

Tesi di Dottorato

SIMULAZIONE DI GUIDA E MICROSIMULAZIONE  
DEL TRAFFICO: SVILUPPO E VALIDAZIONE DI UN  
AMBIENTE INTEGRATO

**Dottorato di Ricerca**

*Ingegneria dei Sistemi Idraulici, di Trasporto e  
Territoriali*

**Indirizzo:** *Infrastrutture Viarie e Sistemi di Trasporto*

**XX Ciclo**

Coordinatore: Prof. Bruno Montella

TUTOR:

PROF. VINCENZO TORRIERI

COTUTOR:

ING. VINCENZO PUNZO

DOTTORANDO:

ING. BIAGIO CIUFFO



*Alla mia dolce Mary*

*Alla mia cara famiglia*



*<<Siamo come nani sulle spalle dei giganti,  
sì che possiamo vedere più cose di loro e più lontane,  
non per l'acutezza della nostra vista,  
ma perché sostenuti e portati in alto dalla statura dei giganti>>*

Bernardo de Chartres, XII sec.



## Premessa

Il concetto della ricerca, a mio modo di vedere è splendidamente concentrato nell'aforisma riportato all'inizio del testo ad opera del filosofo e grammatico francese del XII sec., Bernardo De Chartres. La figura del "nano" sulle "spalle del gigante" lascia spazio a due immagini simboliche: chiunque può dare il suo apporto all'arricchimento della conoscenza generale a patto che il suo lavoro si basi sulla conoscenza già acquisite. La ricerca e la conoscenza in genere non sono e non possono essere, quindi, legate alla grandezza di poche menti illuminate, ma strettamente legate alla condivisione e alla comunicazione delle idee e delle esperienze. La ricerca che si conduce in contesti che per diversi motivi, tendono ad isolarsi e a lavorare per conto proprio, cercando sempre di ripartire dall'origine dei problemi senza far tesoro delle esperienze condotte all'esterno e senza permettere agli altri di giovare delle proprie, è sterile e fallimentare in partenza. L'amore per la scoperta, per l'innovazione, per la sfida, non nasce d'altronde proprio dalla voglia di mettere a disposizione della società o della comunità scientifica i risultati condotti attraverso la propria severa e faticosa esperienza?

Nodo cruciale del ragionamento è, però, l'ipotesi che ciò che si comunica abbia davvero il carattere di arricchimento della conoscenza. Il problema, quindi, sorge nel momento in cui la smania di comunicare prevarica il valore di ciò che si comunica. La ricerca a questo punto diventa sterile e collassa su se stessa.

Nel campo dell'Ingegneria dei Trasporti (non oso generalizzare vista la mia brevissima esperienza nel campo scientifico), a volte, leggendo gli innumerevoli lavori che ogni anno si trovano pubblicati, si ha la sensazione che quanto ci si trova innanzi non aggiunga realmente valore allo stato dell'arte della tematica che si sta affrontando, ma piuttosto che mostri gli stessi risultati già raggiunti conditi da una diversa salsa o osservati da una differente angolazione. La mancanza quasi totale di lavori metodologici in cui si cerca di mostrare la capacità di una procedura incentrandosi sulla validità concettuale delle basi su cui si fonda piuttosto che esclusivamente sui risultati ottenuti e sulla rapidità con cui sono pervenuti ne è la prova più clamorosa. Una branca così giovane ha infatti bisogno di solidificare le sue fondamenta molto di più di ridipingere la sua facciata.

La convinzione di ciò, dentro di me, rappresenta, a mio modesto modo di vedere, il frutto principale della mia esperienza di dottorato di ricerca. Essa si è evoluta durante i tre anni trascorsi presso il Dipartimento di Ingegneria dei Trasporti "L. Tocchetti" dell'Università "Federico II" di Napoli sotto la guida attenta del Prof. Vincenzo Torrieri e del Prof. Vincenzo Punzo. Durante tali anni, infatti, il mio interesse e le mie attività hanno spaziato (seppur con gradi di approfondimento non sempre adeguati) tra le diverse sfere che abbracciano la disciplina: dalla progettazione stradale, all'analisi della domanda di trasporto (statica e dinamica), dalla rappresentazione del sistema dei trasporti per la simulazione di politiche intervento a scala macroscopica alla modellizzazione microscopica del comportamento veicolare, passando per l'analisi di dati di varia origine e natura, fino ad arrivare allo studio del comportamento di guida mediante simulatore di guida e veicolo strumentato. Tale diversificazione delle mie attività ha avuto



origine principalmente dagli stimoli che i miei mentori hanno voluto e saputo offrirmi. In aggiunta, la capacità di questi di pormi sempre degli angoli di osservazione differenti con i quali analizzare un qualsiasi problema ha continuamente consentito dentro di me di ricostruire l'hegeliano processo evolutivo della conoscenza in cui la sintesi finale può avvenire solo attraverso l'attenta disamina della tesi di partenza e della sua antitesi.

Tra i vari problemi affrontati, quello che più di tutti ha fornito degli spunti di ricerca rilevanti è stato quello della calibrazione di un modello di simulazione microscopica del traffico. Tali modelli, infatti, pur avendo ormai raggiunto una diffusione rilevante sia nel campo della ricerca che nel campo della pratica professionale, mancano di procedure e metodologie in grado di calibrarli in modo che riescano a riprodurre condizioni reali. Come risultato tali modelli vengono ampiamente utilizzati in tutto il mondo per simulare l'effetto dell'attuazione di politiche sul sistema dei trasporti senza avere una coscienza sufficiente sulle loro potenzialità. Nel frattempo i risultati prodotti dalla ricerca del settore sull'argomento mostrano risultati dai quali l'entità del problema è solo marginalmente portata alla luce. In questo caso la ricerca manca della comunicazione necessaria ed i risultati adottati finiscono per lo più per aggiungere solo materiale sterile alla conoscenza.

E questo non è un caso isolato all'interno dell'Ingegneria dei Trasporti, ma dalla condivisione delle esperienze condotte dai miei colleghi del Dipartimento e di altri incontrati in vari occasioni sia in contesti nazionali che internazionali, la sensazione che una parte dei lavori disponibili si basi su ipotesi e considerazioni non sufficientemente approfondite è piuttosto

diffusa. E se le gambe del “gigante” non sono sufficientemente solide, anche i “nani” sulle sue spalle rischiano di cadere miseramente.

Per questo motivo durante le attività svolte, grande cura è stata volta ad analizzare i problemi presentatisi nella maniera più completa e rigorosa possibile in modo che il contributo fornito, seppur piccolo, fosse fondato su base solide e stabili su cui basare analisi successive.

Per fare ciò è stato ancora necessario il supporto delle mie due guide, per il loro apporto scientifico, per il loro supporto morale, per avermi saputo caricare nei momenti più delicati, per aver sempre saputo smorzare i miei toni quando fosse stato necessario e per aver fornito una guida anche morale alla mia vita di questi anni.

Voglio inoltre cogliere l’occasione per ringraziare la mia dolce compagna di stanza e di vita Mary, che mi dà la forza di gettare sempre il cuore oltre gli ostacoli che ci si presentano innanzi e mi sprona a stare sempre sveglio per rinnegare le viziate prassi che accompagnano lo svolgimento di molte delle attività che si svolgono in questa parte d’Italia.

Un ringraziamento va inoltre al Prof. Andrea Papola per l’affetto che ha mostrato nei nostri confronti in questi anni e per i preziosi suggerimenti che mi ha fornito durante la sua breve ma intensa esperienza di controrelatore del mio lavoro.

Un saluto ed un abbraccio vanno quindi a tutte le persone che ho frequentato in questi anni di lavoro universitario. In particolare un saluto speciale ai miei “*fidanzatini*” Davide e Antonino, alla mia “*pazza*” conterranea Roberta, al mio *guru spirituale* Salvatore (basta solo che non parli di cinema), al mio amico “*chiattillo*” Valerio, ai miei punti di riferimento scientifici Fulvio e Vittorio.

L'esperienza portata avanti, nel bene e nel male, avrà senza dubbio influenza su tutta la mia vita ed è questo il motivo per il quale a prescindere dalle strade che il mio destino mi porterà ad imboccare, non sarò mai pentito di aver votato questi tre anni alla ricerca e all'attività universitaria.

Biagio Ciuffo



## **Abstract**

I simulatori di guida sono strumenti sempre più utilizzati per condurre esperimenti riguardanti le interazioni tra il veicolo, il guidatore e l'ambiente stradale. La qualità dei risultati ottenibili dagli esperimenti dipendono dalla capacità dell'ambiente simulato di fornire al guidatore degli stimoli (di varia natura) quanto più prossimi a quelli che riceverebbe nella vita reale. Uno degli elementi della simulazione aventi un ruolo fondamentale per una realistica rappresentazione delle condizioni di guida è il comportamento dei veicoli di contorno. Gli sforzi compiuti in questo settore sono stati quindi principalmente centrati a fornire ai cosiddetti veicoli autonomi un comportamento visivamente accettabile più che fisicamente realistico. Se questo è comunque un aspetto importante all'interno di un ambiente di simulazione di guida, non soddisfa appieno le esigenze degli utilizzatori. Il traffico simulato, oltre a mostrare un naturale movimento dei veicoli, deve, infatti, garantire anche la consistenza del deflusso della corrente veicolare in cui il veicolo interattivo è immerso. Inoltre le esigenze sperimentali attualmente esistenti richiedono un grado di approfondimento ancora maggiore. Lo studio di strumenti di aiuto alla guida (ADAS), di sistemi di informazione all'utenza o dei sistemi cooperativi, richiede sempre più la modellizzazione del traffico sull'intera rete di trasporto.

L'idea che si è quindi portata avanti durante l'attività di dottorato è stata quella di integrare l'ambiente di simulazione di guida con un software di simulazione microscopica del traffico. L'integrazione fa sì che il software di microsimulazione scelto, nello specifico, AIMSUN NG, si occupi di

simulare il movimento di tutti i veicoli presenti su una rete, inviando alla simulazione di guida i dati riguardanti esclusivamente i veicoli intorno al veicolo interattivo in linea con le possibilità offerte dai modelli di traffico attualmente esistenti.

Questo tipo di strategia apre alle possibilità effettive di utilizzare il simulatore di guida e i modelli di simulazione microscopica del traffico per il loro mutuo sviluppo: in questo senso il comportamento di guida del veicolo interattivo verrebbe ad essere usato per sviluppare e calibrare modelli comportamentali per il movimento dei veicoli autonomi, mentre il comportamento di questi ultimi potrebbe essere impiegato per validare il comportamento del veicolo interattivo.

Un aspetto importante della fase di validazione della struttura ha riguardato quindi l'analisi di AIMSUN e della sua capacità di riprodurre reali condizioni di deflusso veicolare. Per questo motivo il software è stato inserito all'interno di una procedura di ottimizzazione globale incaricata della ricerca dei valori da attribuire ai parametri del modello in grado di minimizzare la distanza tra gli output dello stesso e i dati reali a disposizione.

L'integrazione è stata quindi progettata utilizzando le possibilità di comunicazione offerte dai due ambienti, e la sua realizzazione è stata effettuata e risulta funzionante. La versione del software di simulazione microscopica del traffico utilizzata, però, non è ancora adatta a svolgere il ruolo cui è chiamata. Il lavoro è stato comunque portato avanti avendo avuto modo di testare gli sviluppi futuri del software che consentiranno in breve tempo di superare le difficoltà esistenti ad oggi.

## Abstract

*PhD Candidate: Biagio Ciuffo*





# ***INDICE***

Premessa.....	vii
Abstract.....	xiii
1. Introduzione.....	7
1.1 Background.....	7
1.2 Finalità del lavoro.....	11
1.3 Limiti applicativi.....	14
1.4 Thesis outline.....	14
2. L'utilizzo del simulatore di guida.....	17
2.1 La simulazione di guida – Cenni storici.....	17
2.2 Architettura dei simulatori di guida.....	19
2.2.1. Il sistema visivo.....	20
2.2.2. Il sistema sonoro.....	22
2.2.3. Il ritorno di forza.....	23
2.2.4. Il sistema dinamico.....	23
2.2.5. Ritorno di forza ai comandi.....	26
2.2.6. Il modello del veicolo.....	26
2.2.7. Il modello di traffico.....	27
2.2.8. Controllo dello scenario.....	27
2.3 Alcune osservazioni su i simulatori di guida.....	28
2.4 Malessere da simulatore.....	28
2.5 Validazione.....	29
2.6 Alcuni simulatori di guida esistenti.....	30
2.6.1. Simulatore Daimler_Chrysler.....	31

2.6.2.	NADS .....	32
2.6.3.	Il simulatore dinamico della Renault .....	34
2.6.4.	VIRTTEX.....	35
2.7	Il simulatore di TEST Scarl.....	36
2.7.1.	Il sistema visuale.....	37
2.7.2.	Cabina .....	38
2.7.3.	Il sistema di moto.....	38
2.7.4.	Il sistema dinamico .....	39
2.8	L'utilizzo Dei Simulatori Di Guida Per La Sicurezza Stradale – Lo Stato Dell'arte.....	40
2.8.1.	Introduzione .....	40
2.8.2.	Panoramica delle esperienze condotte in letteratura .....	42
2.8.3.	La letteratura di recente produzione sulle possibili applicazioni dei simulatori di guida.....	54
3.	La modellizzazione del traffico nei simulatori di guida. Stato dell'arte	59
3.1	Il modello di traffico del simulatore di guida del CRdCT .....	59
3.2	La simulazione dei veicoli autonomi – Background.....	61
3.2.1.	Esempi di moduli per la gestione del traffico creati ad hoc.	64
3.2.2.	Integrazione con software di simulazione del traffico esistenti	67
3.3	Conclusioni .....	77
4.	Analisi delle strategie di integrazione.....	81
4.1	Introduzione .....	81
4.2	Le API di un software .....	82
4.3	Le API dei due strumenti .....	85

4.3.1.	Le API di SCANeR.....	85
4.3.2.	Le Api di AIMSUN.....	89
5.	Progettazione e realizzazione della comunicazione tra AIMSUN e SCANeR.....	93
5.1	Introduzione .....	93
5.2	Le funzioni disponibili nelle API utilizzate nell'integrazione .....	95
5.3	I temi affrontati per l'integrazione .....	97
5.3.1.	Accurata corrispondenza tra le reti stradali dei due ambienti 97	
5.3.2.	Simulazione del traffico i tempo reale .....	98
5.3.3.	Sincronizzazione dei moduli di guida e di traffico .....	99
5.3.4.	Procedura di scambio delle informazioni in tempo reale.....	99
5.3.5.	Consistenza della frequenza di aggiornamento dei dati.....	100
5.3.6.	Gestione della visualizzazione dei veicoli autonomi .....	100
5.4	L'architettura software .....	101
5.4.1.	L'ambiente condiviso.....	101
5.5	Prima applicazione .....	105
5.5.1.	Descrizione del test .....	105
5.5.2.	Problemi riscontrati.....	106
5.6	Conclusioni .....	107
6.	Validazione dell'ambiente integrato .....	109
6.1	Introduzione .....	109
6.2	Approccio alla calibrazione di un modello di simulazione microscopica del traffico.....	110
6.3	Stato dell'arte .....	113
6.3.1.	Il concetto generale di calibrazione.....	113

6.3.2.	I modelli dinamici di simulazione del traffico .....	115
6.3.3.	La calibrazione dei modelli di simulazione microscopica del traffico – breve stato dell’arte .....	120
6.4	Studio di campo eseguito .....	126
6.4.1.	Introduzione all’ambito sperimentale .....	126
6.4.2.	Analisi del modello di microsimulazione utilizzato.....	130
6.5	Analisi di sensitività.....	136
6.5.1.	Metodologia .....	137
6.5.2.	Progettazione degli esperimenti .....	138
6.5.3.	Implementazione e risultati .....	142
6.6	Calibrazione del modello .....	150
6.6.1.	Implementazione della procedura di calibrazione.....	153
6.6.2.	Verifica della procedura – validazione .....	155
6.7	Conclusioni .....	162
6.8	Ulteriori sviluppi .....	165
7.	Conclusioni del lavoro svolto .....	166
8.	Bibliografia .....	170
	Appendice A – Pubblicazioni .....	178





# 1. Introduzione

## 1.1 Background

La sicurezza stradale è un tema di grande interesse internazionale. Il costo sociale degli incidenti che avvengono continuamente sulle strade è elevatissimo. Per far capire l'entità di tale problema basti riflettere sui due dati: in Italia il costo complessivo dovuto ad incidenti stradali si aggira intorno ai 23 Miliardi di € annui [1] e che secondo alcune stime nel 2020 l'incidentalità stradale sarà la terza causa di morti nel mondo.

Molti degli incidenti sono causati dal fallimento dell'interazione tra il guidatore, il veicolo e le condizioni del sistema stradale (inteso come insieme di vari fattori quali stato dell'infrastruttura, condizioni di traffico etc.). Per questo, motivi ingenti risorse sono state impiegate e numerosi sforzi sono stati condotti nella ricerca applicata alla realizzazione di strumenti di supporto alla guida. Come risultato, in pochi anni, gli utenti della strada saranno dotati di sofisticati strumenti quali ad esempio gli intelligent transportation systems (ITS), gli advanced driver assistance systems (ADAS), gli in-vehicle information systems (IVIS). L'effetto auspicato della diffusione di tutti questi strumenti è la realizzazione di un sistema di trasporto più pulito, più sicuro e più efficiente. Ciò però dipenderà molto dall'effettivo utilizzo che gli utenti che se ne saranno dotati ne faranno. Da uno studio effettuato utilizzando un simulatore di guida [1] ad esempio,

risulta che esiste una richiesta da parte dell'utenza di sistemi di ausilio alla guida in caso di traffico intenso (Congestion Assistant Systems). Nello studio viene però mostrato come l'effettivo utilizzo di questi sistemi da parte dell'utenza dipenda dal tipo di aiuto che lo strumento fornisce. In particolare si è visto che l'utente accetta molto volentieri avere informazioni dinamiche ad esempio sull'entità della coda cui ci si sta avvicinando, mentre accetta con molta più fatica quegli strumenti che forniscono un supporto attivo quale ad esempio le limitazioni sull'utilizzo dell'acceleratore in condizioni di traffico intenso.

L'accettabilità di questi strumenti da parte dell'utenza è un fattore quindi da analizzare attentamente per non fornire all'utenza solo dei gadget tecnologici di scarsa utilità.

Nel caso in cui, quindi, questi strumenti venissero effettivamente ad essere introdotti sul mercato, in pochi anni il guidatore si vedrà ad interagire, durante la guida, oltre che con il veicolo e con le condizioni del sistema stradale, anche con vari strumenti di aiuto alla guida e di intrattenimento (oltre a quelli con i quali già interagiscono quali telefoni cellulari, navigatori satellitari, etc.). Tutte queste interazioni influiscono notevolmente, nel bene e nel male, sul comportamento dei guidatori e sulla loro abilità di guida. Analizzarne gli effetti sulla sicurezza stradale diviene quindi un tema di grande interesse scientifico.

Esperimenti a riguardo possono essere realizzati osservando il comportamento dei soggetti testati immersi nel traffico reale, o su circuiti di guida realizzati appositamente oppure mediante l'uso di simulatori di guida. Le sperimentazioni nel mondo reale sono senz'altro quelle da cui si può ottenere una risposta più realistica dal soggetto che viene testato. D'altro



canto, però, tali sperimentazioni sono anche quelle da cui risulta più complicato ottenere conclusioni statisticamente accettabili vista l'aleatorietà legata alle condizioni meteorologiche, della strada e del traffico. Senza considerare poi i rischi che il soggetto testato naturalmente corre e naturalmente crea nel momento in cui si trovi immerso in un traffico veicolare reale.

Le piste realizzate ad hoc offrono un'ambientazione più sicura e la possibilità di garantire una certa ripetitività agli esperimenti. Il realismo che attraverso di esse è possibile fornire alla sperimentazione è, invece, piuttosto basso, visti i costi necessari per l'allestimento.

I simulatori di guida, invece, offrono sempre la perfetta riproducibilità degli esperimenti da condurre, la massima sicurezza, offrono la possibilità di realizzare scenari assolutamente realistici e di testare strumenti ancora non esistenti ma di cui si conoscano le logiche e gli effetti sulla guida. D'altro canto, però, hanno i problemi del realismo delle risposte che l'utente testato riceve dall'ambiente circostante che possono provocare nello stesso un senso di disorientamento a livello inconscio che si può trasformare in vero e proprio malessere da guida (quello che in letteratura è definito *motion sickness* [1]). Inoltre la consapevolezza, da parte del soggetto esaminato, di essere in un ambiente simulato influenza notevolmente il suo comportamento.

Il simulatore di guida in dotazione al Centro Regionale di Competenza Trasporti (CRdCT) offre la straordinaria occasione di affrontare queste problematiche.

Appare evidente che la grande potenzialità teorica dei simulatori di guida diventa effettivamente pratica se i problemi cui si è fatto cenno in precedenza vengono ridimensionati in modo da non fornire false sensazioni all'utente che si sta testando. In particolare l'immersione a livello inconscio dell'utente nell'ambiente simulato avviene se gli stimoli **visivi** e **sonori** che egli riceve dall'ambiente e dagli oggetti che lo circondano sono prossimi a quelli che si aspetterebbe nella realtà e se le **azioni** che sperimenta nella realtà vengono riprodotte con sufficiente accuratezza.



**Figura 1—1 Il simulatore di guida del Centro Regionale di Competenza Trasporti[60]**

Sebbene lo stato dell'arte sull'argomento sia denso di esperienze e tentativi condotti, l'ambiente di simulazione è talmente vasto e complesso che un simulatore in grado di riprodurre fedelmente le condizioni di guida reali non esiste (e forse non esisterà mai). La crescente diffusione di questo strumento unita al costante sviluppo tecnologico lasciano comunque pensare che nel

prossimo futuro gli ampi margini di miglioramento di cui questi strumenti sono dotati saranno, almeno in parte, colmati.

## **1.2 Finalità del lavoro**

Il presente lavoro cerca di fornire un contributo allo stato dell'arte di settore per quel che riguarda la modellizzazione del traffico in cui è immerso l'utente alla guida di un simulatore di guida (il comportamento cioè di quelli che nel prosieguo saranno definiti *veicoli autonomi*). Gli aspetti che si vogliono affrontare sono molteplici: (i) il primo e più immediato è legato alla rappresentazione visuale del traffico di contorno: i veicoli autonomi devono *muoversi in maniera credibile* ovvero mostrando realistiche dinamiche di interazione con gli altri veicoli. Ciò è molto importante, perché influenza notevolmente l'impegno dell'utente alla guida del simulatore di guida (alla guida, cioè, di quello che d'ora in avanti chiameremo *veicolo interattivo*) e quindi la sua abilità, soprattutto quando si affrontino studi che riguardano la progettazione stradale, l'uso di sistemi di aiuto alla guida e l'effetto della fatica o di sostanze eccitanti, che sono notoriamente molto influenzati dall'intensità e dalla composizione del traffico veicolare. (ii) Oltre a quello del singolo veicolo, è inoltre importante che sia realistico il comportamento della corrente veicolare nel suo complesso. Un banale esempio sta nel fatto che un utente alla guida del veicolo interattivo deve vedere decrescere la velocità della corrente veicolare al crescere del numero dei veicoli autonomi che lo circondano. Un altro elemento importante che fino ad oggi è stato trascurato è la possibilità di avere all'interno dell'ambiente simulato una corretta rappresentazione del traffico a livello aggregato ossia avere un comportamento di tutti i veicoli della rete in

grado di fornire una *modellizzazione accettabile dei flussi* che gravano sul sistema. Questo fattore è molto importante per garantire che un utente che si muove in uno scenario simulato trovi le condizioni di traffico che attenderebbe di trovare in quello reale. Quest'aspetto risulta particolarmente interessante allorquando si vogliono eseguire esperimenti su sistemi di informazioni all'utenza (ITS, IVIS) di cui si voglia simultaneamente investigare tanto l'efficacia, quanto l'appetibilità da parte dell'utenza.

L'approccio generalmente utilizzato tra gli studiosi del settore consiste nel creare un modulo specifico che vada ad interfacciarsi con l'ambiente di simulazione di guida e che si occupi della rappresentazione del comportamento visivo dei veicoli autonomi. La logica seguita è quella di considerare i veicoli autonomi alla stregua di qualsiasi altro oggetto che circonda il veicolo interattivo. Per questo motivo ad essere modellizzati sono solo i veicoli autonomi che si trovano all'interno della distanza di visibilità dal veicolo interattivo che occupa sempre il centro della sfera di simulazione. L'interesse ad una modellizzazione del traffico globale che grava su un'intera rete, attualmente non esiste.

Questa logica appare tanto più evidente se si pensa considera che i moduli di gestione del traffico presenti nei pacchetti software commerciali di simulazione di guida non consentono la gestione di più di cento veicoli autonomi, cifra che è almeno di un ordine di grandezza inferiore a quella che sarebbe necessaria per gestire una rete di trasporto. D'altra parte, però, la domanda, da parte delle strutture di ricerca che utilizzano simulatori di guida, di architetture in grado di ospitare moduli per la gestione del traffico ad una scala più ampia è ancora molto limitata, cosicché le aziende che si

occupano della produzione e dello sviluppo dei simulatori non hanno un interesse pratico ad affrontare tale problema.

L'idea alla base del presente lavoro per superare tale problema è quella di integrare l'ambiente del simulatore di guida con uno dei software di simulazione microscopica del traffico presenti in commercio, in maniera tale che questi si occupi sia della corretta modellizzazione del comportamento dei veicoli e sia della loro gestione complessiva. Ad ogni passo di calcolo questi dovrebbe fornire all'ambiente della simulazione di guida, le informazioni sulle caratteristiche cinematiche dei soli veicoli che si muovono intorno a quello interattivo. L'idea segue alcuni rudimentali tentativi che sono stati portati avanti a livello internazionale e che sembra abbiano sortito degli effetti incoraggianti.

Utilizzare questo tipo di approccio nella modellizzazione del traffico della simulazione di guida, consente, tra le altre cose, di aprire le porte alla completa riproduzione del modello di offerta del sistema dei trasporti che è già stata affrontata nei più comuni pacchetti software di simulazione microscopica del traffico. In questo modo, ad esempio, si potrebbe pensare di inserire nell'ambiente di guida simulato, particolarizzazioni del modello di offerta quali ad esempio, coordinazioni semaforiche, sistemi di trasporto collettivo superficiali su sede promiscua, ramp metering, presenza di corsie riservate ad HOV e quant'altro già presente nei più diffusi pacchetti presenti in commercio.

Lo scopo del lavoro è quindi quello di individuare un software di microsimulazione adatto a compiere l'integrazione, analizzarlo e realizzare la struttura di integrazione.

Come sarà mostrato nel prosieguo, non è stato possibile completare l'ultimo punto a causa di intoppi di carattere informatici che sono sopraggiunti.

### **1.3 Limiti applicativi**

Ogni software di simulazione microscopica del traffico ha il proprio specifico campo di applicazione. Questo sarà definito dai particolari modelli da cui è costituito. A seconda del campo applicativo per cui questi modelli sono stati ricavati il software presenterà dei limiti applicativi negli altri ambiti. Questo non vuol dire che un software i cui modelli sono stati ricavati per la rappresentazione dell'ambito autostradale non possa essere utilizzato anche in ambito urbano, ma certamente la sua modellizzazione sarà meno raffinata e per un'applicazione così delicata come quella dei simulatori di guida, ciò deve essere tenuto in stretta considerazione.

Alcuni dei software di simulazione microscopica del traffico più avanzati, in linea con le esigenze di ricercatori e professionisti, offrono ampie possibilità di espansione. Attraverso API e particolari kit di sviluppo è possibile sostituire o customizzare interi pezzi del codice di simulazione esistente. In questo modo è possibile, certamente non a costo zero e con gli opportuni accorgimenti, adattare il software esistente a condizioni per cui non sarebbe predisposto.

Questa possibilità è tenuta in fortissima considerazione all'atto della scelta del microsimulatore da utilizzare per l'integrazione.

### **1.4 Thesis outline**

Partendo dalle premesse esposte, il documento si sviluppa nel seguente modo: nel Capitolo 2 sarà presentato uno stato dell'arte sull'utilizzo dei simulatori di guida in campo internazionale per porre maggiormente in evidenza l'esigenza di una corretta rappresentazione del traffico veicolare all'interno degli scenari di simulazione e sarà descritto brevemente il simulatore di guida del CRdCTras. Nel capitolo 3 invece sarà approfondito la descrizione dei modelli che regolano la rappresentazione del traffico nei simulatori di guida esistenti. Nel Capitolo 4 saranno affrontate nello specifico le problematiche metodologiche inerenti l'integrazione con un modello di simulazione microscopica del traffico. Nel Capitolo 5 sarà dunque esposto il progetto del protocollo di comunicazione necessario per l'integrazione e la realizzazione della struttura integrata. Nel Capitolo 6 sarà mostrata la validazione condotta sulla struttura sia dal punto di vista della corrispondenza tra le reti stradali rappresentate, sia dal punto di vista della frequenza di calcolo e soprattutto per quanto riguarda la capacità del micro-simulatore di traffico di riprodurre le reali condizioni del deflusso stradale. Infine nel Capitolo 7 saranno riportate le conclusioni del lavoro. La bibliografia e le appendici con i codici di calcolo elaborati e con i lavori scientifici pubblicati concludono il lavoro.





## **2. L'utilizzo del simulatore di guida**

### **2.1 La simulazione di guida – Cenni storici**

I simulatori di guida sono stati sviluppati, a partire dal lontano 1910, allo scopo di verificare l'abilità e la competenza degli operatori di mezzi pubblici. Nelle quattro decadi successive, mockup di automobili sono state dotate di apparecchiature per testare le risposte dei guidatori a vari stimoli. In alcuni casi, venivano anche mostrate scene di guida riprodotte in maniera meccanica o immagini filmate. Tuttavia, nonostante un crescente interesse, i limiti tecnologici esistenti (potenza di calcolo e display video) ne frenavano fortemente sia la diffusione che lo sviluppo.

Alla fine degli anni '60, la tecnologia sviluppata dalla "National Aeronautics and Space Administration" (NASA) per supportare i suoi programmi spaziali diede un nuovo e potente impulso al riaffermarsi delle tecniche di simulazione di guida. Diversi simulatori di guida divennero così operativi in tutto il mondo.

Nei primi anni 70, la General Motors e il Virginia Polytechnic Institute and State University realizzarono un lavoro pionieristico sulla simulazione di guida "human-in-loop". Questo lavoro, che continuò per un ventennio, venne implementato usando il simulatore VPI-SU. Nella configurazione più avanzata, il VPI-SU presentava un modello di veicolo a 16 gradi di libertà con una piccola piattaforma mobile.

Nel 1983, la Federal Highway Administration inizia a servirsi dell'Highway Driving Simulator (HYSIM). Questo simulatore a base-fissa è stato utilizzato principalmente per studi sui fattori umani (strumenti di controllo del traffico, percezione dei rischi da parte del guidatore, sistemi intelligenti veicolo-infrastruttura, identificazione delle manovre pericolose). La Figura 2—1 mostra il laboratorio originale con un sistema di proiezione montato sulla cabina del veicolo.

Un simulatore di guida con un complesso sistema di movimento venne inaugurato a metà degli anni 80. Si trattava del simulatore di guida del VTI (Figura 2—1), sviluppato dal Swedish National Road and Transport Research Institute a Linköping. Il simulatore era in grado di fornire un ampio movimento laterale e dei moti di rollio e di beccheggio, inoltre, una tavola vibrante simulava i moti ad alta frequenza.



**Figura 2—1** *Il laboratorio originale HYSIM (sx) e il simulatore di guida del VTI (dx)*

Nel 1985, a Berlino, la Daimler-Benz rese operativo il suo simulatore a 6 gradi di libertà. Una grande quantità di denaro venne spesa per la sua realizzazione e ciò si dimostrò un ottimo investimento. Il simulatore si muoveva con accuratezza ed agevolmente con un attrito molto basso. Nel

1994, il simulatore venne ricostruito allo scopo di realizzare lo stesso ampio movimento laterale del simulatore del VTI.

Tra la fine del 1980 e gli inizi del 1990, molte aziende automobilistiche iniziano a sviluppare i loro propri simulatori di guida. Per esempio, il Ford Research Laboratory realizzò un simulatore di guida a base fissa, che era già pronto all'uso nel 1991. Renault e Chrysler svilupparono contemporaneamente simulatori statici. In Italia, il Centro Ricerche Fiat e la SEPA collaborano alla realizzazione di un simulatore statico dotato di un ampio campo di visione. Anche, tra le università e gli istituti di ricerca divenne comune dotarsi di simulatori di guida.

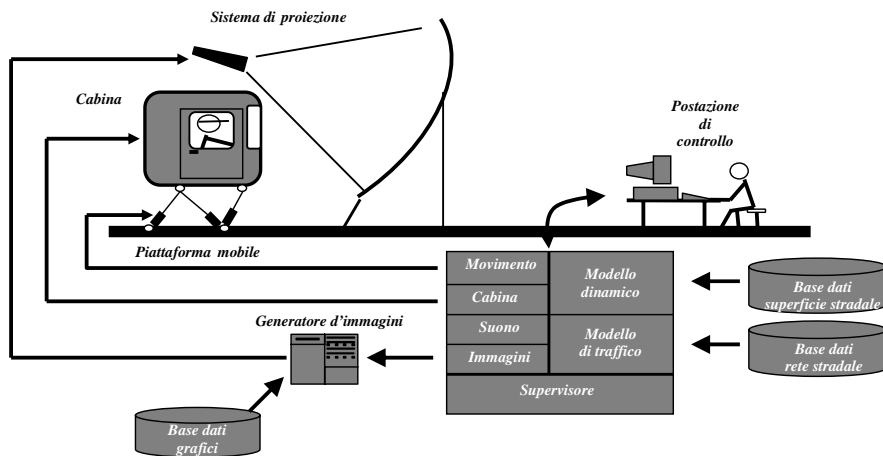
Nell'ultimo decennio, si sta assistendo alla nascita di una nuova generazione di simulatori di guida avanzati, accompagnata dal diffondersi di simulatori meno avanzati privi di sistemi di movimento. Due simulatori molto avanzati sono il NADS, sviluppato dal National Highway Traffic Safety Administration, ed il VIRTTEX al Ford Research Laboratory. Entrambi i simulatori presentano un largo dome mobile. Il NADS è caratterizzato da un enorme sistema di movimento per spostamenti laterali e longitudinali che consente le migliori performance attualmente disponibili. Tra questi sistemi ad elevate prestazioni può essere ascritto il simulatore di guida di TEST Scarl, operante dal 2006 presso l'Istituto Motori del CNR di Napoli, in ragione della sua piattaforma mobile a 6 gradi di libertà e del suo ampio campo di visione.

## **2.2 Architettura dei simulatori di guida**

I simulatori di guida sono composti da diversi sotto-sistemi differenti (Figura 2—2): video, audio, ritorno di forza, modello del veicolo e scenario.

Tutti questi sotto-sistemi operano insieme per creare l'illusione che il guidatore stia davvero controllando un veicolo reale. E' fondamentale che tali sotto-sistemi siano sincronizzati e che abbiano un ritardo (dal azione del guidatore alla reazione del simulatore) minimo di 40-60 ms. Altrimenti, si potrebbe perder l'illusione e diventerebbe elevata la probabilità di disturbare il conducente (mal di testa, confusione, stordimento).

Ovviamente, nel design di un simulatore è fondamentale la conoscenza del tipo di applicazioni in cui lo stesso verrà impiegato.



**Figura 2—2 Architettura tipo di un simulatore di guida avanzato**

### **2.2.1. Il sistema visivo**

Il sistema visivo fornisce tutte le informazioni visive necessarie al guidatore. Condurre un veicolo è un compito essenzialmente visivo, per questo motivo, è vitale che le informazioni visive restituite siano ottimali.

La produzione in tempo reale di immagini dotate di un buon realismo dipende da diversi fattori:

- Transport delay
- Frame rate

- Dimensione dello schermo
- Risoluzione
- Acuità visiva

I ritardi nella restituzione grafica sono spesso suddivisi in: transport delay e frame rate time. Il transport delay è il tempo intercorrente tra l'istante in cui il modello tridimensionale inizia ad essere calcolato a quello in cui l'immagine è pronta per essere visualizzata. Il ritardo totale in un simulatore di guida dovrebbe essere inferiore a circa 40-60 ms. Il frame rate time rappresenta il tempo tra la visualizzazione di due immagini successive. Per un livello di interattività accettabile, il frame rate non dovrebbe scendere al di sotto dei 30 Hz; a 60 Hz si ottengono immagini estremamente fluide. Un ritardo grafico comporta un rischio elevato che il guidatore si ammali.

La risoluzione necessaria dipende da diversi fattori, come la distanza dallo schermo. Diversi test hanno stabilito che un oggetto deve essere almeno 3 arcmin per essere intravisto da un occhio umano. Ciò significa 20 pixels per grado, quindi con uno schermo di 120° avremo 2400 pixels (su due canali/proiettori). Una risoluzione minore può essere adottata per gli schermi laterali e posteriori.

Il campo visivo diretto è di circa 60° x 30°. Alcuni esperti lo ritengono sufficiente, ma la gran parte della sensazione della velocità è situata agli estremi di tale campo di visione; ciò significa che uno schermo di almeno 120° x 30° è necessario (Figura 2—3). Per creare uno schermo di tali dimensioni, sono richiesti diversi proiettori.

È importante che la scena grafica venga processata dal punto di vista del guidatore, altrimenti egli non vedrà ciò che si aspetta. I retrovisori non sono

fondamentali ma possono diventare necessari in alcuni tipi di test (ad esempio: interazione con altri veicoli).

Esistono altre tecniche per la visualizzazione dell'immagini, ad esempio gli head mounted displays ed i sistemi di retroproiezione (i proiettori posizionati dietro gli schermi vi proiettano, tramite specchi concavi, le immagini). Nessuna di queste tecniche viene utilizzata nei simulatori di alto livello. Gli head mounted display visualizzano le immagini su due schermi molto prossimi agli occhi: tali schermi creano problemi in ragione della loro bassa risoluzione e dei grandi ritardi. La retro-proiezione presenta l'inconveniente che l'immagini possono essere osservate solo da un area ristretta.



**Figura 2—3 Sistema visivo a proiezione frontale**

### **2.2.2. Il sistema sonoro**

Pochi soldi e sforzi vengono dedicati al sistema sonoro, poiché esso non fornisce al guidatore molte informazioni dirette su ciò che sta accadendo. È comunque un elemento essenziale, senza il quale l'illusione non sarebbe così buona.

Durante la simulazione, il sistema sonoro riproduce i suoni che si possono udire all'interno di un veicolo. Il suono dominante in un veicolo è tra i 20-500 Hz, ma alcune tonalità e rumori provenienti dai pneumatici, dalla strada, ecc, si manifestano nell'area delle alte frequenze. Di conseguenza, il sistema sonoro deve coprire alcune migliaia di Hertz.

Per ottenere un'illusione realistica, è importante generare una spazializzazione 3D nel suono, così che sia possibile avvertire la direzione della sorgente sonora. Ovviamente, la direzione dei suoni e l'informazione visiva devono essere in accordo o ci sarà un considerevole rischio di disorientare il guidatore. Molti suoni dipendono dalla velocità del veicolo (vento / irregolarità stradale) e dai giri del motore.

Il metodo più pratico di ricostruire il suono, percepito all'interno di un veicolo, consiste nel servirsi di una libreria di campioni sonori del motore, del vento, o delle irregolarità stradali, ecc; tali campioni vengono poi mixati in tempo reale in modo da ottenere una ottima illusione sonora.

### **2.2.3. Il ritorno di forza**

Il ritorno di forza al guidatore può essere diviso in due parti: le forze che si manifestano attraverso il cambio di moto del veicolo e le forze restituite dai comandi del veicolo come il volante, i pedali, etc.

### **2.2.4. Il sistema dinamico**

Il compito del sistema di moto è di riprodurre tutte le forze a cui un guidatore viene esposto a seguito del movimento del veicolo.

I valori reali delle accelerazioni risultanti da tale moto dipendono non solo dal modello del veicolo ma anche dal comportamento di guida e dalle condizioni stradali. Alcuni studi hanno messo in luce che le accelerazioni

lineari (in x,y,z) sono dominanti tra 0-1 Hz con un piccolo picco secondario di circa 2-3 Hz per l'accelerazione verticale. Ciò si verifica a seguito della connessione diretta tra l'azione del guidatore ed i movimenti del veicolo mentre il picco delle accelerazioni verticali è dovuto alle sospensioni. Per le accelerazioni angolari, si è osservato che i moti di rollio e di beccheggio sono prevalentemente ad alta frequenza mentre l'imbardata a bassa frequenza. Tale risultato viene giustificato dal fatto che l'imbardata è generalmente il risultato dall'azione di chi guida mentre il rollio ed il beccheggio sono dovuti più alle vibrazioni che ad una azione diretta. Il moto angolare di rollio e beccheggio è limitato meccanicamente in un veicolo standard a circa  $\pm 6^\circ$ , a causa della corsa limitata delle sospensioni.

Costruire un sistema in grado di simulare ciò in maniera efficace richiede l'uso di due differenti sotto-sistemi: uno per le alte e l'altro per le basse frequenze. Per le alte frequenze, un piano vibrante è il modo più efficace per fornire a chi guida la sensazione della velocità del veicolo attraverso vibrazioni provenienti dalla strada, dal vento e così via.

Il livello di simulazione delle basse frequenze dipende fortemente dai tipi di test a cui il simulatore verrà destinato. Per studi comportamentali in condizioni di guida normali (es.: misura del grado di distrazione dovuta ai telefoni cellulari), si può utilizzare solamente un piano vibrante. In realtà, alcuni esperti ritengono che un simulatore a base-fissa può essere considerato sufficiente per molti test in cui le manovre del veicolo non producono accelerazioni superiori ai 0,3g. Quando si vuole testare la reazione del guidatore, ad esempio in una situazione critica dove azioni estreme sono necessarie per evitare una collisione, la percezione fisica delle



forze diventa il fulcro del test, quindi, un qualche tipo di sistema di moto a bassa frequenza risulta necessario.

Due differenti tecniche sono correntemente usate per simulare le accelerazioni a bassa frequenza. La prima utilizza un exapod (Figura 2—4) dove le accelerazioni di breve periodo vengono restituite tramite il movimento lineare e le accelerazioni di lungo termine inclinando il guidatore e usando la forza gravità per ottenere la forza richiesta (ad esempio, in Figura 2—4, un inclinazione di  $11^\circ$  simula un'accelerazione di  $0,2g$ ). Il secondo si serve di lunghi movimenti lineari e, con questa tecnica, le accelerazioni vengono direttamente simulate.



**Figura 2—4 Esempio di Hexapod (DRL) (sinistra) - Principio di inclinazione per l'accelerazione (destra)**

Molti simulatori di guida usano un fattore di scala allo scopo di ridurre la domanda dei sistemi di moto. Studi hanno dimostrato che il più basso fattore di scale realistico è di  $0,35$  ( $1g$  diventa  $0,35g$ ). Diversi simulatori di alto-livello usano un fattore di scala pari a  $0,5$  o più.

### **2.2.5. Ritorno di forza ai comandi**

Gran parte della sensazione di guida dipende dalle forze restituite ai comandi del veicolo, ad esempio il volante, i pedali e la leva del cambio. Il ritorno di forza al volante dovrebbe essere presente in ogni simulatore, poiché minimizza la varianza nella sterzata.

La tecnica usuale per creare il ritorno di forza al volante consiste nel connettere un servomotore alla colonna dello sterzo: questa soluzione è tanto flessibile quanto economica. Risulta comunque difficile ottenere un buon modello di sistema sterzante, poiché esso dipende da diversi fattori, quali le forze agli pneumatici e la reale dinamica del volante. Il sistema deve anche possedere una elevata frequenza di aggiornamento ( $> 100$  Hz), altrimenti perderà in precisione.

Una soluzione pratica per riprodurre un ritorno di forza ai pedali è quella di servirsi dei sistemi meccanici esistenti: il sistema di frenatura di un veicolo reale può essere usato in un simulatore, in tal modo si ottiene un ritorno di forza reale.

### **2.2.6. Il modello del veicolo**

In un simulatore di guida il veicolo, i suoi componenti meccanici e il loro comportamento sono descritti sotto forma di modelli computerizzati. Tali modelli simulano le dinamiche del veicolo in dettaglio ed con un intervallo di aggiornamento elevato. Il modello, come minimo, descrive una drive-line ed un sistema sterzante, per consentire al guidatore di muoversi in avanti e controllare la sua direzione nel mondo simulato.

Anche in questo caso, il livello di dettaglio del modello dipende dal tipo di esperimenti che si vogliono condurre. Se il simulatore viene utilizzato per

applicazioni di sviluppo del veicolo, dove i parametri veicolari assumono una grande rilevanza, allora, è necessario un modello particolarmente complesso.

Considerare tutti i parametri importanti, può voler dire servirsi di un modello a 15 gradi di libertà: 6 sono per la massa, 4 per la velocità delle ruote, 4 per la corsa delle sospensioni e l'ultimo per la cremagliera dello sterzo.

### **2.2.7. Il modello di traffico**

Il modello di traffico del simulatore di guida viene utilizzato per gestire il movimento dei veicoli autonomi, le loro mutue interazioni e le loro interazioni con il veicolo interattivo. Malgrado le condizioni del traffico giochino un ruolo importante nella maggior parte degli studi che si portano avanti con il simulatore, non tutti i driving simulator sono dotati di tale modulo.

Del modo con cui il movimento dei veicoli venga simulato nei simulatori di guida si parlerà estesamente nel prossimo capitolo.

### **2.2.8. Controllo dello scenario**

Quando si conducono dei test al simulatore di guida è importante avere il controllo di tutti gli oggetti (tutti i parametri) all'interno dello scenario. In questo modo diventa possibile costruire scenari che saranno gli stessi per tutti i soggetti testati.

Questa è una delle chiavi per progettare esperimenti di buona qualità al simulatore di guida. Senza la possibilità di controllare l'intero ambiente, molti dei grandi vantaggi dei simulatori di guida verrebbero oscurati.

### **2.3 Alcune osservazioni su i simulatori di guida**

Il progetto di un simulatore è un compromesso tra prestazioni e costi. Quando un simulatore diventa operativo, è fondamentale avere un gruppo di persone che vi si dedichi. Progettare, condurre e valutare un esperimento significa che all'interno di tale gruppo devono essere rappresentati diversi ruoli, dal tecnico all'esperto di fattori umani. Senza un tale gruppo non è possibile realizzare test di buona qualità: anche il simulatore più avanzato è inutile senza del personale specializzato.

### **2.4 Malessere da simulatore**

Il malessere da simulatore è un problema comune a tutti i simulatori di guida. Il problema ha origine da una varietà di cause interagenti, tra queste vi sono:

- Elevato ritardo tra le azioni del guidatore e le reazioni del simulatore. Ciò significa che il guidatore prosegue la sua azione per un tempo maggiore del solito, così che quando il simulatore reagisce, il movimento è troppo ampio. Il ciclo chiuso, guidatore e simulatore, diventa instabile ed il guidatore deve compiere un grande sforzo solamente per mantenere la strada.
- Il sistema di moto del simulatore produce movimenti nell'area del mal d'auto, frequenze intorno a i 0,2 Hz ed accelerazioni superiori a  $0,2 \text{ m/s}^2$ :

- Le informazioni prodotte dal sistema visivo e dal sistema di moto non sono sincronizzate. Se la scena visuale ed il ritorno di forza non corrispondono, il guidatore diventa facilmente disorientato.

Il malessere da simulatore si verifica quando i soggetti testati sono preda della nausea e rimangono in tale stato per un certo periodo anche dopo la fine del test. Non tutti sono affetti da malessere e il suo verificarsi dipende moltissimo da come il simulatore è stato progettato.

E' importante distinguere tra malessere da moto e da simulatore. Se il guidatore diventa malato a causa del moto nella stessa situazione in una macchina reale, ciò prova che il simulatore emula l'ambiente reale. Il malessere da moto è, quindi, prova di un buon realismo; il malessere da simulatore è soprattutto un grosso problema da risolvere. Ovviamente, il malessere da simulatore intacca i risultati dei test.

## **2.5 Validazione**

La validità di un simulatore rappresenta la misura di quanto un simulatore è realistico. Quando si analizzano i risultati di un esperimento, la validità di un simulatore dovrebbe essere nota. Quello della validazione è un concetto ampio, che, in letteratura, viene suddivisa in:

- Validità fisica: confronto tra il movimento restituito dal simulatore e le dinamiche veicolari simulate.
- Validità percettiva: confronto tra la percezione del movimento al simulatore ed in una situazione reale.

- Validità comportamentale relativa: confronto tra gli effetti della variazione di alcuni parametri ambientali (strada, veicolo, condizioni di traffico) sul comportamento di guida.
- Validità comportamentale assoluta: confronto tra il comportamento di guida al simulatore ed nella realtà.

Nello stabilire il grado di validità di un simulatore, è necessario, in ogni caso riferirsi al tipo di studi a cui lo stesso è stato destinato. Differenti parametri devono essere validi per differenti tipologie di esperimenti.

## **2.6 Alcuni simulatori di guida esistenti**

Nel mondo esistono almeno un centinaio di simulatori di guida con diverse capacità.

I simulatori esistenti possono essere classificati in tre differenti categorie: alto, medio e basso livello. La categoria ad alto livello include tutti i simulatori che soddisfano tutte le richieste di un simulatore avanzato: si tratta di sistemi che garantiscono campi di visione prossimi ai 360° con un'ampia piattaforma mobile. Nella categoria di basso livello si possono inserire i simulatori costituiti da pc ordinari e dotati di un volante ed una pedaliera. I simulatori che si collocano tra queste due categorie vengono definiti di medio livello.

Di seguito, per dare un'idea di quanto di meglio offre il mercato della simulazione di guida verranno descritti alcuni tra i più performanti simulatori di medio/alto livello.

### 2.6.1. Simulatore Daimler\_Chrysler

Il primo modello di simulatore di questa azienda è stato sviluppato nel lontano 1985 e rinnovato nel 1993/94 per aumentarne le prestazioni (Figura 2—5). Durante l'ammmodernamento, il modello dinamico del veicolo e il sistema visivo sono stati sostituiti e il sistema di moto è stato esteso lateralmente.

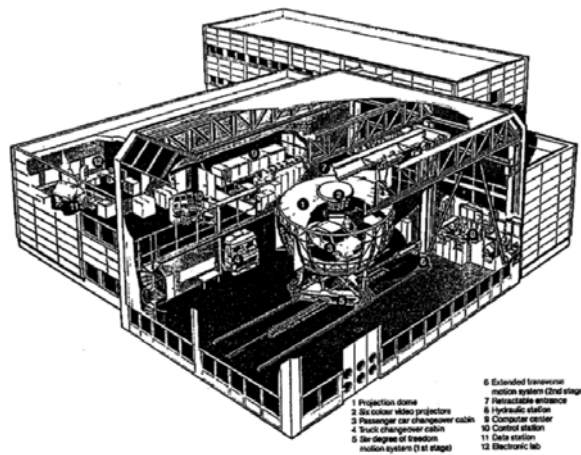


Figura 2—5 Layout laboratorio Daimler-Benz.

Il sistema di proiezione (frame rate 50Hz e transport delay pari a 70-85 ms) è costituito da 6 canali generanti immagini in tempo reale, di questi, cinque sono dedicati alla proiezione frontale (campo di visione: 180° in orizzontale e 35° in verticale). Il canale rimanente è utilizzato per la proiezione sullo schermo posteriore (48° in orizzontale e 35° in verticale), che può essere osservato dallo retrovisore laterale sinistro e da quello centrale. Ogni canale ha una risoluzione di 1024x926 pixels.

Il sistema di moto è costituito da una piattaforma mobile a 6 gradi di libertà e da un sistema di moto che garantisce un ampio movimento laterale. Il

massimo movimento laterale fornito dalla piattaforma è di  $\pm 1,5$  m (accelerazione massima 1,2 g con una frequenza di 3 Hz), a cui è possibile sommare un contributo extra di  $\pm 2,3$  m garantito da un grande attuatore idraulico (accelerazione massima 0,7 g con una frequenza di 5 Hz) che muove lateralmente l'intero sistema (piattaforma, cabina, sistema video).

Il modello dinamico in tempo reale CaSimIR (**Car Simulation In Real Time**) è stato sviluppato presso il centro ricerche Daimler-Chrysler ed è costituito da due tipi di moduli:

- I moduli dato contengono informazioni sulle specifiche del veicolo (informazioni sulla massa, sulle sospensioni, e sulle caratteristiche del motore).
- I moduli struttura (albero motore, contatto ruota/strada, sospensioni e volante) rappresentano la struttura del veicolo in dettaglio ed con grande generalizzazione.

Il modello può simulare differenti autovetture e camion. Il modello di autovettura ha dai 18 ai 37 gradi di libertà e può simulare tutte le possibili opzioni di guida.

Il modello è un modello fisico che sfrutta le leggi del moto di Newton ed in cui vengono immessi solo dati veicolari autentici.

### **2.6.2. NADS**

Il National Advanced Driving Simulator (Figura 2—6) è il più avanzato e costoso (costo totale 50 milioni di dollari) esistente al mondo. Il progetto, iniziato nel 1992 all'Università del Iowa dalla National Highway Traffic Safety Administration, è stato portato a termine nel Natale del 2001.





**Figura 2—6 Sistema di simulazione NADS.**

Ogni elemento di questo simulatore presenta caratteristiche ottimali: un piano XY di moto di circa  $400 \text{ m}^2$  con in cima un exapod di grande dimensioni ed un piano vibrante con frequenze superiori a  $20 \text{ Hz}$  ed escursioni di  $\pm 0,5 \text{ cm}$ .

Il dome montato sulla piattaforma dinamica presenta all'interno un sistema grafico (frame rate  $60\text{Hz}$  e transport delay pari a  $50 \text{ ms}$ ) da 15 canali in grado di coprire l'intero campo di visione ( $360^\circ$ ). Il sistema consente il cambio cabina con facilità, sono disponibili 4 diverse cabine, con un tempo medio per la sostituzione di otto ore.

Il progetto NADS ha sviluppato un nuovo processo per creare modelli dinamici per il simulatore, tramite Simulink e costruendo un modello per l'intero veicolo. Questo processo si serve di un tool di modellizzazione multy-body sviluppato al NADS, chiamato NADSdyna. NADSdyna permette di costruire diversi sottosistemi di veicoli (es.: sospensioni, motore e cambio). Una volta creato un modello di un sotto-sistema, è possibile

validarlo in maniera individuale. L'equipe del NADS ha sviluppato una libreria di modelli di tutti i sotto-sistemi presenti in un veicolo standard e tali modelli possono essere utilizzati per ottenere nuovi veicoli.

### **2.6.3. Il simulatore dinamico della Renault**

Il simulatori di guida vengono utilizzati alla Renault da circa dieci anni. Al momento, Renault possiede tre simulatori di guida (un quarto "Ultimate" è in fase di completamento, vedi Figura 2—7), situati al Technocentre presso la Research Direction a Guyancourt (Parigi). Due dei simulatori sono statici, il terzo è dinamico. Il simulatore dinamico è stato progettato dal gruppo di Vehicle Simulation e Perception presso il Renault Research Department nel 1999.



**Figura 2—7 Immagini del nuovo simulatore della Renault "Ultimate".**

La cabina del simulatore è una versione modificata di una Renault Clio a scala reale. Il campo totale di visione è di  $150^{\circ} \times 40^{\circ}$ , coperto da tre canali con  $1024 \times 768$  pixels. Le immagini vengono proiettate su tre schermi piatti adiacenti. Per mantenere stabile il riferimento visuale relativo della cabina durante il moto della piattaforma, una compensazione dinamica del punto di vista del guidatore viene effettuata del modulo software di generazione delle

immagini. Le immagini dei retrovisori sono proiettate su uno schermo piatto posto dietro la cabina.

La cabina è montata su una piattaforma mobile a 6 gradi di libertà. Gli spostamenti massimi del simulatore sono  $\pm 22$  cm e  $\pm 15^\circ$  in tutte le direzioni lineari ed angolari. L'accelerazione massima consentita è di 0,5 g e 0,4 m/s. per i moti angolari, i valori massimi sono  $300^\circ/\text{s}^2$  e  $30^\circ/\text{s}$ .

Il comportamento del veicolo simulato è fornito da MADA V3.0. Questo modello calcola 250 variabili interne in tempo reale. E' un modello multi-body, che gira a 100 Hz su computer standard.

Per creare uno scenario di traffico realistico, Renault usa il proprio software di simulazione, SCANeR II. Si tratta di un sistema completo che consente il controllo del veicolo e dell'ambiente stradale durante la simulazione.

#### **2.6.4. VIRTTEX**

Il VIRTual Test Track EXperiment (Figura 2—8) è situato al Ford Research Laboratory a Dearborn (USA). Il VIRTTEX è un nuovo modello di simulatore di alto livello costruito tra il 2000 ed il 2001.

Il sistema di moto è costituito da un exapod di grandi dimensioni ad elevate prestazioni ottimizzate per veicoli terrestri: il moto laterale e longitudinale è di  $\pm 1,6$  m ed il tempo di risposta è di 15 ms.

Al di sopra della piattaforma vi è un grande dome; sul soffitto del quale sono installati cinque proiettori che garantiscono un ampio campo di visione sia frontale che posteriore. Potenti schede grafiche consentono una risoluzione frontale di 1600x1200 pixels per canale a 60 Hz. Il transport delay è leggermente alto (circa 80 ms).

Il modello del veicolo tiene conto del telaio/sospensioni, potenza del motore, del volante e dei pneumatici. Il modello dinamico del veicolo è a 6

gradi di libertà a cui si aggiunge un grado di libertà per ognuna delle sospensioni. Il modello dello sterzo ha più di 10 parametri che possono essere cambiati in tempo reale.



**Figura 2—8 Il Ford VIRTTEX.**

## **2.7 Il simulatore di TEST Scarl**

Il simulatore di guida di TEST Scarl (Figura 2—9) è stato assemblato, secondo precise richieste della società committente, dalla ditta francese Oktal nel 2005, a Napoli.



**Figura 2—9 Il sistem di simulazione di TEST Scarl.**

L'idea era quella di costruire un simulatore di guida ad elevate prestazioni ma nel contempo snello ed utilizzabile per diverse tipologie di sperimentazioni. A tale scopo si è pensato di realizzare una architettura "leggera" costituita da una piattaforma mobile di ridotte dimensioni montante solamente una cabina di guida, i grandi schermi di proiezione non sono imbarcati ma fissati al suolo. Una particolarità di questo simulatore è costituita dal fatto che il sistema è stato pensato per ospitare anche una cabina di volo (cambio cabina in circa mezz'ora). Il simulatore, situato presso una delle sale dell'Istituto Motori del CNR di Napoli, è operativo dal marzo 2006.

### **2.7.1. Il sistema visuale**

La visione frontale della strada e dell'ambiente circostante è generata in tempo reale da workstation dotate di schede grafiche (Nvidia Fx Quadro) altamente performanti. Il campo totale di visione è di  $160^{\circ} \times 57^{\circ}$ , garantito da tre canali visuali aventi una risoluzione di  $1400 \times 1050$  pixels ognuno. Le immagini vengono proiettate su tre schermi (dimensioni  $4 \times 3$  m ciascuno) piatti, fissati al terreno: gli schermi laterali sono ruotati di  $30^{\circ}$  rispetto a quello centrale. I retrovisori sono stati sostituiti da tre schermi LCD da  $6,5''$  gestiti da due pc Pentium standard. Il siffatto sistema garantisce un frame rate massimo di 75 Hz, una risoluzione dello schermo di circa  $3,5$  arcmin e una dimensione dei singoli pixels al centro dello schermo di 3 mm. Infine, la cabina è equipaggiata con una video camera direttamente collegata ad una tv  $14''$  posta nella sala di supervisione.

### **2.7.2. Cabina**

La cabina di guida è la metà anteriore di una reale Citroën C2, dotata di un cruscotto, vari indicatori (velocità, livello di carburante, indicatori di direzione, marcia inserita, etc.), due sedili regolabili e retrovisori laterali e centrale (Figura 2—10).



**Figura 2—10 Retrovisori ed interni del simulatore di TEST Scarl.**

Il cambio può essere sia sequenziale che automatico. Volante e pedali sono provvisti di un sistema dinamico di ritorno di forza.

Il sistema sonoro consiste in 2 altoparlanti ed un subwoofer. Infine, un microfono connesso al sistema consente al supervisore di comunicare con il guidatore.

### **2.7.3. Il sistema di moto**

La cabina è montata su di una piattaforma mobile della CueSim, che include 6 attuatori lineari elettrici. L'exapod ha 6 gradi di libertà e può portare un carico di 1000 kg. Le caratteristiche della piattaforma sono riportate nella seguente tabella.

Parametri	Modello (Maxcue 610-450-16-12)		
Max. Carico (kg)	1000		
Altezza (min,max)	1180,1675		
Peso (kg)	625		
Dimensioni (mm)	1980 x 1750		
Performance	Pos.	Vel.	Acc.
Mov. X (mm, mm/s, g)	-491 /+432	718	+/- 1,39
Mov. Y (mm, mm/s, g)	-425 /+425	712	+/- 1,2
Mov. Z (mm, mm/s, g)	-247 /+248	484	+/- 0,59
Rollio ( $^{\circ}$ , $^{\circ}/s$ , $^{\circ}/s^2$ )	-25 /+25	50	575
Beccheggio ( $^{\circ}$ , $^{\circ}/s$ , $^{\circ}/s^2$ )	-24 /+25	48	595
Imbardata ( $^{\circ}$ , $^{\circ}/s$ , $^{\circ}/s^2$ )	-43 /+43	82	1100

#### 2.7.4. Il sistema dinamico

Il sistema dinamico del simulatore di TEST Scarl ha una frequenza di aggiornamento di 100 Hz ed è costituito da diversi sottomodelli: drive-line, volante, pneumatici, confort, forze aerodinamiche, ruote non motrici, freno, chassis. Il modello di chassis, descritto come un sistema multi-body, calcola la dinamica dello chassis, il veicolo, gli assali e i carichi. La modellizzazione multi-body è stata implementata tramite il software MesaVerde. Il modello di drive-line gira a 1000 Hz e contiene i seguenti elementi: motore, cambio, trasmissione differenziale, ruote motrici, limitatore di velocità, frizione, modello di consumo, emissioni inquinanti.

Il modello di pneumatici usato è di tipo Pacejka, nella formulazione di Michelin. I valori usati sono stati forniti da tutti i maggiori produttori di pneumatici.

Il modello globale dei freni è composto da: un sotto-modello idraulico dei freni, in cui gli aspetti meccanici e termici sono tenuti in considerazione nel calcolo dell'efficienza del sistema frenante e un modello di ABS che può essere attivato o disattivato.

Il sotto-modello di confort non è fisico ma solo un effetto speciale e riproduce le accelerazioni verticali dovute all'irregolarità con una frequenza di quasi 30 Hz.

Le forze aerodinamiche sono riprodotte tramite un sotto-modello che tiene in conto il coefficiente di penetrazione frontale e laterale dell'aria.

## **2.8 L'utilizzo Dei Simulatori Di Guida Per La Sicurezza Stradale – Lo Stato Dell'arte**

### **2.8.1. Introduzione**

L'utilizzo di simulatori di guida in campo scientifico e industriale si sta sempre più sviluppando negli ultimi anni. A livello internazionale si ha una sempre maggiore coscienza delle potenzialità applicative di tale strumento e dei vantaggi che dalla sua applicazione conseguono.

I campi di applicazione in cui i simulatori di guida sono più impiegati sono i seguenti:

- i) Studi relativi all'utente
- ii) Studi relativi all'infrastruttura
- iii) Studi relativi al veicolo
- iv) Addestramento alla guida

La sicurezza stradale dipende, in via del tutto generale, dal successo dell'interazione tra utente, infrastruttura e veicolo. La maggior parte degli studi che vengono condotti mediante l'utilizzo di un simulatore di guida sono, quindi, inquadrabili nell'ottica di studio della sicurezza stradale.



Per quanto concerne l'oggetto del presente documento e, quindi, le linee generali di utilizzo del laboratorio di sicurezza stradale di TEST, l'ottica degli studi effettuabili nel campo della sicurezza stradale riguardano principalmente l'interazione tra utente e infrastruttura e hanno, come principale oggetto di analisi, la configurazione fisica e funzionale di quest'ultima.

Le principali esperienze che possono essere condotte attraverso l'uso del simulatore di guida di cui è dotato il laboratorio per la sicurezza stradale della società TEST S.c.a.r.l. sono, dunque, le seguenti:

- a) *studio del comportamento del conducente in presenza di anomalie del traffico*
- b) *studio del comportamento del conducente in presenza di anomalie del tracciato stradale*
- c) *studio del comportamento di guida del conducente in condizioni climatiche avverse*
- d) *studio del comportamento di guida del conducente in condizioni di scarsa e/o fastidiosa illuminazione*
- e) *studio del comportamento di guida del conducente all'interno di gallerie*
- f) *studio della percezione da parte del conducente della segnaletica orizzontale e verticale esistente*
- g) *ricerca della configurazione ottimale della segnaletica orizzontale e verticale*

Ulteriori esperienze che possono essere condotte interfacciando il simulatore di guida con sistemi esterni sono:

- h) studio dell'efficacia di sistemi di ausilio alla guida intelligenti (ADAS)
- i) studio della percezione da parte del conducente delle informazioni riguardanti le condizioni del traffico sulla rete stradale
- j) studio di sistemi di rilevamento delle condizioni di affaticamento del guidatore

Nel seguito sono riportate una serie di esperienze condotte a livello internazionale per fornire un quadro delle tipologie di sperimentazioni comunemente effettuate, delle loro modalità di svolgimento e dei risultati ottenuti. Alla fine vengono elencati i riferimenti bibliografici dei lavori più recenti presenti in letteratura cui fare riferimento per maggiori informazioni.

## **2.8.2. Panoramica delle esperienze condotte in letteratura**

### **(I) Nome del progetto/lavoro**

SAFETY ISSUES OF RED-LIGHT RUNNING AND UNPROTECTED LEFT-TURN AT SIGNALIZED INTERSECTIONS

### **Soggetto attuatore**

CENTER FOR ADVANCED TRANSPORTATION SYSTEMS SIMULATION - UNIVERSITY OF CENTRAL FLORIDA (UFC) – FLORIDA, USA

### **Autore/i**

X. YAN

### **Anno di pubblicazione dei risultati**

2005

### **Descrizione del progetto**

Il lavoro è incentrato sulla disamina degli incidenti stradali dovuti al mancato rispetto del divieto di transito sancito dal rosso di lanterna di un'intersezione semaforizzata e alla presenza di svolte a sinistra pericolose.



Dopo aver esaminato le caratteristiche generali degli incidenti del tipo indicato realmente accaduti, sono state ipotizzate alternative configurazioni di segnaletica orizzontale in grado di dissuadere l'utenza ad effettuare manovre affrettate e pericolose. Lo studio dell'efficacia di tali tipologie di segnalamento è stato portato avanti mediante l'uso del simulatore di guida dell'UFC.

### **Principali conclusioni riportate**

La sperimentazione condotta ha coinvolto 42 soggetti equamente divisi in quattro classi (definite da età e sesso) con almeno un anno di patente. Prima di analizzare i risultati sono stati rimossi i dati riguardanti utenti che avevano percepito condizioni anche leggere di malessere durante la simulazione. Questa è consistita di una guida di circa 30 minuti in uno scenario urbano in cui i guidatori hanno affrontato l'attraversamento di 32 intersezioni

semaforizzate. Nella metà dei casi era presente una segnaletica orizzontale di preavviso. La tipologia di segnaletica variava tra tre alternative. I risultati hanno mostrato che, in presenza di segnaletica orizzontale, gli utenti mantengono un comportamento nettamente più prudente. Il particolare tipo di segnaletica adottato ha avuto invece un'influenza minore. Il dato quantitativo ha mostrato una riduzione del 74.3% del numero di manovre potenzialmente generatrici di incidente.

**(II) Nome del progetto/lavoro**

RED LIGHT RUNNING AND LIMITED VISIBILITY DUE TO LTV'S USING THE UFC DRIVING SIMULATOR

**Soggetto attuatore**

CENTER FOR ADVANCED TRANSPORTATION SYSTEMS SIMULATION - UNIVERSITY OF CENTRAL FLORIDA (UFC) – FLORIDA, USA

**Autore/i**

E. RADWAN, X. YAN, R. HARB, H. KLEE, M. ABDEL-ATY

**Anno di pubblicazione dei risultati**

2005

**Descrizione del progetto**

Il progetto è stato realizzato contestualmente con quello descritto in precedenza. Lo scopo è stato quello di investigare l'effetto della presenza, all'interno del parco circolante, di Light Truck Vehicles (LTVs) (quali ad esempio SUV o autocarri di piccole dimensioni, etc.) e di Larger Size Vehicles (LSV) (quali ad esempio autocarri di grandi dimensioni, autobus, etc.) sulla frequenza di accadimento

di incidenti per mancata visibilità. Entrambi i tipi di veicoli influiscono fortemente sulla visibilità del contesto stradale per gli utenti che si dovessero trovare a guidare dietro di loro. Anche in questo caso è stato utilizzato il simulatore di guida dell'UFC.



### **Principali conclusioni riportate**

L'esperienza è stata condotta sullo stesso scenario utilizzato per il precedente progetto e con gli stessi 42 soggetti. I risultati hanno mostrato che gli LTVs influiscono su una sostanziale riduzione della campo di visibilità orizzontale. Inoltre, a causa della frustrazione dovuta alla difficoltà di eseguire manovre di sorpasso, la velocità sperimentata dai guidatori che si trovano a seguire LTVs è sensibilmente maggiore di quella mantenuta seguendo normali automobili. Di conseguenza, la presenza di LTVs influisce sulla probabilità di collisione e sulla gravità delle stesse. La presenza di LSVs influisce, invece, sulla visibilità verticale. La loro presenza incide sulla capacità di visione della regolazione semaforica se non nelle immediate vicinanze e questo incide sul numero di attraversamenti di intersezioni semaforizzate durante il

rosso di lanterna, con un conseguente aumento del rischio di collisioni. D'altro canto, però, l'aumento della velocità di guida per gli utenti che si trovano a seguire un LSV non è risultato statisticamente significativo. Durante la sperimentazione sono stati inoltre testati posizionamenti alternativi della segnaletica verticale in grado di ridurre l'effetto della mancanza di visibilità.

**(III) Nome del progetto/lavoro**

UNOBTRUSIVE DROWSINESS DETECTION BY NEURAL NETWORK LEARNING OF DRIVER STEERING

**Soggetto attuatore**

THE GEORGE WASHINGTON TRANSPORTATION RESEARCH INSTITUTE - CENTER FOR INTELLIGENT SYSTEM RESEARCH - VIRGINIA, USA

**Autore/i**

R. SAYED, A. ESKANDARIAN

**Anno di pubblicazione dei risultati**

2001

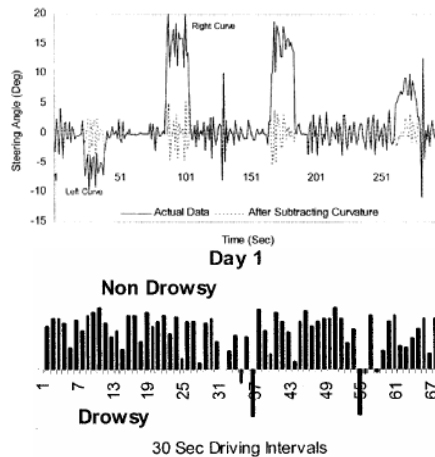
**Descrizione del progetto**

L'obiettivo del progetto è stato quello di testare un sistema non intrusivo di rilevamento di condizioni di affaticamento/sonnolenza da parte del guidatore in grado di accrescere la sicurezza di guida in campo autostradale. Il metodo studiato si basa sull'applicazione delle reti neurali artificiali (ANN). L'andamento del segnale rappresentante la variazione dell'angolo di sterzata, una volta processato, viene utilizzato per alimentare la rete neurale, che

classifica ogni intervallo di tempo come *affaticato* e *non affaticato*. La particolarità del metodo sta proprio nella sua non intrusività, essendo analizzati esclusivamente dati provenienti dal volante dell'autovettura.

### Principali conclusioni riportate

La sperimentazione è stata condotta testando, mediante l'uso di un simulatore di guida, il comportamento di 12 utenti durante 4 giorni in diversi momenti della giornata e con diversi livelli di privazione del sonno.



I risultati hanno mostrato la grande accuratezza con cui la metodologia ha saputo classificare gli intervalli di guida. L'utilizzo del simulatore di guida è stato essenziale per azzerare i rischi che sarebbero sorti nell'effettuare tale sperimentazione su strada.

**(IV) Nome del progetto/lavoro**

INVESTIGATION OF DRIVER REACTIONS TO TREAD SEPARATION SCENARIOS IN THE NATIONAL ADVANCED DRIVING SIMULATOR (NADS)

**Soggetto attuatore**

NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION – VEHICLE RESEARCH AND TEST CENTER – OHIO, USA

**Autore/i**

T. A. RANNEY, G. HEYDINGER, G. WATSON, K. SALAANI, E. N. MAZZAE

**Anno di pubblicazione dei risultati**

2002

**Descrizione del progetto**

L'obiettivo dello studio è stato quello di analizzare la reazione dei guidatori alla rottura di un pneumatico. Lo studio è stato condotto utilizzando il NADS. In particolare si è voluto studiare l'effetto sul comportamento alla guida della presenza una direzione di deriva del veicolo, della consapevolezza di un'imminente rottura del pneumatico, della conoscenza di regole comportamentali da seguire in caso di rottura, dell'età del conducente, dell'errore di localizzazione del pneumatico danneggiato da parte del guidatore.





### **Principali conclusioni riportate**

La sperimentazione è stata condotta su 108 soggetti equamente suddivisi in tre classi di età. Ad ogni utente è stato assegnato un livello di sottosterzo tra i tre prestabiliti. Durante la guida gli utenti hanno sperimentato due rotture del pneumatico ad una velocità di 75 miles/h. La prima rottura è inaspettata. La seconda è attesa e una metà dei soggetti ha avuto informazioni sul comportamento da seguire all'atto della stessa. I risultati hanno mostrato che la specifica ruota di cui si è simulato la rottura e le istruzioni sul comportamento da seguire, non hanno avuto influenza sul comportamento di guida. L'età dei conducenti ha influito in maniera marginale solo sulla capacità di rispondere alla perdita di controllo. Grande influenza risulta invece avere l'attesa della rottura sul comportamento di guida. In questo caso, ovviamente i tempi di reazione sono risultati notevolmente inferiori rispetto al caso di rottura inattesa. Grande influenza ha, inoltre, il grado di sottosterzo del veicolo sulla probabilità di perdita di controllo del veicolo a valle della rottura del pneumatico.

**(V) Nome del progetto/lavoro**

USING AN ADVANCED DRIVING SIMULATOR AS A DESIGN TOOL IN ROAD TUNNEL DESIGN

**Soggetto attuatore**

SWEDISH ROAD AND TRANSPORT RESEARCH INSTITUTE  
VTI, SWEDEN

**Autore/i**

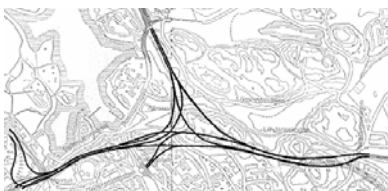
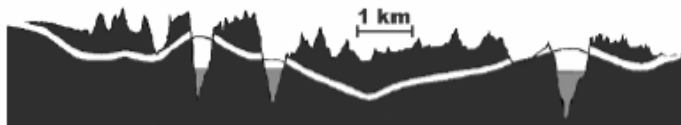
M. LIDSTROM

**Anno di pubblicazione dei risultati**

1998

**Descrizione del progetto**

Lo scopo del progetto è stato quello di testare le soluzioni progettuali adottate per la realizzazione dell'autostrada che transita al di sotto del territorio di Stoccolma.



L'importanza di fornire all'utenza un ambiente di guida in cui sapersi orientare con facilità era un elemento di grande importanza per una strada che si sviluppava completamente in galleria. Un simulatore di guida è stato dunque utilizzato per poter esaminare nel dettaglio tutte le variabili progettuali, altrimenti difficilmente analizzabili.

### **Principali conclusioni riportate**

La possibilità di guidare nell'ambiente simulato attraverso il tracciato progettato ha dato grande impulso alla ricerca condotta. Più di 100 tra progettisti e cittadini della città hanno testato il tracciato in maniera praticamente ininterrotta per 20 giorni consecutivi. Sono state analizzate, tra le altre cose, la distanza di visibilità in ogni punto del tracciato, il posizionamento ottimo della segnaletica stradale, il sistema di controllo della strada, la configurazione complessiva dell'infrastruttura. Lo strumento è stato utilizzato come supporto alla discussione con le autorità locali, con le forze dell'ordine e con i vigili del fuoco per stabilire le migliori strategie di azione in case di evento incidentale. La possibilità, infine, di poter lavorare sulla struttura realizzata già nelle fasi iniziali della progettazione esecutiva ha consentito di eliminare tutta una serie di problemi che si sarebbero incontrati all'atto della realizzazione dell'opera.

### **(VI) Nome del progetto/lavoro**

FREEWAY GUIDE SIGN DESIGN FOR THE CENTRAL  
ARTERY TUNNEL USING A DRIVING SIMULATOR

**Soggetto attuatore**

UNIVERSITY OF MASSACHUSETTS –  
MASSACHUSETTS, USA

**Autore/i**

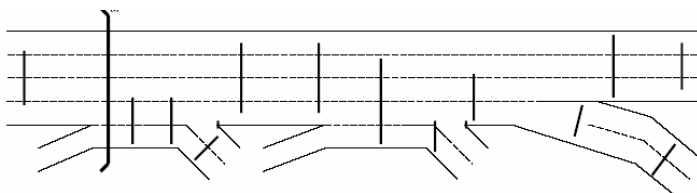
J. UPCHURCH, D. FISHER, R. CARPENTER, A. DUTTA

**Anno di pubblicazione dei risultati**

2002

**Descrizione del progetto**

Il progetto ha riguardato la messa in sicurezza della Central Artery di Boston. Tale strada di grande scorrimento, nella parte centrale, in una fascia della lunghezza di circa 3 miglia (di cui la metà in galleria) presentava 23 ingressi e uscite. Il numero di incidenti causati dagli utenti non abituali, confusi dalla segnaletica presente ha indotto le autorità locali a condurre uno studio approfondito. Il simulatore di guida dell'Università del Massachusetts è stato utilizzato a tale scopo.



**Principali conclusioni riportate**

Lo studio ha avuto due funzioni principali: da un lato ha evidenziato la necessità di ridurre a 14 le uscite e le entrate nel tratto incriminato e dall'altro ha consentito la progettazione ottimale della segnaletica di preavviso degli svincoli.

**(VII) Nome del progetto/lavoro**

RWS-AVV WORK-ZONES

**Soggetto attuatore**

TNO – KNOWLEDGE FOR BUSINESS – DELFT, THE NETHERLANDS

**Autore/i**

R. VAN DER HORST

**Anno di pubblicazione dei risultati**

2005

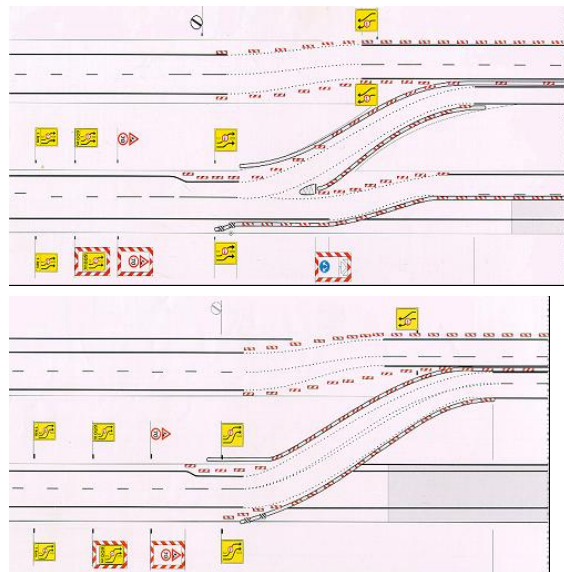
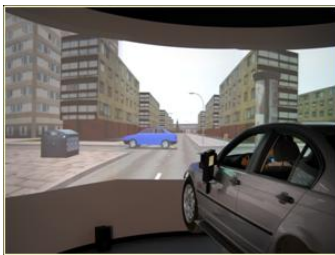
**Descrizione del progetto**

Il TNO è una delle società europee che utilizza maggiormente il simulatore di guida per scopi industriali e progettuali. In questo studio, commissionato dall'ente nazionale per la sicurezza stradale, è stata valutata la migliore configurazione, in termini di sicurezza, di un cantiere stradale da aprire su una strada estremamente trafficata.

**Principali conclusioni riportate**

Sono state analizzate diverse condizioni relative ai due scenari, in base alla larghezza delle corsie, all'estensione del cantiere, alla quantità di traffico presente. Sono state analizzate sia la sicurezza

che il comfort del guidatore. Gli utenti su cui sono stati svolti i test hanno mostrato una generale sensazione di oppressione legata all'angusto spazio di transito. Il comportamento nel primo caso, in cui c'è la separazione delle due corsie, ha comunque mostrato livelli di sicurezza e di comfort più elevati. Una larghezza delle corsie accettabile è risultata dover essere non inferiore ai 2.75m.



### **2.8.3. La letteratura di recente produzione sulle possibili applicazioni dei simulatori di guida**

Le esperienze riportate in precedenza danno un quadro generale delle possibili e più immediate applicazioni legate alla sicurezza stradale di un simulatore di guida. Esse sono state scelte perché condotte da strutture di ricerca estremamente avanzate e titolate in campo internazionale. Nel presente paragrafo si vuole inoltre presentare una selezione delle pubblicazioni del settore di più recente produzione al fine di fornire un

supporto nel caso si cercassero maggiori informazioni. Tale bibliografia non è inserita nella bibliografia generale del testo perché i contenuti dei seguenti lavori non sono inquadrabili nel contesto in cui il presente lavoro cerca di assestarsi.

- Chrysler S. T., Holick A., Williams A., D. Funkhouser. *Driver Comprehension Of Diagrammatic Advanced Guide Signs And Their Alternatives*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Musicant O., Lotan T., T. Toledo. *Safety correlation and implications of an in-vehicle data recorder on driver behaviour*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Ulmann B. R., Ulmann G. L., Dudek C. L., A. A. Williams. *Driver understanding of messages displayed on sequential portable changeable message signs in work zones*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Hesar S. G., Wang J. H., C.E. Collyer. *Adding Graphics To Dynamic Message Sign Messages*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Van der Horst R., S. De Ridder. *The influence of roadside infrastructure on driving behaviour: a driving simulator study*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007

- McAvoy D. S., Schattler K. L., T. K. Datta. *Driving Simulator Validation for Nighttime Construction Work Zone Devices*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Auberlet J. M., Champion A., Mandiau R., Espié S. and C. Kolski. *From perception to conflict resolution: a priority managing mechanism for road intersection*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- Bella F., Garcia A., F. Selves. *Driving Simulator Validation For Deceleration Lane Design*. In Proceedings of the 2007 TRB Annual Meeting. Washington D. C., USA. 2007
- E. Peli, A. R. Bowers, A. J. Mandel, K. Higgins, R. B. Goldstein, and L. Bobrow. *Design for Simulator Performance Evaluations of Driving with Vision Impairments and Visual Aids*. Transportation Research Records n. 1937 (2007).
- J.-H. Wang and Y. Cao. *Assessing Message Display Formats of Portable Variable Message Signs*. Transportation Research Records n. 1937 (2007).



- C. A. MacCarley. *Methods and Metrics for Evaluation of an Automated Real-Time Driver Warning System*. Transportation Research Records n. 1937 (2007).
- J. D. Hoffman, J. D. Lee, D. V. McGehee, M. Macias, and A. W. Gellatly. *Visual Sampling of In-Vehicle Text Messages Effects of Number of Lines, Page Presentation, and Message Control*. Transportation Research Records n. 1937 (2007).
- H. C. Leea, A. H. Leeb, D. Cameron, C. Li-Tsang. *Using a driving simulator to identify older drivers at inflated risk of motor vehicle crashes*. Journal of Safety Research, n.34 (2003)
- A.C.E. Spek, P.A. Wieringa, W.H. Janssen. *Intersection approach speed and accident probability*. Transportation Research Part F, n.9 (2006)
- T. Horberry, J. Anderson, M. A. Regan. *The possible safety benefits of enhanced road markings: A driving simulator evaluation*. Transportation Research Part F, n.9 (2006)
- C. Hoa, N. Reedb, C. Spence. *Assessing the effectiveness of “intuitive” vibrotactile warning signals in preventing front-to-rear-end collisions in a driving simulator*. Accident Analysis And Prevention n. 38 (2006)

- Thiffault P., J. Bergeron. *Monotony of road environment and driver fatigue: a simulator study*. Accident Analysis And Prevention n. 35 (2003)
- Santos J., Merat N., Mouta S., Brookhuis K., D. de Waard. *The interaction between driving and in-vehicle information systems: Comparison of results from laboratory, simulator and real-world studies*. Transportation Research Part F, n.9 (2005)
- Tornos J. *Driving Behaviour In A Real And A Simulated Road Tunnel—A Validation Study*. Accident Analysis And Prevention n. 30 (1998)
- Mindrehoud M. M., P. H. L. Bovy. *Extended time-to-collision safety measures for road traffic safety assessment*. Delft University Press, Delft. 2000

### **3. La modellizzazione del traffico nei simulatori di guida. Stato dell'arte**

#### ***3.1 Il modello di traffico del simulatore di guida del CRdCT***

Il simulatore di guida in dotazione del Centro Regionale di Competenza Trasporti della Regione Campania è un simulatore dinamico molto complesso. È stato acquistato a scatola chiusa dalla Oktal, azienda francese tra le maggiori case produttrici di simulatori di guida. Una breve descrizione della struttura di tale simulatore è stata fornita nel Capitolo 2. Come è stato quindi illustrato il software SCANeR II [42] si occupa della gestione e del coordinamento di tutti i modelli utilizzati per realizzare l'ambiente di simulazione. Uno di questi modelli è il modello di traffico.

Il modello di traffico serve a gestire i veicoli autonomi presenti all'interno di uno scenario. Al momento di creare uno scenario è necessario stabilire il numero di veicoli di cui si necessita e quindi crearli uno per volta. Ogni veicolo è definito dal particolare modello di vettura scelto tra quelli disponibili e tra quelli eventualmente creati e dal tipo di comportamento del guidatore. La procedura è piuttosto macchinosa in quanto al momento non è possibile definire automaticamente delle caratteristiche valide per un gruppo di essi. Inoltre il numero di veicoli gestibili anche volendone creare molti, non deve superare le cento unità. Ogni veicolo viene generato in un punto qualsiasi scelto dal progettista dello scenario e gli viene assegnato un

percorso da compiere. Alla fine di tale percorso scompare. Eventualmente si può utilizzare la funzione di swarm. In questo caso si definisce la dimensione di una finestra mobile centrata sul veicolo interattivo e che si muove in maniera ad esso solidale. Ogni veicolo autonomo viene generato e distrutto sul confine di questa finestra. In pratica una volta che un veicolo autonomo appare nella finestra, vedrà diminuire o aumentare la propria distanza dal veicolo interattivo a seconda delle sue caratteristiche specifiche. Allorquando la sua posizione dovesse trovarsi al di fuori dello swarm, esso sarebbe automaticamente riposizionato in un altro punto del contorno. Il risultato di questo approccio è che il veicolo interattivo incontra veicoli con una certa continuità, ma si tratta sempre degli stessi veicoli che si comportano sempre allo stesso modo e questo non aiuta ad accrescere il realismo della simulazione. Il modello utilizzato per calcolare la cinematica di ogni veicolo non è noto in maniera esplicita. Quello che viene indicato è solo il numero di parametri da settare. I parametri che regolano il comportamento del singolo veicolo riguardano ad esempio la massima velocità desiderata, il grado di accettazione dei limiti di velocità, il grado di accettazione delle regole di precedenza e della segnaletica esistente etc. Dalle simulazioni appare però che il modello, più che utilizzare questi parametri per simulare il traffico, simula il movimento dei veicoli in maniera tale che possano sembrare visivamente accettabili. Il veicolo autonomo, in altre parole, come già detto, viene considerato alla stregua di qualunque altro oggetto dello scenario. Con questo presupposto, non è possibile pensare di utilizzare il simulatore di guida per effettuare degli studi di traffico.

D'altro canto un approccio di questo tipo è quello comunemente utilizzato da tutti i moduli di gestione del traffico presenti in commercio.

### **3.2 La simulazione dei veicoli autonomi – Background**

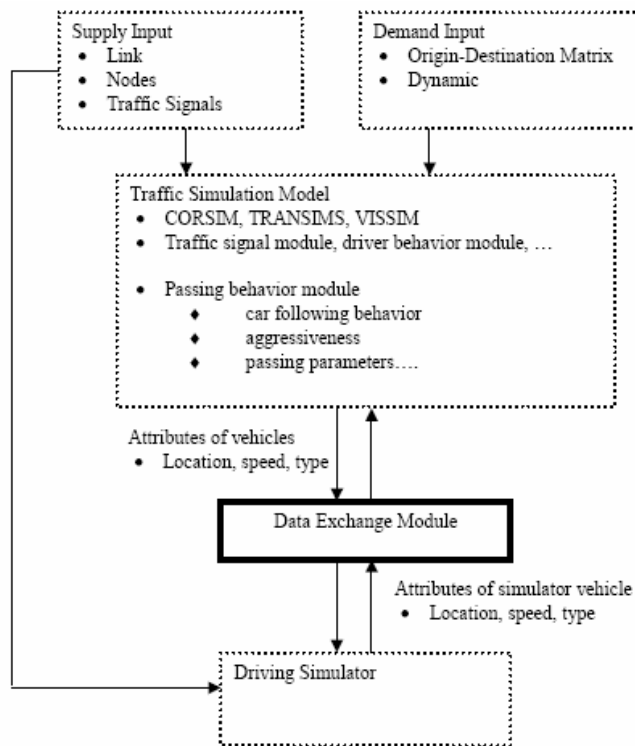
Il problema della simulazione del traffico all'interno dei simulatori di guida non è un tema ampiamente affrontato dagli studiosi del settore. Poche sono le esperienze riportate in letteratura. Questo perché l'idea di utilizzare un simulatore di guida per esperimenti legati al traffico è ancora poco diffusa.

I lavori reperiti possono dividersi in due grandi categorie:

1. *Tentativi di integrare un simulatore di guida con un modello di traffico creato ad hoc*
2. *Tentativi di integrare un simulatore di guida con un modello di traffico esistente*

Un tentativo di fornire una concettualizzazione generale del problema è stato effettuato da Jenkins e Rilett nel 2001 [26]. In tale studio, dopo essere stati esaminati gli effetti degli avanzamenti tecnologici tanto nell'ambito dei simulatori di guida quanto in quello dei modelli di simulazione microscopica del traffico, gli autori si soffermano sul problema della generale cattiva rappresentazione di questo all'interno dei primi. A conferma di quanto detto in precedenza sottolineano il fatto che all'interno dei simulatori di guida il traffico è rappresentato solo per scopi visivi e non contiene gli sforzi teorici ampiamente condotti in letteratura. Vengono quindi discussi i modelli di microsimulazione esistenti con le rispettive capacità di modellizzare un sistema di trasporto e vengono analizzati i

principali software di simulazione microscopica del traffico esistenti e i modelli utilizzati al loro interno. Lo studio, quindi, propone una metodologia di integrazione. Essa non deve essere realizzata inglobando un modello nell'altro, ma piuttosto costruendo una struttura che ne consente la mutua interazione. In questo modo non ci si vincola all'uso né di uno specifico simulatore di guida, né di un particolare modello di microsimulazione. La struttura proposta è quella mostrata in Figura 3—1.



**Figura 3—1 Esempio di Integrazione simulatore di guida – modello di microsimulazione del traffico [26]**

Ciò che è necessario creare è il Data Exchange Module (DEM) e le comunicazioni dei due modelli con esso. L'architettura è basata su di un costante ciclo retroattivo tra gli input del simulatore di guida e gli output del microsimulatore. In particolare il modello di simulazione deve fornire come input al simulatore di guida la domanda di spostamento. La reazione del guidatore testato al traffico presente viene tradotta in maniera che il modello di simulazione la possa comprendere ed inviata ad esso che aggiorna la simulazione. Il grande vantaggio ottenibile è che modello di traffico può essere calibrato con i dati rilevabili nel mondo reale.

Tale architettura è assolutamente condivisibile e presenta i seguenti challenge:

- Le reti di traffico devono coincidere. Se ciò non avviene il guidatore del simulatore di guida potrebbe osservare comportamenti strani nei veicoli autonomi.
- Le informazioni devono viaggiare attraverso il DEM e raggiungere i due modelli senza disturbarli. Inoltre il DEM deve essere in grado di leggere le informazioni relative al veicolo autonomo e scriverle nel microsimulatore e di leggere le informazioni relative agli altri veicoli del microsimulatore e scriverle nel simulatore di guida.
- È probabile che i codici sorgente dei due ambienti siano differenti. Il DEM deve essere in grado di comunicare con entrambi. Se le informazioni inviate al DEM fossero nello stesso formato la cosa si semplificherebbe molto.
- Le velocità di comunicazione devono essere molto maggiori dei tempi necessari ad effettuare i calcoli dei due modelli.

- Il simulatore di guida deve ricevere informazioni inerenti alla posizioni dei veicoli con frequenza pari almeno a 60Hz. I modelli di simulazione del traffico lavorano al massimo a 10Hz. Tra due step di calcolo successivi da parte del microsimulatore, devono essere eseguite delle interpolazioni in modo che il guidatore del simulatore di guida non veda i veicoli intorno muoversi a scatto.

Una volta superati questi problemi applicative bisogna verificare che il modello costruito risponda alle esigenze per cui è stato realizzato. Bisogna cioè implementare un'opportuna fase di validazione della struttura creata.

### **3.2.1. Esempi di moduli per la gestione del traffico creati ad hoc**

Le esperienze condotte e riscontrabili in letteratura per la simulazione del movimento dei veicoli autonomi non sono moltissime e sono riportate di seguito. Un paio di essi ([1] e [15]) hanno usato l'approccio di simulare esclusivamente i veicoli presenti nelle immediate vicinanze del veicolo interattivo. La maggior parte degli altri modelli disponibili si basano sulla struttura proposta da Michon per descrivere la guida [31]. Questa struttura divide la guida in tre livelli: Strategico, Tattico e Operativo. Nel primo livello l'utente compie, in maniera più o meno inconscia, le scelte riguardanti, ad esempio, l'ora il motivo, il modo o il percorso da seguire durante lo spostamento. A questo punto, durante lo spostamento, a seconda degli input che riceve dall'esterno il guidatore compie delle scelte di secondo livello riguardanti le particolari manovre da compiere (sorpasso, etc.). A questo punto il terzo livello riguarda l'applicazione delle scelte effettuate in precedenza. Ovviamente questi livelli si influenzano



vicendevolmente durante la fase di guida. Questo tipo di struttura è stata adottata ad esempio in [58] e [17].

Per quel che riguarda i campi di applicazione considerati, la maggior parte dei modelli utilizzati si riferiscono alla simulazione del traffico su autostrada e su strade urbane ([1], [4], [17], [17], [58] and [58]). Una quantità di lavori molto più ridotta è stata dedicata alla simulazione del comportamento dei veicoli sulle strade a singola carreggiata con possibilità di sorpasso. In [1] è definito un modello di sorpasso molto semplice che utilizza un gap critico per tutti i guidatori ed ipotizza il rispetto dei limiti di velocità durante tutta lo svolgimento di tale fase (cosa che non sempre si verifica). Un modello più complesso per questo tipo di strade è descritto in [51] e [42]. In esso il fenomeno di creazione di plotoni a causa delle difficoltà di effettuare il sorpasso su strade a singola carreggiata con una corsia per senso di marcia è tenuta in considerazione.

Per quel che riguarda la strategia di integrazione utilizzata tra i due modelli di simulazione, gli approcci sono stati diversi. Nell'esperienza condotta dal Prof. Stephan Espiè all'istituto francese dell'INRETS [26] il simulatore di guida è stato incorporato in ARCHISIM, il modello di simulazione microscopica del traffico da loro creato. Ad ogni modo, questo non è la maniera più efficiente di operare [26], in quanto la struttura combinata funziona solo per gli ambiti applicativi del modello di microsimulazione. Nel caso se ne volesse utilizzare un altro bisognerebbe creare l'intera struttura dall'inizio.

Un'altra esperienza legata alla creazione di un modello di simulazione del traffico che gestisca i veicoli autonomi di uno scenario di simulazione è stata portata avanti presso il NADS (National Advanced Driving Simulator)

0. Il NADS è certamente il più famoso simulatore di guida esistente. Gli autori del lavoro propongono di inserire il modello di traffico creato all'interno dell'architettura del simulatore. Nel corso di tale studio, i ricercatori del NADS hanno portato avanti un'intensa campagna di interviste per cercare di cogliere le attese che gli utenti del simulatore ripongono sul comportamento dei veicoli autonomi. Una volta raccolte tutte queste richieste, il modulo di gestione del traffico è stato creato utilizzando i modelli riguardanti le dinamiche veicolari presenti in letteratura più adatti a soddisfare le esigenze complessive.

In generale, però, i casi di studio in cui un modello di simulazione di traffico viene creato ad hoc, a parte strutture di ricerca di grandi dimensioni, come il NADS o il VTI, hanno come unico interesse: la credibilità della rappresentazione del traffico. Secondo Wright et al., infatti, i veicoli autonomi della simulazione devono possedere:

- *Intelligenza*: i veicoli devono dare l'impressione di avere le capacità di navigare autonomamente nell'ambiente simulato;
- *Individualismo*: i veicoli devono variare il loro comportamento nel tempo e devono avere tutti un comportamento almeno leggermente differente;
- *Stile*: si dovrebbe poter definire uno stile di guida comparabile con quello posseduto dagli utenti reali dell'ambito di studio che si sta analizzando.

Un altro lavoro presente in letteratura che parte da presupposti simili al precedente e che si basa sempre sulla creazione di un modello di simulazione ad hoc partendo dal modello di Michon [31] è quello presentato

da Brodin e Carlsson 0. Le osservazioni riportate in questo lavoro non aggiungono, però, nulla di particolare a quelle precedentemente individuate.

### 3.2.2. Integrazione con software di simulazione del traffico esistenti

Ancora meno numerose sono le esperienze condotte in letteratura sull'integrazione di un software di simulazione del traffico con un simulatore di guida per la gestione del moto dei veicoli autonomi. Ciò è dovuto, come anticipato, alla scarsa domanda che attualmente riguarda la gestione del traffico che grava su un'intera rete di trasporto in un simulatore di guida. Gli articoli proposti sono comunque molto interessanti.

Uno studio in cui l'integrazione è stata effettivamente realizzata è stato realizzato in Indonesia [31]. In questo caso trattasi di un simulatore di guida statico gestito da un solo PC. L'integrazione è stata effettuata con Paramics.

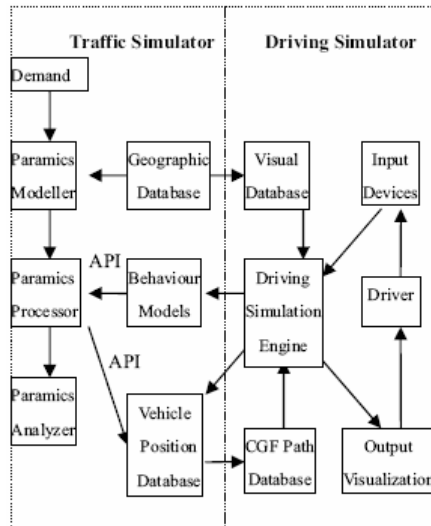


Figura 3—2 Ipotesi di struttura di integrazione tra simulazione di guida e di traffico

[31]

Lo scopo dell'integrazione, in questo caso, è stato proprio quello di utilizzare il simulatore di guida come strumento per testare nuove tecnologie di informazione all'utenza (ITS). Per dare informazioni realistiche all'utente, anche in simulazione, è necessario avere il controllo delle condizioni del traffico sull'intera rete per poter fornire una realistica previsione dei tempi di percorrenza. La strategia d'integrazione è stata la seguente:

- *nessuno dei due modelli è stato alterato.*
- è stata realizzata la loro *comunicazione*. Paramics viene utilizzato per gestire il traffico e le strumentazioni di informazioni all'utenza (pannelli a messaggio variabile e i detector virtuali) presenti nella rete simulata.
- comunicazione realizzata utilizzando *l'API di Paramics* da gestire in linguaggio di programmazione C. Inoltre attraverso l'API è stato possibile fornire ad ogni singolo veicolo un diverso comportamento, in maniera da fornirgli un comportamento oltre che più efficiente, anche più naturale.
- scambio di informazioni è effettuato utilizzando un *database* in cui vengono raccolte le informazioni riguardanti i veicoli da entrambe i modelli.
- il valore dei parametri alla base dei modelli riguardanti il comportamento dei veicoli di fronte alle scelte da effettuare lungo il proprio spostamento, sono, inoltre, *continuamente aggiornati* dai dati derivanti dalle esperienze condotte con il simulatore di guida.

L'esperienza risulta estremamente interessante sia dal punto di vista degli obiettivi perseguiti durante lo studio, sia per quel che riguarda la

metodologia di integrazione. L'unico elemento di perplessità, risiede nel fatto che il simulatore di guida utilizzato fosse un simulatore estremamente semplice (costituito da un software che gestisce in un unico PC la simulazione di guida condotta utilizzando un volantino posizionato su una scrivania).



**Figura 3—3 Tipo di simulazione di guida nell'esperienza indonesiana [31]**

Visti gli obiettivi perseguiti e la tipologia di simulatore di guida utilizzato, è probabile che elementi di notevole importanza quali ad esempio la restituzione visiva del comportamento dei veicoli potesse anche essere meno curata. Ad ogni modo lo studio risulta tra i più interessanti in assoluto condotti nel settore.

Un'altra esperienza di questo tipo è stata condotta presso l'istituto nazionale per la sicurezza stradale norvegese (SINTEF). La loro esperienza nel campo della simulazione di guida è tra le più vaste al mondo. Il lavoro condotto sull'integrazione tra un simulatore di guida ed un software di simulazione del traffico sebbene sia oggi citato in tutti i lavori del settore [8], nel periodo in cui fu svolto, non suscitò un grande interesse tant'è che è rimasto non pubblicato.

Il software di simulazione adottato è stato AIMSUN.



**Figura 3—4 Il simulatore di guida di SINTEF**

Lo studio individua una serie di benefici derivanti dall'integrazione del simulatore di guida con un microsimulatore di traffico. Vengono riportati di seguito:

- facilità di adattamento di varie condizioni di traffico
- facilità di cambiamento delle caratteristiche di una singola classe di veicoli
- possibilità di riprodurre incidenti e condizioni di pericolo
- possibilità di dotare gli altri veicoli dello scenario delle stesse apparecchiature che si stanno testando con il simulatore di guida
- possibilità di validare il comportamento alla guida del simulatore confrontandolo con i comportamenti assunti dagli altri veicoli mossi dal microsimulatore
- in senso duale si può calibrare il microsimulatore attraverso i dati provenienti dal Simulatore di Guida

- Si può immergere chi guida il simulatore in un ambiente più realistico

Analizza quindi le problematiche principali dell'integrazione fornendone una risoluzione:

- La rete utilizzata nei due modelli deve essere esattamente la stessa;
- Il numero di veicoli da simulare deve essere compatibile con le caratteristiche dei due modelli.
- La frequenza di elaborazione degli output deve essere compatibile.
- È necessario imporre al microsimulatore le caratteristiche cinematiche del veicolo interattivo.
- I modelli di simulazione microscopica del traffico in commercio non sono in grado di fornire una modellizzazione accurata di tutti gli scenari di guida.

Il primo punto nasconde non poche insidie. La rete in cui è immerso un simulatore di guida deve essere per sua natura molto dettagliata. I tronchi stradali in cui tale rete è suddivisa, per riuscire a cogliere tutte le particolarità delle condizioni al contorno, possono essere di lunghezza paragonabile a quella dei veicoli che vi si muovono al di sopra (!!!). Le reti dei microsimulatori sono generalmente costituite da tronchi molto più grandi. Il tronco stradale, nei simulatori di guida fornisce al modello dei parametri in base ai quali effettuare il calcolo dei comportamenti di guida. È naturale che meno tronchi esistono, meno informazioni il software dovrà gestire. Con potenze di calcolo sufficienti, però, questo problema rappresenta un ostacolo più limitato.

Per quanto riguarda il controllo dei veicoli, un software di simulazione del traffico può gestire anche diverse migliaia di veicoli, mentre, per il simulatore è sufficiente la gestione di un centinaio di veicoli che si muovono intorno al veicolo interattivo. D'altra parte, la gestione per il simulatore di guida di un numero elevato di veicoli è estremamente onerosa oltre che inutile. La soluzione auspicabile è quella di lasciare la gestione dei veicoli totalmente al simulatore di traffico, che invia al simulatore di guida solo le caratteristiche cinematiche dei veicoli che si trovino ad una certa distanza da quello interattivo (lo studio fornisce anche un suggerimento pratico indicando la presenza in AIMSUN di una funzione che impone un controllo sulla traccia dei veicoli (*veicol tracking*)). Legando le classi dei veicoli impostate al simulatore, con le classi dei veicoli del microsimulatore è possibile la visualizzazione di differenti tipologie di veicoli.

Per quel che riguarda il problema della frequenza di aggiornamento delle informazioni sulla posizione dei veicoli, bisogna considerare che un simulatore di guida, per fornire una buona resa di immagine, è necessario aggiornare ogni elemento almeno 30 volte ogni secondo, mentre all'interno di AIMSUN si può arrivare al massimo a 10 volte al secondo. Inoltre, al tempo dello studio, in AIMUSN l'intervallo di simulazione era inescandibilmente legato ad un parametro del modello di car-following che come vedremo nel prossimo capitolo è chiamato *reaction-time*. Perché la simulazione sia realistica è necessario che il *reaction-time* non sia inferiore a 0.6 sec. Per questo motivo è stato creato un algoritmo in grado di estrapolare la posizione del veicolo del simulatore di traffico in istanti intermedi inserendo un controllo delle deviazioni della tangente alla traiettoria del veicolo all'istante di simulazione dall'asse stradale e dal bordo strada.



Per quel che riguarda la gestione delle caratteristiche cinematiche del veicolo interattivo, in AIMSUN è possibile modificare ad ogni step di simulazione posizione e velocità di uno qualsiasi dei veicoli presenti nella simulazione di traffico, adattandole a quelle del simulatore e quindi adattando in cascata le caratteristiche degli altri veicoli.

Per quel che riguarda la capacità di un modello di simulazione microscopica del traffico di simulare diversi scenari di guida, questo dipende dal particolare scelto. In AIMSUN, ad esempio, la gestione dell'urbano era ed è tutt'ora al quanto problematica.

Il lavoro svolto in SINTEF, come detto, per quanto rimasto non pubblicato, rappresenta un punto di riferimento per quel che riguarda questo specifico campo di applicazione perché fornisce una base tecnica approfondita dei benefici e dei problemi esistenti. Alcune osservazioni possono essere condotte:

- le reti dei microsimulatori sono generalmente costituite da tronchi grandi perché alcuni modelli, quali ad esempio quelli di *lane-changing*, richiedono dimensioni minime degli archi stradali per attivarsi. Per questo motivo all'atto della costruzione della rete bisogna prestare attenzione a costruire archi stradali dalle giuste dimensioni che non vadano ad inficiare la modellizzazione del sistema;
- le reti nei due sistemi devono combaciare anche dal punto di vista del sistema di riferimento, perché altrimenti, lo scambio delle informazioni sui veicoli diviene più difficoltosa;

- la nuova versione di AIMSUN (v. 5) consente di separare simulation step e reaction time, purché questi sia un multiplo del primo. In questo modo si può impostare il sim step al suo valore minimo di (0.1 sec) e usare l'estrapolazione della posizione in maniera molto meno forte;
- la prossima versione di AIMSUN (v.6) sarà strutturata in maniera tale da ridurre sensibilmente i tempi di calcolo, e avrà una migliore rappresentazione del modello di lane-changing.
- con la prossima versione di AIMSUN sarà disponibile un tool chiamato micro SDK, che consentirà di entrare nel codice del software per sostituire intere parti. In questo modo sarà possibile utilizzare i modelli di simulazione che meglio si adattano alle varie condizioni che vogliono essere simulate.

Partendo dunque dalle considerazioni presenti in [8] e aggiornandole con le nuove caratteristiche di AIMSUN, è concreta la possibilità di essere immediatamente operativi alla risoluzione del problema. È necessario però esaminare prima altri lavori interessanti da cui trarre ulteriori utili considerazioni.

Un altro lavoro molto interessante è frutto di una tesi di dottorato svolta presso il VTI [42]. In questo caso, l'integrazione è stata effettuata tra il simulatore presente in Svezia e i modelli di simulazione da loro sviluppati VTISim [15] e TMPA [17].

L'interesse specifico della tesi è rivolto alla una corretta rappresentazione del traffico. Nel lavoro vengono esaminati vari punti:

- Un modello di simulazione microscopica del traffico, in generale, riproduce la propagazione di flussi veicolari sull'intera area geografica che si sta analizzando. Un modello di micro-simulazione per un simulatore di guida deve riprodurre il comportamento dei veicoli che si trovano nelle prossimità del veicolo del simulatore stesso. L'area, quindi, all'interno della quale il traffico dovrà essere simulato, dovrà muoversi in maniera solidale con tale veicolo.
- I moderni software di simulazione microscopica del traffico si definiscono stocastici, nel senso che per quasi tutti gli aspetti della simulazione si definisce una distribuzione di probabilità con media e varianza prefissata. Per quasi tutti gli elementi della simulazione viene quindi effettuata un'estrazione casuale all'interno di queste distribuzioni. Questo permette ad esempio di avere, all'interno della stessa simulazione, i veicoli appartenenti ad una stessa classe che si comportano in maniera leggermente differente tra di loro. Le estrazioni casuali dipendono generalmente da un seme che può essere definito dall'utente. Se questo seme rimane immutato tra due simulazioni successive, il risultato delle due simulazioni rimarrà invariato perché la successione delle estrazioni casuali sarà sempre lo stesso. Se il seme cambia, invece, due simulazioni avranno diversi risultati. Questa caratteristica è molto interessante, ma deve essere utilizzata in maniera corretta: se bisogna confrontare il comportamento di guidatori sottoposti a stesse condizioni esterne, la stocasticità tra le varie simulazioni deve essere

bloccata. D'altro canto se si vuole osservare come varia il comportamento del soggetto esaminato al variare del comportamento dell'ambiente circostante la stocasticità può essere impiegata in maniera interessante.

- I modelli di simulazione microscopica del traffico hanno generalmente uno specifico campo di applicazione, essendo realizzati con un preciso scopo modellistico. L'ambito applicativo più comune è quello extraurbano con infrastrutture stradali a carreggiate separate. I software di simulazione che lavorano bene in queste condizioni sono molteplici (i più famosi sono VISSIM [43], AIMSUN [10], CORSIM [24], Paramics [49]). Per esaminare, invece, condizioni di traffico extraurbane a singola carreggiata con correnti veicolari contrapposte, sono stati sviluppati solo un paio di modelli in letteratura (TWO PAS [33], TRARR [26], VTISim [51]). Esistono anche altri modelli di gestione del traffico che trattano altri specifici ambiti applicativi ([51], [17], [17]).

Dopo aver analizzato i punti precedenti, la tesi si concentra sulla descrizione del modello di traffico creato e sulla sua integrazione con l'ambiente di simulazione di guida. Tale struttura fornisce in input al modello di simulazione di traffico utilizzato la posizione e la velocità del veicolo interattivo, mentre in output restituisce le informazioni riguardanti gli altri veicoli presenti nell'ambiente circostante. Il modello di traffico creato è un modello di tipo meso-micro. Ad essere modellizzata a scala microscopica è solo la dinamica del traffico veicolare che si sviluppa entro una certa

distanza dal veicolo interattivo. Tale distanza deve essere almeno la distanza di visibilità in maniera tale che il guidatore del simulatore non veda spuntare dei veicoli dal nulla. Si considerano quindi due finestre in movimento, una più interna, in cui il comportamento dei veicoli debba risultare il più naturale possibile ed una più esterna in cui se ne percepisca solo la presenza (il loro movimento è regolato da un modello mesoscopico).

Particolare importanza è data alla calibrazione e validazione dei modelli ad un livello microscopico. Questo per assicurare:

- Il comportamento dei veicoli circostanti sembri realistico
- Gli output del modello di comportamento siano in accordo con dati reali
- A loro volta gli output macroscopici siano in accordo con i dati reali raccolti

L'obiettivo che si lo studio si è posto è quello di utilizzare un modello in grado di fornire tanto una corretta rappresentazione visuale del movimento dei veicoli autonomi, quanto una restituzione credibile del traffico nel suo complesso. È necessario quindi avere dati utili sui calibrare sia il comportamento di guida dei singoli veicoli sia il comportamento complessivo del traffico. Una volta che la struttura sia validata, è possibile istaurare un processo di continuo tuning dei parametri di guida, calibrando il modello di microsimulazione con i risultati ottenuti dal simulatore e quindi validando il comportamento che emerge dal simulatore stesso con il comportamento dei veicoli guidati dal micro-simulatore.

### **3.3 Conclusioni**

Le esperienze condotte in letteratura hanno molti interessi e punti comuni. Qui di seguito può essere riassunto quanto analizzato con l'indicazione delle scelte che si porteranno avanti nel seguito.

La creazione di un modello di simulazione ad hoc, è una soluzione, a nostro avviso, da scartare. Lo stato di avanzamento dei software commerciali è molto elevato e molti di essi offrono ampie soluzioni a molti dei problemi effettivamente modellizzabili. Tuttavia, viste le limitazioni intrinseche di ogni modello si può pensare di lavorare a migliorarne alcuni aspetti, anche se questo deve essere fatto in modo da non ostacolare la possibilità di aggiornamento continuo dei software. Il modo migliore per effettuare l'integrazione è, quella di preservare l'autonomia dei due strumenti senza inglobare l'uno nell'altro. Ancora una volta, questo è fondamentale per avere sempre la possibilità di seguire gli sviluppi software e per eventualmente sostituire uno dei due.

I software di simulazione del traffico devono essere calibrabili e validabili, sia da un punto di vista microscopico che macroscopico (devono essere in grado di riprodurre le grandezze aggregate quali flusso e velocità media e quelle disaggregate come traiettorie etc.).

Bisogna fare molta attenzione a progettare adeguatamente le comunicazioni in maniera tale che siano il più veloci e agili possibile. L'ideale sarebbe che le informazioni scambiate fossero in un formato di diffusione del tutto generale.

Le frequenze di calcolo tra i due modelli devono essere le stesse.

Le reti di base degli scenari devono essere le stesse per non inficiare i risultati della comunicazione tra i modelli.

Il microsimulatore dovrà gestire tutta la rete per garantire la congruenza del traffico, mentre al simulatore di guida andranno solo le informazioni inerenti al campo di visibilità del veicolo interattivo.

In pratica, mentre un micro-simulatore deve riprodurre correttamente il movimento dei veicoli per riprodurre il movimento del traffico di una rete, il modello di microsimulazione presente nel simulatore di guida ha lo scopo di fare in modo che il guidatore non sappia se il veicolo che gli viene incontro sia guidato da un essere umano oppure no.

Se si tengono presente queste importanti considerazioni, l'integrazione tra un simulatore di guida ed un modello di microsimulazione offre numerose opportunità:

1. facilitare la realizzazione di scenari con differenti condizioni di traffico
2. facilitare il cambiamento delle caratteristiche delle classi di veicoli presenti
3. rendere possibile la riproduzione di incidenti e condizioni di pericolo
4. rendere possibile la dotazione anche agli altri veicoli dello scenario delle stesse apparecchiature che si stanno testando con il simulatore di guida
5. rendere possibile la validazione del comportamento alla guida del simulatore confrontandolo con i comportamenti assunti dagli altri veicoli mossi dal microsimulatore
6. rendere possibile la calibrazione del microsimulatore attraverso i dati provenienti dal simulatore di guida

7. rendere possibile l'immersione del simulatore in un ambiente più realistico.

Vista l'esperienza condotta presso l'istituto norvegese per la sicurezza stradale (SINTEF) e viste, come già accennato, le nuove possibilità offerte dal software di simulazione microscopica del traffico utilizzato in quel caso, si è deciso di utilizzare, per realizzare l'integrazione, AIMSUN NG. Nel capitolo successivo si procederà quindi ad analizzare le possibili strategie per effettuare l'integrazione.



## **4. Analisi delle strategie di integrazione**

### **4.1 Introduzione**

Dopo aver testato il comportamento di AIMSUN, si è passati allo studio delle possibili soluzioni per portare avanti l'integrazione tra i due software. L'analisi incrociata delle procedure adottate in precedenti esperienze e delle possibilità offerte dai due ambienti ha definito due percorsi alternativi:

- Utilizzare le API di AIMSUN e le API di SCANeR: entrambi i software offrono un Application Program Interface che consente lo scambio di informazioni con l'esterno durante le simulazioni;
- Utilizzare le Enhanced API di AIMSUN e richiedere l'apertura del codice di SCANeR.

La seconda opzione è scaturita a valle di una serie di discussioni portate avanti con gli sviluppatori di AIMSUN. Secondo loro, infatti, questa soluzione è sicuramente da preferirsi, visto che le Enhanced API offrono numerose possibilità di interazioni in più delle normali API. Un'esperienza da loro precedentemente condotta presso l'università di Kejo, in Giappone, ha infatti mostrato risultati, sempre a loro giudizio, estremamente più performanti.

La scelta di questa opzione, però, comporta due problemi:

- da un lato richiede l'acquisto delle Enhanced API, di un software di sviluppo denominato QT e di una forma di supporto da parte di TSS, il cui costo varia da 3500 a 8000 €;
- dall'altro la necessità di disporre di aperture di SCANeR aventi le stesse caratteristiche delle Enhanced API di AIMSUN, cosa che l'OKTAL dovrebbe fornirci in via amichevole e confidenziale e che non è per nulla scontata.

L'esigenza di voler ottenere un prodotto innovativo con gli strumenti immediatamente a disposizione, ci ha spinto ad intraprendere la prima strada. La nostra scelta è stata anche dettata dal fatto che il supporto di AIMSUN non conosce SCANeR e le potenzialità delle sue API, e che quindi la strada da loro indicata potrebbe risultare vana. Ad ogni modo, vista la nostra scarsa competenza in tema di programmazione, una strada alternativa a quella intrapresa avrebbe richiesto necessariamente un intervento esterno per sperare in un buon esito del problema.

## **4.2 Le API di un software**

In un software, le API rappresentano delle “porte” che i produttori del software stesso lasciano aperte agli utilizzatori per consentire l'implementazione di moduli personalizzati. Attraverso le API, un utente può attingere delle informazioni particolari o può forzare il software a compiere delle operazioni differenti.

Le API rappresentano un'interfaccia aperta di un software, ovvero una particolare interfaccia che librerie, software o piattaforme possono usare per interagire con un programma. API è l'acronimo di Application

Program(ming) Interface (Interfaccia di Programmazione di un'Applicazione), indica ogni insieme di procedure disponibili al programmatore, di solito raggruppate a formare un set di strumenti specifici per un determinato compito. È un metodo per ottenere un'astrazione, di solito tra l'hardware e il programmatore, o tra software a basso ed alto livello. Le API permettono di evitare ai programmatori di scrivere tutte le funzioni dal nulla. Le API stesse sono un'astrazione: il software che fornisce una certa API è detto implementazione dell'API.

Le API permettono di espandere le funzionalità di un programma. Per uno sviluppatore mettere a disposizione un set di API di un suo software significa dare la possibilità ad altri di interagire con la sua piattaforma e, soprattutto, estendere le funzioni e le caratteristiche della struttura base della piattaforma.

Le API sono essenziali per i computer come gli standard elettrici lo sono per una casa. Chiunque può inserire la spina del tostapane nella presa a muro della sua casa o dal vicino perché entrambe le case sono conformi ad uno standard. Se non ci fosse una interfaccia standard, occorrerebbe avere una centrale elettrica per fare un toast. Niente vieta che esistano più tipi di interfacce diverse, per esempio un tostapane europeo non può funzionare negli Stati Uniti senza un trasformatore similmente ad un programma scritto per Microsoft Windows che non può essere eseguito direttamente su un sistema UNIX senza un API adapter come WINE.

Esistono vari design model per le API. Le interfacce intese per la massima velocità di esecuzione spesso consistono in una serie di funzioni, procedure,

variabili e strutture dati. Esistono anche altri modelli come gli interpreti usati per valutare le espressioni come con ECMAScript/Javascript. Una buona API fornisce una "scatola nera" o un livello di astrazione che evita al programmatore di sapere come funzionano le API ad un livello più basso. Questo permette di riprogettare o migliorare le funzioni all'interno dell'API senza cambiare il codice che si affida ad essa.

L'utilizzo delle API richiede la scrittura di codice in un linguaggio di programmazione definito dai produttori del software. Nel nostro caso per entrambi i software, il linguaggio di programmazione richiesto è il Visual C++. Di qui la necessità di acquisirne gli elementi base. Naturalmente, utilizzando le API, le possibilità di interazione con i software non è illimitata ma dipende dalla "dimensione" delle porte che i programmatori lasciano aperte. Nello specifico esistono delle funzioni che l'utente può richiamare all'interno del codice, che il programma riconoscerà all'atto della sua esecuzione. Un altro passo essenziale è stato quindi quello di conoscere le funzioni disponibili per capire le effettive potenzialità all'interno della struttura di integrazione. Questa fase è terminata con la presa coscienza che le API di entrambi i software offrono gli strumenti necessari a realizzare l'integrazione. Questo, lungi dal voler significare che attraverso di loro si potrà realizzare una struttura effettivamente capace della gestione integrata dei due strumenti, dà il via alla fase di progettazione concettuale dell'integrazione.

È bene sottolineare che, nel caso in cui l'utilizzo delle API non consentisse un'integrazione efficace ed efficiente, la strada da percorrere richiederebbe

l'apertura dei codici dei software cosa che, come detto, all'atto pratico appare molto difficile da ottenere.

### **4.3 Le API dei due strumenti**

#### **4.3.1. Le API di SCANeR**

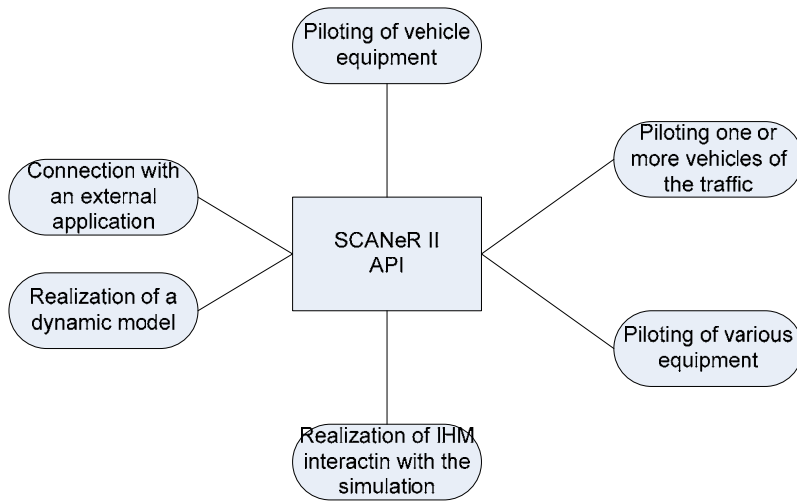
Le API di SCANeR sono state progettate per permettere agli utenti di personalizzare l'ambiente di simulazione con la creazione di propri moduli esterni. Le principali possibilità offerte sono le seguenti:

- Accedere alle informazioni cinematiche del veicolo interattivo;
- Accedere alle informazioni cinematiche dei veicoli autonomi;
- Accedere alle caratteristiche del tracciato;
- Accedere alla memoria di acquisizione condivisa che il software utilizza durante la simulazione;
- Aprire SCANeR a software e ad hardware esterni.

Lo schema di riferimento può essere quello riportato in Figura 4—1.

Le API di SCANeR possono essere suddivise in base alla specifica esigenza funzionale per cui sono progettate:

- Il controllo dei Processi;
- Il controllo della simulazione;
- Il controllo del tracciato stradale;
- Il controllo della comunicazione.



**Figura 4—1 Possibili applicazioni delle API di SCANer**

La parte più importante delle API è il controllo della comunicazione. SCANer II è, infatti, stato progettato come l'insieme di diversi moduli separati in comunicazione tra loro. Ogni modulo svolge un particolare compito all'interno della simulazione e la loro comunicazione è l'elemento essenziale per fare in modo che il loro operato fornisca un risultato consistente. Utilizzando le API si crea un vero e proprio modulo da mettere in comunicazione con tutti gli altri. Uno degli elementi fondamentali di una struttura di questo tipo è la gestione della comunicazione per lo scambio dei dati. Le strategie adottate da SCANer sono molto interessanti se si pensa ai fini dell'integrazione e sono due:

- *Asincrona*, in cui ogni modulo viaggia alla sua propria velocità inviando a SCANer costantemente il valore più aggiornato disponibile;

- *Pseudo-sincrona*, in cui SCANeR riceve i dati solo quando esplicitamente inviati dai vari moduli.

La presenza di strategie di questo tipo è molto importante perché non richiede la perfetta sincronizzazione dei due software e perché consente una certa flessibilità nella simulazione. Il software di simulazione microscopica del traffico tipicamente effettua i suoi calcoli ad una velocità superiore a quella della simulazione di guida (che viaggia in tempo reale). È però possibile che ciò non avvenga nella fase iniziale della simulazione in cui alcuni processi devono essere inizializzati contemporaneamente. Con le strategie a disposizione ciò non è un problema perché le simulazioni vanno comunque avanti e c'è la possibilità che il software di microsimulazione possa recuperare il gap maturato senza inficiare la rappresentazione dei veicoli.

Nella pratica il controllo della comunicazione deve *Inizializzare* la struttura (dichiarando Input, Output ed Eventi), *aggiornare* in continuo i dati da scambiare durante la comunicazione e *rilevare/definire* i valori dei dati richiesti.

I dati possono essere ricavati in due modi:

- Utilizzando i messaggi di rete;
- Utilizzando i dati della memoria condivisa.

I messaggi di rete vengono utilizzati per scambiare informazioni riguardanti il veicolo interattivo, i veicoli autonomi e tutti i dati che i vari moduli di SCANeR si scambiano utilizzando la rete ethernet di comunicazione come illustrato in Figura 4—2 .

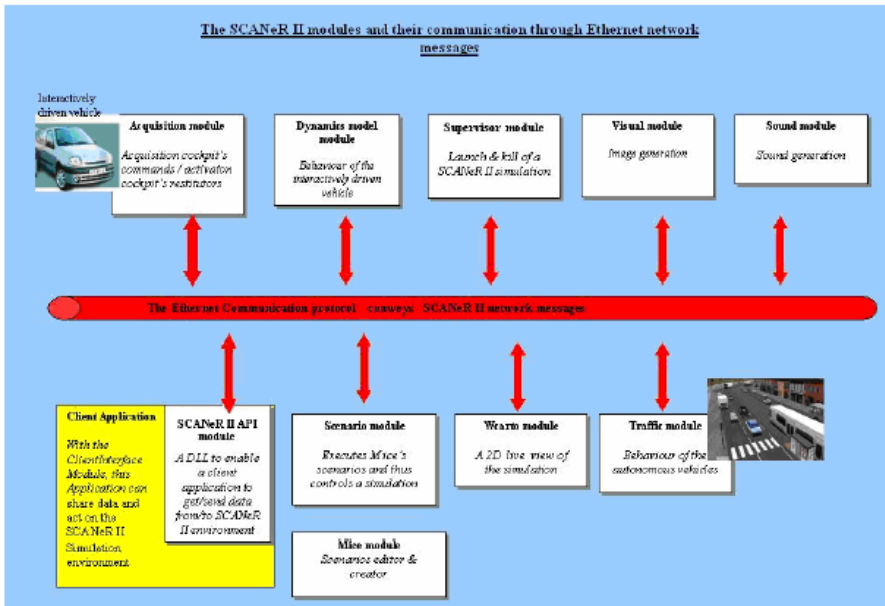


Figura 4—2 La struttura modulare di SCANeR

La memoria condivisa viene invece utilizzata per la comunicazione tra l'interfaccia di guida, la cabina, il modello dinamico e la piattaforma e segue lo schema di Figura 4—3.

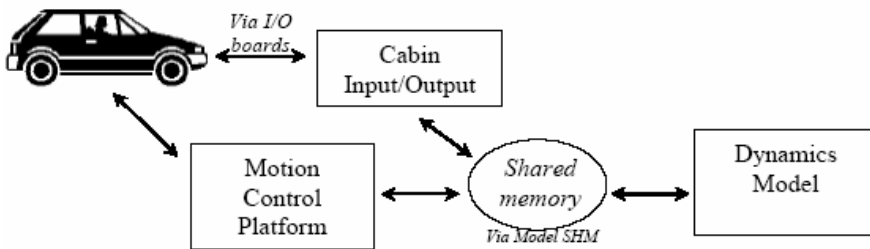


Figura 4—3 La struttura della Shared Memory



I dati richiesti da processo di integrazione sono quindi quelli ottenibili attraverso i messaggi di rete. Questo può rappresentare un limite visto che in questo modo la velocità di scambio è molto inferiore rispetto al caso della memoria condivisa. La velocità è, infatti, un parametro di notevole importanza nel processo di comunicazione. La simulazione del traffico viene effettuata con un intervallo di simulazione dell'ordine del decimo di secondo. Ogni decimo di secondo, quindi, deve avvenire l'intero scambio dei dati e ciò non è per nulla scontato.

L'accesso ai dati della rete di comunicazione richiede la realizzazione di una DLL che vada a costituire il modulo aggiuntivo che compia l'integrazione.

### 4.3.2. Le Api di AIMSUN

La struttura delle API di AIMSUN NG è invece differente. Attraverso le API non si crea alcun nuovo modulo, ma esse vengono utilizzate esclusivamente per ottenere delle informazioni da passare ad eventuali applicativi esterni (Figura 4—4).

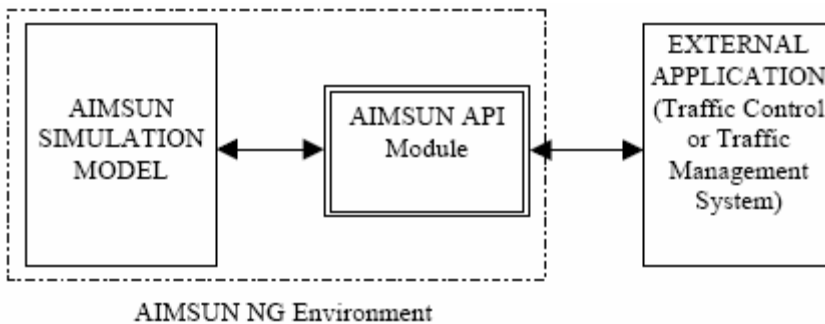


Figura 4—4 Ruolo delle API di AIMSUN

Anche in questo caso per realizzare l'interfaccia è necessario creare una DLL. In questo caso però la sua struttura sarà molto più rigida perché AIMSUN stabilisce sei funzioni all'interno delle quali inserire il codice per ottenere le informazioni desiderate (Figura 5).

Le informazioni reperibili riguardano praticamente tutti gli aspetti della simulazione e le operazioni eseguibili sugli oggetti presenti in AIMSUN sono numerose.

Le Application Program Interface dei due software ci forniscono quindi tutti gli strumenti necessari a compiere l'integrazione.

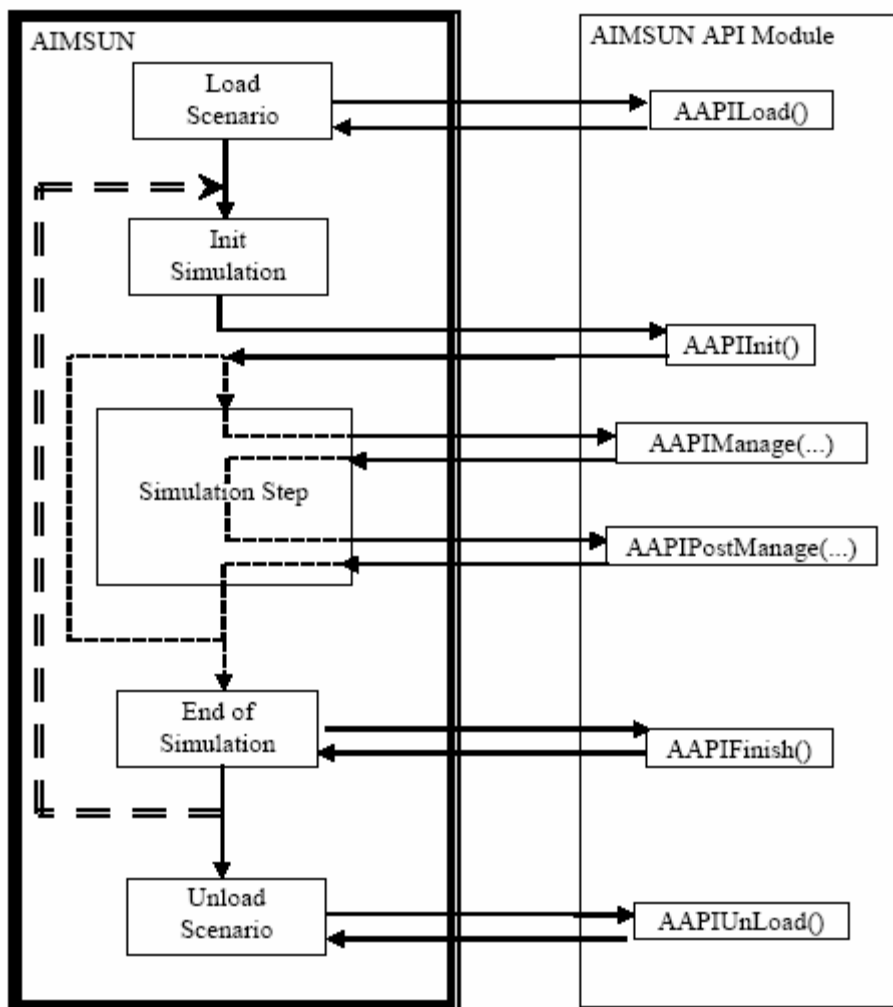


Figura 4—5 Schema di interazione di AIMSUN con le sue API



## 5. Progettazione e realizzazione della comunicazione tra AIMSUN e SCANeR

### 5.1 Introduzione

La realizzazione di un ambiente integrato di simulazione di guida e di traffico consiste nella realizzazione di una struttura di comunicazione tra le due applicazioni come mostrato in Figura 5—1. Tale comunicazione serve a scambiare le informazioni cinematiche dei veicoli autonomi della simulazione, quali ad esempio, la posizione, la velocità, l'accelerazione nelle sue tre componenti spaziali.

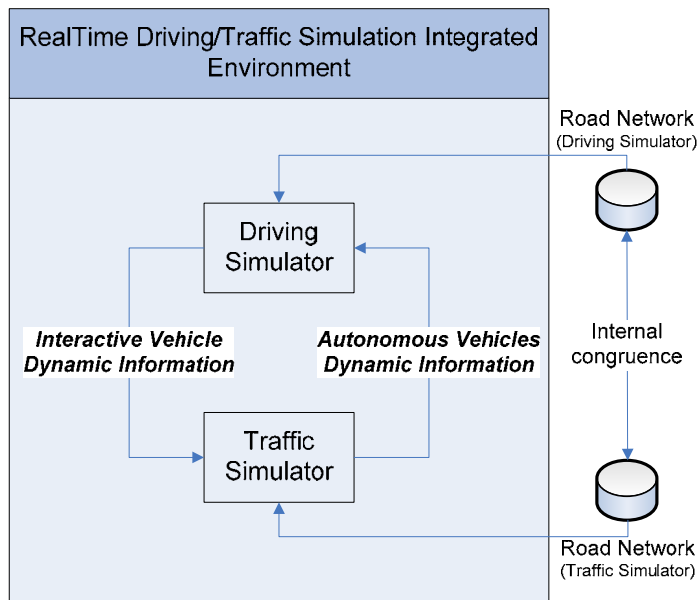


Figura 5—1 Rappresentazione schematica della struttura di integrazione

Perché lo scambio delle informazioni sia efficace è necessario che le reti stradali tra i due scenari siano consistenti sia per quel che riguarda le caratteristiche geometriche che per quel che riguarda il sistema di riferimento in cui sono rappresentate.

Le API dei due software, AIMSUN e SCANeR, secondo quanto visto in precedenza, forniscono, quindi, tutte le funzioni necessarie a realizzare tale comunicazione, cioè danno l'effettiva opportunità di reperire tutte le informazioni riguardanti le caratteristiche cinematiche di tutti i veicoli dello scenario in tempo reale durante la simulazione (l'elenco esaustivo di tutte le funzioni esistenti può essere reperito in [6] e [51]).

Il problema reale dell'integrazione, è legato al coordinamento della comunicazione tra i due ambienti.

Attraverso le API, infatti, non è possibile utilizzare uno dei protocolli di comunicazione esistenti (quali ad esempio il protocollo TCP/IP o UDP), ma bisogna lavorare imponendo cicliche azioni di attesa ai due software. Queste attese, essendo realizzate facendo compiere alle applicazioni (in particolare ad AIMSUN) dei loop di attività mediante comandi software, non rappresentano la maniera migliore di operare perché richiedono la costante utilizzazione della macchina anche quando non sia strettamente necessario, ponendo comunque dei problemi di utilizzazione della memoria.

Per questo motivo, l'unica soluzione perpetrabile con una speranza di successo è quella in cui le due API fossero inglobate in un unico elemento, in maniera tale da poter coordinare dall'interno le relazioni tra i due software ed in maniera tale da realizzare uno scambio di informazioni a livello di variabili di ambiente più che passando per file di testo o database.

Nel seguito si illustrerà brevemente l'approccio che si è tentato di implementare.

## **5.2 Le funzioni disponibili nelle API utilizzate nell'integrazione**

Nel paragrafo 4.3 sono state espone le caratteristiche delle API dei due software. In questa sezione, invece, vengono ripresi alcuni concetti, scendendo nel dettaglio di implementazione pratica.

Le API di AIMSUN, come detto, nascono con l'intento di definire specifiche azioni durante le simulazioni. L'approccio utilizzato è di tipo plug-in: l'utente di AIMSUN può creare una .dll che viene riconosciuta ed eseguita dal software. Questa .dll contiene diverse funzioni (come visto init, config, etc.) che rappresentano i punti di ingresso del software durante diversi momenti della simulazione. Istruendo queste funzioni, l'utente può eseguire le sue specifiche operazione sui risultati della simulazione. AIMSUN richiama queste funzioni alla sua specifica frequenza di calcolo.

```
extern "C" {
    DLLE int AAPILoad();
    DLLE int AAPIIInit();
    DLLE int AAPIManage(float time, float timeSta, float timTrans, float acicle);
    DLLE int AAPIPostManage(float time, float timeSta, float timTrans, float acicle);
    DLLE int AAPIFinish();
    DLLE int AAPILoad();

    DLLE int AAPIEnterVehicle(int idveh, int idsection);
    DLLE int AAPIExitVehicle(int idveh, int idsection);
}
```

**Figura 5—2 Esempio di funzioni richiamate da AIMSUN**

Il nodo cruciale di tutta la faccenda è che AIMSUN non ha un preciso intervallo temporale di calcolo. Esso, semplicemente, esegue tutte le computazioni più velocemente possibile e quindi, al variare del numero di

elementi della simulazione, varierà anche il tempo di calcolo. Per questo motivo, la sincronizzazione dei due software è così delicata ed importante.

A sua volta SCANeR dispone di API da compilare in linguaggio C++. come già visto, le possibili applicazioni di questi strumenti sono volti a:

- Connettere SCANeR con un'applicazione esterna;
- Realizzazione di un nuovo modello dinamico;
- Realizzazione di interfacce grafiche interagenti con la simulazione;
- Pilotaggio di più veicoli presenti nel traffico;
- Gestire diverse strumentazioni (Joystick, strumenti di aiuto alla guida, etc.)

```
int main(int argc, char* argv[])
{
    /*! Initialization of the Process **/
    Process_Init(argc, argv);

    /*! Process State **/
    APIProcessState status;

    do
    {
        /*! Wait for synchronization at the given frequency **/
        Process_Wait();

        /*! Run the process (and update the status) **/
        status = Process_Run();

        if (status == PS_RUNNING)
        {
            // Add user code
        }
    } while (status != PS_DEAD);

    Process_Close();
}
```

**Figura 5—3 Esempio di modulo di SCANeR**

L'utente di SCANeR utilizza le API per implementare un proprio modulo del programma. Tale modulo può lavorare alla sua frequenza oppure essere sincronizzato con un'applicazione esterna.



Nel momento in cui SCANeR lancia il modulo, passa al programma un intero e una stringa caratterizzanti il modulo stesso, dopodichè vengono eseguite le istruzioni che l'utente desidera eseguire.

### **5.3 I temi affrontati per l'integrazione**

Per la realizzazione del prototipo dell'ambiente integrato, numerosi sono stati i temi affrontati. Essi sono riportati di seguito.

#### **5.3.1. Accurata corrispondenza tra le reti stradali dei due ambienti**

I tool che uno strumento come AIMSUN mette a disposizione per rappresentare l'andamento del tracciato sono piuttosto elementari rispetto a quelli necessari a riprodurre in maniera accurata la configurazione reale e quindi bisogna porre molta attenzione durante la fase di costruzione della rete. Il passo principale è la definizione dell'andamento del tracciato in Autocad e la creazione di un file (.dwg) con la linea d'asse. Nel momento in cui un file di questo tipo viene importato in AIMSUN e in SCANeR il sistema di riferimento originario viene conservato. Il problema è che in SCANeR la linea d'asse importata sarà l'asse del tracciato di simulazione, mentre in AIMSUN rappresenterà solo un oggetto posizionato sullo sfondo. Il problema potrebbe essere aggirato importando uno shapefile (.shp) dal quale AISMUN è in grado di importare direttamente la configurazione del tracciato, ma al momento non si hanno informazioni sulle modalità di funzionamento del tool di importazione presente in SCANeR. Seguendo la prima procedura, una volta importato in AIMSUN, è necessario ricostruire il tracciato ripercorrendo in maniera quanto più fedele possibile la linea d'asse. È importante fare attenzione a suddividere il tracciato in tanti tronchi

aventi caratteristiche fisiche omogenee (stessa larghezza, stessa pendenza, etc.), in maniera da compiere meno approssimazioni possibili. Facendo molta attenzione e scegliendo di condurre l'editing con una curva di tipo Bezier, i risultati dovrebbero essere sufficientemente precisi per fornirci una buona rispondenza sulla posizione dei veicoli nei due ambienti di simulazione.

Se il problema, per quel che concerne l'andamento planimetrico del tracciato, con un po' di lavoro, è superabile, per quel che riguarda il profilo altimetrico, la situazione è differente. In AIMSUN l'unica cosa che può essere fatta è la definizione della quota iniziale e di quella finale di ogni segmento con interpolazione lineare dei punti intermedi. Questo vuol dire che per un'accurata rappresentazione del profilo longitudinale della rete è necessario creare segmenti stradali molto piccoli. Questo è un problema se si considera che il numero dei segmenti incide tanto sulla complessità computazionale della simulazione, quanto sui modelli di simulazione, in particolare su quello di lane-changing ([4]).

### **5.3.2. Simulazione del traffico i tempo reale**

Come in precedenza anticipato, AIMSUN non è progettato per lavorare in tempo reale. Al contrario, tutti i calcoli vengono eseguiti alla massima velocità compatibile con la complessità dello scenario di simulazione e con le caratteristiche della macchina sui cui il software è installato. Ad ogni modo in genere il tempo di simulazione è notevolmente inferiore al tempo reale e questo è un presupposto indispensabile per l'integrazione.

### 5.3.3. Sincronizzazione dei moduli di guida e di traffico

Come sarà mostrato nel seguito, per sincronizzare le due applicazioni, si è usato un trucco che rallenta AIMSUN. In pratica, ad ogni step di simulazione attraverso le API, si fa in modo che AIMSUN confronti il tempo di simulazione con il tempo reale finché questi non lo raggiunga. Ovviamente questo vale nell'ipotesi che il tempo di simulazione sia inferiore al tempo reale.

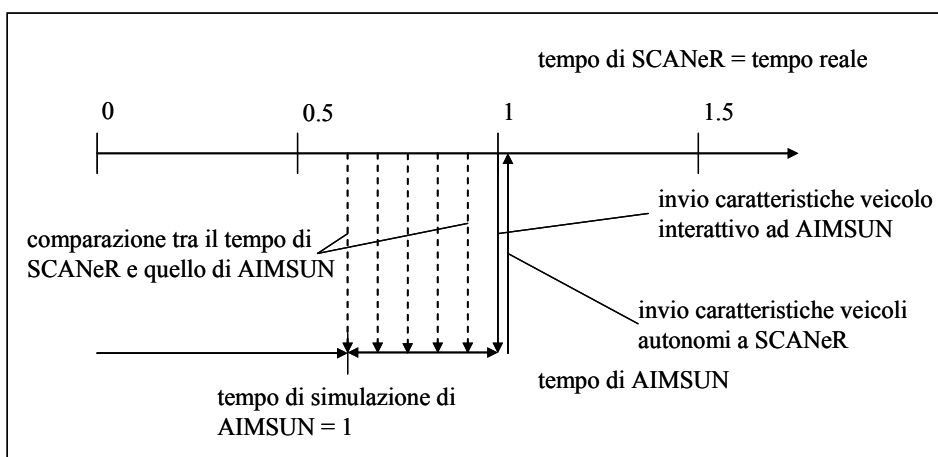


Figura 5—4 Schematizzazione della procedura di integrazione

### 5.3.4. Procedura di scambio delle informazioni in tempo reale

Sebbene l'ipotesi appena enunciata sia verificata praticamente sempre nelle normali applicazioni della microsimulazione del traffico, l'impatto della procedura di scambio dei dati sui tempi di calcolo può in effetti inficiare totalmente l'integrazione.

### **5.3.5. Consistenza della frequenza di aggiornamento dei dati**

Per quanto detto nel precedente capitolo sulle modalità di sincronizzazione delle comunicazioni in SCANeR, la possibilità di adottare una modalità di aggiornamento delle informazioni dei veicoli Pseudo-Sincrona consente lo scambio delle informazioni senza ostacoli. Il problema, semmai, come verrà esplicitato in seguito, è legato all'impossibilità di utilizzare una frequenza di aggiornamento maggiore di 10Hz (troppo bassa per una realistica restituzione visiva dei veicoli).

### **5.3.6. Gestione della visualizzazione dei veicoli autonomi**

Un software di simulazione microscopica del traffico è in grado di gestire migliaia di veicoli contemporaneamente, ma trasferire le informazioni di tutti i veicoli al simulatore sarebbe eccessivamente pesante oltre che inutile. In realtà, le uniche informazioni da scambiare sono quelle relative ai veicoli che si trovano all'interno della distanza di visibilità del veicolo interattivo. Per individuare questi veicoli, si possono utilizzare in AIMSUN le funzioni di *Vehicle Tracking* presenti nelle API, mentre in SCANeR, prima di far partire la simulazione si può creare un certo numero di veicoli (trentaquaranta) da richiamare di volta in volta all'atto di apparire e di scomparire dal tracciato.

Un altro problema a questo punto è la necessità di creare una corrispondenza di tipo uno a uno tra i veicoli di SCANeR e i veicoli di AIMSUN di cui si acquisiscono le informazioni. Questo è possibile solo mantenendo memoria delle informazioni tra due step di simulazione

successivi e creando una procedura in grado di verificare la corrispondenza esistente tra i veicoli.

## **5.4 L'architettura software**

Il problema principale è la rapidità delle comunicazioni. L'utilizzo di un database esterno o comunque di una fonte di dati estranea dai due strumenti può davvero rendere vana l'integrazione. In aggiunta, come detto, è preferibile utilizzare degli strumenti standard dei due software per non inficiare la possibilità di aggiornamento dei due strumenti. Per questo motivo si è scelto di utilizzare le API ma, come si vedrà qui di seguito, in maniera particolare con la definizione di un unico ambiente software.

### **5.4.1. L'ambiente condiviso**

L'idea chiave dell'integrazione è stata l'utilizzare un unico ambiente di sviluppo condiviso. In pratica, come accennato, si è costruito un modulo di SCANeR all'interno di una .dll che serva da Plug-in di AIMSUN.

In Figura 5—5 sono mostrati alcuni comandi di un API di SCANeR da inserire nella .dll di AIMSUN. In pratica, quando AIMSUN richiama le funzioni dichiarate all'interno della sua API, contemporaneamente esegue anche le funzioni di SCANeR. In questo modo è come se AIMSUN stesso fosse il nuovo modulo di SCANeR e questo è davvero interessante da un punto di vista metodologico. In Figura 5—6 è mostrato lo schema concettuale della struttura creata.

```

int main(int argc, char* argv[])
{
    Process_Init(argc, argv);
    APIProcessState status;
    do
    {
        Process_Wait();
        status = Process_Run();
        if (status == PS_RUNNING)
        {
            // Add user code
        }
    } while (status != PS_DEAD);
    Process_Close();
}

```

Da inserire nella funzione AAPINIT  
 Non deve essere richiamata perché la sincronizzazione deve essere comandata dal plugin  
 Da inserire in AAPIMange  
 Da richiamare nella funzione AAPILoad

Figura 5—5 Modifiche da apportare al modulo di SCANeR

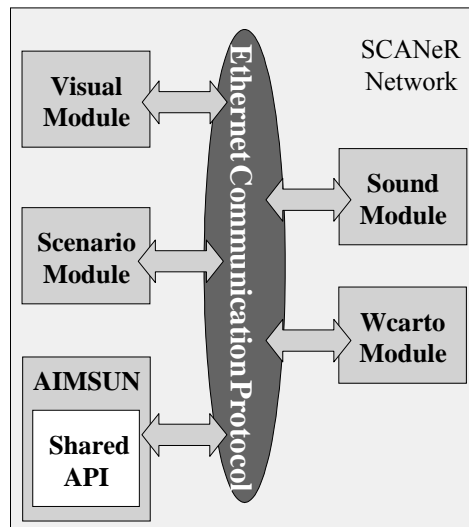


Figura 5—6 Rappresentazione concettuale della struttura di integrazione

Ritornando alle API di AIMSUN, ricordiamo che durante la sua esecuzione l'applicazione richiama sei funzioni [6]:

- AAPILoad();
- AAPIInit();
- AAPIManage(...);
- AAPIPostManage(...);
- AAPIFinish();
- AAPIUnload.

Queste funzioni sono state utilizzate per realizzare l'integrazione nel seguente modo:

- AAPIInit è una funzione richiamata solo all'inizio della simulazione. È stata utilizzata per (i) caricare il modulo di SCANeR, (ii) ricavare la posizione iniziale del veicolo interattivo, (iii) creare un *tracked vehicle* in AIMSUN e (iv) imporre che la sua posizione sia uguale a quella del veicolo interattivo; quindi (v) definire la posizione iniziale di tutti i veicoli presenti in SCANeR.
- AAPIManage è una funzione richiamata ad ogni step di simulazione. È qui utilizzata per (i) sincronizzare le due applicazioni, (ii) recuperare la posizione del veicolo interattivo in SCANeR e quindi (iii) aggiornare la posizione del tracked vehicle in AISMUN.
- AAPIPostManage è una funzione a sua volta richiamata ad ogni simulation step, in particolare alla fine dello stesso. Viene utilizzata per (i) individuare i veicoli presenti nell'intorno del tracked vehicle, (ii) ricavarne le caratteristiche cinematiche, (iii) verificare se durante il precedente step di simulazione tale veicolo

era già nell'intorno e quindi individuare il veicolo di SCANeR da utilizzare o chiamarne in causa uno nuovo, (iv) imporre le caratteristiche cinematiche dei veicoli individuati ai veicoli di SCANeR.

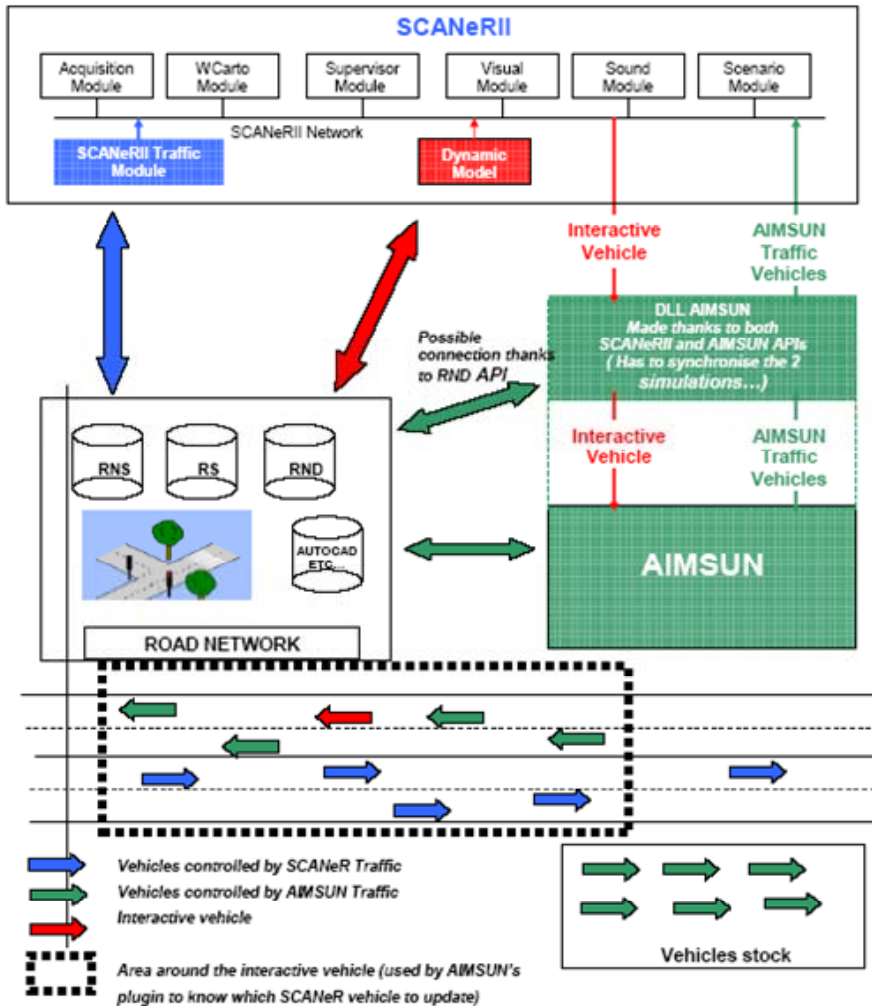


Figura 5—7 Quadro riassuntivo della struttura di integrazione

Un quadro riassuntivo della struttura di integrazione è riportata in Figura 5—7.



## 5.5 Prima applicazione

### 5.5.1. Descrizione del test

Il primo test è stato condotto su un tracciato pianeggiante a doppia carreggiata lungo 6.5km. Un traffico di 1000 veic/h è stato considerato per la simulazione. Le versioni dei software utilizzate sono state AIMSUN 5.1.8 e SCANeR 2.18. La distanza di visibilità considerata è stata di 200m. L'integrazione si è rivelata efficace per un numero di veicoli di SCANeR inferiore a 30. Al di sopra di 30, le procedure di ricerca tra i veicoli hanno rallentato eccessivamente il microsimulatore di traffico. Ad ogni modo 25 veicoli sono risultati sempre sufficienti a simulare i veicoli autonomi di contorno. In Figura 5—8 è mostrato il risultato dell'integrazione attraverso la corrispondenza dei veicoli nei due ambienti.

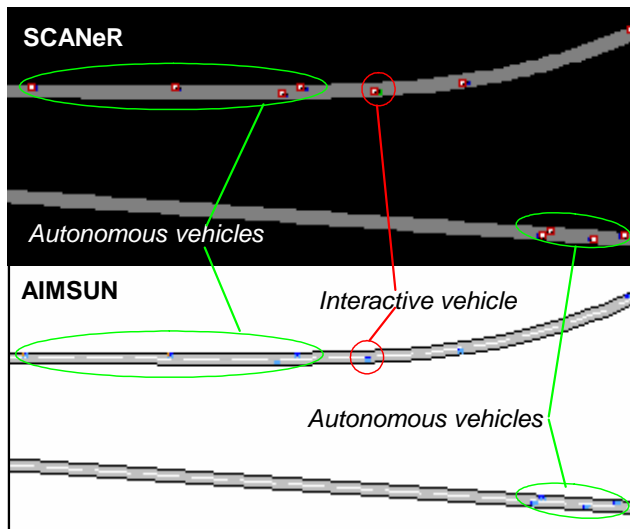


Figura 5—8 Corrispondenza dei veicoli nei due ambienti di simulazione

### **5.5.2. Problemi riscontrati**

Malgrado il generale buon esito della prima esperienza condotta, una serie di problematiche sono state incontrate. Ad ogni modo, per molte di loro si attende una rapida risoluzione con l'uscita della prossima versione di AIMSUN (la n° 6).

#### **Bassa frequenza di aggiornamento**

Come già anticipato, la massima frequenza consentita in AIMSUN è 10Hz. Tale vincolo non ha particolare significato fisico, ma è dettata dal fatto che una frequenza maggiore non trova giustificazione nelle pratiche di microsimulazione. Ad ogni modo la struttura della versione 5.1.8 di AIMSUN non consentirebbe una simulazione efficace ad una frequenza maggiore. D'altra parte, dal punto di vista visuale, è necessario garantire una frequenza di almeno 30Hz e quindi questo rappresenta il principale ostacolo attualmente presente.

#### **Comportamento di lane-changing dei veicoli**

Il principale limite modellistico presente in AIMSUN per quel che riguarda il comportamento dei veicoli è la rozza rappresentazione dei cambi corsia. In aggiunta, in AIMSUN i veicoli occupano esattamente il centro di ogni corsia e questo non aiuta la rappresentazione realistica del movimento dei veicoli. Questi problemi, ad ogni modo, sono già stati affrontati e risolti in AIMSUN 6.

## **Simulazione del traffico su strade a singola carreggiata e ad una corsia per senso di marcia senza vincolo di sorpasso**

Questo rappresenta l'unico elemento che non troverà a breve una risposta, con la conseguente limitazione delle applicazioni possibili del simulatore.

### **5.6 Conclusioni**

Durante la progettazione della struttura di integrazione tra ambiente di simulazione di guida e di traffico, si è seguita prima di tutto la logica della fattibilità tecnica ed economica. Una volta individuata l'opzione di scelta che meglio si adattasse alle immediate capacità del centro, anche grazie allo studio della letteratura di riferimento, si sono andati ad analizzare e ad affrontare tutti i problemi pratici individuati per l'integrazione.

I continui colloqui portati avanti con gli amici del supporto tecnico di AIMSUN e di SCANeR, hanno consentito la realizzazione di un primo prototipo di integrazione. Il principale risultato di questo prototipo è stato l'aver provato la fattibilità dal punto di vista computazionale della comunicazione tra le due applicazioni. I problemi legati alla visualizzazione ed al comportamento in fase di lane-changing, allo stato attuale impediscono che l'integrazione possa venire utilizzata in fase sperimentale. Ad ogni modo l'esperienza è stata comunque portata avanti visti gli sviluppi riscontrati nella prossima versione del software.



## **6. Validazione dell'ambiente integrato**

### **6.1 Introduzione**

La parte probabilmente più importante di un lavoro di questo tipo consiste nella validazione delle strutture realizzate.

Come già detto alla fine del Capitolo 5, però, l'aspetto che può apparire più ovvio per la validazione di quanto portato a termine, e cioè lo studio dell'impatto che tale ambiente integrato può avere sugli utenti non è stato compiuto per le due ragioni precedentemente espresse:

- Impossibilità di usufruire di frequenze di aggiornamento superiori ai 10Hz in AIMSUN 5.1 con conseguente non accettabile restituzione visiva dei veicoli al simulatore di guida;
- Rappresentazione del cambio di corsia ancora troppo rozza nel modello di simulazione microscopica del traffico con conseguente ancora troppo poco raffinata simulazione del movimento dei veicoli.

Come già accennato, però, la nuova versione di AIMSUN, rilasciata in versione beta e già in questa sede esaminata, dovrebbe fornire un contributo importante alla risoluzione di entrambi gli aspetti.

La parte di validazione su cui si è insistito moltissimo e che è stata sviluppata contemporaneamente alla realizzazione della struttura di comunicazione è stata relativa allo studio di AIMSUN e dei modelli di cui è costituito. In particolare ciò che è stato lungamente sperimentato ha

riguardato l'effettiva capacità di AIMSUN di simulare realistiche/reali condizioni del deflusso veicolare (uno degli aspetti, cioè, non ancora affrontati a fondo nella letteratura di riferimento dei simulatori di guida). Per questo motivo si è messa a punto una procedura di ottimizzazione globale per funzioni non convesse che cercasse quei valori dei parametri del modello (se esistono) che riproducono al meglio le condizioni reali in termini di flussi e velocità ai vari detector posti sulla rete. Nel presente capitolo, il background esaminato, la metodologia adottata e i risultati ottenuti saranno presentati e discussi.

## **6.2 Approccio alla calibrazione di un modello di simulazione microscopica del traffico**

Per valutare l'effetto dell'attuazione di politiche sul sistema dei trasporti si fa sempre più affidamento a modelli di simulazione microscopica del traffico. Per questo motivo numerosi sforzi sono stati compiuti nel campo della ricerca di settore nello sviluppo di pacchetti software in grado di soddisfare le sempre maggiori richieste esistenti. Questo ha comportato la realizzazione di strumenti sempre più sofisticati e complessi dipendenti da un numero sempre maggiore di parametri. Per questo motivo, l'esigenza di eseguire un'accurata calibrazione di tali strumenti, prima del loro utilizzo risulta essere molto forte.

La fase della calibrazione di un software di simulazione microscopica del traffico è probabilmente la più complicata nella modellizzazione di uno scenario del traffico. La complessità delle condizioni ambientali e dei fenomeni, e la difficoltà/impossibilità nel raccogliere dati diretti sulle dinamiche che si vanno a riprodurre, sono riconosciuti come i principali

ostacoli per il problema; se si aggiunge l'assenza di strumenti appropriati, tra i software di simulazione esistenti, si ha un chiaro quadro della situazione.

Esiste quindi, una spinta della ricerca sperimentale, di procedure di calibrazione affidabili e snelle che consentano l'assegnazione ai parametri dei valori più adatti a rappresentare la configurazione in esame.

Per questo motivo una parte consistente di questo lavoro è stata dedicata all'implementazione di una procedura sistematica per la calibrazione di un modello di simulazione microscopica del traffico.

Inoltre, un modello calibrato su certi dati, deve essere in grado non solo di riprodurre gli stessi dati, ma in generale di riprodurre il comportamento complessivo del sistema. In questo senso è possibile dire che la calibrazione è una parte essenziale di una struttura di valutazione più grande che deve comprendere necessariamente anche una fase successiva di validazione del modello. Di questa esigenza bisogna tener conto sin dalla fase della raccolta dei dati di campo necessari per effettuare la calibrazione. Nello specifico è infatti necessario adottare uno schema di campionamento duale; in cui si prevede che vengano utilizzati due differenti set di dati, uno per la calibrazione ed uno per la validazione. Come si vedrà di seguito, la struttura di calibrazione sviluppata consiste nei seguenti step:

- *Analisi di Sensitività del modello di simulazione microscopica del traffico*: si necessita l'individuazione di quei parametri per i quali il software risulta essere maggiormente "sensibile" e che spiegano la gran parte della varianza degli output del software stesso.

- *Progettazione della struttura di Calibrazione:* scelto l'algoritmo di ottimizzazione e le MoP da utilizzare per la calibrazione, è stato necessario creare un'appropriata procedura di calcolo in grado di implementare l'algoritmo stesso.
- *Validazione dei dati disponibili:* prima di utilizzarli all'interno della procedura di calibrazione è stata effettuata una verifica sulla consistenza dei dati rilevati.
- *Calibrazione del modello:* sono state eseguite due operazioni, la prima che mira ad una verifica del processo di ottimizzazione, "cuore della procedura", ed una seconda di applicazione della stessa a dati reali.
- *Validazione:* è stato verificato che il software con i parametri calibrati fosse in grado di riprodurre delle condizioni di traffico quanto più reali possibile.

Nei paragrafi successivi sarà dapprima esaminato lo stato dell'arte dei differenti approcci seguiti in letteratura per la calibrazione di modelli di simulazione microscopica del traffico. Sarà quindi descritto in maniera dettagliata l'intero processo applicato al caso studio di nostro interesse; ci sarà quindi la presentazione della rete e delle condizioni sperimentali al contorno, l'autostrada E32 Napoli - Salerno, in corrispondenza di due tratte Portici -Torre Annunziata e Angri – Scafati. Saranno quindi illustrati i modelli e i parametri caratterizzanti il software di simulazione microscopica utilizzato, che ovviamente è stato AIMSUN NG v. 5.1. A questo punto sono riportati i risultati dell'analisi di sensitività effettuata sul modello e dei relativi piani sperimentali redatti per condurla, secondo la teoria DOE ed



analizzata con metodi statistici, che ne garantiscono la consistenza. C'è quindi la definizione dell'algoritmo di ottimizzazione che porta a compimento l'intera procedura di calibrazione dei parametri definiti maggiormente sensibili. Infine saranno riportate le conclusioni derivanti dai risultati ottenuti.

### **6.3 Stato dell'arte**

Di seguito si farà una digressione abbastanza dettagliata sulle tematiche della simulazione microscopica del traffico e della calibrazione in generale. Partendo da un'analisi globale sull'importanza del processo di calibrazione, si passerà ad una fase di dettaglio della procedura stessa; quindi, si presenteranno i modelli, che, nelle loro singole specificità, sono il motore dell'intero processo, e la cui applicazione rappresenta il punto critico. Si esibirà, infine, un excursus sulla letteratura esistente in materia.

#### **6.3.1. Il concetto generale di calibrazione**

Un processo comporta differenti passaggi, tra i quali la fase senz'altro più significativa è quella progettuale di calibrazione. Tale affermazione non è azzardata, anzi, a ben vedere, non fa altro che riprendere quanto, dalla seconda metà degli anni '80 ad oggi, si propone come chiave di volta del management.

“Smettere di dipendere esclusivamente dal controllo in produzione per ottenere qualità”, nel senso di preoccuparsi della qualità dei prodotti durante la loro progettazione, piuttosto che, tardivamente, durante la loro produzione; questo scriveva W. Edwards Deming, statistico americano, padre del settore manageriale.

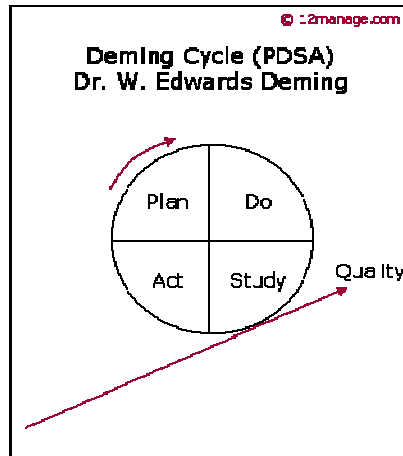


Figura 6—1 Rappresentazione del Ciclo di Deming

*Ciclo di Deming:*

Suddivisione di un intervento migliorativo in quattro macrofasi: plan, do, study, action.

Potremmo, quindi, meglio evidenziare dei passaggi fondamentali dell'intero processo, rivisto alla luce dell'acquisita importanza della calibrazione:

- l'*ideazione* di un sistema, prendendo sempre in considerazione la possibilità di adottare una tecnologia migliore;
- la *progettazione* del sistema, sfruttando tutti i vantaggi dei parametri di progetto;
- *definire e controllare* le tolleranze di produzione, affinché la variabilità non scenda al di sotto di un certo livello.

Al giorno d'oggi, in una realtà sempre più veloce e sempre più competitiva, non ci si può non conformare ai capisaldi precedentemente esposti; in ogni

ramo della scienza e della tecnica, il livello di qualità del prodotto, quale elemento ultimo di una lunga sequenza, è ovviamente l'obiettivo chiave.

Ci si può rendere facilmente conto dell'importanza fondamentale del funzionamento dei processi produttivi e logistici. Tali sistemi sono, infatti, caratterizzati da elevata complessità, numerose interrelazioni tra i diversi sottoprocessi che li attraversano, guasti dei segmenti, indisponibilità, stocasticità dei parametri del sistema.

Spiegato il concetto di calibrazione e della sua importanza nel senso più assoluto, è doveroso fare una digressione su cosa agisce.

### **6.3.2. I modelli dinamici di simulazione del traffico**

L'invarianza nel tempo delle variabili che descrivono il funzionamento del sistema di trasporto, consente di darne una rappresentazione mediante un'unica fotografia; una rappresentazione che potremmo pertanto definire *statica*.

Il risultato di un'assegnazione di equilibrio, per esempio, fotografa quella particolare condizione di funzionamento del sistema in cui flussi e costi di percorso (o di arco) sono mutuamente consistenti e, come detto, stazionari. Una tale configurazione potrebbe essere osservata nel mondo reale solo se domanda, scelte di percorso e sistema di offerta rimanessero costanti per un periodo di tempo sufficientemente lungo, affinché il sistema possa raggiungere una condizione di funzionamento stazionario.

Le semplificazioni indotte da tale assunzione di stazionarietà, non consentono tuttavia di rappresentare il *funzionamento interno* del sistema; tale corrispondenza poi, è indipendente dalla storia precedente del sistema, motivo per cui tali modelli sono anche noti come modelli senza memoria.

Sono stati recentemente sviluppati diversi *modelli dinamici* di offerta, di domanda e di assegnazione, cui si dà generalmente il nome di modelli con dinamica *intra-periodale* (within-day), vista la loro attitudine a riprodurre le dinamiche interne al periodo di riferimento del sistema in esame.

La loro formulazione ed il loro livello di complessità dipende dal tipo di sistema di offerta interessato.

In questi ultimi, in particolare, le semplici relazioni algebriche che, nell'ipotesi di stazionarietà intra-periodale, correlano le variabili in gioco, sono in molti casi sostituite da equazioni differenziali che descrivono l'evoluzione nel tempo delle stesse variabili.

I primi modelli dinamici del deflusso stradale, proposti negli anni cinquanta del secolo scorso, fornivano una rappresentazione del deflusso stradale basata su di un'analogia con il deflusso idrico. In tale approccio i singoli veicoli sono trattati come un fluido continuo (mono-dimensionale), per il quale possono definirsi in ciascun punto dello spazio e del tempo variabili quali la portata, la densità e la velocità. L'evoluzione nel tempo di tali variabili di stato è modellizzata attraverso un'equazione alle derivate parziali che comprende tanto la conservazione della massa (i veicoli) quanto una relazione sperimentale fra portata e densità. In accordo ad una classificazione basata sul livello di dettaglio dei modelli, tale approccio alla modellizzazione del deflusso veicolare è chiamato *macroscopico*.

I modelli macroscopici possono essere ulteriormente divisi a seconda che forniamo una rappresentazione continua o discreta dello spazio, a differenza del tempo sempre trattato in maniera continua.

I modelli *mesoscopici*, invece, rappresentano il deflusso stradale ad un livello di dettaglio del singolo veicolo (o di un gruppo di veicoli

generalmente chiamato pacchetto). In tale caso, sebbene la rappresentazione del traffico sia discreta, il movimento di ciascun singolo veicolo dipende da legge che descrivono le relazioni fra le variabili deflusso del del globale (per esempio, media di velocità dell'in densità di della di funzione), o da funzioni probabilistiche (si vedano che i modelli basati sull'analogia governano che dei di cinetica della gassano, la che descrivono le dinamiche delle distribuzioni di velocità).

I modelli *microscopici* descrivono i movimenti dei singoli veicoli come il risultato di scelte individuali disaggregate e delle interazioni con gli altri veicoli e con l'ambiente stradale. La scelta del percorso, le decisioni di accelerazione o cambio corsia, il comportamento alle intersezioni ecc., di ciascun singolo veicolo, sono in generale modellizzati esplicitamente. Inoltre, ciascuna entità del flusso ha le sue proprie caratteristiche che possono includere: le caratteristiche del veicolo, come il tipo o l'accesso ad informazioni sul viaggio, le prestazioni del veicolo, quali la massima accelerazione o la massima velocità, le caratteristiche del guidatore, quali il tempo di reazione o la velocità desiderata etc.

La classificazione proposta è basata da un lato sulla rappresentazione del flusso di utenti (continua o discreta) e dall'altra sulle funzioni di prestazioni adottate (aggregate o disaggregate).

Rappresentazione del flusso	Funzioni di prestazione	
	Aggregate (capacità esplicita)	Disaggregate
Continua	MACROSCOPICI	
	spazio discreti	spazio continui
Discreta	MESOSCOPICI	MICROSCOPICI

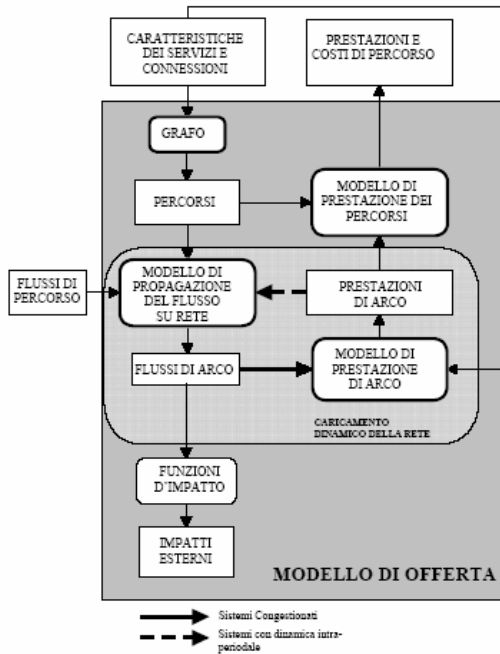
**Figura 6—2 Classificazione delle possibili modellizzazioni dell'offerta di trasporto**

Il campo di applicazione dei diversi approcci varia sensibilmente. Per esempio, i modelli macro- e mesoscopici sono particolarmente indicati per la simulazione del sistema stradale nella progettazione di strategie di controllo: la rappresentazione esplicita dello stato (*statespace modelling*) del sistema da essi consentita, può essere infatti facilmente inclusa in schemi di ottimizzazione.

Anche se funzione delle dimensioni della rete e delle capacità di elaborazione, generalmente il tempo di calcolo per tali modelli è sensibilmente inferiore al tempo reale, consentendone l'uso anche in applicazioni *on-line*. Nei modelli di simulazione microscopica, invece, i tempi di calcolo si incrementano notevolmente col crescere della congestione del sistema stradale – vale a dire con il crescere del numero di veicoli da simulare – restringendo generalmente il loro uso ad applicazioni di verifica *off-line*. A prescindere dalle considerazioni precedenti, la scelta del modello da utilizzare è in generale affetta dal livello di dettaglio richiesto dall'applicazione. Da notare che in tale considerazione nulla è riferito alla precisione attesa dei risultati. Il livello di dettaglio si riferisce alla fase modellistica e alla possibilità di meglio rappresentare l'evoluzione dei fenomeni, ma l'obiettivo atteso deve essere sempre quello di ottenere informazioni sull'efficienza di politiche o di interventi sul sistema dei trasporti a livello aggregato di flussi, livelli di congestione, velocità e quant'altro possa misurarsi sul sistema, in termini di prestazioni dello stesso.

Da un punto di vista modellistico, i modelli di offerta dinamici macroscopici e mesoscopici, che adottano funzioni di prestazioni aggregate, esprimono

flussi e prestazioni del sistema in funzione dei flussi di percorso e delle caratteristiche fisiche del sistema in maniera analoga al caso statico. La Figura 6—3 riproduce la struttura generale di un siffatto modello di offerta dinamico.



**Figura 6—3 Struttura generale modello di offerta dinamico**

Un discorso a parte meritano i modelli di offerta microscopica. Per questi ultimi, infatti, non è possibile individuare una struttura del tipo rappresentato in Figura 6—3. Come anticipato, le funzioni di prestazione di tali modelli sono infatti disaggregate, fanno cioè riferimento al singolo veicolo.

Per loro natura i modelli di deflusso microscopici nascono dall'integrazione di diversi modelli che descrivono il movimento del singolo veicolo

basandosi su definite ipotesi che possono discendere, a seconda del modello utilizzato, tanto da paradigmi comportamentali, quanto da evidenze sperimentali e statistiche. Tutti questi modelli, comunque, per loro natura, dipendono da una serie di parametri dal cui valore dipendono i risultati ottenibili. L'individuazione dei valori dei parametri che avvicinano la simulazione alla realtà, ovvero la fase della calibrazione del modello, è un tema in letteratura tutt'altro che risolto e di cui si parlerà nel seguito.

### **6.3.3. La calibrazione dei modelli di simulazione microscopica del traffico – breve stato dell'arte**

Dall'analisi dei lavori presenti in letteratura, la calibrazione dei modelli di simulazione microscopica del traffico viene effettuata in due modi:

- Tecniche razionali che implicano misure dirette dei parametri.
- Tecniche indirette in cui i valori dei parametri sono dedotti dai confronti tra output del modello e le osservazioni reali.

La prima utilizza tecniche di stima, usualmente modelli econometrici, per definire direttamente i singoli valori dei parametri, come input del modello di simulazione. Questo tipo di tecnica, ampiamente utilizzata, risulta estremamente pericolosa, visto che implica la fisicità dei parametri del modello. In realtà, nella maggior parte dei modelli utilizzati, i parametri hanno un significato prettamente statistico e, sebbene il valore da utilizzare debba essere fisicamente accettabile, non è detto che esprima effettivamente ciò che rappresenta. Un esempio lampante è dato dal Tempo di Reazione associato a ciascuna categoria di veicolo/classe di utenza. Molti modelli dipendono da tale parametro, ma se esso fosse ricavato ad esempio misurando il tempo di reazione effettivo del sistema conducente/veicolo nella realtà e lo utilizzasse in simulazione, i risultati che otterrebbe



sarebbero distanti da ciò che accade in realtà. Questo perché i modelli di simulazione microscopica del traffico non rappresentano realmente tutti fenomeni, le scelte, le decisioni che avvengono nel caso reale, ma, attraverso i loro parametri e i loro sottomodelli, devono mediare in senso statistico i fenomeni reali. Per questo motivo, in letteratura ha avuto un certo impulso l'idea di alcuni studiosi (soprattutto di quelli interessati di simulare il traffico nella simulazione di guida) di introdurre la categoria di modelli cosiddetti *nanoscopici* in cui andare realmente a calare tutti i processi fisici/decisionali che intervengono nella guida reale. La complessità di questo tipo di strutture ha, però impedito, che questi modelli si diffondessero e si pensasse davvero di utilizzarli.

Per quanto detto, l'unica soluzione è quello di utilizzare la seconda tecnica tecnica, indiretta, che in cui i valori dei parametri si ottengono implementando una strategia di ottimizzazione in grado di trovare quei valori in grado di minimizzare la distanza tra valori reali e valori simulati di grandezze aggregate rappresentative del deflusso veicolare quali ad esempio, portate, densità, velocità, lunghezza delle code o altro di quelle che saranno definite le Misure di Prestazione (MoP) del sistema. Tale tipo di tecnica necessita, ovviamente, di raccogliere dati reali per poter essere utilizzati, ma tali dati sono anche quelli generalmente più facilmente misurabili sul sistema dei trasporti, attraverso l'installazione di sensori ampiamente diffusi sul mercato. Se i dati sono abbastanza agevolmente reperibili, la loro qualità non è, purtroppo, sempre adeguata e delle opportune verifiche e correzioni devono purtroppo essere effettuate.

Un problema sui cui porre notevolmente l'attenzione è che il numero dei parametri da cui tali modelli sono influenzati è, usualmente, estremamente

elevato. Poiché è praticamente impossibile applicare una procedura di ottimizzazione su un modello molto complesso come quelli che si stanno considerando, su un numero molto elevato di parametri, occorrerebbe selezionarne un numero appropriato.

Un altro problema è rappresentato dalla forma funzionale e dalla MoP utilizzata all'interno della funzione obiettivo da minimizzare. Essa, deve essere rappresentativa dei fenomeni simulati; per ottenere anche una stima significativa dei parametri.

A partire da questi elementi si riportano i principali elementi reperiti nei più significativi lavori presenti in letteratura.

In [12] viene riportata la procedura di calibrazione di MITSIM sulla rete stradale di Stoccolma. L'approccio utilizzato è l'approccio sequenziale.

In una prima fase si isolano parti elementari di una rete (tronco) sulla quale entrino in gioco le sole dinamiche di deflusso e sulla quale è possibile raccogliere dati disaggregati molto dettagliati relativi a tali dinamiche, ad esempio le traiettorie dei veicoli per il car-following od anche altri parametri comportamentali per il lane-changing.

Utilizzando questi dati, è, quindi, possibile procedere alla stima dei parametri che regolano il deflusso (i parametri dei modelli di car-following, di lane-changing, di gap-acceptance, di merging, etc.).

Nella fase successiva l'attenzione è posta a porzioni di rete più estese in cui entrino in gioco le altre dinamiche legate al comportamento di viaggio, come la scelta del percorso o la distribuzione della domanda di spostamento. Tali dinamiche non sono in generale direttamente rilevabili, per cui si è soliti effettuare una stima indiretta, sfruttando dati di deflusso aggregati come portate e velocità ai detector.

L'utilizzo di dati molto disaggregati, come le traiettorie dei singoli veicoli per calibrare i parametri, ad esempio, dei modelli di car-following, presenta due difficoltà: (i) in primo luogo dati di questo tipo sono molto difficili da reperire e nella pratica professionale la possibilità di utilizzo di questi modelli sarebbe praticamente nulla., e (ii) non è detto che i parametri ricavati in questo modo ci forniscano una simulazione adeguata del deflusso veicolare a scala macroscopica.

Ad ogni modo, ritornando, a parlare dell'approccio sequenziale utilizzato, la scelta di appositi sottosistemi “chiusi” (ossia di breve tratte delle quali si conoscano tutte le condizioni al contorno), oltre a fornire una maggiore maneggevolezza, garantisce la conoscenza di tutti i flussi in ingresso ed in uscita, vale a dire di una matrice OD accurata, ha consentito di svincolarsi dalle problematiche connesse alla scelta ed alla modellizzazione dei flussi di domanda od.

Un approccio simile è stato utilizzato in [31] e [19]. Nel primo lavoro la stima dei flussi di domanda viene applicata, ad una rete di dimensioni ancora più elevate; in queste trattazioni insomma la matrice OD rappresenta all'inizio uno degli input da fornire al modello entrando quindi a far parte del processo di calibrazione in un secondo momento. In questo caso, però, i parametri dei modelli di deflusso sono ricavati non calibrandoli sulle traiettorie, ma su dati aggregati in cui tali dinamiche fossero le uniche presenti, ovvero tratte autostradali in cui tutte le informazioni fossero note e disponibili.

Medesime sono le problematiche che si trovano in [8], ancora concentrato sulla calibrazione di un modello di microsimulazione su larga scala. Il problema della calibrazione è affrontato per gestire un problema

maggiormente settoriale, riguardante l'ottimizzazione dei piani temporali delle intersezioni semaforizzata.

In [12], invece, grande attenzione è rivolta al concetto che un modello di simulazione non potrà mai riprodurre la realtà in maniera fedele, ma solo in quanto approssimazione; che ogni modello di simulazione debba essere sviluppato per il particolare contesto che si sta analizzando, che le MOP (Measure Of Performance) che si utilizzano per la validazione siano effettivamente rappresentative del contesto stesso ed infine che lo sviluppo e la validazione del modello stesso siano eseguiti congiuntamente allo studio che si effettua. In particolare gli autori si soffermano sui passi compiuti nella definizione del modello (nello specifico AIMSUN), e nell'analisi di quelli che sono i passi metodologicamente chiave per rendere tale modello valido e credibile. Ciò che gli autori sottolineano essere essenziale è :

- La rappresentazione geometrica di ciascun componente della rete di traffico stradale, nonché di tutte le strumentazioni ad esse collegate.
- La rappresentazione degli schemi di controllo e gestione del traffico (fasi, tempi-direzioni dei movimenti dei veicoli, svolte permesse, ed altro).
- I modelli comportamentali dei singoli veicoli: car-following, lane-changing, gap-acceptance ...
- La rappresentazione della domanda di traffico.
- Il modello di scelta del percorso, per simulazioni basate su matrici OD.

Gli autori di [27] hanno come scopo quello di presentare una metodologia rigorosa nella calibrazione dei microsimulatori. Si fa l'esempio di microsimulatori quali INTRAS, MITSIM, PARAMICS come casi simbolici dell'adattamento allo specifico simulatore.

La trattazione si pone, però, come importante chiave di lettura nella calibrazione di AIMSUN. È possibile individuare un primo step del processo, che riproduce i flussi esistenti sulla rete; noti i flussi in ingresso, è necessario calibrare i parametri che influenzano il comportamento dei guidatori per fare in modo che, una volta propagato, il flusso simulato in una sezione a valle sia lo stesso di quello rilevato.

Il secondo passo individuato, prevede che le velocità rilevate siano quanto più possibile prossime a quelle simulate, nell'ottica che il nostro modello sia quanto più reale possibile anche riguardo agli accodamenti (velocità che quindi dovranno essere superiori od inferiori ad una certa soglia).

Il lavoro è molto completo, ma presenta una metodologia forse troppo complessa per fornire un'opportunità di utilizzo praticamente effettiva.

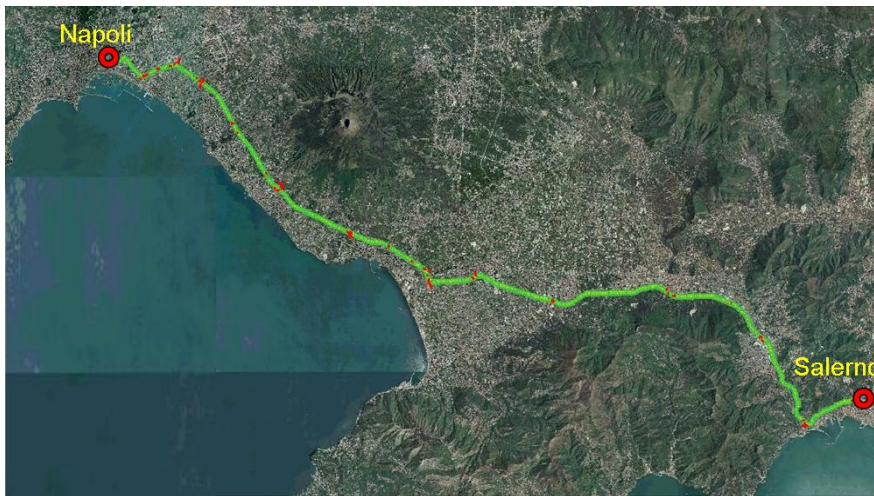
Altri studi sono presenti e disponibili in letteratura mostrano ulteriori aspetti più o meno interessanti del problema. In particolare per chi dovesse essere interessato all'argomento si suggeriscono i lavori presenti in bibliografia ([1], [44], [29], [37], [33], [10], [38] and [52]).

Per maggiori approfondimenti e per una più sintetica e puntuale descrizione del problema si rimanda ai lavori pubblicati riportati in Appendice.

Per quanto le esperienze riportate in letteratura non fossero in numero trascurabile, si è giudicato interessante provare a definire dal principio una metodologia di calibrazione che comprendesse tutta una serie di fasi in letteratura assenti o comunque giudicate di secondaria importanza.

## **6.4 Studio di campo eseguito**

L'obiettivo del lavoro è testare la fattibilità su di una rete reale di una procedura di calibrazione automatica. La raccolta dei dati necessari per la calibrazione è stata resa possibile con l'uso degli strumenti a disposizione sulla Autostrada E32 tra Napoli e Salerno nell'ambito del progetto n. 12897 Legge 593/2000 Art. 12 PON 2000-2006; TEMA 15-TRASPORTI: "Sistema di monitoraggio, controllo ed informazione per la gestione attiva della sicurezza di un'infrastruttura autostradale".



**Figura 6—4 Scenario utilizzato per la sperimentazione**

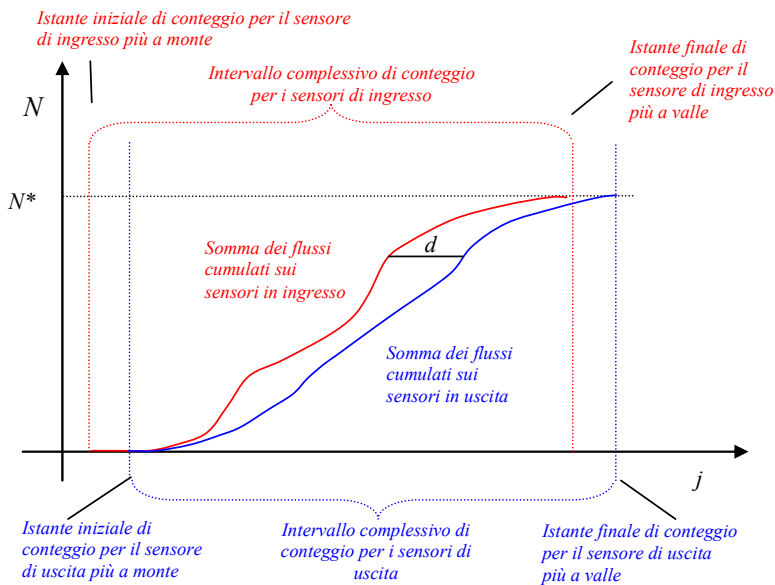
### **6.4.1. Introduzione all'ambito sperimentale**

Particolare attenzione è stata posta alla qualità dei dati rilevati. Vista la mole dei rilievi disponibili, sono state implementate in automatico alcune verifiche di congruenza dei dati.

Tali verifiche hanno consentito di evidenziare e correggere errori sistematici di misura non evidenziati nella fase di calibrazione dei sensori.

Sono verifica di accuratezza e congruenza (consistency) dei dati rilevati dai sensori. In particolare, al fianco della usuale calibrazione dei sensori è stata effettuata una verifica di congruenza dei flussi ai varchi di ingresso e di uscita di ciascun sistema chiuso.

Con riferimento ad un intervallo giornaliero si sono costruiti i diagrammi cumulati della (somma) dei flussi in ingresso e della (somma) dei flussi in uscita traslati del tempo necessario a percorrere la distanza fra i sensori al momento di inizio del conteggio.



**Figura 6—5 Schema per l'analisi di consistenza dei dati**

Le verifiche impostate rispondono a tre ovvie considerazioni effettuate guardando alla Figura 6—5:

- che in ciascun istante la curva cumulata degli ingressi sia al di sopra di quella delle uscite;
- che il valore giornaliero dei due flussi cumulati risulti uguale;

- che per ciascun valore dei flussi cumulati le due curve siano traslate di un tempo non inferiore al tempo di percorrenza a flusso nullo fra il sensore di ingresso più a valle ed il sensore di uscita più a monte.

In particolare, indicando con  $N_{s,j}^i$  il flusso di veicoli appartenenti alla categoria  $i$  transitati fino all'istante  $j$  e rilevati dal sensore  $s$ , la prima verifica può scriversi:

$$\sum_{s \in S_{uscita}} N_{s,j}^i < \sum_{s \in S_{ingresso}} N_{s,j}^i \quad \forall i, j$$

La seconda verifica effettuata sui volumi giornalieri:

$$\sum_{s \in S_{ingresso}} N_{s,T_s}^i = \sum_{s \in S_{uscita}} N_{s,T_s}^i \quad \forall i$$

dove  $T_s$ , è l'istante finale dell'intervallo di misura del sensore  $s$ .

Infine la terza verifica:

$$d \leq t_{(ingresso-uscita)}^0$$

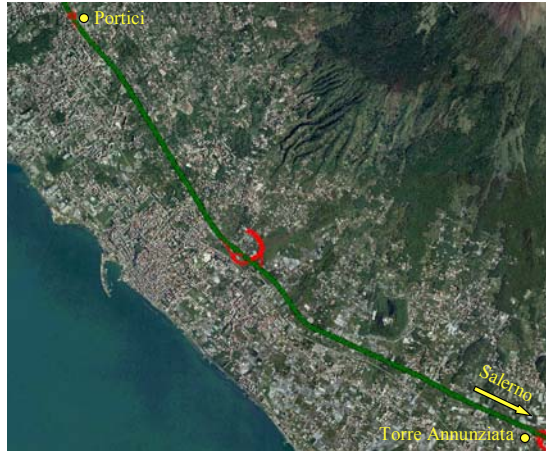
I dati osservati, raccolti in un'intera settimana, si sono rivelati inconsistenti; nel particolare il flusso di traffico cumulato in entrata è risultato essere più alto di quello in uscita. Allora, si è scelti due sottosistemi chiusi per i quali i dati disponibili durante un lungo intervallo di tempo risultassero più accurati di altri. Noi abbiamo scelto questi per la procedura di calibrazione cercando di correggere i dati corrispondenti per preservarne la consistenza.

Tuttavia un non trascurabile errore deve essere tenuto in considerazione, esaminando i risultati di calibrazione.

Il primo sottosistema è un tronco autostradale di 9 Km tra le barriere di pedaggio di Portici e Torre Annunziata in direzione Salerno. In essa sono

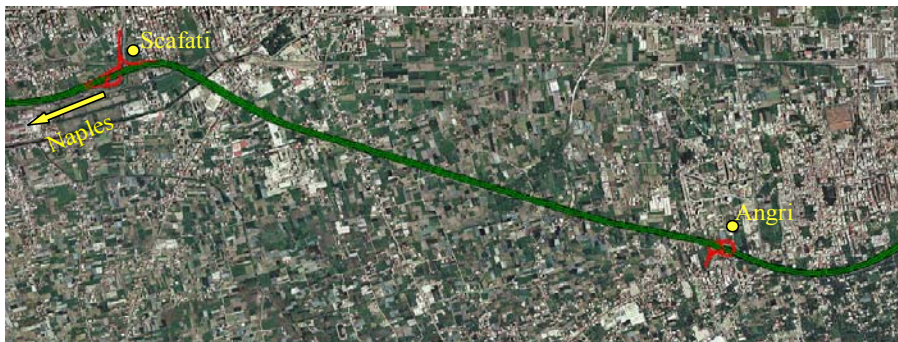


state esaminate le tre ore pomeridiane coincidenti alla condizione di punta (il momento di maggior traffico).



**Figura 6—6 Portici**

Invece il secondo è un tronco autostradale di 4 Km tra le barriere dei pedaggi di Angri e Scafati in direzione di Napoli. Qui al contrario, sono state esaminate le tre ore di punta della mattina.



**Figura 6—7 Angri**

Questa è la stessa rete utilizzata sia per la fase di analisi di sensitività del modello che per la verifica della procedura di calibrazione.

Il vantaggio di effettuare la sperimentazione in un contesto autostradale è quello di non dover tenere in considerazione tutte le variabili introdotte dai modelli di scelta del percorso e di stima della domanda di spostamento. Dal punto di vista metodologico ciò è essenziale per condurre un'analisi statistica adeguata del fenomeno.

#### **6.4.2. Analisi del modello di microsimulazione utilizzato**

Così come descritto i modelli dinamici, evidenziamo le caratteristiche generali della microsimulazione:

- Rappresentazione del flusso discreta (singolo veicolo)
- Dinamica del traffico dettagliata; rappresentazione esplicita di:
- Comportamenti di guida
- Comportamenti di viaggio
- Strategie di controllo e di routing
- Sensori di monitoraggio
- Differenziazione per classi di veicoli/guidatori
- Rappresentazione dettagliata degli elementi d'offerta (reti integrate di autostrade e strade urbane)
- Struttura di simulazione "ibrida"(struttura in cui coesistono una lista di eventi aggiornata in tempo reale, con uno "scanning" delle attività ad intervalli  $\Delta t$  fissi)
- Natura della simulazione stocastica.

Ne consegue come il simulatore microscopico del traffico ci permetta di interagire su differenti reti di traffico: dalle urbane e dalle autostrade a

combinazioni di esse. E' uno strumento per analisi del traffico, in grado di progettare e verificare i suddetti sistemi. Si può però con certezza evidenziare la grande utilità che ne deriva dall'applicazione ai nuovi sistemi di controllo del traffico e alle politiche di gestione, siano esse basate su tecnologie tradizionali o sull'implementazione dell' ITS.

Compiè ampiamente anticipato, il software di simulazione microscopica del traffico utilizzato nello studio volto alla validazione dell'integrazione con il simulatore di guida è AIMSUN NG (Advanced Interactive Microscopic Simulator for Urban and Non-urban Network). Per una specificazione dettagliata del modello si faccia riferimento a [4]. Esso è uno dei più diffusi simulatori microscopici del traffico, capace di riprodurre differenti meccanismi dinamici del settore dei trasporti.

È importante ricordare che AIMSUN è un simulatore stocastico: ciascuna replicazione di una simulazione/esperimento dipende da un particolare seme, che influenza l'intera generazione di numeri pseudo-casuali utilizzata dal suo sistema. Quindi per considerare ciò, quando si vanno a comparare gli output del modello con i dati reali è necessario effettuare, per ogni singola simulazione, un numero di repliche tanto maggiore quanto maggiore è l'affidabilità del dato che si vuole ottenere con l'intenzione di smorzare l'influenza del particolare seme adottato.

Utilizzando un intervallo di confidenza sulla media della misura simulata, può determinarsi il numero minimo di repliche. Infatti, fissato il livello di confidenza, occorre determinare il numero minimo dei gradi di libertà di una funzione di distribuzione di probabilità T di Student, che viene praticamente a coincidere con il numero di repliche eseguite. Questa

funzione di distribuzione deve avere un certo livello di confidenza,  $(1-\alpha)$ , in maniera tale da creare un certo intervallo fiduciario.

Si dice che:

*se la differenza tra la media campionaria e quella reale è minore di una frazione della deviazione standard campionaria*

$$|\bar{X}(n) - \mu| \leq t_{n-1, 1-\frac{\alpha}{2}} \sqrt{\frac{S^2(n)}{n}}$$

*allora la proporzione  $(1-\alpha)$  degli infiniti intervalli di confidenza al  $100(1-\alpha)$ , costruiti ciascuno con  $n$  osservazioni, conterrà la  $\mu$  media reale.*

Durante i vari studi effettuati con il modello si è visto che al fine di smorzare gli effetti della stocasticità è necessario effettuare almeno tre replicazioni con tre semi differenti per ogni simulazione.

Per riprodurre correttamente le dinamiche connesse con la propagazione del flusso dei veicoli, in una simulazione microscopica, le manovre dei veicoli sono modellate in dettaglio usando i modelli di car following, lane changing e gap acceptance. Questi modelli comportamentali sono funzione di diversi parametri che permettono di riprodurre differenti tipi di veicoli: auto, bus, camion ecc; i parametri possono essere divisi in tre categorie legate al livello in cui sono definite:

- *Caratteristiche dei Veicoli*
- *Parametri Locali*
- *Parametri Globali della Modellazione*
- *Generali*
- *Car following per due corsie*
- *Lane Changing*
- *Altri*

Nel particolare le *Caratteristiche dei Veicoli* riguardano parametri definiti in relazione ai diversi tipi di veicoli, non solo nei valori medi ma anche nella deviazione, massimi e minimi. I *Parametri Locali* possono influenzare il comportamento dei veicoli, nonostante non siano definiti al livello dei tipi di veicoli, ma perché sono correlati alle sezioni. I *Parametri Globali della Modellazione* rappresentano un set di parametri correlati ai modelli comportamentali, per cui validi sull'intera rete e durante l'intera simulazione dell'esperimento; infatti nel loro percorrere la rete il comportamento dei veicoli è continuamente aggiornato secondo i modelli di Car following e di Lane Changing.

I guidatori tendono a viaggiare alla loro velocità desiderata in ciascuna sezione, ma l'ambiente circostante condiziona fortemente il loro modo di agire. Il tempo di simulazione è diviso in intervalli minori chiamati simulation cycle o simulation step( $\Delta t$ ). Il simulation step può influire non solo sulle performance di computazione, ma anche su alcuni output di simulazione; ad esempio quanto minore è  $\Delta t$ , tanto maggiore sarà la capacità della sezione.

Il modello di *Car Following* implementato in AIMSUN è basato sul modello di Gipps. Esso può essere attualmente considerato come un sviluppo *ad hoc* di questo modello empirico, nel quale i parametri interessati non sono globali, ma influenzati dai parametri locali dipendenti dal “tipo di guida” (limit speed acceptance del veicolo), dalla geometria della sezione (limit speed della sezione, ecc.), dall'influenza dei veicoli nelle corsie adiacenti, ed altri.

Esso consta essenzialmente di due componenti, accelerazione e decelerazione. La prima rappresenta l'intenzione del veicolo di raggiungere una certa velocità desiderata (principio che sorregge l'intero modello), mentre il secondo riproduce le limitazioni imposte dal veicolo che precede, quando si cerca la velocità desiderata.

Questo modello specifica che la velocità massima alla quale un veicolo- $n$  può accelerare in un tempo  $(t, t+\tau)$  è dato da:

$$v_n^A(t+\tau) \leq v_n(t) + 2,5 \cdot a_n \cdot \tau \cdot \left(1 - \frac{v_n(t)}{V_n}\right) \cdot \sqrt{0,025 + \frac{v_n(t)}{V_n}}$$

dove:  $v_n(t)$  è la velocità del veicolo- $n$  al tempo  $t$

$V_n$  è la velocità desiderata dal veicolo- $n$  per la corrente sezione

$a_n$  è la max accelerazione per il veicolo- $n$

$\tau$  è il tempo di reazione.

Dall'altro lato, la massima velocità che lo stesso veicolo- $n$  può raggiungere durante il medesimo intervallo di tempo  $(t, t+\tau)$ , concordemente con le sue proprie caratteristiche e le limitazioni imposte dalla presenza del veicolo leader è:

$$v_n^B(t+\tau) = b_n \tau + \sqrt{b_n^2 \tau^2 - b_n \cdot \left[2(x_{n-1}(t) - s_{n-1} - x_n(t)) - v_n(t) \cdot \tau - \frac{v_{n-1}^2(t)}{\hat{b}_{n-1}}\right]}$$

dove:  $b_n$  ( $<0$ ) è la max decelerazione desiderata dal veicolo- $n$

$x_n(t)$  è la posizione del veicolo- $n$  al tempo  $t$

$x_{n-1}(t)$  è la posizione del veicolo- $(n-1)$  al tempo  $t$

$s_{n-1}$  è l'effettiva lunghezza del veicolo- $(n-1)$

$\hat{b}_{n-1}$  è la stima della decelerazione desiderata del veicolo- $(n-1)$

In ogni caso, la velocità definitiva del veicolo- $n$  durante l'intervallo  $(t, t+\tau)$  è il valore minimo delle formulazioni precedenti:

$$v_n(t + \tau) = \min \{v_n^A(t + \tau), v_n^B(t + \tau)\}$$

Ne consegue che la posizione del veicolo-n all'interno della corsia sia aggiornato con questa velocità e per vale:

$$x_n(t + \tau) + v_n(t + \tau)\tau$$

Il modello di **Lane Changing** può essere anche considerato come uno sviluppo di quello di Gipps.

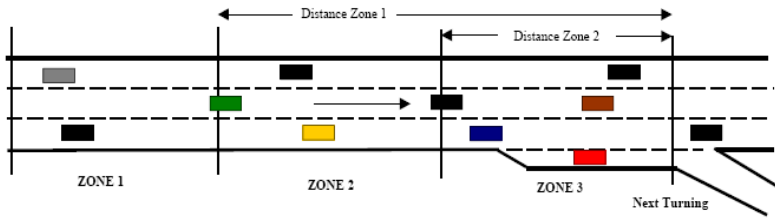
E' strutturato come un processo decisionale, di analisi della possibilità del cambio di corsia, della desiderabilità di cambiare corsia, e della fattibilità delle condizioni per il cambio, condizioni queste che sono anche locali, dipendendo dalla posizione del veicolo sulla rete stradale.

Il comportamento del modello può sintetizzarsi con alcune domande, che esprimono al meglio come possa essere definita la manovra di cambio di corsia eseguita da un guidatore:

- È necessario cambiare corsia?
- È desiderabile cambiare corsia?
- È possibile cambiare corsia?

Queste domande prendono in considerazione diversi fattori come la fattibilità della svolta, la distanza dalla svolta successiva, le condizioni di traffico in termini di velocità e di coda, la possibilità di avere dei miglioramenti nelle condizioni di deflusso e non ultimo la sicurezza della svolta calcolata sullo spazio di frenata.

In quest'ottica il modello si avvale di una rappresentazione accurata della sezione distinta in tre zone



**Figura 6—8 Modello di lane changing. Individuazione zone**

A queste zone sono collegati i parametri *Percent Overtake* e *Percent Recover* che rappresentano la percentuale della velocità desiderata che, rispettivamente, il veicolo che segue, ed il veicolo che precede, devono decidere per superare, oppure rientrare nella corsia più lenta.

Al suddetto modello di Lane Changing si unisce il modello di *Gap Acceptance* che studia appunto lo spazio disponibile a che possa realizzarsi il cambio di corsia; si è soliti far riferimento ad un *gap critico*, funzione delle velocità relativa e dei distanziamenti da monte e da valle.

## **6.5 Analisi di sensitività**

In relazione a quanto detto in precedenza, sulle problematiche della qualità e dell'ottimizzazione strutturale nel contesto della calibrazione in generale, l'analisi di sensitività si inquadra come un momento fondamentale.

La numerosità  $n$  dei parametri da cui i modelli in precedenza descritti rende qualsiasi tentativo di ricerca della soluzione ottima vano sia per la complessità di esplorazione di uno spazio delle soluzioni  $n$ -dimensionale sia per l'effetto negativo che la presenza di variabili aventi peso notevolmente differente sugli output del modello hanno sugli algoritmi che vengono utilizzati in questi casi. Per questo motivo restringere e vincolare lo spazio



delle possibili soluzioni è condizione necessaria per la risolvibilità del problema. Chiaramente tale restringimento deve essere effettuato con la logica di effettuare un trade-off con la capacità del modello di adattarsi al mondo reale: la riduzione esasperata dei parametri da calibrare porta infatti ad una migliore riproducibilità dei dati che si utilizzano in fase di calibrazione, ma ad una minore capacità del modello ad adeguarsi al cambiamento dei suoi input nella fase di validazione.

Un'analisi di sensitività del modello consente dunque di analizzare il peso di ogni singolo parametro del modello in maniera da restringere il numero dei parametri da analizzare avendo una percezione dell'errore che si dovrà scontare sul comportamento del modello.

### **6.5.1. Metodologia**

Il primo passo per la scelta dei parametri da coinvolgere nella procedura del modello di calibrazione è, quindi, un'analisi di sensitività del modello. L'obiettivo è di trovare come il valore dei parametri del modello vadano ad influenzare gli output dello stesso. Quando il modello è rappresentato da una funzione continua, l'analisi è ottenuta studiandone le derivate parziali. Nel caso di un modello di simulazione ciò non può farsi a causa dell'indisponibilità di una formulazione esplicita dello stesso. Un modo per aggirare il problema è eseguire la simulazione ripetutamente, cambiando di volta in volta i parametri del modello e costruendo, dall'interpolazione dei dati ottenuti, un metamodello per il quale possano essere calcolate le derivate parziali.

Un altro metodo, che è quello utilizzato nel presente studio, consiste nel condurre un'analisi statistica degli output del modello al variare del valore assunto dai parametri.

L'obiettivo di fornire la sensitività di un certo numero di parametri, con il minor numero possibile di simulazioni, è raggiunto nell'utilizzo della metodologia della *Progettazione degli Esperimenti* (DOE).

### **6.5.2. Progettazione degli esperimenti**

La progettazione degli esperimenti può vedersi come tecnica finalizzata all'individuazione della combinazione dei parametri di progetto, detti fattori di controllo. Di norma il numero di tali fattori da esaminare si aggira intorno alla decina e per ognuno di questi è necessario considerare almeno 2 o 3 livelli distinti. Questi ultimi devono essere sempre fissati in un range piuttosto ampio, ossia tale da consentire ad eventuali fattori di disturbo di manifestare la loro influenza.

Lo studio sperimentale che è necessario effettuare pone dei grossi problemi di durata, a meno di non ricorrere a specifici piani sperimentali da approntare. Infatti, se, ad esempio, supponiamo che sia 10 il numero di fattori cui compete ciascuno 2 livelli, per un piano sperimentale di tipo fattoriale completo (si veda il par. successivo), occorrerebbero  $2^{10}$  osservazioni sperimentali, cosa che costituisce una difficoltà insormontabile, se non si adottano opportuni accorgimenti.

Risolto il problema della pianificazione degli esperimenti ed ottenuti i risultati, si deve procedere all'analisi di questi ultimi ai fini dell'ottimizzazione della risposta del sistema. Questa valutazione può essere fatta statisticamente usando l'Analisi della Varianza (ANOVA). Tale tecnica è usata per ottenere l'informazione sugli effetti del particolare fattore (i parametri) su una particolare variabile (gli output del modello). L'obbiettivo della tecnica è isolare la varianza sperimentale spiegata per i

fattori e di assicurare, con un certo livello di confidenza, che la variabile sia sensibile nei confronti del fattore.

La progettazione sperimentale fornisce una maniera per decidere, prima che i tentativi siano fatti, quali configurazioni particolari siano tali da ottenere le informazioni desiderate col minimo sforzo. Inoltre seguire una delle tecniche della progettazione degli esperimenti consente:

- di controllare fattori a prima vista incontrollabili;
- di incrementare ulteriormente il controllo sulla fonte della variabilità, conoscendo la natura deterministica dei random-number generators;
- di proteggere dagli errori sistematici sperimentali, casualizzando le combinazioni dei fattori nonché l'ordine delle stesse sequenze.

Maggiormente in dettaglio, nella terminologia della progettazione i parametri di input e l'ipotesi strutturale del modello vengono indicati come *fattori*, mentre le misure di performance di output sono le *risposte*.

I fattori in particolare possono essere *quantitativi* o *qualitativi*, dove naturalmente sono quantitativi, se assumono valori numerici, qualitativi, se esprimono un'ipotesi strutturale non quantizzabile. Sono ancora *controllabili* o *incontrollabili* se rappresentano un'azione che è possibile manipolare.

Di norma, il numero di parametri che influenzano un modello di simulazione è molto grande. Quindi, per la fattibilità di un buon studio di calibrazione, dovrà essere selezionato un appropriato sottoinsieme di parametri. In più, dovrà definirsi una funzione obiettivo caratterizzante il modello o proporre più di una in maniera tale da non perdere di alcuni

parametri. Non da sottovalutare è la considerazione della stocasticità per avere una stima significativa dei parametri; da qui, come si è detto anche in precedenza sui vantaggi apportati, l'utilizzo della tecnica della progettazione degli esperimenti.

Possiamo dire che la suddetta tecnica prevede differenti metodi per la sua costruzione, i quali vanno differenziandosi in base al *modello*; in tal caso per modello s'intende il tipo di funzione con cui si vuole approssimare la risposta (*response*).

Il modello può essere lineare

$$y = c + \sum_{i=1}^k a_i x_i + e \quad i=1 \dots k$$

con i coefficienti  $a_i$  che devono minimizzare l'errore  $e = y - y_{app}$ , dove  $y_{app}$  è la risposta stimata dal modello. Di conseguenza quello che è un modello di *regressione*, e che trova una buona regolazione nella procedura dei *minimi quadrati*, può scriversi in forma matriciale:

$$Y = X \cdot A + E$$

da cui il vettore stimato dei coefficienti

$$A_{app} = (X^t \cdot X)^{-1} \cdot X^t Y$$

Da cui per il principio chiave della progettazione, si cerca di combinare i livelli dei fattori per minimizzare i termini dell'inversa del prodotto matriciale  $(X^t \cdot X)^{-1}$ . Ciò è soddisfatto solo se la matrice è *ortogonale*.

I progetti sperimentali sono detti *ortogonali* (due vettori di identiche dimensioni sono ortogonali se la somma dei prodotti degli elementi corrispondenti è pari a zero) rispetto al modello, se un qualunque fattore si bilancia attraverso gli effetti di altri fattori; in altre parole, se la stima dell'effetto medio di ciascun fattore sia eseguita indipendentemente dai livelli degli altri.

Ritornando sull'importanza della scelta dei livelli, non è consigliabile scegliere i livelli “-“ e “+” di un fattore quantitativo estremamente lontani dagli altri. La ragione è che la larga separazione tra i livelli corre il rischio di dare l'illusione di linearità, quando nei fatti c'è un intervallo interno non-lineare; cosa che ovviamente non potremmo decifrare dai risultati. La larga separazione tra i livelli dei fattori rende discutibile l'interpolazione in quella precisa zona interna. Al di là di aumentare i livelli, quindi, un possibile rimedio è l'uso di un *metamodello*, ossia di un piano più elaborato.

Il proposito di un metamodello è quello di stimare o approssimare la superficie di risposta; lo si utilizza per conoscere come la superficie si comporta su diverse regioni dello spazio definito dai fattori di input, come la risposta (response) può cambiare quando gli input variano singolarmente, od anche per ricercare approssimativamente la combinazione ottima dei fattori di input.

Spesso, un *metamodello* è un modello di regressione standard, dove le variabili indipendenti per la regressione sono i parametri di input della simulazione e la variabile dipendente è la risposta cercata. E' possibile sviluppare differenti modelli di regressione a seconda del response.

E' necessario eseguire almeno *qualche* simulazione di differenti combinazioni di input, per ottenere i dati da cui estrapolare la regressione. Ovviamente nell'utilizzare tali modelli adattati, è possibile disporre di tutti i consueti strumenti, come gli intervalli di confidenza e la stima degli intervalli.

A seconda del grado di accuratezza si può scegliere una regressione lineare o quadratica

$$E[R(x_j, x_i)] = b_0 + b_j x_j + b_i x_i + b_{ji} x_j x_i$$

$$E[R(x_j, x_i)] = b_0 + b_j x_j + b_i x_i + b_{ji} x_j x_i + b_{ii} x_i^2 + b_{jj} x_j^2$$

con  $j$  ed  $i$  variabili di input, i parametri  $b_0 \dots b_{ji}$  stimati ai minimi quadrati.

### 6.5.3. Implementazione e risultati

Come già detto, il primo step dell'analisi di sensitività è definire i parametri che devono essere trattati, e, ancora ripetendo, gli unici parametri che andremo a considerare sono quelli legati alla propagazione dinamica del flusso. In AIMSUN tali parametri sono divisi in varie categorie, in accordo con quanto detto in precedenza per quanto riguarda i modelli utilizzati: parametri globali, parametri locali e attributi della categoria veicolare. I parametri locali sono diversi per sezione stradale. Il loro livello di dettaglio è troppo elevato e la loro definizione non può entrare nella fase di calibrazione, ma piuttosto ad essi deve essere dedicata una fase di studio della rete accompagnata da una serie di rilievi di campo che rendano l'utilizzatore in grado di cogliere i fenomeni esistenti.

I parametri *globali*, sono tali in quanto hanno un unico valore su tutti gli elementi della simulazione. Essi sono essenzialmente i seguenti:

- *simulation step (reaction time) "rT"*
- *percent recover "pR"*
- *percent overtake "pO"*
- *reaction time at stop "rTs"*.

Il reaction time è strettamente legato allo step di simulazione per questioni modellistiche, ma la sua definizione (così come quella del Reaction Time at Stop) può anche essere impostata a livello della singola categoria veicolare.

Per quel che riguarda le caratteristiche dei veicoli, quindi, oltre ai due tempi di reazione appena indicati, i parametri che è possibile definire a livello di singola categoria sono:

- *max desired speed “mDs”*
- *max acceleration “mA”*
- *normal deceleration “nD”*
- *max deceleration “mD”*
- *give way time “gWt”*.

In totale sono stati quindi presi in considerazione nove parametri nella nostra analisi. In realtà, nelle simulazioni assumiamo che il parco veicolare sia assimilabile a due categorie veicolari. Di queste, però solo i parametri di una delle due saranno modificati durante l'analisi di sensitività a causa dell'eccessivo numero di simulazioni che si avrebbero altrimenti.

Diverse analisi sono state compiute per la scelta della MoP da considerare in fase di calibrazione del modello e per la scelta della lunghezza dell'intervallo di misurazione (MIL) delle stesse MoP più indicato. l'obbiettivo è stato quello di cercare la coppia (MoP,MIL) che meglio esprimono l'influenza dei valori dei parametri sull'output del modello. Le MoP considerate sono state le uniche generalmente disponibili nel monitoraggio del traffico: flusso e velocità, per classe di veicoli. Un minuto e cinque minuti sono, invece gli intervalli su cui mediare le misure. Ciò significa portare avanti quattro analisi.

Per non trascurare gli effetti di alcun parametro, si sono considerati solo piani fattoriali completi.

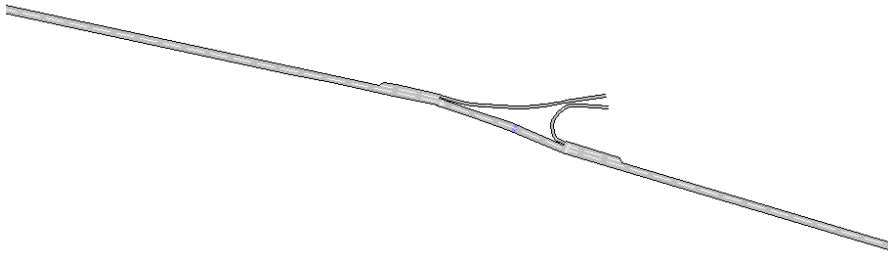
La prima analisi di sensitività è stata condotta sui risultati di un piano fattoriale completo a due livelli. Tale strada viene generalmente utilizzata allorquando il fenomeno da analizzare è approssimativamente lineare. Però la prova è stata fatta anche per capire la robustezza di questi strumenti.

In questo caso  $2^9 = 512$  sono stati gli esperimenti necessari. I risultati ottenuti, però, usando ambo gli output-MoP, sono stati insoddisfacenti: solo pochi parametri risultavano essere significativi e, soprattutto, più del 90% della varianza era spiegata da un unico parametro, il simulation step (in questo caso legato al reaction time). Come precedentemente esposto, il reaction time del guidatore, è sicuramente in grado di condizionare l'intera fase sperimentale ma non con una così netta predominanza rispetto agli altri parametri.

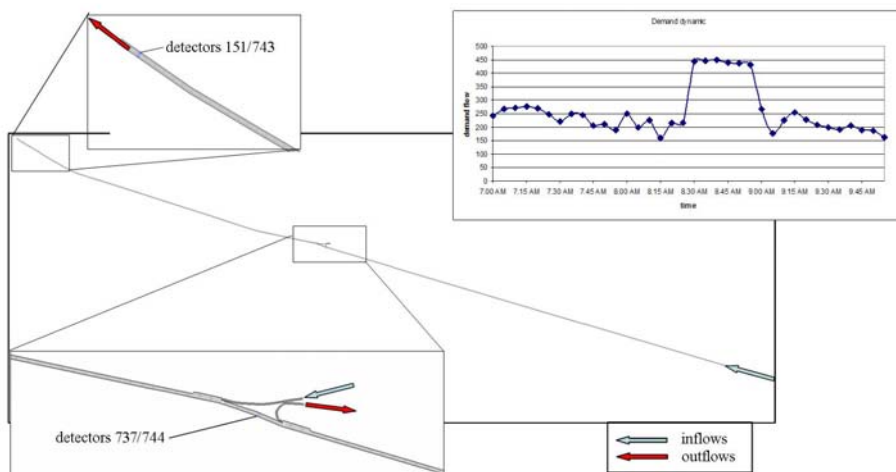
Allora l'analisi è stata ripetuta incrementando il numero dei livelli per parametro, in particolare un piano fattoriale completo a tre livelli, che ha portato migliori risultati, soprattutto per la MoP della velocità. Sono state questa volta eseguite  $3^9 = 19683$  simulazioni, ottenendo risultati completamente differenti.

Per implementare l'analisi di sensitività si è caricato il tronco autostradale rappresentato in Figura 6—9, pienamente rappresentativo delle condizioni dinamiche di traffico; è una tratta di 15 Km circa con una rampa d'accesso e una di uscita al centro. Presenta quattro detector, due dei quali sulle rampe e due alla fine del tronco, che registrano il flusso e la velocità del traffico ogni 5' di simulazione dividendoli per categoria di veicoli. La configurazione dei sensori e la dinamica della domanda che carica la rete sono riportate in Figura 6—10.





**Figura 6—9 Simulazione dello svincolo di Angri**



**Figura 6—10 Posizionamento dei sensori e andamento della domanda sulla rete**

Dopo aver costruito il piano degli esperimenti ed aver eseguito le simulazioni, i valori di output vengono registrati in un database, dal quale sarà poi possibile gestirli per un ANOVA. Tale tecnica prevede che si effettui un test d'ipotesi sui valori di output, dai quali si può calcolare una variabile casuale avente distribuzione Z di Fisher.

L'ipotesi è rigettata se la probabilità di avere un valore più grande di quello calcolato è inferiore ad una fissata probabilità (il rischio che ci assumiamo per rigettare un'ipotesi vera).

$$p = \Pr\{Z_{n,m} > z^*\} > \alpha$$

con  $z^*$  pari al rapporto tra lo scarto quadratico medio del fattore in questione e lo scarto quadratico medio legato all'errore

$Z_{n,m}$  valore tabellato in funzione del livello di significatività  $\alpha$  e dei gradi di libertà  $n$  del fattore  $m$  dell'errore.

Il problema sorge con l'aumentare del numero di parametri da testare, infatti l'implementazione dell'ANOVA diviene difficile da gestire. Nel nostro caso cercando la sensitività di 9 parametri e  $(9 \times 8)/2 = 36$  interazioni si è scelto di utilizzare dapprima il software BOSS/QUATTRO [15] per eseguire il test.

L'analisi richiede che BOSS/QUATTRO comunichi con AIMSUN, cosa resa possibile usando file batch (.bat) e python script (.py). La struttura costruita si è rivelata stabile ed affidabile; l'unico problema è stato l'elevato impegno di risorse che ha reso impossibile eseguire le 19683 simulazioni necessarie nonostante l'utilizzo di potenti strumenti. Si è quindi scelto di eseguire l'analisi sullo stesso numero di parametri ma utilizzando un numero di livelli differenti per ogni parametro: 6 con tre livelli ciascuno e gli altri tre con due livelli, per un totale di 5832 simulazioni, circa 50 ore di elaborazione.

I risultati sono nelle tabelle seguenti.

SOURCES	ANOVA 2 levels (MoP = speed)			
	F-LEVEL	F-LEVEL <sub>0.95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	50	254	NO SENSITIVE	0.64%
<i>mD</i>	0	254	NO SENSITIVE	0.00%
<i>mDs</i>	65	254	NO SENSITIVE	0.84%
<i>nD</i>	21	254	NO SENSITIVE	0.28%
<i>pO</i>	0	254	NO SENSITIVE	0.00%
<i>rT</i>	7026	254	<b>HIGH SENSITIVE</b>	<b>90.99%</b>
<i>rTs</i>	0	254	NO SENSITIVE	0.00%
<i>mD</i> <i>pO</i>	0	254	NO SENSITIVE	0.00%
<i>mD</i> <i>rTs</i>	0	254	NO SENSITIVE	0.00%
<i>mD</i> <i>rT</i>	7	254	NO SENSITIVE	0.09%
<i>mD</i> <i>mDs</i>	0	254	NO SENSITIVE	0.00%
<i>mD</i> <i>mA</i>	0	254	NO SENSITIVE	0.00%
<i>mDs</i> <i>mA</i>	0	254	NO SENSITIVE	0.00%
<i>nD</i> <i>mD</i>	0	254	NO SENSITIVE	0.00%
<i>nD</i> <i>pO</i>	0	254	NO SENSITIVE	0.00%
<i>nD</i> <i>rTs</i>	0	254	NO SENSITIVE	0.00%
<i>nD</i> <i>rT</i>	5	254	NO SENSITIVE	0.07%
<i>nD</i> <i>mDs</i>	0	254	NO SENSITIVE	0.00%
<i>nD</i> <i>mA</i>	0	254	NO SENSITIVE	0.00%
<i>pO</i> <i>rTs</i>	0	254	NO SENSITIVE	0.00%
<i>pO</i> <i>rT</i>	12	254	NO SENSITIVE	0.16%
<i>pO</i> <i>mDs</i>	1	254	NO SENSITIVE	0.01%
<i>pO</i> <i>mA</i>	0	254	NO SENSITIVE	0.00%
<i>rT</i> <i>mDs</i>	11	254	NO SENSITIVE	0.15%
<i>rT</i> <i>mA</i>	32	254	NO SENSITIVE	0.41%
<i>rTs</i> <i>rT</i>	9	254	NO SENSITIVE	0.11%
<i>rTs</i> <i>mDs</i>	0	254	NO SENSITIVE	0.00%
<i>rTs</i> <i>mA</i>	0	254	NO SENSITIVE	0.00%
			<i>unexplained</i>	<b>6.04%</b>

Tabella 6—1 Analisi della varianza a 2 livelli – risultati

SOURCES	ANOVA 3 levels (MoP = traffic counts)			
	F-LEVEL	F-LEVEL <sub>0,95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	28	19	<b>HIGH SENSITIVE</b>	0.07%
<i>mD</i>	0	19	NO SENSITIVE	0.00%
<i>mDs</i>	22463	19	<b>HIGH SENSITIVE</b>	<b>56.59%</b>
<i>nD</i>	4	19	NO SENSITIVE	0.01%
<i>pO</i>	271	19	<b>HIGH SENSITIVE</b>	0.68%
<i>rT</i>	5663	19	<b>HIGH SENSITIVE</b>	<b>14.27%</b>
<i>rTs</i>	38	19	<b>HIGH SENSITIVE</b>	0.10%
<i>mD</i> <i>pO</i>	0	6	NO SENSITIVE	0.00%
<i>mD</i> <i>rTs</i>	0	6	NO SENSITIVE	0.00%
<i>mD</i> <i>rT</i>	0	6	NO SENSITIVE	0.00%
<i>mD</i> <i>mDs</i>	0	6	NO SENSITIVE	0.00%
<i>mD</i> <i>mA</i>	0	6	NO SENSITIVE	0.00%
<i>mDs</i> <i>mA</i>	10	6	<b>HIGH SENSITIVE</b>	0.05%
<i>nD</i> <i>mD</i>	0	6	NO SENSITIVE	0.00%
<i>nD</i> <i>pO</i>	3	6	NO SENSITIVE	0.01%
<i>nD</i> <i>rTs</i>	0	6	NO SENSITIVE	0.00%
<i>nD</i> <i>rT</i>	16	6	<b>HIGH SENSITIVE</b>	0.08%
<i>nD</i> <i>mDs</i>	21	6	<b>HIGH SENSITIVE</b>	0.11%
<i>nD</i> <i>mA</i>	1	6	NO SENSITIVE	0.01%
<i>pO</i> <i>rTs</i>	0	6	NO SENSITIVE	0.00%
<i>pO</i> <i>rT</i>	135	6	<b>HIGH SENSITIVE</b>	0.68%
<i>pO</i> <i>mDs</i>	221	6	<b>HIGH SENSITIVE</b>	<b>1.11%</b>
<i>pO</i> <i>mA</i>	1	6	NO SENSITIVE	0.00%
<i>rT</i> <i>mDs</i>	4581	6	<b>HIGH SENSITIVE</b>	<b>23.08%</b>
<i>rT</i> <i>mA</i>	18	6	<b>HIGH SENSITIVE</b>	0.09%
<i>rTs</i> <i>rT</i>	74	6	<b>HIGH SENSITIVE</b>	0.37%
<i>rTs</i> <i>mDs</i>	9	6	<b>HIGH SENSITIVE</b>	0.04%
<i>rTs</i> <i>mA</i>	1	6	NO SENSITIVE	0.00%
			<i>unexplained</i>	<b>2.63%</b>

**Tabella 6—2** Analisi della varianza a tre livelli – Risultati dei conteggi come MoP

SOURCES	ANOVA 3 levels (MoP = speed)			
	F-LEVEL	F-LEVEL <sub>0.95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	1954	19	<b>HIGH SENSITIVE</b>	<b>9.94%</b>
<i>mD</i>	4	254	NO SENSITIVE	0.01%
<i>mDs</i>	4873	19	<b>HIGH SENSITIVE</b>	<b>24.80%</b>
<i>nD</i>	296	19	<b>HIGH SENSITIVE</b>	<b>1.51%</b>
<i>pO</i>	16	19	NO SENSITIVE	0.08%
<i>rT</i>	5561	19	<b>HIGH SENSITIVE</b>	<b>28.30%</b>
<i>rTs</i>	109	19	<b>HIGH SENSITIVE</b>	0.56%
<i>mD</i> <i>pO</i>	2	254	NO SENSITIVE	0.00%
<i>mD</i> <i>rTs</i>	47	6	<b>HIGH SENSITIVE</b>	0.48%
<i>mD</i> <i>rT</i>	0	19	NO SENSITIVE	0.00%
<i>mD</i> <i>mDs</i>	1	19	NO SENSITIVE	0.00%
<i>mD</i> <i>mA</i>	3	19	NO SENSITIVE	0.01%
<i>mDs</i> <i>mA</i>	34	19	<b>HIGH SENSITIVE</b>	0.17%
<i>nD</i> <i>mD</i>	0	6	NO SENSITIVE	0.00%
<i>nD</i> <i>pO</i>	27	6	<b>HIGH SENSITIVE</b>	0.28%
<i>nD</i> <i>rTs</i>	7	6	<b>HIGH SENSITIVE</b>	0.07%
<i>nD</i> <i>rT</i>	0	19	NO SENSITIVE	0.00%
<i>nD</i> <i>mDs</i>	0	19	NO SENSITIVE	0.00%
<i>nD</i> <i>mA</i>	74	19	<b>HIGH SENSITIVE</b>	0.37%
<i>pO</i> <i>rTs</i>	11	19	NO SENSITIVE	0.05%
<i>pO</i> <i>rT</i>	0	254	NO SENSITIVE	0.00%
<i>pO</i> <i>mDs</i>	17	6	<b>HIGH SENSITIVE</b>	0.17%
<i>pO</i> <i>mA</i>	37	19	<b>HIGH SENSITIVE</b>	0.19%
<i>rT</i> <i>mDs</i>	6	19	NO SENSITIVE	0.03%
<i>rT</i> <i>mA</i>	3	6	NO SENSITIVE	0.03%
<i>rTs</i> <i>rT</i>	2	6	NO SENSITIVE	0.02%
<i>rTs</i> <i>mDs</i>	1	19	NO SENSITIVE	0.01%
<i>rTs</i> <i>mA</i>	0	19	NO SENSITIVE	0.00%
			<i>unexplained</i>	<b>14.55%</b>

**Tabella 6—3** Analisi della varianza a tre livelli – Risultati delle velocità come MoP

Le tabelle suddette presentano un ANOVA per delle MoP raccolte ogni cinque minuti; questo perché per ogni singolo minuto, i valori delle MoP risultano troppo sensibili a ciascun parametro. Confrontando i risultati dell’ANOVA per il piano sperimentali a tre livelli, si vede come ci siano diversi parametri sensitivi; tuttavia usando il flusso come MoP risulta che la maggioranza della varianza spiegata è legata a due soli parametri *maxDesiredspeed* (“mDs”) e *Simulation step* (“rT”).

Al fine di avere un ulteriore riscontro della veridicità di quanto ottenuto sul test dell’ANOVA da parte del software BOSS/QUATTRO, si è

implementata la stessa analisi con l'ausilio del software MATLAB 6.0 [39], riscontrando una perfetta validazione dei risultati di BOSS/QUATTRO.

Una volta realizzata l'analisi di sensitività si è approcciato lo studio per ricavare una metodologia per la calibrazione di AIMSUN. Nel paragrafo seguente si esaminerà tale metodologia e si mostrerà come le varie fasi, tra cui l'analisi appena descritta, interagiscono tra loro per condurre alla stima ottimale dei parametri.

## **6.6 Calibrazione del modello**

La validazione e la calibrazione dei modelli sono rappresentano delle fasi collegate tra loro che teoricamente dovrebbero essere eseguite prima di ogni nuova applicazione del modello in esame. L'approccio generale proposto per la calibrazione e la validazione è illustrato in Figura 6—11.

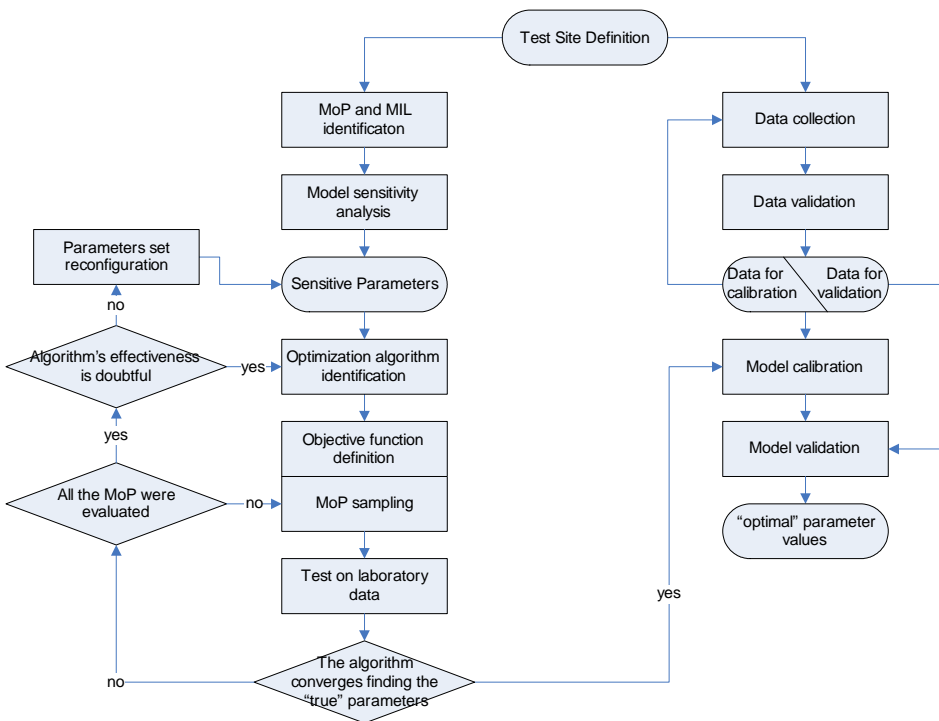
Una volta individuato e rappresentato attraverso gli strumenti del modello di simulazione microscopica del traffico il modello d'offerta e il modello di domanda del sistema da analizzare, la metodologia prevede due fasi. La prima, individuata dagli eventi posti in successioni nella parte sinistra del diagramma, che vuole studiare il modello in relazione allo schema da analizzare senza utilizzare i dati reali su cui eseguire la calibrazione. Durante questa fase, a valle dell'analisi di sensitività del modello descritta in precedenza, utilizzando un test con dati sintetici (cioè ottenuti attraverso una simulazione) si fissano le variabili della calibrazione. In particolare attraverso un processo iterativo che vede al centro il suddetto test, si sceglie l'algoritmo che maggiormente è in grado di affrontare il problema di ottimizzazione, la forma funzionale della funzione obiettivo da inserire nel problema di ottimizzazione, la MoP e il MIL da considerare sui dati a

disposizione. In pratica, variando le variabili progettuali della procedura di ottimizzazione, il test consiste nell'utilizzare i dati ricavati da una simulazione (quindi a parametri noti) in luogo dei dati reali, per verificare che l'insieme delle variabili scelte sia in grado di riprodurre i valori dei parametri impostati. Il vantaggio di questa procedura sta nel fatto che in questo modo si avrà coscienza della qualità dei risultati ottenuti non solo in termini di vicinanza con gli output del modello, ma anche in relazione ai parametri iniziali, a differenza di quanto si osserva in letteratura. Inoltre, in questo modo, si ha effettivamente coscienza delle prestazioni ottenute perché nei dati utilizzati non vi sarà presenza di errori di misura, di errori legati alla distanza tra il modello e la realtà e di errori legati alla stima della domanda che sono sempre presenti.

Una volta che attraverso questa prima fase, tutte le incognite relative alla procedura di ottimizzazione sono state risolte, allora si può passare a calare tale procedura sui dati reali. Ovviamente come già espresso in precedenza nell'utilizzare i dati reali, cura bisogna porre nella verifica della loro consistenza.

La necessità di utilizzare un procedura tanto complessa nasce dal fatto che il modello di simulazione è molto complesso e la funzione obiettivo da minimizzare che ne risulterà sarà a sua volta complessa nel senso che presenterà diverse discontinuità e multimodalità, come si evince dalla Figura 6—12 per soli due parametri del modello. In aggiunta l'ottimizzazione è un'ottimizzazione con alcune variabili intere ed, essendo un modello di simulazione e non disponendo di una formulazione esplicita necessita l'utilizzo di un approccio black-box.

L'obiettivo della procedura è trovare i valori dei parametri che meglio permettono al modello di riprodurre le condizioni reali. Esso è equivalente alla soluzione di un problema di minimizzazione vincolata, dove la funzione obiettivo esprime la deviazione tra le misurazioni di output simulate e quelle osservate. Quando il modello è di simulazione, per calcolare il valore della funzione obiettivo in ciascuno step dell'algoritmo di ricerca del minimo, si andranno ad eseguire una o più simulazioni.

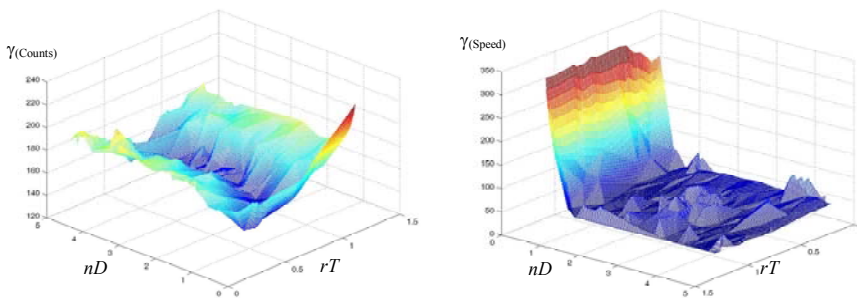


**Figura 6—11 Schema di calibrazione**

La scelta dell'algoritmo migliore per effettuare l'ottimizzazione proposta è stata effettuata interfacciando AIMSUN con il software di ottimizzazione LINDO API. In questo modo, ad ogni iterazione del particolare algoritmo



settato LINDO ipotizza un set di parametri, li passa ad AIMSUN che esegue una simulazione e compara gli output ricavati con i dati a disposizione finché la ricerca del minimo non è conclusa. Vista la complessità della funzione obiettivo che ci si trova ad esplorare l’algoritmo scelto è tale da scegliere diversi punti di partenza in modo da aggirare l’ostacolo dei minimi locali.



**Figura 6—12** Forma della funzione obiettivo al variare di due parametri del modello di simulazione

### 6.6.1. Implementazione della procedura di calibrazione

Come anticipato, con l’ausilio del software LINDO API, una procedura automatica è stata implementata per calibrare AIMSUN.

Abbiamo ottimizzato il modello in maniera implicita, nel particolare con un approccio “*black box*”, noto anche come *input-output*; tale dizione riflette il fatto che ciò che conosciamo del sistema è catturato solo dalla risposta di output  $g(u)$  allo stimolo in input  $u(t)$ . La specifica struttura interna del sistema non è specificata. Usando quindi un algoritmo di ottimizzazione non lineare “multi startpoint”, si cerca di trovare il punto di minimo assoluto, aggirando i minimi locali.

Il software comunica con AIMSUN attraverso un processo di scripting, richiamato sistematicamente in ciascuno step della procedura di ottimizzazione; procedura che lavora secondo questa sequenza:

- 1) legge i valori delle MoP che arrivano da un file di testo
- 2) genera una soluzione – i valori dei parametri – e li scrive in un file di testo
- 3) lancia uno script python che:
  - a. inizializza AIMSUN
  - b. carica la rete
  - c. prende i valori dei parametri dal file di testo
  - d. li assegna al modello
  - e. esegue differenti repliche della simulazione dello scenario
  - f. calcola la media degli output dalle differenti repliche e li immagazzina in un database Microsoft Access
  - g. chiude AIMSUN
- 4) legge risultati dal database
- 5) li paragona ai valori “reali” delle MoP
- 6) se la funzione obiettivo non ha trovato il suo valore minimo, genera un nuovo set valori dei parametri, altrimenti la procedura finisce e la soluzione viene salvata in un log file.

L'esatta formulazione della funzione obiettivo è

$$\text{Min } \gamma = \left[ \sum_{k \in K} \sum_{i \in I} \sum_{j \in J} (v_{ij}^{obs} - v_{ij}^{sim})^2 \right]^{0,5}$$

$$v^{sim} = F(od, rT, mDs, mA, nD)$$

$$rT_{car} = rT_{truck} \geq 0,1$$

$$mDs_{car} \geq mDs_{truck}$$

$$mA_{car} \geq mA_{truck}$$

$$nD_{car} \geq nD_{truck}$$

dove K, I, J sono rispettivamente il set delle categorie di veicoli considerati, i detector e gli intervalli di tempo dei rilievi. *od* è la matrice che include il modello di domanda distinto per ogni categoria di veicoli, da fornire ad AIMSUN come input. *rT*, *mDs*, *mA*, *nD* sono i vettori dei parametri differenziati per ciascuna categoria di veicoli.

I vincoli imposti servono per preservare il significato fisico del fenomeno.

### 6.6.2. Verifica della procedura – validazione

Prima di cominciare la calibrazione con i dati reali che si sono collezionati, vogliamo testare la procedura considerando, come dati reali, i risultati provenienti da una simulazione e cercando di riprodurre i parametri usati.

Questa fase è necessaria per:

- verificare la stabilità e l'efficienza della procedura
- confermare i risultati dell'analisi di sensitività

Per fare ciò, abbiamo utilizzato la stessa rete dell'analisi di sensitività, estratto casualmente i valori dei parametri, e simulato l'esperimento, i cui output-MoP sono stati confrontati con quelli di riferimento; il procedimento è iterato, fino a riprodurre i parametri estratti. In questa maniera non ci sono errori derivanti dai dati disponibili, né dalla stocasticità del modello, ci si prende cura infatti di avere in ogni esperimento replicato lo stesso seme, né

per la capacità del modello di riprodurre un particolare campo di applicazione che possa influenzare i risultati ottenuti.

Abbiamo eseguito diversi test, ciascuno dei quali ha richiesto 15–20 ore, cambiando sia il set di parametri da riprodurre, sia il MoP da usare nella funzione obbiettivo.

Dopo aver eseguito le simulazioni, per misurare la performance del modello globale e controllare se le misurazioni simulate realmente avvicinino quelle osservate, si è usato un approccio statistico.

In generale la letteratura esistente presenta un largo numero di metodi per una validazione statistica del modello di simulazione.

Questi approcci includono le misure di goodness of fit, intervalli di confidenza, intervalli di confidenza e test statistici. In molti casi, però non possono essere applicati perché sia i processi di traffico reali che quelli simulati, sono non-stazionari e auto-correlati. La scelta dei metodi appropriati e le loro applicazioni alla validazione dei modelli di simulazione del traffico dipende dalla natura dei dati di output.

Nonostante dal MoP usato nella procedura di calibrazione, questi test sono da eseguire su più di una misura, per essere sicuri che il modello sia capace di riprodurre il fenomeno reale.

Nella nostra applicazione, per verificare i risultati della calibrazione, abbiamo usato stime di goodness of fit e gli intervalli di confidenza sia sui flussi che sulle velocità.

Le misure di goodness of fit più comunemente usate sono l'errore sullo scarto quadratico medio (RMSe) e i coefficienti di disuguaglianza di Theil.

RMSe tende ad amplificare gli errori più piccoli penalizzando gli errori più grandi

$$RMSe = \sqrt{\frac{1}{N} \sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2}$$

$$U = \frac{\sqrt{\frac{1}{N} \cdot \sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2}}{\sqrt{\frac{1}{N} \cdot \sum_{n=1}^N (Y_n^{sim})^2 + \frac{1}{N} \cdot \sum_{n=1}^N (Y_n^{obs})^2}} = U_m + U_s + U_c$$

$$U_m = \frac{N \cdot (\bar{Y}_n^{sim} - \bar{Y}_n^{obs})^2}{\sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2};$$

$$U_s = \frac{N \cdot (s^{sim} - s^{obs})^2}{\sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2};$$

$$U_c = \frac{2 \cdot N \cdot (1 - \rho) \cdot s^{sim} \cdot s^{obs}}{\sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2}.$$

Il coefficiente di disuguaglianza di Theil U generalmente si trova scomposto in tre sottocoefficienti  $U_m$ ,  $U_s$  e  $U_c$ .

In particolare  $U_m$  rappresenta la tendenza e riflette l'errore sistematico;  $U_s$  è la varianza che indica quanto il modello di simulazione ben riproduce la variabilità dei dati osservati;  $U_c$  è invece la covarianza e rappresenta l'errore rimanente.

È anche possibile utilizzare i classici test d'ipotesi ed intervalli di confidenza. Per applicare un test con due campioni è necessario che ambo gli output, simulato ed osservato, siano indipendenti ed identicamente distribuiti (I.I.D.). Quindi è necessario che siano eseguiti separatamente in ciascun punto spazio-tempo; tuttavia, quando il numero dei dati disponibili in ciascun punto è tale da non ottenere buoni risultati, come suggerito da altri studi, possono essere raggruppati assieme ad osservazioni di un appropriato intervallo di tempo.

Il test da eseguire è sulla media di una variabile casuale ottenuta dalla differenza tra le misure simulate e quelle osservate ( $Y^{obs} - Y^{sim}$ ). Noi si vuole testare se con un certo livello di significatività,  $Y^{obs} = Y^{sim}$ .

Useremo il  $t$ -test nel seguente modo:

$$\Pr \left\{ \frac{\left| (Y^{obs} - Y^{sim}) - (\underline{Y}^{obs} - \underline{Y}^{sim}) \right|}{\sqrt{\frac{(s^{obs})^2}{n^{obs}} + \frac{(s^{sim})^2}{n^{sim}}}} < t_{\alpha/2, n} \right\} > 1 - \alpha$$

dove  $\underline{Y}^{obs} - \underline{Y}^{sim}$  sono le medie delle misure osservate e simulate,  $s^{obs} - s^{sim}$  sono le deviazioni standard dei valori, ancora, osservati e simulati,  $n^{obs}$  e  $n^{sim}$  rispettivamente i gradi di libertà osservati e simulati.

Imposto che  $\underline{Y}^{obs} - \underline{Y}^{sim} = 0$  allora la formulazione diventa:

$$\Pr \left\{ \frac{\left| Y^{obs} - Y^{sim} \right|}{\sqrt{\frac{(s^{obs})^2}{n^{obs}} + \frac{(s^{sim})^2}{n^{sim}}}} < t_{\alpha/2, n} \right\} > 1 - \alpha$$

Assumendo  $\alpha = 0,05$  e considerando di calcolare le misure mediate su 15minuti, ossia su tre gruppi di dati rilevato ciascuno ogni 5', otteniamo:

$$\Pr \left\{ \frac{|(Y^{obs} - Y^{sim})|}{\sqrt{\frac{(s^{obs})^2}{2} + \frac{(s^{sim})^2}{2}}} < 4.303 \right\} > 0.95$$

Da qui abbiamo calcolato la fascia di ammissibilità attorno alle misure osservate per comprendere graficamente i risultati del test sull'intera serie storica.

Questa fase traspare dalle Tabella 6—4 e Tabella 6—5 e dalla Figura 6—13. Nella tabella si nota con maggiore chiarezza la migliore approssimazione fornita con l'utilizzo della velocità come MoP. Le misure di bontà di stima sono mostrate per ciascun detector disponibile sulla rete.

idDet	Calibration with flows			
	RMSe		Theil's coefficient	
	speed	flows	speed	flows
151	31.122041	28.357588	0.176232235	0.108044946
737	25.27974	30.79908	0.162883795	0.147506283
743	12.384591	37.224171	0.072748026	0.286141478
744	35.286715	19.466495	0.225395325	0.230410033

**Tabella 6—4 Risultati della calibrazione utilizzando i flussi come MoP**

In generale ogni tentativo per calibrare il modello ha fornito buoni risultati, soprattutto, come detto, per gli esperimenti eseguiti usando la velocità come misura di performance.

idDet	Calibration with speed			
	RMSe		Theil's coefficient	
	speed	flows	speed	flows
151	21.09322641	10.07127378	0.120192	0.0401331
737	8.627777283	11.658926	0.061239	0.056412
743	5.458896546	10.06023525	0.0303204	0.0755704
744	4.926664217	12.31135429	0.0326094	0.1472187

**Tabella 6—5 Risultati della calibrazione utilizzando le velocità come MoP**

In ciascuna figura abbiamo i dati simulati e osservati per tre ore di simulazione. La linea azzurra rappresenta l'intervallo di confidenza in ciascun istante; è possibile notare che in entrambi gli esperimenti, i dati simulati sono all'interno della fascia di ammissibilità, ma nel caso della calibrazione eseguita usando la velocità, le misure simulate sono più vicine a quelle osservate, in altre parole più realistiche.

In aggiunta i parametri ottenuti dalla procedura di calibrazione stimano al meglio quelli presi in origine: nella tabella seguente viene mostrato il paragone tra i parametri imposti inizialmente ed i parametri trovati attraverso la procedura di calibrazione.

	$rT$	$mDs_1$	$mDs_2$	$mA_1$	$mA_2$	$nD_1$	$nD_2$
<i>imposed values</i>	0.7	120	85	3.5	1	4	4
<i>fund values</i>	0.63	113	90	3.6	1.1	4.4	4.1

**Tabella 6—6 Confronto tra i parametri ipotizzati e quelli ottenuti con la calibrazione**



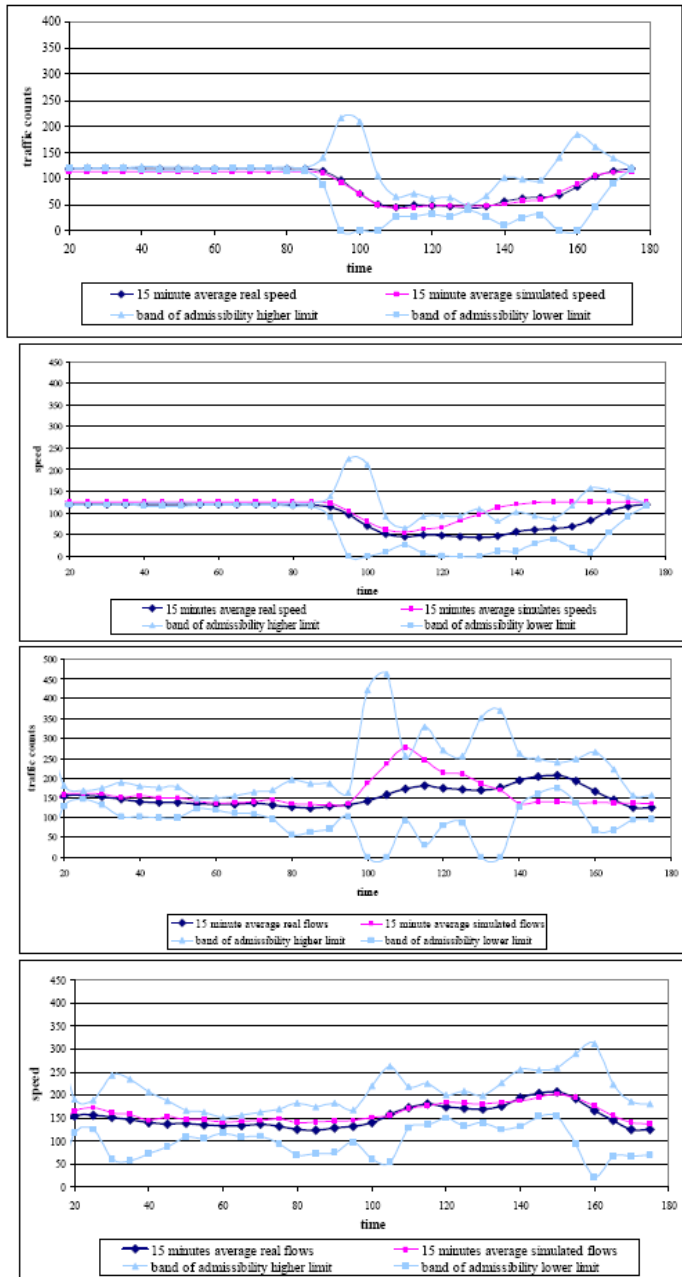


Figura 6—13 Risultati calibrations

Possiamo, quindi, concludere che:

- la procedura di calibrazione è perfettamente in grado di calibrare il modello su di una rete simulata
- il set dei parametri ottimali per la calibrazione è uno di quelli inclusi nell'analisi di sensitività
- la Measure of Performance maggiormente sensitiva è la *velocità*.

Infine, una volta verificata la bontà della procedura costruita, si può passare alla calibrazione su dati reali. I risultati del test sulle nostre due reti, Portici – Torre Annunziata in direzione Salerno e Angri – Scafati in direzione di Napoli, sono riportati rispettivamente nei grafici di Figura 6—14.

Si nota come tutti i punti siano interni all'intervallo di confidenza, e che quindi i valori dei parametri ricavati possono essere utilizzati per le successive applicazioni.

È molto interessante notare che la procedura ha fornito risultati soddisfacenti malgrado la mancata accuratezza dei dati rilevati. Più precisamente, da questi risultati si evidenzia che, anche con raccolte di dati non perfettamente congruenti, il processo di calibrazione del modello è capace di riprodurre il comportamento globale della rete.

## **6.7 Conclusioni**

Questi risultati confermano ancora l'efficienza della procedura costruita e dimostrano chiaramente l'importanza delle analisi sottolineate.

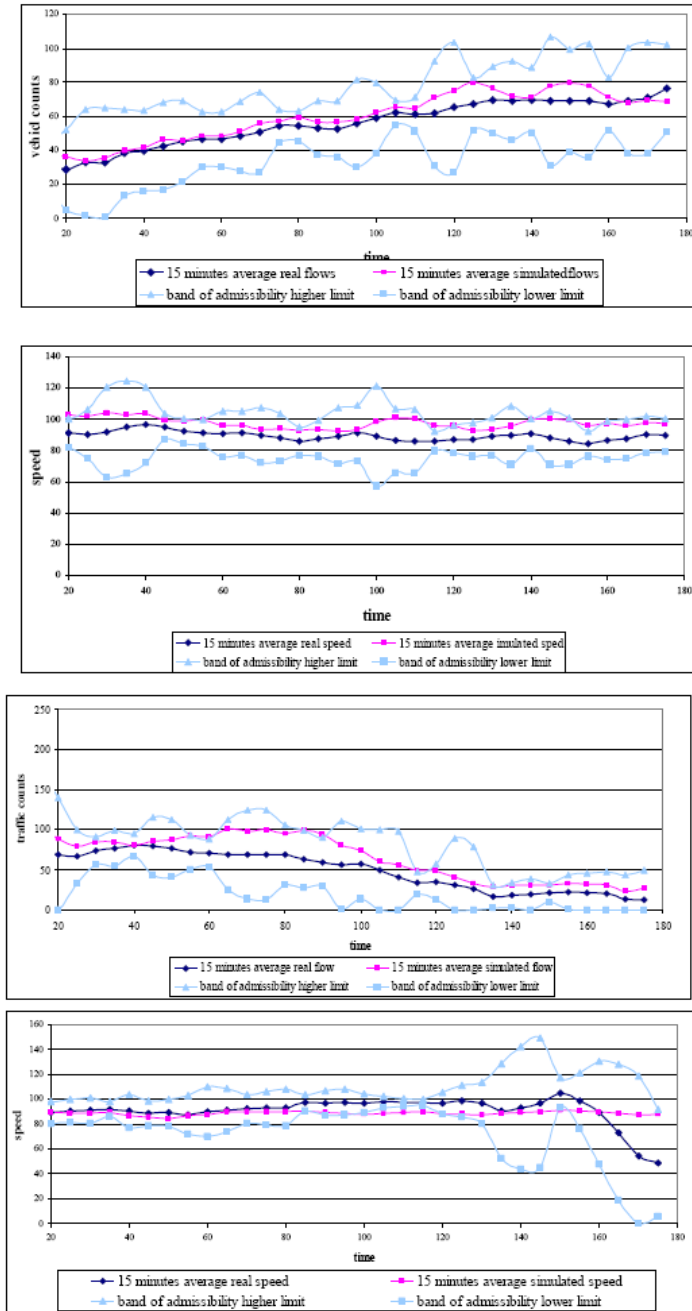


Figura 6—14 Risultati procedura di calibrazione su dati reali

L'obiettivo del presente lavoro è stato verificare l'efficienza e la consistenza della procedura di calibrazione realizzata e la sua applicabilità per riprodurre casi reali. La dimensione del problema ha suggerito fortemente una preliminare analisi di sensitività al fine di ridurre il numero di parametri da investigare. Dai primi risultati traspare come l'utilizzo della velocità in qualità di MoP, rispetto al flusso, permette una calibrazione più efficace.

Prima della calibrazione sui dati reali, si è eseguito un test per verificare la capacità dell'algoritmo di ottimizzazione di ricerca del minimo di una funzione obiettivo così complessa, e che ci conferma i risultati dell'analisi di sensitività. L'esito del test ci ha permessi di ritrovare dei valori dei parametri molto prossimi a quelli "veri" usati nella generazione dei dati di traffico sintetici usati per la calibrazione.

Alla fine è stata eseguita la calibrazione sui dati reali. Estrema cura si è avuta nella messa a punto di una procedura in grado di consentirci di scegliere dati consistenti e non "rumorosi", e di due differenti siti da investigare. L'applicazione di questa procedura ha messo in luce una complessiva scarsa qualità dei dati a disposizione. Ad ogni modo i risultati preliminari si sono mostrati più che accettabili, come confermato dalle indagini statistiche e da quelle visive sulle serie storiche simulate ed osservate.

In conclusione a questo capitolo si può senza dubbio affermare che il software di simulazione microscopica del traffico AIMSUN NG, abbia le caratteristiche sufficienti a garantire un'adeguata rappresentazione del

traffico per quel che riguarda le sue caratteristiche aggregate ancor più se prima di ogni applicazione una procedura quale quella descritta viene ripetuta per trovare i valori dei parametri che maggiormente permettono al modello di riprodurre la realtà.

### **6.8 Ulteriori sviluppi**

In appendice è riportato il lavoro che sarà presentato al prossimo TRB annual meeting 2008. In tale lavoro sono mostrati gli ulteriori sviluppi delle nostre applicazioni. In effetti la procedura mostrata, seppur risultata efficace ha presentato dei problemi in termini di efficienza computazionale richiedendo svariati giorni per essere compiuta. L'obiettivo del lavoro mostrato è stato quello di provare ad utilizzare la stessa procedura su un'altra rete per saggiarne la bontà e di aprire le porte ad una procedura alternativa che passa per la calibrazione cosiddetta model-based perché basata sull'applicazione dell'equazione di stato stazionario del modello di car-following alla base di AIMSUN e cioè il modello di Gipps.

Ulteriori studi si stanno tutt'ora portando avanti per sviluppare una nuova procedura, e saranno oggetto dell'interesse del tesista o di chi dovesse succedergli nel prossimo futuro.

## 7. Conclusioni del lavoro svolto

Il lavoro svolto durante i tre anni concernenti l'attività del dottorato di ricerca ha portato alla luce numerosi spunti su cui soffermare ulteriormente l'attenzione della ricerca.

L'obiettivo delle attività svolte è stato quello di studiare la simulazione del traffico nell'ambiente di simulazione di guida.

L'esperienza condotta ha, per grandi linee, mostrato i seguenti risultati pratici/concettuali:

1. i simulatori di guida sono strumenti che cominciano ad essere diffusamente utilizzati negli studi riguardanti l'interazione tra guidatore, veicolo ed infrastruttura nell'analisi della sicurezza stradale. Uno degli aspetti della simulazione che ha più impatto sul realismo della stessa è la qualità della rappresentazione del movimento dei veicoli cosiddetti *autonomi* che sono di contorno alla simulazione.
2. la simulazione del traffico all'interno degli scenari di simulazione di guida è un aspetto ancora solo marginalmente affrontato. Gli studi reperiti sono pochi ed in genere portati avanti da istituti di ricerca del settore di notevoli dimensioni e prestigio. Le strade individuate sono due e riguardano la creazione di un modello di traffico ad hoc o l'integrazione con uno dei modelli di traffico esistenti.
3. l'integrazione degli ambienti di simulazione di guida e di traffico è estremamente interessante, perché apre le porte alla completa modellizzazione dell'offerta di trasporto all'interno della

simulazione di guida, cosa che oggi viene effettuata con grande fatica e lentezza.

4. i benefici, in termini di simulazione del deflusso veicolare, dell'integrazione inoltre consentono finalmente di aprire a studi di traffico con il simulatore di guida e di mettere in pratica l'idea agognata del mutuo utilizzo dei due strumenti, l'uno per sviluppare modelli di traffico e l'altro per validare la simulazione di guida.
5. i benefici dell'integrazione si scontrano con l'inadeguatezza dei modelli di simulazione microscopica del traffico di rappresentare realisticamente il movimento dei singoli veicoli. Ad ogni modo la strada è sicuramente da intraprendere visto che tali strumenti si stanno sempre più sviluppando in questo senso avendo colto l'importanza di fornire un output grafico realistico per mostrare l'efficacia delle simulazioni.
6. la scelta del particolare modello di traffico esistente da utilizzare nell'integrazione è una fase fondamentale. A tale scelta è infatti legato il tipo di utilizzazione che del simulatore di guida si intende fare, visto che ogni modello ha un suo, più o meno vasto, specifico campo di applicazione. Una volta scelto, però, ci si deve assicurare che esso sia in grado di simulare anche in termini quantitativi il comportamento di una corrente veicolare e del traffico in generale.
7. la fase dell'integrazione è una fase molto delicata. Lo studio di tutte le possibilità e la scelta di quella che permette la maggiore flessibilità e la maggiore velocità di comunicazione è essenziale per poter usufruire di uno strumento in grado di simulare le più

complesse simulazioni di traffico su rete e i più sofisticati sistemi di aiuto alla guida.

8. l'utilizzo delle API per integrare i due software si è rivelato efficace ed efficiente da tutti i punti di vista. Tutti i problemi presentatisi sono stati risolti salvo quelli di rappresentazione del movimento dei veicoli che purtroppo è ancora strutturale nei modelli di microsimulazione.
9. la calibrabilità di un modello di simulazione microscopica del traffico è una caratteristica per nulla scontata. Il problema in letteratura, malgrado quanto ci si possa aspettare è ancora irrisolto sia da un punto di vista pratico che metodologico. Gli studi individuabili mancano ancora della proprietà di generalizzazione dei risultati ottenuti e ciò è anche testimoniato dall'assenza di appositi tool nei pacchetti commerciali.
10. il problema è stato quindi affrontato dal principio ponendo attenzione a tutte le assunzioni fatte e da farsi. Gli ostacoli riscontrati hanno spinto le nostre attività verso un approfondimento molto marcato in cui si sono evidenziate una serie di tematiche essenziali per la produzione di una simulazione di traffico di buona qualità.

L'attività svolta ha, inoltre, consentito al dottorando di accrescere notevolmente le proprie capacità e le proprie competenze tanto nel settore che si è trovato ad affrontare, tanto nella sfera generale dell'ingegneria dei trasporti.

La consapevolezza che ha acquisito tanto dei propri mezzi, quanto delle possibilità e delle potenzialità che la ricerca ha nel campo dello sviluppo



della società lo spronano a provare, per quanto irta di difficoltà e di dolore, a perpetrare la strada appena intrapresa.

## 8. Bibliografia

- [1] Abdullhai, B., J. B., Sheu, W. Recker. “*Simulation of ITS on the Irvine FOT Area Using Paramics 1.5 Scalable Microscopic Traffic Simulator: Phase I: Model Calibration and Validation*”. California PATH Research Report. 1999
- [2] Ahmad O., Papelis, Y. “*An autonomous driver model for the overtaking manoeuvre for use in microscopic traffic simulation*”. In Proceedings of the Driving Simulation Conference, Paris, France. 2000.
- [3] Ahmad O, Y. Papelis. “*A Comprehensive Microscopic Autonomous Driver Model for Use in High-Fidelity Driving Simulation Environments*”. In Proceeding of 81<sup>o</sup> TRB Annual Meeting. 2002.
- [4] AIMSUN 5.0 Microsimulator User’s Manual 5.0.6. Transport Simulation Systems 1997-2005. November 2005.
- [5] AIMSUN NG Scripting Version 0.7. Transport Simulation Systems © 2005. November 2005.
- [6] AIMSUN NG API Manual Version 5.1.2. Transport Simulation Systems © 2006. October 2006.
- [7] Al-Shinabi T., Mourant R. R. “*Toward More Realistic Driving Behavior Models for Autonomous Vehicles in Driving Simulators*”. In Proceedings of the 2002 TRB Annual Meeting. 2002.
- [8] Ambadipudi R., Dorothy P., R. Kill, Development and validation of large – scale microscopic models. TRB Annual Meeting, 2006.

- [9] Bang, B. and T. Moen. “*Integrating a driving simulator and a microscopic traffic simulation model*”. Trondheim, Norway, SINTEF (unpublished paper). 2004
- [10] Balakrishna, R., Antoniou, C., Ben-Akiva, M., Koutsopoulos H. N., Y. Wen. “*Calibration of Microscopic Traffic Simulation Models: Methods and Application*”. Transportation Research Records, 1999, 2007.
- [11] Barcelo J., J. Casas. “*Dynamic network simulation with AIMSUN*”. In Proceedings of: International Symposium on Transport Simulation, Yokohama. 2002.  
[http://www.aimsun.com/Yokohama\\_revised.pdf](http://www.aimsun.com/Yokohama_revised.pdf)
- [12] Barcelo J., J. Casas. Methodological notes on the calibration and validation of microscopic traffic simulation models. TRB Annual Meeting, 2004.
- [13] Ben Akiva M., Darda D., Jha M., Koutsopoulos H., T. Toledo, Calibration of Microscopic Traffic Simulation Models with Aggregate Data. TRB Annual Meeting, 2004.
- [14] Bonakdarian, E., J. Cremer, J. Kearney, Willemsen, P. “*Generation of Ambient Traffic for Real-Time Driving Simulation*”. In Proceedings of the IMAGE Conference, Scottsdale, Arizona, USA. 1998.
- [15] BOSS/QUATTRO 5.0 manual. Samcef, 2004.
- [16] Brodin, A. and Carlsson A. “*The VTI traffic simulation model – A description of the model and programme system*”. VTI meddelande 321A. Swedish National Road and Transport Research Institute (VTI), Linköping, Sweden. 1986

- [17] Champion, A., R. Mandiau, C. Kolski, A. Heidet, and A. Kemeny. Traffic generation with the SCANeRII simulator: towards a multi-agent architecture, in *Proceedings of Driving Simulator Conference, DSC'99*, Paris, France, 1999.
- [18] Champion, A., S. Espié, and J.-M. Auberlet. Behavioral Road Traffic Simulation with ARCHISIM, in *Proceedings of Summer Computer Simulation Conference*, Orlando, USA, 2001.
- [19] Chu L., Liu H. X., Oh J. S., W. Recker, A Calibration Procedure for Microscopic Traffic Simulation. TRB Annual Meeting, 2004.
- [20] Davidsson, F., Kosonen I. and Gutowski A. “*TPMA Model-1 final report*”. Centre for Traffic Research, Royal Institute of Technology, Stockholm, 2002.  
[http://www.infra.kth.se/ctr/projekt/tpma/tpma\\_en.htm](http://www.infra.kth.se/ctr/projekt/tpma/tpma_en.htm)
- [21] Dowling R., Skabardonis A., Halkias J., Mc Hale G., G. Zammit. Guidelines for Calibration of Microsimulation Models: Framework and Applications. Transportation Research Record, 1876, 2004.
- [22] Erto P. Probabilità e statistica per le scienze e l'ingegneria. McGraw-Hill, 2004.
- [23] Espié, S. “*ARCHISIM: Multiactor parallel architecture for traffic simulation*”. In Proceedings of the second world congress on Intelligent Transport System, Yokohama. 1995
- [24] FHWA. “*CORSIM User Manual VI.01*”. Federal Highway Administration, McLean, USA. 1996

- [25] Gregersen, N. P., T. Falkmer, J. Dols and J. Pardo (2001), *Driving simulator scenarios and requirements*. TRAINER-project: System for driver Training and Assessment using Interactive Evaluation tools and Reliable Methodologies. [http://www.trainer.iao.fraunhofer.de/deliverables/TRAINER%20D4.1\\_Final.zip](http://www.trainer.iao.fraunhofer.de/deliverables/TRAINER%20D4.1_Final.zip)
- [26] Hoban C. J., Shephard R. J., Fawcett G. J., G. K. Robinsson. “*A model for simulating traffic on two lane rural roads – User guide and manual for TRARR version 3.2*” Technical Manual ATM No. 10B. Australian Road Research Board, Victoria, Australia. 1991.
- [27] Hourdakis J., Michalopoulos P. G., J Kottommanni, A practical procedure for calibrating microscopic traffic simulation models. *Transportation Research Record*, 1852, 2003.
- [28] INRETS (2001). *Driving simulator and simulation*. [on-line] [http://www.inrets.fr/ur/sara/Pg\\_simus\\_e.html](http://www.inrets.fr/ur/sara/Pg_simus_e.html)
- [29] Jayakrishnan, R., J.-S. Oh, A.-E.-K. Sahroui. “*Calibration and Path Dynamics Issues in Microscopic Simulation for Advanced Traffic Management and Information Systems*”. *Transportation Research Record*, 1771, 2001.
- [30] Jenkins J. M., Rilett L. R. “*Integrating Driving Simulators and Micro-Simulation Models: A Conceptualization*”. In *Proceeding of 82<sup>o</sup> TRB Annual Meeting*. 2002
- [31] Jha M., Goplan G., Garms A., Mahanti B. P., Toledo T., M. Ben Akiva. Development and calibration of a large – scale microscopic traffic simulation model. *Transportation Research Record*, 1876, 2004.

- [32] Jin M., Lam S.-H. “*A Virtual-Reality Based Integrated Driving-Traffic Simulation System to Study the Impacts of Intelligent Transportation Systems (ITS)*”. In Proceedings of the 2003 International Conference of Cyberworlds. 2003.
- [33] Kim, S.-J., Kim, W., L. R. Rilett. “*Calibration of Microsimulation Models Using Nonparametric Statistical Techniques*”. In Transportation Research Records: Journal of the Transportation Research Board, No. 1935, 2005.
- [34] Law A. M., W. D. Kelton. Simulation Modeling and Analysis. McGraw-Hill, International Edition, 2000.
- [35] Leiman, L., Archilla A. R., A. D. May. “*TWOPAS model improvements*”. University of California, Berkeley, USA.
- [36] LINDO API manual, Lindo Systems inc.1804.
- [37] Ma, T., B. Abdullhai. “Genetic Algorithm-Based Optimization Approach and Generic Tool for Calibrating Traffic Microscopic Simulation Parameters”. Transportation Research Record, 1800, 2002.
- [38] Ma, J., Dong, H., H. M. Zhang. “*Calibration of Micro Simulation with Heuristic Optimization Methods*”. In Transportation Research Records, 1999. 2007.
- [39] MATLAB 6 User Manual. The MathWorks ©. 2000.
- [40] Michon, J. A. “*A critical view of driver behavior models: What do we know, what should we do?*”. L. Evans & R. C. Schwing: Human behaviour and traffic safety. (pp. 485–520). New York: Plenum Press. 1985

- [41] Min B.-C., Chung S.-C., Min Y.-K., K. Sakamoto (2004), *Psychological evaluation of simulator sickness evoked by a graphi simulator*. Applied ergonomics, n. 35, pp. 549-556
- [42] Olstam J. J. “*A model for simulation and generation of surrounding vehicles in driving simulators*”. Linköping Studies in Science and Technology Licentiate Thesis No. 1203. 2005
- [43] Park B. B., J. D. Schneeberger. Microscopic Simulation Model Calibration and Validation: A Case Study of VISSIM for a Coordinated Actuated Signal System. Transportation Research Record, 1856, 2003.
- [44] Park B., H. Qi. “*Development and evaluation of a procedure for the calibration of simulation models*”. Transportation Research Record, 1934, 2005.
- [45] Pindyck, R.S., Rubinfeld, D.L.,. Econometric Models and Economic Forecasts, fourth ed. McGraw-Hill, 1998.
- [46] PTV. “*VISSIM User Manual – Version 3.70*”. PTV Planung Transport Verkehr AG, Karlsruhe, Germany. 2003
- [47] Punzo V., F. Simonelli. Analysis and Comparison of Car-following Models Using Real Traffic Microscopic Data. Transportation Research Records 1934, pp. 53-63. TRB, Washington, 2006.
- [48] Quadstone Ltd. “*Paramics Modeller 3.0 User Guide*”. Edinburgh: Quadstone Ltd. (2000).
- [49] Quadstone. “*Quadstone Paramics V5.0 – Viewer User Guide*”. Quadstone Limited, Edinburgh, Scotland. 2004
- [50] SCANer II. User’s Manual. Release 2.12. 2006 February 23.

- [51] SCANeR II API Programming guide v2.16a. Oktal © 2006.  
28 Novembre 2006.
- [52] Schultz, G.G., L.R. Rilett. “*Calibration of Distributions of Commercial Motor Vehicles in CORSIM*”. Transportation Research Record, 1934. 2005.
- [53] SIKa, *Swedish Institute For Transport and Communications Analysis*, [http://www.sikainstitute.se/english\\_fr.html](http://www.sikainstitute.se/english_fr.html), Accessed June 21, 2004.
- [54] Tapani, A. “*A versatile Model for Rural Road Traffic Simulation*”. In Proceedings of the 84<sup>th</sup> Annual meeting of the Transportation Research Board, Washington D. C., USA. 2005
- [55] Toledo T., Koutsopoulos H.N., Davol A., Ben-Akiva M.E., Burghout W., Andréasson I., Johansson T. and C. Lundin. Calibration and Validation of Microscopic Traffic Simulation Tools: Stockholm Case Study. Transportation Research Record, 1831, 2003.
- [56] Toledo T., H. N. Koutsopoulos, Statistical validation of traffic simulation models. Transportation Research Record, 1876, 2004.
- [57] Van Driel, C.J.G. and Van Arem, B. (2005) *Integrated driver assistance from the driver’s perspective; Results from a user needs survey*, University of Twente, Centre for Transport Studies, Enschede, The Netherlands
- [58] Van Wolffelaar, P.C. “*Functional Aspects of the Driving Simulator at the University of Groningen*”. University of Groningen, Groningen, The Netherlands. 1999.



- [59] Wright S., Ward N. J., Cohn A. G. “*Enhanced Presence in Driving Simulators Using Autonomous Traffic with Virtual Personalities*”. *Presence*, Vol. 11, No. 6, December 2002, 578–590. Massachusetts Institute of Technology. 2002
- [60] [www.centroregionaledicompetenzatrasporti.unina.it](http://www.centroregionaledicompetenzatrasporti.unina.it)
- [61] [www.istat.it](http://www.istat.it) – [www.aci.it](http://www.aci.it)

## Appendice A – Pubblicazioni

La produzione scientifica scaturita dalle esperienze raccontate nel presente documento è la seguente:

1. Ciuffo, B., Punzo, V., V. Torrieri. “*A framework for calibration of microscopic traffic flow models*”. In Proceedings of the 86<sup>th</sup> Transportation Research Board Meeting. Washington D. C. 2007.
2. Ciuffo, B., Punzo, V., V. Torrieri. “*Towards the integration of simulation-based and model-based calibrations of traffic micro-simulation models*”. Accettato per la presentazione durante l’87<sup>th</sup> Transportation Research Board Meeting. Washington D. C. 2008.
3. Ciuffo, B., Punzo, V., Torrieri V. “*Integrated Environment of Driving and Traffic Simulation*”. Accettato per la presentazione al Road Safety and Simulation International Conference. Roma, 7-9 Novembre 2007 e presentato al XII SCANeR User Group Meeting, Napoli, 25-27 Ottobre 2007.
4. Ciuffo, B., Galante F., Perneti M., Punzo, V., Torrieri V. “*Tunnel in fire, evaluation of different VMS warning messages by driving simulator*”. Accettato per la presentazione al Road Safety and Simulation International Conference, Roma, 7-9 Novembre 2007.

I primi tre sono riportati in calce nel seguito della presente sezione.

## **A framework for the calibration of microscopic traffic flow models**

### **Biagio Ciuffo**

PhD Student  
Department of Transportation Engineering  
Università di Napoli “Federico II”  
Via Claudio, 21 – 80125 Napoli – ITALY  
Ph.: +39.081.7683771  
Fax: +39.081.7683740  
E-mail address: [bciuffo@unina.it](mailto:bciuffo@unina.it)

### **Vincenzo Punzo\***

PhD, Assistant Professor  
Department of Transportation Engineering  
Università di Napoli “Federico II”  
Via Claudio, 21 – 80125 Napoli – ITALY  
Ph.: +39.081.7683948  
Fax: +39.081.7683946  
E-mail address: [vinpunzo@unina.it](mailto:vinpunzo@unina.it)

### **Vincenzo Torrieri**

Full Professor  
Department of Transportation Engineering  
Università di Napoli “Federico II”  
Via Claudio, 21 – 80125 Napoli – ITALY  
Ph.: +39.081.7683368  
Fax: +39.081.2390366  
E-mail address: [torrieri@unina.it](mailto:torrieri@unina.it)

Words: 5864  
Plus 2 Tables: 500  
Plus 5 Figures: 1250  
TOTAL: **7614**

\* Corresponding author

**ABSTRACT**

In view of the widespread use of microscopic traffic simulation models in common practice an ever more attention on their calibration related issues is bearing. These models, indeed, depend on many parameters, and the value chosen per each of them can sensibly affect model output realism. The only way to undertake is, therefore, their calibration on real traffic data.

Hence the work proposes a framework for calibrating microscopic traffic flow models that aims, in particular, to address the issues of MoP choices and calibration parameters. The whole procedure focuses on the execution of one or more tests with laboratory data. This is the only way to overcome the effect of the errors that arise when using empirical data and that regards mainly *o/d* flow estimation, the data measuring process and mathematical modelling of real dynamics. A crucial element of the proposed procedure is model sensitivity analysis as well, performed by using an Analysis of Variance (ANOVA). Preliminary results, obtained using the microscopic traffic simulation model AIMSUN in freeway conditions, show the capability of the framework of successfully facing the calibration problem.

## INTRODUCTION

While microscopic traffic simulation models are increasingly used in common practice, they are becoming ever more complex to satisfy existing demand. Hence there is a dependence on a higher number of parameters whose values are not known *a priori*, and an increasing need of calibration [(1),(2)].

Calibration is likely to be the most complex phase in the micro-simulation of a traffic scenario. The complexity of the environment, phenomena and corresponding models on the one hand, and the difficulty/impossibility in collecting (direct) data on dynamics to be reproduced on the other, are recognised as the main hurdles for problem solution. The absence of suitable automatic tools for calibration in commercial simulation software is the main evidence for this.

Models used in micro-simulation of traffic flows have to reproduce complex human behaviour such as that resulting from driving or travel tasks. In addition, the whole modelling procedure generally uses *o/d* flows as input that are always estimates of the “true” ones. On the other hand, to calibrate the different models separately, detailed and uncorrelated data would be required on each single phenomenon e.g. detailed vehicle trajectories for traffic flow models (3), drivers’ path choices for path choice models, etc. Instead, data available in practice are almost always aggregate measurements of traffic characteristics (e.g. detector counts and speeds, etc.), which are the result of the combination of all the dynamics involved in a traffic environment (4). Therefore their use does not in principle allow independent calibration of micro-simulation sub-models. Hence, despite the number of works existing on this topic, an approach to problem solution is not yet well established.

The common approach to limiting all existing problems consists in isolating elementary sectors of a network in which dynamics could be treated separately. In this way some studies deal only with the calibration on network sectors in which only flow dynamics are included (e.g. a road stretch or freeway links) [(5),(6),(7),(8)]. Other works use “*sequential*” approaches in which elementary network sectors are first isolated to carry out only flow propagation model parameter estimation, and then more extended network portions are considered, in which other dynamics are included, such as path choice or demand pattern [(4),(9),(10)]. In such works, the approach to the optimization procedure was investigated. Trial and error or automatic procedures and their algorithms were variously tested [(4),(6),(11)].

Other fundamental issues in microscopic traffic simulation model calibration, such as the choice of the parameters to calibrate, or the choice of the Measures of Performance (MoP) to use in the optimization problem formulation, are only marginally dealt with in the literature [(11)].

Regarding the first issue, as anticipated, traffic simulation software depend on a huge amount of parameters. The calibration of all of them is practically impossible since computing efforts are considerable. Conversely, the wrong choice of what number and which parameters do not have to be considered for calibration can compromise the calibration results themselves and the ability of the model to reproduce real traffic conditions. In other words the choice of parameters for calibration is the result of a trade-off between optimization feasibility and model accuracy. The choice of the MoP to be used in the objective function is relevant as well. Usually the objective function is made of a combination of available MoPs [(4),(9),(12)], but the effects of these choices on calibration results are not well defined.

Therefore this work proposes a framework for calibrating microscopic traffic flow models that aims, in particular, to address the issues of MoP choices and calibration parameters. The whole procedure focuses on the execution of one or more tests with laboratory data. This is the only way to overcome the effect of the errors that arise when using empirical data and that regards mainly *o/d* flow estimation, the data measuring process and mathematical modelling of real dynamics. A crucial element of the proposed procedure is model sensitivity analysis as well, performed by using an Analysis of Variance (ANOVA). The choice of the optimization algorithm was made *a priori* and does not concern this paper.

In the following, the methodology of the proposed framework is first described. Then the procedure’s implementation is presented, the results are discussed and conclusions drawn.

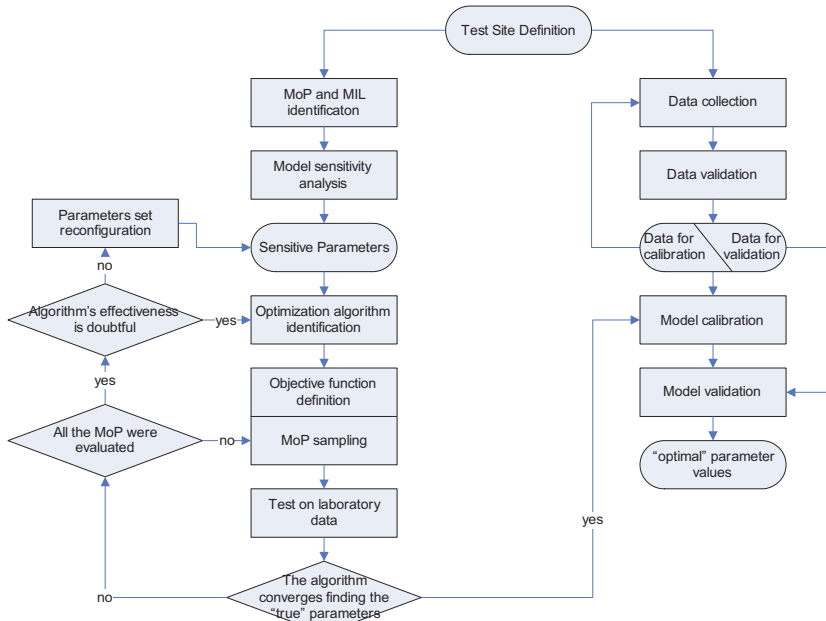
## METHODOLOGY

The conceived framework for calibration and validation of traffic simulation models is represented in Figure 1. The procedure is divided into two main parts. The first part (on the left-hand side in Figure 1) concerns *calibration procedure set-up*; the second part (on the right) concerns *model calibration on a real scenario*.

The calibration procedure set-up deals with all the calibration related problems. During this phase, the form of the objective function and the Measures of Performance to use are established, the parameters involved in the calibration are chosen, the optimization algorithm is identified and the actual possibility of calibrating the model is analyzed. During the second phase the procedure is applied on a real site.

However, the test site definition has to be a preliminary phase in the overall calibration framework. All the simulation model analyses, indeed, have to be carried out on the same test bed, that is, the same that will be used in model calibration on a real scenario.

In the remainder, attention will be mainly focus on the calibration procedure set-up. Results of the procedure application on a real site will also be shown at the end of the paper.



**Figure 1 Overall calibration and validation framework**

**Test Site Definition**

Test site definition is a preliminary phase of the calibration procedure. It not only establishes the data to be collected but also influences all the analyses to be carried out. In order to obtain valid results, both the model sensitivity analysis and the test on laboratory data were performed on the same test bed that is the real site on which the simulation model has to be calibrated. In particular, in the present work a freeway scenario was considered to deal only with flow dynamics.

**Calibration Procedure Set-up**

*MoP and MIL identification*

The development of the calibration procedure starts with identifying a set of MoPs and their measurement interval length (MIL). This identification depends on the particular case study and on actual MoP availability (e. g. in a freeway context, counts and speeds could be the most suitable and easily available data). However, which specific MoP to use in the calibration procedure cannot be chosen a priori but, in an iterative way, only at the end of the calibration set-up. Hence, in the next phases all the useful MoPs have to be considered.

*Model sensitivity analysis*

The second step is the choice of the “optimal” set of parameters for calibration. It consists of the minimum number of input parameters able to explain most of the output’s variance. Therefore, it is necessary to isolate the model’s parameters to analyze and to understand their influence on the model’s output. With this purpose, a *sensitivity analysis* of the model has to be carried out (1). Different parameters obviously have different influence on the model, that is to say that each parameter explains its proper share of the model output’s variance. Moreover, the effects of parameters interaction also explain part of the output’s variance, which should not be neglected. To obtain these information it is possible to use a statistical technique like the Analysis of Variance (ANOVA) (13).

In order to fully evaluate all the effects on the model's output, once the number of values (levels) each parameter can assume has been defined, one should first run as many simulations as the possible combinations of values. In the *experimental design* literature, this method is known as *full factorial design* (13). Other methods exist to reduce the number of simulations to carry out, but they also lead to a less accurate parameters analysis.

Once all the simulations results have been collected, the ANOVA technique can be used. It isolates the experimental variance due to each parameter, and to its interaction with the others (13). Having estimated the variance of each parameter, it is possible to test the hypothesis whether or not this variance is negligible with respect to the variance due to the whole model. The test is made using a Fisher probability distribution. If the hypothesis can be rejected, it is possible to consider, with a certain level of significance, the model sensitive with respect to that parameter (13). As a result of model sensitivity analysis, the more sensitive parameters are highlighted.

In the traffic simulation modelling literature, the use of the ANOVA technique to isolate the more sensitive parameters is not very widespread. In Park et al. (11) ANOVA was used to catch the sensitivity of eight parameters. Five levels per parameter were considered. To perform a full factorial design  $5^8=390,625$  simulations need to be run. To reduce the number of simulations, a Latin Hypercube Sampling algorithm was used: only 200 simulations were performed. Furthermore, during analysis of variance the effect of parameters interaction was not evaluated. Though interesting, the reliability of results from an ANOVA carried out on 200 simulations instead of 390.625 needs to be thoroughly investigated. It would be probably better to reduce the number of levels per parameter and to perform a full factorial design. Besides, the effect of parameters interactions on model outputs, as will be highlighted, is not negligible.

#### *Optimization algorithm identification and objective function definition*

The procedure set-up continues identifying the objective function and the optimization algorithm to use.

The aim of the calibration procedure is to find parameters values that best allow the model to reproduce real conditions. It is equivalent to the solution of a constrained minimization problem where the objective function  $\gamma$  expresses the deviation of simulated output measurements from those observed. Formulation of the constrained minimization problem is chosen by reference to the literature [(4),(14)]:

To solve this optimization problem it is possible to use trial and error or automatic procedures. The second way requires definition of an algorithm which, for each step, imposes parameter values and calculates the value of the objective function until a minimum is reached. When the model is a simulation one, in order to calculate the value of the objective function, at every step of the algorithm, one or more simulations have to be performed. Besides, as the model is a nonlinear, non-convex one, the optimization problem has two main difficulties:

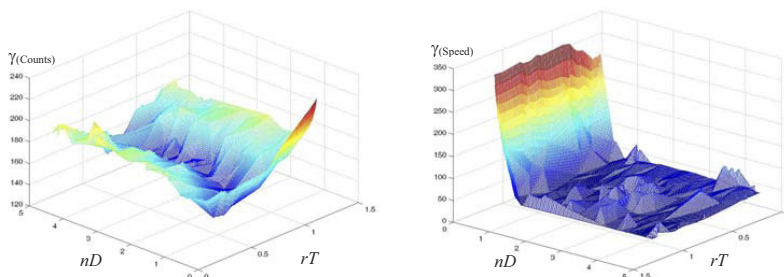
- there is an unknown number of locally optimal solutions in the feasible solution set;
- the quality of these local solutions varies substantially.

For this reason, the choice of the algorithm has to be made accurately.

#### *MoP sampling*

In the next phase, it is necessary to sample the MoP to use in the objective function and the length of its measurement interval.

The MoP chosen affects the effectiveness of calibration procedure as it influences the shape of the objective function. As an example in Figure 2, two model response surfaces for two different MoPs are shown. The surfaces obviously refer to the same model parameters (Simulation Step ("rT") and Normal Deceleration ("nD")).



**Figure 2 Model's response surfaces using traffic counts (on the left) or speed (on the right) as MoP**

The presence of several local minima and the strong difference between the two surfaces are highlighted in the figure. It is evident that the same algorithm could achieve different results using one MoP rather than another, or rather than their combination. The importance in choosing the most suitable MoP for the optimization algorithm is then evident. This choice may be effected by performing a *test* of the calibration procedure with *laboratory data*.

#### *Test with laboratory data*

The test is built as follows: parameters values are set and a simulation is run. Simulation outputs are then collected and considered as real data coming from virtual detectors. Parameters values are then changed and a calibration is performed in order to test its ability to find the starting parameters values once again.

This simple test is necessary to confirm results from the sensitivity analysis and to look at the effectiveness of the calibration structure. Thus, all the real life uncertainties are eliminated: the demand pattern and the real parameters values are completely known and there is no model error in reproducing real life conditions. It is also possible to avoid the effect of model stochasticity, using the same random seed for each simulation. It is thereby possible to focus exclusively on the effect of the sampled MoP, the algorithm's behaviour and the parameters chosen after sensitivity analysis. In spite of its usefulness, a similar approach is not commonly used in the literature.

When the test is unable to find again the parameters starting values again, then only the previous factors have to be investigated. Therefore in the calibration framework, an iterative procedure is presented to evaluate all possible problems.

Three iteration levels are considered. In the first level, the MoP influence on the minimum searching is tested. If the starting parameters values are not found, this level ends when all the identified MoPs have been tested. In this phase, also the MIL effect on the procedure is examined. In the second level, the algorithm's ability in exploring the objective function is analyzed. This level ends when the most appropriate algorithms have already been tested. Finally, the third level concerns the chosen parameter set. In this case, the results of sensitivity analysis have to be re-examined and the parameter set reconfigured. This probably means that not enough of the model's explained variance had been taken into account.

At the end of the *Calibration Procedure Set-up*, the form of the objective and the algorithm to use are set, the parameters to calibrate are known, the MoP and the length of its measurement interval to use in the objective function are also established. It is then possible to start the calibration on real life data.

### **Model calibration on a real scenario**

#### *Data collection and validation*

The first phase in model calibration on a real scenario is data collection. Once data have been collected, their validation is necessary. This phase is necessary to detect any measurement errors that may be present. A first simple verification consists in verifying that all the inflows and outflows of a network coincide. Another more accurate check is made by constructing cumulative diagrams for the inflows and outflows. In this case, such verifications are:

- the cumulative curve for inflows always has to be above that of outflows;
- the cumulative daily value of the two flows has to be the same for both of them;
- for each flow cumulative value, the two curves must have a time distance not less than the running time between the downstream inflow sensor and the upstream outflow sensor.

#### *Model calibration*

When data have been validated, it is possible to calibrate the model by using them. Once the procedure has found parameters values that best allow the model to reproduce real traffic conditions, it is necessary to check whether simulated measurements really match the observed ones. Despite the MoP used in the calibration procedure, different tests can be carried out. Among others, goodness-of-fit and confidence intervals approaches can be used (15).

#### *Model validation*

The last phase of the whole calibration framework is model validation. During this phase the model's ability to reproduce different traffic conditions is tested. Unlike calibration, validation consists of a simple simulation in which the model tries to reproduce real data by using parameters calibrated on the basis of other real data. With this aim, the model can be validated by comparing the results of a simulation with data collected in the same network used for calibration, but in a different moment. Again, it can be done on another network



with the same characteristics of the network used for calibration. The approach to follow for the validation problem can be chosen among the several ones found in literature [(14),(15)].

## IMPLEMENTATION AND RESULTS

AIMSUN 5 (16) is the traffic simulation model used in this work. It is a microscopic and stochastic simulator capable of simulating several traffic conditions and road layouts. In particular, freeway conditions were simulated.

### Test Site Definition

Freeway E32 from Naples to Salerno in the southern Italian region of Campania was the test site for this work. All the analyses were carried out on a 15 km stretch, in the Naples direction, with the Angri interchange approximately at its centre (Figure 3). Existing detectors ensured the complete knowledge of traffic counts and speed divided by vehicle categories. The measurement interval was one minute long. To obtain the outputs to use in the analyses, four detectors among the existing ones were simulated in AIMSUN: two are positioned between the interchange ramps (one per lane) and other two at the end of the stretch. During the morning peak hours, heavy traffic conditions exist on the stretch. Hence, half a simulation was constructed with light traffic conditions and the other half with congested traffic. A bottleneck was simulated on the on ramp for one hour. In Figure 3 the dynamic of counts on a detector is also shown.

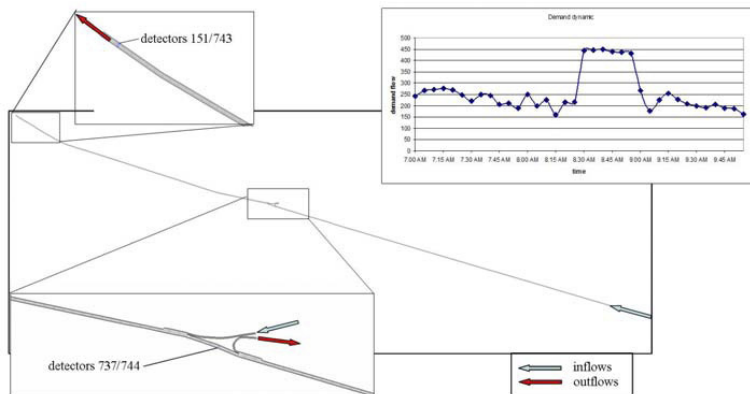


Figure 3 Network selected and inflow dynamic used in the analyses

### Calibration Procedure Set-up

#### MoP and MIL identification

Due to the particular test site, traffic counts and speed were used as MoP. These data are available for a 1-minute measurement interval. However, in aggregating available data, a MIL of 5 minutes was also considered.

#### Model sensitivity analysis

Parameters to analyze during sensitivity analysis depend on the model used and on the considered field of application. In freeway conditions, only flow propagation dynamics have to be considered. To correctly reproduce them, AIMSUN uses several parameters. They can be divided as follows (according to the AIMSUN's notation):

- *Vehicle Attributes;*
- *Local Parameters (those parameters able to affect vehicle behaviour but related to the single section (16));*
- *Global Modelling parameters (those parameters that are valid for all the vehicles moving on the whole network during the entire simulation (16)):*
  - *General*
  - *2-lanes Car – Following*
  - *Lane Changing*
  - *Others*

Of the above parameters, only *vehicle attributes* and *global modelling parameters* were considered. The value of local parameters was considered known. In particular nine parameters were chosen:

- Global modelling parameters
  - *Simulation Step (Reaction Time) "rT"*;
  - *Percent Recover "pR"*;
  - *Percent Overtake "pO"*;
  - *Reaction Time At Stop "rTs"*.
- Vehicle attributes
  - *Max Desired Speed "mDs"*;
  - *Max Acceleration "mA"*;
  - *Normal Deceleration "nD"*;
  - *Max Deceleration "mD"*;
  - *Give Way Time "gWi"*.

If 2 or 3 vehicle categories are considered in the simulation, given that *Vehicle Attributes* differ for each of them, the number of parameters to be calibrated grows to 14 or 19. Hence the current phase has an important meaning.

Simulations were run using a full factorial design. First, two levels per parameter were considered, therefore needing  $2^9=512$  simulations. As will be shown, ANOVA application on the obtained outputs was not satisfactory. For this reason, a 3-level full factorial design was also implemented, requiring  $3^9=19.683$  simulations. In order to manage such a large number of simulations the software BOSS/QUATTRO (17) was used. Connection between BOSS/QUATTRO and AIMSUN was possible using batch files (.bat) and python scripts (.py) [(17),(18)]. The built structure of communication proved stable allowing several thousands of simulations to be performed (and approximately 13 days to be executed). To avoid AIMSUN's stochasticity effects, attention was paid to using the same random seed in each simulation.

The results collected from the simulations were the measurements available at the test site: counts and speed. 1 minute and 5 minutes were the MIL considered. For each output and each MIL, an analysis of variance was carried out. The main results are shown in Table 1, which also presents sensitivity analysis of the six most sensitive parameters and their first level interactions. F represents the Fisher distribution used in the ANOVA to carry out the hypothesis test. F-Level represents the value of the Fisher distribution calculated during the analysis; F-Level<sub>0.95</sub> is the value of a Fisher distribution at a residual probability of 0.05. 0.95 is the level of significance chosen in the test.

As anticipated, the results obtained from the analysis of variance using a 2-level full factorial experimental design, with both counts and speeds as MoPs weren't satisfactory: few parameters were significant and more than the 90% of the variance was explained by only one parameter, the *Simulation Step* (Table 1-A). In AIMSUN, the simulation step has direct or indirect influence on the drivers' *Reaction Time* and this makes it an important parameter, though perhaps not so much as to justify the results obtained. Results of the ANOVA on the outputs coming from a three-level factorial design were completely different (Table 1-B/C), demonstrating the inappropriateness of using a two-level design to test non-linear models.

All the results shown refer to MoPs collected with a measurement interval length of 5 minutes since, using MoPs collected every minute, the model proved too sensitive to each parameter, without an actual physical justification.

The main results of the analysis are:

- in a freeway scenario, the most important parameters are the *Reaction Time*, the *Max Desired Speed* and the *Max Acceleration*: their influence on the model's outputs is much higher than that of the others. *Max Deceleration* has no influence at all since it influences emergency brakes that are not present here. The others (*Percent Recover*, *Percent Overtake*, *Give Way Time* and *Normal Deceleration*) have a certain influence on the model, but their explained variance is very small. However, for the first three parameters this is probably due to the *jam flow conditions* existing on 5 km of the stretch for about one hour and a half starting from the entry ramp that reduces the influence of the lane-changing model and consequently the influence of its parameters. As a conclusion, the influence of *Percent Overtake* and *Normal Deceleration*, which explain slightly higher speed variance (2-3%) will be tested in the calibration procedure. The others were definitely removed from the "optimal" parameters set;
- using speed as an MoP, 94% of its variance is explained by three parameters: *Reaction Time*, *Max Desired Speed* and *Max Acceleration*. On the other hand, using counts as a measure of performance, 94% of its variance is explained by only two parameters: *Reaction Time* and *Max Desired Speed*. This is a good result because the two MoPs depend on the same parameters. It allows only one of them to be used in the objective function of the calibration procedure, simplifying its form and consequently the optimization problem. Indeed, the use of average speed in the objective function should lead to better model calibration, as it depends on a higher number of parameters than counts.

A	ANOVA 2 levels (MoP = speed per 5 minutes)			
	F-LEVEL	F-LEVEL <sub>0.95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	50	254	NOT SENSITIVE	0.64%
<i>mDs</i>	65	254	NOT SENSITIVE	0.84%
<i>nD</i>	21	254	NOT SENSITIVE	0.28%
<i>pO</i>	0	254	NOT SENSITIVE	0.00%
<i>rT</i>	7026	254	<b>HIGHLY SENSITIVE</b>	<b>90.99%</b>
<i>rTs</i>	0	254	NOT SENSITIVE	0.00%
<i>mDs</i> <i>mA</i>	0	254	NOT SENSITIVE	0.00%
<i>nD</i> <i>pO</i>	0	254	NOT SENSITIVE	0.00%
<i>nD</i> <i>rTs</i>	0	254	NOT SENSITIVE	0.00%
<i>nD</i> <i>rT</i>	5	254	NOT SENSITIVE	0.07%
<i>nD</i> <i>mDs</i>	0	254	NOT SENSITIVE	0.00%
<i>nD</i> <i>mA</i>	0	254	NOT SENSITIVE	0.00%
<i>pO</i> <i>rTs</i>	0	254	NOT SENSITIVE	0.00%
<i>pO</i> <i>rT</i>	12	254	NOT SENSITIVE	0.16%
<i>pO</i> <i>mDs</i>	1	254	NOT SENSITIVE	0.01%
<i>pO</i> <i>mA</i>	0	254	NOT SENSITIVE	0.00%
<i>rT</i> <i>mDs</i>	11	254	NOT SENSITIVE	0.15%
<i>rT</i> <i>mA</i>	32	254	NOT SENSITIVE	0.41%
<i>rTs</i> <i>rT</i>	9	254	NOT SENSITIVE	0.11%
<i>rTs</i> <i>mDs</i>	0	254	NOT SENSITIVE	0.00%
<i>rTs</i> <i>mA</i>	0	254	NOT SENSITIVE	0.00%
			<i>unexplained</i>	<b>6.04%</b>
B	ANOVA 3 levels (MoP = speed per 5 minutes)			
	F-LEVEL	F-LEVEL <sub>0.95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	1954	19	<b>HIGHLY SENSITIVE</b>	<b>9.94%</b>
<i>mDs</i>	4873	19	<b>HIGHLY SENSITIVE</b>	<b>24.80%</b>
<i>nD</i>	296	19	<b>HIGHLY SENSITIVE</b>	<b>1.51%</b>
<i>pO</i>	16	19	NOT SENSITIVE	0.08%
<i>rT</i>	5561	19	<b>HIGHLY SENSITIVE</b>	<b>28.30%</b>
<i>rTs</i>	109	19	<b>HIGHLY SENSITIVE</b>	0.56%
<i>mDs</i> <i>mA</i>	9	6	<b>HIGHLY SENSITIVE</b>	0.09%
<i>nD</i> <i>pO</i>	7	6	<b>HIGHLY SENSITIVE</b>	0.07%
<i>nD</i> <i>rTs</i>	2	6	NOT SENSITIVE	0.02%
<i>nD</i> <i>rT</i>	57	6	<b>HIGHLY SENSITIVE</b>	0.58%
<i>nD</i> <i>mDs</i>	47	6	<b>HIGHLY SENSITIVE</b>	0.48%
<i>nD</i> <i>mA</i>	33	6	<b>HIGHLY SENSITIVE</b>	0.33%
<i>pO</i> <i>rTs</i>	0	6	NOT SENSITIVE	0.00%
<i>pO</i> <i>rT</i>	17	6	<b>HIGHLY SENSITIVE</b>	0.17%
<i>pO</i> <i>mDs</i>	27	6	<b>HIGHLY SENSITIVE</b>	0.28%
<i>pO</i> <i>mA</i>	4	6	NOT SENSITIVE	0.04%
<i>rT</i> <i>mDs</i>	144	6	<b>HIGHLY SENSITIVE</b>	<b>1.46%</b>
<i>rT</i> <i>mA</i>	1436	6	<b>HIGHLY SENSITIVE</b>	<b>14.62%</b>
<i>rTs</i> <i>rT</i>	84	6	<b>HIGHLY SENSITIVE</b>	0.85%
<i>rTs</i> <i>mDs</i>	3	6	NOT SENSITIVE	0.03%
<i>rTs</i> <i>mA</i>	10	6	<b>HIGHLY SENSITIVE</b>	0.10%
			<i>unexplained</i>	<b>14.55%</b>
C	ANOVA 3 levels (MoP = traffic counts per 5 minutes)			
	F-LEVEL	F-LEVEL <sub>0.95</sub>	CONCLUSION	Allocation of variance
<i>mA</i>	28	19	<b>HIGHLY SENSITIVE</b>	0.07%
<i>mDs</i>	22463	19	<b>HIGHLY SENSITIVE</b>	<b>56.59%</b>
<i>nD</i>	4	19	NOT SENSITIVE	0.01%
<i>pO</i>	271	19	<b>HIGHLY SENSITIVE</b>	0.68%
<i>rT</i>	5663	19	<b>HIGHLY SENSITIVE</b>	<b>14.27%</b>
<i>rTs</i>	38	19	<b>HIGHLY SENSITIVE</b>	0.10%
<i>mDs</i> <i>mA</i>	10	6	<b>HIGHLY SENSITIVE</b>	0.05%
<i>nD</i> <i>pO</i>	3	6	NOT SENSITIVE	0.01%
<i>nD</i> <i>rTs</i>	0	6	NOT SENSITIVE	0.00%
<i>nD</i> <i>rT</i>	16	6	<b>HIGHLY SENSITIVE</b>	0.08%
<i>nD</i> <i>mDs</i>	21	6	<b>HIGHLY SENSITIVE</b>	0.11%
<i>nD</i> <i>mA</i>	1	6	NOT SENSITIVE	0.01%
<i>pO</i> <i>rTs</i>	0	6	NOT SENSITIVE	0.00%
<i>pO</i> <i>rT</i>	135	6	<b>HIGHLY SENSITIVE</b>	0.68%
<i>pO</i> <i>mDs</i>	221	6	<b>HIGHLY SENSITIVE</b>	<b>1.11%</b>
<i>pO</i> <i>mA</i>	1	6	NOT SENSITIVE	0.00%
<i>rT</i> <i>mDs</i>	4581	6	<b>HIGHLY SENSITIVE</b>	<b>23.08%</b>
<i>rT</i> <i>mA</i>	18	6	<b>HIGHLY SENSITIVE</b>	0.09%
<i>rTs</i> <i>rT</i>	74	6	<b>HIGHLY SENSITIVE</b>	0.37%
<i>rTs</i> <i>mDs</i>	9	6	<b>HIGHLY SENSITIVE</b>	0.04%
<i>rTs</i> <i>mA</i>	1	6	NOT SENSITIVE	0.00%
			<i>unexplained</i>	<b>2.63%</b>

Table 1 Sensitivity analysis results for the six most sensitive parameters

### Optimization algorithm setting and objective function definition

After sensitivity analysis, the optimization problem formulation is:

$$\text{Min}_a \gamma = \left[ \sum_{k \in K} \sum_{i \in I} \sum_{j \in J} (X_{ij}^{obs} - X_{ij}^{sim})^2 \right]^{0.5}$$

$$X^{sim} = F(od, rT, mDs, mA, nD/pO)$$

in which  $X$  is the MoP that will be settled. The use of an objective function with more than one MoP was not evaluated.  $od$  is the demand vector,  $rT$ ,  $mDs$ ,  $mA$ ,  $nD$  and  $pO$ , are the parameters used, differentiated for two vehicle categories. Constraints on the parameters values are the following:

$$rT_{car} = rT_{truck} \geq 0.1$$

$$mDs_{car} \geq mDs_{truck}$$

$$mA_{car} \geq mA_{truck}$$

$$nD_{car} \geq nD_{truck}$$

$$pO \leq pR$$

To solve this optimization problem, LINDO API (19) software was used. It uses a huge list of algorithms to solve several kinds of problems. Due to the nonlinear, non-convex nature of the model under investigation, a *multistart point approach* was used. This method explores stochastically the feasible solution space, searching better local optimal solutions.

### MoP sampling

In this phase the MoP (counts or speed) used in the test is sampled.

### Test with laboratory data

In the test, two vehicle categories were considered: cars and trucks. The same demand pattern as in the sensitivity analysis was used and the following parameters values have been set:  $rT_{car} = rT_{truck} = 0.7$ ,  $mDs_{car} = 120$ ,  $mDs_{truck} = 85$ ,  $mA_{car} = 3.5$ ,  $mA_{truck} = 1$ ,  $nD_{car} = 4$ ,  $nD_{truck} = 4$ ,  $pO = 90$ . A random seed was fixed and a simulation was run. Traffic counts and speed collected every 5 minutes on the 4 available virtual detectors were considered “*real data*”. Then, parameters’ values were changed and the calibration was performed.

An automatic procedure was created using LINDO API. Its communication with AIMSUN was possible through a scripting function recalled at every iterative optimization procedure step.

Starting parameters values were not found exactly using speed as MoP, nor using counts. Nor by alternating *Percent Overtake* and *Normal Deceleration* in the parameter set, was perfect reproduction of the parameters values found. This was probably due to the complexity of the response function and to the difficulties the algorithm found exploring it. However, no other algorithm was used since parameter values found using speed as MoP and Normal Deceleration in the parameter set are really close to the starting ones:  $rT_{car} = rT_{truck} = 0.63$ ,  $mDs_{car} = 113$ ,  $mDs_{truck} = 90$ ,  $mA_{car} = 3.6$ ,  $mA_{truck} = 1.1$ ,  $nD_{car} = 4.4$ ,  $nD_{truck} = 4.1$ . Besides, also the outputs of the simulation performed using these parameters were compared with the outputs of the first simulation in order to test their closeness using goodness of fit measures and confidence intervals.

Common goodness of fit measures are the root mean square error (RMSe) and Theil’s inequality coefficient ( $U$ ) [(15),(20)]:

$$RMSe = \sqrt{\frac{1}{N} \sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2} \quad U = \frac{\sqrt{\frac{1}{N} \sum_{n=1}^N (Y_n^{sim} - Y_n^{obs})^2}}{\sqrt{\frac{1}{N} \sum_{n=1}^N (Y_n^{sim})^2} + \sqrt{\frac{1}{N} \sum_{n=1}^N (Y_n^{obs})^2}}$$

in which  $Y^{obs}$  and  $Y^{sim}$  are the simulated and observed measures respectively and  $N$  is their total number.

It is also possible to use confidence intervals. Under some hypotheses [(15),(20)], it is possible to carry out a statistical test on the mean of the random variable obtained from the differences between the simulated and the observed measures  $Y^{obs} - Y^{sim}$ . A  $t$ -test can be used in the following way (1):

$$\Pr \left\{ \frac{\left( \overline{Y^{obs}} - \overline{Y^{sim}} \right) - \left( \overline{Y^{obs}} - \overline{Y^{sim}} \right)}{\sqrt{\frac{(S^{obs})^2}{n^{obs}} + \frac{(S^{sim})^2}{n^{sim}}}} < t_{\alpha/2, n} \right\} > 1 - \alpha$$

where  $\underline{Y}^{obs} - \underline{Y}^{sim}$  is the difference between the observed and the simulated means,  $s^{obs}$  and  $s^{sim}$  are the observed and simulated standard deviations,  $n^{obs}$  and  $n^{sim}$  are the observed and simulated degrees of freedom,  $(1-\alpha)$  is the confidence interval used in the test and  $t_{\alpha/2, n}$  is the value of a Student probability distribution with  $n$  degrees of freedom corresponding to a probability queue equal to  $\alpha/2$ . Setting  $\underline{Y}^{obs} - \underline{Y}^{sim} = 0$ , assuming  $\alpha = 0.05$  and calculating average measurements on a 15-minutes moving window (in which there are 3 collections),  $t_{\alpha/2, n}$  assumes value of 4.303. From here the band of admissibility around the observed measures was calculated to graphically show the test results on the whole time series. In Figure 4 and Table 2 calibration's results using both speed and counts are shown.

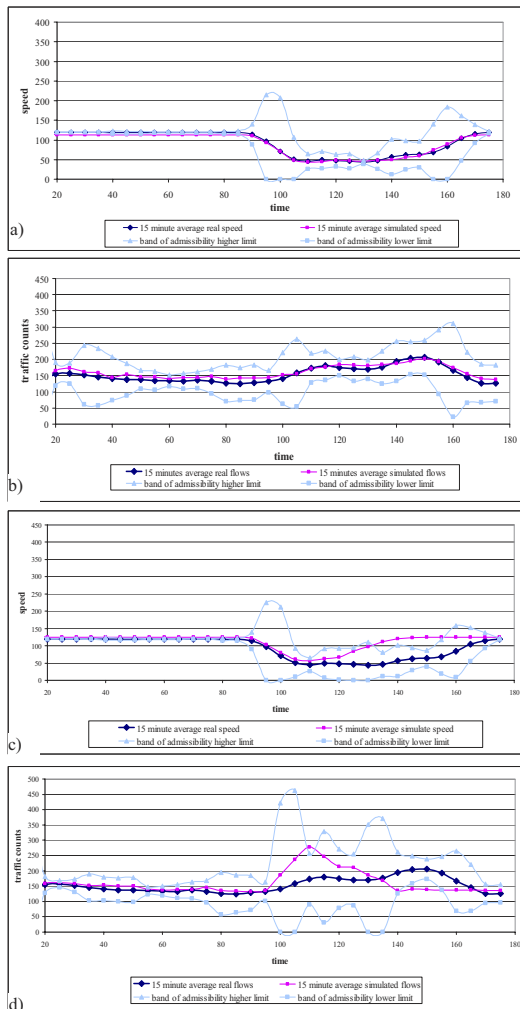


Figure 4 Results of the test of calibration procedures (MoP used: speed (a,b) and flow (c,d))

idDet	Calibration with speed				Calibration with flows			
	RMSe		Theil's coefficient		RMSe		Theil's coefficient	
	speed	flows	speed	flows	speed	flows	speed	flows
151	21.09322641	10.07127378	0.120192	0.0401331	31.122041	28.357588	0.176232235	0.108044946
737	8.627777283	11.658926	0.061239	0.056412	25.27974	30.79908	0.162883795	0.147506283
743	5.458896546	10.06023525	0.0303204	0.0755704	12.384591	37.224171	0.072748026	0.286141478
744	4.926664217	12.31135429	0.0326094	0.1472187	35.286715	19.466495	0.225395325	0.230410033

Table 2 Comparison of test results

In each figure, the evolution of simulated and collected data over the three hours of simulation is presented. The azure lines represent the confidence intervals calculated at each time point. It may be noted that in both experiments, simulated data lie within the band of sensitivity, but in the case of calibration carried out using speeds, simulated measures are nearer the collected ones than if traffic counts are used as an MoP. In the table, this phenomenon is highlighted in clearer fashion. The goodness of fit measures are shown for each detector available on the network. Both RMSE and U referring to speed calibration are always less than those obtained using flows and show a real goodness of fit (note that  $U=0$  implies perfect fit between the observed and simulated measurements). The calibration procedure set-up ends with the following main results:

- the implemented procedure is able to calibrate the model using laboratory data;
- the MoP to use in the objective function is *speed*. Results obtained using flows confirm the sensitivity analysis' results and don't suggest to use a combined objective function;
- the algorithm used found a fair parameters set, although it was not able to find the absolute minimum solution;
- the set of parameters to calibrate was established: results obtained using Normal Deceleration are better than those obtained using Percent Overtake even if it is necessary to calibrate one more parameter.

### **Model calibration on a real scenario**

#### *Data collection and validation*

On the Freeway E32, data are continually collected and differentiated by 4 vehicle categories and by lane every minute using a system composed by synchronized camcorders. The validation of data collected during an entire week revealed the existence of an incongruence in the internal collection system: traffic counts at entries are constantly higher than those at exits. An attempt was made to correct data to preserve their consistency. Nevertheless, a non-negligible error has to be taken into account in examining the calibration results.

#### *Calibration of the simulation model*

Despite the data inconsistency, two calibration attempts were made. Besides the test site described at the beginning of the paper, the calibration procedure was also applied (Figure 5) on another subsystem of the same Freeway to test its robustness, namely the 9km stretch between the toll barriers of Portici and Torre Annunziata in the Salerno direction.

All the points lie within the respective confidence interval, but the difference between simulated and real measures is not negligible. Data related errors render any calibration attempt void. However, in spite of the lack of accuracy in data collection, the results confirm the effectiveness of the procedure developed. Further applications on validated data are necessary to better confirm the effectiveness of the proposed procedure. The level of accuracy in data collection should also be investigated.

#### *Validation of the simulation model*

Because of the impossibility in recovering accurate data, this phase was not performed at all.

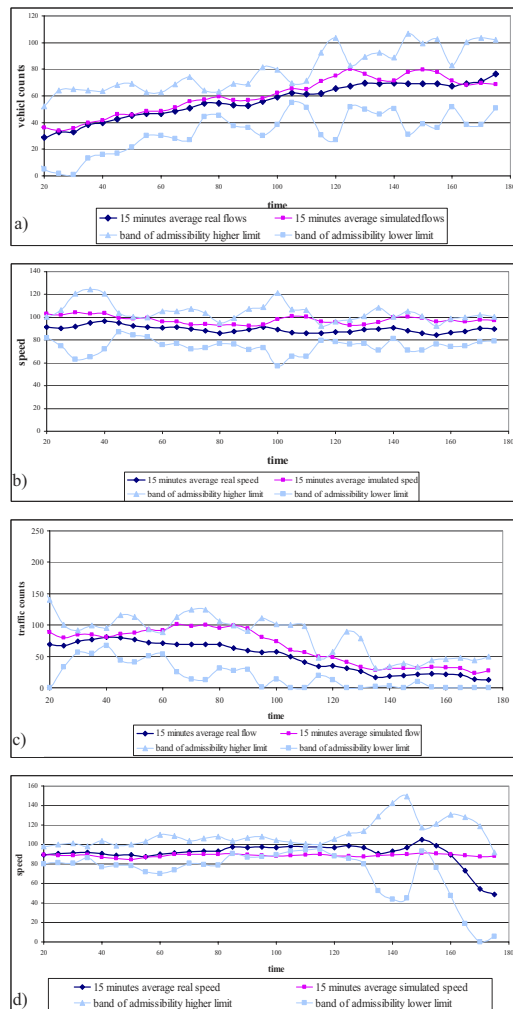


Figure 5 Results of the calibration procedure on the Portici (a,b) and Angri (c,d) stretches

**SUMMARY**

This paper describes preliminary results and efforts of implementing an automatic calibration procedure for microscopic simulation model. The microscopic simulation tool used has been AIMSUN.

Dimensions of the problem strongly suggested a preliminary model sensitivity analysis to reduce the number of parameters to be investigated. Other hurdles of the problem are the choice of the algorithm to use in the optimization problem and the Measure of Performance to consider in its formulation. Both of them influences the effectiveness of the procedure. These problems were overcome using an iterative structure with a calibration test on laboratory data in its centre. The aim of the test is (a) to verify the ability of the optimisation algorithm in finding the minimum for such a complex response function, (b) to individuate the MoP more suitable in the optimization problem and (c) to confirm results of the sensitivity analysis. The test succeeded as the algorithm was able to find value really proximal to the “true” ones used to generate the traffic data used for model calibration. In particular, results showed the better performance of using speed as MoPs. Hence, at the end of this phase the analyst knows the “optimal” parameter set, the MoP and the algorithm to use in the calibration problem. The application on a real site is then possible. Great care was devoted in verifying the consistence of the data to use. The impossibility in finding accurate data, made void each attempt of calibrating the model. However, also using noisy data the calibration procedure showed its effectiveness. At this stage of the work validations across

the different sites and over different time periods have not yet performed due to the lack of accuracy of the available data. When they will be available, will allow arguing something more on the validity of the calibration procedure (and of the simulation model used) and on the significance of values of parameters obtained in calibration.

## ACKNOWLEDGEMENTS

The work has been funded by the project n. 12811, Legge 593/2000 Art.12 PON 2000 – 2006; “Sistema robotizzato di spegnimento incendi e monitoraggio ambientale permanente per gallerie stradali e ferroviarie”. Instead traffic data have been available under the project n. 12897 Legge 593/2000 Art.12 PON 2000 – 2006; TEMA 15 – Trasporti: “Sistema di monitoraggio, controllo ed informazione per la gestione attiva della sicurezza di un’infrastruttura autostradale”.

## REFERENCES

- (1) Law A. M., W. D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, International Edition, 2000.
- (2) Rakha H., Hellinga B., Van Aerde M., W. Perez. Systematic verification, validation and calibration of traffic simulation models. *Transportation Research Record*, 1997.
- (3) Punzo V., Formisano D. J., V. Torrieri. Non-stationary Kalman filter for the estimation of accurate and consistent car-following data. *Transportation Research Records* 1934, 2005.
- (4) Ben Akiva M., Darda D., Jha M., Koutsopoulos H., T. Toledo, Calibration of Microscopic Traffic Simulation Models with Aggregate Data. *Transportation Research Records* 1876, 2004.
- (5) Abdulhai B., Sheu J.B. and W. Recker. Simulation of ITS on the Irvine FOT Area Using ‘Paramics 1.5’ Scalable Microscopic Traffic Simulator: Phase I: Model Calibration and Validation. *California PATH Research Report*, 1999
- (6) Hourdakis J., Michalopoulos P. G., J Kottommannil, A practical procedure for calibrating microscopic traffic simulation models. *Transportation Research Records* 1852, 2006.
- (7) Dowling R., Skabardonis A., Halkias J., Mc Hale G., G. Zammit. Guidelines for Calibration of Microsimulation Models: Framework and Applications. *Transportation Research Records* 1876, 2004.
- (8) Park B., J. D. Schneeberger. Microscopic Simulation Model Calibration and Validation: A Case Study of VISSIM for a Coordinated Actuated Signal System. *Transportation Research Records* 1856, 2003.
- (9) Jha M., Goplan G., Garms A., Mahanti B. P., Toledo T., M. Ben Akiva. Development and calibration of a large – scale microscopic traffic simulation model. *Transportation Research Records* 1876, 2004.
- (10) Toledo T., Koutsopoulos H.N., Davol A., Ben-Akiva M.E., Burghout W., Andréasson I., Johansson T. and C. Lundin. Calibration and Validation of Microscopic Traffic Simulation Tools: Stockholm Case Study. *Transportation Research Records* 1831, 2003.
- (11) Park B., H. Qi. Development and Evaluation of a Procedure for the Calibration of Simulation Models. *Transportation Research Record* 1934. TRB, Washington, 2005.
- (12) Jayakrishnan R., Oh J.-S., A.-E.-K. Sahroui. Calibration and Path Dynamics Issues in Microscopic Simulator for Advanced Traffic Management and Information Systems. *Transportation Research Record* 1771, 2001.
- (13) Neter J., Wasserman W., M.H. Kutner. *Applied linear statistical models: regression, analysis of variance, and experimental designs*. Richard Irwin, Homewood. 1990
- (14) Punzo V., F. Simonelli. Analysis and Comparison of Car-following Models Using Real Traffic Microscopic Data. *Transportation Research Records* 1934, 2005.
- (15) Toledo T., H. N. Koutsopoulos, Statistical validation of traffic simulation models. *Transportation Research Records* 1876, 2004.
- (16) AIMSUN 5.0 Microsimulator User’s Manual 5.0.6. Transport Simulation Systems 1997-2005. November 2005.
- (17) BOSS/QUATTRO 5.0 manual. Samcef, 2004.
- (18) AIMSUN NG Scripting Version 0.7. November 2005.
- (19) LINDO API manual, Lindo Systems inc.1804.
- (20) Pindyck, R.S., Rubinfeld, D.L., *Econometric Models and Economic Forecasts*, fourth ed. McGraw-Hill, 1998.



1 **A comparison between simulation-based and model-based calibrations of traffic flow**  
2 **micro-simulation models**  
3  
4  
5

6 **Biagio Ciuffo**

7 PhD Student  
8 Department of Transportation Engineering  
9 Università di Napoli “Federico II”  
10 Via Claudio, 21 – 80125 Napoli – ITALY  
11 Ph.: +39.081.7683770  
12 Fax: +39.081.7683740  
13 E-mail address: [bciuffo@unina.it](mailto:bciuffo@unina.it)

14 **Vincenzo Punzo\***

15 PhD, Assistant Professor  
16 Department of Transportation Engineering  
17 Università di Napoli “Federico II”  
18 Via Claudio, 21 – 80125 Napoli – ITALY  
19 Ph.: +39.081.7683948  
20 Fax: +39.081.7683946  
21 E-mail address: [vinpunzo@unina.it](mailto:vinpunzo@unina.it)

22 **Vincenzo Torrieri**

23 Full Professor  
24 Department of Transportation Engineering  
25 Università di Napoli “Federico II”  
26 Via Claudio, 21 – 80125 Napoli – ITALY  
27 Ph.: +39.081.7683368  
28 Fax: +39.081.7683946  
29 E-mail address: [torrieri@unina.it](mailto:torrieri@unina.it)

30 Words: 6092  
31 Plus 2 Tables: 500  
32 Plus 4 Figures: 1000  
33 TOTAL: **7592**

34 \* Corresponding author  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52

**ABSTRACT**

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52

Parameter calibration of traffic micro-simulation models usually takes the form of a simulation-based optimization problem, that is, an optimization in which every objective function evaluation calls for a simulation. It is recognized that such a problem is computationally intractable. Running time grows exponentially both in the number of parameters and in the digits accuracy. In addition, considerable computing time is required by each objective function evaluation. This means that only heuristic techniques can be applied.

Accordingly in this paper results of the application of OptQuest/Multistart algorithm (OQMS) to the calibration of AIMSUN micro-simulation model parameters on a freeway network are first presented.

Furthermore, it is claimed that the search for an effective solution to the calibration problem cannot be exhausted by the choice of the most efficient optimization algorithm. Indeed, the use of available information concerning the phenomenon could allow calibration performance to be enhanced e.g. by reducing dimensions of the domain of feasible solutions. In the remainder of the paper it is argued that this goal could be achieved by using information from the stationary counterpart of microscopic traffic flow models that depict the aggregate variables of traffic flows as a function of drivers' microscopic parameters. Indeed, as they have a closed analytical formulation they are well suited for faster calibrations. Results show that values of parameters from stationary model-based calibrations are not really far from the optimal ones. Thus the integration of the two approaches cannot be excluded but it is worth investigating.

## INTRODUCTION

In most engineering fields the simulation of complex stochastic systems has been boosted by technological improvements in computing and power memory size. In the field of transportation engineering as well, the micro-simulation of traffic flows is growing in importance. The target sector of applications of such techniques has shifted towards the analysis of complex networks, compared to early experiences which were restricted to contexts whose dimensions and complexity were usually small, such as isolated intersections, toll plazas, etc. The wide use of such techniques, for example in assessing or forecasting interventions on road networks or ITS strategies, calls for careful inspection of their ability to faithfully represent the systems under investigation. The identification of model parameters or calibration is therefore a crucial issue. Such a problem generally takes the form of an optimization problem in which values are sought for a set of parameters which minimize the distance measurements between system and model outputs.

Since the micro-simulation model is not analytical, we are dealing with an objective function  $f$  in the form of a black-box model: given a vector of parameters  $\beta$  in the feasible domain the simulation model yields  $f(\beta)$ . Accordingly, such a problem is generally referred to as black-box or simulation-based optimization and it is a (constrained) non-linear, non-convex optimization problem. Its computational complexity i.e. the amount of resources (typically running time) required for the solution is known to be exponential. Indeed, Vavasis (*1*) proves lower bound complexity results for both global and local optimization. In the first case the problem turns out to be intractable since running time grows exponentially both in the number of parameters and in the digits accuracy. "In contrast to global minimization, the problem of finding an approximate local minimum, under suitable definitions, has complexity that is polynomial" in the number of parameters (and exponential in the digits accuracy) (*1*). It is worth remembering that complexity results depend on the problem and on the digits accuracy, but not on the algorithm (see e.g. (*2*)).

Moreover, in traffic flow micro-simulation each single evaluation of the objective function is very costly, depending on the size of the network and on the number of vehicles simulated. Computational complexity, in addition to the cost of single information regarding the objective function, has therefore steered calibration studies towards the use of heuristic optimization techniques. Among these, experiences carried out using Genetic Algorithms (*3-5*), Box's Complex Algorithm (*6*) and also Trial and Error techniques (*7*) are known. These approaches lead to a rather broad exploration of the feasible domain and consequently they have a probability of being trapped in a local minimum lower than gradient-based algorithms. Moreover, such exploration makes them time-consuming, whereas nothing is known about the quality of the solution found. Hence, other approaches have been tested. In particular, use has recently been made of another heuristic optimization method (*8, 9*): the Simultaneous Perturbation Stochastic Approximation algorithm. This method computes an approximation of the gradient function after perturbing all the parameter values simultaneously. In this way it uses information on the descending direction to obtain a new parameter set. SPSA preserves some positive features of Genetic Algorithms with the least computational effort.

An alternative approach to the problem resolution can be to adopt a heuristic technique which combines a global search of the minimum with a local one. Such an approach is in accordance with findings on complexity by (*1*) and allows the complementary qualities of population-based and gradient-based heuristics to be exploited. For example, the OptQuest/Multistart algorithm (OQMS) (*10*) employs a Scatter Search meta-heuristic (*11*) to provide starting points for a Generalized Reduced Gradient NLP solver (*12, 13*). Such an algorithm is also suitable for mixed-integer optimization problems which arise when simulating different classes of vehicles in the same scenario (see the Methodology section). Moreover, the Scatter Search can be repeated  $m$  times to increase the probability that the best local minimum sought is the global one. Accordingly, the OQMS has been used in this work both for its capability to manage discrete decision variables and because it returns a landscape of the objective function, by enumerating all local minima found.

In this paper it is also claimed that due to its theoretical and practical complexity, the search for an effective solution to the calibration problem cannot be exhausted by the choice of the most efficient algorithm. Both the computational time and the quality of the solution can be improved by means of proper techniques and by using all the available information concerning the phenomenon.

First of all the need to reduce the calibration process only to the sensitive parameters, or even to the most sensitive ones (naturally the latter choice can affect the solution quality as well) is stressed. Indeed, the issue of how many and which input parameters have to be calibrated in a micro-simulation model should be neither disregarded nor left to intuitive considerations.

Indeed, for example, either recognising that traffic dynamics are strongly influenced by traffic composition (especially by the percentage of heavy vehicles) or being simply interested in representing the different vehicle classes in the simulation, each behaving differently from the others (see e.g. (14)), it may be rather hard to think of adopting a single vehicle class in the simulation phase. For instance, it has been recently pointed out that it is impossible to suitably represent traffic flow in micro-simulations using a number of vehicle classes different from that observed in the real context (15). In the same work the calibration of a double-class stationary traffic flow model, derived from a car-following model, produced inaccurate results, compared to the case in which four vehicle classes were considered. Indeed, it was verified that among the same type of vehicles, e.g. cars, driving behaviour sharply changed moving from a lane to another, suggesting that two different types of cars (and trucks) be considered in the simulation. Clearly, therefore, the number of parameters to calibrate can rapidly grow to unacceptable values (for instance, only focusing on the car-following model, in (15) it meant doubling the parameters from 12 to 24).

Several authors have raised the issue of model sensitivity analysis as a preliminary step to the calibration process (e.g. 3, 5, 7, 16, 17). The most widely applied technique is Analysis of Variance (ANOVA), (5, 16, 17). A key point of the analysis is the design of experiments. (5) performs a 5-level design on 8 parameters. However, since a full factorial plan would imply  $5^8$  (i.e. 390,625) simulations, they adopt Latin Hypercube Sampling which reduces the number of simulations to 200. By contrast, (16) inspects a full factorial plan with only two levels. Finally in (17) two full factorial plans, respectively with 2 and 3 levels, are designed. It is shown that a two-level design is not adequate to capture the influence of model parameters. In the following sections we will rely on results from (17).

In addition to the reduction in parameter number, a practicable approach, usually adopted in engineering applications of optimization, is to reduce the dimensions of the domain of feasible solutions. In microscopic traffic simulation the aim is to decrease the range of the intervals in which parameters can vary (i.e. the box constraints). This approach requires *a priori* knowledge of the phenomenon and of the behaviour of the model which one intends to use. For example, the objective can be achieved when the stationary traffic flow model, corresponding to the microscopic one used in the simulations, is known (see e. g. (15) which provides the steady state solutions of the Gipps' car-following model). Such stationary models depict the aggregate variables of traffic flows as a function of microscopic driving parameters. In contrast to the corresponding microscopic models they present a closed analytical formulation and are then very suitable for swift calibration. Therefore it is claimed here that they could be thought of as the *a priori* knowledge on the phenomenon we are looking for and then that they could be used to enhance performances of a black-box calibration. Indeed, if it were shown that values of parameters calibrated in such a way were near to the global optimal ones then it would be possible to use these values both as starting values for the black-box calibration and for reducing the amplitude of the box constraints around them when setting the black-box calibration.

Interestingly, the effectiveness of a calibration process as well as the preceding considerations can be evaluated at best only by performing "controlled" experiments, in which all the model inputs are known. This would lead to the execution of tests based on laboratory data. A practical procedure aimed at verifying all the steps involved in a calibration process that makes use of tests with laboratory data has recently been proposed in (17). Such methodology will be followed throughout this paper.

The developed work was led by calibrating the microscopic parameters of traffic flow in AIMSUN, although the methodological background is valid regardless of the model taken into account. Then the remainder of the present paper is organized as follows. The study methodology is shown, particularly accounting for the adopted solution approach and for the significance of tests with laboratory data. A case study is then presented and results of calibrations carried out following both the black-box approach and the stationary one are detailed and commented. The comparison between results from the two approaches will be the main contribution of the paper and it is intended as a prerequisite before any attempt to integrate the approaches. Thus the paper ends with a summary of findings and prospects for future research, especially regarding the issue of integration.

## METHODOLOGY

The problem of parameter calibration of traffic flow micro-simulation models may be formulated in the following optimization framework:

$$\underset{\beta, \gamma}{\text{Minimize}} \quad f(\mathbf{M}^{obs}, \mathbf{M}^{sim}) \quad (1)$$

subject to the following constraints:

$$l_{\beta} \leq \beta \leq u_{\beta}$$

$$l_{\gamma} \leq \gamma \leq u_{\gamma}$$

where  $\beta$  and  $\gamma$  are, respectively, the vectors of continuous and discrete decision variables (i.e. the model parameters to calibrate);  $f(\cdot)$  is the objective function or goodness-of-fit function, which measures the distance between the observed and simulated sensor measurements,  $M^{obs}$  and  $M^{sim}$ ;  $l_{\beta}$ ,  $l_{\gamma}$ ,  $u_{\beta}$ ,  $u_{\gamma}$ , are lower and upper bounds on model parameters and  $M^{sim} = S(x_1, \dots, x_H, \beta, \gamma)$ , in which  $S(\cdot)$  is the micro-simulation model and  $x_h$  the vectors of OD flows departing in interval  $h$ .

It is worth noting that discrete parameters appear when considering several vehicle classes. Indeed, to simplify calculations, the integration step of the delay-differential equations of car-following models is generally assumed equal to the value of delay, namely to the reaction time. Considering different classes of vehicles, each with its peculiar reaction time, means choosing values of these reaction times that are integer multiples of the common simulation step (generally equal to 0.1 seconds). On the other hand, the calibration of reaction times is a crucial issue, since this parameter has a strong influence on road capacity (see e.g. (15)).

Provided that  $f$  is an input to the minimization algorithm, a part of this input is a black-box model, to be precise the simulation model  $S$ . Since  $S$  is a complex, stochastic, non-linear simulation model, the optimization problem by (1) turns out to be a constrained non-linear, non-convex, mixed-integer optimization problem. As anticipated in the introduction, the global minimum search is really complex and may be achieved only by means of heuristic techniques. In the next sub-section the algorithm employed in this study is briefly presented.

### Solution Algorithm

The algorithm used herein is the OQMS heuristic (OptQuest/Multistart). A detailed description of the algorithm may be found in (10). The algorithm employs a Scatter Search meta-heuristic (11) to provide starting points for a Generalized Reduced Gradient NLP solver (12, 13). In this way it tends to combine the superior accuracy and feasibility seeking behaviour of gradient based local NLP solvers with the global optimization abilities of a Scatter Search. In particular, for smooth problems, the relative advantages of a search method like Scatter Search over a gradient-based NLP solver are its ability to locate an approximation to a good local solution (often the global optimum), and to handle discrete variables. Gradient-based NLP solvers converge to the “nearest” local solution, and do not have facilities for discrete variables unless they are embedded in a rounding heuristics or branch-and-bound method. A relative disadvantage of scatter search methods is their limited accuracy. Search methods also require an excessive number of iterations to find approximations to local or global optima accurate to more than 2 or 3 significant figures, while gradient-based solvers usually rapidly achieve 4 to 8-digit accuracy (10). The selection of starting points is made more efficient by means of a Random Linkage (RL) multi-start algorithm (18) whose function is avoiding choosing starting points converging to the same local minimum.

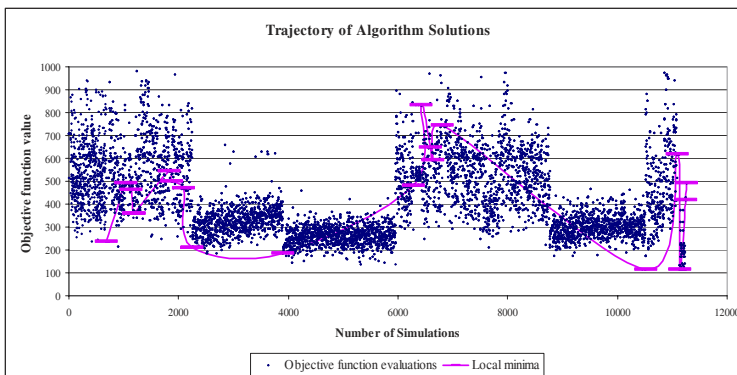


FIGURE 1. Objective function evaluations and trajectory of local minima

The algorithm functioning is evident in Figure 1 which shows the values of the objective function related to each algorithm iteration, in the case of an application later described in the Case Study section. In Figure 1 scatter plots differentiating the  $m$  iteration of the Scatter Search are well distinct (in such a case the stopping rule arrested the algorithm at the sixth search of 10 planned). Given the algorithm's peculiarity, the decrease in the objective function values is not identifiable. Instead in the figure the trajectory of the local minima found is highlighted. The knowledge of such minima in general may suggest the quality of the solution provided and in all the conducted experiments the algorithm proved good capabilities in approximating the global minimum of the function, conversely generating a huge amount of iterations.

### Test with laboratory data

The quality of the solution found is the second major issue of the problem (beyond running time). It refers to the following unanswered questions: assuming that none of the solution techniques assure that the global optimum is reached, to what extent may the provided solution be trusted? And eventually, how far does the sub-optimal solution differ from the optimal one, affecting successive simulations?

A well known technique used for verifying the ability of the algorithm to provide an accurate solution and showing good insight into the phenomenon consists in performing tests with laboratory data (also referred to as synthetic data). In the context of calibrating traffic micro-simulation models it has been recently applied in (17). It consists in calibrating the model using synthetic measurements. Once "realistic" inputs to the simulation model have been assumed (i.e. time varying O/D matrices, model parameter values, etc.), the model is run and outputs from the one-shot simulation become the synthetic measurements. Then the calibration process aims at finding parameter values that minimize the distance between the model outputs and the synthetic data. The calibration is successful when values of calibrated parameters equal the values used for generating the measurements.

Since the optimisation problem is intractable, it is never possible to guarantee finding the global optimum i.e. the values of parameters used for generating the measurements. However, unlike calibrations performed on actual measurements of the system, in the case of calibrations on laboratory data, optimal values of parameters are known and a comparison can be performed with the presumably sub-optimal values provided by the calibration.

Therefore, assuming that the system model and the inputs used in the generation of the laboratory data are "realistic", this technique allows possible flaws to be identified in the calibration process as well, as detailed in (17). For example, being able to draw the whole objective function it is possible to get an insight into the phenomenon, hence, to compare calibration results obtained using different measures  $M^{obs}$  in equation (1) (i.e. different measures of performances – *MoPs*). For instance, in Figure 2, the landscape of the objective function is shown for two different measures: time series of counts and speeds.

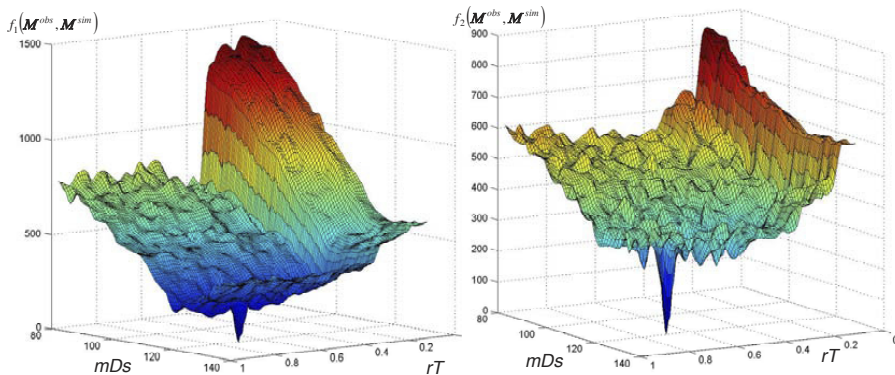


FIGURE 2. Response surfaces of the model as functions of drivers' reaction time and desired speed (see AIMSUN car-following model (19))

### CASE STUDY: A FREEWAY SCENARIO

A freeway scenario was chosen as a case study: the absence of route choices and reliability of the estimated OD flows reduced the problem complexity. Actually, all the analyses were carried out relying on laboratory data, in order to eliminate modelling errors from the investigations and to keep exact knowledge of all the inputs to the model. However, in order to perform “realistic” tests, careful attention was paid in accurately reproducing the network and the main inputs to the model. For example, minute by minute inflows to the freeway distinguished by vehicle class were all known and used as input demand to the model. They were previously processed in order to preserve the consistency between inflows and outflows over a wider time interval.

Experiments consisted in the calibration of AIMSUN 5.1.8 (19) on the highly congested Italian freeway E45, Naples-Pompei-Salerno. The mean daily traffic volume is about 60,000 vehicles with an average percentage of heavy vehicles of 15%.

Especially in the morning peak hours, drivers experience congested traffic conditions in the Naples direction. Detector data show that at the Portici entry-ramp a bottleneck becomes active at 8.00 a.m. and lasts for about 1 hour, spilling back for several kilometres. For this reason the test-site considered was an 11 km freeway stretch (see Figure 3) going from the Portici entry-ramp upstream to the Variable Message Sign located at Pinetina.

The second phase of the calibration procedure was model sensitivity analysis. In spite of the general reliability of results previously obtained for a freeway scenario (17), in order to slavishly follow the proposed methodology, a new sensitivity analysis of the model on the network was performed. In this phase the vehicle fleet was considered as composed of an only one vehicle type. Measures of Performance used as vehicle outputs for testing the parameter influence were both traffic counts and speed surveyed every five minutes on the detectors {8561, 8562}. Parameters whose influence on the model outputs was investigated are the AIMSUN vehicle modelling parameters: *global parameters* and *vehicle attributes* (19). Obviously path-choice parameters were not considered in the analysis.

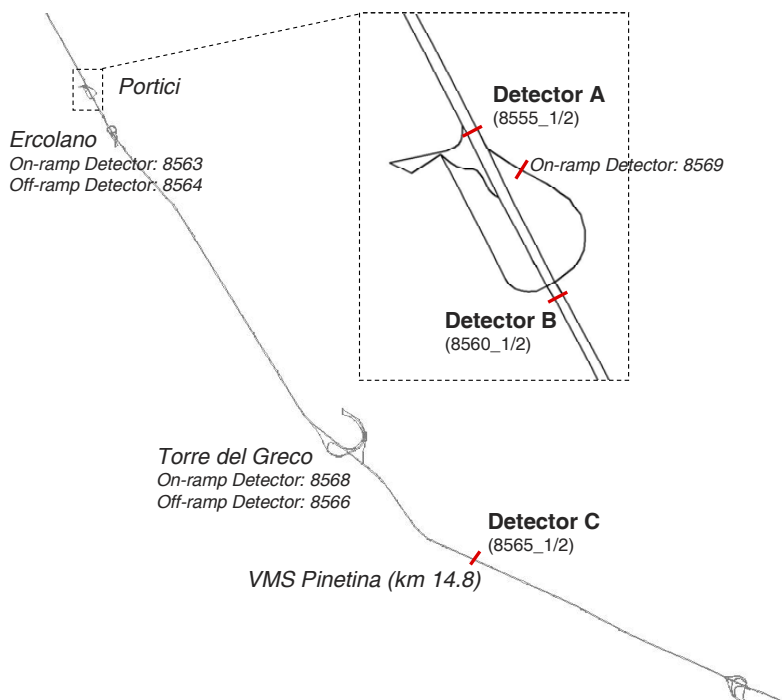


FIGURE 3. Test site layout – E45 Naples-Pompei-Salerno Freeway

### Global Parameters

It is a set of parameters related to the vehicle behaviour models that is valid throughout the whole network: the parameters are used for all vehicles driving anywhere in the network during the entire simulation (19). The main global parameters are the following (see (19) for further details):

- Driver's Reaction Time ( $rT$ )
- Driver's Reaction time at Stop ( $rTs$ )
- Queuing Up Speed ( $qUs$ )
- Queuing Leaving Speed ( $qLs$ )
- Percent Overtake ( $pO$ )
- Percent Recover ( $pR$ )

According with results by (17) only  $rT$ ,  $rTs$ ,  $qUs$  and  $pO$  were involved in the successive analysis.

### Vehicle Attributes

These parameters are defined per vehicle type. For each of them a truncated Normal probability density function can be defined by setting its mean, its deviation and its bounds. Vehicle attributes are the following:

- Maximum Desired Speed ( $mDs$ )
- Maximum acceleration ( $mA$ )
- Normal Deceleration ( $nD$ )
- Maximum Deceleration ( $mD$ )
- Speed Acceptance ( $sA$ )
- Minimum distance between vehicles
- Maximum give-way time ( $gWt$ )

Maximum Deceleration was discarded because in previous analyses it had never had any influence; minimum distance between vehicles was discarded since it was derived from field observations. Particular attention was instead devoted to setting the free-flow speed of drivers along the freeway. In AIMSUN, this speed, i.e. the desired speed of the Gipps' car-following model, is the minimum between the Maximum Desired Speed parameter (constant over the whole network for a given vehicle/driver) and the product of the Speed Limit on that section by the Speed Acceptance parameter (and has great influence on model behaviour as shown below).

Then, to allow a driver to attain a variable desired speed among different sections (and not a unique value over the network), a speed profile along the road was evaluated as a function of road geometric features like e.g. curvature and section width; see (20) for the methodology. The Speed Limit parameter of each section was thus set according to such a speed profile. To avoid the global parameter Maximum Desired Speed having any influence on the desired speed (refer to the calculation in AIMSUN as mentioned above), it was set at an unrealistically high value. Thus the desired speed of each vehicle on a section becomes the product of the Speed Limit by Speed acceptance. The Speed Limit is a constant over the vehicles (albeit variable among sections and consistent with the road alignment) while the Speed Acceptance accounts for the variability over the different vehicle classes. Since the Speed Limit is fixed once calculated, the only parameter influencing free-flow speed involved in subsequent analyses was Speed Acceptance,  $sA$ . The parameters involved in the analysis were therefore  $mA$ ,  $nD$ ,  $sA$  and  $gWt$ .

Analysis was carried out as described in (17): first a 3-level full factorial experimental plane was constructed with the parameters chosen for the analysis, the simulations corresponding to each parameter value combination were then performed (a total of  $3^8 = 6561$  simulations) and the outputs were retrieved. Analysis of variance (ANOVA) was subsequently performed on these outputs in order to obtain the influence on the model of each parameter and of their first order interactions.

Finally, to confirm the results obtained, two other ANOVAs were carried out on the outputs of other experimental planes: a 4-level full factorial design on the five parameters proved to affect the measured speed ( $5^4 = 625$  simulations) and a 4-level full factorial design on the 6 parameters affected traffic counts measured on the network ( $5^4 = 1296$  simulations).

The results obtained substantially confirm previous research (17). However, analysis performed with 4-level experimental plans showed a reduction in the variance explained by the Maximum Acceleration on the measured speeds.



Actually, although the model proved sensitive to 5/6 parameters (depending on the output chosen) more than 95% of the output variance was explained by only two parameters, Driver's Reaction Time and Speed Acceptance (i.e. our proxy for the Maximum Desired Speed) and their interaction. Hence the number of parameters taken into consideration during the calibration phase was limited to two per vehicle type. As stated above, allowing the Reaction Time to vary among vehicle types turns the problem into a Mixed Integer Problem (MIP) (values have to be integer multiples of 0.1 seconds).

An important issue in setting up the calibration procedure is the choice of the mathematical structure of the objective function  $f$  to use in the optimization problem. Several Measures of Goodness of Fit ( $MGF$ ) are commonly used with this purpose. Its choice has great significance due to its influence on the objective function shape and thus on the optimization algorithm effectiveness. For this reason different attempts were performed in order to investigate the choice of the most suitable  $MGF$ . In particular, the Root Mean Square Error (RMSE) and the Root Mean Square Percentage Error (RMSPE) were used. Their definition is reported below:

$$RMSE(MoP) = \sqrt{\frac{1}{(T \cdot K \cdot D)} \sum_{t=1}^T \sum_{k=1}^K \sum_{d=1}^D (MoP_{tkd}^{obs} - MoP_{tkd}^{sim})^2} \quad (2)$$

$$RMSPE(MoPs) = \sqrt{\frac{1}{(T \cdot K \cdot D \cdot M)} \sum_{t=1}^T \sum_{k=1}^K \sum_{d=1}^D \sum_{m=1}^M \left( \frac{MoP_{tkdm}^{obs} - MoP_{tkdm}^{sim}}{MoP_{tkdm}^{obs}} \right)^2} \quad (3)$$

in which  $T$ ,  $K$ ,  $D$  and  $M$  are, respectively, the number of measurement intervals, the number of vehicle classes, the number of detectors and the number of  $MoPs$  in the  $MGF$ .  $MoP_{tkdm}^{obs}$  and  $MoP_{tkdm}^{sim}$  represent respectively the observed and simulated values of the Measure of Performance  $m$ . Note that, given the nature of equation (3), different  $MoPs$  can be combined without resorting to homogenization coefficients.

### Collecting Laboratory Data

Following the findings made by (15) on the same data sets, it was decided to consider four vehicle types: two types of cars and two types of trucks. Input demand data to the model were set up accordingly.

In truth, available traffic data do not obviously take such a classification into account, but only distinguish between cars and trucks. The problem was overcome by assuming that "slow" drivers usually stay in the rightmost lane, while more "aggressive" drivers use the leftmost one. The proportion of each class was then retrieved by traffic data available on the different lanes (at detectors 8555 and 8565). The input demand was then split according to the proportions derived from the data.

After setting the "true" values for the parameters of each vehicle type (see Table 1) a simulation was run. The simulation reproduces traffic conditions during the peak morning period (6.00-9.30). Traffic data on virtual detectors (8561 and 8562) were collected and considered the model output measurements to use for subsequent calibrations (i.e.  $M^{obs}$  in equation (1)). Data collected were time series of counts and speeds over 5-minute intervals (see also (17) for the choice of the interval length).

Interestingly, despite the ability of the virtual sensors to distinguish among the different vehicle types (e.g. between "fast" and "slow" cars), for subsequent calibrations the data collected were aggregated to form only 2 classes: cars and trucks. More specifically, car counts on each lane were obtained as the sum of fast and slow car counts on that lane and car speeds as the space mean speeds of fast and slow car speeds (the same holds for trucks). This was assumed to maintain consistency with traffic data available from the real detectors located on the freeway.

### Objective functions

Three calibrations were performed using the vector of observed "laboratory" measurements: two simulation-based calibrations and one (stationary) model-based calibration. The two simulation-based calibrations, in the following referred to as  $F_1$  and  $F_2$ , differ in the objective function.

In  $F_1$ , the objective function has the form of equation (3) and includes both traffic counts and speeds at detectors as  $MoPs$  ( $q_{tkd}^{obs}$  and  $v_{tkd}^{obs}$ , respectively). It relies upon the time series of measurements over 42 time intervals, two vehicle classes and two detectors (one per lane). In this case equation (3) is specified as:

$$RSSPE(q, v) = \sqrt{\frac{1}{(42 \cdot 2 \cdot 2)} \sum_{t=1}^{42} \sum_{k=1}^2 \sum_{d=1}^2 \left( \frac{q_{tkd}^{obs} - q_{tkd}^{sim}}{q_{tkd}^{obs}} \right)^2 + \frac{1}{(42 \cdot 2 \cdot 2)} \sum_{t=1}^{42} \sum_{k=1}^2 \sum_{d=1}^2 \left( \frac{v_{tkd}^{obs} - v_{tkd}^{sim}}{v_{tkd}^{obs}} \right)^2}$$

The second simulation-based calibration, namely  $F_2$ , and the model-based calibration  $F_3$ , adopt an objective function in the form of equation (2), in which the speed, per time interval, per vehicle class, per detector was considered as MoP. Equation (2) is then specified as:

$$RMSE(v) = \sqrt{\frac{1}{(H \cdot 2 \cdot 2)} \sum_{t=1}^H \sum_{k=1}^2 \sum_{d=1}^2 (v_{tkd}^{obs} - v_{tkd}^{sim})^2}$$

where the number of time intervals to which the first summation is extended,  $H$ , is equal to  $T$  in the case of  $F_2$  and it is equal to the number of stationary intervals in the case of  $F_3$ .

## Results

Eight parameters were calibrated; the two (most sensitive) parameters per vehicle class (see Table 1). In the simulation-based calibrations the OQMS algorithm launches AIMSUN at each iteration for evaluating the objective function. By contrast, the model-based calibration relies on the analytical stationary traffic flow model (for a detailed description of the stationary multi-class traffic flow model calibration refer to (15)). It is worth noting that the stationary model was derived i.e. is consistent with the car-following model in AIMSUN (the Gipps' c.f. model).

Results (Table 1, Table 2 and Figure 4) highlight three kinds of information: values of calibrated parameters and their deviation from the "true" ones in Table 1; the Measures of Goodness of Fit ( $MGF$ ) between the time series of observed "laboratory" data and the simulated ones in Table 2, and a visual comparison of these time series of data in Figure 4.

The  $MGF$  used were the Root Mean Square Error (RMSE) and the Root Mean Square Percentage Error (RMSPE) given respectively by equations (2) and (3) and the GEH statistic. A reference on the use and the meaning of GEH can be found e.g. in (9). Its expression is:

$$GEH(MoP_{tdk}) = \sqrt{\frac{2 \cdot (MoP_{tdk}^{obs} - MoP_{tdk}^{sim})^2}{(MoP_{tdk}^{obs} + MoP_{tdk}^{sim})}} \quad (4)$$

Regarding the contents of Table 1, we may first observe that both in  $F_1$  and  $F_2$  calibrations the quality of the minimum found is acceptable: parameter values are close enough to the "true" ones and their values are consistent with their physical meaning (e.g.  $rT_1^F$  is always lower than  $rT_2^F$ ).

This is confirmed by results reported in Table 2 and in Figure 4: values of RMSE and RMSPE show a close relationship between observed and simulated outputs. Even more impressive is the GEH distribution: its value is less than 3 for 100% of the simulated outputs (an acceptable goodness of fit is considered if with 85% of the simulated outputs its value is less than 5 (9)). Lastly, Figure 4 shows a very good reproduction of the observed time series. These results also show that the higher complexity of the objective function in  $F_1$ , does not improve solution accuracy. For this reason, as already pointed out in (17), an objective function with only the speed as  $MoP$  still remains preferable.

Regarding the model-based calibration ( $F_3$ ), values found for parameters are undoubtedly farther from the "optimal" ones with respect to the other approaches, and the measures of goodness of fit are poor.

The behaviour exhibited by the simulation model, using values of parameters calibrated in  $F_3$  (lower part of Figure 4) is particularly interesting. Indeed, such values of parameters cause a different distribution of vehicles between the two lanes, with respect to the case with the optimal values. In particular on the right-hand lane (that depicted) the counts are higher than those expected.

**TABLE 1. Optimal and calibrated parameter values**

Parameters	Unit	"True" Values	Calibrated Values					
			$F_1$		$F_2$		$F_3$ (Stationary)	
$rT_1^F$	seconds	0.5	0.6	-20%	0.4	20%	0.64	-28%
$sA_1^F$	--	1.3	1.49	-15%	1.21	7%	1.02	22%
$rT_1^S$	seconds	0.8	0.7	13%	0.9	-13%	0.91	-14%
$sA_1^S$	--	1	1.02	-2%	1.04	-4%	0.88	12%
$rT_2^F$	seconds	1.2	1.1	8%	0.9	25%	0.93	23%
$sA_2^F$	--	0.85	0.86	-1%	0.87	-2%	1.02	-20%
$rT_2^S$	seconds	1.5	1.4	7%	1.4	7%	1.5	0%
$sA_2^S$	--	0.7	0.72	-3%	0.72	-3%	0.88	-26%

**TABLE 2. Measures of Goodness of fit for the three calibrations**

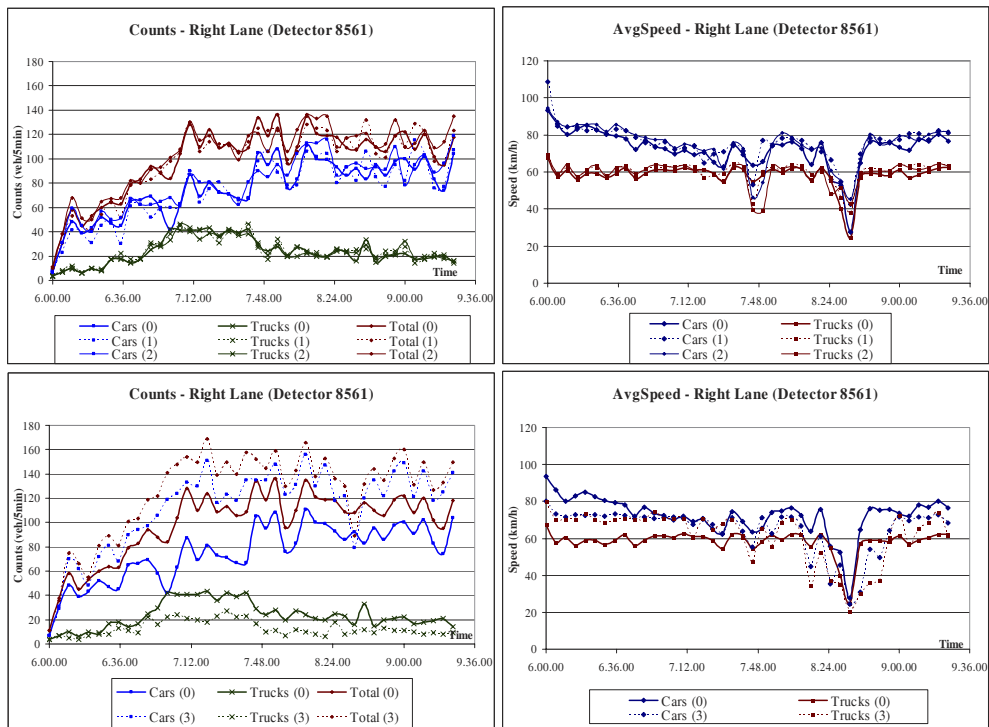
MGF	MoP	$F_1$	$F_2$	$F_3$ (Stationary)
RMSE	Counts (Total)	7.93	8.01	30.99
	Speed (Total)	8.40	5.05	12.06
	Counts (Cars)	10.70	10.83	41.94
	Speed (Cars)	5.07	5.15	11.15
	Counts (Trucks)	3.34	3.30	12.69
	Speed (Trucks)	10.74	4.95	12.91
RMSPE	Counts (Total)	17%	17%	62%
	Speed (Total)	11%	11%	16%
	Counts (Cars)	14%	13%	49%
	Speed (Cars)	9%	10%	15%
	Counts (Trucks)	19%	20%	72%
	Speed (Trucks)	13%	11%	17%
GEH Distribution	Counts			
	Speed			

**CONCLUSIONS**

The contribution of the present paper is two-fold.

On one hand the paper reports results of the application of OptQuest/Multistart algorithm (OQMS) to the problem of simulation-based calibration of traffic flow micro-simulation models. The algorithm has shown its ability in reaching sub-optimal solution not far from the global one. It is worth noting that the optimization problem was set as a mixed-integer problem with 4 discrete and 4 continuous decision variables. Even if the discrete variables (the vehicle reaction times) exhibit higher deviations than the others from the optimal values, they were always consistent with their physical meaning (e.g. reaction times of cars always lower than those of trucks). The verification of the solution quality has been possible thanks to the execution of tests with laboratory

data. However computing time was not satisfactory and higher than the one expected (the MI calibrations lasted 10 days circa, each AIMSUN run taking approximately 3 minutes).



**FIGURE 4. Comparison of time series of observed and simulated outputs: (0) stands for the observed data, (1), (2) and (3) for the simulated outputs, respectively, in the two simulation-based calibrations and in the model-based one**

On the other hand results from the stationary model-based calibration have exceeded the expectations, also provided the procedure simplicity and that they pay for the assumptions behind the model: mainly the one of stationarity and thus the absence of lane-changing. Nevertheless the values of parameters calibrated have an average deviation of 18% and never deviate more than the 28% from the optimal ones. Such numbers allow to not completely dismissing results from a stationary model-based calibration. Rather it seems they can be at least used in place of default parameter values in common practice simulations, also considering that they preserve consistency with the observed road capacity (see (15) for further details).

Further investigations are needed to confirm preliminary results from this study. Moreover it is worth investigating if results from a stationary model-based calibration can be used to enhance performances of a simulation-based one. As mentioned before they could be used both as starting values for a simulation-based calibration and for reducing the amplitude of the box constraints around them when setting the simulation-based calibration. Finally it is desirable that future research would cover a comparison among heuristics available for simulation-based calibrations and an investigation of which heuristic could more benefit from the integration with a model-based calibration.

## AKNOWLEDGEMENTS

Authors are grateful to Davide Mattera for the invaluable discussions about optimisation. Traffic data were available under the project no. 12897, PON 2000 – 2006: “Sistema di monitoraggio, controllo ed informazione per la gestione attiva della sicurezza di un’infrastruttura autostradale”.

## REFERENCES

1. Vavasis, S. A. *Nonlinear Optimization: Complexity Issues*, Oxford University Press, New York, 1991.
2. Traub, J.F., and A.G. Werschulz. Information-based complexity and information-based optimization in *Encyclopedia of Optimization* C.A. Floudas and P.M. Pardalos Eds, Kluwer Academic Publishers, 2001, Vol. 2, pp. 485-489
3. Ma, T., B. Abdulhai. Genetic Algorithm-Based Optimization Approach and Generic Tool for Calibrating Traffic Microscopic Simulation Parameters. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1800*, TRB, National Research Council, Washington, D. C., 2002, pp. 6-15.
4. Kim, S.-J., Kim, W., L. R. Rilett. Calibration of Microsimulation Models Using Nonparametric Statistical Techniques. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1935*, TRB, National Research Council, Washington, D. C., 2005, pp. 111-119.
5. Park, B., H. Qi. Development and Evaluation of a Procedure for the Calibration of Simulation Models. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1934*, TRB, National Research Council, Washington, D. C., 2005, pp. 208-217.
6. Toledo, T., Ben-Akiva, M., Darda, D., Jha, M., H. Koutsopoulos. Calibration of Microscopic Traffic Simulation Models with Aggregate Data. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1876*, TRB, National Research Council, Washington, D. C., 2004, pp. 10-19.
7. Hourdakis, J., Michalopoulos, P. G., J. Kottommannil. Practical Procedure for Calibrating Microscopic Traffic Simulation Models. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1852*, TRB, National Research Council, Washington, D. C., 2003, pp. 130-139.
8. Balakrishna, R., Antoniou, C., Ben-Akiva, M., Koutsopoulos H. N., Y. Wen. Calibration of Microscopic Traffic Simulation Models: Methods and Application. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1999*, TRB, National Research Council, Washington, D. C., 2007.
9. Ma, J., Dong, H., H. M. Zhang. Calibration of Micro Simulation with Heuristic Optimization Methods. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1999*, TRB, National Research Council, Washington, D. C., 2007.
10. Ugray, Z., Lasdon, L., Plummer, J., Glover F., Kelly, J., R. Marti. A Multistart Scatter Search Heuristic for Smooth NLP and MINLP Problems. In *INFORMS Journal of Computing*. July 25, 2002.
11. Glover, F., A Template for Scatter Search and Path Relinking in *Artificial Evolution, Lecture Notes in Computer Science 1363*, J.-KRelinking. J.-K. Hao, E. Lutten, E. Ronald, M. Schoenauer and D. Snyers, eds., Springer Verlag, 1998, pp. 13-54.
12. Smith, S., L. Lasdon, Solving Large Sparse Nonlinear Programs Using GRG. *ORSA Journal on Computing* 4 1, 1992, pp.3-15.
13. Drud, A. S. CONOPT – A large-Scale GRG Code. In *ORSA Journal of Computing, Vol. 6, No. 2*. Operation Research Society of America, 1994, pp. 207-216.
14. Schultz, G.G., L.R. Rilett, Calibration of Distributions of Commercial Motor Vehicles in CORSIM. In, *Transportation Research Record: Journal of the Transportation Research Board, No. 1934*, TRB, National Research Council, Washington, D.C., 2005, pp.246-255
15. Punzo, V., Tripodi, A., Steady-state solutions and multi-class calibration of Gipps’ microscopic traffic flow model, *Transportation Research Record: Journal of the Transportation Research Board, No. 1999*, TRB, Washington D.C., 2007.
16. Kurian, M., Calibration of a Microscopic Traffic Simulator. *Master Thesis*, Massachusetts Institute of Technology, 2001.
17. Ciuffo, B., Punzo, V., V. Torrieri. A framework for calibrating a microscopic simulation model. *Proceedings of the 86<sup>th</sup> Annual Meeting of the Transportation Research Board*, Washington, D. C., 2007.

18. Locatelli, M., F. Schoen, Random Linkage: a Family of Acceptance/Rejection Algorithms for Global Optimization, *Mathematical Programming* 85 2, 1999, pp. 379-396.
19. *AIMSUN 5.1 Microsimulator User's Manual*, TSS-Transport Simulation Systems, Barcelona, Spain, 2006
20. Levison, W.H. Interactive Highway Safety Design Model – Issue Related to Driver Modelling. In *Transportation Research Records: Journal of the Transportation Research Board, No. 1631*, TRB, National Research Council, Washington, D. C., 1998.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52

# Integrated Environment of Driving and Traffic Simulation

B. Ciuffo, V. Punzo, V. Torrieri  
University of Napoli "Federico II"

Department of Transportation Engineering "L. Tocchetti"

Via Claudio, 21, 80125, Napoli, Italy

[bc Huffo@unina.it](mailto:bc Huffo@unina.it), [vinpunzo@unina.it](mailto:vinpunzo@unina.it), [torrieri@unina.it](mailto:torrieri@unina.it)

## I. INTRODUCTION – TRAFFIC REQUIREMENTS

Major advantage of using a driving simulator is the chance of carrying out safe experiments, in a controlled environment, assuring equal driving conditions to all the test drivers. Nonetheless the reliability of results is seriously affected by the likelihood of the simulation environment. Behaviour of surrounding vehicles, generally referred as the traffic scenario, has been recognized as a key element in a realistic simulation.

A realistic traffic flow generally requires that:

(i) *autonomous* vehicles move around the *interactive* one in a realistic manner, i.e. showing realistic kinematics and vehicles interactions. In particular, when testing Intelligent Transportation Systems (ITS), traffic in the driving simulation should behave accordingly. For instance, in case of Advanced Driver Assistance Systems (ADAS), the ability of autonomous vehicles of moving accordingly to such systems is clearly advisable for a comprehensive evaluation of the impacts on drivers as well as on traffic flows.

(ii) vehicles move according to macroscopic behaviour of traffic streams, e.g. the higher the number of vehicles on the road, the lower the average speed of the traffic stream.

(iii) while moving on a network, the driver of the interactive vehicle should meet the "same" traffic he expects or he would find in the real network, i.e. the dynamic pattern of traffic flows on the simulated network should closely resemble the actual one.

In order to fulfil the first requirement accurate driver models are necessary (e.g. see [1-6]). Then the second issue requires that the aggregate behaviour of a traffic stream, resulting from individual behaviours, was consistent with the macroscopic laws of traffic flows (see for ex. [5]). This can be achieved, for example, by calibrating individual driver model parameters by means of aggregated traffic data (rather than microscopic data, e.g. [7]), like time series of speeds at detectors, e.g. [8, 9]. Finally, the third point requires that the traffic simulation was extended to the whole network [5].

When this is accomplished by means of a macroscopic or mesoscopic model, effects of some ITS strategies on the network flow propagation cannot be properly accounted for. It is the case of the design and evaluation of systems affecting individual behaviours and choices of drivers, like in-vehicle information systems (e.g. [10]).

In the aim of providing the simulation environment with

reliable and comprehensive traffic scenarios, the microscopic traffic simulation software AIMSUN [11] has been integrated within the dynamic driving simulator of the Road Safety Laboratory at TEST [12]. The integration allows to easily cope with the complex traffic scenarios resulting from the implementation of ITS strategies like, for example, traffic responsive control strategies. Moreover, the opening provided in AIMSUN for plugging-in any user-defined driver's model into the simulation framework, turns out to be essential for easily implementing and emulating systems like ADAS.

The integration opens to a newer scenario in which a driving simulator and a microscopic traffic simulator are alternatively used for their mutual development: driving behaviour of the interactive vehicle used for developing and calibrating autonomous vehicles behavioural models and autonomous vehicles behaviour for validating interactive vehicle way of driving. For these kind of analyses the integrated environment shows to be also suitable for the great advantage of being able to perform controlled experiments, without the need of complex data estimation procedure (e.g. [11]).

This paper therefore describes criteria, features and technical issues of such an integration.

## II. THE INTEGRATION FRAMEWORK OF A DRIVING SIMULATOR AND A TRAFFIC SIMULATOR

The integration of a driving simulation and a traffic simulation environment is possible by implementing a communication structure able to perform an effective data exchange between the two applications (Fig. 1). The integrated environment has to allow that, during the driving simulation, the autonomous vehicles are moved according to AIMSUN congruently with the interactive vehicle movements. More in detail it is necessary that, during the simulation, the software that manages the driving simulator (namely SCANeR [13]), would send continuously to AIMSUN the cinematic characteristics (i.e. position, angle, speed, etc.) of the interactive vehicle. Once received such information AIMSUN can calculate the cinematic characteristics of all the other vehicles and then send them to SCANeR that plays the final role of updating these information during the driving simulation.

In order to fulfill these objectives it is necessary to choose a communication strategy able to effectively perform the data exchange in real time and to ensure the internal congruence between the road networks created in both environments.

#### A. Issues of integration

In order to implement the Communication Framework it is necessary to cope with several issues. These issues are briefly summarized below.

##### 1) Accurate road matching between traffic and driving modules

First preliminary issue for the integration of the two applications is the congruence between their road networks. Position related data to be exchanged, indeed, have to be, first of all, in a well known reference system in order to allow an appropriate transformation between the two environments during the communication. Then the two road networks have to match in an exact way in order to have a strict correspondence between the two simulations.

For what concerns the first aspect, there are two simple ways to overcome the problem. The network creation, indeed, starts in both the software with the importation of the road axis. Both applications are able to import data from a .dwg file or a .shp file (i.e. from a drawing file or from a shapefile). Thus it is only important to avoid any geographical information transformation during the importation procedure to ensure the consistency of the two reference systems. In this way data retrieved by the two environments are ready to be exchanged.

For what concerns the road matching problem, it is mainly due to the fact that a driving simulator requires very detailed road geometry to present as far as possible a realistic road for the driver. This is obviously far from the purposes of a microscopic traffic simulator. However, the tools available in AIMSUN for the network creation already allow the refinement required at least for the road horizontal alignment. On the contrary, right now, for what concerns the road vertical alignment it is only possible to set the initial and final altitude of every road section (i.e. it is only possible to set the

slope). For this reason in order to reach a high accuracy also of the vertical profile of the road it is necessary to create

a big number of road segments and this can create problems for the AIMSUN *lane-changing* models and for the time needed in performing a simulation (see [11] for more details).

##### 2) Real-time traffic simulation

As anticipated the communication framework has to allow the real time data exchange needed to perform an efficient driving simulation. However, AIMSUN is not designed to work in real time but, according to the hardware performances and to the scenario complexity, it performs the required calculations as fast as possible. This means that, usually, the time needed for a simulation is lower than the real time. For this reason it was necessary to make AIMSUN running at a lower speed.

##### 3) Synchronization of traffic and driving modules

The criterion used to slow down AIMSUN consists of its synchronization with SCANeR. Practically, as will be shown in the remainder, at each simulation step, AIMSUN compares its simulation time with the SCANeR simulation time (i.e. the real time) until it is bigger than this. This procedure ensures the temporal consistency of the data exchanged between the two applications during the simulations.

##### 4) Real-time data exchange

As already stated the microscopic traffic simulation usually requires less than the real simulation time to be performed. However, if the data exchange is not efficient (or if the amount of data to be sent from an application to the other is too big) the time required can excessively slow down the simulation. As a consequence, this can totally void the integration framework.

##### 5) Consistency of the data updating frequency

Each simulation model updates the results of its calculation at a given frequency. The finer are the expected results, the higher is such frequency. Both a microscopic traffic simulation model and a driving simulation software work in this way. In the former the frequency is explicitly set by means of a parameter, namely the *simulation step*. For the objectives of this simulation the higher possible frequency, right now, is 10 Hz. In the latter, instead, the situation is a bit more complex. The simulation software is made up of different sub-modules each of which working at its own frequency. The communication among the different modules can be achieved in an Asynchronous or in a Pseudo-Synchronous way. In both of them the SCANeR process runs at its own frequency, but in the former it always gets the most recent values for data, while in the latter it is able to get values of data only when they are explicitly sent by each module. The latter allows AIMSUN to send data to SCANeR just after each calculation without any care of the specific instant in which this happens.

##### 6) Management of the autonomous vehicles visualization (in the driving environment)

This issue consists of two different topics. First of all, a microscopic traffic simulation is able to manage thousands of vehicles contemporary. The management of a so high number of vehicles in a driving simulation environment is infeasible and also useless due to the driving simulation purpose. For this reason, only information related to the vehicles surrounding the interactive one are exchanged during the

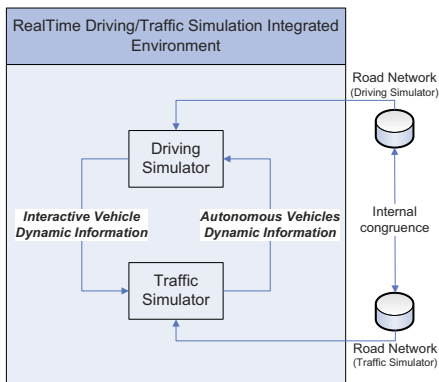


Fig. 1 – Schematic representation of the Communication Framework



simulation. For this reason it is necessary to create in SCANer a certain number of vehicles and to position them out of the network. When a vehicle in AIMSUN enters the surroundings of the interactive one, one of the autonomous vehicles in SCANer instantaneously changes its position and appears in the driving simulation. This procedure needs for an unambiguous correspondence, from a simulation step to another, between each vehicle in AIMSUN and in SCANer. More specifically, until a vehicle in the AIMSUN simulation remains within the surroundings of the interactive vehicle it has to be represented always by the same vehicle in SCANer.

These issues don't constitute the exhaustive list of problems to deal with the integration implementation. They however are the most basic issues in order to obtain an effective framework. In the following, the framework implemented will be shown and the way by which all these issues have been coped with detailed.

### B. Software architecture

Several strategies can be used for implementing the communication framework. However, not all the possible options can be actually carried out for several reasons. In particular, as already explained, a fundamental requirement is the communication rapidity. A high time consuming strategy would cause problems during the simulation for what concerns the real time input restitution.

Moreover the strategy should involve as long as possible functions and methods already available in both software. Indeed the use of tools created ad hoc for the users by the software developers could nullify, for example, the possibility for the software upgrading.

For such reasons the way chosen in this work has exclusively make use of the standard Application Program Interfaces (APIs) available in both software.

One of the main shortcomings of using the APIs concerns with the rapidity allowed, but in this case, due to their peculiar structure the problem was overcome, as will be showed in the remainder, using a shared environment (Fig. 2).

### C. The shared environment

The problem of using the APIs is mainly connected to the

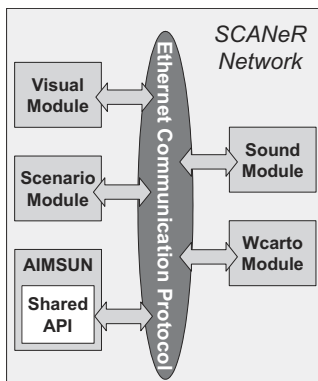


Fig. 2 – Schematic representation of the software architecture

way by which storing data to be exchanged. If an external database is used, the time required would make impossible the real time transmission. The only solution is to use a unique environment for both the APIs in order to directly use data retrieved during the simulations. This was possible thanks to their particular structure.

In SCANer, each model used during the simulation is included in a specific module. The modules communicate each other by means of messages sent and collected on the network connecting them during the simulation. The API (based on the C++ programming language) is used to create an additional module (an executable application) able to communicate with the others in the same way [14].

On the other hand, in AIMSUN, this modular structure does not exist and the API (also here based on the C++ programming language) consists of a Dynamic Load Library (DLL) in which the user can update the content of six functions that the software calls during its execution [15].

In the light of these features the strategy carried out has consisted of using the AIMSUN API as shared environment in which including all the programming code. Obviously it requires the inclusion in DLL of all the libraries needed by SCANer. In this way, practically, AIMSUN itself is the new SCANer (Fig. 2) module and this is very suitable from a conceptual point of view.

During its execution AIMSUN calls six functions from the DLL used as an API. These functions are [15]:

- AAPIOload();
- AAPIOinit();
- AAPIManage(...);
- AAPIOPostManage(...);
- AAPIOfinish();
- AAPIOunload().

These functions allow the user to access information regarding every simulation phase. In this way it is possible to manage all the simulation phases of both the applications in a correct manner. In particular:

- AAPIOload and AAPIOunload are called when the module created by the API is loaded and unloaded.
- AAPIOinit is called just at the beginning of the simulation and have been used here to (i) load the SCANer module, to (ii) retrieve the starting position of the interactive vehicle in SCANer, to (iii) create a tracked vehicle in AIMSUN and to (iv) set its position to the starting position of the interactive vehicle, to (v) set the starting position of all the autonomous vehicles in SCANer.
- AAPIManage is called every AIMSUN simulation step at its beginning and thus it is used to retrieve the position of the interactive vehicle in SCANer and to update the position of the tracked vehicle in AIMSUN.
- AAPIOPostManage is called every AIMSUN simulation step at its end and thus is used to retrieve the position of all the AIMSUN vehicles and to update the position of all the autonomous vehicles in SCANer.
- AAPIOfinish is called at the end of the simulation and it

is used to close the SCANeR module.

### III. FIRST TESTS

The first tests have been carried out on a 6.5km two carriageways road of a flat (i.e. constant altitude) French landscape. The traffic flow considered in the simulation has consisted of 1.000 vehicles per hour. SCANeR 2.18 and AIMSUN 5.1.8 releases have been used in the integration framework. Information have been exchanged for vehicles within a distance of 200m from the interactive one. Different attempts have been carried out with different numbers of vehicles in SCANeR. The integration framework has efficiently allowed the real time driving simulation with a number of SCANeR vehicle lower than 30. However 25 vehicles resulted enough to simulate the surrounding vehicles of the interactive one. In Fig. 3 a snapshot of the two networks with the correspondence of the simulated vehicles is shown.

### IV. SHORTCOMINGS

Despite of the overall good performance of the integration framework a number of shortcomings have been encountered. Most of them were expected before the implementation, and will be overcome with the release of AIMSUN 6 version.

#### A. Low Updating Frequency

As anticipated the maximum updating frequency allowed in AIMSUN is 10Hz. This constraint hasn't any physical meaning. It is only linked to the model purposes (a simulation step less than 0.1s is considered ineffective). Moreover AIMSUN 5.1.8 is not fast enough to run a complex simulation at a frequency of 20-30 Hz in real time.

For visual purposes during the driving simulation, on the contrary, an updating frequency of at least 30Hz is required. This represents a strong limitation for the integration framework, because the visual restitution is the most basic element of the simulation.

#### B. Vehicles Lane-Changing Behavior

For the traffic simulation purposes the lane changing-behavior of the vehicle is very coarse from a visual point of

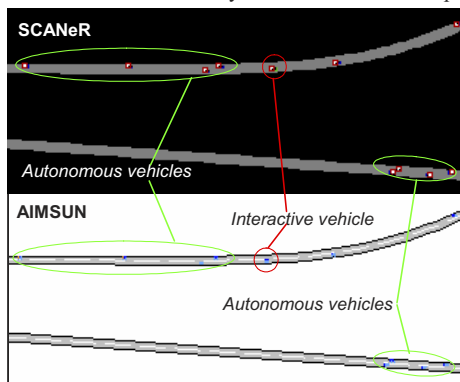


Fig. 3 – Vehicles correspondence in the two simulation environments

view. In addition, in AIMSUN each vehicle occupies exactly the central position of the lane on which it is. In order to achieve a vehicle behavior suitable for a driving simulation it is needed:

#### 1) Robust implementation of the interactive vehicle

AIMSUN has to deal with "not valid" positions as the interactive vehicle can move outside the network (e.g. in the side walk).

#### 2) The real car can block more than one lane and/or turning.

3) Vehicles behavior in curve sections and while turnings has to be improved

#### 4) While changing the lane, the autonomous vehicles have to behave realistically.

For example they should rotate a bit, depending on the speed.

All these features will be available in AIMSUN 6.

#### C. Two lanes rural highway simulation

This is probably the main issue in integrating driving and microscopic traffic simulation. Indeed, at the moment, commercial micro-simulation packages do not simulate two-lane rural highways with overtaking on the opposing lane, despite of some works exist on the topic [16].

### V. CONCLUSION

The present paper has described the conceptual path followed during the implementation of the first prototype of software integration between the environments of driving simulation and microscopic traffic simulation. The fundamental issues dealt with during the work are described and main shortcomings analyzed. The implemented framework is able to make the two applications effectively communicating. However the visual restitution obtained doesn't allow yet a correct driving simulation. However the major improvements expected at the release of AIMSUN 6 and the new research perspectives in driving and traffic simulations that such integration opens suggest the necessity for further deepen investigations.

### VI. ACKNOWLEDGMENTS

Authors are grateful to Alexandre Troale from OKTAL for the precious support to the work.

### REFERENCES

- [1] Espié, S. "ARCHISIM: Multiactor parallel architecture for traffic simulation". In *Proceedings of the second world congress on Intelligent Transport Systems '95*. Yokohama, Japan, 1995
- [2] El hadouaj, S. and S. Espié. "A generic road traffic simulation model". In *Proceedings of ICTTS (Traffic and transportation studies)*, Guilin, China, 2002.
- [3] Ahmad, O. and Y. Papelis. "A Comprehensive Microscopic Autonomous Driver Model for Use in High-Fidelity Driving Simulation Environments". In *Proceeding of 82<sup>nd</sup> TRB Annual Meeting*. Washington DC, 2002.
- [4] Wright S. "Supporting intelligent traffic in Leeds driving simulator". *PhD thesis* at the school of Computing, University of Leeds, 2000.
- [5] Olstam J. J. "Generation and Simulation of surroundings vehicles in a driving simulator". In *Proceedings of Driving Simulator Conference, DSC'06*, Paris, 2006.
- [6] Jenkins, J. M. "Modelling the interaction between passenger cars and trucks". *PhD thesis* at the Texas A&M University, Texas, 2004.

- [7] Punzo V., Formisano D. J., Torrieri V. "Non-stationary Kalman Filter for the estimation of accurate and consistent Car-Following Data", *Transportation Research Records 1934*, TRB, Washington D.C., 2005, pp. 3-12.
- [8] Toledo, T., Ben-Akiva, M., Darda, D., Jha, M., H. Koutsopoulos. "Calibration of Microscopic Traffic Simulation Models with Aggregate Data. In *Transportation Research Records*, No. 1876, TRB., Washington, D. C., 2004, pp. 10-19.
- [9] Punzo V., Tripodi A., "Steady-state solutions and multi-class calibration of Gipps' microscopic traffic flow model" In *Proceedings of the 86<sup>th</sup> TRB annual meeting*, Washington D.C. 2007.
- [10] Jin M., Lam S.-H. "A Virtual-Reality Based Integrated Driving-Traffic Simulation System to Study the Impacts of Intelligent Transportation Systems (ITS)". In *Proceedings of the 2003 International Conference of Cyberworlds*. 2003.
- [11] AIMSUN NG 5 Microsimulator User's Manual 5.1. *Transport Simulation Systems 1997-2005*. November 2006.
- [12] TEST scarl, Centro Regionale di Competenza "Trasporti" [www.crdetest.it](http://www.crdetest.it)
- [13] SCANer II User's guide v2.18a. Oktal © 2006. 28 Novembre 2006.
- [14] SCANer II API Programming guide v2.18a. Oktal © 2006. 28 Novembre 2006.
- [15] AIMSUN API Manual 5.1.2. *Transport Simulation Systems 1997-2005*. October 2006.
- [16] Tapani A. "A Versatile Model for Rural Road Traffic Simulation". In *Proceedings of the 84<sup>th</sup> Annual Meeting of the Transportation Research Board*, Washington D.C., 2005.