypoint, i quali vengono definiti dal pianificatore. Se tutti i parametri di guida che interessano la missione non vengono impostati correttamente, il veicolo non riuscirà a rispettare la traiettoria. La simulazione permette di comprendere l'influenza di ogni singola impostazione senza che si verifichino danni o imprevisti

⁹In ambito di sviluppo software, il termine inglese *build* indica il processo di trasformazione del codice sorgente in un artefatto eseguibile.

come in una sperimentazione sul campo. Questo tema è affrontato al paragrafo 3.2.



Figura 2.6: Schermata di Mission Planner con il simulatore SITL in esecuzione

2.2 Interconnessione e protocolli di comunicazione

Le periferiche hardware individuate nel paragrafo 2.1.2 sono di tipo plug and play. L'unico componente che il *Pixhawk* non riconosce automaticamente è il modulo GNSS ad alte prestazioni *Piksi Multi*. In ogni caso, *Ardupilot* supporta *Piksi Multi*: il loro funzionamento dipende dalla corretta configurazione dei due componenti.

Per realizzare il posizionamento differenziale RTK, è necessario collegare un secondo ricevitore Piksi Multi al computer: è la stazione master definita al paragrafo 1.8.3. Il Piksi produce in output dei messaggi nel protocollo proprietario SBP - Swift Binary Protocol: si tratta di un protocollo binario minimale realizzato per la comunicazione tra dispositivi prodotti dall'azienda Swift Navigation.

In un classico sistema RTK i due ricevitori sono direttamente collegati tra loro. In questo genere di applicazioni invece il collegamento non è diretto. Il Piksi Multi a bordo del UGV è connesso esclusivamente al Pixhawk sulla porta SERIAL 4/5, e deve ricevere le correzioni da quella stessa porta. A sua volta, il Pixhawk è connesso alla GCS via telemetria: le correzioni dovranno essere inviate tramite Mission Planner. Per poter trasmettere correttamente i messaggi, la GCS deve ricevere le correzioni differenziali con protocollo UDP - $User\ Datagram\ Protocol$. Questa trasmissione è resa possibile dal software $Swift\ Console$. Si rimanda al paragrafo 2.5.2 per la trattazione dettagliata.

Cablaggi e connettori

Le periferiche devono essere collegate al Pixhawk nelle rispettive porte seriali. Ogni porta seriale è costituita da un insieme di contatti (pin) e dal telaio di plastica che li circonda, realizzato secondo uno standard definito dalla sigla del connettore. Il connettore, in generale, rappresenta l'interfaccia fisica del collegamento pin-topin: ogni cavo è composto da più fili conduttori, ogni filo conduttore collega un pin a un'estremità con un altro pin all'altra estremità. Con il termine piedinatura (pinout) si descrivono le funzioni e le caratteristiche del singolo contatto (pin) in un connettore elettrico. Per connettere correttamente le periferiche, si deve disporre di cavi con connettore DF13 ad almeno un'estremità (per il Pixhawk) e si devono conoscere i pinout di ogni porta, sia del controllore sia della periferica.

Protocollo di comunicazione

Con il termine protocollo si indica l'insieme di regole che devono essere rispettate da due o più interlocutori digitali tra loro connessi per poter dialogare. I protocolli di comunicazione sono fondamentali per lo scambio di messaggi tra diversi sistemi informatici. Possono includere funzioni di autenticazione e segnalazione/correzione degli errori. Inoltre, possono descrivere la sintassi e la modalità di sincronizzazione delle comunicazioni digitali. Per consentire la corretta trasmissione, i dispositivi di comunicazione devono anticipatamente concordate molti aspetti fisici dei dati da scambiare [18]. Tra le proprietà che sono definite dal protocollo, in particolare:

- velocità di trasmissione;
- dimensione del pacchetto;
- tecniche di sincronizzazione:
- mappatura degli indirizzi;
- tipi di correzione degli errori.

Nell'architettura moderna dei sistemi informatici, i protocolli sono stratificati in livelli (*layers*): ogni livello svolge una specifica funzione.

Protocollo di rete

Quando gli interlocutori digitali sono remoti¹⁰, le regole di comunicazione sono definite da un *protocollo di rete*. In particolare, definisce il modo in cui devono essere impacchettati i dati per la trasmissione in rete: permette al ricevente di consegnarli e spacchettarli nella maniera più opportuna.

¹⁰La connessione remota viene realizzata per mezzo di una rete informatica (*Internet*): i dispositivi connessi dialogano senza essere fisicamente collegati tra loro.

SBP Swift Binary Protocol è il protocollo nativo utilizzato dal Piksi Multi per trasmettere soluzioni, osservazioni, stato, messaggi di debug¹¹ e correzioni differenziali. È quindi un'interfaccia fondamentale tra il ricevitore GNSS e il sistema con cui è connesso. Il protocollo SBP è suddiviso in due parti:

- un over-the-wire message framing format¹²: indicazione della suddivisione dei bit in pacchetti;
- definizione strutturata del carico utile (payload): è la parte di dati trasmessi destinata all'utilizzatore.

Si riporta la struttura del messaggio SBP in tabella 2.2. Si noti che la variabile N denota la dimensione variabile del messaggio.

Offset (bytes)	Dimensione (bytes)	Nome	Descrizione			
0	1	Preamble	Indica l'inizio del messaggio. Ha sempre valore 0x55.			
1	2	Message Type	Identifica il contenuto del Payload			
3	2	Sender	Identifica in maniera univoca il mittente. Nei dispositivi Piksi è impostato sui 2 byte meno significativi del numero seriale del dispositivo. Di default i mittenti hanno valore 0x42: in questo modo si individuano i software di controllo.			
5	1	Length	Lunghezza, in termini di bytes, del campo <i>Payload</i> .			
6	N	Payload	Contenuto del messaggo binario.			
N + 6	2	CRC	Controllo di ridondanza ciclica (<i>Cyclis Redundancy Check</i>) della stratificazione dei dati binari dal <i>Message Type</i> sino alla fine del <i>Payload</i> (che non include il <i>Preamble</i>).			
N + 8			Lunghezza totale del messaggio.			

Tabella 2.2: Struttura del messaggio SBP

¹¹I messaggi di debug segnalano la presenza di errori durante le operazioni di misura

 $^{^{12}}$ Con il termine framing si indicano l'incapsulamento dei dati con una intestazione (header) e l'interpretazione dei bit presenti in dette intestazioni.

UDP User Datagram Protocol è uno dei principali protocolli di rete: utilizza il protocollo Internet¹³ per attuare la comunicazione tramite datagrammi (pacchetti di dimensioni limitate a 8 kbit) da un dispositivo all'altro, attraverso una rete.

È un protocollo leggero: fornisce una procedura, ai programmi applicativi, per trasmettere dati con un meccanismo di protocollo minimo[30]. Ha una velocità di trasferimento elevata, dovuta alla leggerezza dei pacchetti e alla struttura minimale dell'intestazione (*Header*). UDP invia pacchetti al ricevitore indipendentemente dal fatto che sia in grado di riceverli completamente o meno: non chiede una risposta da parte del ricevitore.

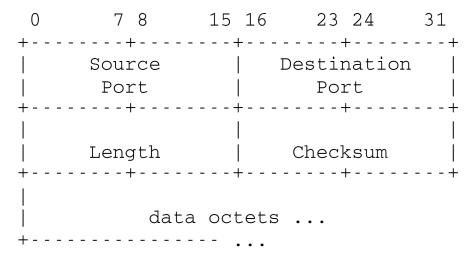


Figura 2.7: Intestazione (*Header*) del protocollo UDP[30]

Il protocollo UDP è ideale per lo streaming di dati in tempo reale: non permette il rinvio di pacchetti eventualmente persi e per questo motivo ha elevata velocità di trasmissione, fondamentale per questo tipo di applicazioni. Difatti, le osservazioni SBP sono trasmesse tramite UDP, per poterne disporre nel minor lasso di tempo possibile.

MAVLink Micro Air Vehicle Link (MAVLink) è un protocollo di comunicazione leggero, utilizzato per trasmettere dati tra un AutoPilot hardware (del *UGV* in questo caso) e la GCS. I messaggi MAVLink possono essere inviati su quasi tutte le connessioni seriali e non dipendono dalla tecnologia sottostante: possono essere trasmessi anche via telemetria[41].

Come per il protocollo UDP, anche in questo caso non è garantito che i messaggi inviati vengano recapitati dal ricevente: la GCS deve controllare continuamente

 $^{^{13}}$ Il protocollo *Internet* è un protocollo di rete che determina su quale porta o interfaccia debba essere inviato un elemento di comunicazione ricevuto.

lo stato del veicolo per determinare se i comandi vengono eseguiti. Ogni messaggio MAVLink può raggiungere una lunghezza massima di 263 byte; si riporta la struttura in figura 2.9.

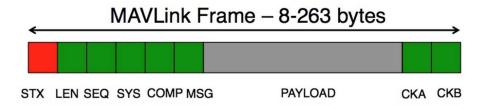


Figura 2.8: Composizione dei bytes del messaggio MAVLink

Byte Index	Content	Value	Explanation			
0	Packet start sign	v1.0: 0xFE (v0.9: 0x55)	Indicates the start of a new packet.			
1	Payload length	0 - 255	Indicates length of the following payload.			
2	Packet sequence	0 - 255	Each component counts up his send sequence. Allows to detect packet loss			
3	System ID	1 - 255	ID of the SENDING system. Allows to differentiate different MAVs on the same network.			
4	Component ID	0 - 255	ID of the SENDING component. Allows to differentiate different components of the same system, e.g. the IMU and the autopilot.			
5	Message ID	0 - 255	ID of the message - the id defines what the payload "means" and how it should be correctly decoded.			
6 to (n+6)	Data	(0 - 255) bytes	Data of the message, depends on the message id.			
(n+7) to (n+8)	Checksum (low byte, high byte)	ITU X.25/SAE AS-4 hash, excluding packet start sign, so bytes 1(n+6) Note: The checksum also includes MAVLINK_CRC_EXTRA (Number computed from message fields. Protects the packet from decoding a different version of the same				
		packet but with different variables).				

Figura 2.9: Struttura del messaggio

Il mittente compila sempre i campi System ID e Component ID per permettere al destinatario di conoscere la provenienza del pacchetto. Il System ID assume il valore predefinito 1 per il veicolo, mentre vengono assegnati valori elevati (come 255) alle stazioni di terra: questo permette sempre una rapida identificazione.

Altri dispositivi collegati al sistema (come unità di elaborazione NUC), capaci di comunicare tramite il protocollo MAVLiknk, assumono lo stesso System ID del controllore di volo (del veicolo): vengono distinti con diversi valori del Component ID, che vale generalmente 1 per il controllore di volo e per la stazione di terra.

2.3 Implementazione

Nei precedenti paragrafi sono state illustrate le modalità di comunicazione e le caratteristiche dei diversi componenti individuati per la creazione dell'UGV oggetto di questo progetto. Per poterli assemblare correttamente e quindi giungere al prodotto finale funzionante, devono essere seguite specifiche regole di connessione e configurazione dei componenti.

Si deve dapprima individuare uno schema di realizzazione, in cui devono essere definiti i collegamenti dei componenti, sia fisici sia di rete, e la loro collocazione. Successivamente, la realizzazione del veicolo autonomo deve essere realizzata secondo un approccio di tipo step-by-step: si inizia dalla configurazione del solo controllore di volo. Successivamente si collegano e configurano i componenti essenziali. Infine, si configura il modulo GNSS ad alte prestazioni Piksi Multi.

2.3.1 Schema di realizzazione

Per realizzare e utilizzare il Rover - UGV è necessario realizzare un sistema connesso che prende il nome di Ground Station - Statione di terra¹⁴: comprende l'insieme di tutti gli elementi, collegati alla GCS, che comunicano con il Rover via telemetria durante la missione.

Se con il termine UGV si identifica il veicolo di terra, compreso l'hardware fisicamente collegato, per svolgere le missioni autonome è necessario che il Rover sia connesso alla GCS: si può definire $sistema\ completo$ l'insieme di queste due entità.

Di conseguenza, si rende necessaria un'ulteriore precisazione nell'utilizzo del termine *autonomo*: riguarda la sola capacità da parte del veicolo di prendere delle decisioni riguardo le manovre da compiere per percorrere un tragitto individuato dal progettista (o pianificatore).

Il veicolo autonomo non può esistere senza la *Ground Station*: è un elemento necessario per la configurazione del controllore e delle periferiche, per la pianificazione delle missioni e per il monitoraggio del *Rover* durante le missioni.

In questo progetto, la stazione di terra è composta dai seguenti elementi:

- software Mission Planner;
- notebook¹⁵ Dell XPS 15 che esegue il programma sul sistema operativo Windows 10;

 $^{^{14}}$ Da non confondersi con la *Ground Control Station - GCS*, che rappresenta il solo software in esecuzione sul computer.

¹⁵L'utilizzo di un computer portatile, rispetto a un fisso, è necessario per lo svolgimento delle missioni all'aperto.

- correzioni RTK, generate da un ricevitore master o scaricate da NTRIP CASTER;
- radio di telemetria, collegata tramite connessione usb al computer;
- radio trasmittente.

e il Rover - UGV è composto dai seguenti elementi:

- veicolo Traxxas X Maxx, compreso di tutte le sue componenti originali, fatta eccezione per la carenatura e la ricevente;
- controllore Pixhawk;
- modulo GNSS Ublox con bussola integrata (GNSS secondario);
- Piksi Multi (GNSS primario);
- ricevente;
- radio di telemetria.

Ogni elemento deve essere installato in una posizione ben definita: deve essere facilmente accessibile, sufficientemente distante dagli altri componenti e prontamente sostituibile. Analizzando il telaio originale del veicolo, non è stata individuata una collocazione che soddisfacesse tutti i requisiti.

Per accogliere tutti gli elementi in una sede stabile e ordinata, è stato progettato un supporto, con l'ausilio di un software di modellazione 3D. La prima versione sarà realizzata tramite stampa 3D. Qualora risulteranno necessarie successive modifiche, si provvederà alla realizzazione di un secondo modello. Una volta individuato il disegno definitivo, si procederà alla costruzione del supporto in fibra di carbonio.



Figura 2.11: Supporto delle periferiche hardware

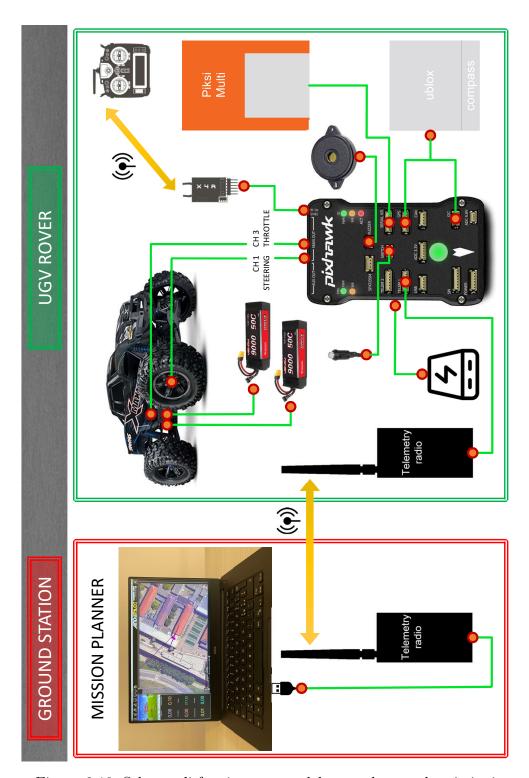


Figura 2.10: Schema di funzionamento del rover durante le missioni

Modifica del supporto in corso d'opera

La fase di progettazione del supporto, come riportato nell'Introduzione, è stata variata, rispetto all'idea originale, a causa dell'impossibilità di utilizzare la stampante 3D. Trattandosi di un componente essenziale per lo svolgimento dei test di collaudo, è stata individuata una diversa soluzione utilizzando i mattoncini Lego.

"Nella mia esperienza universitaria sono stato coordinatore del team studentesco Mi Lego al Territorio. Non avendo accesso e non essendo autorizzato all'utilizzo
e la lavorazione di altri materiali necessari alla fabbricazione di un adeguato supporto, e non potendo attendere altro tempo, vista l'impossibilità di poter disporre
di attrezzature e permessi prima della data di discussione della tesi, complice la
determinazione nel voler collaudare il prototipo e includere tale fase nell'elaborato,
ho pensato di utilizzare i mattoncini Lego, gentilmente concessi in prestito dal team.
In questo modo, ho potuto includere nella mia tesi una delle esperienze più significative e caratterizzanti del mio percorso formativo al Politecnico di Torino, e una
delle passioni che mi accompagnano da sempre, le costruzioni Lego, sicuramente
tra i responsabili della mia ambizione a voler diventare Ingegnere. E ho potuto fare
anche un particolare omaggio al progetto ArduPilot, il quale nacque con la creazione
di un drone realizzato con mattoncini e controllore (Mindstorm) Lego[2][3][38]."

2.4 Configurazione Pixhawk

Prima di poter configurare il controllore di volo, è necessario:

- installare l'ultima versione disponibile del software Mission Planner[39] sul Notebook selezionato per lo sviluppo dell'intero progetto;
- collegare le periferiche essenziali.

L'installazione del software si svolge con la procedura classica comune alla maggior parte dei programmi Windows: si scarica e si avvia il file di installazione esecutivo (.exe), seguendo la procedura guidata di installazione wizard.

2.4.1 Collegamenti

Le periferiche devono essere collegate al controllore tramite le porte illustrate in figura 2.12. Il modulo GNSS secondario con bussola integrata *U-Blox Neo-M8N Dual Compass* comunica con il controllore tramite una porta di output GPS/I2C con connettore JST-GH da 6 pin. Questa soluzione è stata concepita per controllori aventi una porta di input di tipo GPS/Compass. Il Pixhawk non possiede questa specifica porta: devono essere analizzati i pinout del modulo GNSS per comprendere come realizzare il collegamento.



1 Radio control receiver input

signal

- 2 S.Bus output
- 3 Main outputs
- 4 Auxiliary outputs

Figura 2.12: Dettaglio dei connettori sul Pixhawk

	Pixha	wk	u-Blox		
Porta	Pin	Segnale	Porta	Pin	Segnale
GPS	1	VCC		1	VCC
	2	TX(OUT)		2	GND
	3	RX(IN)	GPS/I2C	3	SDA
	4	CAN2 TX	GP5/120	4	SLC
	5	CAN2 RX		5	RX
	6	GND		6	TX
12C	1	VCC			
	2	SCL			
	3	SDA			
	4	GND			

Tabella 2.3: Pinouts GNSS secondario

Si riportano in tabella 2.3 le piedinature di interesse per il collegamento del modulo GNSS secondario. Ogni contatto del modulo GNSS deve essere collegato al corrispondente contatto sul Pixhawk, fatta eccezione per i pin TX e RX: la sigla TX sta per trasmettitore, la sigla RX sta per ricevitore: un pin TX dovrà sempre essere collegato a un contatto RX e viceversa, per la natura del tipo di comunicazione. Note tutte le piedinature, il collegamento deve essere diviso nelle porte GPS e I2C, con connettori DF13 rispettivamente a 6 pin e 4 pin, separando i fili del cavo secondo lo schema riportato in figura 2.13.

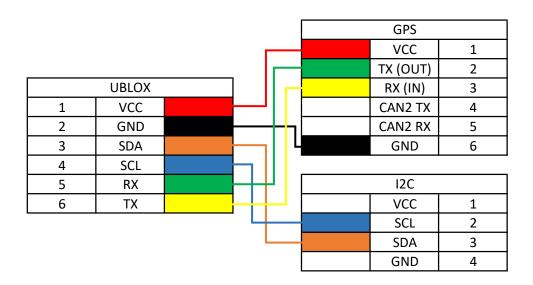


Figura 2.13: Collegamento modulo GNSS secondario - Pixhawk

La ricevente deve essere collegata, tramite il canale di SBUS, alla porta RC IN posizionata sul dorso superiore del controllore, che eroga anche l'alimentazione. Il collegamento si realizza per mezzo di un cavo per servomotori, avente connettore JR10 femmina a entrambe le estremità, a 3 pin: Alimentazione +, conduttore di massa GND - e segnale. La ricevente deve essere univocamente associata alla trasmittente tramite un'operazione chiamata binding:

- si accende la ricevente tenendo premuto il tasto F/S;
- si rilascia il tasto F/S (il led rosso della ricevente deve illuminarsi maniera intermittente);
- si accende la trasmittente e si attiva la modalità binding selezionando l'opzione dal menù (il led verde della ricevente deve accendersi e restare fisso);
- si spengono e riaccendono entrambi gli strumenti (il led rosso sulla ricevente deve risultare spento, il led verde acceso fisso).

Il servo motore di sterzo e il controllore di velocità (ESC) sono comandati tramite il proprio cavo a 3 pin, di cui generalmente è accessibile l'estremità che li collega alla ricevente. Il collegamento si realizza con terminale JR10 femmina. Solitamente, all'estremità del cavo collegata al veicolo, non vi è una legenda che permetta di identificare i diversi fili. Il verso di collegamento corretto è garantito dalla forma del connettore, il quale consente un unico verso. Inoltre, si può sempre fare affidamento alla colorazione convenzionale che ne identifica la destinazione:

- rosso: alimentazione in corrente continua, alimentazione, positivo + (rispetto a GND);
- nero: conduttore di massa GND;
- bianco: segnale di azionamento S.

Per consentire al controllore di comandare i motori, di devono collegare i terminali nei pin (canali o channels) presenti nel dorso superiore del Pixhawk:

- canale 1: servomotore di sterzo (steering servo);
- canale 3: controllore elettronico di velocità (Electronic Speed Control ESC)

Il **buzzer** deve essere collegato alla omonima porta BUZZER. È solitamente prodotto con cavo e connettore DF13 da 2 pin integrati: il suo collegamento è diretto, non richiede ulteriori operazioni. L'interruttore Safety Switch+LED deve essere collegato alla porta SWITCH, ed è sempre plug and play.

Infine, il collegamento del controllore di volo al **notebook** può essere realizzato tramite cavo USB o via telemetria. La connessione USB è contemporaneamente in

grado di alimentare il Pixhawk. La connessione via telemetria non è in grado di alimentare il controllore che, in questo caso, può essere alimentato in due modi: sempre tramite la porta Micro-USB, senza scambio di dati in questo caso, o tramite la porta POWER. In genere, è da preferirsi la modalità via telemetria: rappresenta la configurazione finale dell'UGV ed elimina il collegamento fisico con il veicolo, fattore limitante nelle operazioni.

Le due **radio di telemetria**, una da collegare al Pixhawk e l'altra da collegare al notebook, possiedono due porte: una Micro-USB e una DF13, come illutrato in figura 2.14. Nel Pixhawk, la radio deve essere collegata nella porta TELEM 1

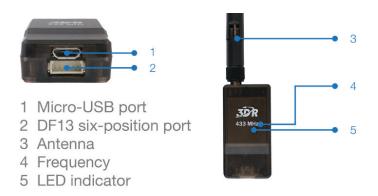


Figura 2.14: Radio di telemetria

tramite un cavo da 6 pin, avente connettore DF13 a entrambe le estremità. Nel notebook il collegamento si realizza tramite un cavo USB/Micro-USB. Il collegamento del modulo GNSS primario richiede un'analisi approfondita della documentazione tecnica fornita da Swift Navigation Inc., pertanto è rimandata al paragrafo 2.5.

2.4.2 Configurazione tramite Mission Planner

Il controllore di volo deve essere configurato tramite il software $Mission\ Planner\ (MP)[40]$. Una volta collegato il controllore al computer, il sistema operativo gli assegna automaticamente un numero di porta COM. Per avviare la comunicazione, si deve selezionare la corrispondente porta COM^{16} dal menù a tendina sito nella regione in alto a destra del programma MP e premere il pulsante $CONNECT^{17}$.

 $^{^{16}}$ Si consiglia di non collegare altre periferiche USB al computer durante la fase di connessione: in questo modo l'unica porta COM selezionabile sarà quella del Pixhawk.

¹⁷Sia per la connessione USB sia per la connesione via telemetria, MP rileva automaticamente la velocità di trasmissione dei dati (rispettivamente 115 200 BPS e 57 600 bps), la quale può essere selezionata anche automaticamente tramite un menù a tendina affiancato al menù delle porte.

Firmware

La prima operazione da compiere è l'installazione del **firmware**: si apre la scheda INITIAL SETUP, si seleziona l'opzione Install Firmware quindi, per configurare un UGV, si sceglie $ArduRover\ V4.0.0$. Mission Planner offre in automatico l'ultima versione stabile disponibile (dati aggiornati al 15 luglio 2020). Se si ritiene necessario, è possibile caricare un firmware specifico tramite il comando Load custom firmware, collocato nella stessa schermata.

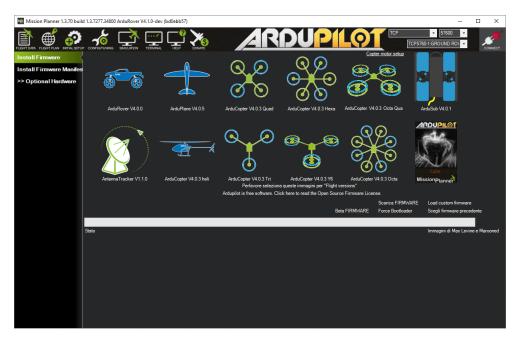


Figura 2.15: Schermata installazione Firmware

Durante l'installazione il programma chiede all'utente se si desidera caricare il sistema operativo ChibiOS. In questo progetto non si è considerata questa opzione in quanto si preferisce il sistema NuttX: ChibiOS è più leggero ma carente di alcune funzionalità. Successivamente devono essere effettuate tutte le calibrazioni delle periferiche: dalla scheda INITIAL SETUP si seleziona l'opzione Mandatory Hardware.

Accelerometro

L'accelerometro è un sensore essenziale per la guida autonoma, di conseguenza ArduPilot ne richiede obbligatoriamente la calibrazione, da effettuarsi con il veicolo non armato¹⁸. Dal menù Accel Calibration si esegue il comando Calibrate

¹⁸Nella pratica RC, con il termine *armare* si indica l'operazione di attivazione dei motori.

Accel: si dovrà posizionare il veicolo (o i solo pixhawk qualora le dimensioni del veicolo rendano difficoltosa tale operazione) nelle diverse posizioni richieste dal programma, attendendo qualche secondo in una condizione di riposo tra una posizione e l'altra. Le posizioni di calibrazione sono illustrate in figura 2.16.



Figura 2.16: Posizioni di calibrazione dell'accelerometro

Bussola

Un UGV necessità di almeno una bussola per poter navigare. Il Pixhawk ne possiede una interna e anche il modulo GNSS secondario possiede la propria.

Prima di cominciare la fase di calibrazione della bussola, ci si deve assicurare di essere distanti da oggetti che generino campi magnetici. Una volta accertata questa condizione, si apre il menù Compass e si esegue il comando Live Calibration: facendo riferimento a una barra di progresso, si deve ruotare il veicolo (o il controllore) in aria lentamente, facendogli assumere le posizioni illustrate in figura 2.16, e monitorare l'avanzamento della barra sino al completamento della procedura.

Trasmittente

Successivamente, si procede alla calibrazione della radio trasmittente, la quale permette il controllo remoto del veicolo e l'esecuzione di specifici comandi.

La fase di calibrazione prevede l'acquisizione dei valori massimo, minimo e trim ¹⁹ di ciascun canale radio, consentendo così ad ArduPilot di interpretare correttamente i segnali ricevuti. Per avviare la calibrazione si apre il menù Radio Calibration si esegue il comando Calibrate Radio. In questa schermata del programma, ogni canale viene rappresentato da una barra campita con due differenti colori (verde e grigio): quando una leva è impostata sul suo valore centrale 0, la barra deve risultare colorata di verde nella metà comprendente il valore minimo e grigia nell'altra metà. In funzione di come vengono mossi gli stick la barra verde aumenta o diminuisce: si muovono tutte le leve dall'escursione minima alla massima, si registrano i valori massimo e minimo raggiunti quindi si conclude l'operazione di calibrazione. Generalmente i canali hanno escursione compresa tra il valore minimo di circa 1100 e il valore massimo di circa 1900.

¹⁹Il termine ingelse *trim* si traduce con il verbo *regolare*: si riferisce alla possibilità di correggere il valore zero di ogni stick(manopola di direzione), qualora non fosse correttamente centrato.

Sterzo e motore

Avendo collegato lo sterzo nel canale 1 e il motore nel canale 3, ci si deve assicurare che il controllore abbia impostata la medesima associazione. Nel menù Servo Output è possibile monitorare gli 8 canali posizionati sul dorso superiore del Pixhawk. Selezionando il menù a tendina del canale 1, si deve impostare la voce GoundSteering. Nel canale 3 deve essere impostata la voce Throttle. È buona norma verificare la corretta associazione eseguendo i test dei motori: si deve selezionare l'opzione Optional Hardware quindi aprire il menù Motor Test ed eseguire Test motor A per verificare il funzionamento del motore di trazione e Test motor B per verificare il servomotore di sterzo.

Failsafe

Con il termine Failsafe si indica una procedura di sicurezza tramite la quale si protegge un sistema in seguito alla manifestazione di un determinato problema. Più precisamente, in funzione del sistema che si sta progettando, si individua una lista di problemi potenzialmente verificabili, e per ognuno di essi si determina una specifica serie di azioni da intraprendere una volta riconosciuta una di queste condizioni.

Nel caso di un ArduPilot UGV si configurano i filesafe da rispettare nelle missioni autonome. Dal menù Failsafe è possibile attivare i filesafe per la perdità del segnale dalla trasmittente, per lo stato della batteria e per la perdità di comunicazione con la GCS. Per attivare il filesafe della **trasmittente**, si deve impostare il parametro FS_THR_ENABLE al valore 1. Il sistema di sicurezza si attiva quando, per un lasso di tempo pari al numero di secondi impostati nel parametro FS_TIMEOUT, si verifica una delle seguenti condizioni:

- la ricevente non manda alcun segnale al controllore;
- il valore del canale 3 (Throttle) scende al di sotto del valore impostato nel parametro FS_THR_VALUE.

In caso di attivazione di questo filesafe, il veicolo compie l'azione corrispondente al valore del parametro FS_ACTION: quando impostato su 1 il *Rover* torna alla posizione di partenza compiendo un percorso diretto, quando impostato su 2, 3 o 4 il veicolo tenterà di tornare alla posizione di partenza secondo un percorso sicuro, basato sul tragitto compiuto. Il filesafe della **batteria** si attiva in due situazioni:

• la tensione è inferiore al valore impostato nel parametro BATT_LOW_MAH²⁰ per un tempo superiore a 10 secondi;

²⁰Nelle versioni firmware precedenti la nomenclatura di alcuni parametri può essere differente.

• la capacità rimanente (carica) della batteria scende al di sotto del valore impostato nel parametro BATT_LOW_MAH.

In caso di attivazione di questo filesafe, il veicolo compierà seguenti azioni:

- il buzzer emetterà un suono corrispondete a questo specifico problema²¹;
- il LED lampeggerà di colore giallo;
- verrà visualizzato un messaggio di avviso nella GCS;
- verrà svolta l'azione corrispondente al valore del parametro BATT_FS_LOW_ACT:
 per il valore 0 non viene compiuta alcuna azione, per il valore 1 il veicolo tornerà alla posizione di partenza, per il valore 2 il veicolo si arresta con lo sterzo
 puntato in avanti, per i valori 3 e 4 tenterà di tornare alla posizione di partenza secondo un percorso sicuro, infine per il valore 5 verranno disarmati i
 motori.

2.5 Configurazione Piksi Multi

Per integrare il modulo GNSS ad alte prestazioni nel UGV, si consulta la guida di supporto realizzata da Swift Navigation Inc. [36].

Il procedimento prevede:

- la realizzazione del cavo di collegamento;
- la configurazione dei parametri del Piksi;
- la configurazione dei parametri del Pixhawk.

2.5.1 Cavo artigianale

Il modulo GNSS Piksi Multi deve comunicare con il controllore tramite una porta di output GPS/I2C con connettore TLE-110-01-G-DV-A da 20 pin. Il Pixhawk non possiede una porta che permetta un collegamento diretto tramite un cavo da 20 pin: deve essere utilizzata la porta SERIAL 4/5²² con connettore DF13 da 6 pin.

Lo scambio di dati avviene tramite i contatti UART1, pertanto devono essere essere analizzati i pinout di entrambe le porte per comprendere come realizzare il

 $^{^{21} \}text{Il}$ buzzer produce suoni associati a specifici messaggi, la documentazione è consultabile al sito $\texttt{https://docs.px4.io/v1.9.0/en/getting_started/tunes.html}$

²²Questa porta include i contatti di SERIAL 4 (GPS2) e SERIAL 5 (DEBUG) in funzione di come viene configurata.

collegamento. Si riportano le piedinature del connettore 20 pin del Piksi e della porta SERIAL 4/5 del Pixhawk in tabella 2.4. Note le piedinature, si deve procedere alla

		-					
				Piksi Multi			
			Pin	Segnale	Segnale	Pin	
Pi	xhawk	-	1	VIN	2	NC	
Pin	Segnale		3	USBO -	4	USBO +	
1	VCC		5	/RESETIN	6	CAN O RX	
2	TX(#4)		7	CANO TX	8	CAN1 RX	
3	RX(#4)		9	EVENT IN	10	GND	
4	TX(#5)		11	UARTO TX	12	UARTO RX	
5	RX(#5)		13	DIG	14	UART1 TX	
6	GND		15	UART1 RX	16	GND	
			17	PV	18	GND	
			19	PPS	20	CAN1 TX	

Tabella 2.4: Pinouts GNSS primario

modifica di un cavo 20 pin con connettore TLE-110-01-G-DV-A ad almeno una delle estremità. Tagliando l'estremità lontana dal connettore citato (qualora presente solo a un capo), si devono individuare i fili di interesse e collegarli, per mezzo di saldatura, con i corrispondenti fili di un secondo cavo 6 pin con connettore DF13. I fili non utilizzati risulteranno spaiati. In figura 2.17 è illustrato uno schema del collegamento finale. L'ultimo collegamento fisico da realizzare riguarda l'antenna elicoidale, realizzato tramite il cavo presente nella dotazione Piksi.

2.5.2 Configurazione UGV Piksi

Il funzionamento del modulo Piksi da installare sul *Rover* è assicurato dalla corretta configurazione del modulo stesso, tramite il software Swift Console, e della porta SERIAL 4/5 del Pixhawk a cui sarà collegato, tramite il software Mission Planner.

Swift Console

Per essere configurato, il modulo GNSS deve essere collegato al notebook. È necessario inserire il modulo Piksi nella Evaluation Board: è quest'ultima a comunicare direttamente con il computer, dalla porta RS232 1, tramite un cavo USB/RS-232. Realizzato il collegamento, si alimenta la Evaluation Board con l'alimentatore incluso nel kit, quindi si avvia il programma Swift Console, il quale chiederà inizialmente a quale dispositivo ci si vuole connettere. Una volta collegato il Piksi al computer,

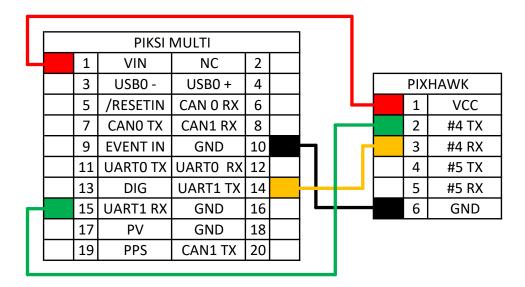


Figura 2.17: Collegamento modulo GNSS primario - Pixhawk

il sistema operativo gli assegna automaticamente un numero di porta COM. Per avviare la comunicazione, si deve selezionare la corrispondente porta COM²³ dal menù a tendina Serial Device, quindi premere OK. La velocità di trasmissione dei dati si imposta pari al valore predefinito: si sceglie 115 200 bps nel menù Baudrate. Successivamente si seleziona la scheda Settings e si impostano i parametri, come riportato in tabella 2.5, quindi si esegue il comando Save to Flash.

Sezione	Parametro	Valore	Descrizione
solution	soln freq	5	Impostazione della frequenza di output a 5 Hz.
solution	output every n obs	1	Genera una soluzione per ogni osservazione.
uart1	enabled sbp messages	65535, 258, 526, 522, 520, 257	Messaggi SBP utilizzati da Ardupilot ²⁴ .

Tabella 2.5: Parametri per le osservazioni del Piksi - UGV

 $^{^{23}}$ Si consiglia di non collegare altre periferiche USB al computer durante la fase di connessione: in questo modo l'unica porta COM selezionabile sarà quella del Piksi.

Mission Planner

Per utilizzare le informazioni di posizionamento del modulo Piksi devono essere impostati gli opportuni parametri, nel Pixhawk, tramite Mission Planner.

Si collega il controllore al computer con le modalità descritte al paragrafo 2.4.2. Dalla scheda CONFIG/TUNING si seleziona il menù Full Parameter List e si impostano i valori dei parametri riportati in tabella 2.6.

Parametro	Valore	Descrizione
EK2_ALT_SOURCE	2	ArduPilot usa il dato di altitudine calcolato dal Piksi per il filtro di Kalman esteso (EKF2): stima tutti i parametri di stato del veicolo.
GPS_AUTO_SWITCH	1	Il Pixhawk utilizza il modulo GNSS che fornisce la posizione con maggior precisione (descritta dagli indici DOP) ²⁵ .
GPS_SBP_LOGMASK	-1	Salva tutti i messaggi SBP nella memoria del Pixhawk (dataflash logs on dataflash memory).
GPS_TYPE	1	Rilevamento del modulo GNSS collegato al- la porta GPS automatico (GNSS secondario in questo caso).
GPS_TYPE2	1	Rilevamento del modulo GNSS collegato alla porta SERIAL 4/5 automatico (GNSS primario in questo caso).

Tabella 2.6: Parametri Pixhawk per corretta comunicazione con Piksi Multi

Radio di telemetria

L'impostazione predefinita per le comunicazioni via telemetria con ArduPilot non è ottimizzata per l'invio delle correzioni RTK: possono verificarsi notevoli perdite di dati e latenze elevate tali per cui la precisione del posizionamento diminuisce. Questo fenomeno dipende direttamente dalle impostazioni delle radio di telemetria. Senza connettere il Pixhawk a Mission Planner, si accende il UGV e si collega la

²⁴L'impostazione di default abilita una serie di messaggi che include anche quelli indispensabili per ArduPilot, quelli ulteriori possono essere utili per applicazioni di post processamento.

²⁵Se si imposta il parametro GPS_AUTO_SWITCH al valore 0 si utilizza il Piksi solo per *logging*: vengono salvate le osservazioni nella memoria, il posizionamento avviene con il GNSS secondario

radio di telemetria della stazione base al notebook. Dalla scheda INITIAL SETUP si seleziona il menù Optional Hardware quindi si apre il sottomenù Sik Radio: il programma mostra le impostazioni relative alla radio Local (quella della stazione base, connessa al notebook) e Remote (quella del UGV). Per entrambe le radio si deve: disattivare l'errore di correzione del codice ECC e selezionare la voce RawData dal menù a tendina Mavlink. In particolare, l'ultima impostazione ottimizza la trasmissione per un determinato tipo di pacchetto. I dati grezzi (RawData) non subiscono alcun tipo di controllo, perciò sono trasmessi nel minor lasso di tempo possibile. Saranno direttamente processati dal Piksi Multi. Si conclude la procedura eseguendo il comando Save Settings.

2.5.3 Configurazione Piksi stazione base

La prima operazione da eseguire è l'installazione dell'antenna, in dotazione nell'evaluation kit, per il modulo Piksi Multi collegato al notebook.

È di fondamentale importanza il suo posizionamento: deve essere installata lontano dal computer, per evitare che il segnale venga alterato da interferenze elettromagnetiche, e deve avere una vista pulita del cielo.

Swift Console

Il Piksi della stazione base dovrà essere sempre inserito nella evaluation board. Si collega il modulo al computer con un cavo USB/RS-232 e si avvia la comunicazione con il software Swift Console con le stesse modalità illustrate al paragrafo precedente.

Selezionando la scheda Settings, si impostano i parametri come riportato in tabella 2.7. Per poter disporre di una posizione rilevata della stazione base sem-

Sezione	Parametro	Valore	Descrizione
solution	soln freq	5	Impostazione della frequenza di output a 5 Hz.
solution	output every n obs	5	Genera una soluzione ogni 5 osservazioni.
surveyed position	broadcast	True	Abilita la trasmissione della posizione rilevata della stazione base.

Tabella 2.7: Parametri per le osservazioni del Piksi - stazione base

pre aggiornata, è necessario eseguire il comando Auto Survey, il quale compare

una volta selezionato il parametro broadcast, e salvare l'impostazione eseguendo, successivamente, il comando Save to Flash.

Le correzioni RTK devono essere ricevute dal UGV via telemetria, quindi devono prima essere inviate al software Mission Planner con protocollo UDP.

Selezionando la scheda Advanced, si accede alla sezione dedicata dalla sottoscheda Networking, quindi si impostano i parametri, come riportato in tabella 2.8. Si conclude la procedura eseguendo il comando Start. In questo modo, le osserva-

Parametro	Valore
Messages to broadcast IP Address Port	Observations 127.0.0.1 13320

Tabella 2.8: Parametri per l'invio delle correzioni RTK dal Piksi - stazione base

zioni vengono inviate su una porta virtuale del notebook: l'indirizzo IP 127.0.0.1 identifica il proprio computer, rappresenta l'entità definita localhost. In ambito informatico, il numero di porta rappresenta un punto (nodo) di comunicazione, chiamato communication endpoints. Le porte sono rappresentate da numeri interi a 16 bit compresi tra 0 e 65535 e identificano un processo specifico o un servizio di rete. I pacchetti appartenenti a una connessione sono identificati da 4 valori:

[<indirizzo IP sorgente>, <indirizzo IP destinazione>,
 <porta sorgente>, <porta destinazione>]

La porta 13320 fa parte delle *porte registrate*, utilizzate come riferimento fra applicazioni, non assegnate. Questo specifico numero è il valore predefinito per la ricezione delle correzioni RTK su Mission Planner, pertanto è ragionevole non sceglierne di diversi, si è sicuri del suo funzionamento.

Mission Planner

Per inoltrare le correzioni RTK nella telemetria si deve configurare la sezione dedicata nel software Mission Planner. Si connette il Pixhawk con le modalità descritte al paragrafo 2.4.2. Dalla scheda Initial Setup si seleziona il menù Optional Hardware e si apre il sottomenù RTK/GPS Inject: il programma mostra le impostazioni relative al tipo di sorgente. Si seleziona la sorgente UDP Host dal primo menù a tendina, si abilità la casella Inject MSG Type e quindi si esegue il comando Connect: si apre una finestra in cui deve essere digitato il numero di porta scelto come sorgente, 13320, da confermare eseguendo il comando OK. Completata la procedura, nella sezione Link Status i valori di dati ricevuti Input data rate e dati inviati Output data rate sono diversi da 0, a conferma del corretto invio delle correzioni RTK.

Comportamento atteso

Per interpretare il comportamento del Piksi Multi ci si può affidare allo stato dei led, illustrati in figura 2.18, descritto in figura 2.19. Nella configurazione operativa,

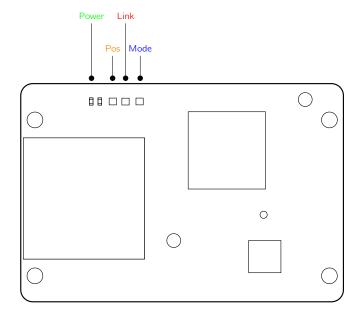


Figura 2.18: Led Piksi Multi

LED Name	Color	State	Description	
POWER	LED Off	Off No Power		
	Green	Continuously On	Module receiving power	
POS	LED Off	Off	No solution, antenna not detected, no GNSS signal received	
	Yellow	Slow Blink	No solution, Antenna detected, no GNSS signal received	
	Yellow	Fast Blink	No solution, GNSS signal received	
	Yellow	Continuously On	GNSS Solution Available (any kind)	
LINK	LED Off	Off	No incoming corrections or Internet access	
	Red	Flashing	Incoming corrections, no Internet access	
	Red	Continuously On	No Incoming corrections, Internet access	
	Red Continuously On Internet Access and incoming corrections		Internet Access and incoming corrections	
		and Flashing (occulting)		
MODE	LED Off	Off	No RTK	
	Blue	Blinking	Float RTK mode	
	Blue	Continuously On	Fixed RTK mode	

Figura 2.19: Significato stato led Piksi Multi

i led devono presentarsi nel seguente modo:

- POWER: verde fisso;
- POS: giallo fisso; LINK: rosso intermittente;
- MODE: blu fisso.

2.6 Costo e caratteristiche del prototipo realizzato

L'UGV realizzato ha un peso complessivo di $10 \,\mathrm{kg}$. In tabella 2.9 si riepilogano i costi sostenuti per l'acquisto di tutti i componenti implementati, IVA compresa.

Componente	Produttore	Modello	Costo	
Veicolo	Traxxas	X Maxx	1.000	€
Controllore di volo ricevitore GNSS ublox, buzzer, safety switch, radio di telemetria, cavi, microsd	mRobotics	Pixhawk Cool Kit	290	€
Ricevitore GNSS a elevate prestazioni	Swift Navigation	Piksi Multi Evaluation Kit	2.100	€
Ricevente	FrSky	X4R	50	€
Trasmittente	FrSky	Taranis X9D Plus	260	€
4 Batterie	FullPower	4S LiPo	300	€
Antenna elicoidale GNSS Multifrequenza	Emlid	Multi-band GNSS antenna	200	€
Supporto	Lego	Pick a brick	300	€
Totale			4.500	€

Tabella 2.9: Costi sostenuti per la realizzazione dell' UGV, IVA compresa.

Autonomia L'autonomia di funzionamento è direttamente dipendente dalla capacità delle batterie utilizzate, superiore all'ora intera di utilizzo impiegando due batterie a quattro celle (4S) da 3300 mAh. L'operazione di sostituzione delle batterie richiede pochi secondi, pertanto l'autonomia in missione, in definitiva, dipende dal numero di batterie di cui si dispone.

Carico trasportabile Il veicolo è capace di trainare oggetti aventi masse molto superiori alla propria. L'impossibilità di svolgere un'estesa campagna di indagini sperimentali, dovuta alle misure di contenimento adottate dal Politecnico di Torino durante l'emergenza *COVID-19*, rende necessario documentare tale abilità del veicolo mediante un metodo non scientifico, ma sufficientemente esaustivo.

La comunità di utenti riporta numerose testimonianze della capacità del veicolo di trainare persone, sedute su carrelli trainati o calzanti pattini, a velocità sostenute e pendenze superiori al 10 %. In particolare, un esperimento americano[27] dimostra che il Traxxas X Maxx riesce a trainare un uomo, di media statura e corporatura, su

pattini, in una pavimentazione piana, realizzata in conglomerato bituminoso, senza difficoltà, raggiungendo una velocità massima pari a 31 mph, equivalenti a 50 km/h. In queste condizioni il coefficiente di attrito è sicuramente il minore riscontrabile. Si considera poi il peso della persona trasportata superiore a 65 kg. Nello stesso studio[27] si dimostra la capacità del veicolo di trainare, sullo stesso suolo, un Pickup Toyota Tacoma, avente una massa a secco dichiarata di 1600 kg. Nella prova di traino dell'automobile, il Traxxas ha avuto notevoli difficoltà di aderenza: le sue ruote slittano con conseguente impossibilità di impostare la direzione desiderata. In conclusione, il risultato di questo test dimostra una capacità del veicolo molto superiore ai requisiti di questo progetto.

Un altro esperimento[9] dimostra come l'X Maxx sia capace di trasportare una persona del peso di 190 lb, equivalenti a 86 kg, in salita, sempre su una pavimentazione realizzata in conglomerato bituminoso, a velocità sostenuta.

In definitiva, si ritiene che le capacità dimostrate negli esperimenti citati coprano, con ampio margine, l'assenza di specifiche verifiche sulla capacità di trasporto di eventuali strumentazioni di rilievo, o altri carichi, permettendo di stabilirne l'adeguatezza. Si documenta di seguito il prototipo di UGV realizzato.



Figura 2.20: Prima realizzazione prototipo UGV

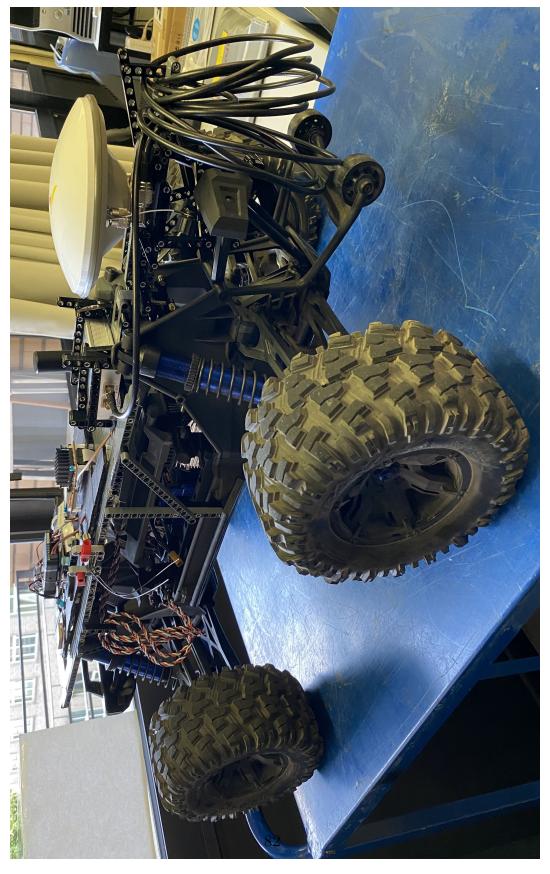


Figura 2.21: Prototipo UGV realizzato con $\mathit{Tersus}~BX316$

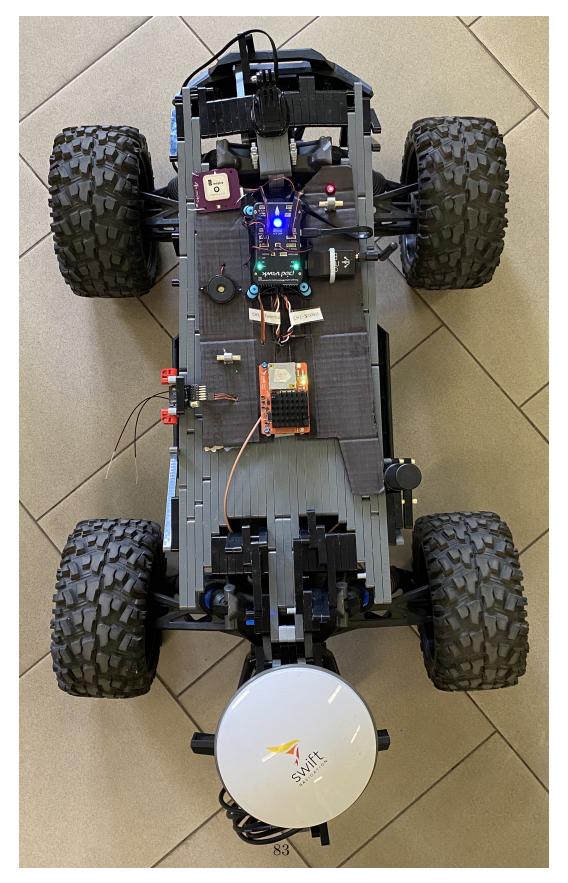


Figura 2.22: Dettaglio indicatori luminosi LED prototipo UGV realizzato

Capitolo 3

Validazione del Prototipo

Il funzionamento dell'UGV deve essere verificato tramite lo svolgimento di una $missione \ autonoma$. Per missione si intende lo svolgimento da parte del veicolo, in totale autonomia, delle istruzioni assegnate da un operatore in una precedente fase di pianificazione. Comprende il passaggio per determinati punti di riferimento, definiti tramite coordinate geografiche, e lo svolgimento di specifiche manovre.

Per ritenere affidabile il collaudo, è necessario stabilire il **contesto** e lo **scopo** di una missione che sia rappresentativa dello scenario operativo per il quale è stato creato il Rover: località devastate da un terremoto. Questi due fattori influenzano in diverso modo lo svolgimento della missione.

Contesto

A seguito di un evento sismico di intensità notevole, le strade possono subire ingenti danni, tra cui:

- piano stradale sconnesso;
- presenza di voragini;
- presenza di macerie.

Questi fattori influenzano significativamente i parametri di guida del veicolo, i quali devono essere tali da assicurare stabilità e sicurezza:

- velocità: deve essere contenuta entro bassi valori per poter assorbire dolcemente le irregolarità del terreno;
- velocità di percorrenza delle curve: deve essere bassa per generare bassi valori di accelerazione laterale;
- accelerazione nel verso del moto: deve essere contenuta entro bassi valori sia in positivo sia in negativo (frenata).

È inoltre necessario individuare un percorso sicuro, a seguito di accertamenti ricavati con ispezioni da svolgersi in sicurezza. Questo tema è illustrato nel dettaglio al paragrafo 3.2.

Scopo

La versatilità degli UGV permette anche l'installazione di strumenti per il rilievo topografico, utilizzabili senza essere messi in comunicazione con il controllore di volo. Di conseguenza, il Rover può essere utilizzato principalmente in due modi.

Per svolgere operazioni durante la guida Lungo il tragitto percorso dal Rover possono essere svolti rilievi di vario tipo, a seconda delle necessità. Si tratta quindi di una acquisizione di dati in movimento, i parametri di guida sono condizionati dalle modalità di rilievo e dalla precisione che deve essere raggiunta. In generale, deve essere garantito il rispetto della traiettoria pianificata entro minime tolleranze.

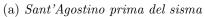
Per trasportare la strumentazione Sul veicolo possono essere installati specifici strumenti di misura necessari per svolgere operazioni di rilievo statico. Il Rover può trasportare la strumentazione in un determinato punto, stazionarvi per il tempo necessario quindi ritornare alla base. I parametri di guida sono influenzati dalla delicatezza degli strumenti trasportati, il rispetto della traiettoria non è severo come nel caso precedente.

3.1 Scenario di intervento

Si pianifica una missione nella città di Amatrice, ipotizzando di poter intervenire nelle giornate immediatamente successive al terremoto verificatosi il 24 agosto 2016. A seguito del sisma si sono verificati ingenti danni, concentrati nel centro storico del comune. In particolare, le chiese di Sant'Agostino e San Francesco si trovavano in uno stato di parziale collasso. Pertanto era necessario un tempestivo intervento al fine di poterne valutare i danni e la messa in sicurezza, e di prevenire ulteriori danneggiamenti, garantirne la conservazione e il successivo restauro, in quanto monumenti di importanza storica. Una rappresentazione è illustrata in figura 3.1.

Lo scopo della missione è l'ottenimento di un rilievo tridimensionale, da realizzare con un laser-scanner *LIDAR*, della facciata della chiesa. Il veicolo dovrà quindi trasportare l'attrezzatura sino al punto definito per la scansione, stazionare per il tempo necessario per effettuare la scansione (comandata da remoto) quindi tornare al punto di partenza. Per entrambe le chiese, si ipotizza di far partire il Rover dal centro operativo comunale (COC). Entrambi i percorsi che collegano le chiese con il COC prevedono il passaggio per strade strette, che si intersecano con angoli di 90°, e la cui sede stradale risulta danneggiata, con presenza di macerie. Pertanto, occorre definire percorso, manovre e parametri di guida del Rover molto prudenziali.







 ${\rm (b)}\ Sant'Agostino\ dopo\ del\ sisma$



(c) San Francesco prima del sisma



(d) San Francesco dopo del sisma

Figura 3.1: Chiese di Sant'Agostino e San Francesco ad Amatrice a seguito del terremoto del 24 agosto $2016\,$

3.2 Pianificazione della missione tramite Mission Planner

Come illustrato al paragrafo 2.1.4, il software selezionato per lo svolgimento di questo progetto è *Mission Planner (MP)*. Più precisamente, la configurazione della missione si compone di due procedure distinte: la configurazione dei parametri di guida e il disegno della traiettoria.

La prima fase da affrontare riguarda la definizione dei parametri di guida globali, da cui dipende il comportamento del veicolo, a prescindere la percorso.

Successivamente si procede con il disegno della traiettoria, la quale viene generata automaticamente dal programma mediante l'unione diretta (segmenti rettilinei) di punti di riferimento, chiamati waypoint, definiti tramite coordinate geografiche: latitudine, longitudine e altitudine.

Il rispetto della traiettoria però non è garantito a priori: un'errata configurazione dei parametri di guida può rendere impossibile il rispetto della traiettoria. Questa tematica è affrontata nel dettaglio al paragrafo 3.2.1.

Di conseguenza, concluse le prime due fasi, segue una prima simulazione della missione, nella quale si osserva il comportamento del veicolo. Al seguito, si comprende quali parametri devono essere affinati per ottener il miglior rispetto della traiettoria pianificata e si valutano le modifiche da apportare al percorso.

Può risultare necessaria la variazione di alcuni parametri globali, o una modifica locale: in diverse zone del percorso possono insorgere particolari necessità, dovute alla geometria del tracciato o alla presenza di ostacoli, tali per cui le impostazioni globali non assicurano il rispetto della traiettoria. In questi casi, spesso non è giustificato modificare il comportamento dell'UGV per l'intero percorso: certe limitazioni influiscono negativamente sull'efficienza della missione. Questa tematica è affrontata nel capitolo successivo, della discussione, 4.2.1.

La procedura si può ritenere conclusa quando la traiettoria percorsa soddisfa i requisti di scarto, massimo e medio, dal percorso pianificato imposti dal progettista, i quali sono definiti sulla base di scopo e necessità della missione.

Lo studio della simulazione è stato concentrato sulla traiettoria per la chiesa di San Francesco, in quanto, rispetto alla chiesa di San'Agostino, risulta più distante dal COC e richiede un percorso più articolato. Affinati i parametri per il tragitto più complesso, le impostazioni saranno valide anche per le situazioni più semplici.

3.2.1 Parametri di guida globali

Per impostare i parametri globali si apre la scheda CONFIG/TUNING e si accede al menù Basic Tuning, illustrato in figura 3.2. I parametri possono essere espressi da numeri interi o reali, utilizzando la virgola come separatore decimale. Di seguito sono riportati i criteri di scelta delle impostazioni di carattere globale.



Figura 3.2: Parametri di guida globali

Raggio dei waypoint

Durante una missione autonoma, il software, in esecuzione sul controllore di volo, impone all'UGV il passaggio per tutti i waypoint, determinando le manovre da compiere nel rispetto di una determinata precisione, definita a priori in fase di pianificazione. La scelta della posizione dei waypoint determina la geometria del percorso, ma non necessariamente il veicolo dovrà passare esattamente sopra il punto specificato. Il programma consente di stabilire il raggio di una circonferenza, centrata sul waypoint, la quale serve da vincolo per la navigazione: l'autopilota può iniziare la manovra di svolta solo una volta giunto all'interno di questo cerchio. Il parametro che descrive il raggio della circonferenza (WP_RADIUS) influisce chiaramente sulla determinazione del percorso globale, pertanto la sua definizione è compresa nelle impostazioni iniziali. Volendo ottenere una navigazione di precisione, si imposta un valore basso, non nullo, in modo da non vincolare troppo la traiettoria:

$$WP_RADIUS = 1,0$$

Valori elevati del parametro permetterebbero il calcolo di percorsi molto distanti dalla traiettoria di riferimento, comportando grandi scostamenti, non ammessi dal contesto della missione: non si riuscirebbe a evitare ostacoli e edifici.

Raggio di sterzata

Il raggio di sterzata minimo si determina in funzione del massimo angolo di sterzo che si impone all'autopilota per il calcolo della traiettoria. Considerato un angolo di sterzata massimo per il Traxxas X Maxx pari a $\theta=35^{\circ}$, si calcola il corrispondente raggio di sterzata minimo:

$$raggio = \frac{interasse}{tg(\theta)} = \frac{0.48 \text{ m}}{tg(35^{\circ})} = 0.69 \text{ m}$$

Si imposta il parametro, espresso in m, a un valore leggermente superiore, in modo tale da avere un margine qualora, in eventuali impreviste circostanze, il veicolo non riesca a raggiungere l'angolo di sterzata massimo:

$$TURN_RADIUS = 1,0$$

Velocità di percorrenza

La velocità di percorrenza dell'intero tragitto deve essere contenuta, in modo da poter ritenere la guida sicura e stabile, per proteggere sia la strumentazione di rilievo trasportata sia il veicolo stesso. Inoltre, devono essere tenute in considerazione le condizioni del tracciato, tra cui il terreno dissestato, che aumenta il grado di instabilità durante la guida. Infine, una velocità elevata produce un comportamento ingovernabile, con ritardi in frenata e sbandamenti: l'autopilota cercherà sempre di passare per ogni waypoint alla velocità massima, anche in corrispondenza dei punti che determinano un cambio significativo di traiettoria. In questo modo il veicolo non sarebbe capace di effettuare correttamente le svolte, oltrepasserebbe costantemente i waypoint delle curve per poi svolgere manovre impreviste per reindirizzarsi verso il waypoint successivo. Di conseguenza, il parametro, espresso in m/s, si imposta un valore massimo equivalente a 22 km/h:

$$WP_SPEED = 6,0$$

Velocità di sterzata

La velocità di sterzata massima deve essere impostata a un valore non eccessivo per evitare brusche sterzate, le quali potrebbero destabilizzare il veicolo e provocare danni allo stesso. Il parametro, espresso in deg/s, si imposta al valore:

Accelerazione

Il veicolo utilizzato per la creazione di questo UGV è molto potente. Si deve evitare che l'autopilota svolga brusche accelerazioni in quanto provocherebbero slittamenti, i quali ridurrebbero stabilità, sicurezza e efficienza del veicolo in missione.

Di conseguenza, si limita la massima potenza di accelerazione, espressa in percentuale % al minor valore consentito:

 $MOT_THR_MAX = 30$

3.2.2 Disegno della traiettoria

Per accedere alla sezione di pianificazione della missione si seleziona la scheda FLIGHT PLAN: viene mostrata una mappa di background e l'elenco dei waypoint (WP), come illustrato in figura 3.3. Nello stesso elenco, si definiscono manovre e



Figura 3.3: Definizione della traiettoria mediante waypoint

azioni specifiche da far compiere all'UGV a seguito del passaggio in uno specifico waypoint. Generalmente, le coordinate dei waypoint si determinano mediante inserimento diretto nella mappa di background: cliccando con il tasto sinistro del mouse nella posizione desiderata, MP salva il WP riportandone automaticamente le coordinate nell'elenco. Qualora si renda necessario, è possibile la definizione manuale dei WP, digitandone le coordinate negli appositi campi in elenco. E allo stesso modo è possibile modificare le coordinate di un punto precedentemente inserito.

Considerato il danneggiamento della sede stradale e la presenza di macerie generati dal sisma, in questo particolare scenario sarebbe incauto fare affidamento alla sola mappa satellitare di background: difficilmente risulterà aggiornata, di conseguenza non mostra le condizioni attuali delle strade. Inoltre, in queste situazioni le operazioni di soccorso sono continue e comportano lo spostamento di macerie: le condizioni delle strade variano in continuazione. Pertanto, si rende necessario

disporre di un'immagine aerea che sia il più recente possibile rispetto al momento di avvio della missione. Una soluzione molto pratica prevede la collaborazione con un'altra tipologia di drone: un UAV. È possibile programmare un UAV per compiere volo autonomo dal quale ricavare foto aeree ad alta risoluzione. Affiancando l'immagine ricavata dal drone alla mappa satellitare di background, è possibile individuare un percorso sicuro e impostare il comportamento del Rover in prossimità di punti critici, riducendo ulteriormente la velocità e infittendo i WP, così da descrivere traiettorie adatte all'aggiramento di ostacoli. Inoltre, l'UAV può affiancare il



Figura 3.4: Ingrandimento di una ripresa aerea, ottenuta da quadricottero, utilizzata per lo studio del percorso

rover anche durante la missione, monitorando l'avanzamento e offrendo un punto di vista alternativo, utile in caso si manifestassero problematiche. Il monitoraggio può avvenire tramite guida sia manuale sia autonoma, sfruttando nel secondo caso le tecnologie di *Active Tracking*.

Scelta iniziale dei Waypoint

Nella prima fase di definizione della traiettoria, deve essere scelto il minor numero di punti possibile: la priorità iniziale riguarda la comprensione del comportamento del veicolo. Le traiettorie semplici mostrano meglio la risposta dell'autopilota ai parametri di guida globali. Diversamente, un percorso articolato richiama leggi di

guida autonoma che possono sovrastare le impostazioni globali. Il primo percorso pianificato pertanto è stato realizzato inserendo un waypoint per ogni cambio direzione, come illustrato in figura 3.5.



Figura 3.5: Traiettoria iniziale per il raggiungimento della chiesa di San Francesco

3.3 Ottimizzazione dei parametri di guida tramite simulazione

Al termine della configurazione iniziale, si esegue una prima simulazione e si valuta il risultato confrontando la traiettoria percorsa con quella pianificata, tramite analisi software. Per comparare i percorsi è stato scelto il programma *CloudCompare*¹: un software open source, in accordo con la filosofia del progetto, progettato per la comparazione di nuvole di punti 3D.

Il programma chiede in input due liste di punti, che possono avere 2 o 3 coordinate: la comparazione può essere svolta nel piano o nello spazio. In particolare, le distanze sono state calcolate utilizzando lo strumento Cloud2Cloud (C2C): un algoritmo che permette di calcolare la distanza assoluta tra punti vicini. La navigazione degli UGV è vincolata al terreno, pertanto si confrontano le coordinate piane.

¹La sua distribuzione e documentazione è contenuta nel sito https://www.danielgm.net/cc/.

3.3.1 Simulazione missione

Per avviare la simulazione, si apre la scheda SIMULATION e si seleziona la tipologia di veicolo Rover. Successivamente si aggiornano i parametri di guida individuati nella precedente fase (paragrafo 3.2.1) aprendo la scheda CONFIG/TUNING, selezionando il menù Basic Tuning e eseguendo il comando Scrivi Parametri. Si prosegue con la scrittura dei waypoint: è il procedimento con cui si carica la missione sul controllore. Si apre la scheda FLIGHT PLAN, si carica la lista di punti (precedentemente salvata) con il comando Load WP File quindi si trasferisce la lista al $Pixhawk^2$ eseguendo il comando Scrivi i WPs.

Per eseguire la missione si apre la scheda FLIGHT DATA, si seleziona il menù Azioni, si arma il veicolo, tramite il comando Arma/Disarma e si avvia la missione eseguendo il comando AUTO. Conclusa la missione, il programma consente un primo confronto visivo, in quanto disegna la traccia del percorso simulato con una linea di colore viola (lo stesso del disegno rappresentante l'UGV). Una rappresentazione è riportata in figura 3.6.

Da una prima analisi qualitativa, è possibile osservare e comprendere alcune problematiche relative all'impostazione dei parametri di guida, ma una valutazione più accurata deve essere svolta effettuando un confronto numerico.



Figura 3.6: Risultato della prima simulazione

²Nell'intera fase di simulazione, il controllore di volo è sempre virtualizzato dal simulatore SITL.

3.3.2 Trattamento dei dati per la comparazione

Per procedere al confronto numerico si deve disporre di due file separati, contenenti le liste di punti che descrivono le traiettorie rispettivamente percorsa e pianficata.

Traiettoria percorsa

Per procedere al confronto numerico, si scaricano i Dataflash logs: contengono tutti i dati registrati dal controllore di volo durante la missione, tra cui le coordinate GNSS delle posizioni registrate. Dal menù DataFlash Logs si esegue il comando Download DataFlash Log Via Mavlink: selezionando il log desiderato viene generato un file .BIN. Per estrarre le coordinate GPS si genera un file .mat eseguendo l'apposito comando Create Matlab File e selezionando il log appena generato. I file .mat possono essere aperti con programma MATLAB o con l'alternativa open source FreeMat³.

In quanto la procedura prevede la realizzazione di molteplici simulazioni per valutare miglioramenti o peggioramenti conseguenti alla modifica di un parametro, è stato creato uno script .mat capace di aprire il file generato da Mission Planner, leggere le coordinate di Latitudine e Longitudine di ogni posizione registrata con il modulo GNSS quindi scriverle in un file .txt da collocare in una specifica cartella contenente gli output.

I punti però sono espressi in coordinate angolari, le quali non sono adatte per la comparazione. Pertanto devono essere convertite in coordinate metriche impiegando algoritmi di proiezione. Per questa procedura è stato utilizzato il software gratuito $ConveRgo^4$, il quale ha consentito la conversione nel sistema di riferimento UTM - WGS84. Per la corretta esecuzione degli algoritmi sono stati caricati i grigliati IGM dell'area di Amatrice. In figura 3.7 è riportata la schermata del software.

Traiettoria pianificata

La traiettoria pianificata è stata definita dall'unione di pochi punti, individuati sempre tramite coordinate angolari. Una traiettoria così descritta è inadatta alla comparazione con *CloudCompare*: la densità dei punti generata durante la navigazione è superiore alla lista di punti pianificata di due ordini di grandezza.

Per ottenere un confronto corretto la traiettoria di riferimento deve essere più densa della percorsa, in modo tale che ogni singolo punto della traiettoria percorsa sia compreso tra due punti successivi della pianificata.

³Per questo lavoro di tesi è stato utilizzato il software *MATLAB*, ma gli script generati sono completamente eseguibili anche dall'alternativa gratuita *FreeMat*, la cui distribuzione e documentazione è contenuta nel sito http://freemat.sourceforge.net/.

⁴La distribuzione e documentazione del programma *ConveRgo* è contenuta nel sito http://www.centrointerregionale-gis.it/

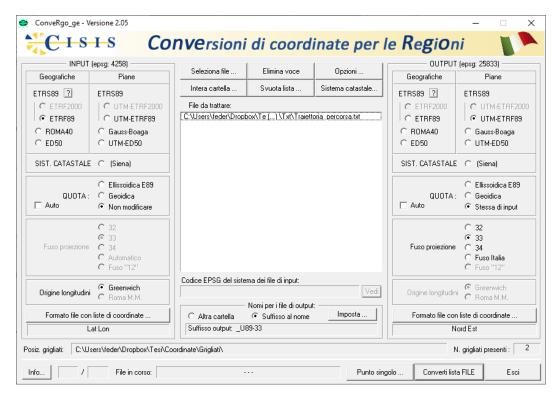


Figura 3.7: Conversione delle coordinate tramite ConveRgo

Per infittire il percorso pianificato è stato scritto uno script .mat, il quale calcola i punti intermedi per mezzo di interpolazione lineare, i quali saranno quindi disposti secondo un percorso rettilineo tra due punti successivi della traiettoria di riferimento. Questa operazione rispetta la geometria del percorso originale, il quale è generato sempre tramite segmenti rettilinei, e pertanto comprende sicuramente tutti i punti generati dallo script. L'elaborazione è stata eseguita direttamente nel sistema di riferimento UTM - WGS84, a seguito della conversione iniziale delle coordinate dei waypoint. Il risultato è riportato in figura 3.8.

Valutazione della prima simulazione

La prima missione ha evidenziato la notevole difficoltà dell'UGV nell'impostazione e mantenimento di una traiettoria rettilinea. Le problematiche sembrano essere relative alle impostazioni dello sterzo, pertanto la prima fase di affinamento riguarderà tali parametri. In tabella 3.1 sono riportati i risultati della comparazione numerica fra le due traiettorie.

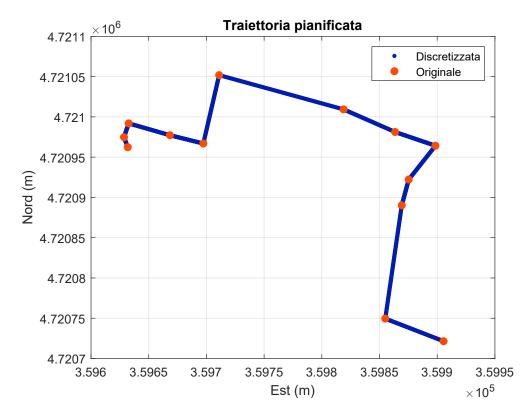


Figura 3.8: Discretizzazione della traiettoria pianificata

Parametro	Valore
Distanza massima	7.83 m
Distanza media	1.65 m
Deviazione standard	1.83 m

Tabella 3.1: Risultati della comparazione relativi alla prima simulazione

3.3.3 Affinamento della velocità di sterzata

Dall'analisi della prima traiettoria percorsa si nota la particolare difficoltà del veicolo nell'impostazione e percorrenza della curve. La svolta inizia sistematicamente oltre il punto di intersezione tra due segmenti rettilinei successivi. In una traiettoria correttamente percorsa, il punto di intersezione dovrebbe coincidere circa con il punto centrale della curva percorsa dal veicolo. Pertanto, l'UGV dovrebbe iniziare la svolta anticipatamente al punto di intersezione. Analizzando i possibili fattori causa di questo ritardo nella manovra di svolta, la velocità di sterzata appare essere la principale responsabile.

Il parametro impostato all'inizio, ACRO_TURN_RATE = 80, è sufficiente a garantire un'adeguata velocità di sterzata per una corretta percorrenza. Il problema quindi risiede in un altro parametro, il quale governa la variazione di velocità nell'operazione di sterzata: ArduRover regola il rapporto tra la velocità di sterzata desiderata ACRO_TURN_RATE e i segnali di output inviati al servomotore di sterzo tramite il parametro ATC_STR_RAT_FF, il quale ha un intervallo di valori assegnabili compreso tra un minimo di 0,0 e un massimo di 3,0.

Inizialmente il parametro è impostato a un valore di 0,2. Per l'affinamento, è stato gradualmente aumentato, incrementando il valore di 0,1 unità per ogni tentativo, sino al raggiungimento del valore ottimale:

$$ATC_STR_RAT_FF = 1,2$$

individuato considerando il miglior rispetto della traiettoria. In figura 3.9 è riportata l'evoluzione della traiettoria generata dalla variazione del parametro FF.

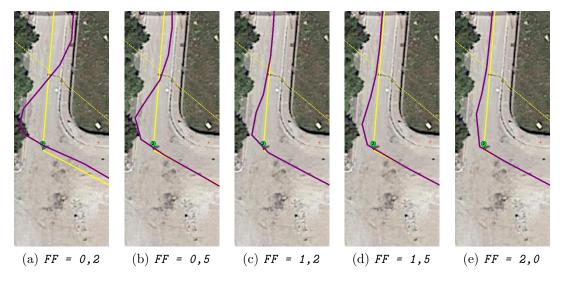


Figura 3.9: Confronto delle traiettorie percorse (viola) rispetto alla pianificata (giallo) al variare del parametro FF

In particolare, con bassi valori del parametro ATC_STR_RAT_FF (da qui in avanti abbreviato in FF) il veicolo si allinea alla traiettoria rettilinea compiendo più cambi di direzione. Al valore ottimale riesce ad allinearsi compiendo un'unica svolta e in uno spazio contenuto rispetto al punto di intersezione. Per valori elevati invece, si evidenzia la difficoltà del veicolo nell'allinearsi alla traiettoria rettilinea con precisione: risulta leggermente distaccato. Nonostante in questa ultima situazione gli scostamenti nell'intorno della curva risultino più contenuti rispetto al valore considerato ottimale, questi risultano comunque notevoli per essere considerati di precisione. Questo accade perché il fenomeno non è più legato al valore del parametro FF,

ma alla geometria della curva. Pertanto, la successiva fase di affinamento riguarda il disegno delle curve. In tabella 3.3 sono riportati i risultati della comparazione numerica calcolata sull'intero percorso:

Parametro	Valore
Distanza massima	$7.82\mathrm{m}$
Distanza media Deviazione standard	0.76 m 1.66 m

Tabella 3.2: Risultati della comparazione numerica relativi alla traiettoria percorsa a seguito dell'affinamento della velocità di sterzata

3.3.4 Affinamento della traiettoria in curva

Per contenere gli scostamenti in curva, la traiettoria deve presentare ovunque angoli ottusi di ampia estensione. Prima di ridisegnare le curve, occorre stabilire un criterio, ovvero un valore limite dell'angolo di intersezione, al di sotto del quale la traiettoria deve essere modificata. Dalle simulazioni effettuate, si stabilisce tale valore pari a 120°: angoli di ampiezza maggiore o uguale garantiscono scostamenti dalla traiettoria pianificata simili tra loro e contenuti. Di conseguenza, le curve con angoli inferiori a 120° sono state realizzate aggiungendo un waypoint in ingresso e uno in uscita. Nella nuova traiettoria l'angolo di minore ampiezza misura 120°.

Con questa modifica la percorrenza è diventata più graduale e la traiettoria percorsa è geometricamente più simile al percorso pianificato. Resta comunque uno scostamento non trascurabile nella seconda metà di curva, come è possibile notare in figura 3.10.

Per contenere gli scarti è necessario ridurre la velocità in curva. Si può operare seguendo due approcci differenti:

- modificando l'estensione della svolta
- riducendo la velocità in curva di una quantità definita

i quali sono trattati rispettivamente ai paragrafi 3.3.5 e 3.3.6.

In tabella 3.3 sono riportati i risultati della comparazione numerica calcolata sull'intero percorso:

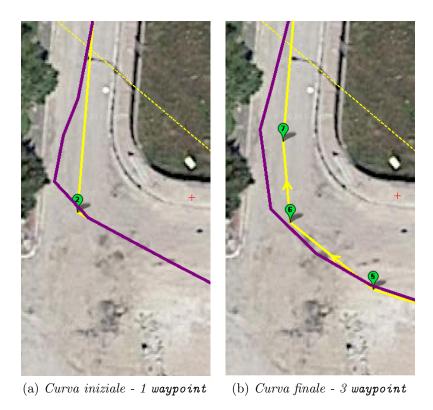


Figura 3.10: Modifica di una curva con aggiunta di waypoint in ingresso e uscita

Parametro	Valore
Distanza massima	7.81 m
Distanza media	0.90 m
Deviazione standard	1.91 m

Tabella 3.3: Risultati della comparazione relativi alla traiettoria percorsa disegnando le curve strette con 3 punti

3.3.5 Estensione della svolta

Il parametro WP_OVERSHOOT è specificamente utilizzato per far rallentare il veicolo prima del raggiungimento di un waypoint. In particolare, descrive la distanza massima, espressa in metri, che il veicolo può percorrere oltre il waypoint mentre termina la svolta. Per avere il maggior rispetto della traiettoria, il parametro deve essere impostato al suo valore minimo:

 $WP_OVERSHOOT = 0,0$

In questo modo il veicolo decelera sensibilmente al raggiungimento di ogni waypoint per impostare la traiettoria verso il punto successivo in uno spazio minimo. Si raggiunge un'elevata precisione nella navigazione, e lo scostamento dalla traiettoria si riduce molto. In figura 3.11 è illustrato il confronto sulla stessa curva analizzata nei precedenti paragrafi.

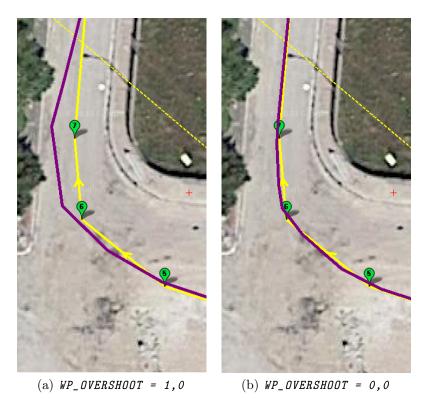


Figura 3.11: Confronto della traiettoria percorsa (viola) rispetto alla pianificata (giallo) al variare del parametro WP_OVERSHOOT

In tabella 3.4 sono riportati i risultati della comparazione numerica calcolata sull'intero percorso:

Parametro	Valore
Distanza massima	$3.67\mathrm{m}$
Distanza media	$0.14\mathrm{m}$
Deviazione standard	$0.66\mathrm{m}$

Tabella 3.4: Risultati della comparazione numerica relativi alla traiettoria percorsa con WP_OVERSHOOT = 0,0

3.3.6 Riduzione della velocità in curva

Un'alternativa alla riduzione automatica generata dal parametro WP_OVERSHOOT si realizza imponendo manualmente la velocità in ingresso curva e aumentandola in uscita. In presenza di curve strette la velocità di percorrenza deve essere ridotta al fine di contenere l'accelerazione laterale e evitare slittamenti o sbandamenti causati dalla perdita di aderenza. Se poi si considera la possibilità di installare delle strumentazioni di rilievo sull'UGV, queste andrebbero posizionate in un livello superiore al telaio e al piano hardware. Di conseguenza, considerando che le masse di molti strumenti non sono trascurabili 5 , il baricentro delle masse risulterebbe molto superiore rispetto alla configurazione originale del veicolo. E questo richiede basse velocità di percorrenza, in modo da non generare valori elevati di forza centrifuga. Per determinare la velocità da impostare nella percorrenza della curva è stata considerata dapprima la verifica al ribaltamento, la quale impone che il momento ribaltante deve essere inferiore al momento stabilizzante:

$$M_S \ge M_R \longrightarrow p \cdot b \ge F_c \cdot h \longrightarrow FS = \frac{F_p \cdot b}{F_c \cdot h} \ge 1$$

dove:

- F_c è la forza centrifuga
- \bullet h è l'altezza del baricentro delle masse
- p è la forza peso
- ullet è il braccio della forza peso, calcolato rispetto al polo di rotazione, ovvero il punto di contatto col terreno più esterno rispetto alla curva
- \bullet FS è il fattore di sicurezza

In figura 3.12 sono illustrate le forze a cui è soggetto il veicolo in curva 6 . Considerato il peso dell'UGV pari a 9.0 kg, immaginando di installarvi un Laser Scanner FARO CAM2 del peso di 4.2 kg, si considera una massa totale, approssimata per eccesso, a favore di sicurezza, pari a

$$m = 15.0 \, \text{kg}$$

Anche l'altezza del baricentro delle masse è stata determinata con un'approssimazione per eccesso a favore di sicurezza. Considerate le seguenti altezze:

⁵Un esempio classico è il laser scanner *FARO CAM2*, avente un peso di 4.2 kg, in dotazione, e utilizzato spesso, dal gruppo di ricerca del *Laboratorio Fotogrammetria*, *Geomatica e GIS* del *Politecnico di Torino*.

⁶La forza di attrito non è stata menzionata nella trattazione in quanto, avendo braccio nullo con il polo di rotazione, non apporta contributo nel fenomeno del ribaltamento.

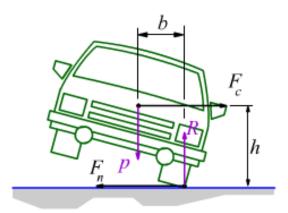


Figura 3.12: Equilibrio del veicolo in curva

- $\bullet \ h_1 = 0.35\,\mathrm{m}$ altezza del piano di appoggio del laser scanner dal suolo
- $h_2 = 0.19 \,\mathrm{m}$ altezza di un eventuale supporto (treppiede) compatto da fissare solidale al piano di appoggio
- $h_3 = 0.21 \,\mathrm{m}$ altezza totale dello strumento LIDAR

si considera la condizione idealmente più gravosa, per cui il baricentro delle masse dell'intero sistema coincide con il punto più alto del veicolo⁷:

$$h = h_1 + h_2 + h_3 = 0.75 \,\mathrm{m}$$

La forza peso p ha un valore costante, pari a:

$$p = m \cdot g = 15 \,\mathrm{kg} \cdot 9.81 \,\mathrm{m/s^2} \simeq 150 \,\mathrm{N}$$

dove:

- \bullet *m* è la massa totale del veicolo
- q è l'accelerazione di gravità

La forza centrifuga F_c invece varia in funzione del raggio della curva e della velocità di percorrenza:

$$F_c = \frac{m \cdot v^2}{r}$$

dove:

⁷È un'approssimazione eccessivamente cautelativa, giustificata dall'impossibilità di svolgere sperimentazioni comportata dalle misure di contenimento adottate dal Politecnico di Torino durante l'emergenza *COVID-19*: un baricentro più elevato non può esistere.

- m è la massa totale del veicolo
- \bullet v è la velocità di percorrenza in curva
- r è il raggio della curva

Si osservi come la dipendenza dalla velocità sia più influente, in quanto è proporzionale al quadrato della velocità, rispetto al raggio della curva, il quale compare alla prima potenza.

Di conseguenza, sono stati calcolati:

- velocità massima di percorrenza per una curva di raggio $r = 1 \,\mathrm{m}$;
- fattore di sicurezza FS per una curva di raggio r=1 m percorsa a una velocità v=1 m/s;
- raggio minimo delle curve per la velocità di percorrenza massima sull'intero percorso $v_{max}=6\,\mathrm{m/s}.$

Considerato il raggio di curvatura minimo assegnato all'autopilota, tramite il parametro TURN_RADIUS = 1,0, nella definizione dei parametri iniziali al paragrafo 3.2.1, si determina la velocità massima ammissibile curva imponendo:

- FS = 1
- $r = 1 \,\mathrm{m}$

la quale risulta:

$$v = 1.9 \, \text{m/s}$$

e genera una forza centrifuga pari a:

$$F_c = 54.2 \,\mathrm{N}$$

Per comprendere l'efficacia di questo approccio è stata effettuata una simulazione nella quale la velocità massima del veicolo in curva è stata imposta pari a 1.9 m. La condizione si realizza utilizzando il comando DO_CHANGE_SPEED, espresso in m/s, nella lista dei waypoint, modificando la velocità in ingresso e in uscita dalla curva. In tabella 3.5 è riportata la modifica relativa alla curva analizzata in tutta la procedura di affinamento, la cui traiettoria è confrontata in figura 3.13. In tabella 3.6 sono riportati i risultati della comparazione numerica calcolata sull'intero percorso. Rispetto all'approccio precedente sono diminuite la distanza media e la deviazione standard.

ID waypoint	Comando	Valore
:		
5	DO_CHANGE_SPEED	1,9
6	WAYPOINT	0
7	WAYPOINT	0
8	WAYPOINT	0
9	DO_CHANGE_SPEED	6,0
:		

Tabella 3.5: Modifica puntuale della velocità per curve strette

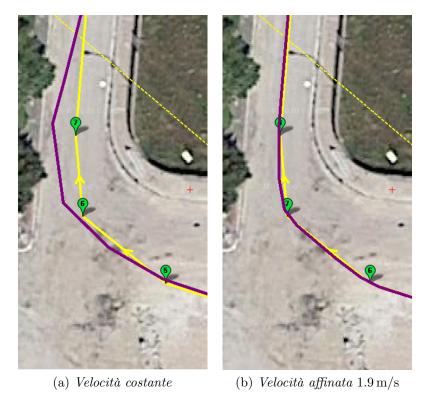


Figura 3.13: Confronto della traiettoria percorsa (viola) rispetto alla pianificata (giallo) al variare della velocità in curva

Affinamento finale

Si vuole ulteriormente ridurre la velocità di percorrenza in curva. Si può ritenere affidabile un fattore di sicurezza unitario qualora siano stati utilizzati ulteriori

Parametro	Valore
Distanza massima Distanza media	3.68 m 0.10 m
Deviazione standard	$0.10\mathrm{m}$ $0.58\mathrm{m}$

Tabella 3.6: Risultati della comparazione numerica relativi alla traiettoria percorsa riducendo la velocità massima in curva a $1.9\,\mathrm{m/s}$

coefficienti di sicurezza nelle precedenti fasi del calcolo⁸. Non avendo seguito tale procedura, la prima motivazione a supporto dell'ulteriore decremento della velocità è volta all'aumento di FS. Secondariamente, anche se uno scostamento medio di $10\,\mathrm{cm}$ si può ritenere molto soddisfacente, una distanza massima di $3.68\,\mathrm{m}$ dalla traiettoria di riferimento si ritiene troppo elevata.

Sulla base di queste valutazioni, la velocità di percorrenza per tutte le curve strette, disegnate con 3 waypoint, è stata fissata pari a:

$$v = 1 \,\mathrm{m/s}$$

la quale genera una forza centrifuga pari a:

$$F_c = 15.0 \,\mathrm{N}$$

In questo modo il fattore di sicurezza assume un incremento notevole, risultando pari a:

$$FS = 3.6$$

I risultati della comparazione numerica, riportati in tabella 3.7 dimostrano una notevole riduzione della distanza massima e un'ulteriore riduzione della distanza media e della deviazione standard. In figura 3.14 è possibile osservare il miglioramento della traiettoria percorsa al progressivo diminuire della velocità di percorrenza in curva.

⁸Se anche l'altezza del baricentro delle masse sia stata ricavata in maniera eccessivamente cautelativa, non potendo svolgere le indagini sperimentali non si conosce l'entità di tale sovradimensionamento, pertanto non può essere preso in considerazione come coefficiente di sicurezza in approccio rigoroso.

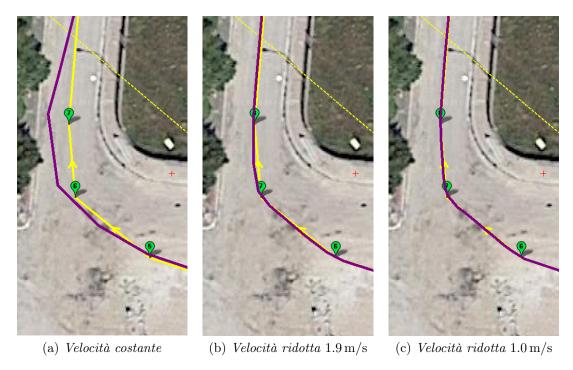


Figura 3.14: Confronto della traiettoria percorsa (viola) rispetto alla pianificata (giallo) al variare della velocità in curva

Parametro	Valore
Distanza massima Distanza media	2.60 m
Distanza media Deviazione standard	$0.08\mathrm{m}$ $0.44\mathrm{m}$

Tabella 3.7: Risultati della comparazione numerica relativi alla traiettoria percorsa riducendo la velocità massima in curva a $1.0\,\mathrm{m/s}$

Limitazione dell'accelerazione trasversale massima

Si può limitare l'accelerazione laterale massima in curva tramite un opportuno parametro: TURN_MAX_G, utilizzato dall'autopilota durante la guida autonoma.

Questa limitazione è stata introdotta per evitare un eventuale incidente e non è in grado di aiutare l'autopilota nella determinazione della traiettoria ideale.

Se i sensori misurano un valore di accelerazione laterale oltre il limite imposto, l'autopilota intraprende delle azioni volte ad abbassare tale valore, a scapito della traiettoria pianificata: le manovre compiute in questa fase di emergenza comportano una deviazione dal percorso di riferimento, e potrebbero causare un

incidente, la problematica potrebbe verificarsi in un contesto con poco margine di manovra. Si imposta il parametro al suo valore originale TURN_MAX_G, espresso in $G=9.81\,\mathrm{m/s/s}$, al valore:

$$TURN_MAX_G = 0.5$$

3.3.7 Parametri definitivi

Le numerose simulazioni di missione effettuate hanno permesso di determinare i criteri di comportamento dell'UGV e i valori dei parametri che assicurano il miglior rispetto della traiettoria pianificata.

In tabella 3.8 sono riportati i parametri modificati, con i valori originali e finali, unitamente alla loro descrizione.

Parametro	Unità	Opzioni	Valore iniziale	Valore finale
ACRO_TURN_RATE	deg/s	0 - 360	180	80
ATC_STR_RAT_FF		0,0 - 3,0	0,2	1,2
CRUISE_THROTTLE	%	0 - 100	100	30
TURN_RADIUS	\mathbf{m}	0 - 10	10	1
WP_OVERSHOOT	m	0 - 10	2	1
WP_RADIUS	\mathbf{m}	0 - 100	2	1
WP_SPEED	m/s	0 - 100	10	6
TURN_MAX_G	m/s	0,1 - 10,0	0,5	0,5

Tabella 3.8: Parametri finali assegnati all'*UGV*

3.4 Analisi delle pendenze

L'UGV ha una navigazione vincolata al terreno, pertanto si deve prestare particolare attenzione nel pianificare percorsi nei quali la pendenza sia compatibile con le capacità del veicolo. Il Traxxas X Maxx, nella sua configurazione originale, riesce a scalare pendenze oltre il 30 % senza significanti perdite di prestazioni. Considerando lo stesso peso aggiuntivo ipotizzato per la verifica al ribaltamento laterale, si limita la pendenza massima dei percorsi in missione al $20\,\%^9$.

 $^{^9}$ È un'approssimazione cautelativa, la determinazione di questo valore è fortemente dipendente da posizione e massa dell'oggetto trasportato, condizioni non determinabili a causa dell'impossibilità di svolgere sperimentazioni comportata dalle misure di contenimento adottate dal Politecnico di Torino durante l'emergenza COVID-19.

Per determinare la pendenza del percorso è necessario disporre di un modello digitale di elevazione del terreno (DEM). Diversi enti pubblici forniscono DEM del territorio italiano, con bassa risoluzione, gratuitamente. Per questo progetto, sono stati utilizzati i DEM distribuiti dal servizio *TINITALY*[19] dell'istituto nazionale di geofisica e vulcanologia (INGV), realizzati con una risoluzione di 10 m.

Per svolgere l'analisi è stato utilizzato il software libero QGIS[31]. Importando i punti della traiettoria pianificati nel sistema di riferimento WGS84 - UTM Fuso 33N, è stato caricato il DEM nel formato vettoriale Raster con estensione .tif, ovvero con compressione di dati senza perdita (lossless). Il DEM introdotto è stato georeferenziato nel sistema di riferimento globale WGS84, pertanto è stato necessario riproiettarlo nel WGS84 - UTM Fuso 33N. Dal modello di elevazione digitale del terreno sono note le informazioni di quota. Per ricavare i dati di pendenza, espressi in valore percentuale %, è stato utilizzato lo strumento di analisi Pendenze.

In conclusione, sono stati associati i dati di quota e pendenza ai punti della traiettoria di riferimento tramite il plugin Point Sampling Tool. In questo modo è stato possibile associare i valori di quota e pendenza alla distanza progressiva che il rover deve percorrere. In figura 3.15 è riportato il profilo altimetrico della traiettoria pianificata, dal quale sono state determinate le pendenze, riportate in figura 3.16. In tutto il tracciato la pendenza è inferiore al 14%, fatta eccezione per un tratto di estensione molto contenuta in cui si raggiunge la pendenza massima:

pendenza massima = 14.77%

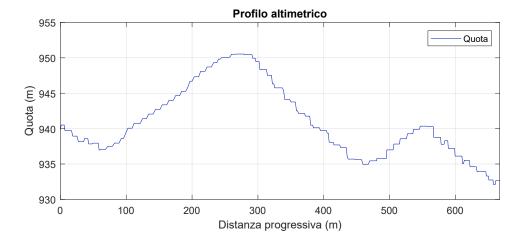


Figura 3.15: Profilo altimetrico del percorso pianificato - Chiesa San Francesco

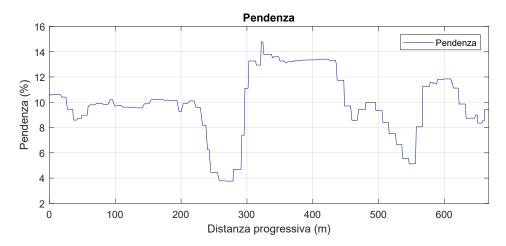


Figura 3.16: Variazione della pendenza del percorso pianificato - Chiesa San Francesco

3.5 Percorso per la chiesa di Sant'Agostino

Avendo determinato i parametri ottimali e i criteri di comportamento dell'UGV, è stata pianificata la missione volta al rilievo della chiesa di Sant'Agostino. In particolare, il veicolo guiderà autonomamente sino al punto prestabilito di fronte alla chiesa, stazionerà per il tempo necessario al compimento della scansione, quindi ritornerà al punto di partenza.

L'azione di stazionamento si assegna con il comando DELAY, espresso in secondi, da inserire nell'elenco dei waypoint. Si riporta la traiettoria pianificata in figura 3.17, sovrapposta alla discretizzazione calcolata per la comparazione.

In questo percorso alcune curve sono state disegnate utilizzando un numero maggiore di waypoint: i 3 punti sono considerati il numero minimo, utilizzato per tracciare curve strette in prossimità dell'intersezione diretta di due strade. In queste situazioni, gli assi delle due carreggiate non sono raccordati da una curva, per cui è il veicolo autonomo, e in generale anche il comune automobilista, che devono disegnare una curva nella manovra di svolta¹⁰.

Nei tratti in cui la strada sviluppa una curva, il numero di punti di riferimento necessari per disegnare la manovra di svolta deve essere incrementato: difatti si deve seguire una curva di ampio raggio, quindi con ampia estensione. Le traiettorie si pianificano con una linea spezzata, curve estese hanno necessità di molti punti per essere ben approssimate con segmenti rettilinei.

¹⁰È lo stesso fenomeno che avveniva per la guida manuale prima della nascita delle clotoidi: nelle curve a raggio costante, l'automobilista disegna la clotoide con il volante.

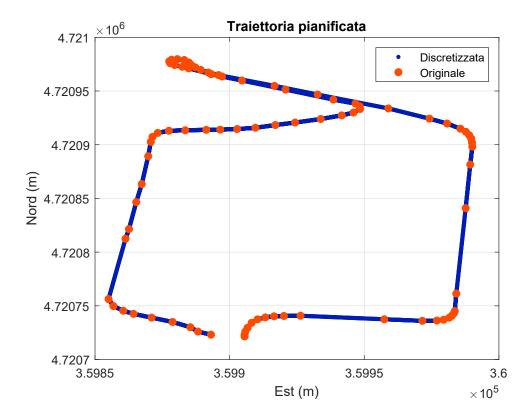


Figura 3.17: Traiettoria pianificata per la chiesa di Sant'Agostino

I risultati della comparazione numerica, riportati in tabella 3.9, sono allineati con i risultati del percorso pianificato per la chiesa di San Francesco, e dimostrano un elevato rispetto della traiettoria pianificata, con una distanza media inferiore a 10 cm. In tutto il tracciato la pendenza è inferiore al 11 %, fatta eccezione per un

Parametro	Valore
Distanza massima Distanza media	2.04 m 0.09 m
Deviazione standard	$0.41\mathrm{m}$

Tabella 3.9: Risultati della comparazione numerica relativi della traiettoria pianificata per il rilievo della chiesa di Sant'Agostino

tratto di estensione molto contenuta in cui si raggiunge la pendenza massima:

pendenza massima =
$$13.27\%$$

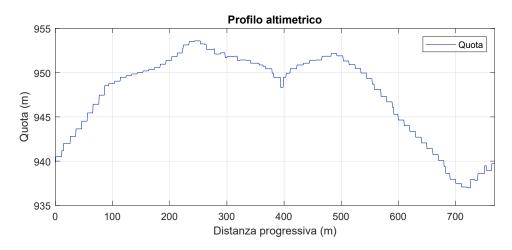


Figura 3.18: Profilo altimetrico del percorso pianificato - Chiesa Sant'Agostino

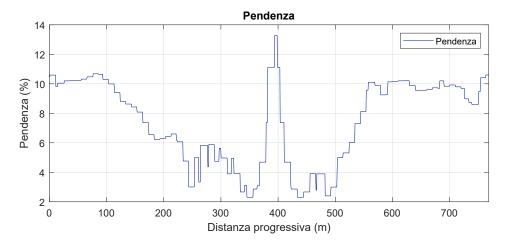


Figura 3.19: Variazione della pendenza del percorso pianificato - Chiesa Sant'Agostino

In figura 3.20 è riportato un esempio di scansione 3D realizzabile con l'UGV. Il rilievo illustrato è stato realizzato dal gruppo di ricerca del Laboratorio Fotogrammetria, Geomatica e GIS del Politecnico di Torino tramite UAV[6]. In perfetta analogia con le modalità già testate, la stessa tecnica di rilievo può essere realizzata dall'UGV progettato in questo lavoro di tesi, in grado di offrire una diversa prospettiva e numerosi altri vantaggi, discussi nel capitolo 5, quello finale, delle Conclusioni.



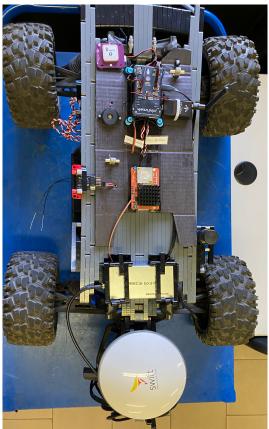
Figura 3.20: Rilievo 3D della chiesa di Sant'Agostino, acquisito con Laser Scanner dal gruppo di ricerca del *Laboratorio Fotogrammetria*, *Geomatica e GIS* del *Politecnico di Torino*[6]

3.6 Collaudo con missione autonoma

La parziale ripresa delle attività di ricerca sperimentale presso il *Politecnico di Torino*, a seguito dell'emergenza *COVID-19*, ha permesso di effettuare una fase di collaudo nei pressi della sede centrale, sita in Corso Duca degli Abbruzzi a Torino. È stato possibile testare la sola navigazione in SPP in quanto un componente di uno dei due ricevitori *Piksi Multi*, acquistati per realizzare il posizionamento RTK, risulta danneggiato e non è stato possibile sostituirlo né ripararlo nei tempi stabiliti per il termine di questo lavoro di tesi.

Nella fase di collaudo sono state programmate diverse missioni autonome, volte a determinare lo stato di operatività del veicolo. Le prime missioni, molto semplici, hanno permesso di stabilire il corretto funzionamento dell'UGV. In seguito, è





(a) Collaudo

(b) Dettaglio strumentazione

Figura 3.21: Prototipo UGV nella configurazione di collaudo

stata pianificata una missione da svolgere trasportando un terzo ricevitore GNSS, il *Tersus BX316*, utilizzato per acquisire e post processare le posizioni percorse dal veicolo durante la prova. I dati acquisiti dal *Tersus* costituiranno il riferimento di maggior precisione, con cui comparare i dati registrati dall'AutoPilota. Oltre a voler determinare lo scostamento ottenuto dalla traiettoria pianificata, si è interessati all'analisi delle posizioni registrate dal Pixhawk, per comprendere meglio il comportamento del veicolo.

Si riporta la traiettoria pianificata in figura 3.22, sovrapposta alla discretizzazione calcolata per la comparazione con *CloudCompare*. Si precisa che non è stato possibile simulare la missione in quanto è stata svolta in una strada pubblica, dalla visione satellitare non è possibile conoscere l'attuale condizione delle strade, la presenza di eventuali automobili parcheggiate e il flusso di persone. Pertanto, la missione è stata pianificata e avviata sul luogo, a seguito di un'attenta ricognizione e con un continuo monitoraggio.

In figura 3.23 è riportato il profilo altimetrico della traiettoria pianificata, dal

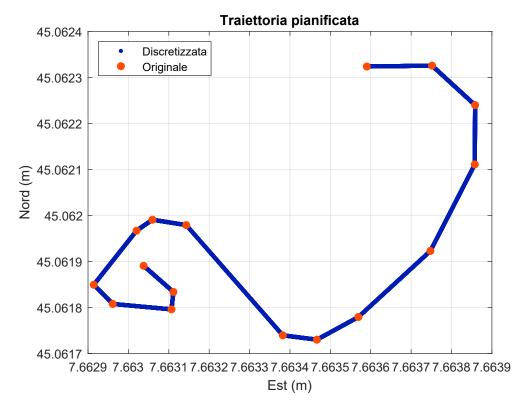


Figura 3.22: Traiettoria pianificata per il collaudo alla statua del Fante

quale sono state determinate le pendenze, riportate in figura 3.24. Il tracciato è prevalentemente piano, le pendenze risultano inferiori al punto percentuale.

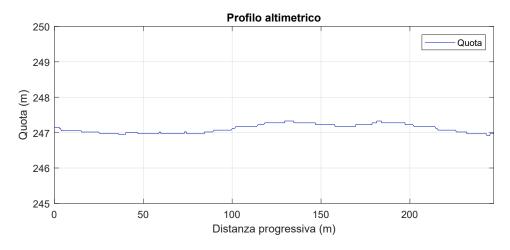


Figura 3.23: Profilo altimetrico del percorso pianificato - Collaudo

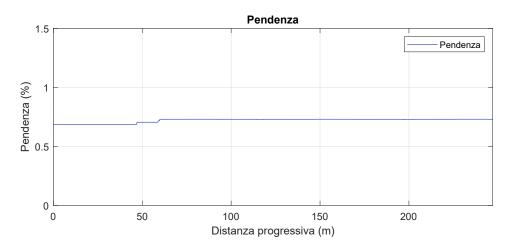


Figura 3.24: Variazione della pendenza del percorso pianificato - Collaudo

I risultati della missione di collaudo sono discussi nel successivo capitolo, dei Risultati, al paragrafo 4.2.1.

Capitolo 4

Risultati

In questo capitolo sono esposti i risultati ottenuti dalla comparazione numerica delle traiettorie e dalle analisi svolte sui dati acquisiti durante le missioni autonome di collaudo.

4.1 Risultati delle simulazioni

Come esposto nel precedente capitolo 3, la comparazione numerica non è stata svolta per ogni variazione di un singolo parametro. Nella fase di affinamento iniziale la comparazione è stata principalmente di tipo visivo, in quanto la modifica dei parametri comportava notevoli modifiche al comportamento dell' UGV, comprensibili dalla traiettoria percorsa disegnata sulla mappa di $Mission\ Planner$.

Per gli affinamenti finali, invece, le modifiche comportavano incrementi di precisione non valutabili per via esclusivamente grafica, pertanto sono state svolte le comparazioni numeriche, i cui risultati, relativi alla distanza media e alla deviazione standard, sono stati riportati caso per caso.

La tabella 4.1 sintetizza gli incrementi di precisione ottenuti per ogni stadio (step) di affinamento, nella navigazione pianificata per la chiesa di San Francesco. E evidenzierà la correttezza della procedura, la quale ha permesso di individuare la traiettoria ottimale per la chiesa di Sant'Agostino al primo tentativo.

La tabella, così come la procedura, è organizzata nei seguenti stadi:

- 1. Simulazione con i parametri iniziali
- 2. Velocità di sterzata
- 3. Modifica delle curve strette
- 4. Estensione della svolta
- 5. Riduzione della velocità massima in curva a 1.9 m/s

- 6. Riduzione della velocità massima in curva a $1.0\,\mathrm{m/s}$
- 7. Pianificazione unica per la chiesa di Sant'Agostino

I grafici riportati nelle figure da 4.1 a 4.7 illustrano i risultati delle comparazioni numeriche, espressi con il supporto di una legenda a colori.

In particolare, le figure sono composte da due grafici (a) e (b), nelle quali sono riportati gli scostamenti, calcolati come distanza assoluta tramite l'algoritmo C2C, in due rappresentazioni differenti:

- (a): istogrammi della distribuzione degli scostamenti, suddivisa in 36 classi;
- (b): rappresentazione planimetrica della distribuzione degli scostamenti, lungo la traiettoria.

Step	Distanza massima	Distanza media	Deviazione standard
1	$7.83\mathrm{m}$	$1.65\mathrm{m}$	$1.83\mathrm{m}$
2	$7.82\mathrm{m}$	$0.76\mathrm{m}$	$1.66\mathrm{m}$
3	$7.81\mathrm{m}$	$0.90\mathrm{m}$	$1.91\mathrm{m}$
4	$3.67\mathrm{m}$	$0.14\mathrm{m}$	$0.66\mathrm{m}$
5	$3.68\mathrm{m}$	$0.10\mathrm{m}$	$0.58\mathrm{m}$
6	$2.60\mathrm{m}$	$0.08\mathrm{m}$	$0.44\mathrm{m}$
7	$2.04\mathrm{m}$	$0.09\mathrm{m}$	0.41 m

Tabella 4.1: Risultati delle comparazioni numeriche sulle traiettorie percorse - Simulazione

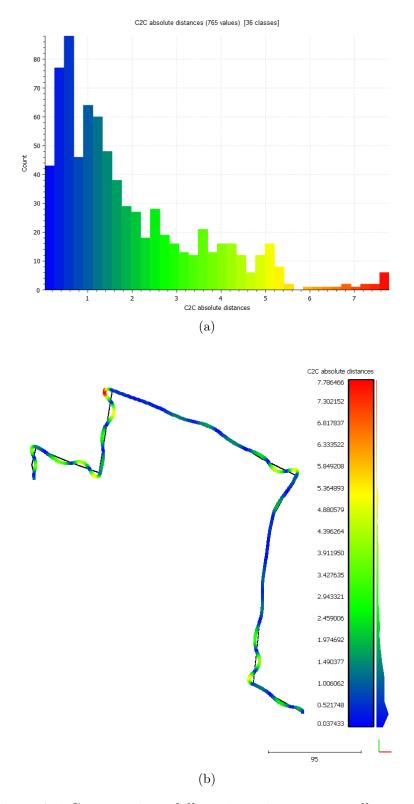


Figura 4.1: Comparazione della traiettoria percorsa nello step $1\,$

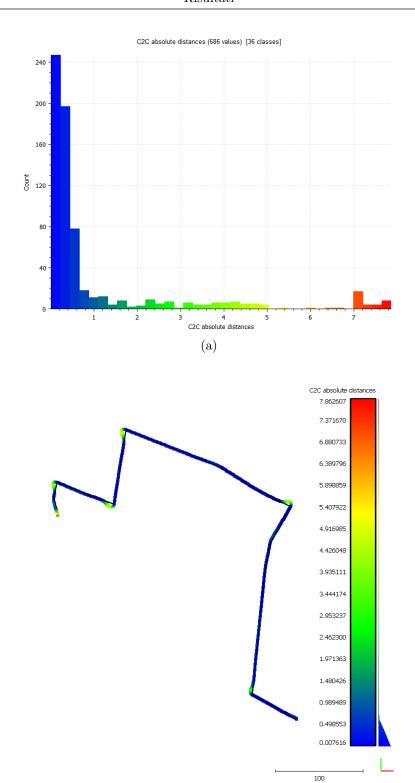


Figura 4.2: Comparazione della traiettoria percorsa nello step $2\,$

(b)

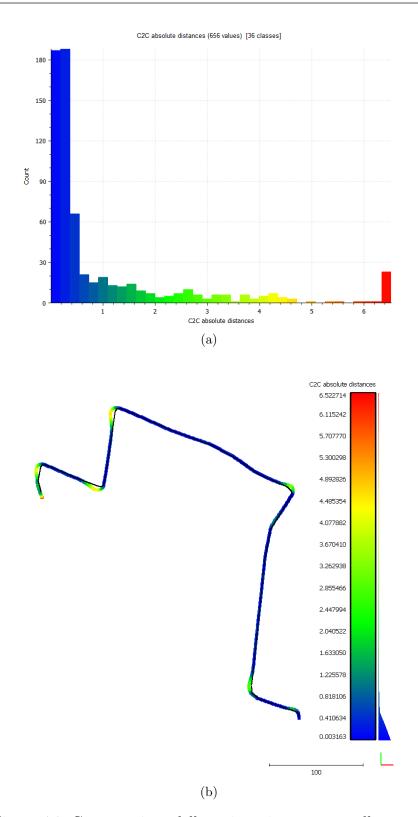


Figura 4.3: Comparazione della traiettoria percorsa nello step $3\,$

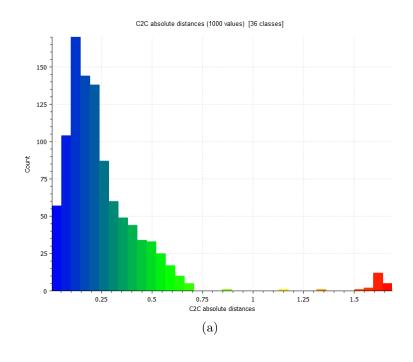
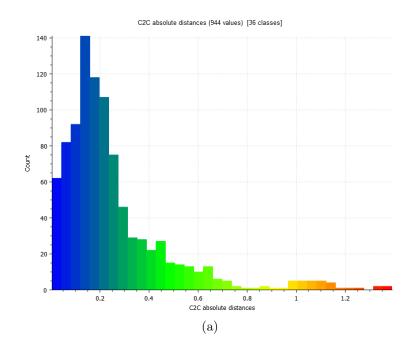




Figura 4.4: Comparazione della traiettoria percorsa nello step $4\,$



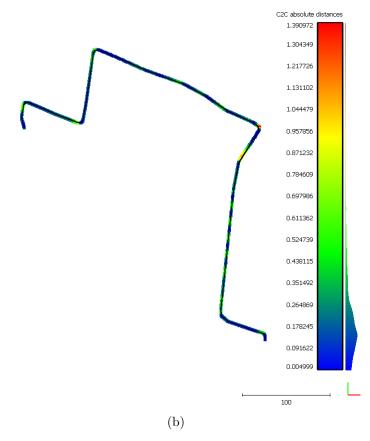
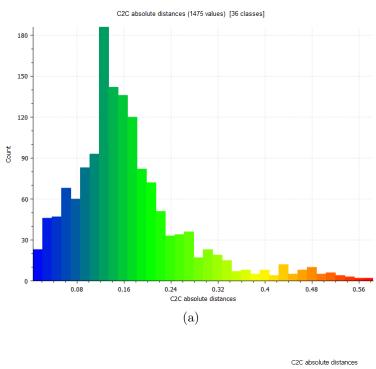


Figura 4.5: Comparazione della traiettoria percorsa nello step $5\,$



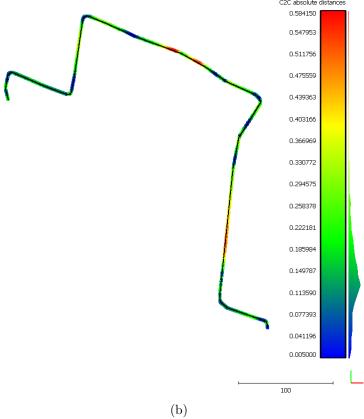
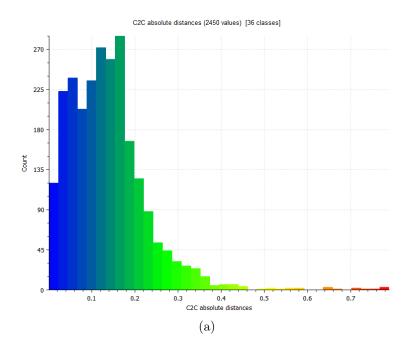


Figura 4.6: Comparazione della traiettoria percorsa nello step $6\,$



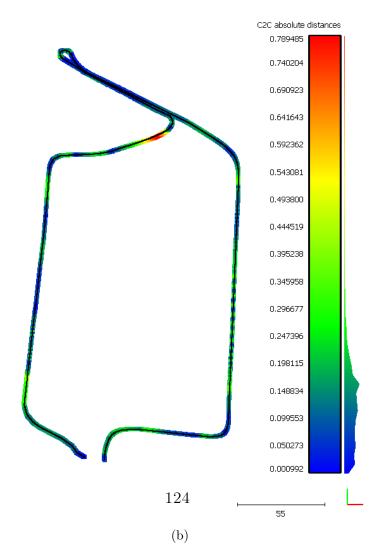


Figura 4.7: Comparazione della traiettoria percorsa nello step $7\,$

4.1.1 Discussione

La fase iniziale, di definizione dei parametri di guida globali, deve inizialmente preliminarmente dei criteri di scelta dei parametri che siano compatibili con il raggiungimento dell'obiettivo prefissato: in questo caso, il minor scostamento possibile della traiettoria percorsa rispetto alla pianificata. Questi criteri sono fondamentali per la corretta impostazione della procedura di affinamento.

Come dimostrato nella procedura di affinamento, la determinazione dei soli parametri di guida globali non è sufficiente per il raggiungimento dello scopo. Di seguito saranno discussi i benefici e le problematiche riscontrati durante la tale fase.

Velocità di sterzata

Il parametro ATC_STR_RAT_FF ha una forte influenza sul comportamento del veicolo. Si deve evitare di impostarlo su valori troppo elevati in quanto comporterebbe
velocità di sterzate elevate. Si rischia lo sbandamento continuo del veicolo, in quanto il servomotore di sterzo non sarebbe più capace di generare piccoli angoli di
sterzata, necessari per l'allineamento alla traiettoria. Questo parametro deve essere
necessariamente testato con missioni di prova e, se necessario, ulteriormente affinato, facendo fede alla traiettoria percorsa reale, la quale potrebbe differire dalla
simulata.

Nella fase di collaudo, i cui risultati, riportati nel prossimo paragrafo 4.2, sono discussi al paragrafo 4.2.1, il parametro non è stato affinato: l'impostazione al valore individuato nella fase di simulazione ha conferito un ottimo comportamento all'UGV, permettendo di procedere direttamente alla pianificazione di una missione.

Traiettoria in curva

L'affinamento della traiettoria in curva è uno dei fattori più importanti per il raggiungimento dello scopo. Gli angoli ampi comportano manovre più semplici, permettendo all'autopilota di scegliere una traiettoria con scostamento minimo da quella pianificata: la situazione ideale è rappresentata dalla traiettoria rettilinea, e le curve sono disegnate unendo segmenti rettilinei: minori è la variazione di direzione, e quindi l'angolo, tra segmenti successivi, maggiore sarà la fedeltà del veicolo nel rispettare la traiettoria disegnata.

Inoltre questa soluzione possiede un'importante caratteristica: il progettista traccia un percorso molto verosimile alla traiettoria che percorrerebbe alla guida di un autovettura. Questa familiarità consente una maggior supervisione e consapevolezza del comportamento dell'UGV.

Nella navigazione autonoma, una delle maggiori difficoltà per i progettisti consiste nella comprensione del percorso pianificato dal robot: costruendo le curve con

più waypoint si vincola la traiettoria e si elimina l'incertezza della manovra di svolta che l'UGV potrebbe compiere.

In un ambiente in cui la navigazione è resa difficile per la presenza di macerie, un elevato controllo sulla traiettoria percorsa è fondamentale per riuscire a evitare, con efficacia, gli ostacoli presenti sul territorio.

Estensione della svolta e riduzione della velocità in curva

Come illustrato nel paragrafo 3.3.6, per ottenere la traiettoria ottimale è necessario ridurre la velocità in curva, e per apportare tale modifica sono stati valutati due approcci differenti.

Il primo approccio riduce al minimo l'estensione della svolta per mezzo del parametro WP_OVERSHOOT. In questo modo, l'autopilota è costretto a diminuire la velocità all'arrivo in ogni waypoint in quanto deve orientarsi verso il successivo nel minor spazio possibile. Anche se questo approccio ha restituito buoni risultati numerici, in termini di distanza media e deviazione standard fra le due traiettorie sono emerse alcune problematiche, di fondamentale importanza per la navigazione.

In particolare, la riduzione di velocità in ogni waypoint genera un comportamento altamente incostante, con frenate e successive accelerazioni che si alternano di frequente, come illustrato in figura 4.8. Una navigazione così condotta comporta un

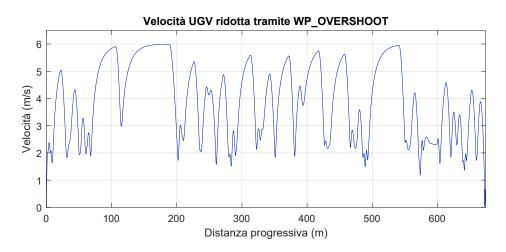


Figura 4.8: Influenza del parametro globale WP_OVERSHOOT sulla velocità nell'intero tracciato

elevato consumo di energia, a scapito dell'autonomia, in quanto il veicolo rallenta anche nei tratti rettilinei, in cui non si ha interesse a ridurre la velocità: piuttosto la si vuole tenere al valore massimo stabilito.

Inoltre, lasciando questo genere di autonomia all'autopilota non si ha controllo sull'accelerazione laterale e quindi sul ribaltamento¹, problematica dominante nelle missioni che prevedono l'installazione di strumenti pesanti sopra il piano hardware.

Si preferisce quindi limitare la velocità nelle sole curve strette, o in generale nei tratti che presentano particolari difficoltà di navigazione, intervenendo localmente con il comando DO_CHANGE_SPEED, e garantendo al contempo la stabilità.

Come illustrato in figura 4.9, si ottiene una navigazione più costante e conservativa nei confronti dell'autonomia, assicurando la bassa velocità nei soli tratti stabiliti. Nella missione simulata si è deciso di far partire il veicolo con una velocità moderata, in modo da consentire un corretto allineamento con la traiettoria pianificata, e di applicare la stessa riduzione di velocità al raggiungimento del punto di rilievo, sempre per consentire lo svolgimento di manovre più precise.

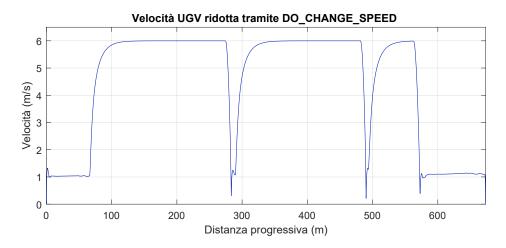


Figura 4.9: Influenza del parametro locale DO_CHANGE_SPEED sulla velocità nell'intero tracciato

Confrontando i due approcci, nel secondo, in cui la riduzione di velocità è stata imposta manualmente, la durata della missione è aumentata in una misura non penalizzante: la prima simulazione ha avuto una durata 3 minuti e 31 secondi, per la seconda missione sono stati impiegati 4 minuti e 32 secondi. L'aumento di 1 minuto e 1 secondo, essendo a favore della sicurezza, si considera preferenziale.

¹Come riportato al paragrafo 3.3.6, non si fa affidamento sull'accelerazione laterale rilevata per ritenere la navigazione sicura nei confronti del ribaltamento laterale.

4.2 Risultati delle missioni autonome

Per comprendere i risultati delle missioni autonome, sono stati analizzati i dati registrati nel file .log, scaricato dal *Pixhawk*, dopo aver concluso la missione. Si descrivono di seguito i log riportati:

- AHRS: Altitude Heading Reference System, è un risultato dell'elaborazione svolta dal *Pixhawk* durante la navigazione, riferita alle misure dei sensori collegati;
- GPS: sono le posizioni registrate dal modulo GNSS economico, compreso nell'acquisto del *Pixhawk*;
- GPS2: sono le posizioni registrate dal modulo GNSS ad alte prestazioni *Piksi Multi*;
- POS: sono le posizioni finali determinate da *ArduPilot*, costituiscono la posizione considerata vera (reale) dall'*AutoPilota*;
- TERSUS: sono le posizioni registrate terzo ricevitore GNSS, il *Tersus BX316*, utilizzato per post processare i dati.

Si riassumono i risultati delle comparazioni numeriche rispetto alla traiettoria di riferimento nella tabella 4.2, la quale riporta gli scostamenti, calcolati come distanza assoluta tramite l'algoritmo C2C. I grafici riportati nelle figure da 4.10 a 4.14 illustrano gli stessi risultati, espressi con il supporto di una legenda a colori.

Log	Distanza massima	Distanza media	Deviazione standard
AHRS GPS GPS2	$2.97\mathrm{m}$ $2.97\mathrm{m}$ $11.79\mathrm{m}$	$0.51 \mathrm{m}$ $0.96 \mathrm{m}$ $0.83 \mathrm{m}$	$0.62\mathrm{m}$ $0.76\mathrm{m}$ $1.34\mathrm{m}$
POS TERSUS	$3.20\mathrm{m}$ $3.03\mathrm{m}$	$\begin{array}{c} 0.36\mathrm{m} \\ 0.81\mathrm{m} \end{array}$	$0.54\mathrm{m}$ $0.70\mathrm{m}$

Tabella 4.2: Risultati delle comparazioni numeriche rispetto alla traiettoria di riferimento - Collaudo

Inoltre, in tabella 4.3 sono riportati i risultati della comparazione numerica rispetto alle posizioni registrate dal *Tersus BX316*, post processate con tecnica differenziale quindi più precise e fedeli al reale percorso compiuto dal veicolo.

Log	Distanza massima	Distanza media	Deviazione standard
AHRS	$3.25\mathrm{m}$	$0.81\mathrm{m}$	$0.63\mathrm{m}$
GPS	$3.76\mathrm{m}$	$0.92\mathrm{m}$	$0.77\mathrm{m}$
GPS2	$8.98\mathrm{m}$	$1.08\mathrm{m}$	$1.15\mathrm{m}$
POS	$3.25\mathrm{m}$	$0.75\mathrm{m}$	$0.66\mathrm{m}$
REFERENCE	$3.03\mathrm{m}$	$0.81\mathrm{m}$	$0.70\mathrm{m}$

Tabella 4.3: Risultati delle comparazioni numeriche rispetto alla traiettoria del $Tersus\ BX316$ - Collaudo

Nelle figure da 4.15 a 4.20 sono riportati gli ingrandimenti di alcuni tratti percorsi, in cui è possibile osservare il risultato dell'elaborazione interna del software AutoPilota, discussa nel successivo paragrafo.

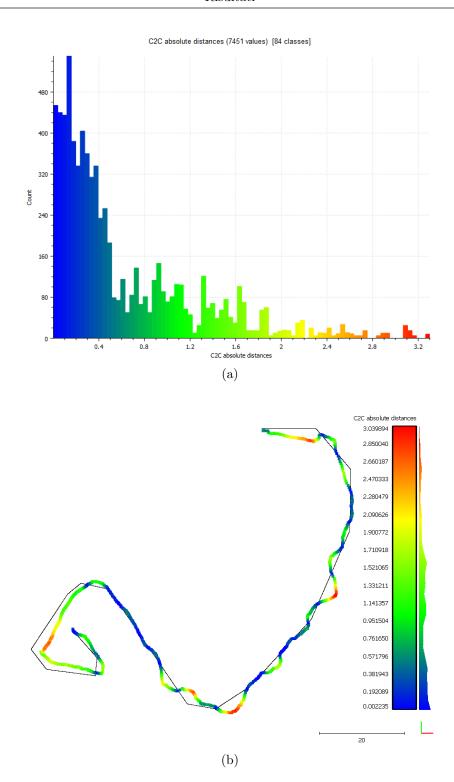


Figura 4.10: Comparazione della traiettoria registrata AHRS

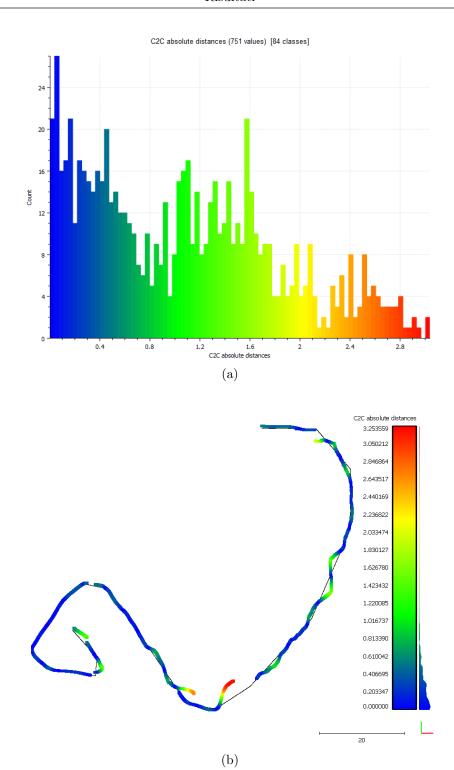


Figura 4.11: Comparazione della traiettoria registrata GPS $\,$

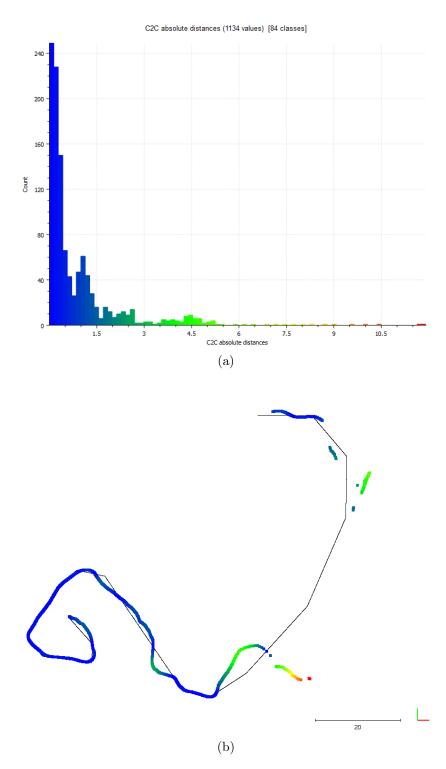


Figura 4.12: Comparazione della traiettoria registrata ${\it GPS2}$

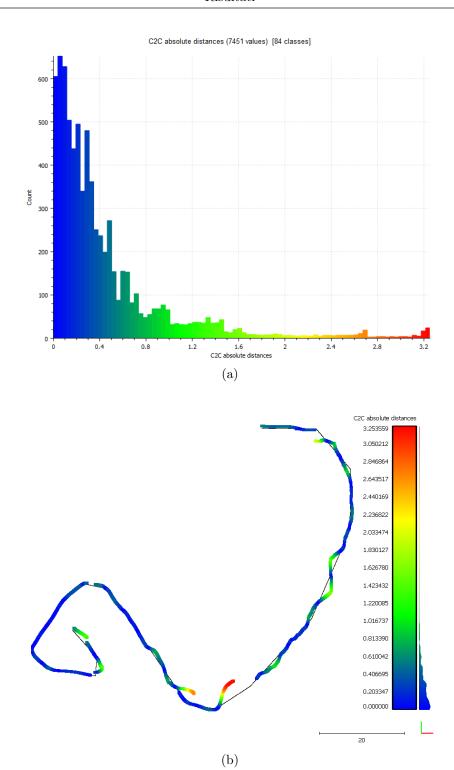


Figura 4.13: Comparazione della traiettoria registrata POS

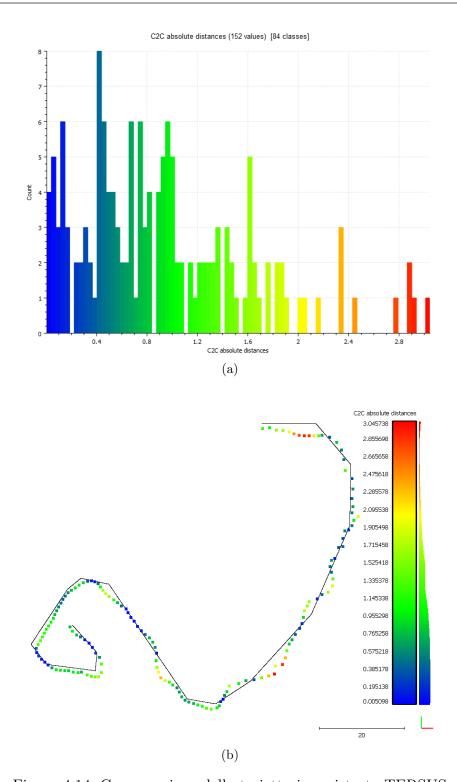


Figura 4.14: Comparazione della traiettoria registrata TERSUS

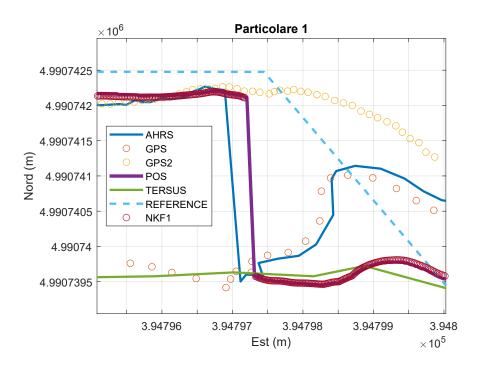


Figura 4.15: Ingrandimento della traiettoria - Particolare 1

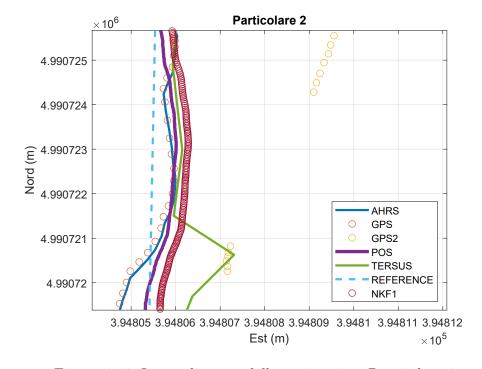


Figura 4.16: Ingrandimento della traiettoria - Particolare 2

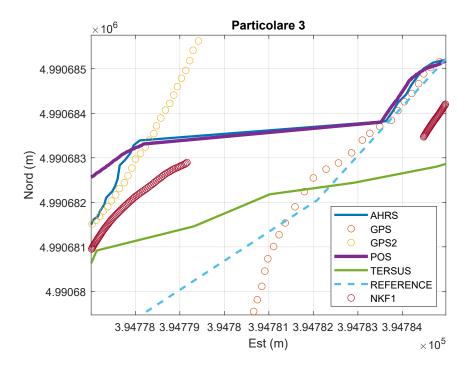


Figura 4.17: Ingrandimento della traiettoria - Particolare 3

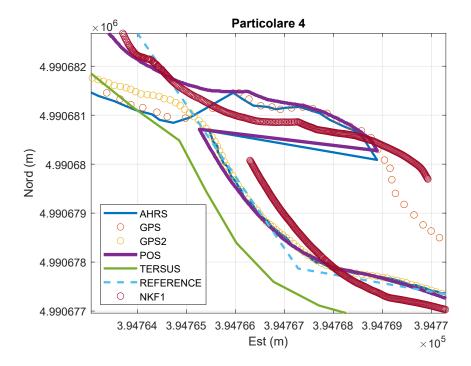


Figura 4.18: Ingrandimento della traiettoria - Particolare $4\,$

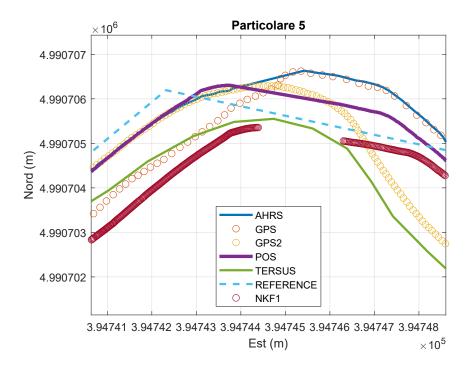


Figura 4.19: Ingrandimento della traiettoria - Particolare 5

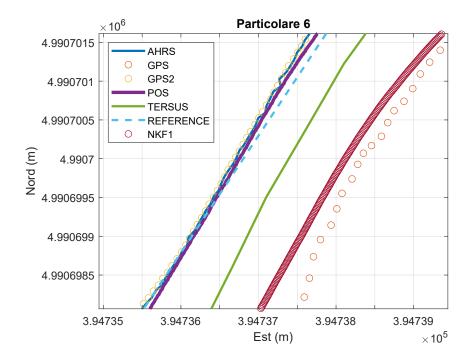


Figura 4.20: Ingrandimento della traiettoria - Particolare 6

4.2.1 Discussione

Lo scopo finale di questa analisi approfondita consiste nell'esprimere un giudizio sulla qualità della navigazione autonoma e nel comprendere quali siano i fattori dominanti per ottenere il miglior comportamento possibile.

La prima osservazione deducibile riguarda la contenuta distanza media: per ogni traiettoria misurata, lo scostamento medio risulta sempre inferiore a 1 m.

Successivamente, si osserva un comportamento instabile del GPS2, il *Piksi Multi*. Si attribuisce tale comportamento sia alla presenza di alberi sul percorso sia alla probabile inadeguatezza dell'antenna elicoidale installata.

Inoltre, considerando i risultati delle comparazioni di POS e TERSUS, si può osservare come la distanza media rilevata dal ricevitore *Piksi Multi*, il modulo a elevate prestazioni, sia maggiore al doppio di quella calcolata e utilizzata dall' *AutoPilota*. Questo risultato evidenzia l'importanza del contributo aggiunto dal posizionamento RTK, il quale si rende necessario per raggiungere precisioni elevate.

Osservando i particolari, si può notare anche come l'output del filtro di Kalman accumuli uno scostamento crescente rispetto alle posizioni vere POS, le quali presentano una traiettoria di uguale forma a NKF1.

Infine, si osserva come la posizione POS, considerata la reale dall'AutoPilota, sia caratterizzata da improvvise discontinuità (salti di posizione), come illustrato nei particolari 1, 3 e 4 (rispettivamente figure 4.15, 4.17 e 4.18). Questo comportamento si ritiene essere dipendente dal log AHRS, il quale, tendenzialmente, si affianca alla soluzione di GPS o GPS2, a seconda dell'accuratezza stabilità dal sistema, per ogni istante. Quest'ultima evidenza, infatti, è riscontrabile in tutti i particolari riportati.

La soluzione del ricevitore TERSUS costituisce le posizioni più precise realmente occupate dal veicolo, consentendo di formulare specifiche considerazioni. Si discutono di seguito i singoli particolari.

Particolare 1

Nel particolare 1, illustrato in figura 4.15, si osserva la prima discontinuità dei log AHRS e POS: il software stabilisce che la posizione più accurata non è più quella di GPS2, decidendo di utilizzare GPS. Si può osservare come, nella prima metà del tratto analizzato, la soluzione di GPS sia allineata con quella del TERSUS, quindi effettivamente corretta.

Se prima del salto il *Pixhawk* considerava POS allineata con GPS2 e AHRS, dopo la discontinuità POS non risulta allineato né con GPS né con AHRS. Interviene, infatti, il contributo del filtro di Kalman NKF1, il quale elabora un ulteriore output che considera anche i dati delle unità inerziali *IMU*. L'*AutoPilota* comprende che il veicolo non ha compiuto un cambio di direzione marcato come riportato la soluzione GPS, pertanto stabilisce che POS debba restare fedele alla soluzione del filtro di Kalman NKF1.

Particolare 2

Nel particolare 2, illustrato in figura 4.16, si può osservare un improvviso salto di posizione del TERSUS, in accordo con le posizioni rilevate dal GPS2, seppur con discontinuità. In questo tratto del percorso, la presenza di alberi disturbava il segnale. Seppur in un tratto isolato, il *Piksi Multi* è riuscito a registrare la posizione vera, senza RTK. Ulteriori test, volti a risolvere l'incostanza delle misurazioni del Piksi, sono necessari per riuscire a sfruttare al meglio la precisione di questo modulo. Si ritiene che l'antenna elicoidale collegatavi potrebbe non essere la più performante.

Particolare 3

Nel particolare 3, illustrato in figura 4.17, si osserva un ulteriore salto di posizione di AHRS, il quale si sposta da GPS per appoggiarsi su GPS2: in quell'istante la soluzione del *Piksi Multi* è stata ritenuta più precisa e affidabile. Difatti, si può osservare come il conseguente cambio di percorso di POS disegni una traiettoria parallela a TERSUS, mentre GPS mostra una divergenza marcata. Si osservi, inoltre, come la stessa discontinuità di AHRS e POS sia presente in NKF1.

Particolare 4

Nel particolare 4, illustrato in figura 4.18, avviene lo stesso fenomeno riscontrato nei particolari 1 e 3. Si può osservare come, nel punto di discontinuità, la soluzione di GPS2 inizi a divergere dalla traiettoria di riferimento. Questo dimostra l'efficienza di ArduPilot nel selezionare l'output più affidabile.

Particolare 5

Nel particolare 5, illustrato in figura 4.19, si può osservare come il salto di posizione di AHRS sia più contenuto rispetto ai precedenti casi. Anche in questo particolare si osserva la contemporanea discontinuità di NKF1.

Particolare 6

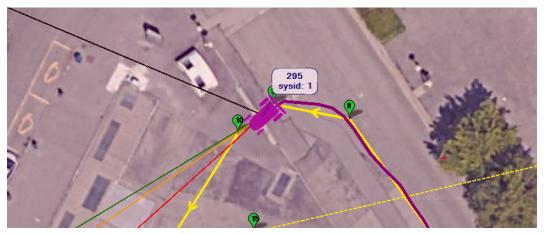
Nel particolare 6, illustrato in figura 4.20, si osserva come la soluzione di GPS2, diventata più accurata e stabile dall'inizio della missione, costituisca il riferimento più solido per l'AutoPilota, il quale ha determinato la propria posizione perfettamente allineata con REFERENCE, AHRS e appunto GPS2. Il TERSUS risulta avere uno scostamento costante, determinato in fase di post processamento. Ancora una volta, si dimostra l'importanza di avere una soluzione RTK: la traiettoria è corretta, affetta da uno scostamento costante rimediabile con il posizionamento differenziale.

Valutazione visiva della navigazione in tempo reale

Le comparazioni numeriche costituiscono un supporto importante per il giudizio della navigazione e per la comprensione delle azioni intraprese dall'UGV. Un fondamentale elemento di giudizio, che non deve mai mancare nella valutazione della navigazione di un qualsiasi Robot, è l'esame visivo dei progettisti che monitorano la fase di collaudo: alcuni comportamenti non sono osservabili dagli registri .log.

Dall'esame visivo, sono state osservate alcune incertezze nell'impostazione della traiettoria dritta tra un waypoint e il successivo: il veicolo ha alternato, velocemente, leggere sterzate a destra e sinistra, compiendo un moto di serpreggiamento.

Nel complesso, il veicolo ha rispettato con successo la traiettoria pianificata e la velocità impostata. Un punto di interesse nella missione di collaudo riguardava il passaggio per uno scivolo, il quale è stato compiuto con successo e precisione, come osservabile in figura 4.21.



(a) Vista in tempo reale su Mission Planner



(b) Fotografia al corrispondete istante

Figura 4.21: Attraversamento di uno scivolo - Collaudo

Capitolo 5

Conclusioni

Nel presente lavoro di tesi è stato assemblato, configurato e collaudato un prototipo di UGV, capace di navigare autonomamente in ambienti esterni, con elevata precisione, utilizzando la tecnica del posizionamento GNSS.

Il veicolo realizzato si è dimostrato molto versatile, pratico e affidabile. Una delle sue caratteristiche di maggior valore è la fattibilità della sua realizzazione: gli UGV impiegati nei progetti citati nello Stato dell'Arte sono molto complessi, risultato di anni di progettazione e dell'impiego di ingenti risorse economiche. La scelta di adottare la filosofia $Open\ Source$ per questo progetto ha permesso di ottenere un veicolo autonomo con una spesa inferiore a $5.000 \in$.

Le maggiori difficoltà riscontrate hanno riguardato la comunicazione tra i diversi dispositivi e la loro alimentazione: durante le sperimentazioni alcuni componenti hanno subito danni irreversibili. La causa è stata attribuita al modulo di alimentazione (BEC), di conseguenza ritenuto insicuro: la configurazione operativa finale, infatti, prevede l'alimentazione del controllore di volo tramite la porta microUSB, collegata a un comune powerbank.

La fase di simulazione si è dimostrata di fondamentale importanza per il corretto utilizzo dell'UGV: ha permesso di comprendere il comportamento del veicolo e il valore da attribuire ai parametri di configurazione. Questa procedura ha consentito di svolgere una fase di collaudo senza inconvenienti, riuscendo a salvaguardare il prototipo e l'attrezzatura installata, e ottenendo da subito un elevato rispetto della traiettoria.

Nella fase di collaudo è stato possibile osservare la qualità dell'Autopilota selezionato per il progetto: ArduRover combina i dati acquisiti dai molteplici sensori installati e li utilizza per prevedere le posizioni future del veicolo. Come osservato nel dettaglio dei particolari, il successo dell'AutoPilota è dovuto proprio alla ridondanza di sensori, che rende possibile applicare con successo il filtro di Kalman esteso (EKF). Difatti, nella fase di collaudo, lo scostamento medio della posizione POS dalla traiettoria di riferimento è stato pari a $0.36\,\mathrm{m}$.

Comparando tale risultato alla soluzione del ricevitore *Tersus BX316*, assunto come il più preciso della prova, si osserva una distanza media reale di 0.81 m, più del doppio del valore riportato dall'*AutoPilota*.

La navigazione autonoma realizzata dal prototipo si considera nel complesso soddisfacente. Per ottenere una navigazione più precisa è necessario utilizzare la tecnica del posizionamento RTK sul Piksi Multi. Nella configurazione in SPP non è stato possibile sfruttare la superiorità del modulo a elevate prestazioni, le cui informazioni sono state più volte scartate dal software, preferendo quelle del modulo ublox.

Infine, si sottolinea l'importanza della ridondanza dei sensori. Il punto di forza di ArduPilot *installato* su *Pixhawk* è proprio la possibilità di realizzare una navigazione accurata e precisa utilizzando i dati di molteplici sensori, non necessariamente costosi e particolarmente performanti.

5.1 Limiti e sviluppi futuri

Il software *Mission Planner*, utilizzato per la pianificazione delle missioni di guida autonoma, non svolge l'analisi delle pendenze, fondamentale per stabilire la percorribilità del tragitto.

Inoltre, il simulatore integrato *SITL* non consente di scegliere la precisione con cui simulare le missioni: è di fondamentale importanza svolgere una campagna di indagini sperimentali per stabilire le differenze di comportamento tra la navigazione affidata al posizionamento RTK e la navigazione affidata al posizionamento SPP. Prima di ogni missione, la simulazione è fondamentale, e, qualora non si potesse realizzare la soluzione RTK, si deve essere in grado di prevedere le conseguenti differenze di comportamento rispetto alle traiettorie simulate.

Il maggior limite del veicolo realizzato è l'impossibilità di navigare autonomamente in ambienti chiusi, in generale dove non è disponibile il segnale GNSS. L'utilizzo di *Pixhawk* e *ArduPilot* non prevede la realizzazione diretta di questa soluzione: applicazioni molto avanzate, tramite l'utilizzo di *ROS* (*Robot Operating System*), consentono l'implementazione degli algoritmi di slam. Tuttavia, la navigazione *indoor* tramite *Pixhawk* non è attualmente sviluppata.

Soluzioni più adatte alla navigazione *indoor*, possibili sviluppi futuri di questo progetto, sono gli *UGV Jackal*[17], realizzati da *Clearpath Robotics*, e i controllori *ArbotiX*: progettati per l'utilizzo di *ROS*, sono adatti all'implementazione degli algoritmi di SLAM, Visual Odometry e l'utilizzo di *anchor* o *nodi* Ultra Wide Band.

Inoltre, per la navigazione in ambienti chiusi sono adatti veicoli di dimensioni contenute, molto inferiori a quelle del Traxxas X Maxx. In questo progetto, l'obiettivo è stato quello di sfruttare le caratteristiche del veicolo per le sue capacità di trasporto, a scapito della sua inadeguatezza per un eventuale futuro utilizzo *indoor*.

Le grandi dimensioni, il costo di realizzazione contenuto, la robustezza e la potenza hanno reso l'X Maxx il veicolo ideale per lo scopo.

Tra i possibili sviluppi futuri di questo progetto, di particolare interesse è la realizzazione della navigazione basata su posizionamento differenziale NRTK: con l'ottenimento delle correzioni differenziali NTRIP da rete, sarà possibile incrementare ulteriormente la precisione della navigazione, rinunciando al contempo al secondo ricevitore GNSS, prima necessario per il posizionamento differenziale RTK.

Un'altra interessante applicazione prevede l'installazione di ruote cingolate: nella comunità RC sono state individuate alcune testimonianze di questa particolare modifica apportata al Traxxas X Maxx. La stabilità e l'aderenza del veicolo migliorano notevolmente, sia sul suolo sconnesso, sia su superfici innevate. Questa realizzazione permetterebbe, in generale, di migliorare il comportamento dell'UGV, e consentirebbe di estendere il contesto di utilizzo dell'UGV a suoli particolarmente impervi.

Infine, si prevede l'implementazione di sensori Sonar, per l'aggiramento di ostacoli nel percorso, non compresi nella dotazione acquistata.

Acronimi

AHRS Altitude Heading Reference System

APM ArduPilot Mega

APR Aeromobile a Pilotaggio Remoto

ARP Antenna Reference Point

ASCII American Standard Code for Information Interchange

BEC Battery Eliminator Circuit

BKG Bundesamt für Kartographie und Geodäsie

CAGR Compound Annual Grouth Rate

CAN Controller Area Network

COC Centro Operativo Comunale

CORS Continuously Operating Reference Station

CPC Carrier Phase Correction

CRC Cyclis Redundancy Check

DEM Digital Elevation Model

DoD Department of Defense

DOP Diluition Of Precision

ECC Error Correction Code

ECEF Earth-Centered Earth-Fixed

EKF Extended Kalman filter

ESA European Space Agency

ESC Electronic Speed Control

FP Framework Program

FPV First Person View

GCS Ground Control Station

GEO Geostationary Earth Orbit

GIS Geographic Information System

GLONASS GLObalnaya NAvigatsionnaya Sputnikovaya Sistema

GND Ground

GNSS Global Navigation Satellite System

GPS Global Positioning System

HTTP HyperText Transfer Protocol

HW Hardware

I2C Inter Integrated Circuit

IMU Inertial Measurement Unit

INGV Istituto Nazionale di Geofisica e Vulcanologia

IP Internet Protocol

IVA Imposta sul Valore Aggiunto

LED Light Emitting Diode

LIDAR Light Detection And Ranging

LiPo Litio Polimero

MAVLink Micro Air Vehicle Link

MEMS Micro Electro-Mechanical Systems

MP Mission Planner

MSG Message

NASA National Aeronautics and Space Administration

NIFTi Natural human-robot cooperation in dynamic environments

NMEA National Marine Electronics Association

NRTK Network Real Time Kinematic

NTRIP Networked Transport of RTCM via Internet Protocol

NUC Next Unit Computer

PPM Pulse Position Modulation

PRC PseudoRange Correction

PRN Pseudo Random Noise

RC Radio Controlled

RINEX Receiver Independent Exchange Format

RRC Range Rate Correction

RTCM Radio Technical Commission for Maritime Services

RTK Real Time Kinematic

RTR Ready To Run

RX Receive

SAPR Sistema Aeromobile a Pilotaggio Remoto

SAR Search And Rescue

SD Secure Digital

SITL Software In The Loop

SLAM Simultaneous Localization and Mapping

SPB Swift Binary Protocol

TRADR Long-Term Human-Robot Teaming for Disaster Response

TTFF Time To First Fix

TX Transmit

UART Universal Asynchronous Receiver-Transmitter

UAV Uncrewed Aerial Vehicle

 $\mathbf{UDP}\quad \mathrm{User\ Datagram\ Protocol}$

UGV Uncrewed Ground Vehicle

USB Universal Serial Bus

UTC Coordinated Universal Time

UTM Universal Transverse of Mercator

VR Virtual Reality

 \mathbf{WGS} World Geodetic System

Elenco delle figure

1.1	Prima dimostrazione di un'automobile radiocomandata - $UGV[46]$.	17
1.2	Primo veicolo radiocomandato - UAV[37]	17
1.3	Costo totale di un UGV [5]	22
1.4	Controllori di volo compatibili con ArduPilot	23
1.5	Segmenti dei sistemi GNSS	28
1.6	Sequenza di generazione del segnale GPS	31
1.7	Effetti della legge di Snell sul percorso del segnale[33]	34
1.8	Centro di fase elettronico dell'antenna e ARP[15]	36
1.9	Differenze tra segnale diretto e indiretto[33]	37
1.10	Determinazione del tempo di volo del segnale[33]	39
1.11	Rappresentazione della portante sinusoidale[8]	41
1.12	Misura di fase agli istanti t_0 e $t[8]$	42
1.13	Schema del posizionamento differenziale[15]	43
1.14	Componenti software del protocollo NTRIP[45]	46
0.1	m vn	40
2.1	Traxxas X Maxx	49
2.2	Dimensioni globali del veicolo	50
2.3	Pixhawk 1	52
2.4	Piksi Multi, fronte (sinistra) e retro (destra)	54
2.5	Schermata di configurazione della GCS Mission Planner	56
2.6	Schermata di Mission Planner con il simulatore SITL in esecuzione	57
2.7	Intestazione (<i>Header</i>) del protocollo UDP[30]	60
2.8	Composizione dei bytes del messaggio MAVLink	61
2.9	Struttura del messaggio	61
2.11	Supporto delle periferiche hardware	63
2.10	Schema di funzionamento del rover durante le missioni	64
	Dettaglio dei connettori sul Pixhawk	66
	Collegamento modulo GNSS secondario - Pixhawk	67
	Radio di telemetria	69
	Schermata installazione Firmware	70
	Posizioni di calibrazione dell'accelerometro	71
2.17	Collegamento modulo GNSS primario - Pixhawk	75

Elenco delle figure

	Led Piksi Multi	79
2.19	Significato stato led Piksi Multi	79
2.20	Prima realizzazione prototipo UGV	81
2.21	Prototipo UGV realizzato con $Tersus\ BX316$	82
2.22	Dettaglio indicatori luminosi LED prototipo UGV realizzato	83
3.1	Chiese di Sant'Agostino e San Francesco ad Amatrice a seguito del	
	terremoto del 24 agosto 2016	86
3.2	Parametri di guida globali	88
3.3	Definizione della traiettoria mediante waypoint	90
3.4	Ingrandimento di una ripresa aerea, ottenuta da quadricottero, uti-	
	lizzata per lo studio del percorso	91
3.5	Traiettoria iniziale per il raggiungimento della chiesa di San Francesco	92
3.6	Risultato della prima simulazione	93
3.7	Conversione delle coordinate tramite ConveRgo	95
3.8	Discretizzazione della traiettoria pianificata	96
3.9	Confronto delle traiettorie percorse (viola) rispetto alla pianificata	
	(giallo) al variare del parametro FF	97
3.10	Modifica di una curva con aggiunta di waypoint in ingresso e uscita	99
	Confronto della traiettoria percorsa (viola) rispetto alla pianificata	
	(giallo) al variare del parametro WP_OVERSHOOT	100
3.12	-	102
	Confronto della traiettoria percorsa (viola) rispetto alla pianificata	
		104
3.14	Confronto della traiettoria percorsa (viola) rispetto alla pianificata	
		106
3.15	(0)	108
	Variazione della pendenza del percorso pianificato - Chiesa San Fran-	
		109
3.17	Traiettoria pianificata per la chiesa di Sant'Agostino	110
		111
	Variazione della pendenza del percorso pianificato - Chiesa Sant'A-	
		111
3.20	Rilievo 3D della chiesa di Sant'Agostino, acquisito con Laser Scanner	
	dal gruppo di ricerca del Laboratorio Fotogrammetria, Geomatica e	
		112
3.21	r i	113
		114
	•	114
	1 1	115
4.1	Comparazione della traiettoria percorsa nello step 1	118

Elenco delle figure

4.2	Comparazione della traiettoria percorsa nello step 2	119
4.3	Comparazione della traiettoria percorsa nello step 3	120
4.4	Comparazione della traiettoria percorsa nello step 4	121
4.5	Comparazione della traiettoria percorsa nello step 5	122
4.6	Comparazione della traiettoria percorsa nello step 6	123
4.7	Comparazione della traiettoria percorsa nello step 7	124
4.8	Influenza del parametro globale WP_OVERSHOOT sulla velocità nell'in-	
	tero tracciato	126
4.9	Influenza del parametro locale DO_CHANGE_SPEED sulla velocità nel-	
	l'intero tracciato	127
4.10	Comparazione della traiettoria registrata AHRS	130
4.11	Comparazione della traiettoria registrata GPS	131
4.12	Comparazione della traiettoria registrata GPS2	132
4.13	Comparazione della traiettoria registrata POS	133
4.14	Comparazione della traiettoria registrata TERSUS	134
4.15	Ingrandimento della traiettoria - Particolare 1	135
4.16	Ingrandimento della traiettoria - Particolare 2	135
4.17	Ingrandimento della traiettoria - Particolare 3	136
4.18	Ingrandimento della traiettoria - Particolare 4	136
4.19	Ingrandimento della traiettoria - Particolare 5	137
	Ingrandimento della traiettoria - Particolare 6	137
4.21	Attraversamento di uno scivolo - Collaudo	140

Elenco delle tabelle

1.1	Caratteristiche dei segnali GPS [33]	30
1.2	Errori nelle misure GNSS	32
1.3	Errori baseline indotti dall'errore d'orbita	33
2.1	Specifiche Pixhawk	51
2.2	Struttura del messaggio SBP	59
2.3	Pinouts GNSS secondario	67
2.4	Pinouts GNSS primario	74
2.5	Parametri per le osservazioni del Piksi - UGV	75
2.6	Parametri Pixhawk per corretta comunicazione con Piksi Multi	76
2.7	Parametri per le osservazioni del Piksi - stazione base	77
2.8	Parametri per l'invio delle correzioni RTK dal Piksi - stazione base	78
2.9	Costi sostenuti per la realizzazione dell' UGV , IVA compresa	80
3.1	Risultati della comparazione relativi alla prima simulazione	96
3.2	Risultati della comparazione numerica relativi alla traiettoria per-	
3.3	corsa a seguito dell'affinamento della velocità di sterzata Risultati della comparazione relativi alla traiettoria percorsa dise-	98
0.0	gnando le curve strette con 3 punti	99
3.4	Risultati della comparazione numerica relativi alla traiettoria percorsa con WP_OVERSHOOT = 0,0	100
3.5	Modifica puntuale della velocità per curve strette	100
3.6	Risultati della comparazione numerica relativi alla traiettoria per-	104
5.0	corsa riducendo la velocità massima in curva a 1.9 m/s	105
3.7	Risultati della comparazione numerica relativi alla traiettoria per-	
	corsa riducendo la velocità massima in curva a 1.0 m/s	106
3.8	Parametri finali assegnati all' UGV	107
3.9	Risultati della comparazione numerica relativi della traiettoria pia-	
	nificata per il rilievo della chiesa di Sant'Agostino	110
4.1	Risultati delle comparazioni numeriche sulle traiettorie percorse -	
	Simulazione	117

Elenco delle tabelle

4.2	Risultati delle comparazioni numeriche rispetto alla traiettoria di	
	riferimento - Collaudo	128
4.3	Risultati delle comparazioni numeriche rispetto alla traiettoria del	
	Tersus BX316 - Collaudo	129

Bibliografia

- [1] 3DR. 3DR. 2020. URL: https://www.3dr.com/.
- [2] 3DR. About. 2020. URL: https://www.3dr.com/company/about-3dr/.
- [3] Chris Anderson. Lego UAV First Flight. 2007. URL: https://www.youtube.com/watch?v=GC2qsOWpL7w.
- [4] ArduPilot. ArduPilot. 2016. URL: https://ardupilot.org/.
- [5] Naval Studies Board, National Research Council et al. Autonomous vehicles in support of naval operations. National Academies Press, 2005.
- [6] F Chiabrando et al. "Multitemporal 3D modelling for cultural heritage emergency during seismic events: Damage assessment of S. Agostino church in Amatrice (RI)". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 42.5/W1 (2017), pp. 69–76.
- [7] A Cina e A Manzino. "Misure GPS in tempo reale: applicabilità e precisioni". In: *Bollettino della SIFET* 4 (2000), pp. 7–20.
- [8] Alberto Cina. Dal GPS al GNSS (Global Navigation Satellite System): per la geomatica. Celid, 2014.
- [9] Fetters Photography & Aerial Cinematography. TRAXXAS X-MAXX PUL-LING ADULT IN A WAGON! 2016. URL: https://www.youtube.com/watch?v=c6HSaiELoDw.
- [10] American Red Cross. Drones for disaster response and relief operations. 2015.
- [11] Geert De Cubber et al. "Search and rescue robotics-from theory to practice". In: (2017).
- [12] Yahoo! Finance. Unmanned Ground Vehicles (UGV) Market to Rise at 14.95% CAGR till 2026; Increasing Number of Company Collaborations Will Provide Impetus to Market Growth, says Fortune Business InsightsTM. 2020. URL: https://finance.yahoo.com/news/unmanned-ground-vehicles-ugv-market-152822747.html.

- [13] Douglas W Gage. UGV history 101: A brief history of Unmanned Ground Vehicle (UGV) development efforts. Rapp. tecn. NAVAL COMMAND CONTROL, OCEAN SURVEILLANCE CENTER RDT e E DIV SAN DIEGO CA, 1995.
- [14] Martial H Hebert, Charles E Thorpe e Anthony Stentz. *Intelligent unman*ned ground vehicles: autonomous navigation research at Carnegie Mellon. Vol. 388. Springer Science & Business Media, 2012.
- [15] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger e Elmar Wasle. GNSS-global navigation satellite systems: GPS, GLONASS, Galileo, and more. Springer Science & Business Media, 2007.
- [16] HS Hopfield. "Two-quartic tropospheric refractivity profile for correcting satellite data". In: *Journal of Geophysical research* 74.18 (1969), pp. 4487–4499.
- [17] Clearpath Robotics Inc. Jackal Umnanned Ground Vehicle. 2020. URL: https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/.
- [18] Technopedia Inc. Communication Protocol. 2012. URL: https://www.techopedia.com/definition/25705/communication-protocol.
- [19] INGV. Tinitaly DEM. 2020. URL: http://tinitaly.pi.ingv.it/.
- [20] Fortune Business Insights. Unmanned Ground Vehicles Market Size, Share and Industry Analysis, By Mobility (Wheeled, Tracked, Hybrid, and Legged), By Size (Small, Medium, Large, Very Large, and Extremely Large), By Mode of Operation (Tethered, Teleoperated, and Autonomous), By System (Payload, Controller System, Navigation System), By Application (Military, Commercial, Law Enforcement, and Federal Law Enforcement) and Regional Forecast, 2019-2026. 2020. URL: https://www.fortunebusinessinsights.com/unmanned-ground-vehicless-market-102525.
- [21] Geert-Jan M Kruijff et al. "Rescue robots at earthquake-hit Mirandola, Italy: A field report". In: 2012 IEEE international symposium on safety, security, and rescue robotics (SSRR). IEEE. 2012, pp. 1–8.
- [22] Ivana Kruijff-Korbayová et al. "Deployment of ground and aerial robots in earthquake-struck amatrice in italy (brief report)". In: 2016 IEEE international symposium on safety, security, and rescue robotics (SSRR). IEEE. 2016, pp. 278–279.
- [23] mRobotics. mRobotics Pixhawk. 2020. URL: https://store.mrobotics.io/category-s/113.htm.
- [24] Ulrich Nehmzow. Robotica mobile: un'introduzione pratica. Springer Science & Business Media, 2008.

- [25] NIFTi. NIFTi returns from earthquake deployment in Italy. 2012. URL: http://www.nifti.eu/news/nifti-returns-from-earthquake-deployment-in-italy.
- [26] Nils J Nilsson. Shakey the robot. Rapp. tecn. SRI INTERNATIONAL MEN-LO PARK CA, 1984.
- [27] Fivo Nine. Traxxas Xmaxx 8S pulling Roller blade's and towing a Truck. 2020. URL: https://www.youtube.com/watch?v=Hw6L09S3aPY.
- [28] Panagiotis Papadakis. "Terrain traversability analysis methods for unmanned ground vehicles: A survey". In: Engineering Applications of Artificial Intelligence 26.4 (2013), pp. 1373–1385.
- [29] B. Parkinson, J. Spilker e P. Enge. Global Positioning System, Vols I and II, Theory and Applications. American Institute of Aeronautics, Reston, 1996.
- [30] Jon Postel et al. "User datagram protocol". In: Internet Engineering Task Force (1980).
- [31] QGIS. QGIS Un Sistema di Informazione Geografica Libero e Open Source. 2020. URL: https://www.qgis.org/it/site/.
- [32] J Saastamoinen. "Contributions to the theory of atmospheric refraction". In: Bulletin Géodésique (1946-1975) 105.1 (1972), pp. 279–298.
- [33] J. Sanz Subirana, J.M. Juan Zornoza e M. Hernández-Pajares. *GNSS Data Processing, Volume I: Fundamentals and Algorithms*. ESA Communications, 2013.
- [34] Thomas B Sheridan. Telerobotics, automation, and human supervisory control. MIT press, 1992.
- [35] Inc. Swift Navigation. About us. 2020. URL: https://www.swiftnav.com/about-us.
- [36] Inc. Swift Navigation. Piksi Multi ArduPilot Integration Guide. 2018. URL: https://support.swiftnav.com/support/solutions/articles/44001850784piksi-multi-ardupilot-integration-guide.
- [37] John William Ransom Taylor e Kenneth Munson. Jane's pocket book of remotely piloted vehicles: robot aircraft today. Collier Books, 1977.
- [38] ArduPilot Dev Team. *History of ArduPilot*. 2020. URL: https://ardupilot.org/rover/docs/common-history-of-ardupilot.html.
- [39] ArduPilot Dev Team. Installing Mission Planner (Windows). 2019. URL: https://ardupilot.org/planner/docs/mission-planner-installation.html#install-mission-planner.
- [40] ArduPilot Dev Team. Mandatory Hardware Configuration. 2019. URL: https://ardupilot.org/copter/docs/configuring-hardware.html.

- [41] ArduPilot Dev Team. MAVLink Basics. 2019. URL: https://ardupilot.org/dev/docs/mavlink-basics.html.
- [42] PX4 Dev Team. *Pixhawk 1.* 2020. URL: https://docs.px4.io/v1.9.0/en/flight_controller/pixhawk.html.
- [43] A Turing. "Computing machinery and intelligence-AM Turing". In: *Mind* 59.236 (1950), pp. 433–460.
- [44] European Union. Un progetto dell'UE impiega i robot dopo il terremoto in Italia. 2016. URL: https://cordis.europa.eu/article/id/120405-eu-project-successfully-deploys-robots-following-italy-earthquake/it.
- [45] Georg Weber, D Dettmering e H Gebhard. "Networked transport of RTCM via internet protocol (NTRIP)". In: A Window on the Future of Geodesy. Springer, 2005, pp. 60–64.
- [46] "World Wide Wireless: Volume 2 October 1921". In: New York: Radio Corp. of America 2 (1921), pp. 18–19.