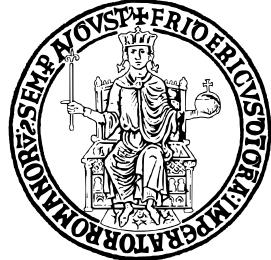


UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II



SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE
DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN INFORMATICA

JAMACT

UN MODELLO DI INDUZIONE DINAMICA
DEL PHANTOM TRAFFIC JAM PER
AGEVOLARE IL TRANSITO DEI VEICOLI DI
TRASPOSTO PUBBLICO

Relatore

Prof. Walter BALZANO

Candidato

Biagio SCOTTO DI COVELLA

N86/3605

Anno Accademico

2022–2023

A mia nipote

Indice

1	Introduzione	7
2	Aspetti preliminari	11
2.1	Reti Veicolari ad Hoc: Vanet	12
2.2	Smart Cities	14
2.3	Veicoli Connessi e Veicoli Autonomi	14
2.4	Prolog	15
2.5	Python	16
3	Il Phantom Traffic Jam	19
3.1	jamiton	20
3.2	Letteratura scientifica sul Phantom Traffic Jam	20
3.3	Il traffico come un fluido: il modello Lighthill-Whitham-Richards	23
3.4	Approcci alla riduzione dell'impatto del traffico fantasma	25
4	Il modello Jamact	27
4.1	Le basi del modello	28
4.2	Modello Teorico	32
4.3	Criticità del sistema e possibili soluzioni	36
4.3.1	Approcci alla coordinazione tra veicolo prioritario e veicoli non prioritari	37
4.3.2	Approcci alla coordinazione di più veicoli prioritari	38
4.3.3	Approcci alla gestione di interferenze tra più jamiton	40
4.4	Esempio Teorico	43
4.5	Jamact e Safety Car: un isomorfismo nella gestione veicolare	47
4.6	Jamact e l'ottimizzazione delle risorse	48
5	Simulatore per il modello Jamact	51
5.1	Panoramica sul codice	51
5.2	Esempio di applicazione dinamica	57
5.3	Risultati simulazioni Jamact	60
6	Modello Jamact per Autobus	67
6.1	Una panoramica sull'utilizzo degli autobus	68
6.2	Estensione modello Jamact – Autobus	71

6.3	Il problema del Commesso Viaggiatore	73
6.4	Isomorfismo con il Trasporto Ferroviario e le Coincidenze dei Treni	76
6.5	Esempio di applicazione dinamica del modello esteso	76
6.6	Risultati simulazioni modello Jamact per autobus	79
7	Conclusioni e sviluppi futuri	89
Appendice A		91
A.1	Moto Uniformemente Accelerato: Concetto e Applicazioni	91
A.2	Equazioni del moto uniformemente accelerato	91
A.3	Moto Uniformemente Accelerato nel Calcolo della Decelerazione	93
Appendice B		95
B.1	Grafi e Tipologie di Grafi Utilizzati	95
B.2	Libreria Python OSMnx	96
Bibliografia		97
Ringraziamenti		101

Elenco delle figure

2.1	Esempio di rete vanet, [10]	12
2.2	Esempio di Smart Cities, [17]	14
2.3	Classificazione livelli di automazione di un veicolo, [18]	15
3.1	MIT: Traffico fantasma strada circolare	21
3.2	The Wide Moving Jam	22
3.3	The Butterfly Effect	22
3.4	The Bottleneck	23
4.1	Elementi fondanti del sistema Jamact	28
4.2	Classificazione strade	31
4.3	Diagramma di flusso – Spiegazione funzionamento Jamact	32
4.4	Calcolo punti statici	35
4.5	Casi particolari	36
4.6	Esempio di una struttura dati Coda.	38
4.7	Esempio di una Coda di priorità con valori interi.	39
4.8	Diagramma di flusso – Gestione occupazione incrocio	40
4.9	Static Provisioning	44
4.10	Dynamic Adaption	45
4.11	Attivazione Jamact	45
4.12	Avanzamento veicoli	46
4.13	Disattivazione Jamact	46
4.14	Esempio percorso 1	47
4.15	Esempio percorso 2	47
4.16	Safety Car Formula 1	48
5.1	Mappa stradale Fuorigrotta	52
5.2	Mappa stradale del percorso	54
5.3	Simulazione con Jamact Attivo	58
5.4	Log della simulazione.	59
5.5	Grafici impatto Jamact bassa densità	62
5.6	Grafici impatto Jamact media densità	62
5.7	Grafici impatto Jamact alta densità	63
5.8	Grafici impatto Jamact	64
5.9	Grafici performance Jamact	64

6.1	Utilizzo mezzi di trasporto	68
6.2	Soddisfazione utenti per Autobus	69
6.3	Grado di soddisfazione per i requisiti chiave	69
6.4	Cosa chiedono gli utenti	70
6.5	Orari Autobus linea 102	79
6.6	Percorso Autobus linea 102	80
6.7	Grafici impatto Jamact bassa densità	83
6.8	Grafici impatto Jamact media densità	83
6.9	Grafici impatto Jamact alta densità	84
6.10	Numero medio di veicoli interferenti	85
6.11	Grafico performance – densità	85
6.12	Grafici impatto Jamact per Autobus	86
B.1	Esempio di applicazione della libreria osmnx.	96

Elenco delle tabelle

4.1	Spiegazione delle variabili del modello	30
5.1	Risultati Simulazioni Jamact	61
5.2	Impatto Jamact per densità di traffico	63
6.1	Risultati Simulazioni Jamact per Autobus	82
6.2	Impatto Jamact per Autobus per densità di traffico	86

–1–

Introduzione

La mobilità sostenibile rappresenta una delle sfide più importanti del nostro tempo, in un contesto caratterizzato da una crescente urbanizzazione, da una domanda sempre maggiore di spostamenti e dall'urgente necessità di ridurre l'impatto ambientale dei trasporti. Il traffico stradale, in particolare, è responsabile di una quota significativa delle emissioni di gas serra, dell'inquinamento acustico e dell'occupazione di spazio urbano; in questo scenario, emerge la necessità di implementare modelli innovativi di gestione del traffico stradale che possano contribuire a rendere le città più vivibili, riducendo i tempi di viaggio, i livelli di inquinamento e i costi associati alla mobilità.

Nel contesto italiano attuale il fenomeno del traffico stradale è evidenziato anche da uno studio condotto dall'Eurostat nel 2019, che colloca l'Italia al secondo posto in Europa per numero di auto per abitante, con una media di 663 auto ogni 1.000 abitanti [1]. Questo dato evidenzia non solo il predominio dell'automobile come mezzo di trasporto più diffuso e utilizzato nel nostro paese, ma anche come il traffico stradale sia principalmente costituito da auto. Queste, oltre a inquinare più dei mezzi collettivi quali autobus, treni e tram, occupano anche più spazio sulle strade, trasportando un numero inferiore di persone.

Per esempio, come dimostrato da un rapporto della National Association of City Transportation Officials (NACTO) [2], gli autobus pubblici svolgono un ruolo fondamentale nel trasporto urbano, trasportando fino a 8000 persone in un'ora, rispetto alle 600-1200 persone trasportate dalle auto private nello stesso periodo. Questo dato evidenzia come gli autobus, occupando meno spazio stradale e trasportando un numero maggiore di persone, rappresentino una fondamentale soluzione per contrastare la congestione stradale, ma anche per diminuire l'apporto del traffico urbano sull'inquinamento ambientale.

Inoltre l'Italia si confronta con una realtà in cui il tempo perso nel traffico è del 24% in più del necessario, come riportato dall'articolo di AGI.it [3]. Questo dato è particolarmente significativo considerando che oltre il 50% degli spostamenti nel paese avviene ancora attraverso l'uso dell'auto privata, secondo quanto riportato da uno studio di Kyoto Club [4]. Ciò non solo conferma l'esigenza di diminuire il numero di auto presenti sulle strade, ma sottolinea anche come è sempre più necessario incentivare l'uso dei mezzi pubblici a discapito dell'auto privata.

Tra le principali cause dell'elevato uso dell'auto privata per muoversi, e quindi

dello scarso utilizzo di mezzi di trasporto pubblico, sicuramente c'è l'inefficienza del trasporto pubblico su gomma. Secondo un articolo del quotidiano La Repubblica [5], l'Italia ha un problema di ricambio dei mezzi pubblici, con un'età media degli autobus di 12,2 anni, il doppio rispetto alla Germania, e solo un terzo degli autobus classificati Euro 3, contribuendo significativamente all'inquinamento ambientale; questo dato sottolinea come il settore degli autobus sia soggetto a scarsi investimenti, e questo incide notevolmente sulla qualità del servizio offerto. La preferenza per l'auto è spesso legata anche alla comodità, alla libertà di movimento, alla percezione di maggiore sicurezza che offre rispetto ai mezzi pubblici, ma anche al maggior senso di rapidità nello spostarsi rispetto all'autobus. Inoltre in troppe parti del paese bus e treni non sono ritenuti adeguatamente funzionanti [6].

Un altro fenomeno generato dall'elevato numero di veicoli presenti sulle strade è il traffico fantasma (o Phantom Traffic Jam). Questo fenomeno si verifica quando una lieve variazione nella velocità di un singolo veicolo all'interno di una colonna di traffico può generare un blocco improvviso e apparentemente inspiegabile. In altre parole, il flusso di veicoli sembra subire un improvviso rallentamento o addirittura fermarsi, anche se non vi è alcuna causa diretta, come un incidente o un semaforo rosso. Inoltre, in base alla densità di auto presenti, questo ingorgo fantasma può viaggiare all'indietro nella colonna di traffico e assumere dimensioni sempre più grandi, fino a creare un vero e proprio blocco di traffico.

In questo lavoro si è deciso di avere un approccio al traffico fantasma diverso rispetto a quelli proposti nella letteratura scientifica. Infatti, l'idea è quella di non considerare il Phantom Traffic Jam come un problema da risolvere, ma di sfruttarne le sue caratteristiche per agevolare il passaggio di veicoli prioritari, come i mezzi di soccorso, gli autobus o altri veicoli di rilevante importanza, attraverso punti critici della rete stradale, come incroci, rotatorie, entrate o uscite autostradali. L'obiettivo è quello di rallentare temporaneamente il flusso di veicoli non prioritari in prossimità di un punto critico della strada per consentire ai veicoli con alta priorità di superarlo senza fermarsi, diminuendone il tempo di percorrenza e migliorando l'efficienza complessiva del sistema soccorso o di trasporto pubblico.

La mia tesi si propone di affrontare questa sfida proponendo un sistema innovativo che, inserendosi in un contesto di Smart Cities e di veicoli autonomi, favorisca l'utilizzo dei mezzi pubblici, in particolare degli autobus, come alternativa preferenziale all'auto privata. L'obiettivo è sviluppare un sistema basato sul concetto del "Phantom Traffic Jam" che consenta agli autobus di percorrere le strade in modo più rapido ed efficiente, garantendo un servizio più veloce e affidabile rispetto all'uso dell'auto privata, e che vada a migliorare i tempi di percorrenza attuali. Questo sistema non solo migliorerebbe la mobilità urbana, ma potrebbe anche incentivare una maggiore adozione dei mezzi pubblici, riducendo così la congestione stradale e l'inquinamento atmosferico.

Struttura e Obiettivi

Il documento è organizzato in diverse sezioni che, partendo dall'esplorazione dei vari aspetti del traffico veicolare fino ad arrivare alla spiegazione di varie tecnologie emergenti in questo campo, introducono al modello di gestione del traffico veicolare

proposto. In particolare i vari capitoli affrontano i vari aspetti del modello Jamact e le sue applicazioni nel contesto delle Smart Cities, con un particolare focus sui veicoli autonomi e connessi. La struttura del documento è concepita per fornire una comprensione completa del fenomeno del traffico fantasma, delle soluzioni proposte e dell'implementazione del modello Jamact.

Inizialmente, il primo capitolo [1, Introduzione] fornisce una panoramica generale degli argomenti trattati, delineando gli obiettivi principali e sottolineando l'importanza dello studio. Questo capitolo introduce il lettore ai temi chiave, stabilendo il contesto e le motivazioni alla base della ricerca.

una panoramica del problema del traffico fantasma e introduce il modello Jamact come una soluzione innovativa. Vengono descritti gli obiettivi generali del documento e l'importanza della gestione intelligente del traffico nelle Smart Cities.

Il secondo capitolo [2, Aspetti preliminari] fornisce una base teorica e tecnica sugli elementi fondamentali necessari per comprendere il contesto in cui si inserisce il modello Jamact. Include una discussione sulle reti veicolari ad hoc (Vanet), le Smart Cities, i veicoli connessi e autonomi, e gli strumenti di programmazione utilizzati, come Prolog e Python. Questi aspetti preliminari sono essenziali per contestualizzare le tecniche utilizzate nella modellazione e simulazione.

Il terzo capitolo [3, Il Phantom Traffic Jam] si concentra sul fenomeno del traffico fantasma (Phantom Traffic Jam), esaminando i cosiddetti "jamitons" e la letteratura scientifica rilevante. Viene inoltre presentato il modello Lighthill-Whitham-Richards, che descrive il traffico come un fluido, e vengono analizzati vari approcci per ridurre l'impatto di questo tipo di congestione.

Successivamente, il quarto capitolo [4, Il modello Jamact], dedicato al Modello Jamact, descrive nel dettaglio il modello proposto. Si parte dalle basi teoriche del modello, analizzando le criticità del sistema e proponendo possibili soluzioni. Viene presentato un esempio teorico e viene discusso come il modello Jamact possa essere isomorfico alla gestione della Safety Car nelle gare automobilistiche, oltre che alla sua applicazione per l'ottimizzazione delle risorse.

Nel capitolo successivo [5, Simulatore per il modello Jamact] si fornisce una panoramica sul codice del simulatore sviluppato per testare il modello, per simulare la differente gestione dei veicoli in un incrocio ideale. Viene mostrato un esempio di applicazione dinamica e vengono discussi i risultati delle simulazioni eseguite, evidenziando l'efficacia del modello nella gestione del traffico. Inoltre i dati presentati tengono conto delle simulazioni prima applicando le regole del modello Jamact e poi applicando semplice regole di precedenza e di arrivo al punto critico.

Il sesto capitolo [6, Modello Jamact per Autobus] estende l'applicazione del modello Jamact al trasporto pubblico. Viene presentata una panoramica sull'utilizzo degli autobus, seguita da un'estensione del modello Jamact per questo tipo di veicoli. Viene affrontato il problema del Commesso Viaggiatore e viene discusso l'isomorfismo con il trasporto ferroviario e le coincidenze dei treni. Il capitolo si conclude con un esempio di applicazione dinamica del modello esteso e i risultati delle simulazioni.

Infine, il settimo capitolo [7, Conclusioni e sviluppi futuri] riassume i principali risultati ottenuti e propone possibili direzioni per future ricerche e miglioramenti del modello.

Le appendici [A, B] forniscono ulteriori dettagli tecnici e teorici, mentre la bibliografia elenca tutte le fonti citate.

L’obiettivo di questo lavoro non è solo comprendere il fenomeno del Phantom Traffic Jam, ma anche esplorarne le potenziali applicazioni come soluzione innovativa per migliorare la mobilità urbana e garantire un transito più fluido e sicuro per i veicoli con priorità di passaggio. Attraverso i vari capitoli, il lettore acquisirà una chiara comprensione sia del fenomeno in sé, sia delle modalità con cui viene analizzato per sviluppare il modello Jamact. Grazie a un approfondimento graduale e dettagliato degli argomenti, il modello Jamact risulterà di facile comprensione per qualsiasi lettore. Inoltre, il modello viene testato attraverso un simulatore appositamente sviluppato, e i risultati delle simulazioni vengono analizzati per valutare l’efficacia delle soluzioni proposte. Un ulteriore obiettivo è l’estensione del modello Jamact al trasporto pubblico, con particolare attenzione agli autobus, per migliorare l’efficienza e la puntualità del servizio. L’obiettivo finale è quello di fornire una chiara comprensione dell’idea alla base del modello Jamact, delle sue possibili implementazioni teoriche, e dei suoi potenziali campi di applicazione e sviluppo futuro.

–2–

Aspetti preliminari

CONTENUTI: **2.1 Reti Veicolari ad Hoc: Vanet.** **2.2 Smart Cities.** **2.3 Veicoli Connessi e Veicoli Autonomi.** **2.4 Prolog.** **2.5 Python.**

In questo capitolo sarà fornita una breve panoramica sulle reti veicolari intelligenti (VANET), le Smart Cities e i veicoli autonomi nel contesto dell’evoluzione urbana. Inoltre verranno spiegate le tecnologie principali utilizzate nello sviluppo del modello Jamact.

Il modello proposto in questa tesi si basa sull’utilizzo di svariate tecnologie che stanno emergendo nel campo dei trasporti e della mobilità. Nel contesto dell’evoluzione delle infrastrutture urbane e dei sistemi di trasporto, è importante citare concetti chiave come l’Internet delle cose (IoT)¹ e i Sistemi di Trasporto Intelligenti (ITS)². Queste tecnologie hanno rivoluzionato il modo in cui si interagisce con l’ambiente urbano e ci si sposta da un luogo all’altro. In questo ottica, sono di notevole importanza le reti veicolari intelligenti (VANET) [7, 8], le quali hanno assunto un ruolo centrale nell’evoluzione delle Smart Cities e nell’integrazione dei veicoli autonomi nelle infrastrutture urbane. Questo capitolo si propone di esplorare in dettaglio le VANET e il loro impatto sulle Smart Cities, nonché l’evoluzione dei veicoli autonomi e le diverse forme di interazione con essi. Saranno inoltre esaminati approfonditamente i protocolli di comunicazione che costituiscono il fondamento delle VANET. Infine, verranno presentati gli strumenti di sviluppo software utilizzati nell’ambito di questa ricerca, con particolare attenzione a Python e Prolog, le cui funzionalità e vantaggi verranno analizzati in relazione agli obiettivi proposti.

Al fine di comprendere meglio i motivi per cui vengono utilizzati, ciascun elemento discusso sarà accompagnato da spiegazioni su come si intende utilizzarlo nel modello Jamact.

¹L’Internet delle cose (IoT) è un sistema di interconnessione di dispositivi fisici attraverso Internet, dotati di sensori, attuatori, software e connettività che consentono loro di raccogliere, scambiare e analizzare dati.

²I Sistemi di Trasporto Intelligenti (ITS) integrano tecnologie dell’informazione e della comunicazione nei trasporti per migliorare efficienza, sicurezza e sostenibilità. Utilizzano sensori e sistemi di gestione dei dati per raccogliere e analizzare informazioni sul traffico e le condizioni stradali, fornendo soluzioni intelligenti per la gestione del traffico, la navigazione e la sicurezza stradale.

2.1 Reti Veicolari ad Hoc: Vanet

Le reti ad hoc rappresentano una forma di rete informatica dinamica e decentralizzata, in cui i nodi comunicano direttamente tra loro senza ricorrere all'utilizzo di un'infrastruttura centralizzata o di un punto di accesso. Questo le rende particolarmente adatte per situazioni in cui l'accesso alla rete tradizionale è limitato o inesistente. Le reti ad hoc sono caratterizzate da diverse proprietà, tra cui l'autonomia dei nodi, la capacità di auto-organizzazione e la flessibilità nel cambiare la topologia della rete. In una rete ad hoc, ogni nodo funge sia da trasmettitore che da ricevitore, e i nodi possono cooperare tra loro per instradare i dati verso la loro destinazione finale [9]. Un esempio di reti ad-hoc wireless sono le reti VANET (Vehicular Ad-hoc Networks).

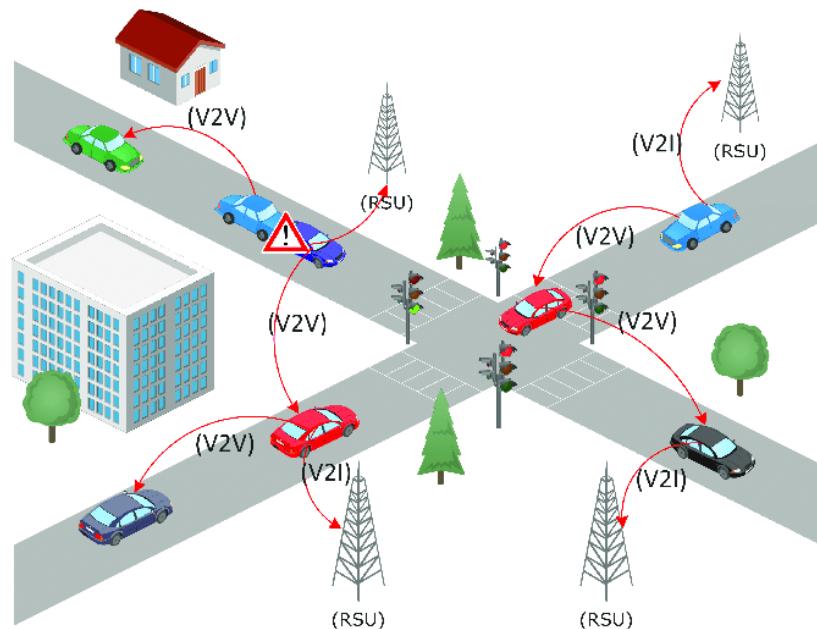


Figura 2.1: Esempio di rete vanet, [10]

Le VANET sono considerate un tipo di rete ad-hoc wireless, poiché non esiste un'entità di coordinamento centrale che gestisca la comunicazione tra i nodi, e si basano sulla comunicazione tra veicoli e infrastrutture fisse, come ad esempio semafori o stazioni di servizio. In queste reti i nodi si autorganizzano e si scambiano informazioni tra loro, utilizzando protocolli di comunicazione specifici come ad esempio lo standard IEEE 802.11p. Le reti VANET sono state applicate in diversi contesti, come ad esempio la gestione del traffico, la sicurezza stradale e la gestione delle emergenze [11, 12].

Le VANETs permettono agli autoveicoli di comunicare tra loro e con le infrastrutture stradali (V2X), ma richiedono una specifica infrastruttura per funzionare:

- Le Unità di Bordo dei Veicoli (OBUs) sono fondamentali, montate sui veicoli per consentire comunicazioni via Wi-Fi, Bluetooth e DSRC, facilitando lo scambio di informazioni istantaneo e affidabile.

- Le infrastrutture stradali, che includono sensori, telecamere e semafori intelligenti, fungono da punti di accesso per i veicoli, fornendo dati in tempo reale sul traffico e sull'ambiente circostante.
- Le infrastrutture di comunicazione, come server centrali e stazioni base, agiscono come nodi intermedi per la trasmissione e l'elaborazione dei dati.
- Le RSU (Road Side Unit), sono infrastrutture statiche che fungono da punti di accesso per i veicoli nelle reti di comunicazione veicolare (VANET).

In breve, OBUS, infrastrutture stradali e di comunicazione lavorano insieme per creare un ecosistema interconnesso nel quale sono di fondamentale importanza i tipi di comunicazione che possono instaurarsi tra i veicoli e qualsiasi elemento dell'ecosistema stradale [13]:

- V2V (Vehicle-to-Vehicle): Questo si riferisce alla comunicazione diretta tra veicoli. I veicoli connessi V2V possono scambiare informazioni tra loro, come dati sulla velocità, la posizione, le condizioni del traffico e gli avvisi di sicurezza, al fine di migliorare la sicurezza stradale e coordinare il movimento.
- V2I (Vehicle-to-Infrastructure): Questo riguarda la comunicazione tra i veicoli e l'infrastruttura stradale, come semafori intelligenti o segnaletica stradale connessa. Gli veicoli possono ricevere informazioni da queste infrastrutture per ottimizzare il flusso del traffico, evitare congestioni e migliorare la sicurezza.
- V2R (Vehicle-to-Roadside): Questo termine è spesso usato in modo intercambiabile con V2I. Si riferisce alla comunicazione tra veicoli e apparecchiature sulla strada, come sensori o sistemi di controllo del traffico.
- V2S (Vehicle-to-Server): Questo rappresenta la comunicazione tra veicoli e server remoti o sistemi centralizzati. Può essere utilizzato per aggiornamenti software, servizi connessi, mappe in tempo reale e altre funzionalità basate su cloud.
- V2X (Vehicle-to-Everything): Si riferisce alla capacità dei veicoli di comunicare con qualsiasi altra entità nell'ecosistema stradale.
- DSRC (Dedicated Short-Range Communications): Il DOT definisce la comunicazione V2I e V2V attraverso il termine DSRC, una tecnologia wireless per collegare veicoli e infrastrutture su brevi distanze.

Nel modello Jamact, le reti VANET serviranno per garantire una comunicazione rapida ed efficiente tra i veicoli presenti nel sistema e per permettere lo scambio di informazioni tra veicolo e server.

2.2 Smart Cities

Le Smart Cities [14] sono un concetto che si sta sempre più diffondendo in tutto il mondo, con l'obiettivo di creare città più intelligenti, connesse e sostenibili. Questo fenomeno è caratterizzato dall'utilizzo di tecnologie innovative come Internet of Things (IoT), Intelligenza Artificiale, Big Data, Cloud Computing, Digital Twins, e molti altri strumenti tecnologici per migliorare la gestione dei servizi pubblici e la qualità della vita dei cittadini. Le moderne città stanno affrontando diverse sfide, tra cui la gestione intelligente della mobilità, il potenziamento della sicurezza, l'attenzione all'ambiente e alla gestione sociale ed economica più equa. Le Smart Cities stanno cercando di utilizzare le tecnologie per risolvere questi problemi e migliorare la qualità della vita dei loro abitanti, e si caratterizzano per l'integrazione di sistemi di comunicazione, mobilità, ambiente ed efficienza energetica [15, 16]. Le Smart Cities, inoltre, si basano su reti di comunicazione elettronica all'avanguardia, come le reti 5G e le reti veicolari ad-hoc (VANET), per abilitare servizi innovativi e connettività, e grazie a queste reti raccolgono e analizzano grandi quantità di dati provenienti da sensori e dispositivi connessi per ottimizzare i servizi e prendere decisioni informate.

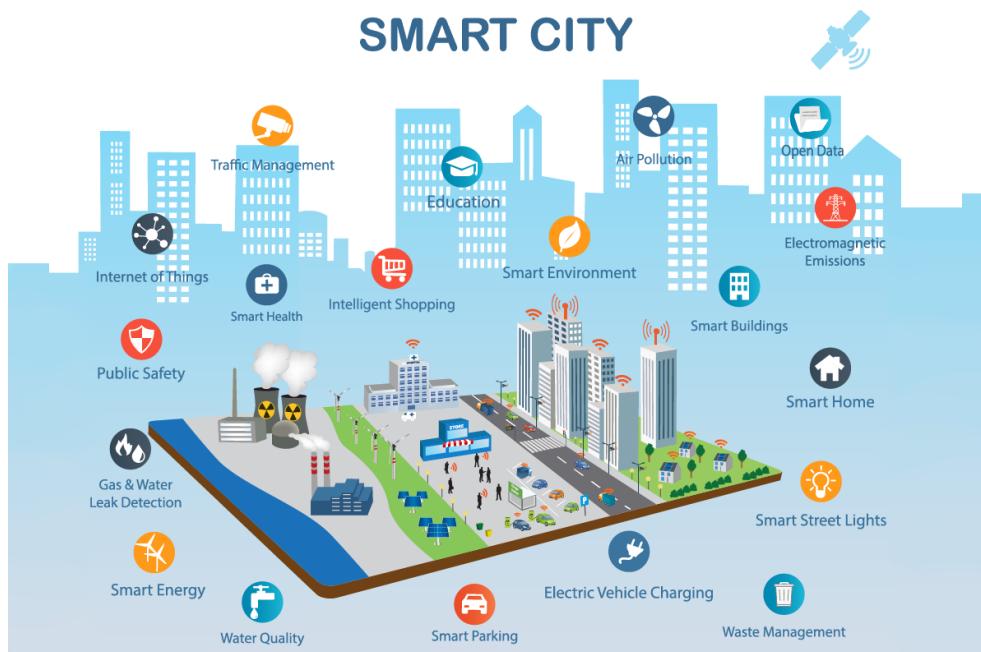


Figura 2.2: Esempio di Smart Cities, [17]

2.3 Veicoli Connessi e Veicoli Autonomi

I veicoli connessi e autonomi rappresentano una tecnologia innovativa con un enorme potenziale nel ridurre gli incidenti stradali, migliorare l'efficienza dei trasporti e aumentare la qualità della vita. Un *veicolo connesso* è un'automobile equipaggiata con tecnologie di comunicazione e connettività che consentono lo scambio di dati e informazioni con altri veicoli, infrastrutture stradali e sistemi centralizzati. Questa

comunicazione avviene attraverso reti wireless come il 4G o il 5G e coinvolge una varietà di sensori e dispositivi incorporati nel veicolo. I *veicoli autonomi* sono veicoli capaci di guidare senza necessità di un conducente umano. Grazie all'intelligenza artificiale e alla sensoristica avanzata, i veicoli autonomi sono in grado di guidare in modo sicuro e efficiente senza l'intervento umano diretto. La differenza con i veicoli autonomi è che quest'ultimi sono in grado di guidare anche senza un conducente umano. La Society of Automotive Engineers (SAE) ha delineato sei livelli di automazione [fig. 2.3] veicolare che indicano il grado di autonomia del veicolo nelle sue funzioni di guida. Questi livelli vanno dal primo, che richiede la presenza umana per la guida, al sesto, in cui il veicolo è completamente autonomo e non richiede interventi umani.

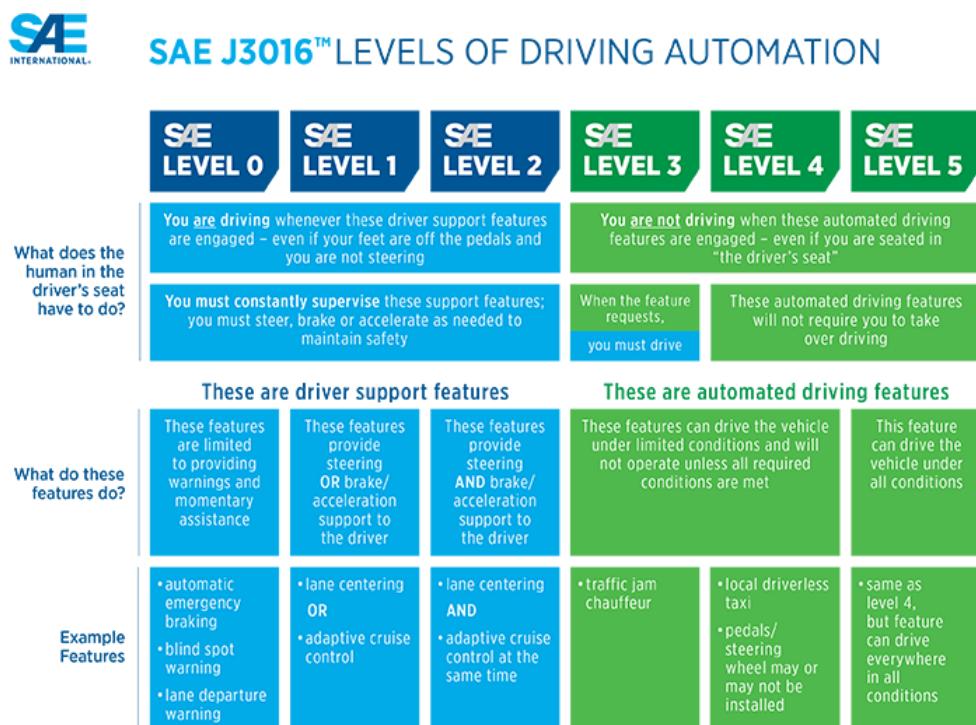


Figura 2.3: Classificazione livelli di automazione di un veicolo, [18]

I veicoli connessi, in particolare quelli automatizzati e autonomi, impiegano diverse tecnologie di bordo per determinare la posizione, monitorare le condizioni circostanti al veicolo, percepire oggetti e ostacoli o comunicare con altri veicoli e infrastrutture stradali. Nel modello verranno sfruttate queste funzionalità per comunicare informazioni fondamentali ai veicoli, i quali regoleranno il loro comportamento in base a queste informazioni durante la guida.

2.4 Prolog

Prolog è un linguaggio di programmazione logica utilizzato principalmente per l'intelligenza artificiale e la linguistica computazionale. È basato sul calcolo predicativo e

permette di definire relazioni logiche tra dati e di interrogare tali relazioni. La programmazione in Prolog si fonda su fatti, regole e interrogazioni. In sintesi, Prolog è un linguaggio di programmazione basato sulla logica, ideale per problemi che possono essere espressi tramite relazioni logiche e regole.

In Prolog, i programmi sono espressi come insiemi di regole logiche e fatti. Un fatto è una dichiarazione che è vera. Una regola è una dichiarazione che specifica una condizione sotto la quale qualcosa è vera. Ad esempio:

```
umano(Socrate).
mammifero(cane).
padre(Abramo, Isacco).
genitore(X, Y) :- padre(X, Y).
```

La prima riga dichiara che Socrate è umano, la seconda che i cani sono mammiferi. La terza riga dichiara che Abramo è il padre di Isacco. L'ultima riga definisce una regola che afferma che X è un genitore di Y se X è il padre di Y.

Un aspetto cruciale di Prolog è il meccanismo di *inferenza*, che utilizza la risoluzione per dimostrare query logiche. Quando si pone una domanda a Prolog, il sistema cerca di dimostrare la verità della query basandosi sui fatti e le regole definite. Ad esempio, ponendo la domanda `mortale(Socrate)`, Prolog cercherà di verificare se esiste una dimostrazione a partire dai fatti e dalle regole disponibili.

Prolog è particolarmente potente per la gestione di problemi che richiedono la ricerca di soluzioni in spazi di possibilità, come i puzzle, i giochi e l'analisi del linguaggio naturale. La sua sintassi semplice e la sua capacità di esprimere direttamente la logica dei problemi lo rendono uno strumento prezioso per i ricercatori e i programmatore.

Un altro concetto importante in Prolog è la *ricorsione*, che consente di definire regole che si richiamano. Questo è utile per lavorare con strutture dati come liste o alberi. Ad esempio, una regola per calcolare la lunghezza di una lista può essere definita ricorsivamente in Prolog, consentendo di elaborare liste di qualsiasi dimensione.

La capacità di Prolog di esprimere direttamente queste regole logiche rende il linguaggio particolarmente adatto per applicazioni in cui le decisioni devono essere prese in base a un insieme di condizioni complesse. Invece di scrivere codice procedurale dettagliato, si definiscono semplicemente i fatti e le regole logiche che governano il comportamento desiderato. Questo approccio non solo riduce la complessità del codice, ma lo rende anche più leggibile e facile da mantenere.

Nel contesto di questa tesi, Prolog si propone come linguaggio ideale per la definizione di alcune regole di comportamento del sistema, utilizzate per descrivere condizioni sotto le quali certe azioni sonomesse o proibite.

2.5 Python

Python è un linguaggio di programmazione di alto livello, interpretato e a tipizzazione dinamica, creato da Guido van Rossum e rilasciato per la prima volta nel 1991.

Python è conosciuto per la sua sintassi chiara e leggibile, che favorisce uno stile di programmazione elegante e conciso, consentendo ai programmatore di esprimere concetti in meno righe di codice rispetto a molti altri linguaggi. La filosofia di design di Python enfatizza la leggibilità del codice, rendendolo un linguaggio ideale sia per i principianti che per gli esperti.

Python possiede diverse caratteristiche che lo rendono un linguaggio versatile e potente:

- **Semplicità:** La sintassi semplice e lineare di Python facilita la scrittura e la lettura del codice, riducendo la complessità associata alla programmazione.
- **Interprete:** Python è un linguaggio interpretato, il che significa che il codice può essere eseguito direttamente senza la necessità di una fase di compilazione. Questo permette una rapida sperimentazione e sviluppo.
- **Tipizzazione Dinamica:** In Python, le variabili non necessitano di una dichiarazione esplicita del tipo; il tipo viene determinato automaticamente durante l'esecuzione del programma.
- **Librerie Standard e di Terze Parti:** Python possiede una vasta libreria standard che fornisce moduli e funzioni per molte attività comuni. Inoltre, esistono numerose librerie di terze parti che estendono le capacità di Python in vari campi, come la scienza dei dati, l'intelligenza artificiale, il web development, e molti altri.
- **Portabilità:** Python è un linguaggio multi piattaforma, il che significa che il codice scritto in Python può essere eseguito su diverse piattaforme senza necessità di modifiche.
- **Community e Supporto:** Python ha una delle community più attive e ampie nel mondo della programmazione. Questo garantisce un supporto continuo e una vasta disponibilità di risorse e documentazione.

Nel contesto delle Smart Cities e della mobilità urbana, Python è utilizzato per sviluppare modelli di simulazione, analizzare flussi di traffico, implementare algoritmi di ottimizzazione e gestire grandi quantità di dati provenienti da sensori e dispositivi IoT. Grazie alla sua versatilità e alle potenti librerie di analisi dei dati, Python è uno strumento ideale per affrontare le sfide complesse e dinamiche della gestione del traffico.

Python si distingue come uno strumento fondamentale nella ricerca e nello sviluppo tecnologico grazie alla sua semplicità, versatilità e potenza. La vasta gamma di librerie e il forte supporto della community lo rendono un linguaggio ideale per affrontare problemi complessi, come la gestione del traffico nelle Smart Cities. Nel contesto di questa tesi, Python sarà utilizzato per implementare e testare il modello Jamact, sfruttando le sue capacità di prototipazione rapida, analisi dei dati e integrazione con altre tecnologie.

—3—

Il Phantom Traffic Jam

CONTENUTI: 3.1 jamiton. 3.2 Letteratura scientifica sul Phantom Traffic Jam. 3.3 Il traffico come un fluido: il modello Lighthill-Whitham-Richards. 3.4 Approcci alla riduzione dell'impatto del traffico fantasma.

In questo capitolo sarà fornita una breve panoramica sugli ingorghi fantasma, analizzando la loro formazione e l'approccio della letteratura scientifica a tale argomento. Infine verrà discusso il fenomeno del Phantom Traffic Jam sia come problema che come soluzione ad un problema.

Il fenomeno del "Phantom Traffic Jam" è un'esperienza comune per molti automobilisti, che si manifesta come un ingorgo di traffico improvviso e apparentemente senza causa, spesso senza alcun segnale di incidente o lavori in corso. Questo tipo di ingorgo è stato oggetto di studio da parte di ricercatori e scienziati, che hanno cercato di comprenderne le cause e le caratteristiche.

Secondo gli studiosi, il Phantom Traffic Jam è causato da una serie di fattori che interagiscono tra loro, tra cui la densità del traffico, il comportamento dei conducenti e le caratteristiche delle strade. In particolare, quando il traffico è molto denso, i conducenti tendono a reagire in modo più aggressivo alle variazioni del traffico, come ad esempio un'auto che rallenta improvvisamente. Questa reazione può creare una catena di eventi che si propagano indietro lungo la strada, causando un ingorgo di traffico che può durare per diversi chilometri.

La letteratura scientifica ha affrontato questo fenomeno utilizzando diversi approcci, tra cui la modellazione matematica, l'analisi dei dati di traffico e gli esperimenti in laboratorio. Inoltre, sono state anche esplorate possibili soluzioni per ridurre l'impatto e la formazione di ingorghi fantasma. Ad esempio, gli studiosi hanno suggerito di mantenere una distanza di sicurezza sufficiente tra le auto, di non accelerare e frenare bruscamente e di utilizzare tecnologie di guida assistita per aiutare i conducenti a mantenere una velocità costante e una distanza di sicurezza sufficiente.

Ad oggi, la maggior parte degli studi condotti sul Phantom Traffic Jam vede un approccio volto a risolverlo come un problema. La maggior parte degli esperimenti e dei modelli sviluppati hanno avuto come unico obiettivo la ricerca di una soluzione innovativa e affidabile a questo fenomeno. Diversi fattori sono stati identificati come contribuenti alla formazione degli ingorghi fantasma, tra cui il comportamento dei

conducenti, come la mancata osservanza della distanza di sicurezza e i repentina cambi di corsia; le condizioni della strada, quali la presenza di buche o superfici irregolari, strade strette o con numerose curve; le condizioni meteorologiche, come pioggia o nebbia che possono ridurre la visibilità.

3.1 jamiton

Il termine jamiton fu coniato per la prima volta nel 2009 da un gruppo di ricercatori della National Science Foundation ed è utilizzato per descrivere le onde di congestione che si propagano all'indietro attraverso il flusso del traffico stradale, causando ingorghi e rallentamenti. Queste onde nascono da piccole perturbazioni in un flusso di traffico uniforme, come un dosso sulla strada, o un guidatore che frena dopo un momento di disattenzione. Anche quando le auto abbandonano quest'onda, essa non scompare ma si sposta gradualmente all'indietro, contro la direzione del traffico e inizia solitamente con i veicoli che incontrano un improvviso aumento di densità iniziale e un calo di velocità [19]. Queste onde di congestione sono simili ai solitoni¹ [20] nella loro natura, poiché si comportano come entità distinte e conservano la loro forma e velocità mentre si propagano attraverso il traffico.

Sebbene i solitoni si verifichino in sistemi fisici diversi, come le onde oceaniche o le onde luminose in una fibra ottica, e i jamiton si verifichino nel traffico stradale, entrambi condividono alcune caratteristiche chiave:

1. **Mantenimento della forma e velocità:** come i solitoni, i jamiton mantengono la loro forma e la loro velocità mentre si propagano attraverso il flusso del traffico. Anche se causano congestione e ingorghi, conservano le loro caratteristiche distinte mentre si muovono lungo la strada.
2. **Propagazione senza dispersione:** i jamiton si propagano all'indietro attraverso il flusso del traffico senza dissiparsi o deformarsi significativamente lungo il percorso. Questo è simile al comportamento dei solitoni che mantengono la loro integrità mentre si muovono attraverso un mezzo.
3. **Effetti localizzati:** sia i solitoni che i jamiton hanno effetti localizzati sulle loro vicinanze. Mentre attraversano il mezzo, influenzano il comportamento dei sistemi circostanti, causando ingorghi o deformazioni localizzate nel caso dei jamiton, o altre interazioni nel caso dei solitoni.

3.2 Letteratura scientifica sul Phantom Traffic Jam

Gli scienziati del MIT hanno condotto una serie di studi teorici e sperimentali per comprendere e modellare gli ingorghi di traffico fantasma, noti anche come "jamiton".

¹I solitoni sono onde che mantengono la loro forma e velocità durante il movimento, senza disperdersi o fondersi con altre onde. Questo fenomeno è cruciale per comprendere il comportamento delle onde in contesti come fibre ottiche e canali oceanici, e ha applicazioni pratiche in telecomunicazioni e fisica dei fluidi.

Questi studi [21] si concentrano sui modelli continui del traffico che descrivono il flusso dei veicoli attraverso equazioni matematiche simili a quelle utilizzate per i flussi di fluidi. I modelli principali utilizzati sono il modello di Payne-Whitham e il modello di Aw-Rascle, che descrivono il traffico come un flusso continuo, considerando la densità e la velocità dei veicoli come funzioni continue. Gli scienziati del MIT hanno scoperto che, sopra una certa soglia critica di densità del traffico, il flusso diventa instabile e piccole perturbazioni si amplificano, portando alla formazione di onde di alta densità di traffico, appunto i jamiton. La velocità del traffico diminuisce con l'aumentare della densità e i conducenti hanno un tempo di adattamento per reagire alle nuove condizioni del traffico. Anche con conducenti che seguono le stesse leggi in modo prevedibile, gli ingorghi possono ancora verificarsi.

Tramite un video molto esemplificativo [22], gli studiosi hanno utilizzato un'autostrada circolare per spiegare il comportamento del traffico. Come si può notare dalle immagini [fig. 3.1], piccole variazioni nel traffico hanno creato onde di traffico che si propagano indietro lungo la strada, fino a rallentare e quasi fermare le auto senza un motivo particolare.

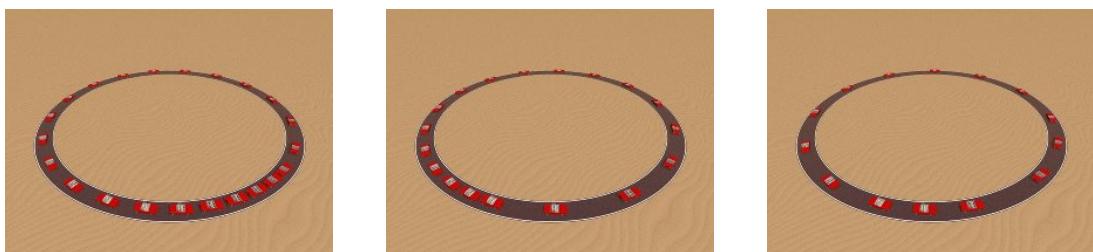


Figura 3.1: MIT: Traffico fantasma strada circolare

Altri studi hanno identificato tre tipi di ingorghi fantasma che contribuiscono al Phantom Traffic Jam:

- Wide Moving Jam [fig. 3.2]
- Butterfly Effect [fig. 3.3]
- Bottleneck [fig. 3.4]

The Wide Moving Jam

Il fenomeno conosciuto come "wide moving jam" si origina dalle variazioni di accelerazione e decelerazione dei veicoli presenti in ciascuna corsia della strada. Queste fluttuazioni non uniformi di velocità generano un'onda di congestione che si propaga lungo l'arteria stradale, determinando alla fine il blocco del flusso veicolare. Un fattore chiave che alimenta questo tipo di ingorghi è il "tailgating", praticato da conducenti che seguono troppo da vicino il veicolo davanti. Quando il veicolo antecedente riduce bruscamente la velocità, i conducenti che lo seguono, privi dello spazio di frenata adeguato, sono costretti a fermarsi repentinamente. Tale situazione innesta una reazione a catena che può condurre alla formazione di un "wide moving jam".

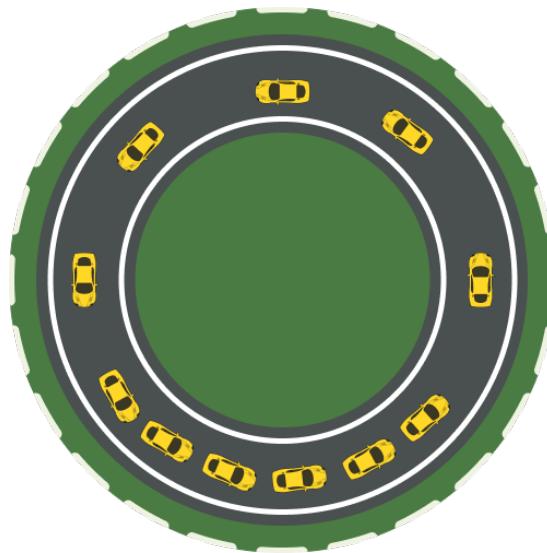


Figura 3.2: The Wide Moving Jam

The Butterfly Effect

Questo fenomeno si manifesta quando una piccola perturbazione nel regolare flusso del traffico, come un veicolo che cambia corsia, innesta una sequenza di eventi che porta a un rallentamento generale. Il meccanismo di funzionamento tipico è il seguente: un veicolo che sta per cambiare corsia riduce la velocità, causando un rallentamento nella corsia di partenza. Durante il cambio di corsia, il veicolo che precede il cambiante di solito deve ridurre la velocità per consentire l'ingresso nella sua corsia. Questo comporta un ulteriore rallentamento nella corsia adiacente. Nel frattempo, i veicoli di questa seconda corsia tenderanno a spostarsi nella corsia percepita come più veloce, se possibile, perpetuando così il ciclo in ciascuna corsia.

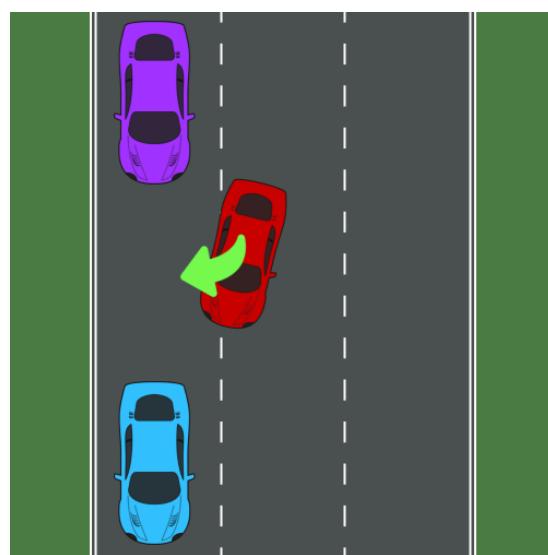


Figura 3.3: The Butterfly Effect

The Bottleneck

Il Bottleneck si verifica quando la capacità di una carreggiata si riduce improvvisamente, come avviene quando un tratto di autostrada a tre corsie si restringe a due corsie. Questo repentino restringimento costringe i veicoli a rallentare mentre il flusso del traffico si adatta alla nuova configurazione. Di conseguenza, il traffico che precede il punto di restringimento, magari viaggiando senza intoppi, è costretto a decelerare e, in ultima istanza, a fermarsi.

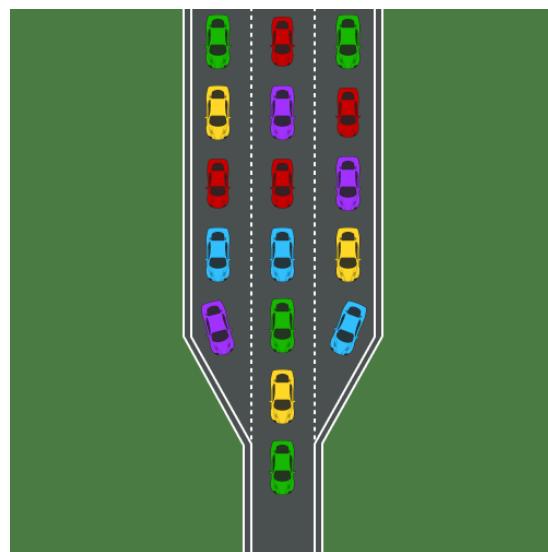


Figura 3.4: The Bottleneck

3.3 Il traffico come un fluido: il modello Lighthill-Whitham-Richards

Il modello Lighthill-Whitham-Richards (LWR) è una teoria che modella il traffico stradale come un flusso continuo, simile al movimento di un fluido in un tubo. Questo modello fornisce un modo per comprendere e prevedere il comportamento del traffico utilizzando concetti analoghi a quelli della fluidodinamica.

Nel modello LWR, le analogie tra traffico e fluido emergono nei seguenti aspetti:

1. **Densità del traffico e densità del fluido:** La densità del traffico sulla strada è assimilabile alla densità del fluido nel tubo. Entrambe le grandezze rappresentano il numero di veicoli (nel caso del traffico) o molecole (nel caso del fluido) presenti in una certa area.
2. **Velocità del traffico e velocità del fluido:** La velocità media dei veicoli che si muovono lungo la strada è paragonabile alla velocità media del fluido nel tubo. Entrambe le grandezze indicano quanto velocemente si muovono i veicoli o le molecole di fluido in una determinata area.

3. **Flusso di traffico e flusso di fluido:** Il flusso di traffico, cioè il numero di veicoli che attraversa una certa sezione della strada in un dato intervallo di tempo, è analogo al flusso di fluido, che rappresenta il numero di molecole di fluido che attraversano una sezione del tubo in un certo intervallo di tempo.
4. **Equazioni di conservazione del flusso:** Le equazioni utilizzate nel modello LWR, come l'equazione di conservazione del flusso e la relazione flusso-densità, sono simili alle equazioni di conservazione della massa e alle leggi di flusso della fluidodinamica. Queste equazioni descrivono come il flusso (traffico o fluido) è regolato dalla densità e dalla velocità del mezzo.

Quindi, basandoci sul modello LWR, si può dire dire che matematicamente il traffico assomiglia molto a un fluido. Utilizzando le equazioni della fluidodinamica, ogni macchina è sostituita concettualmente da una piccola molecola d'acqua, trasformando la strada in una sorta di fiume. È stato dimostrato che esiste una densità critica di veicoli, al di sopra della quale anche una piccola perturbazione, come un rallentamento, può amplificarsi fino a provocare un ingorgo. La formazione e la crescita di questi ingorghi dipendono dalla densità di veicoli, dalla loro distanza e dalla qualità delle azioni dei guidatori, inclusi i loro tempi di reazione. Se la densità di veicoli supera quella critica, ovvero c'è molto traffico, l'ingorgo può propagarsi all'indietro nel traffico, fino a che non raggiunge una zona con densità inferiore, dove può dissolversi.

Teoria delle Onde di Shock

La teoria delle onde di shock, originariamente sviluppata in ambito fluidodinamico, è applicata anche ai modelli di traffico per spiegare la formazione e la propagazione degli ingorghi fantasma. Questo approccio consente di comprendere come perturbazioni nel flusso di traffico possano evolversi in onde di congestione che si propagano contro il flusso del traffico stesso.

La base matematica della teoria delle onde di shock nel contesto del traffico è data dall'equazione di continuità, che esprime la conservazione del numero di veicoli su un tratto di strada. L'equazione è la seguente:

$$\frac{\partial \rho(x, t)}{\partial t} + \frac{\partial(\rho(x, t)v(x, t))}{\partial x} = 0 \quad (3.1)$$

dove:

- $\rho(x, t)$: è la densità del traffico, ovvero il numero di veicoli per unità di lunghezza, in funzione della posizione x e del tempo t .
- $v(x, t)$: è la velocità media del traffico alla posizione x e al tempo t .

Questa equazione descrive come la densità dei veicoli cambia nel tempo e nello spazio in relazione alla velocità del flusso di traffico.

Relazione Velocità-Densità

Un elemento chiave nella teoria delle onde di shock è la relazione tra la velocità del traffico v e la densità ρ . Questa relazione è solitamente espressa come:

$$v = V(\rho) \quad (3.2)$$

dove $V(\rho)$ è una funzione decrescente della densità del traffico. Un esempio tipico di funzione velocità-densità è:

$$V(\rho) = v_{\max} \left(1 - \frac{\rho}{\rho_{\max}} \right) \quad (3.3)$$

dove:

- v_{\max} : è la velocità libera, ovvero la velocità massima che i veicoli possono raggiungere in condizioni di traffico libero.
- ρ_{\max} : è la densità massima, corrispondente alla situazione in cui i veicoli sono così vicini che la velocità si riduce a zero.

3.4 Approcci alla riduzione dell'impatto del traffico fantasma

Gli studiosi hanno adottato diversi approcci per risolvere o gestire al meglio il problema del traffico fantasma. I ricercatori hanno sviluppato diverse strategie innovative che si concentrano sulla gestione del traffico, sull'ottimizzazione delle infrastrutture e sulla modifica dei comportamenti dei conducenti.

Uno studio recente del 2024 [23] ha indagato sulla possibilità di risolvere il Phantom Traffic Jam utilizzando l'Haptic Shared Control (HSC), che coinvolge sia l'automazione che l'intervento umano. Nell'esperimento sono stati testati gli effetti sulla dinamica del flusso del traffico, confrontandolo con il controllo manuale e l'automazione completa, coinvolgendo 24 partecipanti in un simulatore di guida.

I risultati mostrano che l'HSC contribuisce a dissipare gli ingorghi fantasma meglio del controllo completamente manuale, ma meno efficacemente dell'automazione completa. Inoltre si è osservato che l'HSC riduce la frequenza di situazioni pericolose causate da guasti silenziosi dell'automazione rispetto all'automazione completa.

Parallelamente agli studi menzionati, recenti ricerche hanno ulteriormente esplorato le strategie per mitigare i 'Phantom Traffic Jam'. Un approccio innovativo [24] si è concentrato sull'uso di limiti di velocità dinamici e consigli di velocità in-car, processando i dati di traffico attraverso tecniche di logica fuzzy e clustering. Questo metodo ha dimostrato di stabilizzare il flusso del traffico e ridurre l'incidenza dei 'Phantom Traffic Jam', sottolineando l'importanza di interventi preventivi piuttosto che risolutivi. Tali risultati supportano l'idea che una gestione proattiva del traffico possa essere decisiva nel contrastare le congestioni improvvise e migliorare la sicurezza e l'efficienza delle nostre strade.

Infine un’ulteriore soluzione [25] proposta dagli scienziati coinvolge il mantenimento di uno spazio equo tra i veicoli sulla strada, anziché guidare troppo vicino. Mantenere una distanza adeguata tra i veicoli può agire come un ammortizzatore sugli ingorghi fantasma, impedendo che l’effetto di rallentamento si propaghi. Inoltre, l’uso di tecnologie come i sistemi di gestione del traffico intelligente e la collaborazione tra enti pubblici e privati sono cruciali per ridurre l’impatto dei traffici fantasma sulla congestione stradale e sull’economia.

—4—

Il modello Jamact

CONTENUTI: **4.1 Le basi del modello.** **4.2 Modello Teorico.** **4.3 Criticità del sistema e possibili soluzioni.** **4.3.1 Approcci alla coordinazione tra veicolo prioritario e veicoli non prioritari – 4.3.2 Approcci alla coordinazione di più veicoli prioritari – 4.3.3 Approcci alla gestione di interferenze tra più jamiton.** **4.4 Esempio Teorico.** **4.5 Jamact e Safety Car: un isomorfismo nella gestione veicolare.** **4.6 Jamact e l'ottimizzazione delle risorse.**

In questo capitolo sarà presentato il modello Jamact, il quale si focalizza sull'induzione controllata e strategica di jamiton temporanei al fine di favorire il transito senza impedimenti dei veicoli ad alta priorità. Quindi utilizza a proprio vantaggio gli ingorghi fantasma e le loro caratteristiche, inducendoli in situazioni critiche per agevolare il transito dei veicoli prioritari.

Il modello Jamact si propone di sfruttare le caratteristiche del Phantom Traffic Jam per indurlo in situazioni di criticità veicolare, rappresentando una soluzione innovativa per mitigare le congestioni del traffico in ambienti di Smart Cities. L'obiettivo è l'induzione controllata di rallentamenti temporanei al fine di favorire il flusso dei veicoli autonomi prioritari, che sono connessi attraverso reti VANET (Vehicular Ad-hoc Network). A differenza dei tradizionali jamiton, che emergono spontaneamente a causa di piccoli rallentamenti, il jamiton indotto viene creato strategicamente per ottimizzare la mobilità urbana e garantire un passaggio efficiente ai veicoli prioritari.

Il nome "Jamact" trae ispirazione proprio dalla funzione del modello di indurre un ingorgo e rallentare temporaneamente un blocco di auto. La parola "Jamact" combina "jam" da "jamiton" e "act" da "actuate", indicando l'attivazione intenzionale di un ingorgo.

L'idea di sviluppare il modello Jamact è nata dopo un'attenta valutazione delle problematiche causate dagli ingorghi fantasma e delle numerose proposte di soluzione esistenti in vari ambiti. La chiave di questa riflessione consiste nell'osservare tali congestioni non solo come un problema, ma anche come una potenziale soluzione per altre criticità nel contesto della mobilità urbana intelligente. Utilizzando veicoli autonomi e connessi in reti VANET, Jamact può migliorare significativamente la gestione del traffico, trasformando un fenomeno negativo in una risorsa strategica per la mobilità urbana.

Il modello Jamact si integra, quindi, con le tecnologie avanzate delle Smart Cities

e dei veicoli autonomi, sfruttando le reti VANET per garantire una comunicazione e un coordinamento efficienti. Le reti VANET permettono ai veicoli di comunicare tra loro e con le infrastrutture stradali. Nel modello Jamact, i veicoli autonomi prioritari e non prioritari scambiano continuamente informazioni sulla loro posizione, velocità e condizioni del traffico. Questo flusso di dati consente al sistema di monitorare in tempo reale il traffico e gestire dinamicamente i jamiton in base alla posizione del veicolo prioritario e alla sua andatura, garantendo un rallentamento controllato dei veicoli non prioritari, e ricalcolando costantemente l'intensità e la durata del rallentamento in funzione delle condizioni del traffico e di eventuali variazioni nella velocità di percorrenza dei veicoli prioritari.

4.1 Le basi del modello

Prima di passare alla spiegazione del modello Jamact nel dettaglio, vengono presentati qui i quattro punti fondamentali del sistema. Questi elementi costituiscono i concetti centrali su cui si basa l'intera spiegazione teorica del modello. In particolare, i principi chiave di Jamact sono spiegati nella figura [fig. 4.1]:

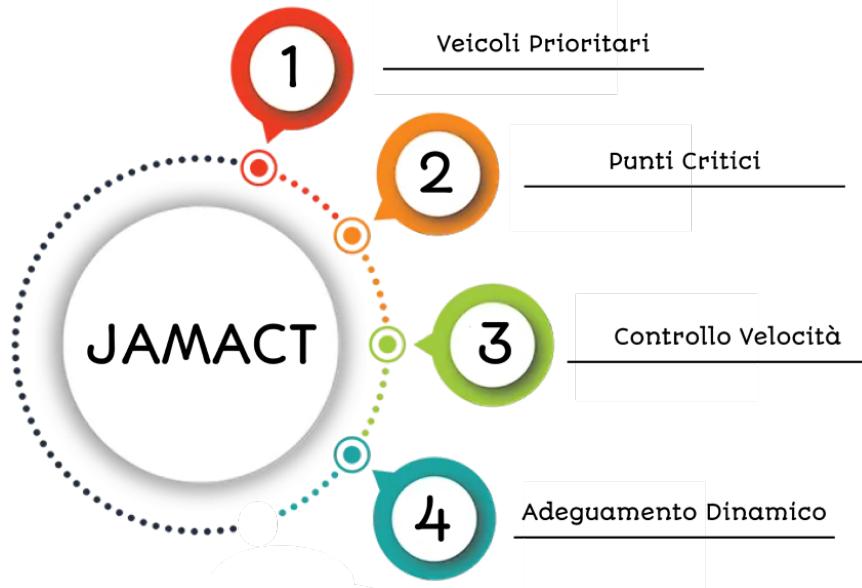


Figura 4.1: Elementi fondanti del sistema Jamact

Questi elementi sono interconnessi e lavorano insieme per creare un sistema di gestione del traffico che non solo riduce le congestioni per i veicoli prioritari, ma ne favorisce anche il transito senza impedimenti.

La priorità dei veicoli è uno degli aspetti essenziali del modello Jamact. Questo elemento identifica i veicoli con alta priorità, come veicoli di emergenza, mezzi pubblici o altre tipologie di veicoli di interesse. La loro identificazione permette di stabilire quali veicoli devono essere favoriti nel flusso del traffico.

Un altro elemento fondamentale è il rilevamento dei punti critici lungo il percorso, dove il transito di veicoli ad alta priorità potrebbe essere ostacolato da congestioni

esistenti. La rilevazione di questi punti è essenziale per poter applicare il modello Jamact in modo mirato e preciso. Sapere dove si trovano i punti critici permette di intervenire esattamente dove è necessario, evitando interventi inutili o inefficaci.

Il controllo della velocità è l'elemento che permette di indurre un jamiton e di far funzionare il modello. Questo elemento utilizza algoritmi di controllo della velocità per rallentare temporaneamente i veicoli di intralcio e creare uno spazio temporale in cui i veicoli prioritari possono transitare senza ostacoli. L'importanza di questo elemento risiede nella sua capacità di gestire il traffico in modo dinamico e adattivo, rispondendo in tempo reale alle condizioni del traffico. Questo controllo preciso della velocità consente di gestire il flusso veicolare in modo tale da ottimizzare il passaggio dei veicoli prioritari, minimizzando al contempo l'impatto sugli altri utenti della strada.

Infine, l'adeguamento dinamico è essenziale per garantire l'adattabilità del modello Jamact. Questo elemento sincronizza l'induzione del jamiton con il transito dei veicoli prioritari, adeguandone l'intensità in base ad eventuali variazioni di velocità di questi ultimi, e assicurando che il modello sia flessibile e possa adattarsi alle condizioni variabili del traffico. L'adeguamento dinamico è fondamentale anche per assicurare che i veicoli prioritari possano sempre transitare in modo efficiente, indipendentemente dalle variazioni nelle condizioni del traffico. Inoltre, questo elemento consente di regolare l'intensità del jamiton in funzione della velocità del veicolo prioritario, garantendo nel contempo che i veicoli non prioritari non subiscano un rallentamento eccessivo. Questo approccio mira a evitare l'arresto completo dei veicoli non prioritari, riducendo al minimo l'impatto complessivo del rallentamento. Nel caso in cui il veicolo prioritario subisca un rallentamento significativo, il sistema si riadatta di conseguenza, liberando alcuni veicoli dal rallentamento e aggiornando dinamicamente lo stato del jamiton indotto.

Variabili del Sistema

Un ultimo elemento da introdurre prima di analizzare teoricamente il funzionamento di Jamact, sono le variabili utilizzate nel sistema, le quali sono fondamentali per comprendere il processo di induzione dei jamiton [tab. 4.1]. Queste variabili rappresentano i parametri chiave che influenzano il comportamento del sistema.

Variabile	Spiegazione
Punto Critico (CP)	Un punto lungo il percorso in cui si possono verificare condizioni di traffico che possono causare rallentamenti o congestioni.
Punto di Attivazione jamiton (JAP)	Il punto nel percorso del veicolo prioritario da cui viene attivato il sistema di rallentamento dinamico.

Tabella 4.1 – Continua alla pagina successiva

Variabile	Spiegazione
Punto di Inizio jamiton (JSP)	Il punto esatto dove inizia l'area in cui viene attivato il jamiton per i veicoli non prioritari.
Aampiezza Punto Critico (CPW)	Distanza massima tra l'inizio del punto critico e la fine del punto critico.
Velocità minima di percorrenza	Rappresenta il limite di velocità minimo consentito per quel tipo di strada per garantire un flusso di traffico sicuro e scorrevole.
Limite di velocità	La velocità massima consentita per quel tipo di strada.
Velocità veicolo prioritario	La velocità del veicolo prioritario.
Velocità primo veicolo non prioritario	La velocità del primo dei veicoli che non hanno priorità nel sistema e che possono subire il jamiton.
Distanza CP - JAP	La distanza tra la fine del punto critico e il punto di attivazione del jamiton, che determina il momento in cui inizia il rallentamento dei veicoli non prioritari prima di raggiungere il punto critico.
Distanza CP - JSP	La distanza tra l'inizio del punto critico e il punto di inizio del jamiton, che determina il momento in cui i veicoli non prioritari devono rallentare prima di raggiungere il punto critico.

Tabella 4.1: Spiegazione delle variabili del modello

In Italia non è previsto un limite di velocità minima di percorrenza per tutte le tipologie di strade, ma la legge consente agli enti proprietari o gestori della strada di fissare una soglia sotto la quale i veicoli non possono circolare [26]. L'unica eccezione è rappresentata dalle autostrade a tre corsie, dove il limite minimo di percorrenza è fissato a 90km/h per la corsia di sinistra, 60km/h per la corsia di destra e invece non è previsto per la prima corsia [27].

Al fine di determinare un limite inferiore per il modello Jamact per tutte le tipologie di strade, necessario a scopo dimostrativo per il modello Jamact, verrà presa in considerazione una proporzione basata sul limite minimo e massimo di velocità in autostrada, in particolare verrà fatta una media dei limiti minimi di velocità presenti

per avere un valore unico di riferimento. Questo si traduce nel seguente calcolo:

$$\text{media_velocita_minima} = \frac{90 \text{ km/h} + 60 \text{ km/h} + 0 \text{ km/h}}{3} = 50 \text{ km/h}$$

Il rapporto tra questa velocità minima e il limite massimo di velocità è:

$$\text{Rapporto minimo/massimo autostrada} = \frac{50}{130} \approx 0,385$$

Moltiplicando questo valore per i limiti massimi di velocità di ogni strada si ottengono i limiti minimi proporzionali per il modello Jamact:

- Strade Extraurbane Principali (110km/h): $110 \text{ km/h} \times 0,385 \approx 42 \text{ km/h}$
- Strade Extraurbane Secondarie e Locali (90km/h): $90 \text{ km/h} \times 0,385 \approx 35 \text{ km/h}$
- Strade Urbane di Scorrimento (70km/h): $70 \text{ km/h} \times 0,385 \approx 27 \text{ km/h}$
- Strade Urbane di Quartiere (50km/h): $50 \text{ km/h} \times 0,385 \approx 19 \text{ km/h}$

Inoltre, viene fornita anche una descrizione dettagliata dei vari livelli di intensità del jamiton in base alla tipologia di strada [fig. 4.2]. Questi livelli di intensità rappresentano le diverse modalità di rallentamento applicate ai veicoli non prioritari in prossimità dei punti critici.

Classificazione	Limite di velocità	Velocità minima	High Jamiton	Medium Jamiton	Low Jamiton
Autostrada	 130	 50	50-77	78-103	104-130
Strade Extraurbane Principali (SS o SP)	 110	 42	42-65	66-87	88-110
Strade Extraurbane Secondarie e Locali	 90	 35	35-53	54-71	72-90
Strade Urbane di scorrimento	 70	 27	27-41	42-55	56-70
Strade urbane di quartiere	 50	 19	19-29	30-39	40-50

Figura 4.2: Classificazione strade

Per le principali tipologie di strade sono stati individuati tre range di intensità del jamiton:

1. **High jamiton:** rappresenta un livello di intensità elevato, ossia i veicoli non prioritari dovranno andare ad una velocità molto ridotta rispetto al limite previsto.

2. **Medium jamiton:** rappresenta un livello di intensità medio, ossia i veicoli non prioritari dovranno andare ad una velocità ridotta rispetto al limite previsto
3. **Low jamiton:** rappresenta un livello di intensità basso, ossia i veicoli non prioritari dovranno andare ad una velocità pari o quasi rispetto al limite previsto.

È importante notare che in fase dinamica i veicoli subiranno un rallentamento in base alla loro velocità di crociera, ma nel caso in cui quest'ultima consentirebbe anche di non rallentare il veicolo, allora il veicolo non subirebbe nessun rallentamento.

4.2 Modello Teorico

Dopo aver fornito una panoramica sul traffico fantasma, sulle sue caratteristiche e sugli elementi principali del sistema Jamact, si procede ora a descrivere il funzionamento del sistema a livello teorico. La figura [fig. 4.3] presenta un diagramma di flusso che sintetizza il processo operativo di Jamact.

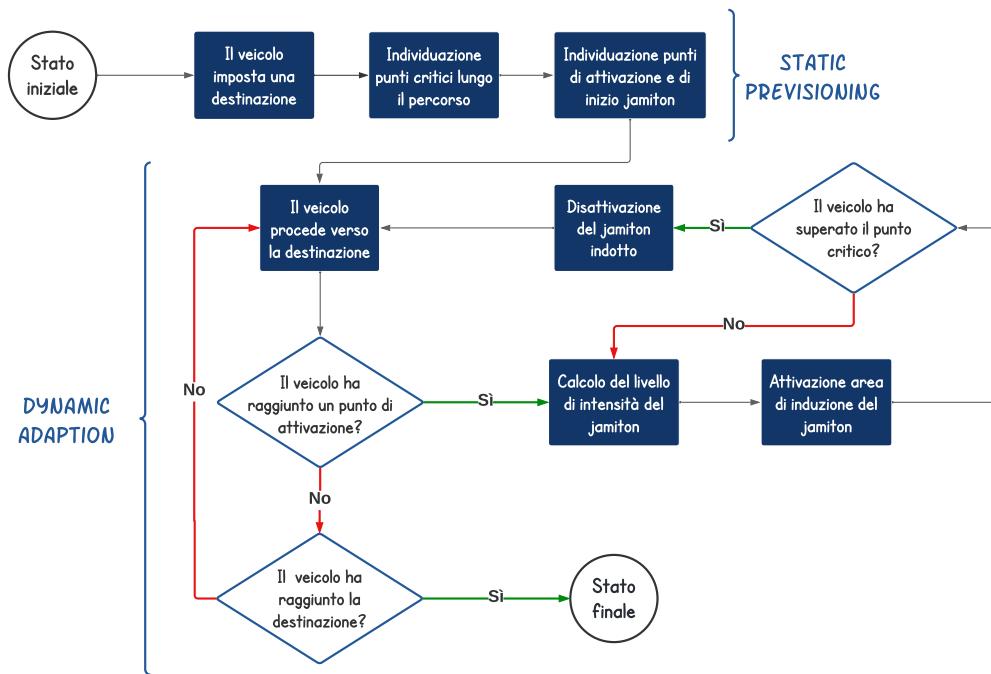


Figura 4.3: Diagramma di flusso – Spiegazione funzionamento Jamact

Come individuato dal diagramma di flusso, il modello Jamact si articola in due fasi principali di lavoro: la fase di "Static Previsioning" e la fase di "Dynamic Adaption".

Nella fase iniziale di "Static Previsioning", che inizia quando il veicolo prioritario imposta la destinazione, il sistema calcola una serie di punti chiave. In particolare:

- **Calcolo dei Punti Critici:** Il sistema identifica tutti i punti critici lungo il percorso previsto del veicolo prioritario, come incroci congestionati, zone di costruzione o altre aree ad alta densità di traffico.

- Individuazione dei Punti di Attivazione e Inizio jamiton: Vengono individuati i punti di attivazione e di inizio del jamiton, ossia il punto nel percorso dove viene attivato dai veicoli prioritari il sistema di rallentamento dinamico, e il punto dove inizia l'induzione del jamiton per i veicoli non prioritari.

La fase di "Static Previsioning" si concentra dunque sulla pianificazione iniziale del percorso del veicolo prioritario, preparando il terreno per la successiva fase di adattamento dinamico.

Una volta che il veicolo prioritario è in movimento, il sistema entra nella fase di "Dynamic Adaption". Durante questa fase, il sistema monitora costantemente l'avanzamento del veicolo prioritario lungo il percorso e controlla se si trova in prossimità di un punto critico dove è necessario attivare il jamiton.

- Monitoraggio dell'Avanzamento del Veicolo Prioritario: Il sistema tiene costantemente traccia della posizione e della velocità del veicolo prioritario lungo il percorso previsto.
- Attivazione del jamiton: Quando il veicolo prioritario si avvicina a un punto critico, e quindi quando supera un punto di attivazione, il sistema attiva il jamiton per rallentare i veicoli non prioritari.
- Ricalcolo dell'Intensità di Rallentamento: Una volta attivato il jamiton, il sistema ricalcola l'intensità di rallentamento che devono avere i veicoli al suo interno. Questo calcolo tiene conto della posizione del veicolo prioritario, della presenza di altri veicoli nel jamiton e delle condizioni attuali del traffico.
- Disattivazione del jamiton: Una volta superato il punto critico, il jamiton viene disattivato e il flusso veicolare torna alla normalità. Questo permette ai veicoli non prioritari di riprendere la normale velocità di transito.

Questo processo di adattamento dinamico si ripete continuamente mentre il veicolo prioritario avanza lungo il percorso, garantendo un transito efficiente e sicuro fino alla destinazione finale.

Static Previsioning

Durante la fase di "Static Previsioning", come detto prima, il sistema calcola preventivamente tutti i punti necessari al funzionamento ottimale. Una volta impostata la destinazione, il sistema identifica i punti critici lungo il percorso del veicolo prioritario, determinando per ciascun punto critico i relativi punti di attivazione e di inizio del jamiton.

Per calcolare il punto di inizio del jamiton, il sistema si basa sulle velocità massima e minima consentite per il veicolo su quel tipo di strada. Utilizzando queste informazioni, il sistema determina il tempo necessario per decelerare [A.3] dalla velocità massima alla velocità minima. Questo tempo di decelerazione è usato per calcolare la distanza

che il veicolo non prioritario deve percorrere per rallentare fino all'inizio del punto critico, definendo così l'ampiezza della zona di jamiton.

Basandoci ancora sul tempo di decelerazione, si determina anche la distanza che percorrerà il veicolo prioritario nel medesimo intervallo di tempo, mantenendo la velocità massima consentita sulla strada. In questo caso questa misura ci dirà quanto distante si trova il punto di attivazione dalla fine del punto critico.

È importante notare che la distanza percorsa dal veicolo prioritario parte dal punto di attivazione e si estende fino alla fine del punto critico. Invece, l'inizio della zona di jamiton, ossia la zona dentro la quale i veicoli non prioritari sono rallentati, ha una distanza calcolata fino all'inizio del punto critico. Questa differenza è dovuta al fatto che i veicoli non prioritari devono essere rallentati finché il veicolo prioritario non ha completamente superato il punto critico.

Una volta individuati questi due punti, si può essere certi che i veicoli non prioritari raggiungeranno il punto critico dopo il veicolo prioritario. Questo perché entrambi i tipi di veicoli viaggeranno per lo stesso intervallo di tempo a velocità diverse.

Per calcolare i tempi di decelerazione, è stata utilizzata la formula [A.4] del moto uniformemente accelerato [A.3], considerando un'accelerazione di $1m/s^2$. Questo valore è stato estrapolato da una ricerca condotta dal Politecnico di Milano [28]. Lo studio ha identificato lo stile di guida ottimale, determinando che l'accelerazione media di $1m/s^2$ è l'accelerazione ideale per minimizzare l'impatto ambientale delle emissioni di CO_2 ¹.

Per calcolare il punto di inizio jamiton, basandoci sul tempo di decelerazione, si utilizza la formula della seconda equazione del moto uniformemente accelerato [A.2]. Invece, per calcolare il punto di attivazione si utilizza la formula inversa della velocità, calcolando quanto spazio viene coperto andando per un certo tempo ad una certa velocità.

I calcoli effettuati durante la fase statica considerano lo scenario più sfavorevole possibile, utilizzando le velocità massime consentite per entrambi i tipi di veicoli e applicando il livello di jamiton più elevato (high), con una velocità all'interno del jamiton impostata al valore minimo consentito. Questo approccio garantisce che il sistema sia progettato per gestire le condizioni più critiche.

Dynamic Adaption

In questa fase, il sistema monitora dinamicamente l'avanzamento del veicolo prioritario lungo il percorso. Qualora si verifichino ostacoli, rallentamenti o variazioni rispetto allo studio statico effettuato prima della partenza del veicolo prioritario, il sistema si adatta di conseguenza. Questo adattamento comporta la modifica dei valori di intensità del jamiton, dei punti di attivazione e dei punti di inizio jamiton, in modo da rispondere efficacemente ai cambiamenti osservati durante il viaggio del veicolo prioritario.

In particolare, il punto di attivazione, una volta superato, avanza con la velocità del veicolo prioritario. Tuttavia, se quest'ultimo rallenta, il sistema ricalcola la velocità a cui le auto all'interno del jamiton devono procedere. Nel caso in cui si raggiunga già

¹Nelle pagine successive si terrà conto di questo valore per tutti i calcoli che si andranno a fare.

la velocità minima consentita e non sia possibile ulteriori riduzioni, il jamiton rilascia alcuni veicoli e si riadatta in base alla nuova posizione del veicolo prioritario.

Per quanto riguarda l'intensità assegnata al jamiton [fig. 4.2], questa varierà durante la fase dinamica in base alla velocità del veicolo prioritario. Nella fase statica, come precedentemente menzionato, l'intensità assegnata è quella massima (high). Durante la fase dinamica, l'intensità del jamiton varierà in base al modo in cui il veicolo prioritario percorre il tragitto dal punto di attivazione al punto critico. Se il veicolo procede più velocemente di quanto previsto, l'intensità del jamiton può essere ridotta, consentendo quindi una maggiore velocità di percorrenza nello stesso, viceversa se il veicolo procede più lentamente l'intensità del jamiton aumenta. Inoltre, il punto di inizio jamiton avanza con l'arrivo dei veicoli all'interno del jamiton. Se il jamiton si attiva quando non sono presenti veicoli al suo interno ma arrivano successivamente, i veicoli successivi subiranno un jamiton ridotto rispetto a quello previsto sempre in base al veicolo prioritario. Il sistema garantisce che un veicolo non prioritario non raggiunga il punto critico prima che il veicolo prioritario lo abbia superato, evitando così che sia costretto a fermarsi per consentire il passaggio di quest'ultimo. Quindi, il sistema assicura che il veicolo non prioritario rallenti progressivamente, mantenendo sempre il suo movimento, e arrivi al punto critico solo dopo il passaggio del veicolo prioritario.

Calcolo statico dei punti del modello

L'analisi del modello teorico si conclude presentando il calcolo dei punti di attivazione e inizio jamiton che staticamente saranno uguali per ogni strada a parità di limiti di velocità. Nella figura [fig. 4.4] per ciascuna categoria di strada sono stati determinati i punti di attivazione e di inizio del jamiton, insieme al tempo di decelerazione associato.

Classificazione	Limite di velocità	Velocità minima	Punto di attivazione jamiton	Punto di inizio jamiton	Tempo decelerazione
	130	50	794 metri	552 metri	22 secondi
	110	42	581 metri	400 metri	19 secondi
	90	35	375 metri	263 metri	15 secondi
	70	27	233 metri	161 metri	12 secondi
	50	19	125 metri	85 metri	9 secondi

Figura 4.4: Calcolo punti statici

4.3 Criticità del sistema e possibili soluzioni

Dall'analisi del modello Jamact e del suo funzionamento, e attraverso la simulazione di vari scenari reali, sono state individuate una serie di criticità potenziali del sistema. In questa sezione, vengono presentati tali scenari problematici insieme a possibili soluzioni per affrontarli. In particolare, sono stati identificati i seguenti scenari, illustrati anche nelle figure [fig. 4.5] che possono creare problemi al funzionamento di Jamact:

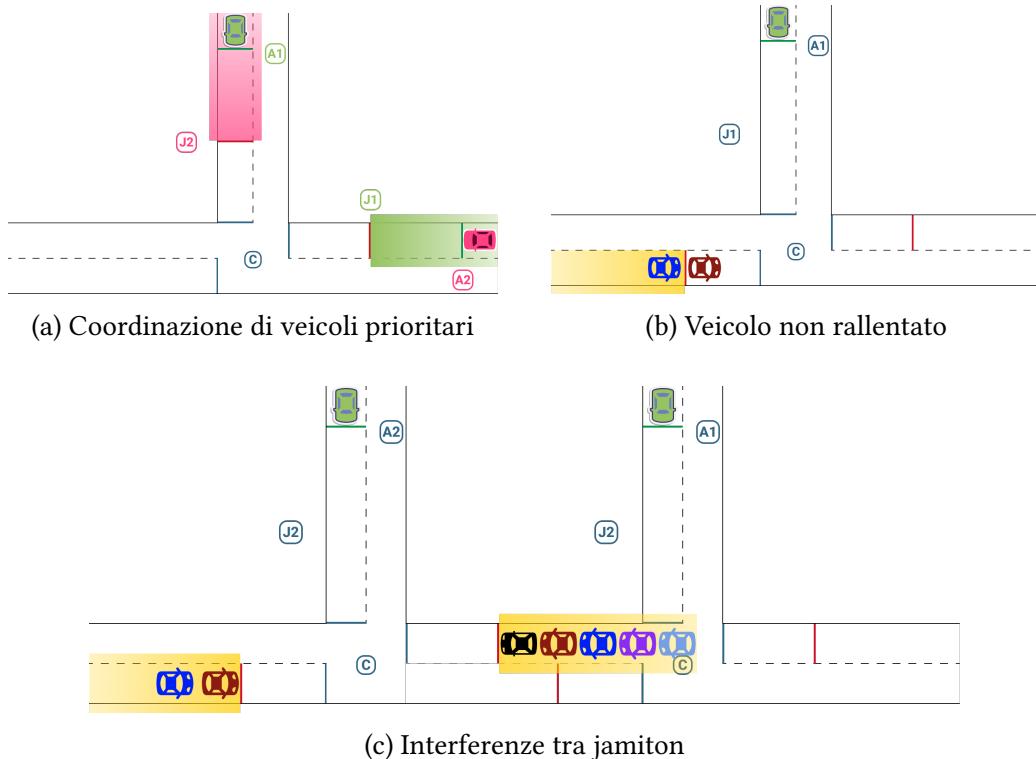


Figura 4.5: Casi particolari

- Coordinazione tra veicolo prioritario e veicoli non prioritari [fig. 4.5a]: Questa criticità si verifica quando un veicolo non prioritario supera la linea di inizio jamiton poco prima che il sistema venga attivato. In questo caso, il veicolo non prioritario potrebbe non rallentare in tempo, interferendo con il transito del veicolo prioritario. Una possibile soluzione è l'implementazione di un sistema di avviso anticipato per i veicoli non prioritari, che segnali l'imminente attivazione del jamiton.
- Coordinazione di più veicoli prioritari [fig. 4.5b]: Questo problema emerge quando più veicoli prioritari devono transitare contemporaneamente attraverso lo stesso punto critico. La gestione simultanea di più veicoli prioritari può complicare l'efficacia del jamiton. Una soluzione potrebbe essere l'implementazione di un sistema di gestione delle priorità, che assegna una sequenza di transito ai

veicoli prioritari in base alla loro urgenza o importanza, e in base a questa si decide anche l'ordine dei jamiton da attivare.

- Coordinazione di più incroci consecutivi [fig. 4.5c]: Questa criticità si presenta quando una coda di jamiton si estende e interferisce con altri punti critici, influenzando così il transito di altri veicoli prioritari. Per mitigare questo problema, è possibile utilizzare un sistema di monitoraggio avanzato che regoli dinamicamente il transito dei veicoli non prioritari in funzione del flusso del traffico e dei jamiton attivi, assicurando che le code non interferiscano con altri punti critici.

4.3.1 Approcci alla coordinazione tra veicolo prioritario e veicoli non prioritari

Uno scenario critico nella fase dinamica del modello Jamact si verifica quando un veicolo non prioritario supera la linea di attivazione del jamiton poco prima che il veicolo prioritario attivi il sistema. Questo può generare veicoli che non sono più sotto il controllo del sistema.

Un'altra situazione problematica potrebbe sorgere se la velocità di un veicolo non prioritario è così alta da non poter essere rallentato, poiché una riduzione di velocità lo porterebbe sotto il limite minimo consentito. Tuttavia, tale veicolo potrebbe comunque risultare di intralcio trovandosi all'interno del punto critico quando il veicolo prioritario vi arriva.

Per risolvere queste due problematiche, è stata ideata una possibile soluzione comune. In particolare sarebbe possibile introdurre una linea di pre-allerta situata a una distanza pari all'ampiezza del punto critico sia dal punto di attivazione che dal punto di inizio jamiton. La funzione principale della linea di pre-allerta² è quella di effettuare diversi calcoli sui veicoli che la attraversano, ottimizzando la gestione del traffico non prioritario sia all'arrivo alla linea di inizio jamiton, sia nella zona del jamiton stesso. In particolare, i calcoli includono:

- **Valutazione della velocità:** Se il sistema prevede che il veicolo non prioritario supererà la linea di inizio jamiton prima che il sistema venga attivato, vengono preventivati degli interventi per evitare conflitti col veicolo prioritario, modificando la velocità del veicolo non prioritario.
- **Induzione del preemptive jamiton al punto di pre-allerta:** nel caso in cui il veicolo non prioritario arrivi alla linea di inizio jamiton prima dell'attivazione di quest'ultima e la sua velocità attuale non sia sufficiente per superare il punto critico prima del veicolo prioritario, il sistema induce un jamiton preventivo (preemptive jamiton) per ridurre la velocità del veicolo in maniera adeguata.

²La distanza del punto di pre-allerta dal punto di attivazione e dal punto di inizio jamiton è pari all'ampiezza del punto critico per permettere al sistema di determinare se un veicolo non prioritario ha una velocità sufficiente per superare il punto critico prima che il veicolo prioritario vi arrivi, garantendo un controllo efficace e tempestivo del traffico.

- **Induzione del preemptive jamiton tra punto di pre-allerta e punto di inizio jamiton:** se il sistema rileva che il veicolo non prioritario arriverà alla linea di inizio jamiton con una velocità troppo elevata per essere efficacemente rallentato ma comunque non abbastanza veloce da superare il punto critico prima dell'arrivo del veicolo prioritario, viene applicato un ulteriore rallentamento nel tratto tra la linea di pre-allerta e la linea di inizio jamiton, al fine di farlo arrivare alla linea di inizio jamiton con una velocità adatta.

Questo approccio garantisce che il veicolo non prioritario raggiunga la linea di inizio jamiton con una velocità appropriata per essere intercettato e subire l'azione del jamiton.

4.3.2 Approcci alla coordinazione di più veicoli prioritari

In una situazione dinamica, potrebbe sorgere un'ulteriore problema nel funzionamento del sistema, in particolare quando due veicoli prioritari devono attraversare contemporaneamente lo stesso punto critico. Per gestire questo scenario, è necessario assegnare una priorità aggiuntiva ai veicoli prioritari. Questo permetterà di stabilire un ordine tra i veicoli prioritari, indicando quale veicolo deve essere gestito per primo. In questo modo, sarà chiaro quale veicolo è autorizzato a creare un "jamiton" sugli altri, e quale veicolo deve prima subire il "jamiton" dall'altro veicolo prioritario prima di poter prendere il controllo e attivare il proprio "jamiton".

La gestione del problema potrebbe avvenire prendendo in considerazione uno tra le seguenti possibili categorie di priorità individuate: priorità per posizione, per tempo/distanza dall'arrivo, per gerarchia e per segnaletica.

1. **Position-Based Prioritization:** In questo approccio, la precedenza viene assegnata al veicolo che si è inserito per primo nel sistema. Per gestire questa priorità, si utilizza una struttura dati chiamata Coda (*Queue*). La coda è una struttura dati FIFO (First In, First Out), che garantisce che i veicoli vengano trattati nell'ordine in cui arrivano. La coda è ideale per questo scenario perché assicura che il veicolo che arriva per primo al punto critico sia anche il primo a passare.



Figura 4.6: Esempio di una struttura dati Coda.

2. **Remaining Travel Time/Distance-Based Prioritization:** In questo approccio, i veicoli vengono ordinati in base a chi è più vicino, in termini di tempo o distanza, alla destinazione. Per gestire questa priorità, si utilizza anche in questo

caso una coda di priorità (*Priority Queue*), dove ogni elemento viene estratto in base alla sua priorità e non necessariamente nell'ordine in cui sono stati inseriti; in questo caso quindi, i veicoli con tempo o distanza dalla destinazione più piccoli saranno estratti prima.

3. **Hierarchy-Based Prioritization:** In questo approccio, i veicoli vengono categorizzati in base a diversi livelli di priorità. La decisione su quale veicolo deve cedere il passo all'altro si basa su questa gerarchia, stabilita da criteri specifici quali l'urgenza del servizio svolto o altri fattori rilevanti. Una struttura dati adatta a gestire questi comportamenti è una coda di priorità (*Priority Queue*), dove ogni elemento viene estratto in base alla sua priorità e non necessariamente nell'ordine in cui sono stati inseriti; in questo caso quindi, i veicoli con priorità maggiore saranno estratti prima.

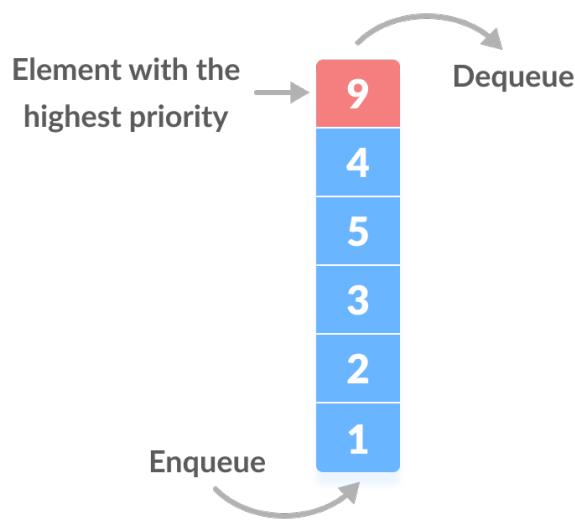


Figura 4.7: Esempio di una Coda di priorità con valori interi.

4. **Sign-Based Prioritization:** Questo approccio si basa sull'osservanza delle segnaletiche stradali di precedenza. Il veicolo prioritario che ha la precedenza secondo il codice della strada sarà il primo ad attraversare il punto critico. Questo tipo di priorità può essere utilizzata in maniera esclusiva (magari in casi in cui non è necessario utilizzare i precedenti metodi), oppure può essere utilizzata in combinazione con un altro metodo e fungere da supporto in caso di errori.

Queste tipologie di priorità possono essere implementate in modo esclusivo, specialmente in situazioni in cui non è richiesto l'utilizzo di altri metodi. Tuttavia, è altrettanto possibile integrarle con altri metodi per rendere il sistema ancora più adattabile alle varie condizioni che possono presentarsi. La decisione sull'approccio da adottare può dipendere dal contesto del traffico, dalle esigenze operative e dalle priorità stabilite.

4.3.3 Approcci alla gestione di interferenze tra più jamiton

Un ulteriore scenario critico che il sistema potrebbe affrontare riguarda la gestione simultanea di diversi jamiton attivati da molteplici veicoli prioritari. È essenziale garantire che i jamiton non interferiscano tra loro, mantenendo la sicurezza e l'affidabilità del sistema. Per questo, il sistema deve assicurarsi che i veicoli attraversino gli incroci solo quando possono liberarli immediatamente. Se il traffico sulla strada in cui devono entrare è congestionato, i veicoli devono essere fermati prima di occupare il punto critico, aspettando che la strada si liberi.

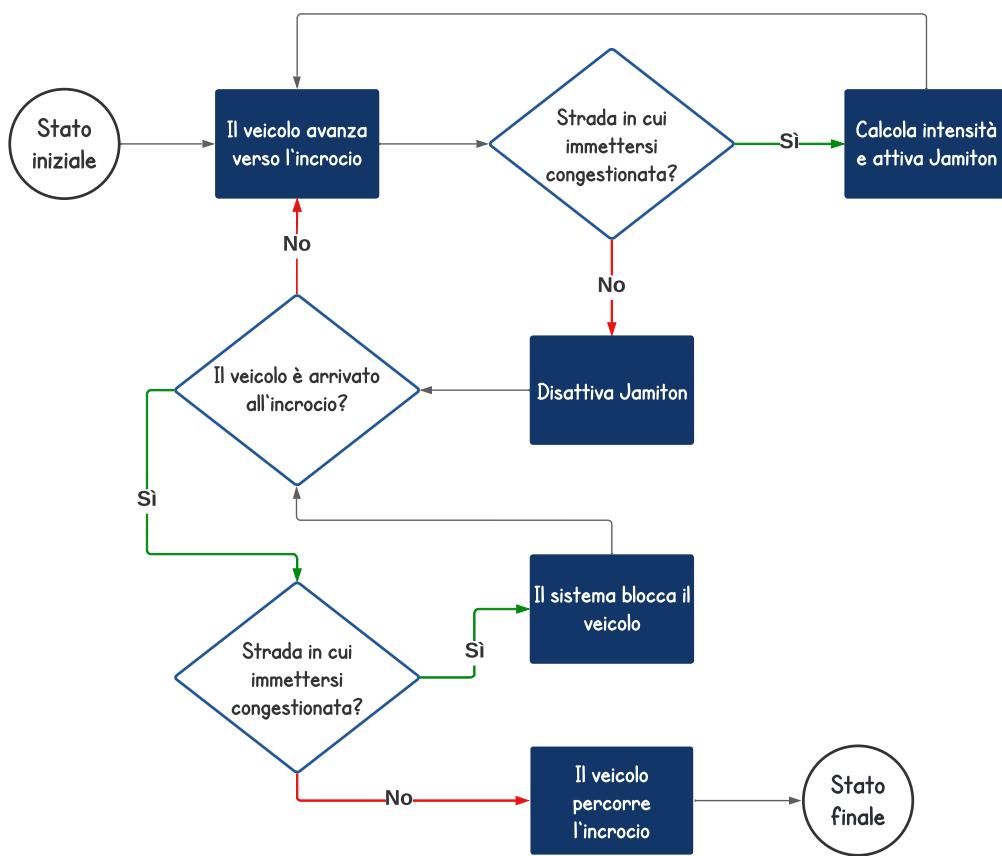


Figura 4.8: Diagramma di flusso – Gestione occupazione incrocio

Per garantire sicurezza, affidabilità e minimizzare le interferenze tra i jamiton dei veicoli prioritari, il sistema deve mantenere costantemente liberi gli incroci, impedendo che vengano congestionati. A tal fine, le strade, soprattutto le sezioni che conducono agli incroci, saranno dotate di sensori in grado di monitorare il livello di traffico. Quando un veicolo si avvicina a un incrocio, il sistema valuta la direzione e il percorso del veicolo per determinare se la strada in cui è diretto è congestionata. Se la strada è congestionata, il sistema rallenterà il veicolo fino a che non si liberi la strada in cui deve immettersi. Nel caso in cui il veicolo non può più essere rallentato

ma non può neanche superare il punto critico, verrà bloccato prima del punto critico, evitando che lo occupi, fino a quando la strada non si libera, garantendo così che non si creino ingorghi.

Per capire meglio come dovrebbe funzionare questa regola, viene introdotto un diagramma di flusso esplicativo [fig. 4.8]. In particolare, si vede come il sistema controlla se la strada in cui deve immettersi il veicolo è libera o meno man mano che avanza verso il punto critico. Se la strada è congestionata, al veicolo viene applicato un jamiton; viceversa, il veicolo può riprendere la sua velocità di percorrenza e viaggiare verso l'incrocio. Una volta arrivato al punto critico, può percorrerlo solo se la strada in cui deve immettersi non è congestionata, altrimenti viene bloccato.

In sintesi, il sistema Jamact utilizza sensori di traffico e algoritmi di controllo per garantire che i veicoli prioritari possano attraversare gli incroci in modo sicuro e senza ostacoli, evitando che questi punti critici diventino congestionati. Questo approccio permette di coordinare efficacemente numerosi jamiton, assicurando che i veicoli prioritari possano transitare in modo fluido e senza interruzioni, minimizzando l'impatto sul traffico generale.

Per simulare il comportamento spiegato nel diagramma di flusso [fig. 4.8], viene presentato un codice Prolog che fornisce una simulazione del comportamento dei veicoli in prossimità degli incroci, utilizzando regole per gestire il traffico in base alla congestione delle strade. La logica implementata permette di rallentare, bloccare o permettere il transito dei veicoli in funzione dello stato della strada. Di seguito una descrizione dettagliata del funzionamento e una versione ottimizzata del codice:

Le regole `is_congested` e `is_clear` controllano lo stato delle strade, rispettivamente se la strada è congestionata e se la strada è libera:

```
is_congested(Road) :-  
    congested(Road).  
  
is_clear(Road) :-  
    clear(Road).
```

La regola `update_vehicle/1` aggiorna la posizione del veicolo man mano che si avvicina all'incrocio:

```
update_vehicle(Vehicle) :-  
    vehicle(Vehicle, Distance, Road),  
    Distance > 0,  
    NewDistance is Distance - 10,  
    retract(vehicle(Vehicle, Distance, Road)),  
    assert(vehicle(Vehicle, NewDistance, Road)).
```

Le regole `slow_down`, `block_vehicle` e `allow_transit` gestiscono il comportamento dei veicoli in base allo stato delle strade. In particolare la prima verifica

se il veicolo si trova su una strada congestionata e, in tal caso, rallenta il veicolo decrementando la distanza di 10 unità, aggiornando le informazioni nel database. La regola `block_vehicle` si attiva quando un veicolo raggiunge l'incrocio (distanza pari a 0) e la strada risulta ancora congestionata, bloccando così il veicolo. Infine, la regola `allow_transit` permette al veicolo di transitare attraverso l'incrocio se la strada è libera, confermando che non ci sono congestioni che impediscono il passaggio.

```

slow_down(Vehicle) :-  

    vehicle(Vehicle, Distance, Road),  

    is_congested(Road),  

    NewDistance is Distance - 10,  

    retract(vehicle(Vehicle, Distance, Road)),  

    assert(vehicle(Vehicle, NewDistance, Road)).  
  

block_vehicle(Vehicle) :-  

    vehicle(Vehicle, 0, Road),  

    is_congested(Road),  
  

allow_transit(Vehicle) :-  

    vehicle(Vehicle, 0, Road),  

    is_clear(Road),
  
```

Infine la regola `manage_traffic` gestisce il flusso di tutti i veicoli, controllando lo stato delle strade e aggiornando le loro posizioni di conseguenza:

```

manage_traffic :-  

    forall(vehicle(Vehicle, Distance, Road),  

        ( Distance > 0 ->  

            ( is_congested(Road) -> slow_down(Vehicle)  

                ; update_vehicle(Vehicle)  

            )  

        ; (is_congested(Road) -> block_vehicle(Vehicle)  

            ; allow_transit(Vehicle)  

        )  

    )
).
  
```

Questa regola utilizza la funzione `forall` per iterare su tutti i veicoli presenti nel database e applica azioni diverse in base alla distanza del veicolo dall'incrocio e allo stato della strada. Se la distanza del veicolo è maggiore di zero, viene verificato lo stato della strada: se la strada è congestionata, il veicolo viene rallentato con la regola `slow_down`, altrimenti la posizione del veicolo viene aggiornata con `update_vehicle`. Se invece la distanza del veicolo è pari a zero, viene controllato

nuovamente lo stato della strada: se la strada è ancora congestionata, il veicolo viene bloccato con `block_vehicle`; se la strada è libera, il veicolo può transitare utilizzando `allow_transit`.

4.4 Esempio Teorico

Per chiarire ulteriormente il modello Jamact e il suo funzionamento, si presenta un esempio teorico che chiarisce l'idea su cui si basa il modello.

Si immagini di trovarsi su una strada urbana di scorrimento, dove il limite di velocità è di 70km/h e la velocità minima consentita nel nostro esempio è di 27km/h. Come già spiegato, durante la fase statica del calcolo verranno determinati i valori basati sul caso peggiore possibile. Tenendo conto di ciò, prendendo in esame un tipico punto critico e considerando un'accelerazione di 1 m/s², si va a calcolare il punto di attivazione del jamiton e il punto di inizio jamiton³.

Per prima cosa, si calcola il tempo necessario per decelerare da 70km/h a 27km/h usando la formula del moto uniformemente accelerato [eq. (A.4)]. Le velocità vengono convertite da km/h a m/s:

$$v_0 = 70 \text{ km/h} \times \frac{1000 \text{ m}}{3600 \text{ s}} = 19 \text{ m/s}$$

$$v_f = 27 \text{ km/h} \times \frac{1000 \text{ m}}{3600 \text{ s}} = 7.5 \text{ m/s}$$

Per calcolare il tempo si usa la formula $t = \frac{v_0 - v_f}{a}$, e sostituendo i valori si ottiene

$$t = \frac{19.44 \text{ m/s} - 7.5 \text{ m/s}}{1 \text{ m/s}^2} = \frac{11.5 \text{ m/s}}{1 \text{ m/s}^2} = \approx 12 \text{ secondi}$$

Successivamente, viene calcolata la distanza percorsa durante il tempo di decelerazione usando la seconda equazione del moto uniformemente accelerato:

$$s = v_0 t + \frac{1}{2} a t^2$$

Sostituendo i valori:

$$s = 19.44 \text{ m/s} \times 12 \text{ s} + \frac{1}{2} \times (-1 \text{ m/s}^2) \times (12 \text{ s})^2$$

$$s = 228 \text{ m} - 72 \text{ m} = \approx 161 \text{ m}$$

Quindi, il tempo necessario per decelerare da 70km/h a 27km/h è di circa 12 secondi, e la distanza percorsa durante questa decelerazione è di circa 161 metri.

Infine, viene calcolato il punto di attivazione jamiton calcolando quanto spazio compre il veicolo prioritario andando a 70km/h per il tempo di decelerazione, ossia

³Questi punti sono già stati calcolati nella tabella della figura [fig. 4.4], ma qui si andrà a dare una spiegazione completa del funzionamento del sistema

12 secondi. Per calcolare lo spazio percorso viaggiando a 70km/h per 12 secondi, si usa la formula della distanza percorsa a velocità costante:

$$s = v \times t$$

Si converte la velocità da km/h a m/s:

$$v = 70 \text{ km/h} \times \frac{1000 \text{ m}}{3600 \text{ s}} = 19.44 \text{ m/s}$$

Si usa la formula con $v = 19.44 \text{ m/s}$ e $t = 12 \text{ s}$:

$$s = 19.44 \text{ m/s} \times 12 \text{ s} = 233 \text{ m}$$

Quindi, lo spazio percorso viaggiando a 70km/h per 12 secondi è di 233 metri.

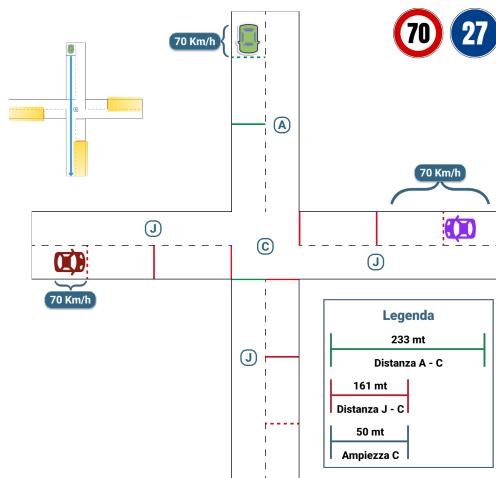


Figura 4.9: Static Previsioning

Individuati questi punti, ben rappresentati in figura [fig. 4.9], si conclude la fase statica del modello e si passa a quella dinamica. Quest'ultima comporta il calcolo dinamico dei punti precedentemente discussi e il loro adattamento in tempo reale in base alle variazioni di andatura del veicolo prioritario. Prima di vedere cosa succede in fase dinamica, è importante sottolineare che la zona di jamiton non ha un inizio ed una fine, ma solo un inizio poiché in un caso reale il rallentamento del primo veicolo influenzereà anche i veicoli che lo precedono e quindi un numero variabile. Si aggiunge anche che se non ci sono veicoli sulla strada allora l'intensità del jamiton potrebbe diminuire. Questo perché se il veicolo prioritario ha attivato il jamiton ma in quell'istante non ci sono veicoli su cui attivarlo, allora il punto di attivazione avanza con l'avanzare del veicolo prioritario e così anche il punto di inizio jamiton potrebbe variare sia di intensità che di altezza sulla strada.

Si passa ora alla fase dinamica, nella quale per rendere il caso più realistico, sono stati leggermente variati i valori di velocità dei veicoli rispetto alle previsioni fatte in fase statica. Come si può vedere dalla figura [fig. 4.10], il veicolo prioritario raggiunge il punto di attivazione con una velocità di 65km/h, inferiore rispetto a quella considerata in fase statica. Analogamente, i veicoli non prioritari arrivano

con una velocità di 60km/h e di 50km/h. A seguito della riduzione della velocità del veicolo prioritario, il sistema ricalcola l'intensità del jamiton, mantenendo inalterate le distanze dal punto critico per i punti di attivazione e di inizio jamiton.

In figura [fig. 4.11] si può osservare l'attivazione del jamiton quando il veicolo prioritario raggiunge il punto di attivazione. Da quel momento, il punto di inizio jamiton (indicato dalla linea rossa) inizia ad avanzare e a rallentare i veicoli fino a raggiungere il punto critico (o incrocio). Durante questo processo, il jamiton e la sua intensità possono adattarsi alle variazioni di velocità del veicolo prioritario.

Nel nostro caso, considerando i cambiamenti dinamici spiegati in figura [fig. 4.10], viene ricalcolata l'intensità del jamiton per tutti i veicoli non prioritari. In particolare, in figura [fig. 4.11] si nota come i tre punti di jamiton (J_1, J_2, J_3) subiscano un'intensità differente in quanto arrivano al punto di inizio jamiton con velocità differenti. Questo comporta anche un differente avanzamento verso il punto critico con tempi diversi. Si nota anche come il J_3 in figura [fig. 4.12] non avanza in quanto non sono presenti auto al punto di inizio jamiton.

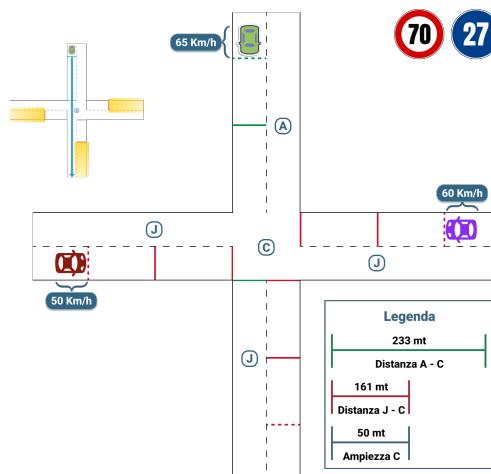


Figura 4.10: Dynamic Adaption

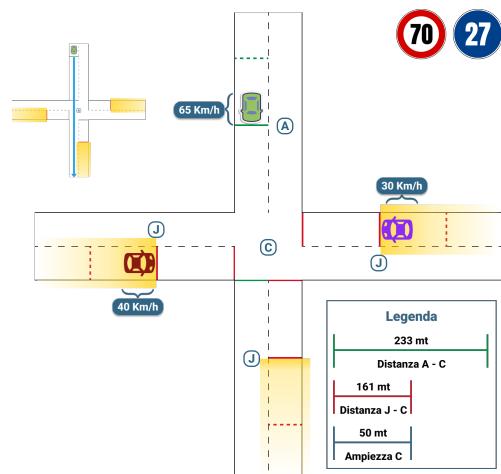


Figura 4.11: Attivazione Jamact

Nel dettaglio, tenendo conto che i veicoli non prioritari arrivano al punto di inizio jamiton con una velocità di 60km/h e 50km/h e che il veicolo prioritario viaggia ad una velocità di 65km/h, si ricalcola l'intensità del jamiton.

Per prima cosa si trova il tempo che il veicolo prioritario impiegherà per percorrere i 233 metri che lo separano dalla fine del punto critico:

$$\text{tempo} = \frac{233 \text{ m}}{(65/3.6) \text{ m/s}} = 13 \text{ s}$$

Il sistema, conoscendo la distanza percorsa e il tempo impiegato, effettua il ricalcolo trovando la velocità media e convertendola in km/h in questo modo:

$$v_{media} = \frac{s}{t} = \frac{161 \text{ m}}{13 \text{ s}} = 12.38 \text{ m/s}$$

$$v_{media} = 12.38 \times 3.6 \text{ km/h} = 45 \text{ km/h}$$

Avendo trovato la velocità media, si può utilizzare questa informazione per calcolare la velocità finale nel caso di una decelerazione uniforme nel seguente modo:

$$v_{media} = \frac{v_{iniziale} + v_{finale}}{2}$$

$$v_{finale} = 2 \times v_{media} - v_{iniziale} = 2 \times 45 \text{ km/h} - 60 \text{ km/h} = 30 \text{ km/h}$$

$$v_{finale} = 2 \times v_{media} - v_{iniziale} = 2 \times 45 \text{ km/h} - 50 \text{ km/h} = 40 \text{ km/h}$$

Quindi, per percorrere 233 metri in 13 secondi, è necessario decelerare da 60km/h a circa 30km/h. Analogamente, per percorrere 233 metri in 13 secondi partendo da una velocità iniziale di 50km/h, bisognerebbe decelerare a circa 40km/h.

I veicoli non prioritari, quindi, viaggeranno verso punto critico con una velocità ridotta e ricalcolata dinamicamente in base ai dati nuovi a disposizione in questa fase. Ciò consente di regolare l'intensità del jamiton a seconda di tutte le situazioni che possono occorrere, garantendo flessibilità e dinamicità al sistema.

Una volta che il veicolo prioritario ha superato punto critico, come in figura [fig. 4.13], il jamiton si disattiva immediatamente permettendo a tutti i veicoli di riprendere la loro andatura.

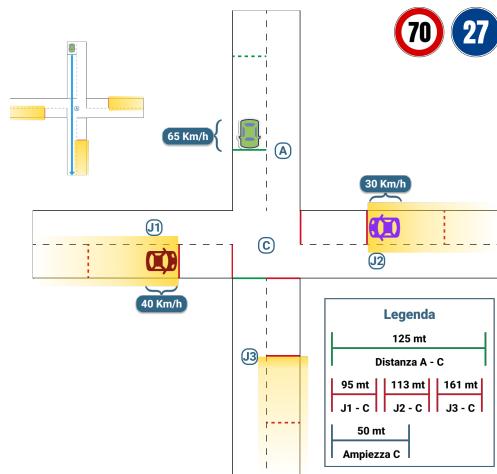


Figura 4.12: Avanzamento veicoli

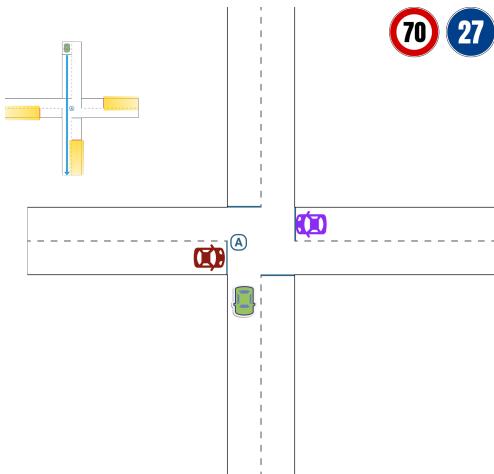


Figura 4.13: Disattivazione Jamact

Tipologie di percorsi

La trattazione sull'esempio di applicazione del modello Jamact si conclude con l'analisi dei due scenari possibili che possono verificarsi in un punto critico. In base alla presenza di veicoli che potrebbero interferire con il percorso del veicolo prioritario, può essere attivato il jamiton lungo quella specifica strada; altrimenti, il jamiton rimane inattivo come evidenziato nell'illustrazione [fig. 4.15].

Nel primo scenario, illustrato in figura [fig. 4.14], si nota come il percorso del veicolo prioritario, e la sua occupazione del punto critico, richiede di attivare il jamiton lungo tutte le strade che confluiscono nel punto critico per consentire il passaggio del

veicolo prioritario senza impiedimenti. Nel secondo scenario, delineato nell'immagine [fig. 4.15], il jamiton non viene attivato lungo tutte le strade. Ciò avviene quando, in base al tragitto del veicolo prioritario e al modo in cui deve occupare il punto critico, non tutti i veicoli presenti sulle strade che convergono nel punto critico costituiscono un ostacolo per il veicolo prioritario. In questa circostanza, il veicolo prioritario attiverà un numero limitato di jamiton, garantendo così che il suo passaggio influisca sul minor numero possibile di veicoli.

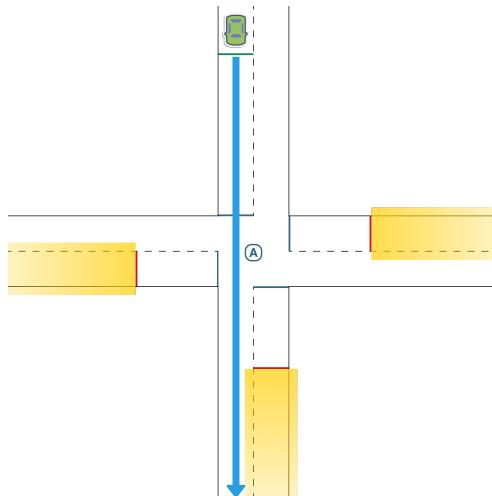


Figura 4.14: Esempio percorso 1

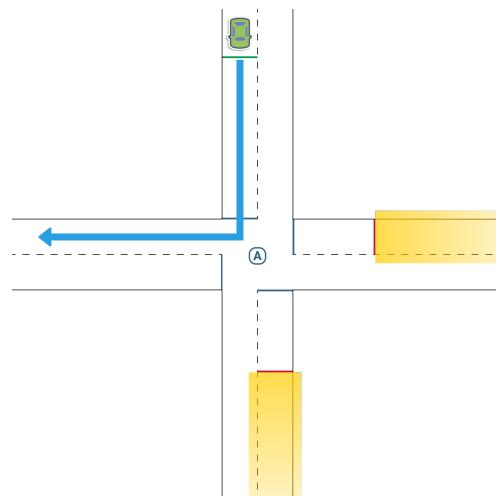


Figura 4.15: Esempio percorso 2

Questi due scenari dimostrano la flessibilità e l'adattabilità del modello Jamact nel gestire situazioni di traffico dinamiche, ottimizzando il flusso dei veicoli prioritari e minimizzando l'impatto dell'induzione di rallentamenti sui veicoli non prioritari.

4.5 Jamact e Safety Car: un isomorfismo nella gestione veicolare

Per comprendere meglio il funzionamento della zona di jamiton e del suo punto di inizio, si presenta qui un parallelo con la Formula 1 introducendo un isomorfismo con la Safety Car. In altre parole, nelle prossime righe verrà analizzato come la zona di jamiton si comporta analogamente alla Safety Car nella Formula 1.

La Safety Car è un elemento cruciale nelle gare di Formula 1 e viene impiegata in determinate situazioni per garantire la sicurezza dei piloti e degli ufficiali di gara. Nel dettaglio, la Safety Car viene attivata in risposta a situazioni di pericolo che rendono necessaria la sospensione temporanea della gara. Queste situazioni includono incidenti gravi, presenza di detriti sulla pista, condizioni meteorologiche estreme o altri eventi imprevisti che mettono a rischio la sicurezza dei partecipanti.

Quando si verifica una situazione di pericolo, i commissari di gara decidono se è necessario inviare la Safety Car in pista. Se la decisione viene presa, la Safety Car viene annunciata ai team e ai piloti attraverso segnali visivi e radio. La Safety Car

entra in pista e si posiziona davanti alla vettura del leader della gara, riducendo la velocità del gruppo e neutralizzando la competizione.

Una volta in pista, la Safety Car guida il gruppo di piloti a una velocità ridotta, che di solito è significativamente inferiore rispetto alla velocità di gara normale. Questo garantisce che i piloti abbiano il tempo di reagire in modo sicuro alle condizioni di pericolo e che i commissari di gara possano lavorare in sicurezza per risolvere la situazione.

Durante il periodo di Safety Car, i piloti devono mantenere una distanza costante tra le vetture e rispettare le istruzioni dei commissari di gara. È vietato sorpassare fino a quando la Safety Car non rientra ai box e la gara viene ripresa.

Una volta che la situazione di pericolo è stata risolta e la pista è stata resa sicura, la Safety Car rientra ai box. Prima del rientro, i piloti vengono avvertiti tramite segnali e comunicazioni radio. Quando la Safety Car lascia la pista, i piloti possono riprendere la gara e tornare alla velocità di gara normale.

Come si può notare dalla spiegazione fatta, la Safety Car e la linea di inizio jamiton (la linea rossa in figura [fig. 4.9] hanno un comportamento simile. Le vetture non prioritarie che raggiungono il punto di inizio della zona di jamiton vengono rallentate fino a quando il veicolo prioritario non ha superato il punto critico. Una volta che il veicolo prioritario ha attraversato il punto critico, la zona di jamiton si disattiva e le vetture non prioritarie possono riprendere la loro velocità normale. Quindi le auto subiscono un rallentamento finché i veicoli prioritari lo richiedono, e la linea di inizio jamiton avanza fino ad arrivare al punto critico dove, secondo i calcoli, il veicolo prioritario è già transitato.

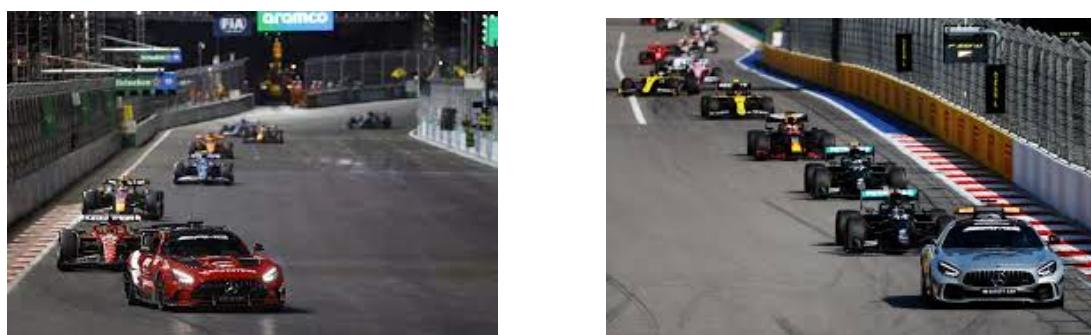


Figura 4.16: Safety Car Formula 1

Grazie a questo isomorfismo, dove la Safety Car è rappresentata dalla linea di inizio jamiton (rossa) e le auto di Formula 1 dai veicoli non prioritari, si ottiene una visione ancora più chiara del funzionamento del sistema Jamact.

4.6 Jamact e l'ottimizzazione delle risorse

Il modello Jamact si propone come una soluzione innovativa per l'ottimizzazione e la gestione del traffico veicolare, in particolare per i veicoli prioritari, minimizzando l'impatto sui veicoli non prioritari. Questo approccio trova fondamenta solide nella

teoria dell'ottimizzazione, la quale si concentra sull'allocazione efficiente delle risorse per raggiungere determinati obiettivi predefiniti. L'ottimizzazione è una branca della matematica applicata che si occupa di trovare la migliore soluzione possibile, data una serie di vincoli e un obiettivo da raggiungere. In contesti reali come la gestione del traffico, l'ottimizzazione permette di ridurre tempi di viaggio, minimizzare il consumo di carburante, e migliorare la sicurezza stradale. Modelli matematici e algoritmi sono strumenti essenziali per risolvere questi problemi complessi. Il modello Jamact può essere visto come un modello efficiente di assegnazione delle risorse e paragonabile ai più noti algoritmi utilizzati nei sistemi operativi, come l'algoritmo del banchiere. Sviluppato da Edsger Dijkstra nel 1965, questo algoritmo gestisce l'allocazione delle risorse assicurando che ogni richiesta possa essere soddisfatta senza compromettere la stabilità del sistema.

L'algoritmo del banchiere prevede che ogni processo (o veicolo, nel nostro contesto) faccia richieste di risorse che devono essere soddisfatte in modo tale da mantenere il sistema in uno stato sicuro. Un sistema è considerato sicuro se esiste un ordine in cui tutte le richieste possono essere soddisfatte senza causare deadlock. L'algoritmo controlla continuamente la disponibilità delle risorse e rifiuta le richieste che potrebbero portare il sistema in uno stato non sicuro.

Nell'ambito del traffico urbano, un esempio di applicazione dell'algoritmo del banchiere nel contesto della gestione delle congestioni è stato affrontato nello studio condotto nell'articolo "A Resource Allocation Technique for VANETs Inspired to the Banker's Algorithm"[29]. In questo contesto, i veicoli sono trattati come processi che richiedono risorse (strade), e l'algoritmo distribuisce i veicoli sui percorsi disponibili per ottimizzare il flusso di traffico e ridurre la congestione. Similmente, il modello Jamact utilizza questi principi per allocare porzioni di strada in maniera prioritaria a quei veicoli che ne hanno necessità, per poi liberare la risorsa e fornirla ai veicoli non prioritari subito dopo il passaggio del veicolo prioritario.

—5—

Simulatore per il modello Jamact

CONTENUTI: 5.1 Panoramica sul codice. 5.2 Esempio di applicazione dinamica. 5.3 Risultati simulazioni Jamact.

In questo capitolo sarà presentato il simulatore Jamact e le sue funzionalità. Verranno inoltre simulati vari casi di applicazione e saranno discussi i risultati ottenuti.

Il simulatore è stato sviluppato con l'obiettivo di fornire una linea guida per lo sviluppo di un futuro modello Jamact destinato all'uso quotidiano, mostrando i vantaggi dell'implementazione del modello e gli svantaggi derivanti dalla sua assenza.

Il simulatore è articolato in due fasi: la fase statica e la fase dinamica. Nella fase statica, vengono utilizzate librerie specifiche per tracciare la rete stradale di una porzione di città, individuando tutti i punti critici. In base al percorso del veicolo prioritario, vengono selezionati solo i punti critici di interesse e calcolati i vari punti di attivazione ed inizio jamiton presenti nella rete. Questa fase si conclude con la creazione di un grafo che evidenzia il percorso del veicolo prioritario e tutte le strade che convergono in un punto critico di interesse.

La fase dinamica, invece, si basa su una serie di algoritmi che, grazie all'uso di thread, simulano l'avanzamento di una serie di veicoli non prioritari e del veicolo prioritario verso un punto critico. In base a queste simulazioni, vengono generati e visualizzati una serie di risultati che illustrano lo stato del sistema e il traffico veicolare simulato. Il simulatore, nella fase dinamica, prevede sia l'utilizzo con Jamact attivo che con Jamact non attivo. Questo permette di valutare l'efficacia del modello implementato e di individuare eventuali criticità. Tutte le simulazioni sono state eseguite tenendo conto di diverse combinazioni di dati, applicati in modo simile sia al simulatore con Jamact attivo che a quello con Jamact non attivo.

5.1 Panoramica sul codice

Il simulatore Jamact è stato implementato in Python, sfruttando la vasta scelta di librerie offerte da questo linguaggio per sviluppare un codice affidabile e capace di elaborare dati in scenari realistici.

Per la fase statica, la libreria principale utilizzata è stata osmnx [B.2], uno strumento potente per scaricare, modellare, visualizzare e analizzare dati geospatiali di reti stradali e infrastrutture urbane direttamente da OpenStreetMap. Grazie a osmnx, è stato possibile creare il grafo stradale all'interno del quale collocare il veicolo prioritario e il suo percorso, individuando i punti critici relativi. Un'altra libreria fondamentale per la fase statica è stata networkx, che consente di creare, manipolare e studiare la struttura, le dinamiche e le funzioni di grafi complessi, offrendo strumenti per analizzare reti sia orientate che non orientate. Con networkx, è stato possibile visualizzare i dati elaborati nella fase statica tramite la creazione di immagini rappresentative.

Per la fase dinamica, invece, le librerie principali utilizzate sono state collections e threading. La libreria collections è stata impiegata per creare le code di veicoli che devono avanzare verso il punto critico, mentre threading è stata utilizzata per simulare l'avanzamento contemporaneo e parallelo dei veicoli, permettendo una simulazione realistica del traffico.

Static Provisioning

Partendo dalla fase statica del modello, è utilizzata la libreria osmnx per scaricare e creare una rappresentazione della rete stradale di una determinata area entro un raggio specificato, come si può vedere in figura [fig. 5.1]:



Figura 5.1: Mappa stradale Fuorigrotta

La mappa come quella in figura è il risultato della funzione `create_map`, ed è costruita sul grafo stradale fornito da OpenStreetMap. Su questa mappa si andrà poi a rappresentare il percorso di un ipotetico veicolo prioritario e si andranno anche a evidenziare tutti i punti critici presenti nel percorso.

```
def create_map(place_name, distance):
    graph = ox.graph_from_address(place_name, dist
        =distance, network_type='drive')
    graph_undirected = graph.to_undirected()
    return graph, graph_undirected
```

Per identificare i punti critici lungo un percorso sulla rete, è stata utilizzata la funzione seguente:

```
def critical_point(graph, graph_undirected,
                   start_point, end_point):
    start_node = ox.distance.nearest_nodes(graph,
                                            X=start_point[1], Y=start_point[0])
    end_node = ox.distance.nearest_nodes(graph, X=
                                           end_point[1], Y=end_point[0])

    path_nodes = nx.shortest_path(graph,
                                   start_node, end_node, 'length')
    internal_path_nodes = path_nodes[1:-1]

    critical_points = [node for node in
                       internal_path_nodes if 'street_count' in
                       graph_undirected.nodes[node] and
                       graph_undirected.nodes[node]['street_count']
                       > 2]
    return path_nodes, critical_points
```

La funzione `critical_point` analizza un percorso specificato sulla mappa stradale, identificando i punti critici lungo il tragitto. Utilizzando le librerie osmnx e networkx in Python, la funzione calcola il percorso ottimale tra due punti sulla rete stradale e individua i nodi critici in base al numero di strade adiacenti. Questa analisi fornisce informazioni per la comprensione della struttura e della congestione del traffico lungo il percorso definito. Inoltre, la funzione crea una variabile `critical_points` che contiene una lista di tutti i punti critici del percorso (in questo caso, gli incroci).

Utilizzando le funzioni spiegate, è stata creata la mappa stradale relativa al percorso di un veicolo prioritario, individuando anche i relativi punti critici di interesse. Nella figura [fig. 5.2] il percorso del veicolo prioritario è evidenziato in blu, mentre le strade evidenziate in verde rappresentano le vie che confluiscono in un punto critico

lungo questo percorso. I punti di inizio del jamiton sono posizionati sulle strade verdi, mentre i punti di attivazione del jamiton si trovano sulle strade blu.



Figura 5.2: Mappa stradale del percorso

Con la creazione di questa mappa, si conclude la fase statica. I risultati ottenuti in questa fase saranno fondamentali per la successiva fase di adattamento dinamico, che si baserà su queste previsioni per ottimizzare e migliorare il funzionamento del sistema in tempo reale.

Dynamic Adaption

La fase dinamica del simulatore si concentra sulla valutazione del modello Jamact in un incrocio ideale, dove le auto procedono da quattro direzioni diverse. Utilizzando le coordinate dei veicoli, la simulazione tiene conto del loro movimento verso il punto critico, considerando sia la propria velocità che l'impatto dei veicoli che li precedono. L'obiettivo principale è dimostrare che, con l'attivazione di Jamact, il veicolo prioritario arriva all'incrocio e lo supera prima che dei veicoli non prioritari presenti nel sistema. Al contrario, senza l'uso di Jamact, si evidenzia come i veicoli non prioritari, non essendo soggetti a rallentamenti, potrebbero precedere il veicolo prioritario all'arrivo all'incrocio e rallentarne il transito.

La simulazione inizia con la generazione casuale di un numero arbitrario di veicoli mediante la funzione **generateVehicle()**:

```
def generateVehicle(number_vehicle):

    new_vehicle = PriorityVehicle(index,
        vehicle_type, start_number, end_number,
        coordinates_priority, stop_lines_priority,
        activation_point, speed_priority)
    vehicles_queue = deque()

    while number_vehicle > count_vehicle:
        count_vehicle += 1
        vehicle_type = random.randint(0, 4)
        start_number = random.choice([0, 1, 2, 3])
        end_number = random.choice([0, 1, 2, 3])
        coordinates = generateCoordinate()
        stop_lines = next_critical_point(
            coordinates)
        starting_point = next_starting_point(
            stop_lines)
        speed = random.randint(0, 70) / 3.6
        index = len(vehicles_queue)

        new_vehicle = Vehicle(index, vehicle_type,
            start_number, end_number, coordinates,
            stop_lines, starting_point, speed)
        vehicles_queue.append(new_vehicle)

    return vehicles_queue
```

I veicoli creati possiedono una serie di attributi che indicano il tipo di veicolo (auto, moto, autobus, ecc.), il percorso (start_number e end_number) da percorrere, le coordinate di partenza, la velocità di viaggio, il punto di attivazione/inizio jamiton e le coordinate dell'incrocio. In particolare, si noti come la velocità rappresenta un valore compreso tra 0 e il limite massimo della velocità consentita, nel caso specifico ci si trova su una strada con limite 70km/h. Inoltre il percorso dei veicoli viene rappresentato con due numeri da 0 a 4 che indicano le quattro direzioni da cui possono giungere e dove possono arrivare rispetto all'incrocio.

In base ai valori generati dalla funzione generateVehicle(), la funzione **moveAllVehicles()** gestisce i movimenti dei veicoli verso il punto critico. In particolare, questa funzione crea un thread per ogni veicolo, rendendolo indipendente dagli altri, e attiva la funzione **move*()** per ciascuno di essi. In questo modo viene simulato il movimento dei veicoli, rendendoli indipendenti nei movimenti ma allo stesso tempo tenendo conto dei veicoli che precedono per simulare anche una corretta distanza di sicurezza tra di loro.

```

def moveAllVehicles(vehicles_queue , jam):

    def vehicle_thread(vehicle):
        while not stop_event.is_set():
            if jam == 1:
                vehicle.moveJam(vehicles_queue)
            else:
                vehicle.moveNoJam(vehicles_queue)

        threads = []
        for vehicle in vehicles:
            thread = threading.Thread(target=
                vehicle_thread , args=(vehicle,))
            threads.append(thread)
            thread.start()

        for thread in threads:
            thread.join()

```

Le funzioni moveJam() e moveNoJam() gestiscono rispettivamente i casi in cui il sistema Jamact è attivo e quelli in cui il sistema non è attivo. La funzione moveNoJam() è relativamente semplice, poiché sposta i veicoli in base alla loro velocità e direzione; pertanto, il codice non è riportato. Al contrario, la funzione moveJam() è più complessa e interessante. Di seguito viene fornito uno pseudocodice per illustrare e chiarire meglio i suoi compiti¹.

```

def moveJam(vehicle)
    if vehicle.cross_starting_point and
        priorityVehicle.cross_activation_point:
        update_speed = updateSpeed()
        if update_speed <= vehicle.speed:
            vehicle.speed = update_speed

    vehicle.advances()

    if vehicle.arrived_stop:
        print("Veicolo arrivato all'incrocio")

```

La funzione **moveJam()** è progettata per gestire il movimento dei veicoli all'interno di una zona di traffico regolata dal sistema Jamact. La funzione accetta un

¹La funzione completa è costituita da numerose righe di codice, la cui esposizione integrale risulterebbe poco rilevante. È importante comprendere la funzionalità e la logica di base.

veicolo non prioritario come input e esegue una serie di controlli e aggiornamenti. In primo luogo, verifica se il veicolo non prioritario ha raggiunto il punto di inizio jamiton e se il veicolo prioritario ha superato il punto di attivazione del jamiton. Se entrambe queste condizioni sono soddisfatte, la velocità del veicolo non prioritario viene aggiornata tramite la funzione `updateSpeed()` per far sì che entri nella zona di jamiton correttamente. Successivamente, il veicolo avanza e al raggiungimento del punto critico, viene stampato un messaggio di conferma del suo arrivo all'incrocio.

La velocità dei veicoli non prioritari viene aggiornata in base al valore ritornato dalla funzione `updateSpeed()`:

```
def updateSpeed(vel_priority, activation, start):  
  
    time = activation / vel_priority  
    velocity_final = start / time  
  
    return velocity_final
```

La funzione `updateSpeed()` calcola e aggiorna la velocità di un veicolo non prioritario utilizzando diversi parametri: la velocità di un veicolo prioritario, la distanza tra il veicolo prioritario e la fine dell'incrocio, e la distanza dal punto di inizio del jamiton all'inizio dell'incrocio. Utilizzando questi valori, la funzione determina il tempo necessario affinché il veicolo non prioritario raggiunga il punto critico. In seguito, calcola la velocità finale del veicolo non prioritario in base a questo tempo e alla distanza di partenza. La funzione ritorna quindi la velocità aggiornata, alla quale il veicolo non prioritario deve procedere per evitare di ostacolare il veicolo prioritario. È importante notare che la velocità del veicolo non prioritario viene aggiornata solo se il valore restituito dalla funzione `updateSpeed()` è inferiore alla sua velocità corrente, cioè solo nel caso in cui il veicolo non prioritario debba rallentare.

5.2 Esempio di applicazione dinamica

In questa sezione verrà presentata un'applicazione pratica del codice discusso in precedenza, con l'obiettivo di mostrare i risultati ottenuti dalla sua esecuzione. Saranno illustrati i passaggi eseguiti per simulare il movimento dei veicoli in un sistema di traffico regolato da Jamact, evidenziando le funzioni chiave e le loro interazioni. Questa dimostrazione mostrerà come le velocità dei veicoli non prioritari vengono calcolate e aggiornate in base alle condizioni del traffico, e come il sistema garantisce che i veicoli non prioritari non ostacolino quelli prioritari. I risultati ottenuti dall'esecuzione del codice saranno analizzati per valutare l'efficacia e l'efficienza dell'algoritmo implementato. Inoltre, la simulazione portata qui come esempio tiene conto di condizioni di traffico ottimali, dove la congestione stradale è nella media e l'incrocio preso di esempio è un incrocio ideale.

I dati utilizzati per la simulazione con Jamact attivo e quelli senza Jamact attivo sono identici. Questo approccio consente di confrontare direttamente i risultati ottenuti partendo dallo stesso punto di partenza. Come si può ben vedere dalla sequenza di immagini nella figura [fig. 5.3], il simulatore con Jamact attivo permette al veicolo prioritario di transitare per il punto critico senza impedimenti e mantenendo la sua velocità di guida.

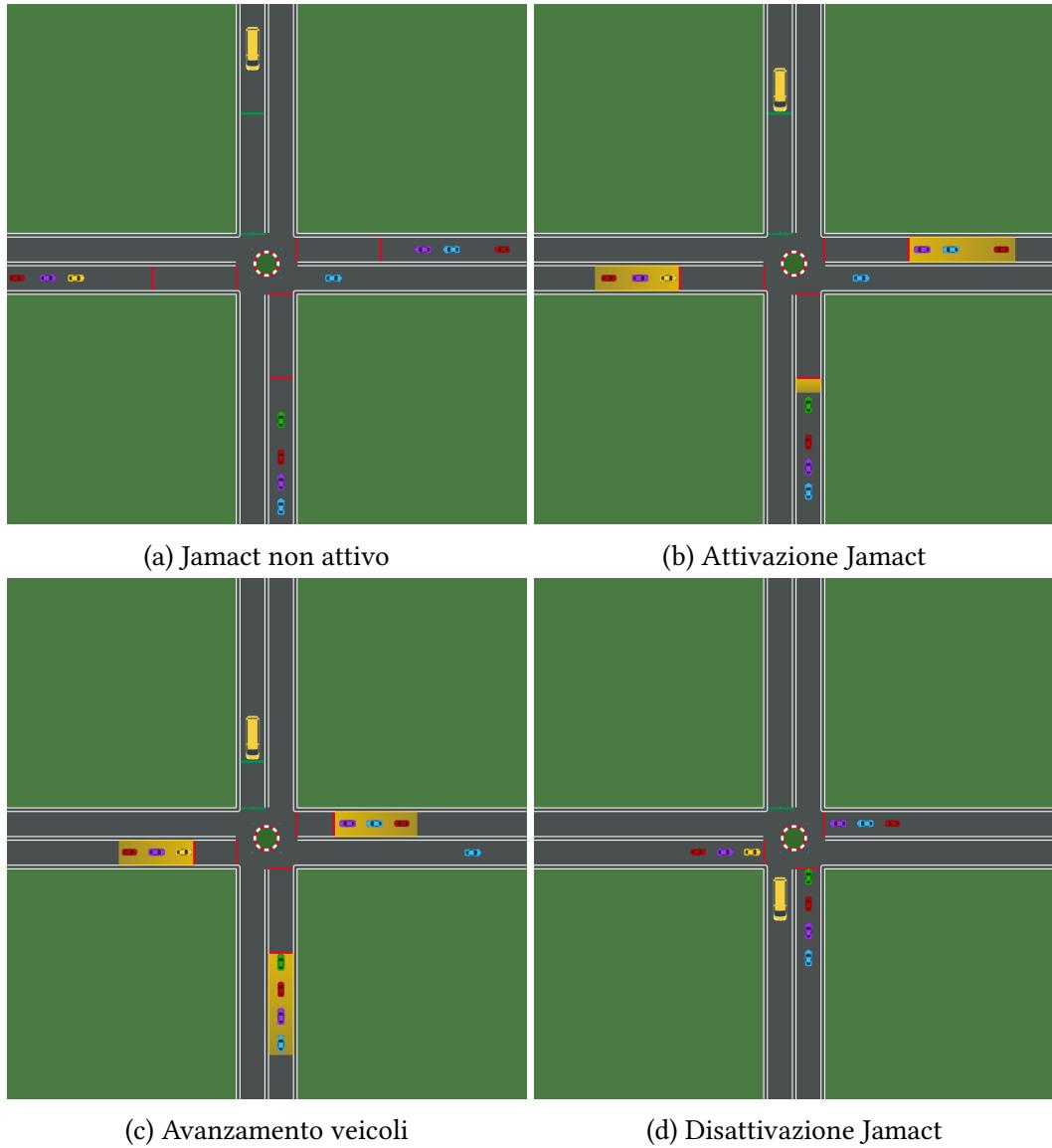


Figura 5.3: Simulazione con Jamact Attivo

Nel dettaglio, nell'immagine [fig. 5.3a] è mostrato lo stato della strada prima che il sistema Jamact entri in funzione. Sulla base dei dati presenti in questa fase, il sistema adatterà il suo comportamento per gestire al meglio la dinamicità della situazione. Nell'immagine [fig. 5.3b], l'arrivo del veicolo prioritario al punto di attivazione attiva il sistema Jamact. Contemporaneamente, i veicoli non prioritari che raggiungono il punto di inizio saranno soggetti al jamiton. L'intensità di quest'ultimo è regolata dina-

micamente in base all'avanzamento del veicolo prioritario. Nell'immagine [fig. 5.3c] si osserva il movimento dei veicoli e dei relativi jamiton e come l'avanzamento del veicolo prioritario sia maggiore rispetto agli altri, proprio grazie al rallentamento applicato. Infine, nell'ultima immagine [fig. 5.3d], i veicoli non prioritari raggiungono l'incrocio dopo che il veicolo prioritario lo ha superato.

In figura [fig. 5.4], è rappresentato un file di log relativo alle immagini sopra riportate, che registra in modo dettagliato e cronologico le attività del sistema, consentendo di monitorarne il comportamento e capire se tutto funziona correttamente.

```
SIMULAZIONE JAMACT DISATTIVATO
Il veicolo right 0 è arrivato all'incrocio: 15:36:11
Il veicolo right 1 è arrivato all'incrocio: 15:36:12
Il veicolo left 0 è arrivato all'incrocio: 15:36:14
Il veicolo left 1 è arrivato all'incrocio: 15:36:14
Il veicolo left 2 è arrivato all'incrocio: 15:36:14
Il veicolo down 0 è arrivato all'incrocio: 15:36:14
Il veicolo prioritario 0 è arrivato all'incrocio: 15:36:14
SIMULAZIONE JAMACT ATTIVATO
Il veicolo prioritario è arrivato al punto di attivazione: 15:50:13 con una velocità di 18.06
Il veicolo right 0 è arrivato al punto di inizio: 15:50:13:13 con una velocità di 17.78
-- Generazione velocità 12.53 veicolo 0
Il veicolo right 1 è arrivato al punto di inizio: 15:50:14:14 con una velocità di 18.89
-- Generazione velocità 13.59 veicolo 1
Il veicolo left 0 è arrivato al punto di inizio: 15:50:14:14 con una velocità di 15.00
-- Generazione velocità 13.59 veicolo 0
Il veicolo down 0 è arrivato al punto di inizio: 15:50:14:14 con una velocità di 14.44
-- Generazione velocità 13.92 veicolo 0
Il veicolo right 2 è arrivato al punto di inizio: 15:50:16:16 con una velocità di 17.50
-- Generazione velocità 16.25 veicolo 0
Il veicolo prioritario è arrivato all'incrocio: 15:50:19
Il veicolo prioritario ha superato l'incrocio: 15:50:25
Il veicolo right 0 è arrivato all'incrocio: 15:50:25
Il veicolo right 1 è arrivato all'incrocio: 15:50:25
Il veicolo left 0 è arrivato all'incrocio: 15:50:25
Il veicolo down 0 è arrivato all'incrocio: 15:50:25
Il veicolo right 2 è arrivato all'incrocio: 15:50:25
```

Figura 5.4: Log della simulazione.

Nello scenario con Jamact non attivo il veicolo prioritario raggiunge l'incrocio prima che gli altri veicoli lo abbiano superato, costringendolo a fermarsi fino a quando l'incrocio non è completamente libero. In questa situazione, il veicolo prioritario deve attendere il passaggio di sei veicoli prima di poter proseguire, subendo un ritardo significativo.

In un sistema senza priorità, ogni veicolo è trattato allo stesso modo ed è quindi soggetto alle stesse regole di precedenza degli altri veicoli. Con Jamact invece, il veicolo prioritario raggiunge e supera l'incrocio prima di qualsiasi altro veicolo. Questo conferma l'ipotesi iniziale secondo cui un veicolo prioritario, rallentando i veicoli non prioritari in base a calcoli opportuni, può creare un varco privilegiato in cui può procedere alla velocità desiderata senza dover mai fermarsi a causa di ostacoli creati da altri veicoli.

5.3 Risultati simulazioni Jamact

Per verificare la validità del modello e valutare i risultati della sua applicazione, sono state condotte una serie di simulazioni, sia con che senza il suo utilizzo, al fine di effettuare confronti e visualizzare eventuali miglioramenti.

Le simulazioni sono state eseguite configurando il simulatore in un contesto di strada urbana di scorrimento, caratterizzata da un limite di velocità di 70 km/h e una velocità minima di 27 km/h [fig. 4.2] e variando sistematicamente la velocità di percorrenza di ogni veicolo. Il punto critico preso in considerazione è un incrocio a quattro strade a doppia corsia (una per senso di marcia). Inoltre sono state condotte diverse simulazioni per diverse densità di traffico, in particolare sono stati individuati tre range di traffico (basso, medio e alto) e per ogni range sono state condotte circa dieci simulazioni utilizzando un numero variabile di veicoli per ogni intervallo di densità. In particolare, per il range di bassa densità di traffico sono stati considerati quattro veicoli; per il range di media densità di traffico sono stati utilizzati otto veicoli; infine, per il range di alta densità di traffico sono stati impiegati dodici veicoli. Questa diversificazione è stata fondamentale per analizzare come variazioni nella densità del traffico influenzino le prestazioni del sistema.

Nella tabella [tab. 5.1] sono riportati i risultati medi delle simulazioni suddivisi per range di densità di traffico. Come si vedrà, i dati raccolti dimostrano come il sistema Jamact abbia un impatto mediamente positivo, e come riesca a favorire il transito dei veicoli prioritari.

	Jamact Attivo	Jamact Non Attivo
Bassa densità di traffico		
Tempo medio percorrenza veicoli non prioritari	13,48 secondi	12,25 secondi
Velocità Media veicoli non prioritari	15,65 m/s	17,23 m/s
Numero medio veicoli interferenti individuati	1,4 su 4	3,2 su 4
Tempo medio percorrenza veicolo prioritario	12,55 secondi	13,12 secondi

Tabella 5.1 – Continua alla pagina successiva

	Jamact Attivo	Jamact Non Attivo
Media densità di traffico		
Tempo medio percorrenza veicoli non prioritari	14,28 secondi	12,92 secondi
Velocità Media veicoli non prioritari	14,78 m/s	16,33 m/s
Numero medio veicoli interferenti individuati	2,2 su 8	4,2 su 8
Tempo medio percorrenza veicolo prioritario	12,24 secondi	13,59 secondi
Alta densità di traffico		
Tempo medio percorrenza veicoli non prioritari	16,41 secondi	14,34 secondi
Velocità Media veicoli non prioritari	12,86 m/s	14,71 m/s
Numero medio veicoli interferenti individuati	4,7 su 12	6,8 su 12
Tempo medio percorrenza veicolo prioritario	11,42 secondi	12,25 secondi

Tabella 5.1: Risultati Simulazioni Jamact

Bassa densità di traffico:

In condizioni di bassa densità di traffico la velocità media dei veicoli non prioritari è leggermente superiore con Jamact non attivo (17,23 m/s) rispetto quando è attivo (15,65 m/s), con un tempo medio di percorrenza più breve senza Jamact (12,25 secondi contro 13,48 secondi con Jamact attivo). Inoltre, con Jamact attivo, la percentuale

di veicoli interferenti identificati è notevolmente inferiore (35% contro l'80% senza Jamact). Questi dati indicano che Jamact migliora il transito dei veicoli prioritari del 4,54%, riducendo il tempo di percorrenza di 0,57 secondi, diminuendo i tempi di percorrenza dell'11,60% (aumento di 1,77 secondi) per i veicoli non prioritari.

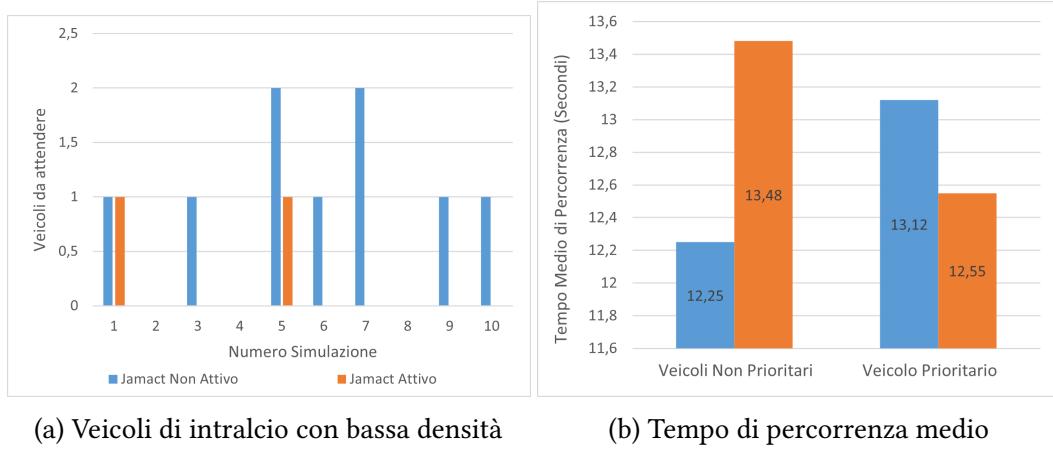


Figura 5.5: Grafici impatto Jamact bassa densità

Media densità di traffico:

Con una densità di traffico media, la velocità media dei veicoli non prioritari diminuisce con Jamact attivo (14,78 m/s contro 16,33 m/s senza Jamact), e il tempo medio di percorrenza è più lungo (14,28 secondi contro 12,92 secondi).

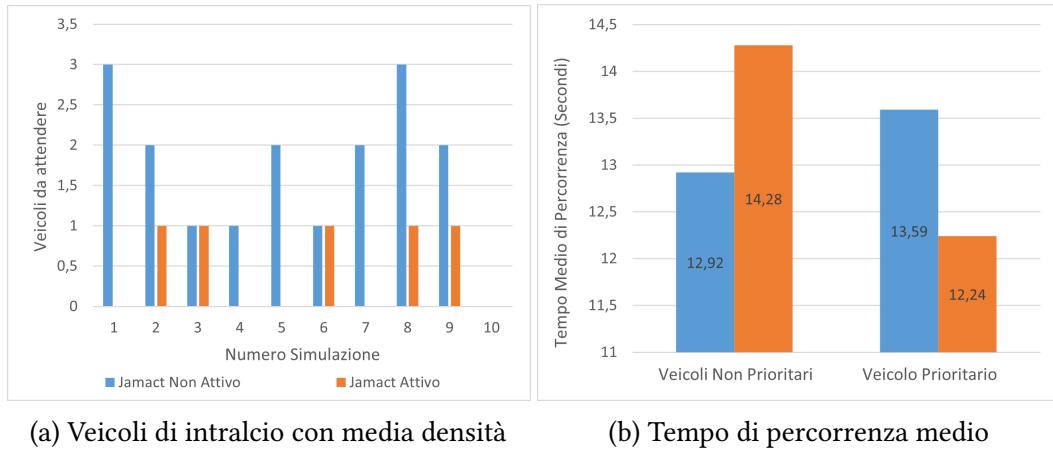


Figura 5.6: Grafici impatto Jamact media densità

La percentuale di veicoli interferenti rilevati scende significativamente con Jamact attivo (27,5% contro 52,5% senza). In questo scenario, Jamact migliora il tempo di percorrenza dei veicoli prioritari dell'11,03%, riducendo il tempo di percorrenza di 1,35 secondi, e diminuendo i tempi di percorrenza per i veicoli non prioritari del 9,52% (aumento di 1,36 secondi).

Alta densità di traffico:

In situazioni di alta densità di traffico, la velocità media dei veicoli non prioritari con Jamact attivo è inferiore (12,86 m/s rispetto a 14,71 m/s senza Jamact), e il tempo medio di percorrenza è più lungo (16,41 secondi contro 14,34 secondi). La percentuale di veicoli interferenti rilevati è inferiore con Jamact attivo (39% contro 56% senza). In quest'ultimo caso i dati mostrano una bassa efficienza del modello Jamact, il quale riesce favorire il transito dei veicoli prioritari mediamente di circa il 7,27%, diminuendo di circa 0,83 secondi il tempo totale impiegato per superare il punto critico. Infine, l'impatto del sistema sui veicoli non prioritari è di un rallentamento di circa il 12,61% (2,07 secondi).

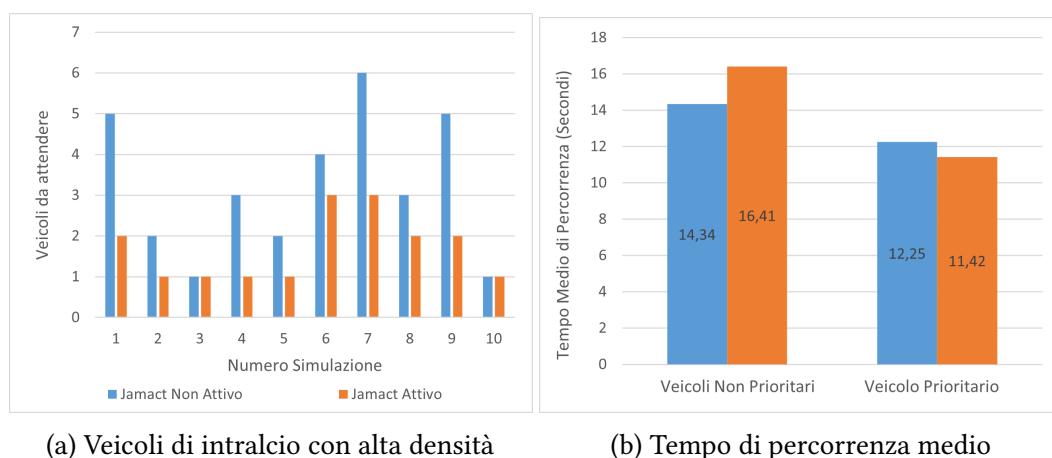


Figura 5.7: Grafici impatto Jamact alta densità

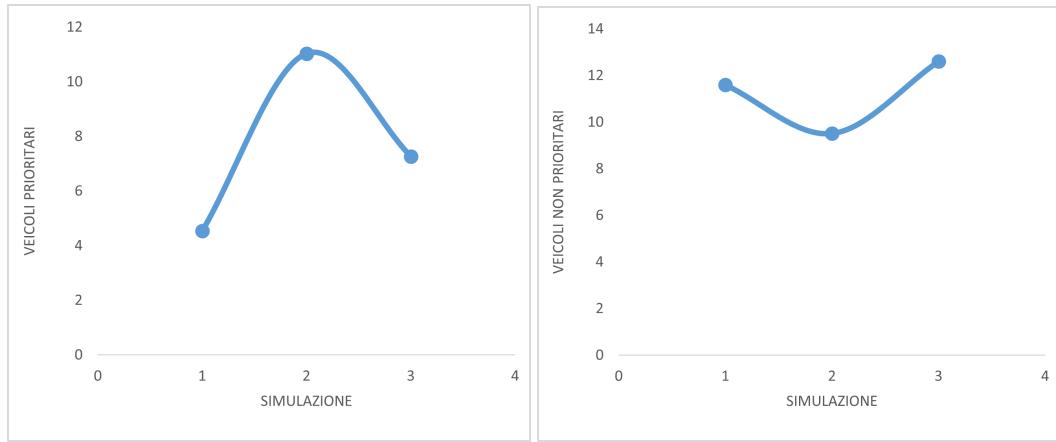
Risultati totali:

I risultati ottenuti dalle simulazioni per i vari range di densità di traffico sono stati elaborati e hanno portato al calcolo di altri dati riassunti nella tabella [tab. 5.2].

	Impatto Veicoli Prioritari	Impatto Veicoli Non Prioritari
Bassa densità di traffico	4,54%	11,60%
Media densità di traffico	11,03%	9,52%
Alta densità di traffico	7,27%	12,61%

Tabella 5.2: Impatto Jamact per densità di traffico

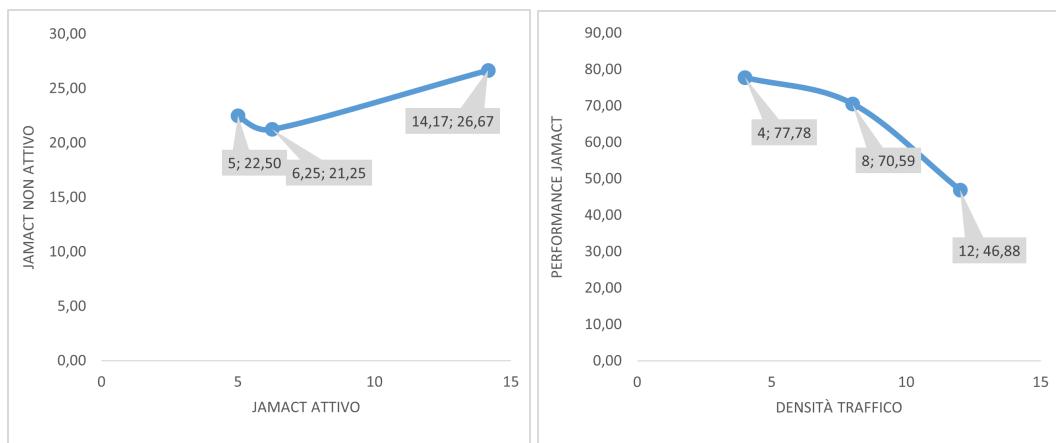
Da questi dati emerge chiaramente come a bassa densità di traffico, l'attivazione di Jamact ha un impatto sui veicoli non prioritari di circa 11,60%, invece sui veicoli prioritari l'impatto è minore e si attesta intorno al 4,54%. Man mano che la densità di traffico aumenta a media densità, l'impatto sui veicoli non prioritari passa al 9,52%, invece l'impatto sui veicoli prioritari è di circa 11,03%. Infine ad alta densità di traffico l'impatto sui veicoli non prioritari raggiunge il 12,61%, mentre per i veicoli prioritari si riduce al 7,27%.



(a) Impatto Jamact sui veicoli prioritari (b) Impatto Jamact sui veicoli non prioritari

Figura 5.8: Grafici impatto Jamact

I grafici della figura [fig. 5.8] evidenziano i dati ottenuti dalle simulazioni e riassunti nella tabella [tab. 5.2]. Dal grafico [fig. 5.8a] si nota come nelle simulazioni con media densità di traffico l'impatto del sistema Jamact è maggiore e più incisivo, rispetto agli scenari con traffico basso e alto. Invece, dal grafico [fig. 5.8b] emerge chiaramente una situazione in cui l'impatto sui veicoli non prioritari è minore nei casi di medio traffico.



(a) Numero medio di veicoli interferenti (b) Grafico performance – densità

Figura 5.9: Grafici performance Jamact

Infine, i risultati evidenziano chiaramente che l'incremento della densità del traffico porta a un aumento del numero di veicoli che interferiscono con il percorso del veicolo prioritario, come illustrato nel grafico della media percentuale nella figura [fig. 5.9a]. Questo fenomeno incide direttamente sulle prestazioni complessive del sistema, come dimostrato nel grafico delle performance in relazione alla densità di traffico nella figura [fig. 5.9b].

Dall'analisi di tutti i dati raccolti e dei grafici elaborati, emergono alcune tendenze chiare:

1. L'efficacia di Jamact nel favorire i veicoli prioritari aumenta con l'aumentare della densità del traffico. Il sistema riesce a garantire tempi di percorrenza più bassi per i veicoli prioritari in tutte le condizioni, ma questo vantaggio diventa particolarmente evidente in condizioni di traffico intenso.
2. Il "costo" di questo vantaggio per i veicoli prioritari si riflette in un rallentamento per i veicoli non prioritari, che diventa più significativo con l'aumentare della densità del traffico.
3. La differenza percentuale tra i tempi di percorrenza dei veicoli prioritari e non prioritari tende ad aumentare con l'intensificarsi del traffico quando Jamact è attivo, suggerendo una maggiore efficacia del sistema in condizioni di congestione.
4. Nonostante l'aumento dei tempi di percorrenza per i veicoli non prioritari, il sistema Jamact sembra riuscire a mantenere un flusso di traffico gestibile anche in condizioni di alta densità, evitando blocchi completi o congestioni estreme.

Le simulazioni condotte hanno dimostrato che il modello di base è efficace nel facilitare il passaggio dei veicoli prioritari, specialmente in situazioni di traffico moderato, permettendo loro di superare rapidamente i punti critici senza incontrare ostacoli significativi. Tuttavia, è emerso che l'impatto del sistema è meno evidente in condizioni di traffico molto leggero o molto intenso rispetto a quelle di traffico medio. Ma nonostante ciò le performance del sistema rilevate tramite le simulazioni restano comunque molto positive. Infine, è importante notare che questo vantaggio comporta un certo livello di sacrificio per i veicoli non prioritari, aspetto che potrebbe richiedere ulteriori considerazioni in termini di ottimizzazione e bilanciamento del sistema.

—6—

Modello Jamact per Autobus

CONTENUTI: **6.1 Una panoramica sull'utilizzo degli autobus.** **6.2 Estensione modello Jamact – Autobus.** **6.3 Il problema del Commesso Viaggiatore.** **6.4 Isomorfismo con il Trasporto Ferroviario e le Coincidenze dei Treni.** **6.5 Esempio di applicazione dinamica del modello esteso.** **6.6 Risultati simulazioni modello Jamact per autobus.**

In questo capitolo sarà presentato una estensione del modello Jamact per un caso d'uso dove i veicoli prioritari sono i mezzi di trasporto, in particolare gli autobus. Vedremo come l'applicazione del modello Jamact possa favorire l'utilizzo dei mezzi di trasporto e come ciò possa incidere sulla vita quotidiana.

In un'epoca in cui le città affrontano sfide crescenti legate alla mobilità, all'inquinamento e alla sostenibilità, il trasporto pubblico rappresenta un valido alleato alla mobilità sostenibile. Tra le diverse opzioni di trasporto pubblico, gli autobus rappresentano una soluzione fondamentale nel contrasto alle emissioni di CO_2 e alla diminuzione del traffico stradale soprattutto nelle grandi città.

Gli autobus offrono numerosi vantaggi sia dal punto di vista ambientale che sociale. Innanzitutto, contribuiscono significativamente alla riduzione delle emissioni di gas serra e degli inquinanti atmosferici. Un autobus, infatti, può sostituire decine di automobili, diminuendo la congestione stradale e migliorando la qualità dell'aria. Inoltre, l'uso del trasporto pubblico promuove uno stile di vita più sano e attivo, incoraggiando le persone a camminare verso le fermate e a interagire maggiormente con l'ambiente urbano.

In questo capitolo, verranno analizzati gli ultimi dati sulla mobilità urbana degli italiani, le loro abitudini e le motivazioni di queste, e in particolare verrà fatto un focus sull'utilizzo degli autobus urbani. Verrà fornita, quindi, una panoramica completa e approfondita delle potenzialità degli autobus come strumento di trasformazione urbana e di miglioramento della qualità della vita.

L'obiettivo, infine, è quello di estendere il modello di gestione del traffico Jamact, incorporando le specifiche problematiche legate agli autobus. Questa estensione mira non solo a migliorare l'efficienza operativa del servizio di autobus, ma anche a incentivare il loro utilizzo da parte dei cittadini. Un incremento nell'uso degli autobus può portare a una riduzione significativa del traffico veicolare privato, contribuendo a

migliorare la qualità dell'aria, ridurre i tempi di spostamento e fornire un'alternativa di trasporto economica e accessibile.

6.1 Una panoramica sull'utilizzo degli autobus

L'utilizzo dei mezzi pubblici rappresenta una soluzione efficace per contrastare l'inquinamento atmosferico, ridurre le congestioni stradali e mitigare i relativi disagi. Secondo i dati ISPRA del 2019, gli autobus hanno contribuito solo al 3,1% delle emissioni di gas serra del traffico stradale, rispetto al 68,7% delle autovetture e al 25,4% dei veicoli commerciali leggeri e pesanti. Tra il 1990 e il 2019, le emissioni di gas serra degli autobus sono diminuite del 22,8%, in contrasto con un aumento generale del 3,9% (+20% per le autovetture). In termini di congestione stradale, un singolo autobus può sostituire oltre 20 automobili, riducendo significativamente il traffico, il consumo di carburante (3% del totale trasporto su strada), e l'occupazione del suolo pubblico (-87% rispetto all'auto), oltre alle emissioni.

Tuttavia, in Italia, prevale ancora l'uso dell'auto privata rispetto a soluzioni più sostenibili. Un'indagine condotta da Isfort, presentata nel "19° Rapporto sulla mobilità degli italiani" [30], rivela che il livello di sicurezza percepito per l'auto raggiunge un punteggio di 8,9, rispetto al 4,7 degli autobus, evidenziando la percezione dell'auto come mezzo più sicuro.

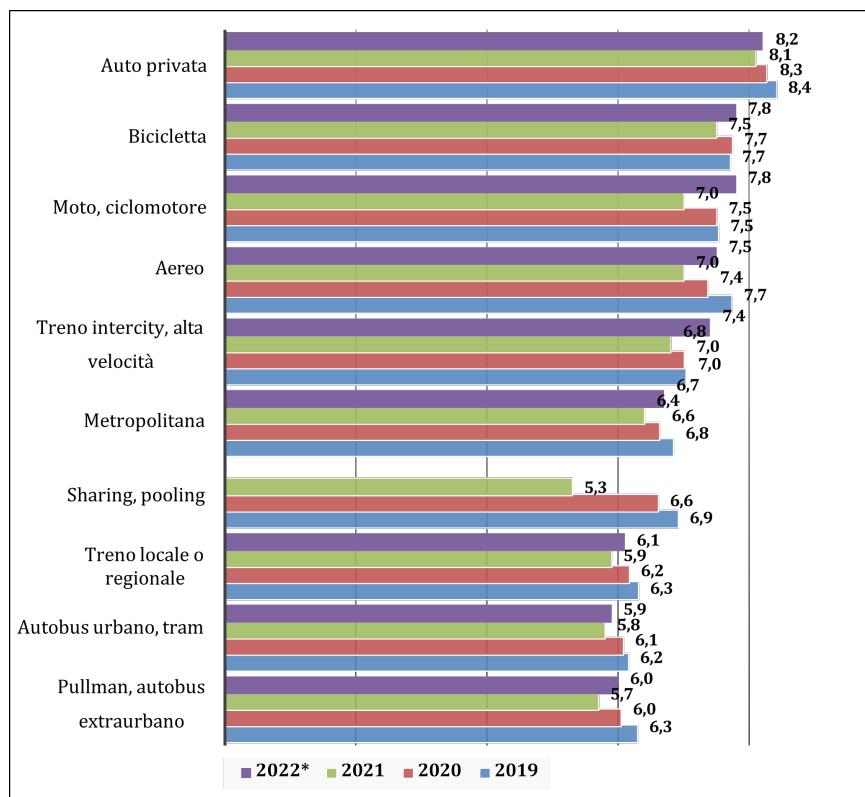


Figura 6.1: Utilizzo mezzi di trasporto

Un report ISTAT del 2018 [31] mostra che tra i cittadini lavoratori o studenti, il 73,7% degli occupati usa esclusivamente mezzi privati per i propri spostamenti, il 7,0% usa solo mezzi pubblici e il 4,1% usa entrambi. Tra gli studenti, il 38,3% utilizza unicamente mezzi privati, il 26,6% mezzi pubblici e il 6,1% entrambi. Anche Isfort conferma che l'automobile è il mezzo di trasporto preferito, con un indice medio di soddisfazione costantemente sopra l'8 su una scala da 1 a 10, mentre i mezzi di trasporto pubblico urbano e locale hanno subito un calo significativo della soddisfazione percepita dagli utenti, attestandosi intorno alla sufficienza [fig. 6.1].

Un'indagine più recente di Isfort, presentata nel "20° Rapporto sulla mobilità degli italiani" [32], evidenzia un quadro poco incoraggiante per la soddisfazione relativa ai mezzi pubblici. Dal grafico [fig. 6.2] emerge che, tra i mezzi pubblici urbani, la soddisfazione per autobus e tram è diminuita di circa 10 punti negli ultimi venti anni (dal 75% al 65% circa).

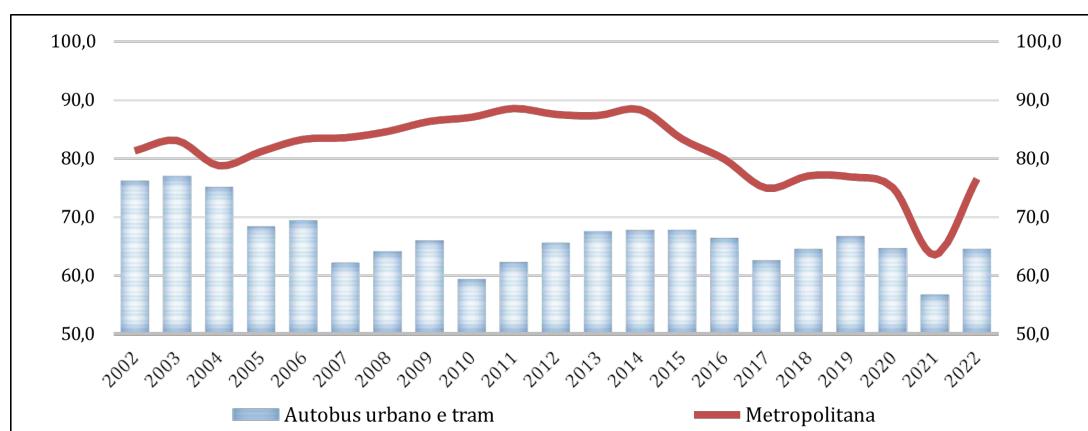


Figura 6.2: Soddisfazione utenti per Autobus

Un ulteriore approfondimento sulla soddisfazione per i mezzi pubblici ha individuato requisiti-chiave di prestazione del trasporto pubblico (frequenza delle corse, puntualità, sicurezza, ecc.) e ha esplorato sia il livello di soddisfazione percepita che l'importanza attribuita a ciascun requisito. Il grafico [fig. 6.3] riassume i risultati del focus.

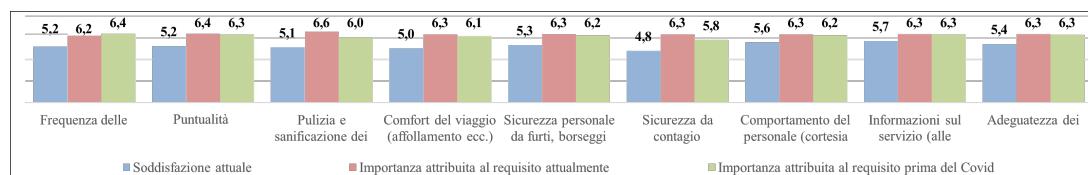


Figura 6.3: Grado di soddisfazione per i requisiti chiave

I dati medi complessivi mostrano un significativo grado di insoddisfazione per tutti i requisiti di qualità del servizio: tutti i punteggi si collocano al di sotto della sufficienza (su una scala da 1 a 10).

Gli intervistati che non utilizzano o utilizzano raramente i mezzi pubblici hanno indicato come principali ragioni del non utilizzo:

- Tempo eccessivo di percorrenza (61,8%)
- Scomodità e scarso comfort dei mezzi (65,9%)
- Mancanza di puntualità (63,8%)
- Scarsa frequenza e regolarità del servizio (poco oltre il 50%)

Da un'indagine condotta dal Sole24Ore emerge che gli utenti chiedono mezzi meno affollati (48%) e tempi di viaggio inferiori (25%). Le priorità includono un servizio più frequente, efficiente, affidabile e confortevole [33] [fig. 6.4].

Cosa potrebbe indurla a (tornare a) utilizzare di più i mezzi pubblici? Indichi al massimo tre risposte



Base: hanno ridotto l'uso di mezzi pubblici (917 casi). Valori in %

Figura 6.4: Cosa chiedono gli utenti

Un sondaggio ANSA [34] conferma che la percezione degli utenti sugli autobus pubblici in Italia è generalmente negativa: il 71% degli italiani ritiene che i mezzi pubblici siano inadeguati e il 55% li considera inquinanti. Questa percezione negativa è dovuta in parte alla scarsa qualità dei servizi, alla mancanza di infrastrutture e alla bassa frequenza, specialmente in alcune regioni del Sud Italia. Inoltre, la gestione dei servizi pubblici è spesso criticata per mancanza di trasparenza ed efficacia, con il 40% dei servizi pubblici che non rispetta i tempi di arrivo e partenza previsti e il 30% dei passeggeri che si lamenta della scarsa qualità dei servizi.

Questi dati evidenziano che il trasporto pubblico su autobus presenta sia vantaggi sia svantaggi. Tra i principali svantaggi vi sono ritardi e cancellazioni, particolarmente comuni nelle grandi città, dove il traffico intenso può causare problemi operativi significativi. La scarsa frequenza degli autobus in molte aree obbliga i passeggeri ad attese prolungate, rendendo il servizio meno pratico. La mancanza di corsie preferenziali in molte città riduce ulteriormente l'efficienza del trasporto pubblico su autobus, rendendoli meno competitivi rispetto ai mezzi privati. Anche la pulizia e il comfort dei mezzi sono frequentemente oggetto di critiche.

D’altro canto, il trasporto pubblico su autobus offre notevoli vantaggi economici e ambientali. È generalmente più economico rispetto all’uso dell’auto privata e, specialmente gli autobus a metano o elettrici, contribuiscono significativamente alla riduzione delle emissioni inquinanti, migliorando la qualità dell’aria. L’uso esteso del trasporto pubblico può anche ridurre il numero di auto in circolazione, diminuendo il traffico e migliorando ulteriormente l’ambiente urbano.

In conclusione, i mezzi di trasporto, in particolare gli autobus, rappresentano una componente cruciale per il futuro della mobilità sostenibile. Grazie alle loro caratteristiche, gli autobus sono fondamentali per combattere l’inquinamento atmosferico e le congestioni stradali. Promuoverne l’uso comporterebbe numerosi vantaggi. In quest’ottica, l’applicazione delle caratteristiche del modello Jamact ai veicoli prioritari come gli autobus può rappresentare uno sviluppo significativo nella mobilità sostenibile.

6.2 Estensione modello Jamact – Autobus

Per estendere il modello Jamact introducendo gli autobus come veicoli prioritari, e quindi adattandolo a questa tipologia di veicoli, è necessario considerarne le caratteristiche specifiche e le implicazioni che ciò comporta sul sistema di gestione del traffico. Gli autobus, essendo veicoli di trasporto pubblico, viaggiano maggiormente all’interno del centro urbano, percorrendo strade spesso molto congestionate e con impedimenti di vario tipo che possono rallentare l’andamento. Inoltre, essendo dei veicoli di trasporto pubblico, sono veicoli che non hanno un andamento nel percorso regolare, ma devono fermarsi e ripartire frequentemente per fare le varie fermate. Oltre a ciò i tempi di fermata possono essere variabili, per esempio a seconda del numero di passeggeri che salgono e scendono dal mezzo. Tutte queste variabili, che sono legate soprattutto alla dinamicità degli eventi, rendono l’applicazione del modello Jamact ai veicoli come gli autobus più ostica.

L’estensione del modello Jamact per gli autobus, deve quindi prendere in considerazione più fattori. Prima di tutto è fondamentale tenere traccia nel percorso anche delle fermate che sono presenti all’interno del tratto tra punto di attivazione e punto critico. Questo perché, staticamente, già è previsto che il veicolo dovrà fermarsi, e quindi rallentare e poi ripartire, e in base a ciò va riadattato il modello per gestire questo comportamento del veicolo prioritario. Oltre a ciò, per quanto riguarda i veicoli come gli autobus, i percorsi che dovranno fare sono già noti. Questo elemento potrebbe essere sfruttato per calcolare staticamente, non solo i punti di attivazione e inizio jamiton, ma potrebbe essere pianificato tutto il percorso basandosi anche sugli orari. Facendo ciò, tutti i veicoli non prioritari che verrebbero a trovarsi nel punto critico nel range di tempo che va da quando gli autobus arrivano al punto critico e poi lo superano, potrebbero essere rallentati con una logica diversa rispetto a quella prevista dal modello di base.

Quindi, per quanto riguarda veicoli prioritari come gli autobus, le informazioni di base conosciute sono molteplici. Come detto, ogni autobus seguirà un percorso prefissato, con degli orari previsti di arrivo e partenza alle varie fermate. Tutti questi

dati sono già calcolati e disponibili per ogni tipologia di autobus, garantendo all'estensione del modello Jamact una maggior precisione. Infatti l'idea di base è quella di comunicare ai veicoli non prioritari, già in fase di definizione del percorso di questi, i punti in cui dovranno rispettare dei limiti. Facendo così si Andrà a raffinare ancora di più il rallentamento dei veicoli non prioritari, impattando sempre meno su di essi. I veicoli non prioritari, conoscendo in anticipo la finestra temporale in cui devono raggiungere il punto critico (arrivando prima dell'autobus o dopo il suo passaggio), possono regolare la velocità di percorrenza in qualsiasi momento, assicurandosi di arrivare a un incrocio solo in determinati orari.

Sarà fondamentale effettuare un training continuo sui dati disponibili, in particolare sugli orari di arrivo e partenza alle varie fermate e sui tempi di percorrenza degli autobus. Con un training continuo su questi dati, i calcoli elaborati in fase statica diventeranno sempre più raffinati e precisi.

Static Provisioning

La fase di "Static Provisioning" sarà estesa per includere la pianificazione del percorso degli autobus. Questo processo considererà non solo i punti critici lungo il percorso, ma anche le fermate degli autobus e le aree di carico e scarico passeggeri. Poiché tutti questi dati sono noti in anticipo e ogni autobus seguirà sempre lo stesso percorso previsto, la fase statica del modello Jamact sarà notevolmente semplificata. Il concetto di linea di attivazione e linea di inizio jamiton sarà ancora presente, ma con un impatto minore, poiché l'obiettivo è permettere ai veicoli non prioritari di autogestirsi. Questi veicoli sapranno in anticipo le finestre di transito disponibili per ogni incrocio e potranno regolare la loro velocità per rispettare tali vincoli. Nel caso in cui ciò non avvenga, le linee di attivazione e di inizio jamiton interverranno per garantire una maggiore robustezza al sistema, applicando la stessa logica del modello di base. La differenza nel calcolo dei punti di attivazione e di inizio jamiton rispetto al modello base consiste nell'inclusione delle fermate degli autobus lungo il percorso negli stessi.

Il sistema calcolerà inizialmente i punti di inizio jamiton e di attivazione come nel modello base. Successivamente, sarà necessario verificare se nello spazio tra il punto di attivazione del jamiton e il punto critico vi siano fermate che potrebbero essere gestite staticamente per ricalcolare il tempo necessario all'autobus per superare l'incrocio. In base a questi dati, i punti di inizio e di attivazione del jamiton verranno ricalcolati già in fase statica.

Dynamic Adaption

Nella fase di "Dynamic Adaption", il sistema dovrà monitorare costantemente la posizione degli autobus lungo il percorso e adattarsi dinamicamente alle variazioni del traffico e alle condizioni stradali. Il sistema monitorerà costantemente la posizione e la velocità degli autobus lungo il percorso, tenendo conto delle fermate pianificate e degli eventuali rallentamenti dovuti al carico e scarico passeggeri o ad altri impedimenti. In base a queste variazioni, anche i dati su cui è stato basato il calcolo statico potrebbero

variare, e il sistema comunicherà a tutti i veicoli non prioritari questo cambio, in modo tale che essi possano riadattarsi alla nuova situazione venutasi a creare.

Inoltre, in fase dinamica, il sistema controlla l'arrivo dei veicoli non prioritari al punto di inizio jamiton e farà entrare in azione il rallentamento solo se i veicoli non prioritari non avranno rispettato le indicazioni fornite inizialmente e che quindi dovranno essere rallentati per permettere un transito agevole e sicuro all'autobus in questione.

6.3 Il problema del Commesso Viaggiatore

Il problema della gestione del traffico per veicoli prioritari, come gli autobus, può essere efficacemente analizzato attraverso l'analogia con il Problema del Commesso Viaggiatore (TSP). Questo isomorfismo aiuta a strutturare e risolvere il problema della pianificazione ottimale dei percorsi dei veicoli per garantire il transito agevole dei veicoli prioritari attraverso punti critici, mantenendo allo stesso tempo un flusso efficiente per i veicoli non prioritari.

Il problema del commesso viaggiatore (TSP - Traveling Salesman Problem) è un classico problema di ottimizzazione combinatoria che cerca di trovare il percorso più breve possibile che visita un insieme di città esattamente una volta e ritorna alla città di partenza. Anche se i percorsi e le fermate degli autobus sono prefissati e non possono cambiare, possiamo utilizzare i principi del TSP per ottimizzare la gestione del traffico attorno agli autobus prioritari.

1. **Punti Critici come Città nel TSP:** Nei modelli di gestione del traffico, i punti critici (incroci, intersezioni) che devono essere attraversati dai veicoli (sia prioritari che non prioritari) possono essere visti come città nel TSP. Gli autobus prioritari devono attraversare questi punti critici in tempi definiti, analogamente alla necessità del commesso viaggiatore di visitare tutte le città nel percorso più breve possibile. L'obiettivo è quindi assicurare che gli autobus prioritari attraversino questi punti critici nel minor tempo possibile, minimizzando eventuali ritardi dovuti al traffico
2. **Percorsi Prefissati e Vincoli:** Nel TSP, i percorsi tra le città sono predeterminati e non possono cambiare durante il viaggio. Analogamente, i percorsi degli autobus prioritari sono fissati e non possono essere modificati per garantire che raggiungano i loro punti critici nel tempo previsto. L'obiettivo è quindi gestire il flusso di veicoli non prioritari in modo da non intralciare gli autobus durante il loro percorso prefissato.
3. **Ottimizzazione del Flusso:** Nel TSP, l'obiettivo è trovare il percorso più breve che visita tutte le città. Nella gestione del traffico, l'obiettivo è ottimizzare il flusso di traffico in modo che gli autobus possano passare attraverso i punti critici senza ritardi e in modo efficiente. L'obiettivo è quindi utilizzare strategie di gestione del traffico che regolino dinamicamente il flusso dei veicoli non prioritari per facilitare il transito degli autobus prioritari.

4. **Strategie di Priorità:** Nel TSP, ci sono strategie per determinare il percorso ottimale. Nella gestione del traffico, si sviluppano strategie per decidere quando e quanto rallentare i veicoli non prioritari per favorire il transito degli autobus. Questa analogia permette di adattare tecniche di ottimizzazione euristica alla gestione del traffico urbano.

In questo contesto, quindi, gli autobus sono i "commessi viaggiatori" e le loro fermate sono le "città" da visitare. Il nostro obiettivo non è cambiare il percorso degli autobus, ma minimizzare il tempo di percorrenza e l'interferenza con altri veicoli non prioritari. Gli obiettivi da raggiungere sono:

- Minimizzare il Tempo di Percorrenza degli Autobus: Ridurre il tempo di percorrenza degli autobus tra le fermate prestabilite.
- Ridurre l'Interferenza dei Veicoli Non Prioritari: Coordinare il traffico dei veicoli non prioritari in modo da ridurre l'interferenza con il percorso degli autobus.
- Ottimizzare il Flusso del Traffico: Gestire il traffico in modo che sia il più fluido possibile per tutti i veicoli, mantenendo la priorità per gli autobus.

Oltre a questi obiettivi, bisogna definire anche i dati su cui modellare il problema:

1. Raccogliere Dati

- Posizione delle Fermate: Coordinate geografiche delle fermate degli autobus.
- Percorsi degli Autobus: Sequenza delle fermate che ogni autobus deve seguire.
- Tempi di Arrivo Previsti: Stime dei tempi di arrivo degli autobus alle varie fermate.

2. Modellare il Problema come un Grafo

- Nodi: Rappresentano le fermate degli autobus.
- Archi: Rappresentano i percorsi tra le fermate con pesi associati ai tempi di percorrenza.

3. Applicare Tecniche di Ottimizzazione

- Modellare le Fermate come Checkpoint: Considerare ogni fermata come un checkpoint che l'autobus deve visitare in un determinato ordine.
- Utilizzare Algoritmi di Scheduling con Vincoli: Pianificare l'attraversamento dei punti critici in modo che gli autobus abbiano un transito prioritario.

Nel contesto della gestione del traffico, l'adattamento del Problema del Commesso Viaggiatore (TSP) per gli autobus si formula in questo modo: gli autobus devono visitare tutte le fermate prefissate in un ordine specifico, minimizzando i tempi di attraversamento dei punti critici e garantendo che non subiscano ritardi significativi. Questo approccio prevede l'uso di variabili di decisione che rappresentano il tempo di arrivo di ogni autobus a ciascuna fermata e punto critico. L'obiettivo è minimizzare la somma dei tempi di viaggio totali degli autobus attraverso i punti critici, ottimizzando così il transito degli autobus prioritari e migliorando l'efficienza complessiva del sistema di traffico.

Un possibile approccio per risolvere il problema della gestione del traffico per veicoli prioritari, come gli autobus, è l'uso dell'algoritmo Greedy. L'algoritmo Greedy è una tecnica di ottimizzazione che costruisce una soluzione passo dopo passo, scegliendo l'opzione migliore disponibile in ogni fase con l'obiettivo di trovare una soluzione globale ottimale. Anche se non sempre garantisce la soluzione ottimale, spesso fornisce buoni risultati in tempi rapidi.

Per applicare l'algoritmo Greedy nel contesto della gestione del traffico:

1. Ordinare i punti critici e le fermate in base alla loro posizione sul percorso dell'autobus.
2. Calcolare i tempi di arrivo stimati per gli autobus in ciascuna fermata e punto critico.
3. Regolare i tempi di arrivo dei veicoli non prioritari per evitare di interferire con il transito degli autobus.

Un altro approccio più sofisticato per risolvere il problema della gestione del traffico è l'uso della programmazione dinamica. Questo metodo consente di risolvere problemi complessi suddividendoli in sottoproblemi più semplici e risolvendo ciascuno di questi sottoproblemi una volta sola, memorizzandone i risultati.

Nel contesto della gestione del traffico per veicoli prioritari, la programmazione dinamica può essere applicata come segue:

1. Creare una tabella per memorizzare i tempi minimi di attraversamento dei punti critici.
2. Iterare attraverso le fermate degli autobus e i punti critici, aggiornando la tabella con i tempi minimi calcolati.
3. Utilizzare la tabella per determinare la sequenza ottimale dei tempi di attraversamento, minimizzando i ritardi complessivi.

L'algoritmo Greedy è particolarmente utile in situazioni in cui si devono prendere decisioni rapide e locali che portano a un miglioramento immediato, come nel caso della gestione del traffico in tempo reale. Ordinando i punti critici e le fermate, si può rapidamente stimare e regolare i tempi di arrivo, riducendo al minimo l'interferenza con il transito degli autobus prioritari. Tuttavia, è importante considerare che mentre

l'algoritmo Greedy è efficace e veloce, la programmazione dinamica offre una soluzione più accurata e ottimale, sebbene a costo di una maggiore complessità computazionale. Utilizzare entrambe le tecniche in combinazione può offrire un equilibrio tra efficienza e ottimalità nella gestione del traffico urbano.

6.4 Isomorfismo con il Trasporto Ferroviario e le Coincidenze dei Treni

Il problema della gestione del traffico per veicoli prioritari, come gli autobus, può essere visto in modo analogo al problema del trasporto ferroviario e delle coincidenze dei treni. In questo contesto, l'obiettivo principale è garantire che i treni prioritari (ad esempio, i treni ad alta velocità) attraversino determinati punti critici (stazioni o segmenti di linea) con il minimo ritardo, mentre i treni non prioritari (ad esempio, i treni regionali) devono attendere il transito dei treni prioritari per evitare interferenze.

Modello del Trasporto Ferroviario

Nel contesto del trasporto ferroviario, i treni seguono percorsi predefiniti e devono fermarsi a varie stazioni. Le coincidenze rappresentano i punti critici dove è necessario coordinare l'arrivo e la partenza dei treni per minimizzare i ritardi e garantire che i treni prioritari non siano ostacolati.

Il problema della gestione del traffico per autobus prioritari può essere isomorfo al problema delle coincidenze ferroviarie nel seguente modo:

- **Percorso dell'autobus ↔ Percorso del treno:** Entrambi seguono un percorso predefinito con fermate o stazioni specifiche.
- **Punti critici ↔ Coincidenze:** I punti critici nel percorso dell'autobus corrispondono alle coincidenze nel percorso del treno, dove i veicoli non prioritari (altri treni) devono attendere il transito dei veicoli prioritari (treni ad alta velocità).
- **Veicoli non prioritari ↔ Treni non prioritari:** I veicoli non prioritari che possono causare interferenze nel traffico stradale sono analoghi ai treni non prioritari (es. treni regionali) che devono attendere il passaggio dei treni prioritari (es. treni ad alta velocità).

Questo parallelismo spiega ancora meglio il problema da cui si parte e la soluzione a cui si vuole arrivare sfruttando le modellazioni fatte per il traffico ferroviario.

6.5 Esempio di applicazione dinamica del modello esteso

Partendo dal codice del simulatore sviluppato precedentemente [chapter 5], sono state implementate diverse funzioni che ampliassero il codice precedente per adat-

tarsi alle nuove esigenze e alle nuove osservazioni fatte. In particolare, come detto nelle precedenti sezioni, sono stati introdotti due algoritmi, l'algoritmo Greedy e la programmazione dinamica, che combinati ritornano i tempi minimi di percorrenza di un punto critico da parte del veicolo prioritario. In base a questi tempi, i veicoli non prioritari regolano la loro velocità per adattarsi alla dinamicità della situazione e non rappresentare un intralcio per il veicolo prioritario.

L'idea alla base di questi algoritmi è quella di creare un dizionario con gli orari di transito del veicolo prioritario per il punto critico (quindi l'orario in cui arriva e l'orario in cui esce dal punto critico). Dinamicamente, i veicoli non prioritari utilizzeranno questi dati per regolare la propria velocità. In prossimità dei punti critici, invece, entrerà in gioco lo stesso algoritmo implementato per il modello base, ossia una gestione dei veicoli tramite le linee di attivazione e di inizio jamiton.

Facendo in questo modo, si punta a raggiungere l'obiettivo che i veicoli non prioritari riescano a regolare la loro velocità e i loro tempi di percorrenza automaticamente senza che il sistema Jamact intervenga attivando i jamiton in prossimità degli incroci. In questo modo i veicoli non prioritari verrebbero rallentati anche in punti molto distanti dal punto critico.

L'algoritmo Greedy calcola una soluzione iniziale ordinando i punti critici e le fermate degli autobus, determinando così i tempi di arrivo e partenza stimati per ciascun punto lungo il percorso. Successivamente, l'approccio della programmazione dinamica affina questa soluzione iniziale ottimizzando i tempi di arrivo e partenza per ridurre le interferenze tra veicoli prioritari e non prioritari, aggiornando i tempi per evitare sovrapposizioni e garantire un transito agevole per i veicoli prioritari. I veicoli non prioritari inizialmente programmano il loro itinerario tenendo conto dei dati elaborati dall'algoritmo Greedy; successivamente adattano il loro percorso ai dati aggiornati dalla programmazione dinamica. Di seguito sono descritti i passaggi principali di ciascuna funzione:

```
def greedyInitialSolution(critical_points ,
                           current_time):
    initial_solution = {}

    for point in critical_points:
        arrival_time = calculateArrivalTime(point ,
                                              current_time)
        departure_time = calculateDepartureTime(
            point , arrival_time)
        initial_solution[point['id']] = (
            arrival_time , departure_time)
        current_time = departure_time

    return initial_solution
```

La funzione `greedyInitialSolution(critical_points, current_time)` implementa

un algoritmo Greedy per calcolare una soluzione iniziale. Ordina i punti critici e le fermate in base alla loro posizione lungo il percorso dell'autobus, e calcola i tempi di arrivo e partenza per ogni punto critico dello stesso. La soluzione iniziale viene memorizzata in un dizionario dove le chiavi sono gli id dei punti e i valori sono tuple di tempi di arrivo e partenza.

```
def dynamicProgrammingRefinement(vehicle, point,
    initial_solution, current_time):
    priority_arrival_time = initial_solution[point
        ['id']][0]
    priority_departure_time = initial_solution[
        point['id']][1]
    arrival_time = calculateArrivalTime(point,
        current_time)
    departure_time = calculateDepartureTime(point,
        arrival_time)

    if checkTime(priority_arrival_time,
        priority_departure_time, arrival_time,
        departure_time):
        arrival_time = max(priority_departure_time
            , arrival_time)
        departure_time = calculateDepartureTime(
            point, arrival_time)
        vehicle.speed = adjust_speed(vehicle,
            arrival_time, departure_time)

    return (arrival_time, departure_time)
```

La funzione **dynamicProgrammingRefinement(vehicle, point, initial_solution, current_time)** affina la soluzione iniziale ottenuta dall'algoritmo Greedy. Utilizza la programmazione dinamica per minimizzare le interferenze. Se c'è interferenza tra veicolo prioritario e veicolo non prioritario, aggiorna i tempi di arrivo e partenza di quest'ultimo in modo da evitare sovrapposizioni, garantendo che non intralci il veicolo prioritario. In base ai nuovi tempi di arrivo e partenza, anche la velocità del veicolo non prioritario viene aggiornata per indurre un rallentamento adeguato.

Esempio di utilizzo

Le fermate degli autobus e i punti critici sono definiti come liste di dizionari, dove ogni dizionario contiene informazioni sull'id del punto, la posizione, e il tempo entrata e di uscita da parte del veicolo prioritario. Inizialmente viene calcolata una soluzione utilizzando la funzione `greedyInitialSolution()`. Questa soluzione ordina i punti lungo il percorso e calcola i tempi di arrivo e partenza in modo rapido e approssimativo.

La soluzione iniziale viene poi raffinata, in fase dinamica, utilizzando la funzione `dynamicProgrammingRefinement()`, che riduce le interferenze tra veicoli e ottimizza i tempi di attraversamento basandosi sulle informazioni in tempo reale ricevute dal veicolo prioritario. In particolare, sulla base della sua velocità di percorrenza si vanno a modificare dinamicamente i tempi di entrata e di uscita da parte del veicolo prioritario in un punto critico. I veicoli non prioritari, quindi, riadattano la loro velocità di percorrenza, calcolando se il loro tempo di entrata e di uscita da un punto critico va ad intersecarsi con la finestra di passaggio del veicolo prioritario.

6.6 Risultati simulazioni modello Jamact per autobus

Per valutare l'efficacia dell'estensione dell'estensione applicata al modello per autobus, sono state condotte diverse simulazioni basate su un caso reale. Utilizzando gli algoritmi implementati, queste simulazioni hanno generato un set di dati che è stato successivamente analizzato. Inoltre, come per il modello base [sez. 5.3], sono stati individuati tre range di densità di traffico (basso, medio e alto) e per ogni range sono state condotte dieci simulazioni utilizzando un numero di veicoli variabile per ogni intervallo di densità: per il range di bassa densità di traffico sono stati considerati venti veicoli; per il range di media densità di traffico sono stati utilizzati sessanta veicoli; per il range di alta densità di traffico sono stati impiegati cento veicoli.

Il percorso analizzato è la tratta Fusaro-Torregaveta dell'autobus di linea 102 [35].

Numeri corse	1	Numeri corse	26
Validità	Ord.	Validità	Ord.
Bacoli - Via Fusaro	6:00	Bacoli - Via Fusaro	14:20
Bacoli - Sella Di Baia	6:03	Bacoli - Sella Di Baia	14:24
Bacoli - Piazza De Gaspari	6:05	Bacoli - Piazza De Gaspari	14:27
Bacoli - Via Risorgimento	6:12	Bacoli - Via Risorgimento	14:34
Bacoli - Centro	6:15	Bacoli - Centro	14:37
Bacoli - Via Lungolago	6:17	Bacoli - Via Lungolago	14:39
Miseno		Miseno	
Bacoli - Via Miliscola (lidi)	6:19	Bacoli - Via Miliscola (lidi)	14:39
Bacoli - Rotonda Miliscola	6:20	Bacoli - Rotonda Miliscola	14:40
Bacoli - Via Mercato di Sabato	6:23	Bacoli - Via Mercato di Sabato	14:47
Torregaveta - Cumana	6:25	Torregaveta - Cumana	14:50

Figura 6.5: Orari Autobus linea 102

Questa tratta include nove fermate (escluse le stazioni di partenza e arrivo), copre circa dodici chilometri e ha un tempo di percorrenza stimato di trenta minuti. La tabella nella figura [fig. 6.5] mostra come lo stesso autobus percorra questa tratta in due fasce orarie diverse, con tempi di percorrenza differenti. Alle 6:00 del mattino, l'autobus impiega 25 minuti viaggiando a una velocità media di 28 km/h. Alle 14:00,

invece, impiega 30 minuti con una velocità media di 25 km/h. Questa differenza è dovuta alla diversa congestione del traffico nelle due fasce orarie. L'obiettivo delle simulazioni è ridurre i tempi di percorrenza nelle fasce orarie più congestionate, avvicinandoli a quelli delle sei del mattino, quando l'autobus può viaggiare senza impedimenti, mantenendo la velocità massima consentita e considerando le fermate.

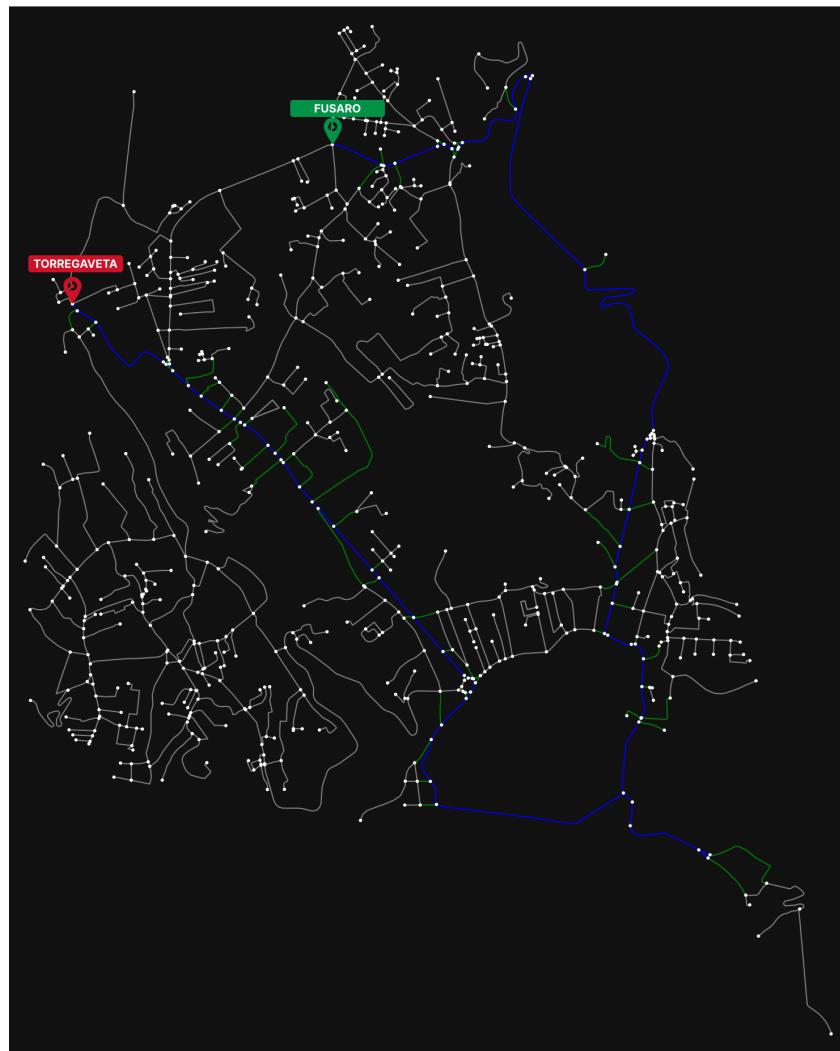


Figura 6.6: Percorso Autobus linea 102

A causa della variabilità dei tempi di fermata, non è possibile simulare perfettamente la realtà operativa di un autobus. Pertanto, le simulazioni effettuate si basano su un confronto con il caso ideale rappresentato dal tempo di percorrenza nella fascia oraria delle sei del mattino, che viene simulato con l'ipotesi ottimale di assenza di veicoli nel sistema, quindi senza impedimenti. Prendendo in considerazione questo caso base, sono state effettuate ulteriori simulazioni con presenza di traffico veicolare, applicando inizialmente il sistema Jamact e valutando i risultati ottenuti, e successivamente confrontando la stessa situazione senza il sistema Jamact attivo. L'impatto sulla percorrenza dell'autobus è stato valutato per entrambe le situazioni.

Per svolgere al meglio le simulazioni, si calcola innanzitutto il percorso che deve compiere il veicolo prioritario. Nella figura [fig. 6.6] è evidenziato in blu il percorso dell'autobus della tratta Fusaro-Torregaveta, lungo 11,95 km e con un limite di velocità prefissato di 30 km/h. A partire da questi dati, viene definito un caso ottimale che serve come riferimento per le simulazioni. Un autobus che percorre 11,95 km a una velocità costante di 30 km/h impiega circa 24 minuti. Questo tempo di percorrenza è il valore di riferimento contro cui verranno confrontati i risultati delle simulazioni.

Nella tabella [tab. 6.1] sono evidenziati i risultati ottenuti mediamente dalle varie simulazioni.

	Jamact Attivo	Jamact Non Attivo
Bassa densità di traffico		
Velocità Media veicoli non prioritari	7,8 m/s	8,1 m/s
Tempo medio percorrenza veicoli non prioritari	25 minuti e 32 secondi	24 minuti e 35 secondi
Percentuale veicoli interferenti individuati	1,5%	3,9%
Numero medio di veicoli interferenti individuati	0,3 veicoli	0,8 veicoli
Tempo medio percorrenza Autobus	24 minuti e 4 secondi	24 minuti e 27 secondi
Media densità di traffico		
Velocità Media veicoli non prioritari	7,12 m/s	7,53 m/s
Tempo medio percorrenza veicoli non prioritari	27 minuti e 59 secondi	26 minuti e 26 secondi

Tabella 6.1 – Continua alla pagina successiva

	Jamact Attivo	Jamact Non Attivo
Percentuale veicoli interferenti individuati	2,67%	6,5%
Numero medio di veicoli interferenti individuati	1,6 veicoli	3,9 veicoli
Tempo medio percorrenza Autobus	24 minuti e 48 secondi	25 minuti e 24 secondi
Alta densità di traffico		
Velocità Media veicoli non prioritari	5,1 m/s	5,8 m/s
Tempo medio percorrenza veicoli non prioritari	39 minuti e 3 secondi	34 minuti e 20 secondi
Percentuale veicoli interferenti individuati	6,6%	9,9%
Numero medio di veicoli interferenti individuati	6,6 veicoli	9,9 veicoli
Tempo medio percorrenza Autobus	25 minuti e 26 secondi	25 minuti e 38 secondi

Tabella 6.1: Risultati Simulazioni Jamact per Autobus

Bassa densità di traffico:

A bassa densità, Jamact dimostra di migliorare significativamente le prestazioni del traffico. La velocità media dei veicoli non prioritari è leggermente superiore con Jamact non attivo (8,1 m/s) rispetto quando è attivo (7,8 m/s), impattando anche sul tempo medio di percorrenza che è più breve con Jamact non attivo (24 minuti e 35 secondi) rispetto a quando è in funzione (25 minuti e 32 secondi). Inoltre con l'attivazione del sistema, la percentuale di veicoli interferenti individuati è significativamente inferiore con Jamact attivo (1,5%) rispetto al 3,9% senza. Questi dati mostrano una buona

efficienza del modello Jamact, il quale riesce a migliorare il tempo di percorrenza degli autobus mediamente di circa il 1,59%, diminuendo di circa 23 secondi il tempo totale impiegato per completare il tragitto. Infine, l'impatto del sistema sui veicoli non prioritari è di un peggioramento di circa il 3,86%, con un aumento dei tempi di percorrenza di circa 57 secondi.

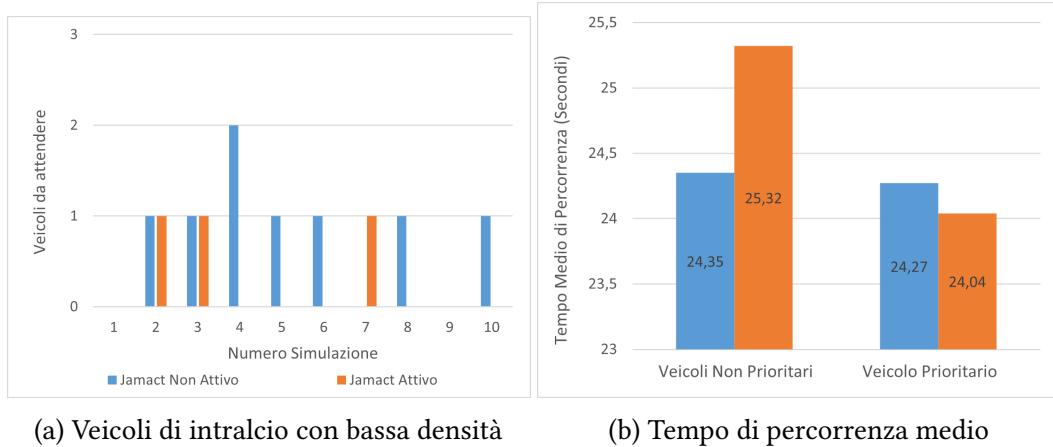


Figura 6.7: Grafici impatto Jamact bassa densità

Media densità di traffico:

Con l'aumento della densità di traffico, Jamact continua a fornire benefici, sebbene in misura minore rispetto alla bassa densità. La velocità media dei veicoli non prioritari è inferiore con Jamact attivo (7,12 m/s) rispetto all'inattività (7,53 m/s), mentre il tempo medio di percorrenza è più lungo con Jamact attivo (27 minuti e 59 secondi) rispetto all'inattività (26 minuti e 26 secondi). Tuttavia, la percentuale di veicoli interferenti individuati è notevolmente ridotta con Jamact attivo (2,67%) rispetto al 6,5% senza.

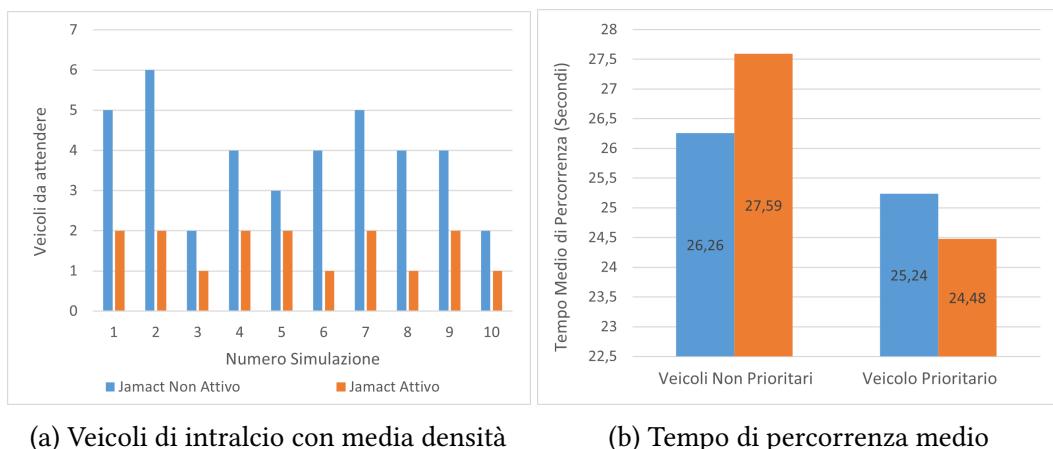


Figura 6.8: Grafici impatto Jamact media densità

Anche in questo caso i dati mostrano un'ottima efficienza del modello Jamact, il quale riesce a migliorare il tempo di percorrenza degli autobus mediamente di circa

il 2,36%, diminuendo di circa 36 secondi il tempo totale impiegato per completare il tragitto. Infine, l'impatto del sistema sui veicoli non prioritari è di un peggioramento di circa il 5,87%, con un aumento dei tempi di percorrenza di circa 93 secondi.

Alta densità di traffico:

A alta densità di traffico, nonostante le sfide più severe, Jamact continua a offrire vantaggi significativi. La velocità media dei veicoli non prioritari è inferiore con Jamact attivo (5,1 m/s) rispetto all'inattività (5,8 m/s), con un tempo medio di percorrenza più lungo con Jamact attivo (39 minuti e 3 secondi) rispetto all'inattività (34 minuti e 20 secondi). La percentuale di veicoli interferenti individuati è notevolmente inferiore con Jamact attivo (6,6%) rispetto al 9,9% senza.

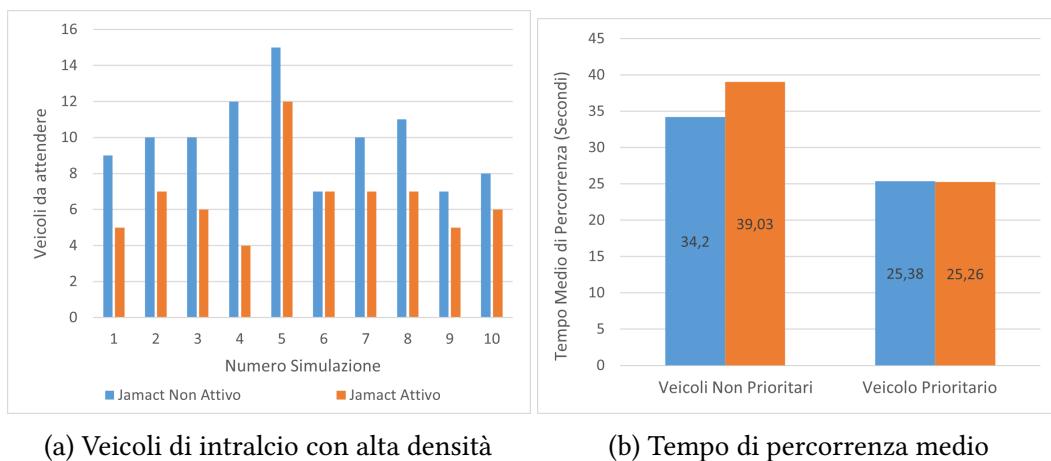


Figura 6.9: Grafici impatto Jamact alta densità

In quest'ultimo caso i dati mostrano una bassa efficienza del modello Jamact, il quale riesce a migliorare il tempo di percorrenza degli autobus mediamente di circa il 0,78%, diminuendo di circa 12 secondi il tempo totale impiegato per completare il tragitto. Infine, l'impatto del sistema sui veicoli non prioritari è di un peggioramento di circa il 13,73%, con un aumento dei tempi di percorrenza di circa 4 minuti e 43 secondi.

Risultati totali:

I dati generali raccolti dalle simulazioni effettuate hanno permesso di analizzare approfonditamente il funzionamento del modello Jamact. In particolare, l'analisi ha portato alla creazione di grafici significativi che illustrano vari aspetti del sistema. Nella figura [fig. 6.10] è evidente come il numero di veicoli interferenti sul percorso dell'autobus aumenti proporzionalmente alla densità di traffico. Questo fenomeno ha un impatto diretto sulle performance complessive del sistema, come evidenziato nel grafico delle performance in relazione alla densità di traffico nella figura [fig. 6.11]. L'analisi dei dati mostra chiaramente che all'aumentare della densità di traffico, le performance del modello tendono a diminuire, sottolineando la necessità di implementare ulteriori

strategie efficaci di gestione del traffico per mantenere elevati standard di efficienza del sistema.

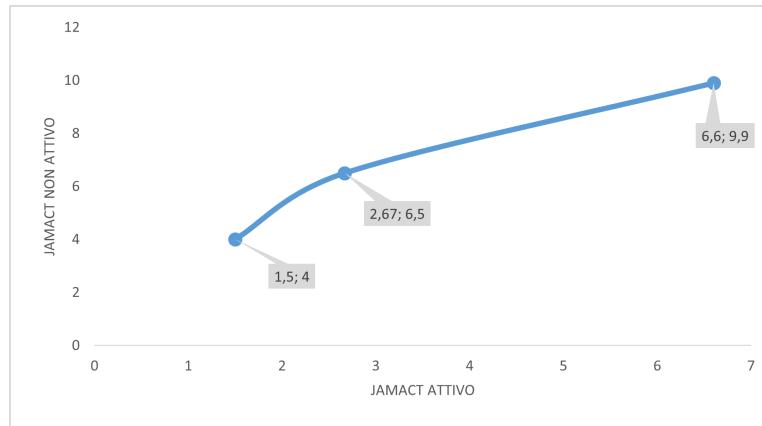


Figura 6.10: Numero medio di veicoli interferenti

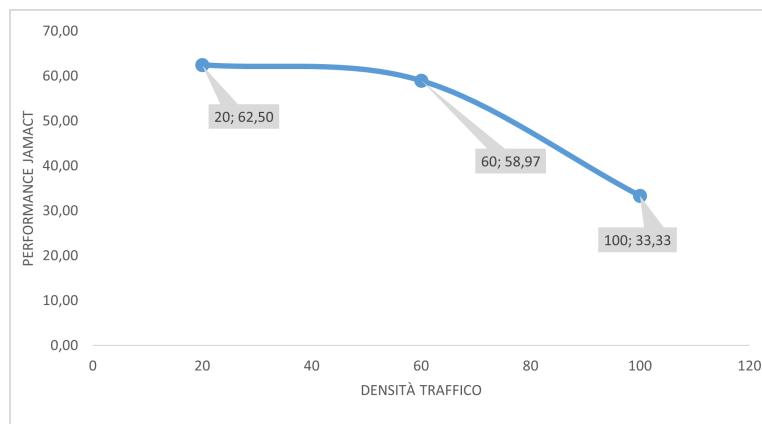


Figura 6.11: Grafico performance – densità

Oltre ai grafici sopra citati, sono stati isolati i risultati generali ottenuti dalle simulazioni per i vari range di densità di traffico individuati. La tabella [tab. 6.2] offre una panoramica dettagliata sull'impatto del sistema Jamact sulla gestione del traffico per autobus ai diversi livelli di densità. I dati raccolti mostrano chiaramente come l'efficacia di Jamact varia in base al contesto del traffico urbano.

A bassa densità di traffico, l'attivazione di Jamact ha un impatto sui veicoli non prioritari di circa 3,86%. Un dato abbastanza contenuto che però permette una gestione ottimale del traffico in condizioni meno congestionate. Inoltre, sui veicoli prioritari l'impatto è minore e si attesta intorno al 1,59%. Quindi il tempo di percorrenza per i veicoli prioritari migliora seppur di poco.

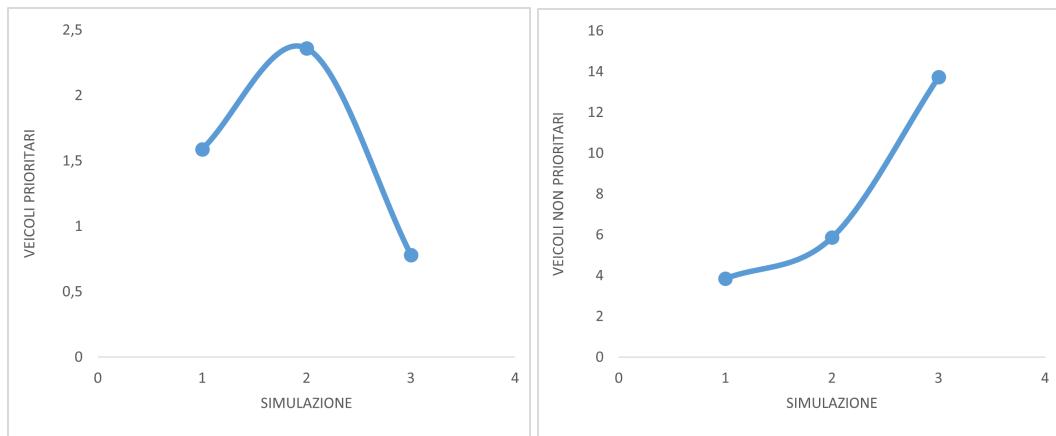
Man mano che la densità di traffico aumenta a media densità, l'impatto di Jamact aumenta leggermente, continuando a fornire benefici sostanziali. Qui, l'impatto sui veicoli non prioritari passa al 5,87%. Migliora, inoltre, anche l'impatto sui veicoli prioritari di circa il 2,36%. Questo indica che Jamact riesce ancora di più a mitigare

le interferenze, garantendo un miglioramento dei tempi di percorrenza maggiore, mantenendo una gestione efficace del traffico.

A alta densità di traffico, Jamact fa emergere un impatto non elevato. Infatti, l'impatto sui veicoli non prioritari raggiunge il 13,73%, mentre per i veicoli prioritari si riduce drasticamente allo 0,78%. Questo rappresenta comunque un miglioramento nell'efficienza complessiva del sistema, anche se evidenzia che Jamact ha meno impatto nella gestione del traffico in situazioni di alta densità, con un miglioramento meno significativo per i veicoli prioritari.

	Impatto Veicoli Prioritari	Impatto Veicoli Non Prioritari
Bassa densità di traffico	1,59%	3,86%
Media densità di traffico	2,36%	5,87%
Alta densità di traffico	0,78%	13,73%

Tabella 6.2: Impatto Jamact per Autobus per densità di traffico



(a) Impatto Jamact sui veicoli prioritari (b) Impatto Jamact sui veicoli non prioritari

Figura 6.12: Grafici impatto Jamact per Autobus

I dati raccolti nella tabella [tab. 6.2] sono evidenziati anche nei grafici della figura [fig. 6.12]. Qui si può ben notare come la curva di impatto sui veicoli prioritari abbia un picco negli scenari di traffico medio, andando a diminuire drasticamente con l'aumentare di quest'ultimo. Al contrario, l'impatto sui veicoli non prioritari mostra

una crescita costante, con un aumento significativo tra gli scenari a traffico medio e quelli ad alto traffico.

L'estensione del modello Jamact si dimostra quindi estremamente efficace nel favorire i veicoli prioritari in scenari di basso e medio traffico, permettendo loro di raggiungere la destinazione in modo più efficiente e facilitandone il transito nei punti critici. Tuttavia, negli scenari ad alta densità di traffico, il sistema mostra una diminuzione di efficienza. Nonostante questo, l'impatto complessivo del sistema sulla gestione del traffico rimane positivo, costituendo una base solida per futuri sviluppi e ottimizzazioni.

—7—

Conclusioni e sviluppi futuri

In questo capitolo saranno esposte le conclusioni e i possibili sviluppi futuri sul lavoro di tesi presentato.

Nel corso di questa tesi, è stato sviluppato e analizzato un nuovo e avanzato modello di gestione del traffico veicolare, mirato a migliorare la sicurezza e l'efficienza del transito in punti critici per i veicoli prioritari. Questo modello ha introdotto l'innovativo concetto di induzione di un rallentamento mirato, sfruttando le caratteristiche del "Phatnom Traffic Jam" a proprio favore, applicandolo esclusivamente ai veicoli non prioritari che causano intralcio. Grazie a questa tecnica di rallentamento selettivo, è stato sviluppato un modello, chiamato Jamact, che mira appunto a rallentare i veicoli non prioritari.

Il modello di base è stato poi esteso e ottimizzato per favorire un transito rapido ed efficiente a veicoli prioritari specifici, cioè gli autobus.

L'obiettivo principale è stato quello di creare un modello che non solo agevolasse il transito degli autobus, ma che incentivasse anche l'utilizzo di questi mezzi rispetto ai veicoli privati. Questo è stato conseguito inducendo rallentamenti strategici sui veicoli non prioritari, diminuendone così la competitività rispetto agli autobus che usufruiscono di un transito privilegiato.

I risultati presentati nel capitolo 6 "Modello Jamact per Autobus" evidenziano che l'applicazione precisa e controllata del modello Jamact ha portato a una importante riduzione dei tempi di percorrenza complessivi da parte degli autobus. Questo miglioramento nella velocità del servizio offerto agli autobus rappresenta un modo per incentivare gli utenti a preferire questo mezzo di trasporto rispetto all'uso dell'auto privata.

Questo miglioramento non solo incentiverebbe l'uso degli autobus, ma porterebbe anche significativi benefici in termini di riduzione della congestione stradale e dell'inquinamento atmosferico. Come discusso nei capitoli precedenti, gli autobus possono trasportare fino a 8000 persone all'ora, rispetto alle 600-1200 persone che le auto private possono trasportare nello stesso periodo. Sostituendo più di 20 automobili. Migliorare il servizio anche in termini di tempi di percorrenza rappresenta uno dei primi passi per favorire l'utilizzo dei mezzi pubblici rispetto all'auto privata.

Naturalmente, oltre a questi miglioramenti, sarebbe necessario investire nella qualità dei mezzi utilizzati e aumentare il numero di veicoli disponibili per ottimizzare l'intero sistema di trasporto pubblico, ma l'introduzione di questo sistema potrebbe portare a piccoli miglioramenti che andrebbero comunque incontro all'esigenza di aumentare l'uso dei mezzi di trasporto pubblici.

In conclusione, il modello sviluppato non solo ha dimostrato di essere efficace nel migliorare l'efficienza del trasporto pubblico, ma ha anche il potenziale per promuovere un cambiamento nei comportamenti degli utenti, favorendo una maggiore adozione dell'autobus come alternativa più conveniente e veloce rispetto all'auto privata.

Sviluppi Futuri

Il presente lavoro ha esplorato e analizzato approfonditamente le potenzialità di un nuovo modello di gestione del traffico veicolare, focalizzato sull'ottimizzazione dei percorsi per veicoli prioritari e sulla riduzione degli ostacoli per migliorare l'efficienza complessiva del sistema di trasporto urbano. Tuttavia, questo studio non è che l'inizio di un percorso verso soluzioni sempre più avanzate e integrate per affrontare problematiche relative al traffico veicolare.

Il modello Jamact si inserisce nell'ambito delle soluzioni innovative per la gestione del traffico, costituendo una base su cui costruire ulteriori sviluppi. Le potenziali direzioni di sviluppo includono una gestione più precisa e mirata dei rallentamenti indotti, per garantire un impatto minimo su tutti i veicoli presenti in strada.

Uno dei possibili sviluppi del modello potrebbe prevedere non solo il rallentamento, ma anche l'aumento di velocità per veicoli che si trovano in una posizione critica tra essere d'intralcio e non esserlo. Questa soluzione, che prevede l'induzione sia di rallentamenti che di aumenti di velocità, potrebbe rappresentare un'importante novità, permettendo di ridurre ulteriormente l'impatto del sistema su tutti gli utenti della strada.

Attualmente, il sistema Jamact è stato presentato principalmente in forma teorica, il che consente l'esplorazione di diversi approcci che, partendo dalle basi del modello, potrebbero svilupparlo in modi differenti. Durante la ricerca non sono stati individuati approcci simili sia al tema del traffico fantasma sia alla gestione della priorità veicolare, rendendo questo modello un innovativo punto di partenza che potrebbe aprire nuove prospettive nella ricerca in questo campo.

Nonostante ciò, è stato sviluppato un simulatore di base per il sistema presentato. Raffinando e dettagliando ulteriormente questo simulatore, si potrebbe creare una solida piattaforma di partenza per sviluppare un sistema più complesso e realistico, utilizzabile nelle Smart Cities.

Per il futuro, gli aspetti da implementare sono numerosi: dall'introduzione di vari tipi di approcci al rallentamento dei veicoli, allo sviluppo di un sistema operativo nella realtà. L'obiettivo finale è disporre di dati sempre più accurati e dettagliati, al fine di rendere il modello Jamact un reale strumento di gestione del traffico, efficiente, sicuro ed estendibile.

-A-

A.1 Moto Uniformemente Accelerato: Concetto e Applicazioni

Il moto uniformemente accelerato è un concetto fondamentale nella descrizione del movimento di un oggetto soggetto a un'accelerazione costante. Questo tipo di moto trova applicazione in una vasta gamma di scenari fisici, dalla cinetica dei corpi in caduta libera alla dinamica dei veicoli in movimento.

Una delle principali utilità del moto uniformemente accelerato è la sua capacità di calcolare varie grandezze cinematiche in un sistema in cui l'accelerazione è costante nel tempo. Utilizzando le equazioni del moto uniformemente accelerato, è possibile determinare velocità, posizioni e tempi relativi al moto di un oggetto in accelerazione.

In particolare, le equazioni del moto uniformemente accelerato permettono di calcolare:

1. **Velocità finale e iniziale:** Determinare la velocità di un oggetto in movimento a un dato istante, conoscendo la sua velocità iniziale, l'accelerazione applicata e il tempo trascorso.
2. **Variazione di posizione:** Calcolare la variazione di posizione di un oggetto in movimento, partendo dalla sua posizione iniziale, la velocità iniziale, l'accelerazione e il tempo trascorso.
3. **Tempo di percorrenza:** Calcolare il tempo impiegato da un oggetto per raggiungere una determinata velocità finale, partendo da una velocità iniziale e sottoposto a un'accelerazione costante.

Questi calcoli consentono la previsione e l'ottimizzazione del movimento di oggetti in accelerazione.

A.2 Equazioni del moto uniformemente accelerato

Le equazioni del moto uniformemente accelerato descrivono il movimento di un oggetto sotto un'accelerazione costante. Sono le seguenti:

- **Prima equazione del moto:**

$$v = v_0 + at \quad (\text{A.1})$$

dove:

- v è la velocità finale,
- v_0 è la velocità iniziale,
- a è l'accelerazione,
- t è il tempo.

- **Seconda equazione del moto:**

$$s = s_0 + v_0 t + \frac{1}{2} a t^2 \quad (\text{A.2})$$

dove:

- s è la posizione finale,
- s_0 è la posizione iniziale,
- v_0 è la velocità iniziale,
- a è l'accelerazione,
- t è il tempo.

- **Terza equazione del moto:**

$$v^2 = v_0^2 + 2a(s - s_0) \quad (\text{A.3})$$

dove:

- v è la velocità finale,
- v_0 è la velocità iniziale,
- a è l'accelerazione,
- s è la posizione finale,
- s_0 è la posizione iniziale.

Queste equazioni sono fondamentali per descrivere il moto di un oggetto quando l'accelerazione è costante e sono utilizzate in una vasta gamma di applicazioni fisiche.

A.3 Moto Uniformemente Accelerato nel Calcolo della Decelerazione

Quando si calcola il tempo necessario per decelerare da una certa velocità iniziale a una certa velocità finale, si applica il concetto di moto uniformemente accelerato. In questo contesto, l'oggetto sottoposto a decelerazione si muove con un'accelerazione costante diretta in senso opposto al moto.

Utilizzando le equazioni del moto uniformemente accelerato, è possibile determinare il tempo impiegato per la decelerazione. In particolare, si utilizza l'equazione:

$$t = \frac{v_f - v_0}{a} \quad (\text{A.4})$$

dove:

- t è il tempo impiegato per decelerare,
- v_0 è la velocità iniziale,
- v_f è la velocità finale, e
- a è l'accelerazione (che è negativa nel caso della decelerazione).

Applicando questa equazione, è possibile calcolare il tempo necessario per passare da una certa velocità iniziale a una certa velocità finale sotto l'effetto di una decelerazione costante. Questo concetto è fondamentale per la risoluzione di problemi relativi al movimento di oggetti soggetti a decelerazione uniforme.

-B-

B.1 Grafi e Tipologie di Grafi Utilizzati

I grafi sono uno strumento di modellazione molto importante, infatti, la soluzione di molti problemi reali può essere ricondotta alla soluzione di opportuni problemi su grafi. Un grafo G è una coppia ordinata (V, E) , dove V è un insieme di nodi chiamati vertici, con $V \neq \emptyset$, ed $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ è un insieme di sottoinsiemi di cardinalità due di V (o insieme di coppie di vertici distinti), detto insieme degli archi.

I grafi possono esprimere due tipi di relazioni:

- Simmetrica: è una relazione binaria R su un insieme A tale che, per ogni coppia di elementi a e b in A , se a è in relazione con b (ossia aRb), allora anche b è in relazione con a (ossia bRa). Formalmente, una relazione simmetrica è definita nel seguente modo:

$$\forall a, b \in A, (aRb \implies bRa)$$

- Antisimmetrica: è una relazione binaria R su un insieme A tale che, per ogni coppia di elementi a e b in A , se a è in relazione con b (ossia aRb) e b è in relazione con a (ossia bRa), allora a deve essere uguale a b . Formalmente è indicata come:

$$\forall a, b \in A, (aRb \wedge bRa \implies a = b)$$

Le relazioni citate possono poi essere rappresentate sotto forma di:

- **Grafi orientati**: gli archi hanno una direzione, rappresentando relazioni assimmetriche.
- **Grafi non orientati**: gli archi non hanno direzione, rappresentando relazioni simmetriche.

In alcune situazioni è utile associare un peso o costo c agli archi di un grafo, tramite una funzione di costo:

$$c : E \rightarrow \mathbb{R}$$

Se un arco ha una un peso allora il grafo viene detto pesato e se tra due vertici non esiste un arco diremo che il costo è infinito.

B.2 Libreria Python OSMnx

La libreria osmnx in Python è uno strumento potente per scaricare e analizzare i dati geografici da OpenStreetMap (OSM). In particolare, osmnx consente di creare grafi delle reti stradali, ciclabili e pedonali, utilizzando i dati OSM.

Per rappresentare le reti stradali, osmnx utilizza principalmente grafi orientati poiché le strade possono avere sensi unici e direzioni specifiche di percorrenza. Ogni nodo nel grafo rappresenta un'intersezione o un punto terminale di una strada, mentre ogni arco rappresenta un segmento stradale tra due nodi.

Il processo di creazione di un grafo della mappa con osmnx può essere riassunto nei seguenti passaggi:

1. **Download dei dati:** osmnx scarica i dati OSM per un'area geografica specifica utilizzando l'API di OSM. È possibile specificare l'area utilizzando un nome di luogo, coordinate geografiche o poligoni.
2. **Creazione del grafo:** I dati scaricati vengono trasformati in un grafo. osmnx identifica i nodi (intersezioni) e gli archi (strade) dai dati OSM e costruisce un grafo utilizzando la libreria networkx. Ogni arco nel grafo contiene attributi come la lunghezza della strada, il tipo di strada, i limiti di velocità, ecc.
3. **Pulizia e semplificazione:** Il grafo può essere pulito e semplificato per rimuovere nodi e archi ridondanti o non necessari, riducendo la complessità del grafo e migliorando l'efficienza delle analisi successive.
4. **Visualizzazione e analisi:** Il grafo risultante può essere visualizzato utilizzando funzioni di plotting integrate in osmnx, e può essere analizzato per vari scopi, come calcolare percorsi ottimali, analizzare la connettività della rete stradale, ecc.



Figura B.1: Esempio di applicazione della libreria osmnx.

Bibliografia

- [1] Eurostat. *Luxembourg leads cars per inhabitant list in the EU*. 2021. URL: <https://ec.europa.eu/eurostat/web/products-eurostat-news/-/edn-20210922-1>.
- [2] National Association of City Transportation Officials. *Move That Bus: Ensuring Transit Priority at Traffic Signals*. 2022. URL: <https://nacto.org/wp-content/uploads/2022/08/MoveThatBus-FINAL.pdf>.
- [3] AGI. *Tempo perso nel traffico: quanto costa e come migliorarlo*. 2019. URL: https://www.agi.it/data-journalism/news/2019-08-26/tempo_perso_traffico-6074418/.
- [4] Kyoto Club. *Libro della Mobilità 2019*. 2019. URL: https://www.kyotoclub.org/mediabinary/LibroMOB2019_digital_sm.pdf.
- [5] Repubblica. *Non sprecare gli autobus: tutto da rifare*. 2021. URL: https://www.repubblica.it/green-and-blue/2021/01/08/news/non_sprecare_autobus_tutto_da_rifare-280962191/.
- [6] Moveo. *Mezzi pubblici in Italia: uno sguardo al trasporto locale*. 2023. URL: <https://moveo.telepass.com/mezzi-pubblici-italia/#:~:text=Il%20ricorso%20al%20trasporto%20pubblico,700.000%20di%20Sud%20e%20isole..>
- [7] M Shahid Anwer e Chris Guy. «A survey of VANET technologies». In: *Journal of Emerging Trends in Computing and Information Sciences* 5.9 (2014), pp. 661–671.
- [8] Michael Lee e Travis Atkison. «VANET applications: Past, present, and future». In: *Vehicular Communications* 28 (2021), p. 100310.
- [9] ICT Security Magazine. *Reti Mobili Ad Hoc - Parte 1: Architettura*. ICT Security Magazine. 2018. URL: <https://www.ictsecuritymagazine.com/articoli/reti-mobili-ad-hoc-parte-architettura/>.
- [10] ResearchGate. *Time Barrier-Based Emergency Message Dissemination in Vehicular Ad-hoc Networks - Scientific Figure*. 2019. URL: https://www.researchgate.net/figure/VANET-communication-architecture_fig3_330772224.
- [11] Bruccoleri Angelo. *Analisi di sicurezza di servizi sviluppati su reti Vanet*. 2017. URL: <https://webthesis.biblio.polito.it/6579/1/tesi.pdf>.

- [12] Giuseppe Diliberto. *Propagazione delle informazioni nelle VANET tramite Population Protocol*. 2021. URL: <https://sites.unipa.it/networks/ndslab/pdf/thesis/diliberto.pdf>.
- [13] Khadige Abboud, Hassan Aboubakr Omar e Weihua Zhuang. «Interworking of DSRC and Cellular Network Technologies for V2X Communications: A Survey». In: *IEEE Transactions on Vehicular Technology* 65.12 (2016), pp. 9457–9470. doi: 10.1109/TVT.2016.2591558.
- [14] ChuanTao Yin et al. «A literature survey on smart cities». In: *Science China. Information Sciences* 58.10 (2015), pp. 1–18.
- [15] Osservatori.net. *Smart City: come funziona la città intelligente*. <https://blog.osservatori.net/smart-city-come-funziona-citta-intelligente>. 2024.
- [16] Il Sole 24 Ore. *Città sempre più smart e connesse*. <https://www.ilsole24ore.com/art/citta-sempre-piu-smart-e-connesse-AFjLF1PB>. 2023.
- [17] Aliga. *What the Heck is a Smart City?* Aliga. URL: <https://aliga.sk/en/what-the-heck-is-a-smart-city/>.
- [18] SAE International. *SAE International Releases Updated Visual Chart for its "Levels of Driving Automation" Standard for Self-Driving Vehicles*. SAE International. 2018. URL: <https://www.sae.org/news/press-room/2018/12/sae-international-releases-updated-visual-chart-for-its-%E2%80%9Clevels-of-driving-automation%E2%80%9D-standard-for-self-driving-vehicles>.
- [19] Morris R Flynn et al. «Traffic modeling-Phantom traffic jams and traveling jamitons». In: *Traffic* 8.29 (2009), p. 2016.
- [20] Octo Telematics. *Phantom Traffic Jams: What They Are and How to Avoid Them*. 2023. URL: <https://www.octotelematics.com/blog/phantom-traffic-jams/>.
- [21] Morris R. Flynn et al. «Traffic Modeling - Phantom Traffic Jams and Traveling Jamitons». In: *Journal Name Volume.Number* (2010), Pagine. doi: DOI. URL: <https://math.mit.edu/traffic/>.
- [22] Massachusetts Institute of Technology (MIT). *The Physics of Traffic Jams*. YouTube video. Accessed: 2024-05-23. 2008. URL: <https://www.youtube.com/watch?v=Q78Kb4uLAdA>.
- [23] Klaas O. Koerten, David. A. Abbink e Arkady Zgonnikov. «Haptic Shared Control for Dissipating Phantom Traffic Jams». In: *IEEE Transactions on Human-Machine Systems* 54.1 (2024), pp. 11–20. doi: 10.1109/THMS.2023.3315519.
- [24] L. J. J. Wismans, L. C. W. Suijs, L. Krol e E. C. van Berkum. «In-Car Advice to Reduce Negative Effects of Phantom Traffic Jams». In: *Transportation Research Record* 2489.1 (2015), pp. 1–10. doi: 10.3141/2489-01. URL: <https://doi.org/10.3141/2489-01>.

- [25] Laura Smith. *Why Phantom Traffic Jams Happen*. URL: <https://www.livescience.com/61862-why-phantom-traffic-jams-happen.html>.
- [26] La Legge per Tutti. *Qual è il limite minimo di velocità in strada?* https://www.laleggepertutti.it/672572_qual-e-il-limite-minimo-di-velocita. 2024.
- [27] Allianz Direct. *Limits di velocità in Italia su Autostrade e Strade (Extra) Urbane*. 2024. URL: <https://www.allianzdirect.it/blog/limiti-velocita-italia/>.
- [28] Francesco Saffioti. «Design and Performance Evaluation of a Control Algorithm for a Robotic Arm». Accessed: 2024-05-21. Tesi di dott. Politecnico di Milano, 2016. URL: https://www.politesi.polimi.it/retrieve/a81cb05b-eded-616b-e053-1605fe0a889a/2016_07_Saffioti.pdf.
- [29] Walter Balzano, Erasmo Prosciutto, Biagio Scotto di Covella e Silvia Stranieri. «A Resource Allocation Technique for VANETs Inspired to the Banker's Algorithm». In: *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*. A cura di Leonard Barolli. Cham: Springer International Publishing, 2023, pp. 222–231. ISBN: 978-3-031-19945-5.
- [30] ISFORT. *Rapporto sulla mobilità 2022*. Rapp. tecn. ISFORT, 2022. URL: https://www.isfort.it/wp-content/uploads/2023/01/221215_RapportoMobilita2022_Def-1.pdf.
- [31] ISTAT. *La mobilità sostenibile in Italia: Indagine sulle principali caratteristiche della mobilità urbana*. Rapp. tecn. ISTAT, 2018. URL: <https://www.istat.it/it/files/2018/11/Report-mobilit%C3%A0-sostenibile.pdf>.
- [32] ISFORT. *Rapporto sulla mobilità 2023*. Rapp. tecn. ISFORT, 2023. URL: https://www.isfort.it/wp-content/uploads/2023/12/RapportoMobilita2023_Def.pdf.
- [33] Carlo Berruti. «Trasporto pubblico locale: 6 condizioni per il salto di qualità». In: (2023). URL: <https://www.econopoly.ilsole24ore.com/2023/02/13/trasporto-pubblico-locale-condizioni/>.
- [34] ANSA. *Dossier: Il trasporto delle persone in Italia*. Rapp. tecn. ANSA, 2021. URL: https://www.ansa.it/documents/1640275471033_Dossier_Trasport_Persone_FINAL.pdf.
- [35] Ente Autonomo Volturno. *Linea 102 Fusaro-Torregaveta*. 2024. URL: https://www.eavsrl.it/web/sites/default/files/eavbus/102_FUSARO-TORREGAVETA.pdf.

Ringraziamenti

Il raggiungimento di questo traguardo e la realizzazione di questa tesi è stata resa possibile grazie al sostegno e all'incoraggiamento di numerose persone, a cui desidero esprimere la mia più sincera gratitudine.

Innanzitutto, desidero ringraziare il mio relatore, il prof. Walter Balzano. Grazie a lui ho avuto la possibilità di conoscere l'ambiente universitario sotto un altro punto di vista, riuscendo a pubblicare lavori di ricerca che hanno arricchito non solo il mio bagaglio di esperienze ma che mi hanno permesso di conoscere altri aspetti dell'informatica. Grazie alla sua fiducia e ai suoi consigli, sono riuscito a completare un lavoro di tesi che mi rende orgoglioso e soddisfatto.

Un sentito ringraziamento va anche ad Erasmo, con il quale ho lavorato alla stesura di parte di questa tesi. La nostra collaborazione ha permesso di creare da zero un lavoro molto interessante. La sua presenza è stata essenziale non solo come collega eccezionale, ma anche come grande amico con cui condividere il futuro professionale e personale.

Desidero ringraziare anche Cristian, con il quale ho condiviso fin dall'inizio la passione per il Napoli e l'entusiasmo per l'ideazione di progetti innovativi. La nostra amicizia, nata durante gli anni universitari, è destinata a durare nel tempo e ad andare anche oltre l'università.

Ringrazio inoltre tutti coloro che ho incontrato in questo percorso, come Gianmarco, Antonio, Oleksander, Marco e tanti altri. Da ognuno ho imparato qualcosa che mi ha arricchito e migliorato, contribuendo alla mia crescita personale e professionale.

Un ringraziamento di cuore va alla mia famiglia, in particolare ai miei genitori, per il loro sostengo costante, e per non avermi mai messo pressione sui risultati universitari. Senza di loro, questo traguardo non sarebbe stato possibile e non avrebbe avuto lo stesso valore. La loro fiducia in me è stata per me molto importante.

Ringrazio inoltre le mie sorelle e il mio "fratello aggiuntivo" che sono stati, sono e saranno sempre un esempio per me. Hanno rappresentato una spalla su cui contare, offrendo supporto e ispirazione in ogni momento.

Ringrazio anche tutti i miei amici con cui ho condiviso parte della mia vita dagli anni del liceo fino ad oggi.

Infine, un ringraziamento speciale va ad Alessia, con cui ho condiviso tante emozioni e gioie durante questo percorso. Il suo sostegno continuo e incondizionato è stato fondamentale. So che anche per te arriveranno presto grandi traguardi e successi, e non vedo l'ora di vederti festeggiarli.

Dedico inoltre tutto questo lavoro a chi il 24 giugno alle 14.56, nascendo, mi ha regalato una grande emozione. Spero che un giorno anche tu possa essere orgogliosa di me.

A tutti voi, il mio più sentito grazie.

Biagio