



UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Corso di Laurea
in Ingegneria Informatica

Relazione Finale

**Simulazione di guida cooperativa
con controllori longitudinali eterogenei**

Relatori: Prof. Renato Lo Cigno
Dott. Lorenzo Ghio

Laureando:
Marco Franceschini
Matricola n. 727808

Anno Accademico 2021/2022

Sommario

La guida cooperativa, ovvero veicoli autonomi che collaborano esplicitamente attraverso lo scambio di messaggi su una rete di telecomunicazione dedicata espressamente allo scopo (V2X), è uno degli ambiti di ricerca e sviluppo più promettenti nell'ambito della mobilità sostenibile. Una delle applicazioni di guida cooperativa più studiata e più matura da un punto di vista tecnologico e di sistema è il Cooperative Adaptive Cruise Control (CACC), noto anche con il termine *platooning*. Un platoon consiste in un gruppo, o treno, di veicoli, dove il *leader* guida effettivamente il veicolo, mentre gli altri seguono la traiettoria del leader a una distanza nettamente inferiore alla distanza di sicurezza necessaria per i guidatori umani e anche di quella consentita dai normali ACC, che, in assenza di comunicazioni, devono fare affidamento solamente sulle misure di distanza, velocità e accelerazione relativa con il veicolo di fronte. In alternativa al modello in cui il primo veicolo (leader) è di fatto autonomo e comanda il comportamento del platoon, il platoon stesso può seguire una traiettoria (profilo di posizione e velocità nel tempo) comunicato a tutti i veicoli da un sistema esterno di coordinamento globale del traffico. Il vantaggio della guida cooperativa e del platooning in particolare risiede soprattutto nell'aumento di sicurezza sulle strade (si elimina l'errore umano), nell'uso più efficiente delle infrastrutture (veicoli più vicini occupano meno spazio effettivo), nella riduzione dei consumi e dell'inquinamento grazie alla diminuzione della resistenza dell'aria e infine in un minore stress per il guidatore, che diventa effettivamente un passeggero. I sistemi di CACC non sono ancora disponibili sul mercato, soprattutto per la mancanza di reti veicolari standard e disponibili sui veicoli commerciali.

Ogni CACC si differenzia dagli altri per la scelta dell'algoritmo di controllo dei veicoli e delle informazioni utilizzate per realizzarlo, inclusa la cosiddetta *topologia di controllo* che definisce, per ciascun veicolo, quale collegamento logico realizza con gli altri veicoli e quindi di quali informazioni dispone sul platoon stesso. Gli studi sul CACC con sistemi di controllo omogenei, ovvero quelli in cui tutti i veicoli del platoon utilizzano lo stesso sistema di controllo, con eventuali variazioni per il primo e l'ultimo veicolo per le diverse condizioni contorno. Questo è probabilmente uno dei motivi per cui l'industria automobilistica non ha fin'ora mostrato grande entusiasmo per questi sistemi, perché costringono a definire, standardizzare e usare tutti lo stesso sistema. Per questo motivo diventa molto interessante lo studio di *platoon eterogenei*, in cui ogni veicolo usa il suo algoritmo di controllo, coordinato, ma indipendente da quello degli altri

veicoli nel platoon. A oggi questa opzione resta sostanzialmente inesplorata, sia a livello teorico (sistemi multi-corpo che non seguono un unico modello di controllo) sia a livello sperimentale o simulativo.

L'obiettivo della tesi è quello di studiare l'interazione di sistemi di controllo differenti in *platoon* misti. Partendo dall'unico studio effettuato in merito all'argomento, sono state eseguite delle simulazioni utilizzando tre diversi sistemi di controllo, di cui due ben noti e dall'interazione già studiata, e uno con una topologia di controllo diversa, sviluppato specificatamente per la sua robustezza alla perdita di informazione tra i veicoli e per poter seguire una traiettoria definita da un eventuale sistema globale di ottimizzazione. I risultati presentati, ovviamente preliminari, indicano come la topologia di controllo sia un fattore chiave per la possibile integrazione di diversi controllori all'interno dello stesso platoon. Allo stesso tempo, simulazioni in presenza di forti disturbi di traffico (variazione della velocità del platoon), indicano che la sicurezza dei passeggeri è garantita anche nelle situazioni più critiche, almeno nel caso dei tre controllori presi in considerazione.

Indice

1	Introduzione	5
1.1	Scopo e obiettivi della tesi	7
2	Guida cooperativa e Platooning	9
2.1	Guida cooperativa	9
2.2	Platooning	10
3	I controllori utilizzati	12
3.1	String Stability	14
3.2	Adaptive Cruise Control (ACC)	14
3.3	Ploeg	15
3.4	PATH	16
3.5	Giordano	17
3.6	Lag di attuazione	19
3.7	Controllori eterogenei in un platoon	19
4	Gli strumenti di simulazione	22
4.1	SUMO	22
4.2	Veins	23
4.3	PLEXE	24
4.3.1	Modelli di sistemi di controllo in SUMO	24
4.3.2	Protocolli e applicazioni per platooning in Veins	25
4.4	Modifiche agli strumenti	25
4.4.1	Modello di Giordano	26
4.4.2	Protocolli e applicazioni per platoon eterogenei	27
4.4.3	Test di verifica	27
5	Risultati degli esperimenti	30
5.1	Obiettivi	30

5.1.1	Considerazioni iniziali	30
5.2	Scenari	31
5.3	Parametri	32
5.4	Risultati delle simulazioni	33
5.4.1	Risultati a velocità costante	33
5.4.2	Risultati in presenza di un disturbo sinusoidale	38
5.4.3	Risultati frenata di emergenza	42
5.5	Considerazioni	44
5.5.1	Possibili modifiche	45
5.6	Analisi parametrica	45
5.7	Risultati modifica parametri	46
6	Conclusioni	49

Capitolo 1

Introduzione

Il trasporto su strada è ormai diventato parte essenziale della vita quotidiana. Basti pensare che i veicoli totali utilizzati nel 2020 solo all'interno dell'Unione Europea si aggirano intorno ai 285 milioni [1] (dati provenienti da ACEA European Automobile Manufacturers' Association). Considerato l'elevato numero di automobili bisogna valutare i numerosi contro che la loro circolazione comporta, e cercare, grazie anche all'innovazione tecnologica, di risolverli. Le principali problematiche sono dovute soprattutto a incidenti, traffico intenso e inquinamento sempre più in crescendo.

Nel 2020, come riportato nella figura 1.1, il numero di vittime per incidenti stradali negli Stati Uniti risulta essere quasi 43.000, circa 13 ogni 100.000 abitanti [2]. Il totale a livello globale di vittime è mediamente di 1.3 milioni annui [3]. La maggior parte di essi è causata da motivi strettamente legati all'errore umano, come guida in stato di ebbrezza, alta velocità, distrazioni alla guida e colpi di sonno.

Un altro problema essenziale è l'inquinamento a causa del traffico stradale. In particolar modo dal 1990 al 2019 le emissioni di gas serra dovute al solo settore dei trasporti sono aumentate del 3.9%, rappresentando nel 2019 il 23.4% delle emissioni nazionali totali di CO₂ [4]. Le zone di traffico intenso con continue accelerazioni e decelerazioni sono le principali responsabili del grande aumento di emissioni di gas serra, oltre che dell'elevato consumo di carburante e in prospettiva di minore autonomia dei veicoli elettrici.

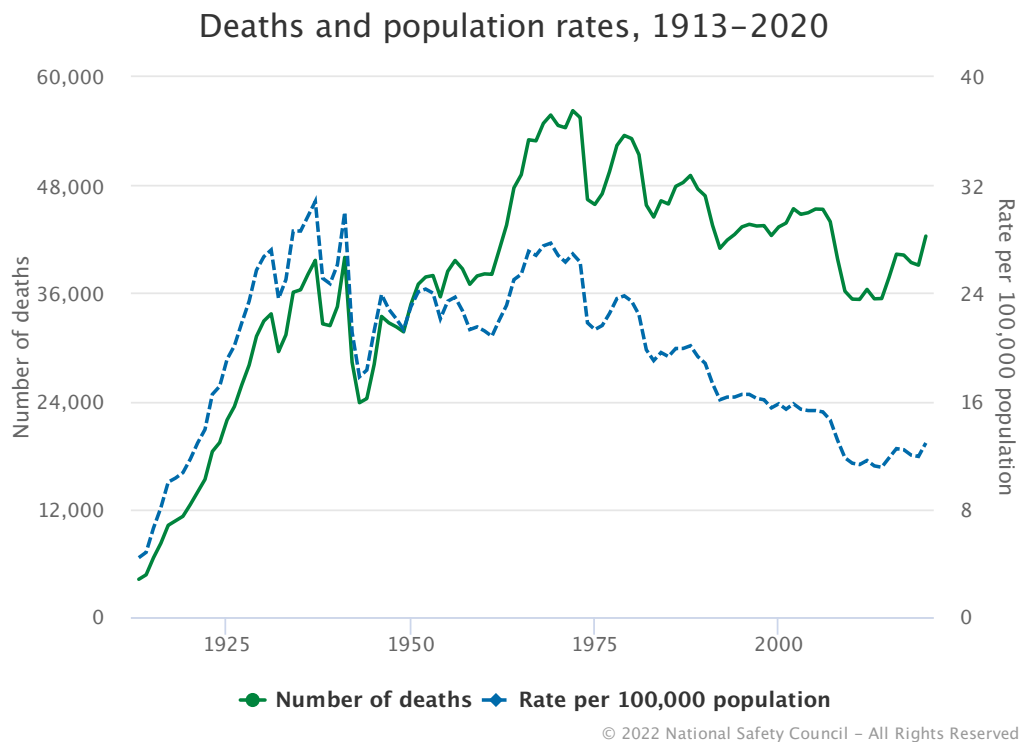


Figura 1.1: Vittime annuali per incidenti stradali negli Stati Uniti confrontati con le vittime ogni 100.000 abitanti

Per ridurre le zone di traffico intenso c'è bisogno di ridurre la congestione delle strade. Il problema principale si ha quindi nel momento in cui il numero di veicoli per secondo che attraversa un tratto di strada supera la sua capacità [5]. L'unica idea elementare per aumentare il fluire dei veicoli e diminuire la congestione delle strade è quella di ridurre la distanza inter-veicolare, cioè aumentare la porzione di strada utilizzabile [6]. In questo modo riusciremmo ad ottenere un netto miglioramento nella gestione del sistema stradale, aumentandone la capacità.

Lo spazio tra veicoli ha un'influenza anche dal punto di vista aerodinamico. Infatti per via del fenomeno fisico della resistenza fluidodinamica le aree a bassa pressione in coda al veicolo, che agiscono in direzione opposta al movimento, ne determinano un maggior consumo di carburante. La dissipazione di energia per via della resistenze aerodinamica ad alta velocità è circa del 20%. Ridurre lo spazio tra i veicoli permetterebbe quindi di diminuire le perdite energetiche [7], [8]. Più è grande il numero di veicoli allineati con brevi distanze, maggiore è il risparmio di carburante. Da uno studio del progetto SARTRE [9] risulta come

con distanze dai 5 agli 8 m tutte le macchine appartenenti al gruppo risparmiano benzina, in particolar modo chi segue il veicolo di testa risparmia fino al 16% del carburante, mentre il leader ne risparmia circa l'8%. Oltre alla riduzione dei consumi, l'allineamento di più veicoli con brevi distanze comporta anche una diminuzione delle emissioni di CO₂. Con 10 m di distanza e un tasso del 40% di veicoli allineati, la riduzione delle emissioni di gas serra è stata studiata essere del 2.1% [10].

La riduzione della distanza di sicurezza non può però essere attuata da un conducente umano, in quanto ad esso non è concesso superare i limiti stabiliti dalla normativa. C'è bisogno quindi di un sistema controllato da computer che sia sicuro ed efficace. Oltre a diminuire i consumi di carburante e massimizzare l'utilizzo delle strade, un sistema elettronico in grado di guidare un veicolo riuscirebbe a ridurre il numero di incidenti e aumentare la sicurezza stradale, eliminando la componente legata all'errore umano [11].

1.1 Scopo e obiettivi della tesi

I sistemi elettronici in grado di assistere la guida di un veicolo sono denominati come ADAS (Advanced Driver Assistance Systems). Il loro obiettivo principale è quello di ridurre gli incidenti su strada e diminuire la pericolosità dei sinistri che non si possono evitare. Anche i sistemi in grado di guidare un veicolo rientrano in questa categoria. Negli anni, grazie all'innovazione tecnologica, ne sono stati sviluppati diversi, tra sistemi di guida autonoma e cooperativa. La tecnologia su cui questi sistemi poggiano è alla base di numerosi studi. Basti pensare come nel tempo si è passati dall'utilizzo dei soli sensori a bordo di un'automobile, per implementare un sistema di guida autonoma, sino all'idea di sfruttare la comunicazione tra veicoli per ridurre le distanze inter-veicolari e consentire una guida cooperativa [12]. Sulla base di quest'ultima sono stati studiati diversi sistemi, tanto più che nel 2011 e nel 2016 è stata realizzata una competizione inerente all'ambito, per lo sviluppo di nuove idee, la Grand Cooperative Driving Challenge [13]. Alcuni di questi sistemi verranno presentati in dettaglio nel capitolo 3.

La presenza di molte tipologie diverse di sistemi ADAS per la guida cooperativa ha reso per ora impossibile la scelta di uno standard da mettere in commercio. Tuttavia la continua ricerca consentirà anche in futuro lo sviluppo di sistemi con caratteristiche diverse rispetto a quelli già esistenti. Proprio per questo una possibile idea è quella di consentire il commercio di ADAS

per guida cooperativa con logiche di funzionamento e topologie comunicative anche differenti, in modo da affiancare il continuo progredire tecnologico. La problematica principale però sarebbe capire come diversi sistemi di controllo possano coesistere. Un'alternativa alla questione, che non verrà analizzata in questa tesi, è l'utilizzo di un algoritmo di consenso distribuito [14].

Un'analisi teorica del problema risulta essere complicata. L'unico studio sperimentale già esistente è stato realizzato sulla base di due sistemi di controllo con topologia comunicativa compatibile [15]. L'obiettivo della tesi è, attraverso un'analisi sperimentale, studiare l'interazione di diversi sistemi di guida cooperativa con comportamenti anche radicalmente opposti.

Capitolo 2

Guida cooperativa e Platooning

L'idea di un veicolo controllato da un sistema elettronico è da anni ormai alla base di numerosi studi che hanno prodotto un sistema di supporto alla guida ADAS (Advanced Driver Assistance Systems) denominato Cruise Control (CC) e già sul mercato dal 1955. Il Cruise Control è un tuttavia un sistema che consente solo il mantenimento di una velocità desiderata, ma non accelerazioni o decelerazioni a seguito dell'andamento di un veicolo di fronte.

Un grande passo verso il raggiungimento di una guida sicura ed efficiente è stato realizzato grazie all'introduzione degli Adaptive Cruise Control (sezione 3.2) nel mercato. Gli Adaptive Cruise Control oltre a mantenere una velocità desiderata sono anche in grado di tenere una distanza di sicurezza dagli altri veicoli. Sfruttando un sensore a bordo come un radar o una telecamera, esso è in grado di rilevare un veicolo frontale e decelerare per evitare collisioni. Tuttavia essendo questa tipologia di sensori lenta nel fornire informazioni, il sistema si regola con un tempo di reazione equiparabile all'essere umano e quindi la distanza di sicurezza risulta essere simile a quella mantenuta in norma dai conducenti umani. Per questo gli Adaptive Cruise Control rappresentano una buona base di partenza per la riduzione delle distanze inter-veicolari e la guida cooperativa, ma sicuramente non un punto di arrivo.

2.1 Guida cooperativa

La componente chiave per realizzare l'obiettivo di ridurre le distanze tra veicoli è la comunicazione inter-veicolare (IVC). Grazie ad essa le macchine sono in grado di comunicare informazioni tra di loro e di cooperare. La condivisione di informazioni circa la dinamica di ogni veicolo è in grado di aumentare

la reattività del sistema, rendendo così possibile la riduzione delle distanze inter-veicolari mantenendo la cosiddetta *stabilità di stringa* (si veda la sezione 3.1).

Sfruttando le potenzialità delle IVC i veicoli sono in grado di creare una rete di informazioni in modo tale da monitorare gli stati di tutti gli altri, e aumentare l'efficacia e la sicurezza del sistema. I Cruise Control adattivi equipaggiati con la tecnologia di comunicazione sono detti Cooperative Adaptive Cruise Control (CACC). Un sistema CACC è caratterizzato da un algoritmo di controllo distribuito, che ne determina le caratteristiche di sicurezza, efficienza e anche comfort di viaggio. Alcuni Cooperative Adaptive Cruise Control verranno analizzati nel capitolo 3.

Alcuni studi presentati nell'articolo [16], mostrano come i veicoli basati su sistemi intelligenti e muniti di comunicazioni riescano a ridurre il numero di incidenti e a migliorare il fluire del traffico. Infatti in ulteriori studi è dimostrato come a peggiorare il tasso di incidenti dovuti a sistemi intelligenti è in realtà l'instabilità degli ACC, mentre l'introduzione della parte comunicativa riduce di molto il rischio di incidenti, come riportato in [17].

2.2 Platooning

I CACC hanno come obiettivi sicurezza, risparmio del carburante e gestione del traffico e, associandoli ad applicazioni globali di coordinamento portano a sistemi generalmente indicati con il termine *platooning*. L'idea del platooning è quella di organizzare i veicoli in gruppi, chiamati appunto platoon, nei quali il primo conducente, il leader, guida (o in maniera autonoma oppure è un guidatore professionista), mentre gli altri seguono con una distanza ridotta. In questo modo, sfruttando anche le IVC, si è in grado di ridurre la distanza inter-veicolare, e quindi gestire al meglio il fluire del traffico e il risparmio del carburante. Due ulteriori vantaggi sono la riduzione del numero di incidenti, in quanto un errore del sistema elettronico è molto meno probabile di un errore umano, e un'esperienza di guida meno stressante.

Per consentire l'applicazione del platooning c'è bisogno di varie componenti. La prima sono i sistemi di controllo, cioè i sistemi che guidano i veicoli in maniera autonoma. Il sistema di controllo si divide in due parti, quella longitudinale e quella laterale. La componente laterale si occupa di sterzare, mentre quella longitudinale, di interesse in questa tesi, si occupa di accelerare e decelerare.

L'altra componente fondamentale per un platoon è l'IVC, senza la quale non sarebbe possibile ridurre la distanza inter-veicolare. Grazie alla comunicazione i veicoli scambiano informazioni sul proprio stato, come velocità, distanza e accelerazione. Queste vengono trasmesse a tutti i veicoli del platoon con una certa frequenza e una velocità maggiore rispetto all'azione di sensori come radar o telecamere. Ogni veicolo del platoon sfrutta, in base al sistema di controllo utilizzato e alla sua topologia comunicativa, alcune di queste informazioni. Tale sistema regola il comportamento del veicolo su cui è installato sulla base delle informazioni ricevute, in maniera molto più rapida rispetto a un Adaptive Cruise Control.

Capitolo 3

I controllori utilizzati

In questo capitolo verranno analizzati i sistemi di controllo alla base di tecnologie ADAS (Advanced Driver Assistance Systems) per consentire la guida cooperativa, prima da un punto di vista generale e poi entrando nei dettagli di quattro diversi tipi di controllo usati in questa tesi.

Un sistema di controllo ha il compito di definire alcuni parametri e ingressi per regolare il sistema controllato, ovvero raggiungere uno specifico obiettivo di stabilità e prestazioni. In particolare i controllori utilizzati per il platooning hanno come obiettivi principali: sicurezza, comfort di viaggio, massimizzazione dell'utilizzo delle strade e gestione del carburante.

Un sistema ADAS già disponibile su molti veicoli è l'ACC (Adaptive Cruise Control). Esso è considerato la diretta evoluzione del Cruise Control (CC). Un veicolo munito di CC è in grado di mantenere una velocità prefissata dall'utente, ma non di rallentare o accelerare in funzione del comportamento del veicolo davanti, o di realizzare decelerazioni di emergenza in caso di presenza di ostacoli. L'ACC invece, grazie alla presenza di un radar a bordo, è in grado di regolare la velocità del veicolo su cui è equipaggiato mantenendo automaticamente una distanza di sicurezza dai veicoli che lo precedono. I controllori ACC hanno prestazioni limitate quando operano in serie –una fila di veicoli tutti controllati da un ACC, ma come vedremo, possono essere migliorati mediante la comunicazione esplicita tra veicoli.

Nelle comunicazioni veicolari i nodi comunicanti possono essere sia i veicoli che le infrastrutture fisse chiamate RSU (Road Side Unit). In questo modo i nodi possono scambiarsi informazioni quali lo stato del traffico, avvisi di sicurezza, ma anche dati relativi al proprio veicolo, come accelerazione, velocità e distanza. Entrambi i nodi utilizzano comunicazioni short-range (DSRC) con protocolli e standard dedicati. Molto comune nelle comunicazioni veicolari è

l'utilizzo dello standard IEEE 802.11p.

Tra le comunicazioni veicolari si possono identificare le seguenti tipologie in base agli agenti coinvolti nella comunicazione:

- V2V (Vehicle to Vehicle): comunicazioni tra soli veicoli;
- V2I (Vehicle to Infrastructure) comunicazioni tra veicoli e infrastrutture fisse;
- V2X (Vehicle to Everything) integrazione di entrambe le tipologie precedenti;

Attraverso moduli di trasmissione e ricezione wireless, i veicoli sono quindi in grado di comunicare tra di loro, allo scopo di realizzare applicazioni il cui fine è fornire supporto alla guida.

Il Cooperative Adaptive Cruise Control (CACC) è definito come estensione dell'ACC, in cui i veicoli sono equipaggiati con tecnologie di comunicazione V2X.

Lo scambio di informazioni tra veicoli all'interno di un platoon può avvenire seguendo diverse *topologie logiche di controllo*, in base alla tipologia di CACC utilizzato. La topologia logica di controllo non va confusa con la topologia fisica delle comunicazioni, che molto spesso sono di tipo broadcast, in cui quindi tutti i veicoli ricevono i dati da tutti gli altri. I dati scambiati includono sempre posizione, velocità e accelerazione, oltre ad altri dati che variano da un controllore ad un altro.

Le principali topologie logiche di controllo analizzate in questa tesi sono:

- Predecessor Following: le informazioni sono trasmesse a cascata a partire dal leader fino all'ultimo veicolo. In questo modo ogni veicolo del platoon utilizza le informazioni esclusivamente di quello davanti;
- Predecessor-Leader Following: oltre alle informazioni trasmesse a cascata ogni veicolo ha a disposizione le informazioni del leader, che si aggiungono a quelle del veicolo frontale;
- Bidirectional: ogni veicolo riceve informazioni sia dal veicolo successivo che dal veicolo che segue. In questo modo ogni elemento del platoon sfrutta i dati sia del veicolo frontale che di quello precedente, eccezion fatta per il leader e l'ultimo veicolo;

Hanno recentemente attirato molta attenzione anche sistemi basati sulla teoria del consenso [18]–[20], in cui la topologia stessa è parte del progetto del controllore, e non un vincolo esterno, ma questi non verranno considerati in questa tesi.

I sistemi di controllo CACC utilizzati in questa tesi sono: Ploeg [21] (3.3), PATH [22] (3.4) e Giordano [23] (3.5).

La scelta dei sistemi di controllo è ricaduta su questi tre in quanto i primi due sono molto conosciuti e utilizzano una topologia di controllo che sfrutta solo le informazioni dei veicoli davanti. Giordano è invece estremamente diverso poiché utilizza una topologia basata anche sui veicoli precedenti, il che lo rende un sistema di controllo di difficile concezione per le normali abitudini di guida.

3.1 String Stability

La proprietà fondamentale di un sistema di controllo per platoon è che sia *string stable*. La string stability è definita in [24] come la capacità di un veicolo, appartenente al platoon, di attenuare i disturbi mentre si propagano lungo la sequenza di veicoli. Al contrario della system stability, che valuta lo stato del sistema nel tempo, la string stability si focalizza quindi sulla propagazione dello stato tra i sottosistemi.

Per esempio supponendo che il leader di un platoon, con distanze intra-veicolari di 5 m, oscilli in velocità e il veicolo successivo non riesca a seguirne il comportamento, allora la distanza non sarà di 5 m ma oscillerà all'interno di un intervallo, che supponiamo essere 4-6 m. Se il sistema è string stable anche tra il secondo e il terzo veicolo l'intervallo di distanza verrà mantenuto, in caso contrario significa che il disturbo si sta propagando. In una platoon di lunghezza infinita la propagazione del disturbo potrebbe portare a due veicoli troppo vicini.

La string stability è quindi una proprietà fondamentale, in grado di evitare collisioni. Essa dev'essere rispettata anche in presenza di altri disturbi come accelerazioni o decelerazioni.

3.2 Adaptive Cruise Control (ACC)

L'ACC sfrutta il radar a bordo per misurare la distanza dal veicolo precedente. Un sistema ACC è quindi autonomo, non dipende da comunicazioni wireless o cooperazione con altri veicoli sulla strada.

L'unico obiettivo dell'Adaptive Cruise Control è mantenere una distanza di sicurezza, regolando la velocità del veicolo su cui è equipaggiato, seguendo la legge del controllore presentata in [25] e definita come:

$$u_i = -\frac{1}{H} (\dot{\epsilon}_i + \lambda \delta_i) \quad (3.1)$$

$$\delta_i = x_i - x_{i-1} + l_{i-1} + H v_i \quad (3.2)$$

$$\dot{\epsilon}_i = v_i - v_{i-1} \quad (3.3)$$

dove u_i rappresenta l'accelerazione desiderata, v_i e v_{i-1} rispettivamente la velocità del veicolo considerato e quello davanti, x_i e x_{i-1} le loro posizioni, l_{i-1} la lunghezza del veicolo di fronte, H è l'headway in secondi e λ un parametro di progetto. δ_i è l'errore della distanza, con $H v_i$ distanza desiderata. $\dot{\epsilon}_i$ è infine la velocità relativa.

L'utente sceglie la velocità desiderata come con Cruise Control (CC). Se il radar rileva un veicolo più lento di fronte, ACC decelera automaticamente sino a mantenere una distanza di sicurezza. Se la strada davanti diventa libera ACC accelera nuovamente sino a raggiungere la velocità prefissata.

Per garantire la string stability un'ACC deve rispettare il seguente vincolo alla 3.1:

$$H \geq 1s \quad (3.4)$$

Ciò comporta elevate distanze tra veicoli all'interno di un platoon, essendo la distanza proporzionale alla velocità. Per esempio alla velocità di 130 km/h è necessario mantenere uno spazio di 36 m.

ACC quindi viola uno degli scopi principali del platooning: la guida ravvicinata.

3.3 Ploeg

Il sistema di controllo progettato da Ploeg e colleghi realizza un sistema CACC il cui scopo è migliorare le prestazioni rispetto a un ACC, mantenendo allo stesso tempo una esperienza di guida simile a un normale ACC in modo da favorire la sua adozione. Dal punto di vista delle prestazioni esso è quindi molto simile a un controllore ACC, ma ne rappresenta una sua versione avanzata.

Sfruttando il sistema di comunicazioni V2X, e in particolar modo una topologia predecessor-following, un veicolo controllato con Ploeg riesce a prevedere le azioni future del veicolo precedente, attraverso l'input al sistema di controllo u_{i-1} . Esso è così in grado di ridurre i tempi di reazione e conseguentemente lo spazio tra i veicoli rispetto ad un ACC.

L'obiettivo principale di Ploeg è mantenere un headway costante e minore rispetto all'Adaptive Cruise Control.

La legge del controllo di Ploeg proposta in [21] è definita come:

$$\begin{aligned} \dot{u}_i = \frac{1}{H} &(-u_i + k_p(x_{i-1} - x_i - l_{i-1} - H v_i) \\ &+ k_d(v_{i-1} - v_i - H a_i) + u_{i-1}) \end{aligned} \quad (3.5)$$

con k_p e k_d parametri aggiuntivi per controllare quanto gli errori di distanza e velocità sono ponderati dal controllore.

Infine l'headway time originariamente proposto nell'articolo è $H = 0.5$ s. In questo modo ad una velocità di 130 km/h la distanza da mantenere non è più di 36 m, ma risulta dimezzata, cioè 18 m.

Così facendo la distanza, seppur inferiore, rimane comunque proporzionale alla velocità, esattamente come per ACC.

3.4 PATH

Un altro approccio ai sistemi di controllo CACC è rappresentato da PATH. Rispetto ad ACC e Ploeg esso mira al mantenimento di uno spazio costante tra i veicoli del platoon, indipendentemente dalla velocità di esso.

Il pattern di comunicazione utilizzato è predecessor-leader following. PATH ha quindi il vantaggio di sapere cosa sta accadendo in testa al platoon, ed è in grado di mantenere spazi minori tra i veicoli a prescindere dalla velocità. ACC e Ploeg non avendo a disposizione questa informazione devono mantenere una distanza maggiore per motivi di sicurezza.

PATH è quindi in grado di massimizzare l'utilizzo delle strade adeguatamente ai principi del platooning.

La legge del controllo definita in [22] è la seguente:

$$\begin{aligned} u_i = \alpha_1 u_{i-1} + \alpha_2 u_0 + \alpha_3 (v_i - v_{i-1}) \\ + \alpha_4 (v_i - v_0) + \alpha_5 (x_i - x_{i-1} + l_{i-1} + d_d) \end{aligned} \quad (3.6)$$

in cui

$$\begin{aligned}\alpha_1 &= 1 - C_1; & \alpha_2 &= C_1; & \alpha_5 &= -\omega_n^2 \\ \alpha_3 &= -\left(2\xi - C_1\left(\xi + \sqrt{\xi^2 - 1}\right)\right)\omega_n \\ \alpha_4 &= -C_1\left(\xi + \sqrt{\xi^2 - 1}\right)\omega_n\end{aligned}$$

con C_1 fattore di peso tra l'accelerazione del leader e i veicoli precedenti, ξ rapporto di smorzamento e ω_n larghezza di banda del controllore. Infine d_d rappresenta la distanza inter-veicolare desiderata.

3.5 Giordano

Ai due sistemi di controllo Ploeg e PATH, ben conosciuti e consolidati, è stato scelto di aggiungerne un terzo.

Giordano, presentato in [23], è un controllore distribuito che assicura la string stability. Rispetto ai due CACC presentati precedentemente, con pattern di comunicazioni unidirezionali, la legge del controllo di Giordano sfrutta una topologia bidirezionale, facendo affidamento sulle informazioni sia del veicolo di fronte che di quello posteriore.

Un'ulteriore peculiarità, che lo rende ancor più differente rispetto a Ploeg e PATH, è l'utilizzo di una velocità di riferimento $v(t)$ condivisa tra tutti i veicoli. Essa può essere imposta dal leader del platoon, ma anche decisa da un qualsiasi altro veicolo, oppure presa da una sorgente esterna. Anche il leader segue $v(t)$.

Il sistema di controllo ha la seguente equazione:

$$\begin{aligned}u_i &= -k(x_i - x_{i+1} - d_d) - k(x_i - x_{i-1} + d_d) \\ &\quad -h(v_i - v_{i+1}) - h(v_i - v_{i-1}) - r(v_i - v).\end{aligned}\tag{3.7}$$

con k coefficiente elastico, h coefficiente di attrito inter-veicolare e r coefficiente d'attrito del veicolo di riferimento. La scelta del parametro r è fondamentale per quanto riguarda il comfort di guida e la convergenza alla $v(t)$. Infatti più r viene scelto elevato, più veloce sarà la convergenza alla velocità di riferimento, ma questo rischia di provocare indesiderate accelerazioni. Per questo il parametro va scelto in modo che a_{max} sia compatibile con il comfort del conducente.

Inoltre utilizzando una topologia bidirezionale, l'input del sistema di controllo varia in base alla posizione del veicolo nel platoon e risulta quindi differente rispetto al u_i generico.

Per il veicolo 1 (il leader):

$$u_1 = -k(x_1 - x_2 - d_d) - h(v_1 - v_2) - r(v_1 - v). \quad (3.8)$$

per i veicoli $i = 2, \dots, N - 1$:

$$\begin{aligned} u_i = & -k(x_i - x_{i+1} - d_d) - k(x_i - x_{i-1} + d_d) \\ & -h(v_i - v_{i+1}) - h(v_i - v_{i-1}) - r(v_i - v). \end{aligned} \quad (3.9)$$

e, per il veicolo N :

$$u_N = -k(x_N - x_{N-1} + d_d) - h(v_N - v_{N-1}) - r(v_N - v). \quad (3.10)$$

Rispetto a PATH e Ploeg anche Giordano risulta *string stable*, ma con un approccio molto diverso. Infatti, sotto l'azione di un disturbo sinusoidale, il massimo dell'attenuazione si verifica nel mezzo del platoon con una dinamica simmetrica rispetto al centro.

Come mostrato nell'articolo di presentazione, possiamo tracciare un'analogia tra il sistema di controllo di Giordano e un sistema meccanico molla-ammortizzatore (3.1), dove k rappresenta la costante elastica della molla, h il fattore di smorzamento tra veicoli e r la rigidità della molla.

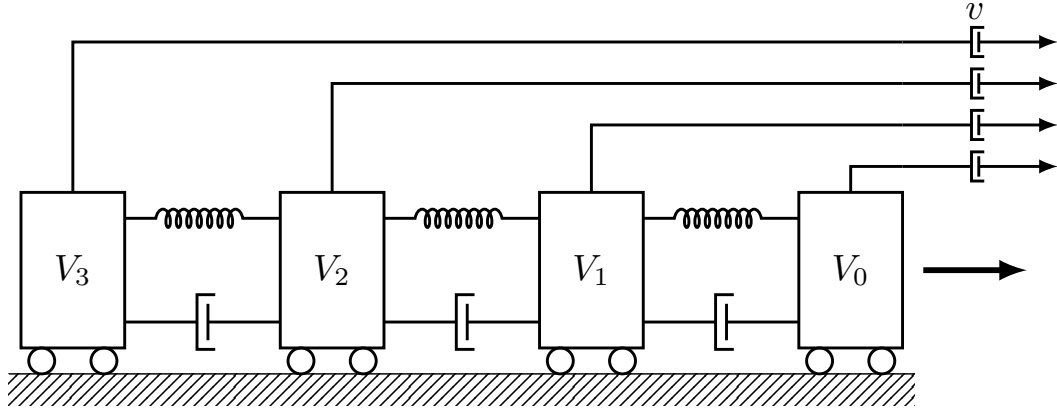


Figura 3.1: sistema molla-ammortizzatore paragonato ad un platoon di veicoli Giordano. Figura esposta nell'articolo di riferimento [23]

3.6 Lag di attuazione

L'accelerazione calcolata mediante le formule dei controllori non è attuata immediatamente, per via di un ritardo di trasmissione delle informazioni dal controllore al motore. Il ritardo di attuazione è modellato come un filtro passa basso del primo ordine. Ciò significa che l'accelerazione applicata al veicolo al passo η è:

$$\ddot{x}[\eta] = \beta \cdot \ddot{x}_{des}[\eta] + (1 - \beta) \cdot \ddot{x}[\eta - 1] \quad (3.11)$$

$$\beta = \frac{\Delta_t}{\tau + \Delta_t} \quad (3.12)$$

Dove τ rappresenta la costante di tempo (di default fissata a 0.5 s), mentre Δ_t è il tempo di campionamento della simulazione.

3.7 Controllori eterogenei in un platoon

Per studiare l'interazione di diversi sistemi di controllo di guida cooperativa bisogna introdurre il concetto di *mixed platoon*, ovvero una stringa di veicoli composta da più di un algoritmo di CACC utilizzato. L'interazione di diversi CACC è di interesse soprattutto per la variazione di sicurezza, comfort di guida e massimizzazione nell'utilizzo delle strade con la progressiva introduzione di algoritmi differenti.

L'unico studio fino ad ora realizzato in merito ai *mixed platoon* è l'articolo [15], basato sull'interazione di PATH(3.4) e Ploeg(3.3). I due sistemi sono tuttavia compatibili dal punto di vista topologico, infatti, come mostra la figura 3.2, sia PATH che Ploeg utilizzano le informazioni solo dei veicoli frontali, l'unica differenza sta nel fatto che PATH usa in più le informazioni del leader del platoon. Per via di questo, in un platoon misto, come mostrato nella topologia 3 della figura, i veicoli con sistema di controllo PATH eleggono a leader da cui ricevere informazioni il primo veicolo non PATH incontrato.

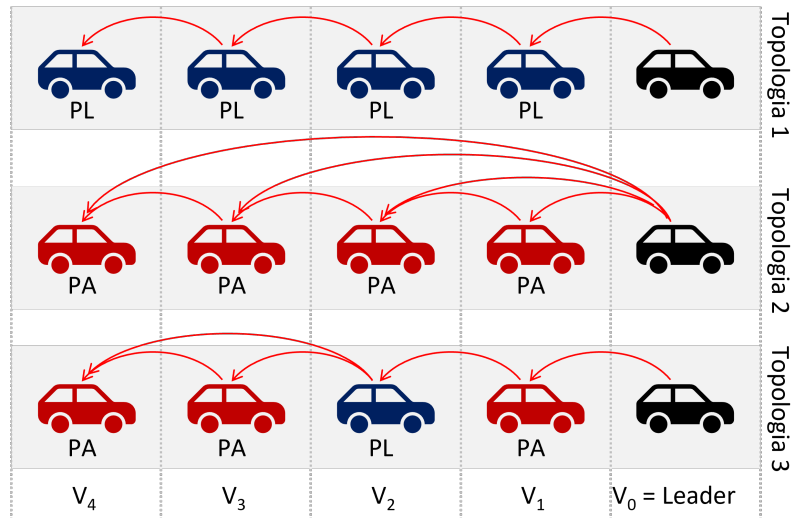


Figura 3.2: Topologia comunicativa di Ploeg (PL) in alto, PATH (PA) in mezzo e di un platoon misto in basso.

Gli studi dell'articolo [15] mostrano infine come PATH e Ploeg insieme in platoon eterogenei mantengano la sicurezza e la massimizzazione nell'utilizzo delle strade rispetto a stringhe composte da un solo algoritmo di controllo, migliorando addirittura i parametri in alcuni casi. Il comfort di guida è invece sempre migliore rispetto all'utilizzo di ACC (3.2).

Visto che due algoritmi con topologia monodirezionale basata sulla *predecessor-following* possono coesistere con buoni risultati, un passo ulteriore potrebbe essere l'introduzione di un sistema di controllo con topologia bidirezionale in platoon misti con i due già studiati.

Giordano (3.5) è un CACC con topologia di tipo bidirezionale che sfrutta le informazioni del veicolo frontale e di quello antecedente, con un riferimento ad una velocità esterna imposta dall'infrastruttura stradale o dal leader del platoon e che si presenta come nella figura 3.3.

In un platoon misto con PATH, figura 3.4 in alto, esso, essendo il primo non PATH incontrato, diventa il leader che fornisce informazioni alla coda del platoon. Esso rispetto al mix tra PATH e Ploeg (figura 3.2) sfrutta anche le informazioni del PATH alle sue spalle. Con Ploeg, figura 3.4 in basso, invece l'unica modifica sta nel fatto che Giordano sfrutta le informazioni del Ploeg antecedente.

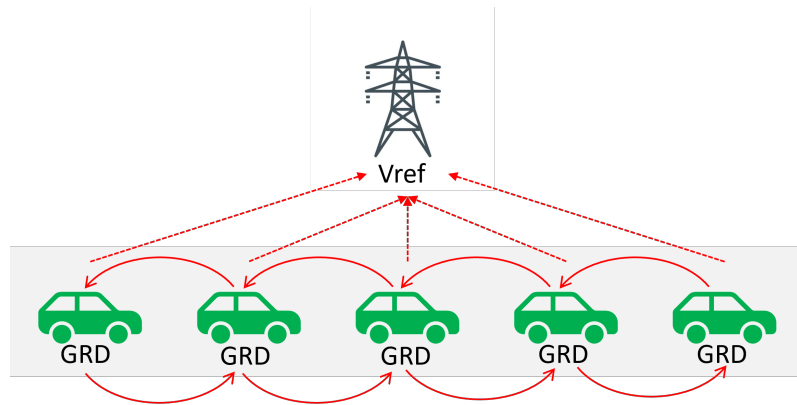


Figura 3.3: Topologia del controllo di Giordano (GRD) con riferimento a una velocità esterna.

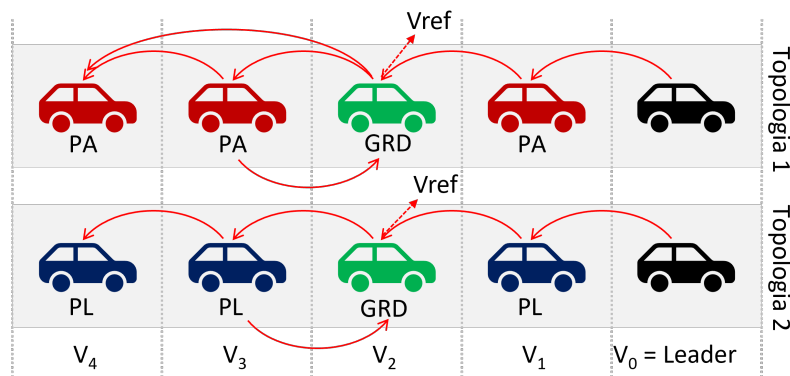


Figura 3.4: Topologia comunicativa di Giordano (GRD) in platoon misto con PATH in alto, e un platoon misto tra Ploeg e Giordano in basso.

Vista la topologia dei *mixed platoon* nella figura 3.3, fare delle previsioni sui risultati risulta essere complicato. Essendo un'analisi in teoria del controllo di un platoon misto ancor più difficile per via della complessa topologia di controllo che si viene a creare, l'obiettivo della tesi è quello attraverso un'analisi sperimentale di verificare le variazioni in sicurezza, comfort e massimizzazione nell'utilizzo delle strade di platoon di soli PATH o Ploeg con un veicolo Giordano al loro interno. La scelta di usare un solo Giordano è data dal fatto che PATH e Ploeg sono due sistemi CACC già conosciuti e studiati, anche dal punto di vista eterogeneo, mentre Giordano essendo meno conosciuto e usando una topologia così diversa potrebbe essere un disturbo per gli altri due algoritmi. Per questo è di interesse in questa tesi capire quanto questo disturbo possa influire sulle performance.

Capitolo 4

Gli strumenti di simulazione

Le simulazioni di platoon eterogenei sono state realizzate grazie all'utilizzo di diversi strumenti coordinati tra di loro. In particolare è stata utilizzata un'infrastruttura di simulatori in grado di replicare il traffico urbano e le reti di comunicazione tra i veicoli.

Per l'emulazione delle reti è stato scelto OMNeT++ [26], un simulatore basato su eventi discreti molto diffuso ed utilizzato. Esso fornisce un modello di programmazione a blocchi simile a quello ISO/OSI, ed è quindi molto intuitivo progettare protocolli di rete cross-layer.

Per il traffico è stato scelto invece SUMO [27] (4.1), accoppiato con OMNeT++ via Veins [28] (4.2), il quale viene poi esteso da PLEXE [29] (4.3) per implementare tutte le funzionalità del platooning.

4.1 SUMO

SUMO [27] (Simulation of Urban Mobility) è un simulatore di traffico microscopico, open source e altamente scalabile, sviluppato in Germania, nell'istituto di ricerca sui trasporti presso il centro aerospaziale tedesco. Esso consente di simulare un dato traffico composto da singoli veicoli che si muovono attraverso una rete stradale. La simulazione consente di affrontare una vasta gamma di argomenti relativi alla gestione del traffico ed è puramente microscopica, cioè ogni veicolo è modellato in modo esplicito, ha un proprio percorso e si muove individualmente attraverso la rete. Le simulazioni sono deterministiche per impostazione predefinita, ma ci sono varie opzioni per introdurre la casualità.

SUMO include una vasta gamma di moduli e programmi necessari per preparare ed eseguire una simulazione del traffico. In particolar modo i due

più utilizzati sono sumo e sumo-gui. Il primo consente di eseguire una simulazione senza visualizzazione, usando solamente la riga di comando, ed è spesso utilizzato insieme a TraCI; il secondo invece consente di simulare il traffico e visualizzarlo mediante un'interfaccia grafica.

TraCI (Traffic Control Interface) dà accesso ad una simulazione del traffico stradale in corso ad applicazioni esterne e permette di recuperare i valori degli oggetti simulati e di manipolarne il comportamento "on-line". Lo scambio di dati tra SUMO e OMNeT++ sfrutta l'interfaccia, ed è basato su una connessione TCP che incapsula i messaggi, composti da set di comandi e risposte.

TraCI è essenziale nelle simulazioni, in quanto consente di assegnare un diverso controllore per ogni veicolo, di impostarne e cambiarne i parametri, e anche di creare complessi scenari con accelerazioni e decelerazioni preimpostate.

4.2 Veins

Veins [28] estende OMNeT++, fornendo uno stack di comunicazione veicolare basato su IEEE 802.11p. Esso è in grado di accoppiare la rete e il simulatore di mobilità, come mostrato nella Figura 4.1, creando un nodo di rete in OMNeT++ per ogni veicolo in SUMO. Ad ogni nodo è associato uno stack che include un'interfaccia di rete IEEE 802.11p, un protocollo di beaconing e una o più applicazioni funzionanti su di esso.

Ogni volta che un veicolo si muove, Veins replica il movimento per il nodo in OMNeT++ corrispondente.

I due simulatori, di rete e di traffico, vengono estesi con un modulo dedicato alla comunicazione. Durante la simulazione i due moduli si scambiano comandi attraverso una connessione TCP. Il set di comandi e risposte scambiati è basato su TraCI, l'interfaccia che SUMO espone.

OMNeT++ è un simulatore basato su eventi, che programma i movimenti di ogni nodo a intervalli regolari. Questo si adatta bene con l'approccio di SUMO, il quale avanza il tempo della simulazione a passi discreti. I moduli integrati in OMNeT++ e SUMO consentono di agire come un buffer che raccoglie tutti i comandi arrivati tra due passi temporali, garantendo il sincronismo tra le due simulazioni.

Veins, utilizzando TraCI, interroga SUMO sullo stato del traffico, chiedendo informazioni come numero di veicoli, posizione e velocità per ognuno di essi, ed è in grado di modificare le dinamiche del traffico, cambiando il percorso di un veicolo o la sua accelerazione.

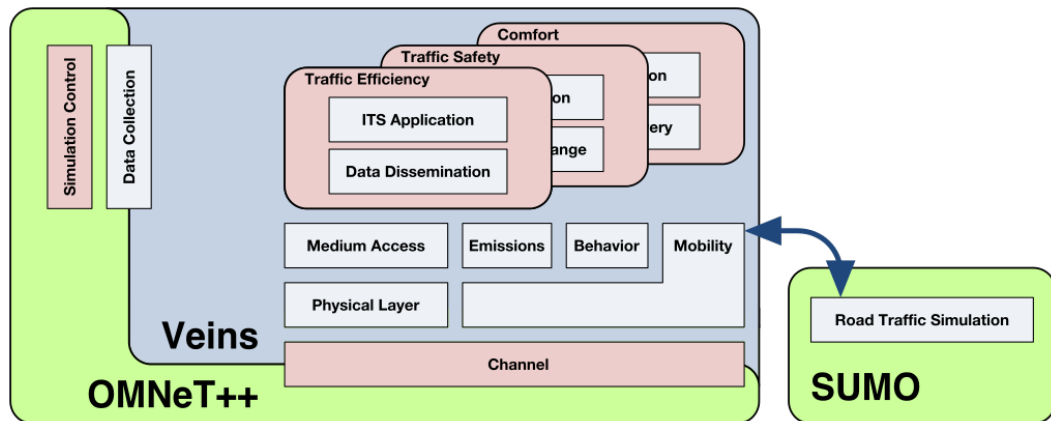


Figura 4.1: Accoppiamento tra OMNeT++ e SUMO tramite Veins (<https://veins.car2x.org>).

4.3 PLEXE

PLEXE [29], [30] è un'estensione di Veins, che permette simulazioni e studi realistici del platooning. Utilizzando le capacità di Veins di simulare sia la rete che il traffico, PLEXE integra tutti i componenti necessari per studiare i platoon di macchine, come i modelli dei sistemi di controllo e le manovre per formare e mantenere i platoon.

PLEXE estende l'interazione tra SUMO e OMNeT++ via TraCI, consentendo di inviare i dati di un veicolo a tutte le altre macchine di un platoon, permettendo così l'utilizzo di protocolli e applicazioni per platooning. È possibile così alimentare i CACC presenti in SUMO utilizzando i dati ricevuti dai veicoli via Veins.

Per consentire le simulazioni di platoon di macchine sono quindi necessarie modifiche ed estensioni da due punti di vista: lato SUMO, per quanto riguarda i modelli dei sistemi di controllo e le manovre di un platoon; lato Veins, per l'implementazione di protocolli e applicazioni per platooning.

4.3.1 Modelli di sistemi di controllo in SUMO

SUMO presenta diversi modelli di movimento per i veicoli, concepiti per mimare il comportamento di un guidatore umano. Alcuni esempi sono IDM e Krauss. La principale modifica apportata da Plexe è l'introduzione di un nuovo modello di controllo chiamato CC (Cruise Controllers) e basato su Krauss. Questo modello di movimento è stato creato apposta per implementare i sistemi di

controllo di guida cooperativa. Per esempio Cruise Control e Adaptive Cruise Control fanno parte di questo modello. Il grande vantaggio di Plexe lato SUMO è l'estensibilità. Il modello è infatti modificabile, permettendo l'introduzione di tutti i CACC desiderati. Per verificare la validità delle modifiche è possibile assegnare diversi comportamenti al leader del platoon, come un'accelerazione costante, una frenata di emergenza oppure un disturbo sinusoidale della velocità, verificano così la *string stability* di un sistema composto dal nuovo algoritmo introdotto.

La versione di SUMO utilizzata comprende, oltre a CC e ACC, anche i modelli di PATH e Pleog, presentati nel Capitolo 3.

Durante la simulazione è possibile scegliere quale sistema controlla la macchina. Infatti, attraverso l'interfaccia TraCI, si può accedere ai modelli implementati da Plexe, e quando un CACC è attivo per un veicolo è possibile inviargli i dati necessari, scambiati via IVC dal simulatore della rete di comunicazione, cioè OMNeT++.

4.3.2 Protocolli e applicazioni per platooning in Veins

Per quanto riguarda Veins, ogni veicolo è fornito di una scheda di rete IEEE 802.11p, un protocollo di base per inviare i messaggi e un'applicazione funzionante sulla loro distribuzione.

L'idea è quella di avere uno strato di protocolli, con radice la classe `BaseProtocol`, che fornisce tutte le funzionalità di base da estendere, e un set di sottoclassi che si concentrano solo sull'implementazione della strategia di beaconing.

Lo stesso principio è utilizzato per lo strato applicativo, dove `BaseApp` è responsabile di caricare i parametri delle simulazioni e passare i dati agli CACC via TraCI. Gli strati sottostanti sono invece incaricati di decidere tutti gli aspetti applicativi di un platoon, come quale veicolo è il leader, la corsia su cui una macchina deve transitare, e molto altro.

4.4 Modifiche agli strumenti

Per consentire le simulazioni di platoon eterogenei, composti dai quattro controllori, presentati nel Capitolo 3, sono state apportate delle modifiche agli strumenti elencati precedentemente. Partendo dalle versioni di Plexe e SUMO utilizzate per [15], già contenenti PATH e Ploeg e abilitate all'utilizzo di platoon

misti, sono state realizzate delle modifiche per consentire in aggiunta anche l'utilizzo di Giordano.

4.4.1 Modello di Giordano

Il modello sviluppato da Plexe in SUMO, Cruise Controllers (CC), è estensibile modificando la classe relativa `MSCFModel_CC`, scritta in `c++`, e implementando il sistema di controllo desiderato. All'interno della classe sono stati inseriti quindi tutti i metodi necessari per l'utilizzo di Giordano dal simulatore di traffico. Tra questi il più importante è sicuramente il metodo che realizza il calcolo dell'accelerazione del controllore. Per la computazione vengono utilizzate le equazioni presentate nel capitolo 3.5, e viene realizzata la distinzione in base alla posizione del veicolo nel platoon come segue:

```

1   if (pos == 0)
2       //leader
3       return -vars->giordanoK * (egoPosition.distanceTo2D(pp) - pd) - vars->
        giordanoH * (v - pv) - vars->giordanoR * (v - v0);
4   else if (pos > 0 && pos < nCars - 1)
5       //middle vehicle
6       return -vars->giordanoK * (-egoPosition.distanceTo2D(fp) + fd) - vars
        ->giordanoK * (egoPosition.distanceTo2D(pp) - pd) - vars->
        giordanoH * (v - fv) - vars->giordanoH * (v - pv) - vars->
        giordanoR * (v - v0);
7   else
8       //last vehicle
9       return -vars->giordanoK * (-egoPosition.distanceTo2D(fp) + fd) - vars
        ->giordanoH * (v - fv) - vars->giordanoR * (v - v0);

```

dove `pp`, `pd`, `vd` rappresentano rispettivamente la posizione, la distanza e la velocità del veicolo precedente, mentre `fp`, `fd`, `fv` quelle del veicolo antecedente; `v0` è invece la velocità di riferimento, con `distanceTo2D` funzione per calcolare una distanza data la propria.

Un'altra importante funzione implementata in SUMO è `computePrediction`. Essendo la frequenza di scambio dei dati tra due veicoli (10 Hz) inferiore a quella dell'anello di controllo a bordo veicolo (100 Hz), i dati forniti all'algoritmo potrebbero essere incoerenti dal punto di vista temporale, è quindi necessario utilizzare un predittore in grado fornire i dati di velocità e posizione mancanti per interpolazione. Essendo \ddot{y}_{t_0} , \dot{y}_{t_0} e y_{t_0} rispettivamente accelerazione, velocità e posizione del veicolo al tempo t_0 , per stimare la velocità e la posizione dello stesso veicolo al tempo t il sistema computa:

$$\dot{y}_t = \dot{y}_{t_0} + \ddot{y}_{t_0}(t - t_0) \quad (4.1)$$

$$y_t = y_{t_0} + \frac{t - t_0}{2}(\dot{y}_t + \dot{y}_{t_0}) \quad (4.2)$$

L'utilizzo delle due equazioni rende la simulazione di PLEXE estremamente realistica.

4.4.2 Protocolli e applicazioni per platoon eterogenei

Per la parte di protocolli e applicazioni è stata modificata, anche in questo caso, la versione di PLEXE utilizzata nell'articolo [15]. In questo modo è bastato inserire i parametri di Giordano nel file `CC_Const.h` contenente tutti i valori scambiati via TraCI con SUMO.

```
1 #define CC_PAR_GIORDANO_H (parameter_prefix + "ccgh")
2 #define CC_PAR_GIORDANO_K (parameter_prefix + "ccgk")
3 #define CC_PAR_GIORDANO_R (parameter_prefix + "ccgr")
4 #define CC_PAR_GIORDANO_V (parameter_prefix + "ccgv")
```

I parametri sono stati introdotti anche nei beacon scambiati tra i veicoli, aggiungendo i pezzi di codice necessari nell'infrastruttura di protocolli e applicazioni, partendo dalle due classi base `BaseProtocol` e `BaseApp`.

Con una piccola aggiunta al `TrafficManager`, creato per la gestione di un platoon misto con `PATH` e `Ploeg`, è ora possibile simulare un platoon eterogeneo anche con Giordano. Per esempio scrivendo nel file `omnetpp.ini` corrispondente alla simulazione prescelta una stringa di controllori scelti come:

```
1 **.traffic.controller = "ACC PATH PLOEG GIORDANO"
```

il sistema assegnerà ad ogni veicolo corrispondente il controllore, partendo dal leader `ACC`.

4.4.3 Test di verifica

Come spiegato nella sezione 4.3.1, una volta inserito un nuovo controllore è necessario testarne il comportamento simulando differenti scenari di guida, come una frenata di emergenza oppure un andamento sinusoidale della velocità del leader.

Per verificare il corretto inserimento del controllore all'interno dell'infrastruttura di simulatori, sono stati testati platoon di solo veicoli Giordano, su diversi scenari. Il primo test è stato eseguito su uno scenario a velocità costante (100 km/h) a 4, 8 e 16 veicoli, su una singola corsia, tutti inseriti ad una distanza maggiore di 5 m, ovvero lo spazio desiderato preimpostato per il sistema di controllo. Il fine della simulazione era verificare la corretta convergenza del

platoon alla distanza desiderata. Come mostrato nella figura 4.2, tutti i veicoli convergono ad un errore di distanza nullo.

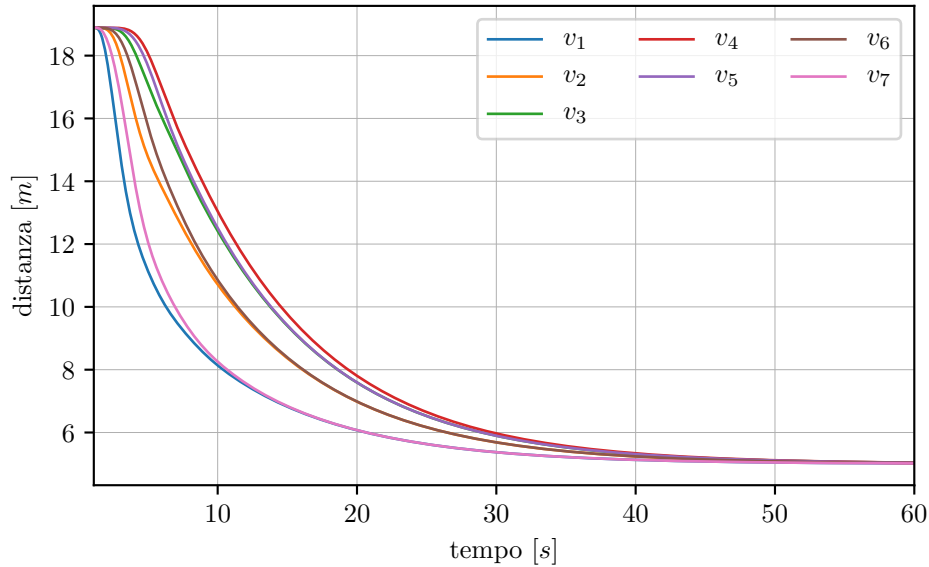


Figura 4.2: Simulazione a velocità costante con 8 veicoli. Nel grafico è rappresentato l'errore di distanza (distanza al tempo t - distanza desiderata) nel tempo.

Il secondo test è stato realizzato con riferimento ai grafici dell'articolo [23], per dimostrare il corretto funzionamento della frenata di emergenza per un sistema di controllo Giordano. La simulazione è stata fatta con 8 veicoli, ad una sola corsia e ad una velocità iniziale di 100 km/h, prima con il parametro r adattato, per ottenere uno stop completo del platoon in maniera più rapida, e poi senza adattamento del parametro. La figura 4.3 mostra come adattando il parametro r lo stop completo del platoon risulta essere molto più efficace e veloce, mantenendo anche la maggiormente la sicurezza, rispetto alla figura 4.4 nella quale invece r non viene adattato. Lo scopo era ottenere i grafici identici come presentati nell'articolo, per avere conferma del fatto che Giordano sia inserito correttamente nei simulatori. Le figure 4.3 e 4.4, se confrontate con l'articolo [23], risultano essere uguali.

In conclusione, si può affermare che il sistema di controllo Giordano utilizzato nelle simulazioni è lo stesso presentato nell'articolo.

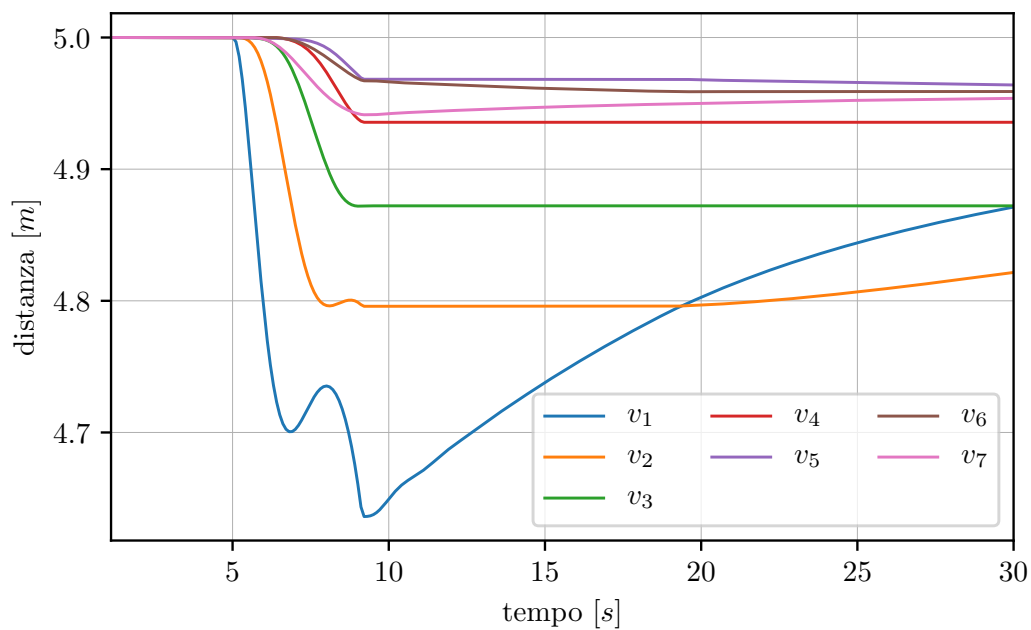


Figura 4.3: Simulazione di una frenata di emergenza a 5 s, con parametro r del controllore Giordano adattato.

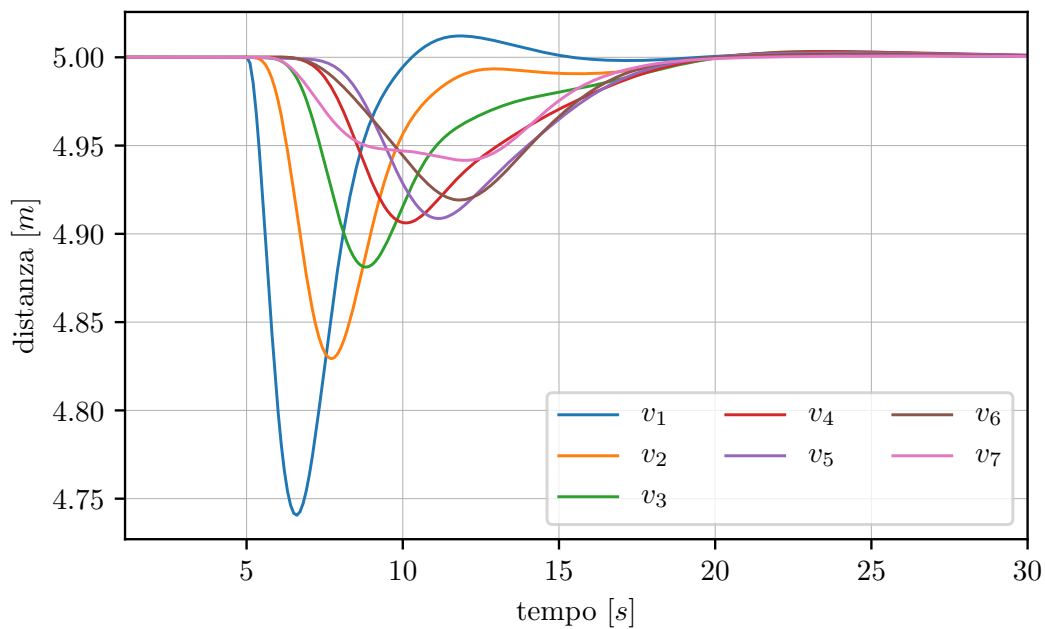


Figura 4.4: Simulazione di una frenata di emergenza a 5 s, con parametro r del controllore Giordano non adattato.

Capitolo 5

Risultati degli esperimenti

5.1 Obiettivi

Sicurezza, comfort di guida e massimizzazione dell'utilizzo delle strade sono gli obiettivi da mantenere e migliorare in un platoon composto da sistemi di controllo eterogenei. Le simulazioni presentate di seguito, sono state realizzate per verificare quanto l'inserimento di un controllore estremamente diverso come Giordano, in platoon di soli PATH o Ploeg, possa modificare gli obiettivi in positivo, o in negativo, e quanti e quali problemi possa causare.

Gli esperimenti terranno in considerazione:

- La sicurezza: dal punto di vista degli incidenti e dell'efficienza delle frenate di emergenza;
- Comfort di guida: dal punto di vista delle oscillazioni in velocità e le conseguenti accelerazioni;
- Massimizzazione dell'utilizzo delle strade: verificando la lunghezza dei platoon e le distanze inter-veicolari;

5.1.1 Considerazioni iniziali

Giordano è un sistema di controllo dalla concezione molto diversa rispetto a PATH e Ploeg. Questo potrebbe portare a difficoltà di integrazione, ma non a incidenti, in quanto tutti e tre i controllori sono string stable. Comfort di guida e massimizzazione delle strade risultano essere invece imprevedibili per via del complesso funzionamento dei parametri di Giordano, e della sua diversa natura.

È inoltre importante capire come la scelta di un determinato algoritmo di controllo possa influenzare la topologia di comunicazione del platoon. Infatti ogni controllore usa solo le informazioni di uno specifico set di veicoli. La figura 3.3 presentata nella sezione 3.7 mostra come la topologia di comunicazione varia, in base alla scelta dei controllori e la loro posizione nel platoon, nei 2 casi. Il platoon in basso è composto da veicoli Ploeg e Giordano, con i veicoli Ploeg che utilizzano solo le informazioni frontali, mentre Giordano utilizza anche i dati dell'ultimo veicolo retrostante. Il platoon ibrido nel mezzo invece ci mostra come i veicoli PATH nelle ultime due posizioni non utilizzino più il leader del platoon per lo scambio di dati, ma eleggano il primo veicolo non PATH come nuovo leader da seguire.

In generale la topologia di comunicazione potrebbe avere dei riflessi importanti sulle prestazioni dei platoon, e gli obiettivi ricercati.

5.2 Scenari

Gli scenari utilizzati nelle simulazioni prevedono un'autostrada in linea dritta con un'unica corsia. Gli esperimenti sono stati realizzati con platoon da 4, 8 e 16 veicoli, numerati da V_0 fino a V_{N-1} , con N dimensione del platoon. Il leader V_0 segue sempre un profilo di velocità indipendente, ed è associato ad un sistema di controllo ACC per evitare incidenti. I veicoli seguenti implementano tutti o solo PATH oppure solo Ploeg, con l'inserimento di un Giordano in ogni posizione del platoon (tranne il leader) per ogni scenario presentato.

La nomenclatura di un platoon sarà indicata tra parentesi graffe a partire da V_i fino a V_{N-1} , usando le sigle PA, PL e GRD, rispettivamente per PATH, Ploeg e Giordano. Per esempio $\{PL, PL, GRD\}$ indica un platoon composto da 4 veicoli con leader ACC e nell'ordine i veicoli: Ploeg, Ploeg e Giordano.

Gli scenari utilizzati per le simulazioni sono 3:

- **Velocità costante:** il veicolo leader segue costantemente una velocità di 100 km/h;
- **Oscillazione:** il veicolo leader segue un andamento oscillatorio, aumentando e diminuendo la sua velocità, con un intervallo di oscillazione di 10 km/h, e una frequenza di 0.2 Hz;
- **Frenata di emergenza:** il veicolo leader, alla velocità di regime di 100 km/h, effettua una frenata di emergenza dopo 5 s di simulazione, con una decelerazione di 8 m/s²;

In tutte le simulazioni i veicoli vengono inseriti direttamente a regime, cioè alla velocità stabilita di 100 km/h e con le distanze desiderate di ogni controllore già impostate. La durata delle simulazioni definita è di 60 s. Inoltre non sono considerate perdite di pacchetti scambiati mediante la comunicazione wireless IEEE 802.11p. Infatti l'interesse delle simulazioni è la coesistenza dei sistemi di controllo, e non la resistenza alle perdite di pacchetti. Infine la frequenza di scambio di messaggi impostata è di 0.1 s.

5.3 Parametri

I parametri impostati per ACC, PATH e Ploeg sono i seguenti:

- $\text{accHeadway}(H)$: Headway time di ACC, impostato a 1.2 s;
- $\text{caccC1}(C_1)$: rappresenta per PATH un fattore di ponderazione tra l'accelerazione del leader e quella del veicolo precedente, è stato impostato a 0.5;
- $\text{caccXi}(\xi)$: è il damping ratio di PATH che descrive come le oscillazioni in un sistema decadono dopo il disturbo (necessariamente maggiore o uguale a 1), è impostato a 1;
- $\text{caccOmegaN}(\omega_n)$: la larghezza di banda di PATH, impostata a 0.2 Hz;
- $\text{caccSpacing}(d_d)$: la distanza desiderata di PATH (usata anche per Giordano) impostata a 5 m;
- $\text{engineTau}(\tau)$: il lag di attuazione (sezione 3.6) scelto a 0.5 s;
- $\text{ploegH}(H)$: Headway time di Ploeg, impostato come nell'articolo di presentazione a 0.5 s;
- $\text{ploegKp}(k_p)$: parametro aggiuntivo di Ploeg per valutare quanto l'errore di distanza è ponderato dal controllore, impostato a 0.2;
- $\text{ploegKd}(k_d)$: parametro aggiuntivo di Ploeg per valutare quanto l'errore di velocità è ponderato dal controllore, impostato a 0.7;

Le scelte seguono le linee guida già presentate nell'articolo [15] e per le quali sono già stati fatti studi di coesistenza tra PATH e Ploeg in platoon eterogenei.

I parametri di Giordano sono stati invece selezionati dall'articolo [23], partendo da quelli più usati negli esperimenti, e sono:

- $\text{giordanoH}(h)$: rappresenta il coefficiente di frizione inter-veicolare, impostato a 0.71;
- $\text{giordanoK}(k)$: coefficiente elastico, scelto a 0.5;
- $\text{giordanoR}(r)$: coefficiente di frizione rispetto al veicolo di riferimento, impostato a $\sqrt{0.5}$;

Tuttavia i parametri potranno essere modificati durante le simulazioni per sfruttare tutte le potenzialità del controllore, o modificarne volontariamente alcuni comportamenti.

5.4 Risultati delle simulazioni

I risultati sono stati elaborati mediante l'utilizzo di Python per la creazione dei grafici, e i tool specifici per l'analisi dei file `.vec` prodotti dalle simulazioni.

Nelle sezioni seguenti verranno presentati i risultati per ognuno degli scenari precedentemente elencati, e per comodità verranno distinti i risultati di Giordano in un platoon PATH e in un platoon Ploeg. I parametri utilizzati per queste simulazioni sono quelli elencati nella sezione 5.3.

5.4.1 Risultati a velocità costante

Le prime simulazioni sono state realizzate con platoon comandati da un andamento costante della velocità del leader, impostata a 100 Km/h. Per PATH e Ploeg è semplicemente la velocità del leader ad essere impostata, mentre per Giordano è necessario stabilire il valore del parametro $\text{giordanoV}(v)$, dalla sezione 3.5, ovvero la velocità di riferimento. In questo caso è stata scelta una diretta associazione con la velocità del leader, in modo da mantenere una coerenza all'interno dei platoon.

Il fine è quello di comprendere come PATH e Ploeg reagiscano all'inserimento di un solo veicolo Giordano, se questo possa creare dei disturbi alla sicurezza e al comfort dei veicoli di un platoon oppure no.

PATH

Il grafico tempo-distanza 5.1 mostra una simulazione di 60 secondi di un platoon composto da 4 veicoli con sistemi di controllo $\{\text{PA GRD PA}\}$. I veicoli sono inseriti tutti alla distanza desiderata d_d di 5 m. Come si può notare Giordano, che è il veicolo v_2 , risulta avere delle variazioni in distanza millimetriche rispetto al veicolo di fronte che sembrano divergere (la distanza

viene misurata in maniera relativa, rispetto al veicolo frontale). Gli stessi andamenti si possono osservare con una simulazione a 16 veicoli nella figura 5.2, dove Giordano è il veicolo v_8 . Per semplicità sono stati presentati i risultati con Giordano come veicolo centrale del platoon, ma in tutte le altre posizioni esso si comporta sempre alla stessa maniera.

Per capire al meglio la natura dei movimenti millimetrici del veicolo Giordano è stata fatta una simulazione a 16 veicoli con una durata di 150 secondi. Il risultato, nella figura 5.3, mostra che Giordano, sempre in posizione v_8 non ha un comportamento divergente, che potrebbe provocare incidenti o malfunzionamenti, ma effettua delle oscillazioni millimetriche in distanza.

In ogni posizione possibile del veicolo Giordano nel platoon le simulazioni riportano sempre lo stesso comportamento, senza alcun incidente o divergenza.

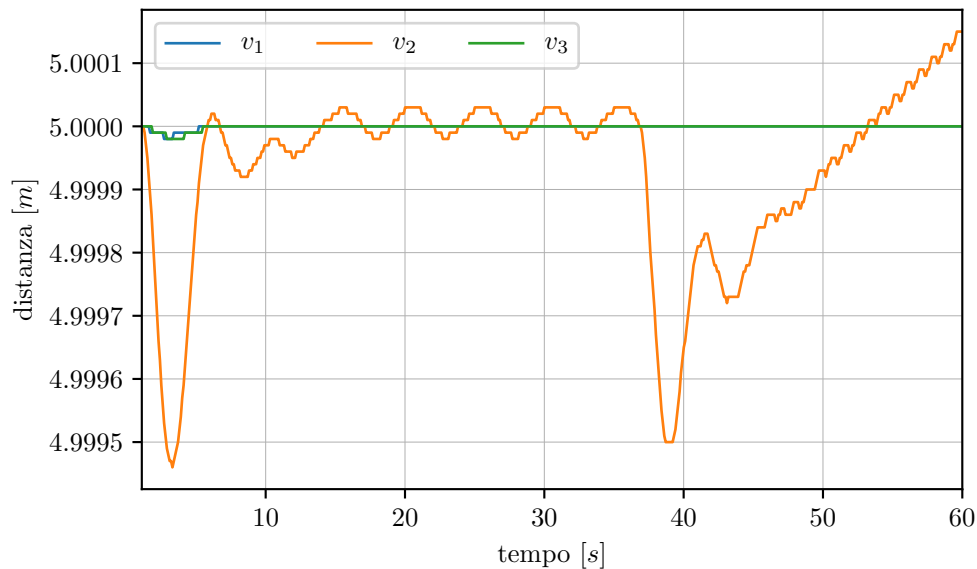


Figura 5.1: tempo-distanza di un platoon PATH a 4 veicoli con Giordano veicolo v_2 . Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

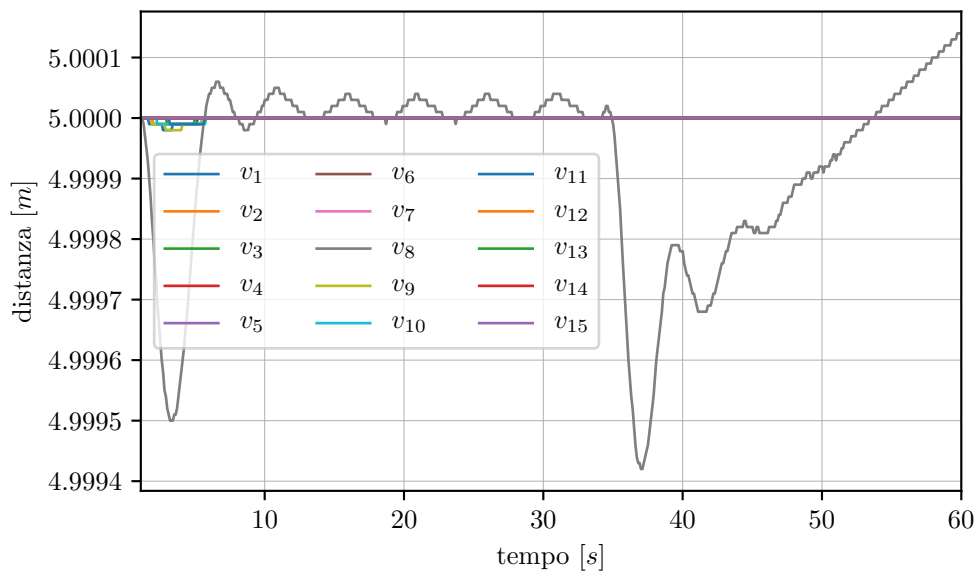


Figura 5.2: tempo-distanza di un platoon PATH a 16 veicoli con Giordano veicolo v_8 . Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

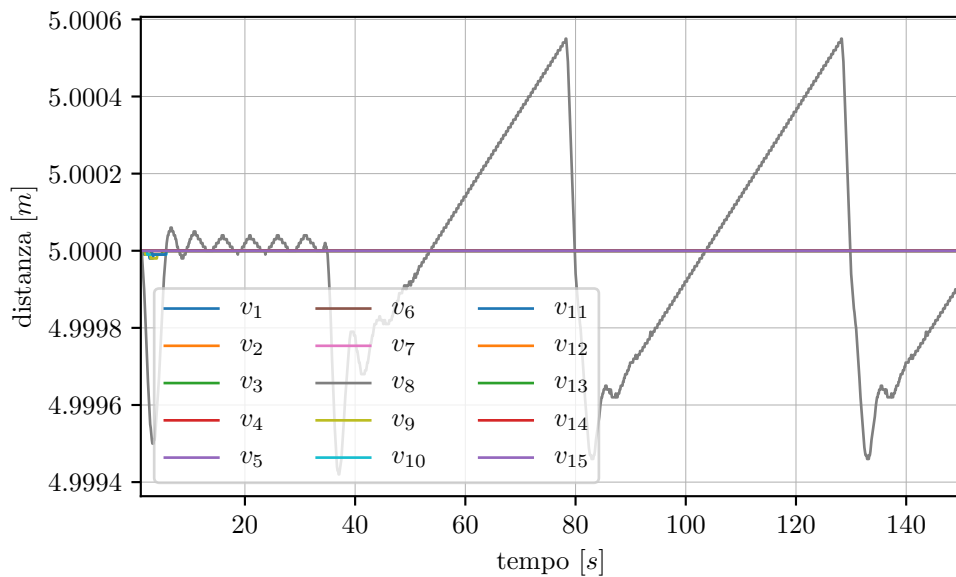


Figura 5.3: tempo-distanza di un platoon PATH a 16 veicoli con durata 150 s e con Giordano veicolo v_8 . Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

Ploeg

Per quanto riguarda la reazione di un platoon Ploeg all'inserimento di un veicolo Giordano sono state fatte delle simulazioni sempre di 60 s, nelle quali però le distanze di inserimento sono di 5 m per Giordano (valore di d_d) e di circa 16 m per Ploeg (valore calcolato in base alla velocità di 100 Km/h e all'Headway time di 0.5 s).

Dal grafico 5.4 si può notare come Giordano, in posizione v_2 , tenda a convergere verso la distanza desiderata di Ploeg, e nel farlo indietreggi avvicinandosi al veicolo precedente v_3 , del quale infatti si misura una distanza relativa al veicolo di fronte di qualche metro inferiore fino a convergenza del veicolo v_2 . Lo stesso comportamento si può notare nel grafico 5.5 nel quale è rappresentata una simulazione di un platoon a 16 veicoli con Giordano in posizione v_8 . Anche in questo caso v_8 tende alla distanza desiderata di Ploeg, e nel convergere porta con se un momentaneo avvicinamento di tutti i veicoli dietro, che gestiscono il disturbo esattamente come un sistema di controllo Ploeg farebbe.

Giordano risulta quindi fortemente influenzato dalla diversa e più ampia distanza dei veicoli con sistema di controllo Ploeg precedenti. Infatti dalla figura 5.6 si può notare come Giordano in ultima posizione, senza quindi più dietro nessuno, rimanga costantemente alla distanza d_d di 5 m.

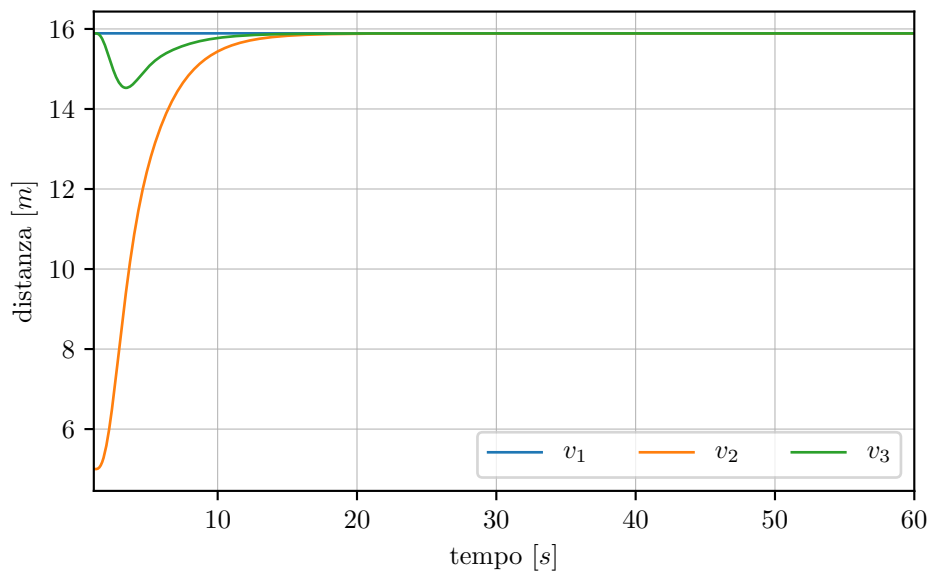


Figura 5.4: tempo-distanza di un platoon Ploeg a 4 veicoli con durata 60 s e con Giordano veicolo v_2 . Distanza target di Ploeg circa 16 metri e distanza target di Giordano 5 metri.

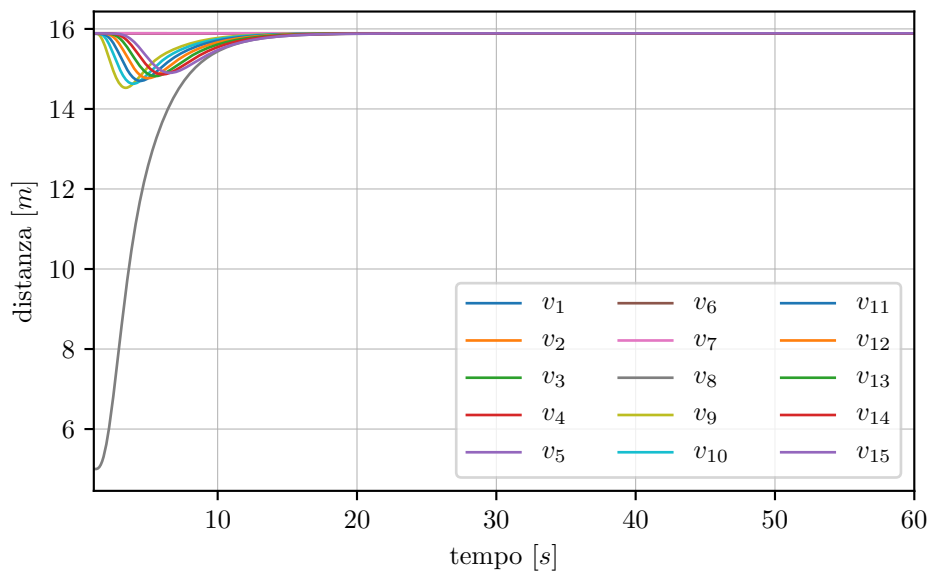


Figura 5.5: tempo-distanza di un platoon Ploeg a 16 veicoli con durata 60 s e con Giordano veicolo v_8 . Distanza target di Ploeg circa 16 metri e distanza target di Giordano 5 metri.

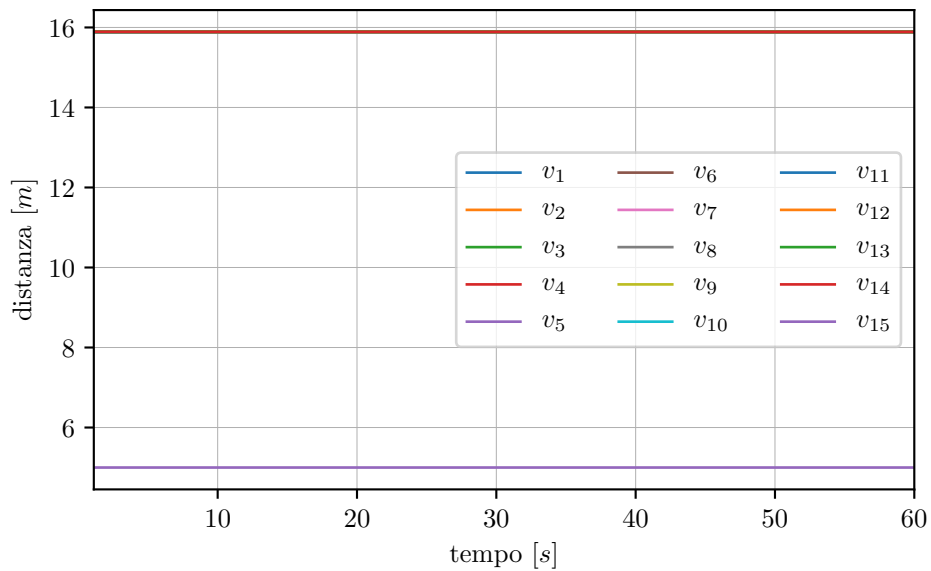


Figura 5.6: tempo-distanza di un platoon Ploeg a 16 veicoli con durata 60 s e con Giordano veicolo v_{15} . Distanza target di Ploeg circa 16 metri e distanza target di Giordano 5 metri.

Nonostante il disturbo creato da Giordano ai veicoli a lui precedenti, essendo i controlli selezionati string stable non risulta alcun incedente tra le simulazioni proposte.

5.4.2 Risultati in presenza di un disturbo sinusoidale

Per studiare la variazione al comfort di guida e la resistenza ad un ulteriore disturbo, sono state realizzate altre simulazioni con andamento della velocità del leader:

$$velocitaLeader + ampiezza * \sin(2 * simTime() * frequenza) \quad (5.1)$$

con la velocità del leader impostata a 100 km/h, l'ampiezza dell'intervallo di 10 km/h e la frequenza di oscillazione a 0.2 Hz. Anche in questo caso per PATH e Ploeg è bastato impostare la velocità del leader come scritto precedentemente, mentre per Giordano è stata nuovamente scelta come velocità di riferimento v l'andamento del leader del platoon.

Il fine delle simulazioni è quello di capire come un disturbo sinusoidale possa modificare il comfort di guida e la sicurezza già studiati e consolidati sia di PATH che di Ploeg, se uniti ad sistema di controllo Giordano. Un altro importante obiettivo è anche capire il comfort di guida del veicolo Giordano inserito nei platoon.

PATH

In un platoon di 16 veicoli PATH con Giordano in posizione v_8 , come mostrato dalla figura 5.7, le oscillazioni del veicolo v_8 risultano essere nettamente più ampie, seppur di soli 50 cm, rispetto agli altri veicoli, il che potrebbe implicare dei problemi di comfort per il veicolo Giordano. Inoltre nella simulazione si può notare come le vistose oscillazioni di v_8 portino anche a dei movimenti millimetrici tra alcuni dei veicoli PATH, che in platoon di soli suoi simili normalmente non si notano.

Dopo un'analisi dell'accelerazione della stessa simulazione, si può notare come nel platoon vengano seguiti due profili diversi, uno dai primi sette veicoli, e l'altro dai veicoli dopo l'ottavo, come mostrato nella figura 5.8. Isolando l'accelerazione di un platoon composto da soli PATH (figura 5.9) e quella del veicolo v_8 Giordano all'interno di un platoon PATH (figura 5.10), si può notare come i primi sette veicoli della figura 5.8 seguano proprio il profilo di velocità

imposto dal leader, mentre i successivi sette, da v_9 a v_{15} , seguano il profilo del veicolo v_8 , ovvero di Giordano.

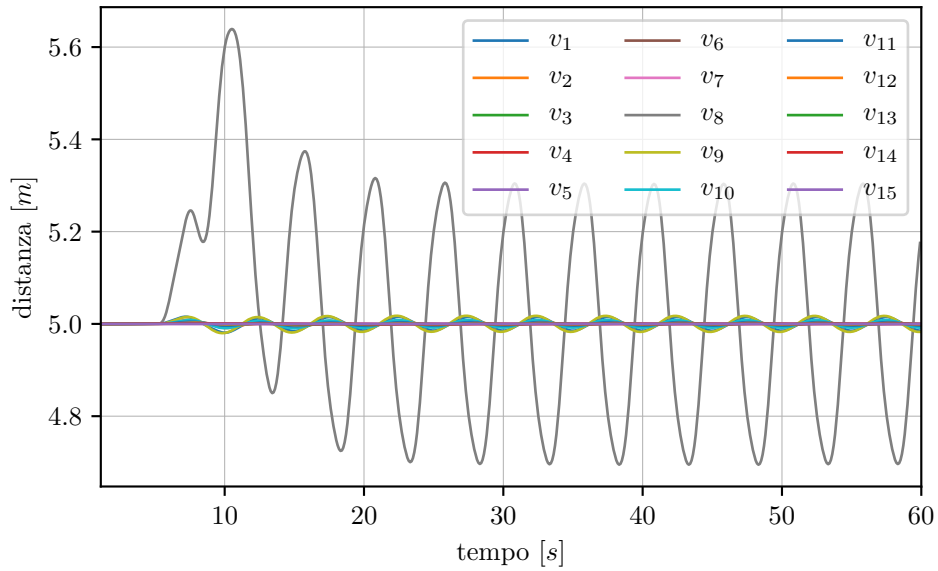


Figura 5.7: tempo-distanza di un platoon PATH a 16 veicoli con durata 60 s e con Giordano veicolo v_8 . Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

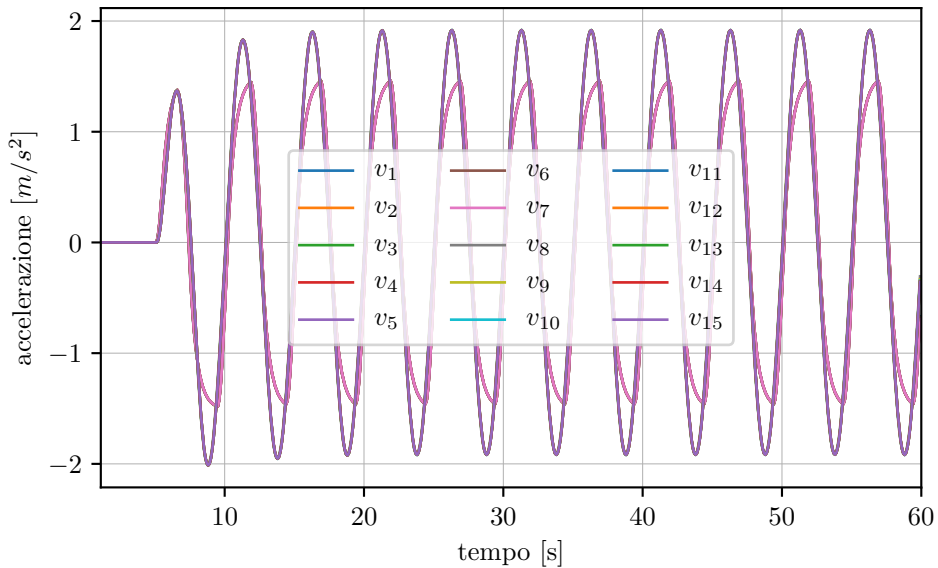


Figura 5.8: tempo-accelerazione di un platoon PATH a 16 veicoli con durata 60 s e con Giordano veicolo v_8 .

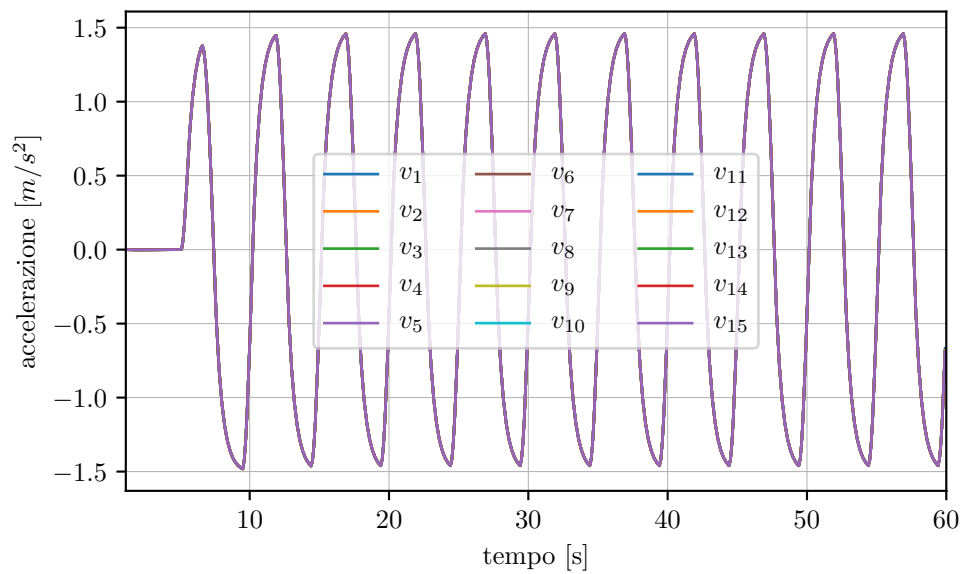


Figura 5.9: tempo-accelerazione di un platoon di soli PATH a 16 veicoli con durata 60 s.

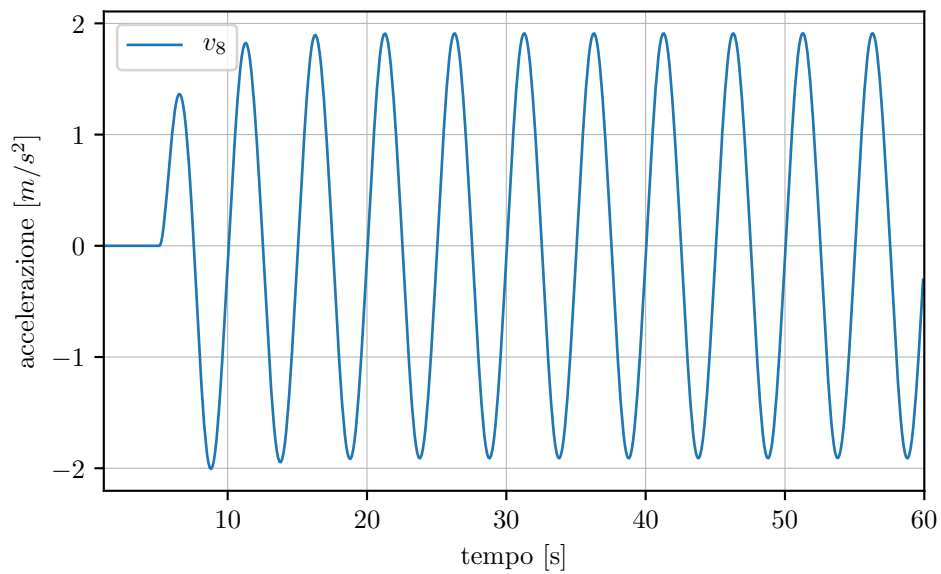


Figura 5.10: tempo-accelerazione di un veicolo Giordano in posizione v_8 in un platoon PATH a 16 veicoli con durata 60 s.

Ploeg

In un platoon a 16 veicoli Ploeg, con Giordano in posizione v_8 , come mostra la figura 5.11, le oscillazioni di Giordano sono più ampie, mentre Ploeg, nonostante venga spezzato in due dalla convergenza di Giordano, risulta procedere come normalmente fa in reazione ai disturbi.

Analizzando le accelerazioni Giordano risulta quello con le oscillazioni in accelerazione più ampie, ma comunque poco più del veicolo più disturbato tra i Ploeg. Le oscillazioni, tolto il transitorio, si mantengono comunque in un intervallo che va dai -2 m/s^2 ai 2 m/s^2 , che è consono ai canoni standard di comfort di guida.

Nonostante il doppio disturbo, non risultano esserci incidenti tra le simulazioni effettuate. Resta comunque da analizzare la natura delle oscillazioni più ampie, rispetto a PATH, nel primo caso presentato.

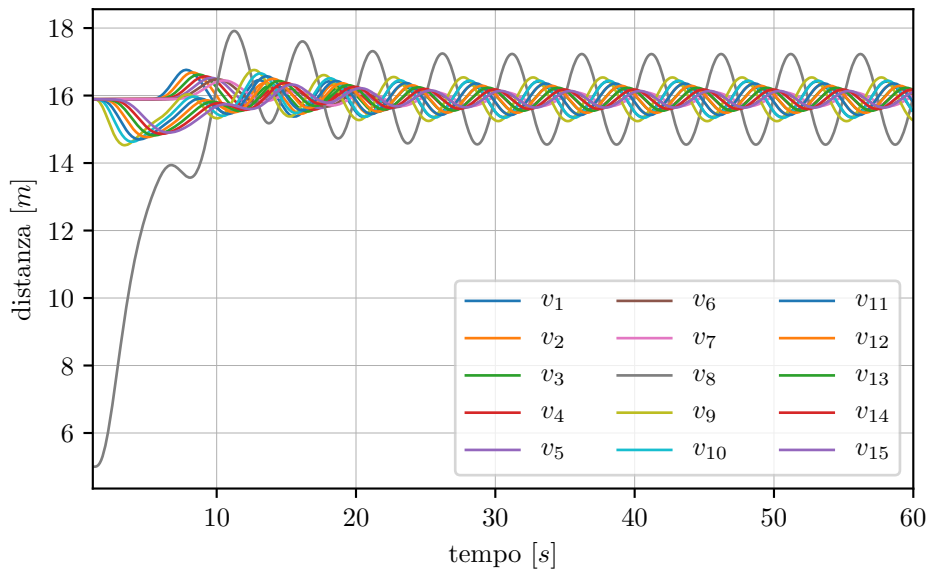


Figura 5.11: tempo-distanza di un platoon Ploeg a 16 veicoli con durata 60 s e con Giordano veicolo v_8 . Distanza target di Ploeg circa 16 metri e distanza target di Giordano 5 metri.

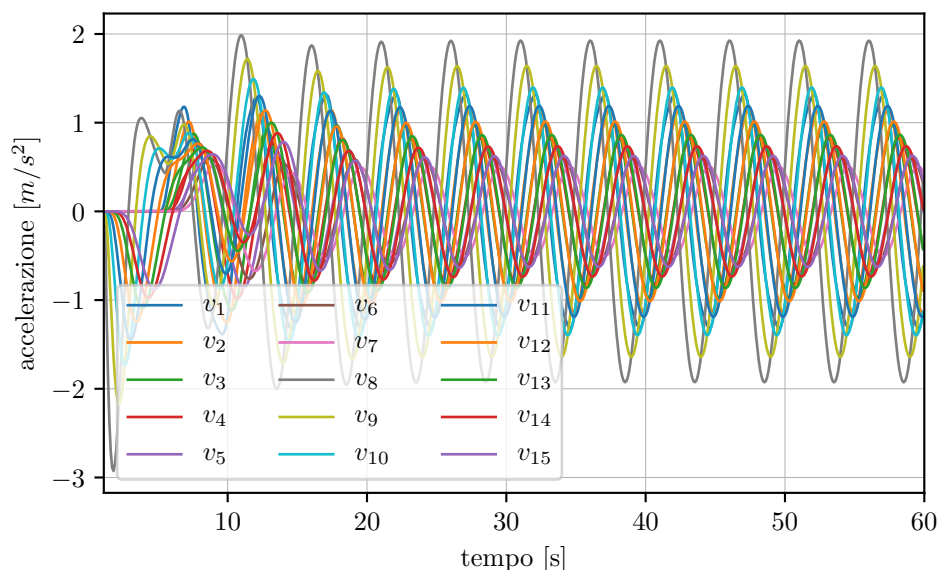


Figura 5.12: tempo-accelerazione di un platoon Pløeg a 16 veicoli con durata 60 s e con Giordano veicolo v_8 .

5.4.3 Risultati frenata di emergenza

Lo studio della frenata di emergenza è essenziale per capire la sicurezza di un sistema. Le simulazioni sono state eseguite con una decelerazione di 8 m/s^2 dopo 5 s di andamento costante a 100 km/h in un platoon di 16 veicoli.

La difficoltà principale nel presentare i risultati delle simulazioni è dovuta al diverso utilizzo dello scenario mostrato nella sezione apposita dell'articolo [23], infatti per le frenate di emergenza vengono presentati due casi possibili: il primo non adattando il parametro r di Giordano, il secondo adattandolo. L'eventuale adattamento di $giordanoR$ avviene nel seguente modo:

$$giordanoR = \text{minimo}(decelerazione/velocita', giordanoRMax); \quad (5.2)$$

dove la decelerazione è quella imposta al leader del platoon per la frenata di emergenza e dal valore di 8 m/s^2 , mentre $giordanoRMax$ è un valore massimo imposto al parametro r per evitare accelerazioni troppo aggressive, e in questo caso è impostato a 8. Essendo la velocità del veicolo tendente a zero, il parametro r tenderà, a partire da un valore piccolo, a convergere verso il valore di $giordanoRMax$.

Come mostrato nell'articolo in assenza di adattamento di r ci potrebbero essere degli incidenti già solo in un platoon composto interamente da sistemi di controllo Giordano. Infatti nella tabella 5.1, con r non adattato nel caso di PATH si hanno 15 incidenti su 15 simulazioni. Per Ploeg invece sono 7, e sono corrispondenti a Giordano nelle prime 7 posizioni del platoon. Adattando invece r gli incidenti risultano essere 0 in ogni caso, anche se con il rischio di avere distanze considerevoli tra un veicolo e l'altro, come mostrato nella figura 5.13.

Lista incidenti		
	Incidenti r adattato	Incidenti r non adattato
PATH	0	15
Ploeg	0	7

Tabella 5.1: Analisi incidenti in platoon composti da 16 veicoli di cui uno Giordano. Le simulazioni per ogni caso sono 15, una per possibile posizione (il leader è sempre ACC)

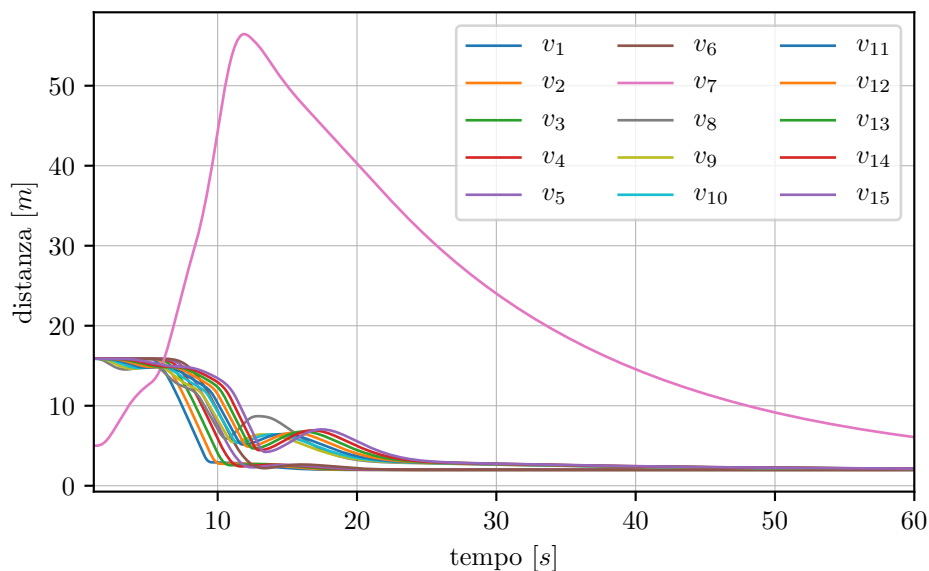


Figura 5.13: tempo-distanza di un platoon Ploeg a 16 veicoli con durata 60 s, con Giordano veicolo v_7 e con parametro giordanoR adattato.

5.5 Considerazioni

La sicurezza è mantenuta in ogni caso, fatta eccezione per i risultati riguardanti le frenate di emergenza (sezione 5.4.3) con il parametro r di Giordano non adattato. I risultati quindi suggeriscono l'utilizzo dell'adattamento di r per evitare incidenti nel caso di decelerazioni brusche. Potrebbe essere un problema non banale la gestione del parametro r di più veicoli Giordano da parte dell'infrastruttura stradale o dai veicoli leader del platoon.

Il comfort di guida invece presenta le maggiori problematiche. Infatti analizzando i risultati a velocità costante (sezione 5.4.1) si può notare come Giordano interagisca con PATH per mezzo di oscillazioni, che seppur piccole potrebbero lo stesso minare il comfort di guida del veicolo Giordano con continui cambi di velocità più o meno aggressivi. Questo disturbo nell'adattamento è probabilmente dovuto alle continue piccole accelerazioni che i veicoli PATH hanno per mantenere forzatamente la distanza di 5 m, indipendentemente dalla velocità.

Certamente il problema più grande, per il comfort di guida, riguarda invece la reazione a un disturbo di tipo sinusoidale. Infatti Giordano mescolato con PATH risulta avere oscillazioni più ampie, seppur consone ai canoni di sicurezza, e con accelerazioni leggermente più grandi. Come accadeva anche nel mix di PATH e Ploeg [15], tutti i veicoli non PATH, partendo dal fondo, diventano il leader di riferimento dei veicoli PATH alle loro spalle. Questo è un esempio evidente di come la topologia di comunicazione influenza fortemente la guida di un platoon. Infatti come si nota nella figura 5.8 i veicoli PATH da v_9 a v_{15} non seguono più l'andamento sinusoidale del leader v_0 , ma quello del veicolo Giordano v_8 (il primo non PATH che incontrano).

Con Ploeg invece c'è un problema nella massimizzazione nell'utilizzo delle strade. Giordano è infatti modellato per avere una distanza desiderata piccola, compatibilmente con gli obiettivi di platooning. Nell'esempio 5.5 è invece evidente come Giordano si trovi a dover lavorare a una d_d molto più grande della sua. Questo è ben spiegato dall'analogia con il sistema molla-ammortizzatore, infatti è come se il Ploeg alle sue spalle fosse una molla molto più allungata di quella frontale, e, come normalmente una molla fa, Giordano cerchi di raggiungere la stabilità nel mezzo tra i due Ploeg.

5.5.1 Possibili modifiche

Nell'analisi del comfort di guida e delle oscillazioni non è stato tenuto conto del complesso funzionamento dei parametri di Giordano. Di default sono infatti stati scelti dei parametri già usati nell'articolo di presentazione [23]. È quindi possibile effettuare un'analisi parametrica e sperimentale per cercare di ridurre al più possibile le oscillazioni di Giordano, o quantomeno la differenza di accelerazione che c'è con PATH.

Un'altra possibile modifica, che non verrà analizzata in questa tesi, potrebbe essere l'eliminazione dalle formule di Giordano dell'influenza della componente posteriore. Questo potrebbe permettere a Giordano di rimanere a 5 m di distanza anche se tra due veicoli Ploeg. È anche vero che così facendo staremmo analizzando un controllore completamente diverso, e senza più topologia bidirezionale.

5.6 Analisi parametrica

Dalle formule presentate nella sezione 3.5 risulta che l'influenza dei parametri r , h e k sul sistema può essere studiata separatamente. Infatti come spiegato nell'articolo [23] è possibile suddividere l'analisi delle formule in due componenti, la prima studia i comportamenti medi di un platoon Giordano, la seconda invece riguarda il vettore delle differenze $z(t)$ dove ogni componente z_i è uguale a $y_{i-1} - y_i$.

Dallo studio della posizione media del platoon se ne ricava che

$$a_{max} = rve^{-rt} \quad (5.3)$$

cioè l'accelerazione massima del controllore Giordano è influenzata solamente dal parametro r e non dagli altri due. Di conseguenza ne risulta che $\tau_a = \frac{1}{r}$, cioè la costante di tempo del sistema risulta essere tanto più piccola quanto più r è grande. Questo significa che all'aumentare di r il sistema converge più velocemente alla velocità di riferimento, ma questo provoca un aumento di a_{max} con conseguente peggioramento del comfort di guida.

Il secondo livello di analisi dell'influenza dei parametri riguarda il vettore delle differenze $z(t)$. Dopo qualche manipolazione matematica se ne ricava che la funzione di trasferimento del sistema Giordano è la seguente:

$$\Gamma(s) = \frac{1}{s^2 + (h\Omega_i^2 + r)s + k\Omega_i^2} \quad (5.4)$$

Quello che viene messo in risalto è che i poli della funzione di trasferimento sono influenzati da tutti e tre i parametri. In particolar modo ne segue che la condizione di stabilità del sistema è $h > \frac{k}{r}$.

Dall'analisi se ne deduce che un'idea potrebbe essere quella di ridurre il parametro r per aumentare il comfort di guida e forse ridurre le oscillazioni. Così facendo però il rapporto $\frac{k}{r}$ aumenterebbe mettendo a rischio la stabilità, per questo bisogna considerare anche come si sposterebbero eventualmente i poli di $\Gamma(s)$ oppure il significato dei parametri k e h . Infatti k è presentato come coefficiente elastico, mentre h come coefficiente di frizione tra due veicoli.

5.7 Risultati modifica parametri

Partendo dall'idea di ridurre il parametro r per aumentare il comfort, è quindi necessario aumentare h o diminuire k .

Come primo studio è stato cercato di capire quali possono essere gli effetti collaterali di una k troppo piccola. Come risulta dal grafico 5.14 dove è stata usata una k di 0.1, paragonato alle oscillazioni studiate nella sezione 5.4.2, l'ampiezza rimane invariata, mentre sembra che al diminuire di k rallenti la convergenza alla posizione base di 5 m.

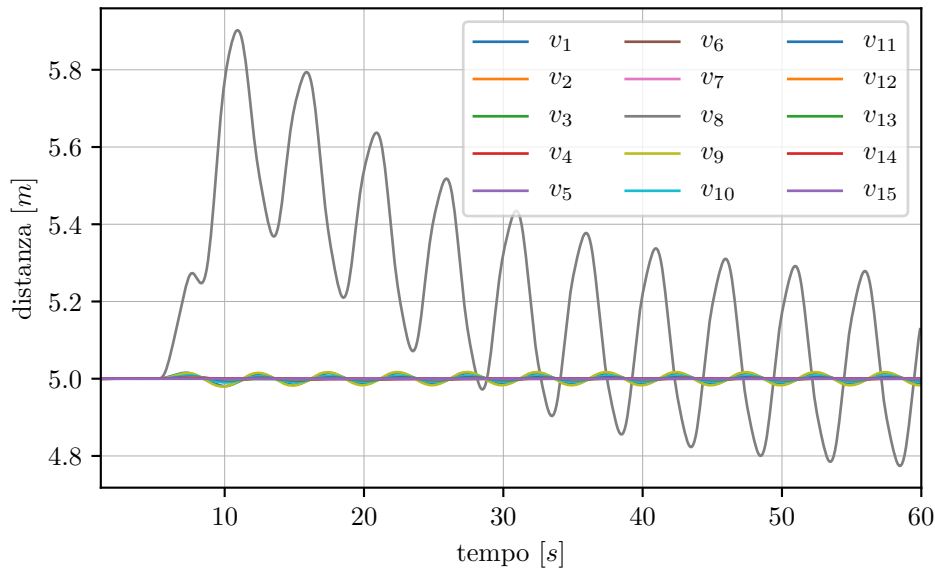


Figura 5.14: tempo-distanza di un platoon PATH a 16 veicoli con durata 60 s, con Giordano veicolo v_8 e con $k = 0.1$. Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

È quindi necessario avere un valore di k non troppo piccolo, per evitare problemi di convergenza. Scegliendo dei parametri $k = 0.2$ e $r = 0.5$, è stata trovata la condizione ottimale, che sperimentalmente non viene mai superata in nessun'altra situazione.

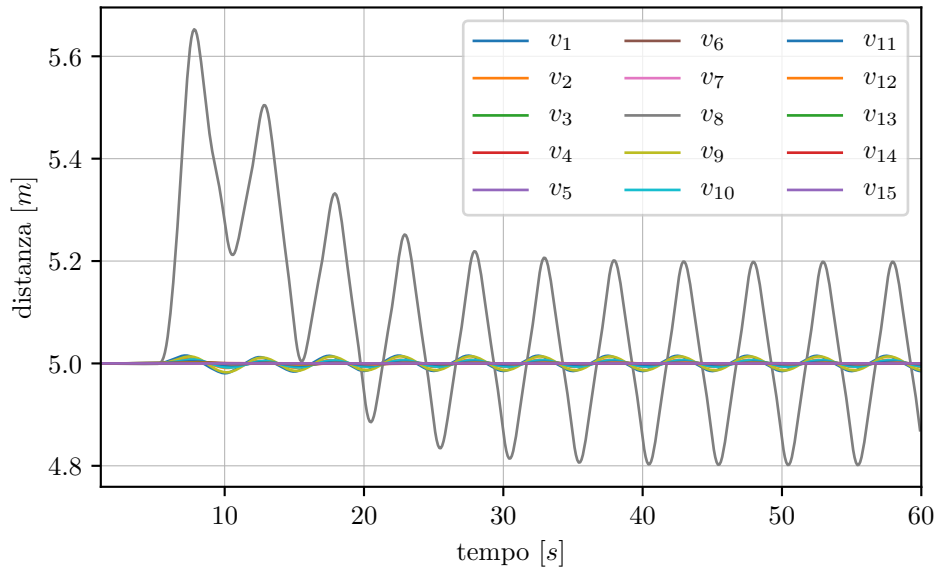


Figura 5.15: tempo-distanza di un platoon PATH a 16 veicoli con durata 60 s, con Giordano veicolo v_8 , con $k = 0.2$ e $r = 0.5$. Distanza target di PATH 5 metri e distanza target di Giordano 5 metri.

Le oscillazioni ottenute sono quindi di circa 40 cm a confronto di quelle ottenute nella sezione 5.4.2 di circa 60 cm. L'ampiezza non è ulteriormente riducibile, probabilmente muovendo ulteriormente i parametri c'è un grosso rischio di influenzare la dinamica del sistema. I parametri scelti nell'articolo [23] erano, come ci si poteva aspettare, molto vicini alla soluzione ottimale in un'idea di trade-off tra comfort e convergenza. Inoltre le oscillazioni non sono riducibili anche perché Giordano rispetto a PATH ha un'azione di ritardo nei confronti del pattern di velocità imposto dal leader. Come risulta nella figura 5.16 tutti i veicoli da v_1 a v_7 , che sono PATH, rispettano l'andamento imposto dal leader, mentre da v_8 Giordano in poi l'andamento cambia in quanto Giordano reagisce in ritardo e non simultaneamente alla velocità di riferimento, e i PATH alle sue spalle, che lo impostano come nuovo leader, ne seguono l'andamento.

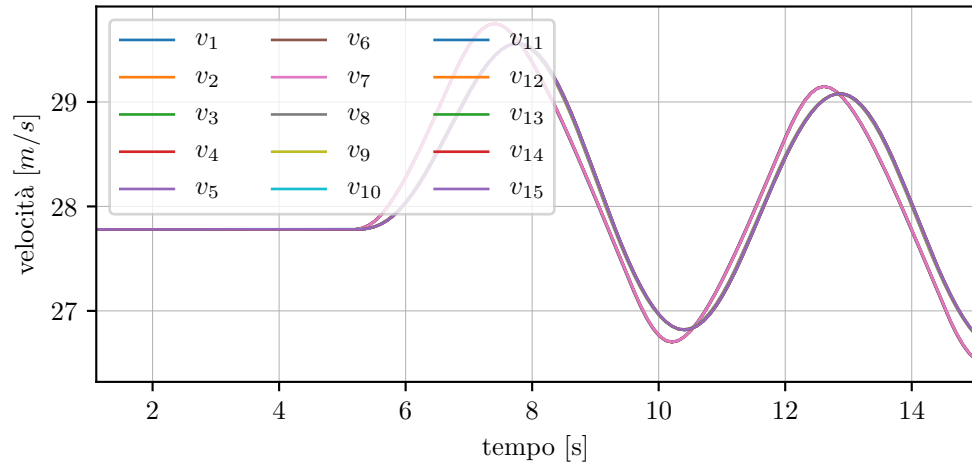


Figura 5.16: tempo-velocità di un platoon PATH a 16 veicoli con durata 60 s, con Giordano veicolo v_8 , con $k = 0.2$ e $r = 0.5$.

Capitolo 6

Conclusioni

In questa tesi è stata analizzata l'integrazione in un platoon di un sistema di controllo chiamato Giordano con i più tradizionali controllori noti come PATH e Ploeg. Giordano segue una logica di progetto alquanto diversa da PATH e Ploeg, quindi l'analisi della sua possibile integrazione è particolarmente interessante.

L'integrazione di un singolo veicolo Giordano comporta, come prevedibile, variazioni delle prestazioni complessive che devono essere considerate con cura. In un platoon di soli PATH esso tende ad avere delle oscillazioni microscopiche, centrate intorno ai 5 m, anche a velocità costante del leader. Questo è sintomo di una non perfetta integrazione e di una continua ricerca della stabilità da parte del veicolo Giordano. La spiegazione del comportamento risiede principalmente nella diversa topologia di controllo utilizzata dai due sistemi. Infatti l'accelerazione di Giordano è influenzata non solo dall'andamento del veicolo frontale, ma anche di quello precedente. Essendo PATH un controllore che cerca, indipendentemente dalla velocità, di mantenere la d_d di 5 m, le sue continue piccole accelerazioni per raggiungere l'obiettivo risultano influenti per il veicolo Giordano.

Per l'integrazione con Ploeg il problema principale riguarda le due diverse distanze desiderate, con cui lavorano i sistemi di controllo. Il problema in realtà è relativo, perché dopo una fase di transitorio, nella quale non risultano esserci incidenti, il platoon simulato procede senza particolari problemi. Lo svantaggio più grande è il fatto che si è costretti ad usare la d_d di Ploeg, che è più del triplo di quella stabilita per Giordano. In questo caso si perde quindi tutto il potenziale di un sistema di controllo che lavora con distanze ridotte per massimizzare l'utilizzo delle strade. Una possibile modifica sarebbe quella di eliminare l'influenza del veicolo posteriore per il sistema di controllo Giordano,

ma bisognerebbe effettuare prima un'analisi delle formule dal punto vista del controllo automatico del nuovo sistema.

La sicurezza viene sempre mantenuta. Questo potrebbe sembrare un risultato ovvio dato che tutti i sistemi sono string stable; tuttavia non c'è nulla che garantisca, da un punto di vista teorico, che veicoli che seguono leggi di controllo diverse tra loro mantengano questa proprietà quando vengono mescolati, quindi il risultato è di grande interesse e importanza. Nello scenario della frenata di emergenza va però considerato necessariamente l'adattamento del parametro r di Giordano, che potrebbe risultare difficile dal punto di vista della gestione da parte dell'infrastruttura, ma che garantisce la sicurezza totale anche nel caso di una brusca decelerazione.

Il comfort di guida è invece l'aspetto più problematico, dopo uno studio parametrico la conclusione è che le oscillazioni in distanza di Giordano risultano ottimali in ampiezza e accelerazione con i seguenti parametri:

- $\text{giordanoH}(h) = 0.71$;
- $\text{giordanoK}(k) = 0.2$;
- $\text{giordanoR}(r) = 0.5$;

L'attenuazione del disturbo sinusoidale risulta comunque peggiore rispetto a PATH ed equiparabile a Ploeg.

Per la massimizzazione nell'utilizzo delle strade invece in un platoon PATH si mantiene la distanza desiderata di Giordano, con Ploeg invece necessita di un transitorio in cui avviene l'adattamento alla distanza inter-veicolare utilizzata da Ploeg.

In conclusione Giordano può certamente coesistere con PATH e Ploeg, perdendo qualche vantaggio in termini di distanze inter-veicolari se integrato in un platoon Ploeg, o a livello di oscillazioni in distanza se integrato in un platoon PATH. Le prestazioni, seppur leggermente peggiori se paragonate a un *platoon omogeneo*, risultano comunque rientrare negli obiettivi del *platooning*. Il risultato è di grande interesse industriale, in quanto ogni casa automobilistica potrebbe produrre e commercializzare il proprio sistema di controllo per guida cooperativa, senza doversi adattare ad uno standard e accordarsi con altri produttori.

Lo studio dei *platoon eterogenei* è appena iniziato e rappresenta un'area di ricerca e applicazione del tutto nuova con significativi risvolti sia teorici che tecnici ed economici. Ad esempio non è ancora mai stato fatto alcuno studio di

topologie e sistemi di controllo in platoon formati da più di due CACC diversi, né è, a mia conoscenza, mai stato formalizzato il problema da un punto di vista teorico.

Rimanendo nell'ambito degli studi euristici per via simulativa ci sono in ogni caso moltissimi scenari di interesse e applicazioni da esplorare, come ad esempio se si possano ottenere prestazioni complessivamente migliori in un platoon composto da veicoli diversi tra loro assegnando a ciascun veicolo il controllore per lui ottimale, anziché lo stesso controllore a tutti i veicoli.

Bibliografia

- [1] *Vehicles in use Europe 2022*, <https://www.acea.auto/publication/report-vehicles-in-use-europe-2022>.
- [2] *Car Crash Deaths and Rates*, <https://www.injuryfacts.nsc.org/motor-vehicle/historical-fatality-trends/deaths-and-rates>.
- [3] *How Many People Die From Car Accidents Each Year?* <https://www.forbes.com/advisor/legal/auto-accident/car-accident-deaths>.
- [4] *Le emissioni del trasporto stradale in Italia*, https://www.isprambiente.gov.it/files2021/eventi/evento16apr2021_emissioni_strada.pdf.
- [5] L. Elefteriadou, *An Introduction to Traffic Flow Theory*. Springer New York, 2014.
- [6] H. Greenberg, «An Analysis of Traffic Flow,» in *Operations Research*, vol. 7, feb. 1959, pp. 79–85.
- [7] V. Vohra, M. Wahba, G. Akarslan, R. Ni e S. Brennan, «An Examination of Vehicle Spacing to Reduce Aerodynamic Drag in Truck Platoons,» in *2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2018, pp. 1–6.
- [8] R. Gnatowska e M. Sosnowski, «The influence of distance between vehicles in platoon on aerodynamic parameters,» *Experimental Fluid Mechanics 2017*, vol. 180, pp. 194–198, gen. 2017.
- [9] *Safe Road Trains for the Environment*, <https://www.roadtraffic-technology.com/projects/the-sartre-project/>.
- [10] S. Tsugawa, «An Overview on an Automated Truck Platoon within the Energy ITS Project,» *IFAC Proceedings Volumes*, vol. 46, n. 21, pp. 41–46, 2013, 7th IFAC Symposium on Advances in Automotive Control. indirizzo: <https://www.sciencedirect.com/science/article/pii/S1474667016383409>.
- [11] G. Sidorenko, J. Thunberg, K. Sjöberg e A. Vinel, «Vehicle-to-Vehicle Communication for Safe and Fuel-Efficient Platooning,» in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 795–802.
- [12] C. Bergenheim, E. Hedin e D. Skarin, «Vehicle-to-Vehicle Communication for a Platooning System,» *Procedia - Social and Behavioral Sciences*, vol. 48, pp. 1222–1233, 2012, Transport Research Arena 2012. indirizzo: <https://www.sciencedirect.com/science/article/pii/S1877042812028340>.

- [13] R. Kianfar, B. Augusto, A. Ebadighajari et al., «Design and Experimental Validation of a Cooperative Driving System in the Grand Cooperative Driving Challenge,» *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, n. 3, pp. 994–1007, 2012.
- [14] M. di Bernardo, A. Salvi e S. Santini, «Distributed Consensus Strategy for Platooning of Vehicles in the Presence of Time-Varying Heterogeneous Communication Delays,» *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, n. 1, pp. 102–112, 2015.
- [15] M. Segata, L. Ghio e R. Lo Cigno, «On the Progressive Introduction of Heterogeneous CACC Capabilities,» in *IEEE Vehicular Networking Conference (VNC)*, Ulm, Germany, nov. 2021, pp. 1–8.
- [16] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe e M. Nakamura, «Cooperative Adaptive Cruise Control in Real Traffic Situations,» *TITS*, vol. 15, n. 1, pp. 296–305, feb. 2014.
- [17] Y.-Y. Qin, Z.-Y. He e B. Ran, «Rear-End Crash Risk of CACC-Manual Driven Mixed Flow Considering the Degeneration of CACC Systems,» *IEEE Access*, vol. 7, pp. 140 421–140 429, set. 2019.
- [18] M. di Bernardo, A. Salvi e S. Santini, «Distributed Consensus Strategy for Platooning of Vehicles in the Presence of Time Varying Heterogeneous Communication Delays,» *IEEE Trans. on Intelligent Transportation Systems*, vol. 16, n. 1, pp. 102–112, set. 2014.
- [19] S. Santini, A. Salvi, A. S. Valente, A. Pescapè, M. Segata e R. Lo Cigno, «A Consensus-Based Approach for Platooning with Intervehicular Communications and Its Validation in Realistic Scenarios,» *IEEE Trans. on Vehicular Technology*, vol. 66, pp. 1985–1999, 3 Mar. 2017.
- [20] —, «Platooning Maneuvers in Vehicular Networks: a Distributed and Consensus-Based Approach,» *IEEE Transactions on Intelligent Vehicles*, vol. 4, pp. 59–72, 1 mar. 2019.
- [21] J. Ploeg, B. Scheepers, E. van Nunen, N. van de Wouw e H. Nijmeijer, «Design and Experimental Evaluation of Cooperative Adaptive Cruise Control,» in *IEEE ITSC 2011*, Washington, D.C.: IEEE, ott. 2011, pp. 260–265.
- [22] R. Rajamani, H.-S. Tan, B. K. Law e W.-B. Zhang, «Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons,» *IEEE Transactions on Control Systems Technology*, vol. 8, n. 4, pp. 695–708, 2000.
- [23] G. Giordano, M. Segata, F. Blanchini e R. Lo Cigno, «The joint network/control design of platooning algorithms can enforce guaranteed safety constraints,» *Elsevier Ad Hoc Networks*, vol. 94, nov. 2019.
- [24] L. Peppard, «String stability of relative-motion PID vehicle control systems,» *IEEE Transactions on Automatic Control*, vol. 19, n. 5, pp. 579–581, 1974.
- [25] R. Rajamani, *Vehicle Dynamics and Control*, 2^a ed. Springer, 2012.
- [26] A. Varga, «Using the OMNeT++ discrete event simulation system in education,» *IEEE Transactions on Education*, vol. 42, n. 4, 11 pp.-, 1999.

- [27] P. A. Lopez, M. Behrisch, L. Bieker-Walz et al., «Microscopic Traffic Simulation using SUMO,» *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 2575–2582, nov. 2018. indirizzo: <https://elib.dlr.de/127994/>.
- [28] C. Sommer, R. German e F. Dressler, «Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis,» *IEEE Transactions on Mobile Computing (TMC)*, vol. 10, n. 1, gen. 2011.
- [29] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler e R. Lo Cigno, «PLEXE: A Platooning Extension for Veins,» in *IEEE Vehicular Networking Conference (VNC)*, Paderborn, Germany, dic. 2014, pp. 53–60.
- [30] M. Segata, R. Lo Cigno, T. Hardes et al., «Multi-Technology Cooperative Driving: An Analysis Based on PLEXE,» *IEEE Transactions on Mobile Computing (TMC)*, feb. 2022.