

Treball de Fi de Grau

Grau en Enginyeria en Tecnologies Industrials

**Simulació de la fluència del trànsit de vehicles
comparant el comportament humà i l'impacte de
vehicles autònoms**

MEMÒRIA

Autor: Jaume Cartró Benavides
Director: Arnau Dòria Cerezo
Convocatòria: Gener 2017



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resum

Aquest projecte té per objectiu el desenvolupament d'un programa informàtic capaç de modelar i simular el trànsit en carreteres no urbanes i veure com es creen congestions. Per dur a terme aquest propòsit s'empren models de seguiment de vehicles. Destacar la gran flexibilitat del model que permet incorporar fàcilment qualsevol algoritme de seguiment.

Per al seu desenvolupament s'han utilitzat les eines que proporciona *Simulink* i *Matlab* així com el seu propi llenguatge de programació. Un cop s'han realitzat tots els càlculs corresponents, el programa disposa d'una interfície d'usuari la qual mostra una animació de l'evolució dels vehicles al llarg del temps. També s'exposen gràfics on es visualitza la velocitat o acceleració dels vehicles que ajuden a entendre com es comporten segons l'algoritme de seguiment triat. Després de cada simulació totes les dades queden enregistrades i poden servir per estudiar l'eficiència del sistema.

Llistat de figures

| | |
|--|----|
| Figura 4-1: Taula de valors dels paràmetres típics del model IDM..... | 15 |
| Figura 4-2: Corba força tracció màxima | 17 |
| Figura 4-3: Diagrama del cos lliure de un vehicle | 17 |
| Figura 4-4: Distribució de la força normal | 18 |
| Figura 5-1: Circuit en el pla | 19 |
| Figura 5-2: Circuit amb alçada amb 3 pics | 20 |
| Figura 6-1: Esquema intern de funcionament..... | 22 |
| Figura 6-2: Esquema conceptual del fitxer de Simulink..... | 24 |
| Figura 6-3: Gràfic adaptació de velocitat | 25 |
| Figura 6-4: Efecte de la part integral del controlador | 26 |
| Figura 6-5: Esquema conceptual de la implementació del model mecànic | 26 |
| Figura 6-6: Integral Anti-Windup del model | 27 |
| Figura 6-7: Simulació que no incorpora Anti-Windup | 27 |
| Figura 6-8: Simulació que incorpora Anti-Windup | 28 |
| Figura 6-9: S_7 , v_7 i a_7 abans i després de l'Anti-Windup | 28 |
| Figura 6-10: Passos per posicionar el rectangle en 2D | 30 |
| Figura 6-11: Part del codi on s'obtenen els paràmetres automàticament..... | 31 |
| Figura 6-12: Matriu 'Par' | 32 |
| Figura 6-13: Taula de paràmetres propis dels vehicles | 32 |
| Figura 6-14: Exemple de distribució de vehicles | 33 |
| Figura 6-15: Distribució de s_0 , V_0 , T , b i a per a una mostra de 1000 vehicles..... | 34 |
| Figura 7-1: Gràfic acceleracions amb Model IDM..... | 36 |

| | |
|---|----|
| Figura 7-2: Disposició dels vehicles al llarg del temps, Model IDM | 36 |
| Figura 7-3: Gràfic de velocitats, Model IDM | 37 |
| Figura 7-4: Disposició dels vehicles al llarg del temps, Model Simple | 37 |
| Figura 7-5: Gràfic de velocitats, Model Simple | 38 |
| Figura 7-6: Taula simulacions variant el nombre de vehicles | 40 |
| Figura 7-7: Exemples del Diagrama Fonamental | 41 |
| Figura 7-8: Diagramas de Flux Fonamental realitzats amb la plataforma de simulació del projecte | 42 |
| Figura 7-9 Taula amb paràmetres de referència | 43 |
| Figura 7-10: Taula de simulacions variant un paràmetre..... | 44 |
| Figura 7-11: Circuits en el pla i amb alçada | 46 |
| Figura 7-12: Gràfic d'acceleracions del model en el pla | 46 |
| Figura 7-13: Gràfic d'acceleracions del model amb alçada | 46 |
| Figura 7-14: Gràfic de velocitats del model en el pla..... | 47 |
| Figura 7-15: Gràfic de velocitats del model amb alçada..... | 47 |
| Figura 8-1: Taula Materials i Software utilitzats | 48 |
| Figura 8-2: Taula del temps de treball emprat..... | 48 |

Taula de continguts

| | |
|--|-----------|
| RESUM | 1 |
| LLISTAT DE FIGURES | 3 |
| TAULA DE CONTINGUTS | 5 |
| 1. GLOSSARI | 9 |
| 2. PREFACI | 10 |
| 2.1. Context del projecte..... | 10 |
| 2.2. Origen del projecte | 10 |
| 2.3. Antecedents | 11 |
| 3. INTRODUCCIÓ | 12 |
| 3.1. Objectius del projecte | 12 |
| 3.2. Abast del projecte..... | 12 |
| 4. MODEL MATEMÀTIC | 13 |
| 4.1. Models de conducció..... | 13 |
| 4.1.1. Model Senzill..... | 13 |
| 4.1.2. Model IDM..... | 14 |
| 4.2. Model mecànic dels vehicles..... | 16 |
| 4.2.1. Força de tracció | 16 |
| 4.2.2. Forces resistents..... | 17 |
| 4.2.3. Velocitat real | 18 |
| 5. MODEL DEL CIRCUIT | 19 |
| 5.1. Model en el pla | 19 |
| 5.2. Model amb alçada | 20 |
| 6. IMPLEMENTACIÓ DEL MODEL | 21 |
| 6.1. Estructura general del programa/sistema | 21 |
| 6.2. Simulink..... | 23 |
| 6.2.1. Implementació del model mecànic..... | 24 |
| 6.2.2. Integral Anti-Windup..... | 27 |
| 6.3. Matlab..... | 29 |
| 6.3.1. Càlcul de distàncies | 29 |
| 6.3.2. Posicionar vehicles | 29 |
| 6.3.2.1. Cas dos dimensions | 29 |

| | |
|--|-----------|
| 6.3.2.2. Cas tres dimensions..... | 30 |
| 6.4. Paràmetres | 30 |
| 6.4.1. Paràmetres propis dels vehicles | 32 |
| 6.4.2. Paràmetres propis dels perfils de conducció..... | 33 |
| 7. SIMULACIONS | 35 |
| 7.1. Simulació Model IDM..... | 35 |
| 7.2. Simulació amb el model Simple..... | 37 |
| 7.3. Simulacions per veure afectació de la densitat de vehicles..... | 39 |
| 7.4. Simulacions per observar l'afectació individual dels paràmetres pertanyents als perfils del conductors..... | 42 |
| 7.5. Simulació amb el model amb alçada | 45 |
| 8. PRESSUPOST ECONÒMIC | 48 |
| CONCLUSIONS | 49 |
| AGRAÏMENTS | 51 |
| BIBLIOGRAFIA..... | 52 |
| Referències bibliogràfiques | 52 |
| Bibliografia complementària | 52 |
| ANNEX | 53 |
| A. Esquemes de blocs Simulink..... | 54 |
| A.1. Esquema de blocs principal | 54 |
| A.2. Esquema dels blocs del model mecànic | 55 |
| A.2.1. Blocs del càlcul de la força de referència..... | 56 |
| A.2.2. Blocs per obtenir la força real | 57 |
| A.2.3. Blocs per obtenir la velocitat real | 58 |
| A.2.4. Blocs per obtenir la fora resistent total | 59 |
| A.3. Blocs de inicialització i reset del l'integrador velocitat-posició | 60 |
| B. Fitxers Matlab | 61 |
| B.1. Traffic flow simulation 2D | 61 |
| B.2. Traffic flow simulation 3D | 63 |
| B.3. Vehicles and Driving behaviour paramers..... | 67 |
| B.4. Input Parameters | 69 |
| B.5. Functions | 70 |
| B.6. Track..... | 71 |
| B.7. Length curve ('Interpreted Matlab Fcn' Simulink block)..... | 71 |

| | | |
|-------|---|----|
| B.8. | Total length ('Interpreted Matlab Fcn' Simulink block)..... | 72 |
| B.9. | IDM Model ('Matlab Function' Simulink block)..... | 72 |
| B.10. | Simple Model ('Matlab Function' Simulink block) | 73 |
| B.11. | Slope ('Matlab Function' Simulink block)..... | 73 |
| B.12. | Conversion to horitzontal speed ('Matlab Function' Simulink block)..... | 73 |
| B.13. | Programa per crear els Diagrames Fonamentals..... | 73 |
| B.14. | Programa per mostrar les propietats normals de cada paràmetre | 74 |
| C. | Gràfiques de les simulacions..... | 77 |

1. Glossari

Flux de trànsit microscòpic: a diferència del flux macroscòpic que usa variables per reflectir l'estat mitjà del trànsit, el flux macroscòpic usa variables per fer focus individualment a cada conductor.

Matlab: plataforma optimitzada per resoldre problemes de ingenieria i científics. El llenguatge MATLAB es basa en matrius per expressar de forma natural les matemàtiques computacionals.

Simulink: entorn de programació visual que funciona sobre l'entorn de programació Matlab. Gràcies a aquesta característica proporciona un nivell d'abstracció de nivell més alt que el llenguatge MATLAB.

Adaptive Cruise Control (ACC): sistema que mesura la distància respecte al vehicle precedent mitjançant un sensor de radar especial i controla la velocitat garantint que no se supera el valor establert.

Controlador PID: mecanisme de control per realimentació el qual calcula la desviació o error entre un valor mesurat i un valor desitjat. L'algoritme de control PID consisteix en tres paràmetres diferents: el proporcional, l'integral i el derivatiu. El Proporcional depèn dels errors actuals, l'Integral depèn dels errors passats i el Derivatiu és una predicció dels errors futurs.

Antiwindup: fa referència a la situació dins d'un controlador PID amb realimentació on es produeix un gran canvi en la situació d'ajust i els termes integrals acumulen un error significatiu durant aquest augment (Windup). La seva tasca es revertir aquest efecte no desitjat.

2. Prefaci

2.1. Context del projecte

Un dels problemes més rellevants que pateixen les grans ciutats i les carreteres de circumval·lació que les envolten, és la gran congestió de vehicles que s'hi produeix. El fet de que el trànsit es col·lapsi repercuteix negativament en diversos aspectes de la societat: augmenten considerablement les emissions de gasos de efecte hivernacle, conductors i passatgers perden moltes hores amb el que es considera una activitat no productiva (cost d'oportunitat), afecta a la salut dels conductors, etc.

Tot i que és evident que el factor més important és la gran quantitat de conductors, un altre factor rellevant és la poca coordinació que hi ha entre ells.

En la actualitat s'intenta combatre el gran problema del trànsit des de varies perspectives, ja sigui mirar de buscar alternatives de transport més eficients, com mirar de canviar els hàbits de conducció de la població. La simulació i recreació de situacions reals de trànsit és una altra manera que pot ajudar a desencallar la situació. Recrear el comportament humà mitjançant simulacions permet analitzar quins són les variables que estan afectant amb més notorietat. Un cop se sap quins són els comportaments que no afavoreixen la fluïdesa del trànsit es poden trobar algorismes de control de moviment els quals, gràcies al previ estudi, es millora la mobilitat conjunta dels vehicles. No obstant, és molt difícil canviar els hàbits de conducció de les persones, és aquí on entra en joc la conducció autònoma.

La conducció autònoma consisteix en vehicles capaços de funcionar sense la intervenció humana. Aquests porten incorporats algorismes de control de moviment per poder funcionar.

Així doncs en aquest projecte es desenvoluparà una plataforma de simulació que per una banda permeti recrear situacions de trànsit real i, per un altra, es puguin implementar algorismes que puguin servir pels cotxes autònoms.

2.2. Origen del projecte

Actualment des de l'Institut d'Organització i Control de Sistemes Industrials (IOC) de la Universitat Politècnica de Catalunya (UPC), està en marxa un projecte de creació de robots seguidors de línia i cada any es van afegint diverses funcionalitats als robots. Fins ara ja s'han dissenyat i posat en marxa un parell d'aquests robots i s'han afegit funcionalitats com la comunicació amb un ordinador pel qual es pot controlar el robot i donar un seguit d'ordres.

Aquest projecte està orientat a la creació i implementació de models de seguiment de vehicles exclusivament pels robots. D'aquesta manera seran capaços de seguir una línia i en cas de trobar un robot al davant poder adaptar-se a la seva velocitat sense col·lisionar.

Un altre dels objectius és crear una plataforma de simulació on es pugui observar la creació de congestions degudes al comportament humà. De certa manera aquest nou enfoc engloba l'inicial ja que per arribar a aquest punt cal prèviament modelitzar el control dels vehicles i aquests models es podran fer servir més endavant pels robots.

2.3. Antecedents

De cara a plataformes de simulació, la que proporciona la pàgina web 'Traffic-Simulation.de' [1] ha estat una clara referència. En la pàgina es permeten recrear diverses situacions de tràfic on es creen colls d'ampolla (carreteres interurbanes, incorporacions de carril, reducció de carrils...). La documentació és molt completa i de gran qualitat i per això s'han agafat idees pel model d'aquest projecte.

3. Introducció

3.1. Objectius del projecte

L'objectiu principal d'aquest projecte és la creació d'una plataforma de simulació de la fluència del trànsit. Aquesta es vol que es concebi de tal manera que els paràmetres que caracteritzin el sistema puguin ser modificats fàcilment. També es vol que no només s'obtinguin resultats numèrics del que pugui estar succeint sinó que també sigui una eina visual, és a dir, que els vehicles es representin gràficament.

Amb la plataforma de simulació creada, la fita a assolir és la de reproduir la fluïdesa del trànsit degut al comportament humà, observar la creació de congestions i veure com es van propagant al llarg del temps. També es té com a objectiu incloure un algoritme de control de moviment per ser implementat en vehicles autònoms i comparar-lo amb el comportament humà.

Avui en dia alhora de fer una nova infraestructura en carreteres és molt habitual la utilització de simulacions per observar possibles defectes d'aquesta i si hi ha algun element que suposi un coll d'ampolla que pugui taponar el trànsit. Per això també es vol recrear un cas en el que, apart de la pròpia descoordinació dels conductors, les característiques del circuit suposin un empitjorament del trànsit.

3.2. Abast del projecte

L'abast del projecte comprèn la creació d'una plataforma de simulació de la fluència del trànsit. En aquesta es podrà escollir si es vol que l'estudi sigui en el pla o amb alçada (l'estudi en tres dimensions permet incloure variacions de inclinació al llarg del circuit). Després també s'inclou un model mecànic dels vehicles simplificat que limitarà la seva actuació per poder obtenir resultats realistes.

Per part dels vehicles només es contemplarà el control de seguiment a un vehicle que estigui situat més endavant. Per tant, no s'inclourà la possibilitat d'avançament ni canvis de carril. Degut a aquesta circumstància el circuit disposarà d'un únic carril de circulació.

Respecte a l'origen d'aquest projecte, el qual consistia en provar algorismes de control de seguiment per poder implementar-los en els robots que la universitat està desenvolupant, no es contempla arribar a implementar-los en els aparells. Aquesta tasca s'haurà de dur a terme en futurs treballs.

4. Model matemàtic

En aquest apartat es mostren les equacions matemàtiques que regeixen el comportament dels vehicles. Per una banda es troben els algorismes de seguiment que descriuran de quina manera avançarà cada vehicle i , per una altra, les equacions físiques o limitacions pròpies dels vehicles.

4.1. Models de conducció

Els models de conducció tractats a continuació consisteixen en algorismes que en aquest cas es centren únicament en seguir al vehicle situat al davant. Aquests seran funció de un seguit de paràmetres com són la velocitat pròpia del vehicle, la velocitat del vehicle a seguir, distància entre ambdós, acceleració màxima, etc. Depenent del nombre de paràmetres que intervinguin dins l'algorisme es considerarà que és més o menys complexa.

Com ja s'ha dit amb anterioritat la plataforma de simulació està dissenyada per poder afegir fàcilment qualsevol model de seguiment. A continuació examinarem dos models de tipus microscòpic, no obstant, es podrien afegir-ne d'altres en futurs treballs.

4.1.1. Model Senzill

S'ha escollit aquest nom per la seva pròpia naturalesa. El model senzill és un model de seguiment que descriu la dinàmica de moviment de un vehicle k seguint l'expressió següent:

$$v_k = k_p \cdot (s_k - s_{ref}) + v_{k+1} \quad (\text{Eq. 4.1.})$$

On v_k és la velocitat del vehicle, v_{k+1} la velocitat del vehicle situat al seu davant, k_p és una constant de proporcionalitat, s_k és la distància que hi ha entre els dos vehicles i s_{ref} és una distància de referència, seguiment o seguretat i aquesta es farà més gran a raó de v_k :

$$s_{ref} = s_0 + T \cdot v_k \quad \text{On } T: \text{ constata de proporcionalitat} \quad (\text{Eq. 4.2.})$$

s_0 : Distància mínima

El que fa aquest algorisme és adequar la velocitat a la del vehicle davanter i situar-se a la distància de referència. Per tant, si la distància real és més gran a la de referència el vehicle anirà accelerant i, al apropar-se, desaccelerarà fins que $v_k = v_{k+1}$ i $s_k = s_{ref}$. La constant k_p marca amb quina rapidesa es vol apropar-se al vehicle següent. Per últim cal afegir una limitació, doncs pot succeir que el vehicle de davant circuli a una velocitat major a la desitjada pel conductor del vehicle k que anomenem V_0 :

$$v_k = \max(k_p \cdot (s_k - s_{ref}) + v_{k+1}, V_0) \quad (\text{Eq. 4.3.})$$

4.1.2. Model IDM

El model IDM, '*Intelligent Driver Model*', és un model de caire determinista desenvolupat per en Treiber, en Hennecke i en Helbing l'any 2000 [2] amb l'objectiu de millorar els resultats provinents d'altres models intel·ligents de conducció.

La dinàmica de un vehicle k circulant a velocitat v_k , a una distància s_k al següent vehicle i que segueix aquest model, ve descrita per la següent equació diferencial:

$$\dot{v}_k = \frac{dv_k}{dt} = a \left(1 - \left(\frac{v_k}{V_0} \right)^\delta - \left(\frac{s^*(v_k, \Delta v_k)}{s_i} \right)^2 \right) \quad (\text{Eq. 4.4.})$$

On s^* és la distància ideal que hauria d'haver entre els dos vehicles i té en compte paràmetres com la diferència de velocitats $\Delta v_k = v_k - v_{k+1}$. Es calcula amb l'expressió:

$$s^*(v_k, \Delta v_k) = s_0 + \max \left(v_k T + \frac{v_k \Delta v_k}{2\sqrt{ab}}, 0 \right) \quad (\text{Eq. 4.5.})$$

V_0 , s_0 , T , a i b són paràmetres propis del model i tenen el significat següent:

- Velocitat desitjada V_0 : la velocitat a la que el vehicle circularia sense altres vehicles.
- Espai mínim s_0 : distància mínima a la que un vehicle es situa darrere d'un altre.
- Temps desitjat de seguiment T : temps de seguretat al vehicle en front que en una situació de trànsit fluid et permet frenar sense col·lisionar.
- Acceleració a : màxima acceleració del vehicle.
- Desacceleració confortable b : restricció de desacceleració confortable per a una persona.

L'exponent δ es fixa a un valor entre 1 i 5.

Valors estàndards dels paràmetres i que utilitzarem en les simulacions són per exemple:

| Paràmetre | Valors per un cotxe | Valors per a un camió | Observacions |
|---------------------------------|---------------------|-----------------------|---|
| Velocitat desitjada V_0 | 120 km/h | 80 km/h | En funció del límit de velocitat variarà, en canvi la resta de paràmetres es mantenen constants fins i tot en carreteres urbanes. |
| Temps desitjat de seguiment T | 1.5 s | 1.7 s | Aquest paràmetre pot variar molt en funció del conductor. Es pot situar entre 0.8 i 2 segons o fins i tot menys. |

| | | | |
|--------------------------------------|------------------------|----------------------|--|
| So Espai mínim | 2.0 m | 2.0 m | Distància que només s'haurà d'assolir en cas d'estar aturat. |
| a Acceleració | 1-1.5 m/s ² | 0.6 m/s ² | Valors molt baixos afavoreixen la formació del trànsit 'stop and go' (parada i posada en marxa) |
| b Desceleració confortable | 3.0 m/s ² | 2.0 m/s ² | Valors molt elevats afavoreixen la formació del trànsit 'stop and go' (parada i posada en marxa) |

Figura 4-1: Taula de valors dels paràmetres típics del model IDM

Donat que l'equació depèn de molts factors, es fa força complicat entendre com funciona a primera vista, per tant, analitzarem algunes de les seves propietats.

En primer lloc l'expressió està formada per dues parts. La primera correspon al terme $a \left(1 - \left(\frac{v_k}{V_0}\right)^\delta\right)$. Aquest s'encarrega de proporcionar l'acceleració necessària per assolir la velocitat desitjada. En el cas d'una carretera sense trànsit l'expressió es simplifica únicament a aquest valor. Al aproximar-se a dita velocitat el terme es va reduint fins a zero.

La resta de l'expressió és un terme de desacceleració que s'encarrega de proporcionar la frenada degut al vehicle situat en front i ho fa comparant la distància real amb la distància ideal. En cas de que la distància real s'aproximi a la distància ideal, el terme compensa la part de acceleració lliure i la acceleració resultant és propera a zero. Com a conseqüència de la seva definició, la desacceleració augmenta quan:

- Disminueix la distància al vehicle de davant per mantenir la distància de seguretat.
- Al augmentar la pròpia velocitat, si s'està seguint un vehicle, la distància de seguretat també augmentarà.
- Al apropar-se a un vehicle a una velocitat superior a la seva, és a dir, la diferència de velocitats és elevada.

4.2. Model mecànic dels vehicles

El programa permet fer les simulacions sense tenir en compte cap limitació pel que fa al moviment dels vehicles. No obstant, si vol fer un estudi rigorós és important considerar que no sempre el vehicle podrà arribar a assolir la velocitat de referència que li marca l'algorisme en qüestió.

Donat que la finalitat del projecte es centra majoritàriament en el control dels vehicles, el model mecànic implementat és força simplificat. No es consideren ni distribucions de pes sobre cada eix de les rodes, ni si els vehicles són de tracció davantera o posterior, si alhora de reduir la velocitat hi ha un repartiment de la frenada a cada eix, etc. Bàsicament es pretén controlar l'acceleració i desacceleració màxima dintre d'uns marges raonables.

4.2.1. Força de tracció

La força de tracció està delimitada per dos factors. Per una banda, per la potència màxima del vehicle. On l'equació que defineix la potència és:

$$P_{vehicle} = F_{tracció} \cdot v_{vehicle} \quad (\text{Eq. 4.6.})$$

I per una altra, per la força màxima que pot realitzar el contacte roda-terra:

$$F_{màx} = \mu_{terra} \cdot N_{vehicle} \quad (\text{Eq. 4.7.})$$

On μ correspon al coeficient de fricció del terra i N a la força normal del vehicle. La força normal dependrà així mateix de la inclinació del terra.

Si ajuntem ambdues condicions:

$$F_{real_màx} = \min(\mu_{terra} \cdot N_{vehicle}, P_{vehicle}/v_{vehicle}) \quad (\text{Eq. 4.8.})$$

A mode d'exemple s'ha fet una simulació de un vehicle per observar la corba de tracció. Se li ha donat una consigna de velocitat 200 km/h i parteix de velocitat nul·la, és a dir que treballarà a l'acceleració més gran possible per arribar a la velocitat de 200 km/h.

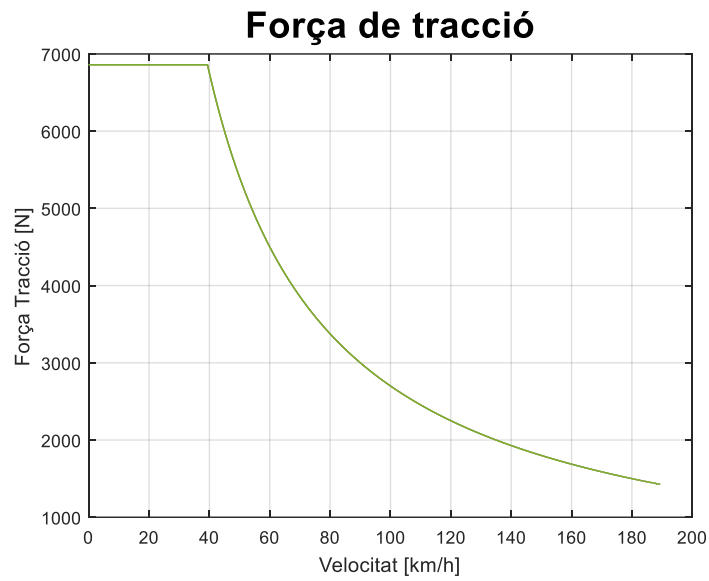


Figura 4-2: Corba força tracció màxima

S'observa com a velocitats baixes la limitació de tracció és deguda al contacte amb al terra i a velocitats majors la potencia màxima és el factor limitant.

4.2.2. Forces resistents

La força resistent que s'oposa al moviment del vehicle es pot desglossar en diverses parts. A la figura següent es mostra el diagrama del cos lliure de un vehicle on es veuen totes les forces que actuen al sistema:

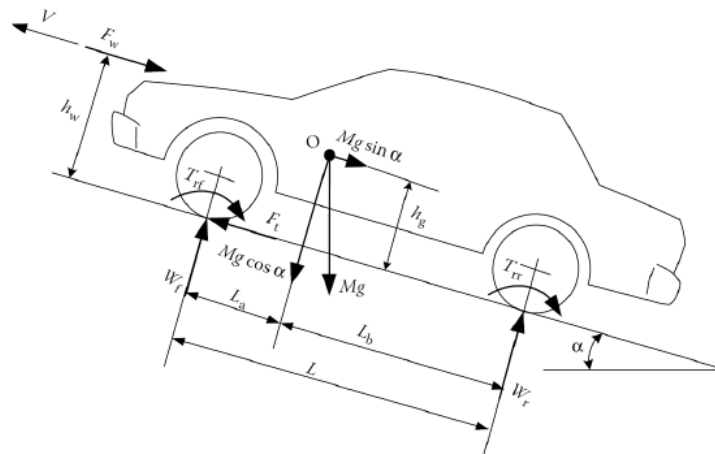


Figura 4-3: Diagrama del cos lliure de un vehicle

Força aerodinàmica, F_a :

És la força deguda a la fricció de l'aire amb el perfil del vehicle. A velocitats baixes serà poc rellevant però a velocitats altes augmentarà enormement ja que depèn de la velocitat del vehicle elevada al cub. El seu valor es troba a partir de l'expressió:

$$F_a = \frac{1}{2} \cdot \rho \cdot A_f \cdot c_d \cdot (v_{vehicle} - v_{vent})^3 \quad (\text{Eq. 4.9.})$$

On ρ és la densitat de l'aire, A_f és l'àrea frontal del vehicle, c_d correspon al coeficient aerodinàmic de resistència, $v_{vehicle}$ la velocitat del vehicle i v_{vent} la velocitat del vent.

Força rodolament, F_r :

El contacte entre la roda i el terra no és un contacte puntual sinó una petita superfície. Degut a que la resultant de la força normal no està situada just a sota de l'eix de la roda, es crea un moment que s'oposa al moviment. Aquesta força depèn molt de la pressió dels pneumàtics. L'expressió que la defineix és:

$$F_r = f_r \cdot M \cdot g \cdot \cos(\alpha) \quad (\text{Eq. 4.10.})$$

On f_r és el coeficient de fricció de rodolament, M la massa del vehicle, g la constant de la força de la gravetat i α la inclinació del terra.

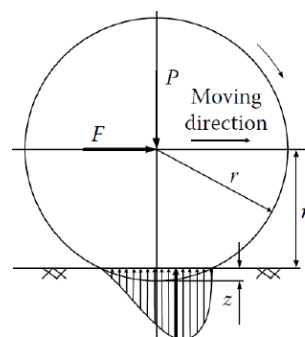


Figura 4-4: Distribució de la força normal

Força de gravetat, F_g :

Quan hi ha cert desnivell el propi pes del vehicle en les pujades s'oposarà al moviment i a les baixades la força anirà en la mateixa direcció al moviment:

$$F_g = M \cdot g \cdot \sin(\alpha) \quad (\text{Eq. 4.11.})$$

4.2.3. Velocitat real

Un cop trobades les forces que actuen al sistema podem calcular la velocitat del vehicle a partir de la segona llei de Newton:

$$\frac{dv}{dt} \cdot M = F_t - (F_a + F_r + F_g) \quad (\text{Eq. 4.12.})$$

De fet obtenim l'acceleració però, gràcies a les eines que *Simulink* ens proporciona, obtenim la derivada fàcilment que correspon a la velocitat.

5. Model del circuit

Primer de tot s'ha de tenir en ment que es pretén simular el comportament de la fluïdesa del trànsit només en zones no urbanes. En aquest escenari les carreteres no tenen interseccions, semàfors, etc. i les accions dels conductors són reduïdes.

Per a aquests cas és innecessari crear una xarxa complexa de carreteres per on circulin els vehicles. Hi ha dos solucions possibles:

- Crear un circuit obert on es definirà una densitat de vehicles per quilòmetre. A l'inici del circuit aniran apareixent vehicles segons el flux definit i al arribar al final desapareixeran, per tant, els vehicles es van renovant.
- Crear un circuit tancat on durant tota la simulació els vehicles seran els mateixos i aquests van donant voltes consecutivament al circuit.

Per simplicitat es va triar la segona opció.

Pel que fa a la forma del circuit qualsevol hauria estat vàlida, tanmateix, per simplicitat s'ha escollit la forma de circumferència.

5.1. Model en el pla

Les equacions paramètriques del model en el pla són:

$$(x, y) = (R \cdot \cos(\theta), R \cdot \sin(\theta)) \quad \text{on } \theta \in [0, 2\pi] \quad (\text{Eq. 5.1.})$$

On R és el radi de la circumferència del circuit i θ és l'angle.

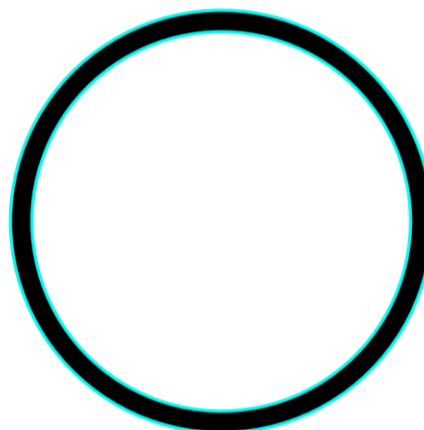


Figura 5-1: Circuit en el pla

5.2. Model amb alçada

Un cop acabat el model en el pla es va decidir fer el mateix en 3 dimensions. Apart del canvi visual dels vehicles i del circuit, el model amb alçada permet afegir canvis de pendents apropant-nos encara més a la realitat. El canvi de gradient del terreny pot arribar a traduir-se en la creació d'un coll d'ampolla, és a dir, d'un decreixement local de la capacitat de la carretera.

Per duu a terme la creació del circuit s'han conservat les coordenades x i y i s'ha modificat la coordenada z que ara serà funció de l'angle. Per aquest objectiu s'ha considerat que la funció senoïdal és ideal ja que permet introduir desnivells periòdicament de manera senzilla i sense haver de fer servir funcions a trossos.

Les equacions paramètriques del model amb alçada són:

$$(x, y, z) = \left(R \cdot \cos(\theta), R \cdot \sin(\theta), \frac{R}{2} + \sin(a * \theta) * \frac{R}{6} \right) \quad \text{on } \theta \in [0, 2\pi] \quad (\text{Eq.5.2})$$

On R és el radi de la circumferència del circuit, θ és l'angle i a és el nombre de pics al circuit. El component $R/2$ s'ha afegit perquè la coordenada z sigui sempre positiva i el factor $R/6$ per suavitzar les pendents.

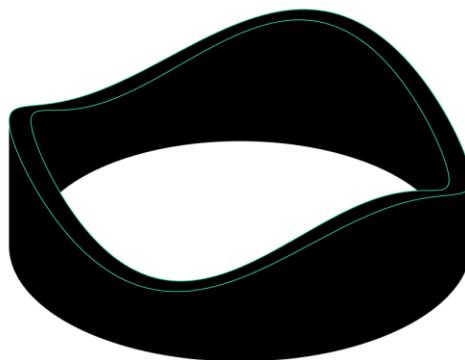


Figura 5-2: Circuit amb alçada amb 3 pics

6. Implementació del model

Per implementar les equacions del model i representar gràficament la simulació calia trobar un programa que complís satisfactòriament els requisits. Donat que en el model trobem equacions diferencials i integrals de les quals algunes resulten no-lineals i difícils de tractar analíticament, el programa ha de disposar d'eines capaces de realitzar integració numèrica.

De tots els programes disponibles s'ha triat *Matlab*. És un programa de gran capacitat de càlcul i molt utilitzat a arreu del món, ja sigui per recerca a universitats o desenvolupar productes dins d'empreses. La gran documentació i comunitat online ha fet molt fàcil aprendre els mètodes necessaris per duu a terme el projecte i trobar solucions als problemes que han anat sorgit al llarg d'aquest.

A més a més, el programa *Matlab* incorpora una plataforma de simulació anomenada *Simulink* ideal pel nostre propòsit. Simulink resulta una eina molt interessant per a la simulació de sistemes dinàmics. El seu entorn de programació visual a partir de blocs presenta un nivell d'abstracció superior al del llenguatge de Matlab, permetent implementar les equacions d'un sistema qualsevol d'una manera senzilla i intuïtiva sense requerir un alt nivell de programació.

6.1. Estructura general del programa/sistema

En aquest apartat s'explicarà quin es el procediment per duu a terme una simulació. Es presentaran els fitxers més rellevants, en quin ordre s'utilitzen i com interaccionen entre ells per compartir la informació. En els apartats 6.2 i 6.3 es veurà amb més detall cada un d'ells (tots els fitxer es troben a l'Annex B).

En total es tenen o creen cinc tipus diferents de fitxers:

- Fitxer d'extensió ``.m'` de tipus 'Function': són fitxers pensats únicament per escriure-hi funcions i, en general, al fer una crida del fitxer només es podrà accedir a la 1a funció en cas de que n'hi hagués més d'una.
- Fitxers d'extensió ``.m'` de tipus 'Script': està pensat per introduir un seguit d'instruccions per generar un programa. No permet declarar funcions però si fer-ne crida d'elles.
- Fitxer d'extensió ``.mat'`: conjunt de cel·les on es pot emmagatzemar informació.
- Fitxer d'extensió ``.mp4'`: es genera al final de cada simulació i consisteix en un vídeo que mostra l'evolució dels vehicles juntament amb un parell de gràfics.
- Fitxer d'extensió ``.slx'`: es un fitxer de la plataforma Simulink.

El primer que s'ha de fer es obrir el fitxer ``Input_Parameters.m'`. Aquest fitxer conté tots

els paràmetres que afecten a la simulació (radi del circuit, durada de la simulació, nombre de vehicles, característiques d'aquests, etc.). Un cop es té definit el tipus de simulació que es vol fer, s'ha d'obrir el fitxer 'Traffic_flow_simulation_2D.m', si es vol fer una simulació en el pla, o el fitxer 'Traffic_flow_simulation_3D.m', si es vol fer una simulació amb alçada. A continuació s'han d'executar pitjant la tecla 'F5' o pitjant el botó 'Run'.

Aquests fitxers fan una crida als paràmetres definits anteriorment i amb aquests es dona la ordre d'executar el model de Simulink. El fitxer de Simulink, 'Schematic_model_of_operation.slx', genera una matriu amb totes les posicions dels vehicles al llarg del temps. Amb aquesta informació es procedeix a la creació del vídeo que conté l'animació dels vehicles i dos gràfics.

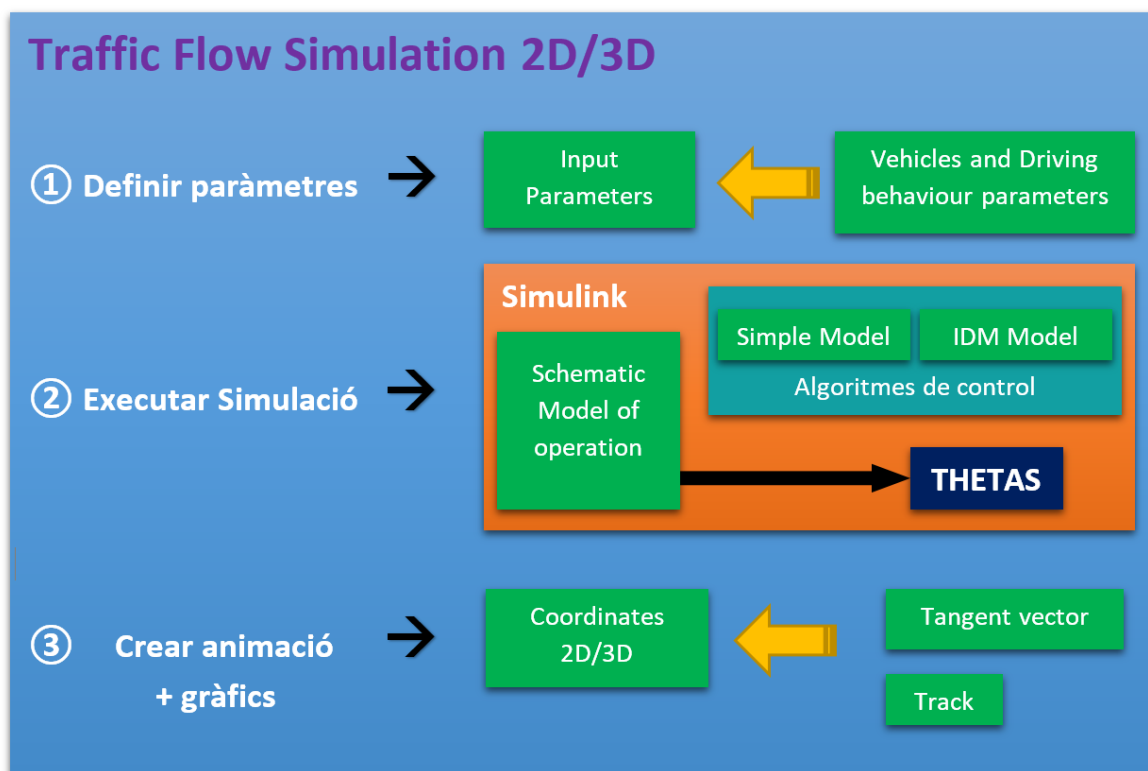


Figura 6-1: Esquema intern de funcionament

6.2. Simulink

Com s'ha esmentat a l'apartat anterior, el fitxer de Simulink 'Schematic_model_of_operation.slx' s'encarrega de calcular la posició de cada vehicle per cada instant de temps, és a dir, en quin angle de la circumferència es troben.

En el fitxer 'Input_parameters.m' s'ha definit prèviament la durada de la simulació i el pas d'integració desitjat. Normalment interessa que aquest sigui el més gran possible per reduir el temps de càlcul.

El sistema de blocs, que es troba a l'annex A, funciona seguint els passos següents:

- 1) Al primer instant, $T = 0$ s, tant la velocitat real com posició dels vehicles són condicions inicials establertes al fitxer 'Input_parameters.m'.
- 2) Coneixent la velocitat dels vehicles i la posició actual, mitjançant un bloc integrador de la llibreria de Simulink, s'obté la posició dels vehicles del següent instant de temps. Aquesta queda enregistrada a la matriu 'Thetas'. La matriu 'Thetas' consisteix en una matriu on cada columna representa un vehicle i cada fila la posició que ocupa en un instant de temps.
 - a. En el cas de fer un estudi 3D es fa una conversió de la velocitat lineal dels vehicles a una velocitat horitzontal ja que més endavant el propi model introdueix la velocitat vertical.
- 3) Un cop tots els vehicles estan posicionats i tenen una velocitat associada, es calculen els paràmetres següents: la distància entre vehicles i la inclinació del circuit a cada posició on s'hi troba un vehicle (la inclinació s'usa només en el cas 3D).
- 4) Juntament amb aquests dos paràmetres i d'altres que són constats durant la simulació (per saber el seu valor se'ls fa una crida al fitxer 'Input_parameters.m') el sistema queda caracteritzat. Amb tota aquesta informació l'algoritme seleccionat troba l'acceleració o velocitat de referència per al nou instant de temps per a cada vehicle.
 - a. En el cas de ser un algoritme que troba l'acceleració, s'usa un integrador per trobar la velocitat de referència.
- 5) Un cop trobada la velocitat de referència, si es fa un estudi que té en compte limitacions dels vehicles, es troba la velocitat real que poden assolir.
- 6) El següent pas seria tornar al pas 2) i tornar a trobar la posició del següent instant de temps.

Successivament es van repetint aquests passos afegint la posició de cada vehicle a la matriu 'Thetas' per a cada iteració fins que s'arribi al temps final de la simulació.

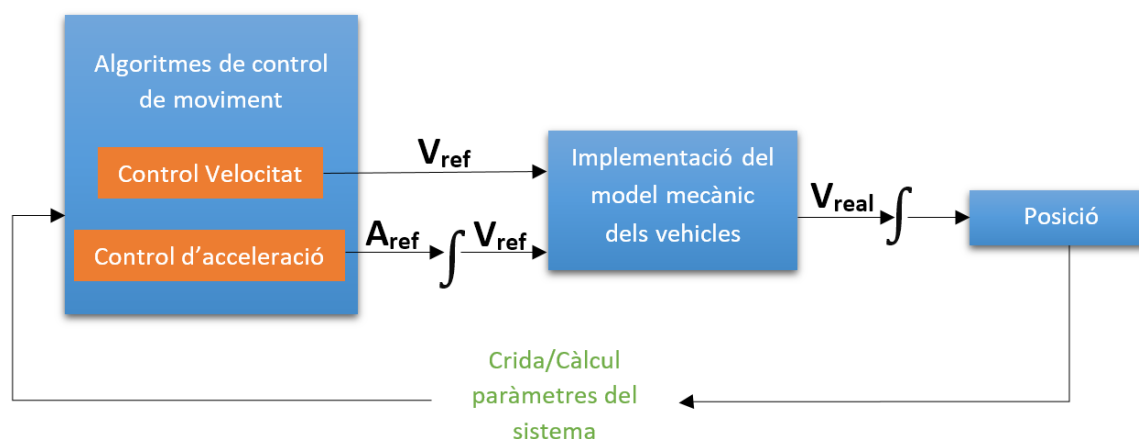


Figura 6-2: Esquema conceptual del fitxer de Simulink

Per últim comentar que en l'esquema de Simulink s'han disposat diversos blocs interruptors. Permeten seleccionar si es vol fer una simulació en 2D o 3D, seleccionar si es vol afegir el model mecànic dels vehicles i quin algoritme de control de moviment es desitja fer servir. Gràcies a la introducció d'aquests blocs es pot agrupar diversos casos en un sol esquema sense necessitat de fer-ne un diferent per cada tipus d'estudi.

6.2.1. Implementació del model mecànic

Per limitar la velocitat en funció de les característiques dels vehicles s'usen un seguit de blocs agrupats en un de sol a l'esquema general. Així mateix, aquest bloc s'organitza en 4 subconjunts (il·lustrats a l'annex A.2.).

El primer subconjunt consisteix en un controlador i es pot entendre des de dos punts de vista:

1. El vehicle és controlat per una persona, és a dir, el conductor en funció de l'entorn intueix com ha d'actuar. Si veu que la velocitat actual és més petita a la que creu adequada pitjarà el pedal d'acceleració, per tant, augmentarà la força de tracció i conseqüentment la velocitat. En cas contrari, si veu que ha de reduir la velocitat, aixecaria el peu de l'accelerador i si fos necessari utilitzaria el pedal del fre.
2. El segon punt de vista consistiria en un vehicle autònom que regularia per si mateix la seva velocitat. El procediment és semblant al cas 1. però en aquest cas per saber què fer utilitzaria sensors per captar informació de l'entorn.

Si es volgués aprofundir més i provar de recrear ambdós comportaments, humà i autònom, alhora d'accelerar es podria anar jugant amb les constants del controlador. Per il·lustrar aquest fet s'ha realitzat un exemple:

Suposem que un vehicle està en una carretera limitada a 90 Km/h i arribat a un punt el límit

puja fins als 120 Km/h. Utilitzant el model amb un controlador PI (proporcional i integral) s'han modificat les constants per veure com s'adeqüen en cada cas al canvi de velocitat.

- En el primer cas, la línia blava de la Figura 6-3, correspondria a un conductor que tendeix a accelerar bruscament de tal manera que sobrepassa el límit de velocitat i ha de corregir la velocitat.
- En el segon cas, la línia verda, es podria tractar d'un cotxe amb ACC (Adaptive Cruise Control) incorporat el qual controla la velocitat del cotxe. Aquest seria capaç de fer servir la potencia màxima del cotxe sense sobrepassar la velocitat de referència de 120 Km/h.
- L'últim cas podria ser el comportament d'un conductor que circula amb més precaució que el primer i accelera amb més moderació fins a arribar a la velocitat de 120 Km/h sense necessitat de rectificar.

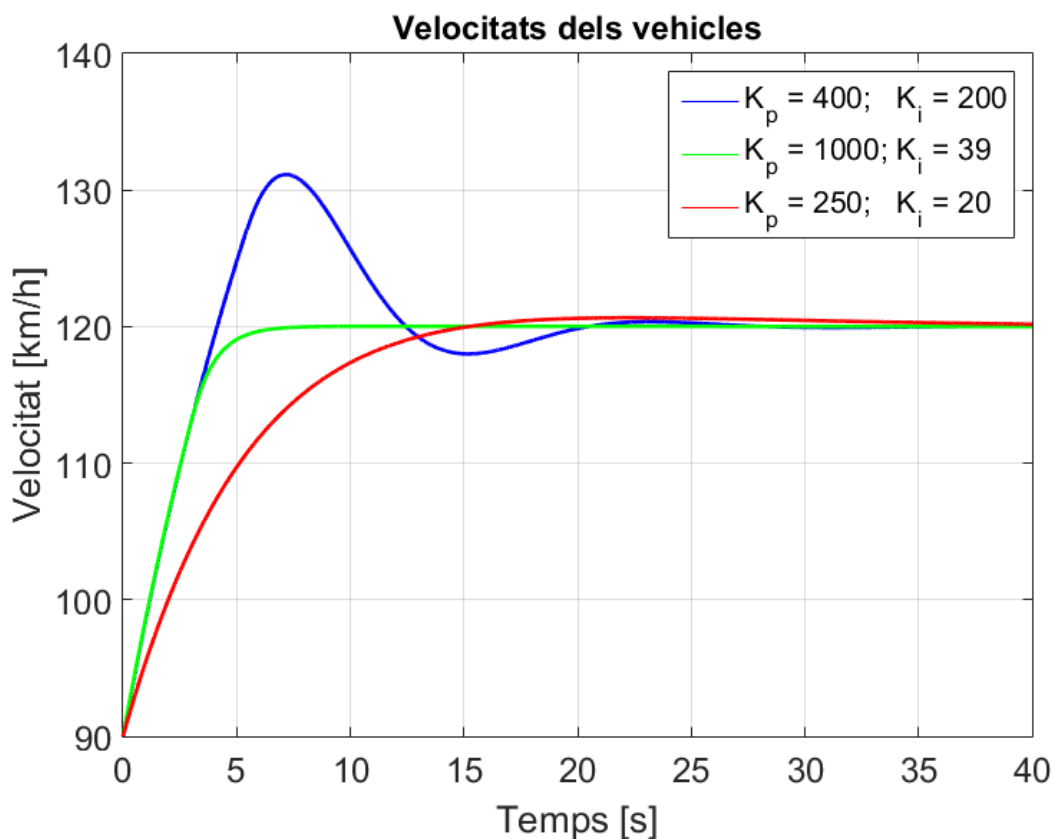


Figura 6-3: Gràfic adaptació de velocitat

Per una banda es va pensar de fer un controlador únicament proporcional. El problema és que per poder funcionar correctament necessita la realimentació del valor de la força resistent. Aquest valor evidentment un conductor no el pot conèixer i un cotxe autònom li podria resultar

difícil de calcular. La solució és afegir la part integral. Aquesta assoleix el valor de la força resistent un cop s'arriba a la velocitat consigna com s'observa a la Figura 6-4:

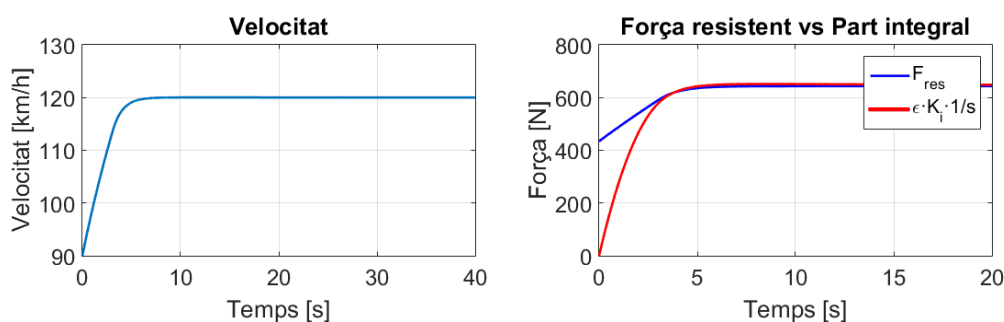


Figura 6-4: Efecte de la part integral del controlador

Donat que en aquest projecte es fa èmfasi en com obtenir la velocitat de referència, s'implementa un controlador PI de resposta ràpida. Per triar les K_p i K_i agafem els valors proposats a [4].

Els tres següents subconjunts de blocs tracten d'imitar el comportament físic del món real. Implementant les equacions del apartat 4.2., primer es limita la força i , amb aquesta juntament amb la força resistent, es calcula la velocitat real.

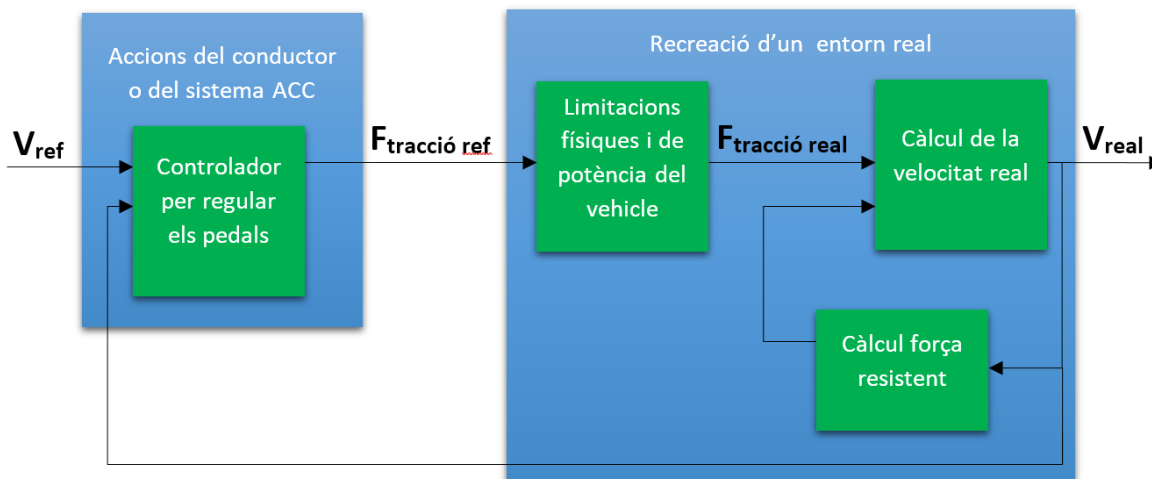


Figura 6-5: Esquema conceptual de la implementació del model mecànic

6.2.2. Integral Anti-Windup

L'Integral Anti-Windup és un mecanisme molt utilitzat als controladors PID amb realimentació. La seva implementació és necessària quan el senyal de sortida es satura, en general per limitacions físiques. En el temps en què la senyal ha estat saturada, l'integrador ha acumulat un error que provoca un retard de la resposta. La tasca de l'Anti-Windup és contrarestar l'efecte de l'integrador i no permetre que l'error vagi augmentant. L'Anti-Windup implementat en aquest model és el següent:

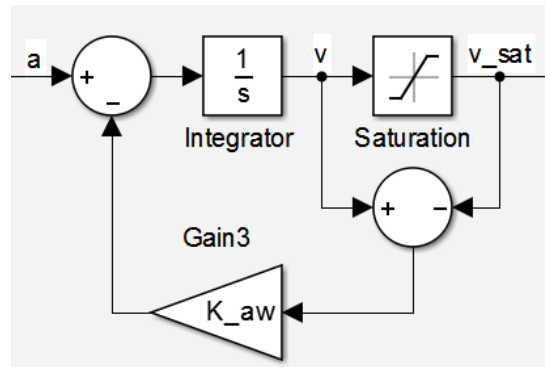


Figura 6-6: Integral Anti-Windup del model

Concretament ha calgut afegir-lo pel fet de fer servir un model mecànic per a cada vehicle. En situacions on els vehicles han de realitzar una forta frenada i aturar-se. Per les limitacions imposades no són capaços de fer la frenada de tal manera que al assolir la velocitat nul·la es mantinguin la distància mínima desitjada. Com que l'algorisme IDM obliga a situar-se a dita separació dóna un ordre d'acceleració negativa i per tant de retrocedir. Com que a la realitat mai un cotxe va marxa enrere per corregir aquesta situació, s'ha saturat inferiorment la velocitat a zero. En cas de no introduir l'Anti-Windup, el vehicle va acumulant error i es tradueix en una espera innecessària quan el vehicle de davant ja fa estona que ha marxat. En la Figura 6-7 i 6-8 es pot observar la diferència en una simulació que no utilitza l'Anti-Windup i la mateixa on si que ho fa.

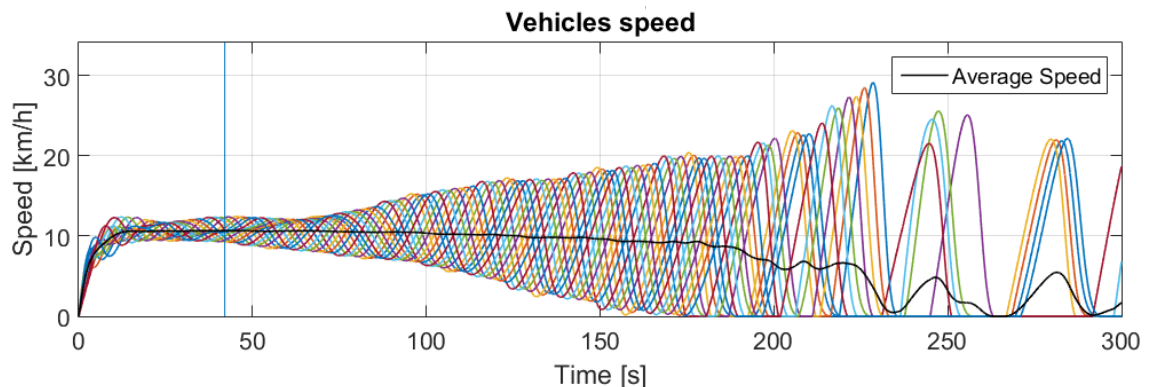


Figura 6-7: Simulació que no incorpora Anti-Windup

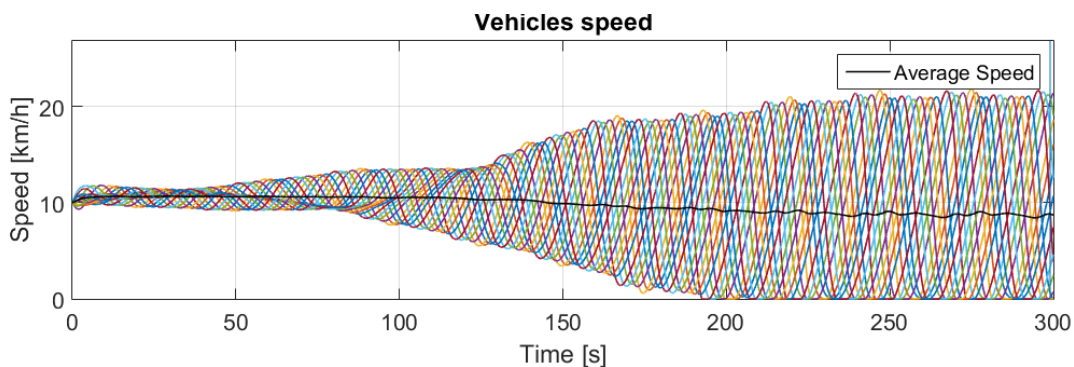


Figura 6-8: Simulació que incorpora Anti-Windup

Per veure de forma més aclaridora el que està succeint, a la Figura 6-9 es gràfica: la distància al vehicle de davant, la velocitat i acceleració del vehicle número 7 d'aquesta simulació.

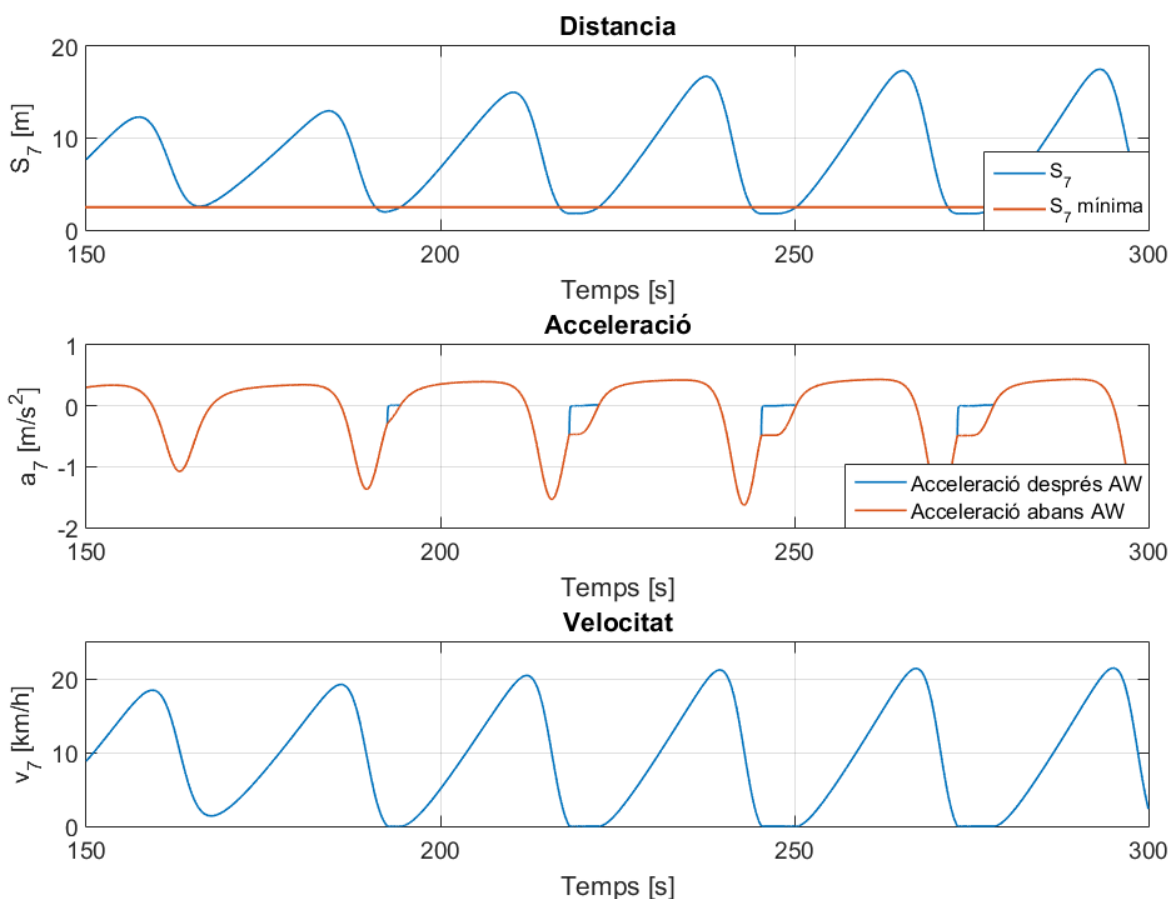


Figura 6-9: S_7 , v_7 i a_7 abans i després de l'Anti-Windup

S'observa que quan la velocitat arriba a zero i la distància és inferior a la distància mínima l'acceleració és negativa. En canvi, gracies a l'Anti-Windup es corregeix el seu valor a zero.

6.3. Matlab

Entre tots els fitxers utilitzats amb llenguatge Matlab, a continuació s'analitzen els que presenten certa complexitat o són necessaris de comprendre per entre el funcionament intern per arribar a fer una simulació. La resta de fitxers, juntament amb els que es tractin seguidament, estaran presents a l'annex B.

6.3.1. Càlcul de distàncies

La distància entre vehicles és un dels paràmetres que fan servir els algorismes de control de moviment. S'ha definit com la distància que ha de recórrer un vehicle des de el seu centre de masses fins al centre de masses del següent vehicle, tot seguint el circuit, menys la semi longitud d'ambdós vehicles.

En el model en el pla la distància no resulta cap problema obtenir-la. La separació de un vehicle al següent és igual a l'arc de circumferència entre tots dos, per tant, es calcula fent la diferència de posició angular dels dos vehicles pel radi del circuit. Com es força senzill fer aquesta operació, amb els blocs de Simulink és suficient. Per contra, al model 3D cal recórrer a la utilització d'un bloc que ens permet fer una crida a una funció escrita amb llenguatge Matlab.

El nom del fitxer és `'Length_curve.m'` (Longitud de corba) doncs la distància correspon a la longitud de la corba del circuit entre dos punts. Gràcies a les eines de Matlab s'implementa les formules de càlcul vectorial [4] que ens permeten obtenir la longitud d'una corba.

6.3.2. Posicionar vehicles

Com ja s'ha dit anteriorment a la matriu 'Thetas' s'emmagatzema angles entre $[0, 2\pi]$ que representen les posicions dels vehicles a una circumferència. Coneixent aquest angle i les equacions 5.1. i 5.2. que descriuen les trajectòries del circuit en 2D i 3D respectivament, es troben les coordenades del circuit sobre les quals s'han de representar els vehicles.

6.3.2.1. Cas dos dimensions

En el cas de les simulacions en el pla els vehicles són representats amb rectangles. Posicionar un vehicle significa doncs, conèixer els quatre vèrtexs d'aquest rectangle. S'obtenen amb els passos següents:

1. Coneixent la longitud i amplada del vehicle es busquen les coordenades dels vèrtexs centrant el rectangle a l'origen (0,0) i orientant-lo de manera que l'eix longitudinal del rectangle queda paral·lel amb l'eix de coordenades y .
2. Fent ús de les propietats geomètriques s'orienta el rectangle perquè el seu eix

longitudinal quedi paral·lel amb la tangent del circuit en el punt on s'ha de situar.

3. Es trasllada el rectangle orientat de tal manera que el centre del rectangle coincideix amb el punt del circuit corresponent.

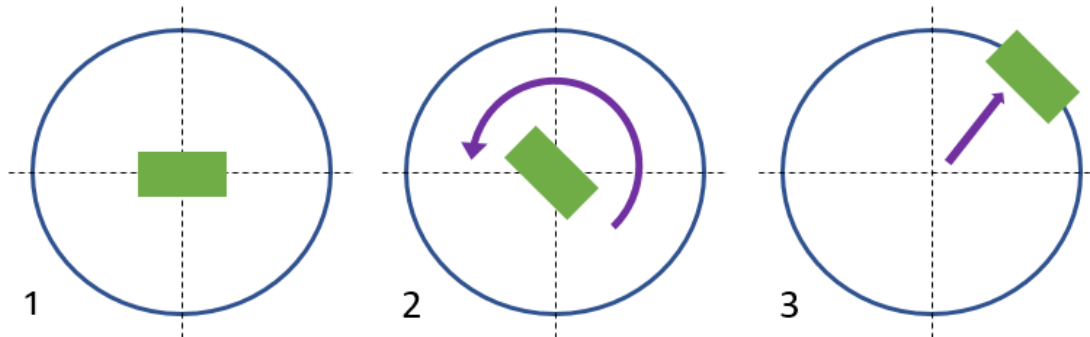


Figura 6-10: Passos per posicionar el rectangle en 2D

6.3.2.2. Cas tres dimensions

En el cas de les simulacions en tres dimensions els vehicles són representats amb paral·lelepèdres rectangulars. En aquest cas caldrà trobar vuit vèrtexs. L'estratègia utilitzada difereix a la que s'ha fet servir en el pla:

1. Trobar els vectors unitaris que defineixen les direccions normals a les cares del paral·lelepèdre.
2. Coneixent els vectors unitaris i les dimensions del paral·lelepèdre es troben els vèrtexs del vehicle centrat a l'origen (0,0,0).
3. Es trasllada de tal manera que el seu centre és coincident al punt del circuit corresponent.
4. S'afegeix la meitat de l'altura del vehicle i una certa distància de separació entre aquest i el circuit.

Per últim, tant en el cas 2D com en el 3D, fent ús del mètode de Matlab 'patch' es representen els vehicles.

6.4. Paràmetres

Per poder duu a terme una simulació cal definir un gran nombre de paràmetres. Tots ells estan presents al fitxer `'Input_parameters.m'`.

En primer lloc tenim paràmetres invariables que romanen constants a totes les simulacions com són: la densitat de l'aire, el coeficient de fricció del circuit, coeficients dels controladors

utilitzats, velocitat del vent, coeficient de rodament dels pneumàtics, etc.

En segon lloc tenim aquells paràmetres que en funció de l'estudi que es vol fer poden anar variant. Dins d'aquest grup trobem dos tipus:

- Paràmetres que cal modificar manualment: Duració de la simulació, radi del circuit, nombre de vehicles, velocitat inicial i velocitat mitja dels vehicles, etc.
- Paràmetres que es calculen de manera automàtica i aleatòria per cada simulació. Consisteixen en un seguit de variables que defineixen cada un dels vehicles i els perfils de conducció propis de cada un.

Aquests últims paràmetres, els quals es creen automàticament i de forma aleatòria, s'obtenen al fer una crida de la funció 'Vehicles_and_Driving_behaviour_parameters.m'. Prèviament al fitxer 'Input_paramerts.m' es defineixen una seguit de variables com es mostra a la figura següent:

```

8      % Creation of parameters involving driving profiles and vehicles
9      % characteristics
10 -   Number_of_Vehicles = 25;
11 -   Average_Speed = 60;           Desviation_Speed = 5;
12 -   Average_Minimum_Space = 2;   Desviation_Minimum_Space = 0.5;
13 -   Average_Time_Headway = 2;    Desviation_Time_Headway = 0.6;
14 -   Average_Vehicles_Acceleration = 1; Desviation_Acceleration = 0.1;
15 -   Average_Vehicles_Decelaretion = 4; Desviation_Vehicles_Deceleration = 0.2;
16
17 -   Par = Vehicles_and_Driving_behaviour_parameters(Number_of_Vehicles,...
18           Average_Speed,Desviation_Speed,Average_Minimum_Space,...
19           Desviation_Minimum_Space,Average_Time_Headway,Desviation_Time_Headway,..
20           Average_Vehicles_Acceleration,Desviation_Acceleration,...
21           Average_Vehicles_Decelaretion,Desviation_Vehicles_Deceleration,Radius);
22 -   save Par;

```

Figura 6-11: Part del codi on s'obtenen els paràmetres automàticament

Tenint en compte que per a cada vehicle es fixen fins a 12 paràmetres diferents, és necessari trobar una manera ordenada d'agrupar tots aquests valors. És per això que s'usa una matriu de nom 'Par' on s'emmagatzema tota aquesta informació. La distribució de la matriu 'Par' és:

| | | | | | |
|----|-----------|-----------|-----|-----------|--------------------------------------|
| | V_1 | V_2 | ... | V_n | |
| 1 | w_1 | w_2 | ... | w_n | → Amplada |
| 2 | l_1 | l_2 | ... | l_n | → Longitud |
| 3 | h_1 | h_2 | ... | h_n | → Altura |
| 4 | m_1 | m_2 | ... | m_n | → Massa |
| 5 | P_1 | P_2 | ... | P_n | → Potència |
| 6 | Pos_1 | Pos_2 | ... | Pos_n | → Posició inicial |
| 7 | V_1 | V_2 | ... | V_n | → Velocitat desitjada |
| 8 | S_{0-1} | S_{0-2} | ... | S_{0-n} | → Distància de separació |
| 9 | T_1 | T_2 | ... | T_n | → Temps desitjat de seguiment |
| 10 | a_1 | a_2 | ... | a_n | → Acceleració màxima |
| 11 | b_1 | b_2 | ... | b_n | → Desacceleració confortable |
| 12 | K_1 | K_2 | ... | K_n | → Tipus de vehicle |

Figura 6-12: Matriu 'Par'

6.4.1. Paràmetres propis dels vehicles

Estan constituïts per les dimensions, masses i potències dels vehicles.

Per arribar a l'objectiu principal del projecte de simular el tràfic real s'han definit diferents tipus de vehicles. En total s'han escollit cinc tipus diferents per representar els vehicles més freqüents que es troben a les carreteres: cotxe de baixes prestacions, cotxe d'altres prestacions, motocicletes, autobusos i camions. S'han establert valors representatius mostrats a la Figura 6-13.

| | Amplada [m] | Llargada [m] | Altura [m] | Massa [Kg] | Potència [Kw] |
|------------------------------------|--------------------|---------------------|-------------------|-------------------|----------------------|
| <i>Cotxe de baixes prestacions</i> | 1,7 | 3,9 | 1,4 | 1000 | 70 |
| <i>Cotxe d'altres prestacions</i> | 1,9 | 5 | 1,5 | 1.900 | 200 |
| <i>Motocicleta</i> | 0,8 | 2 | 1,1 | 180 | 70 |
| <i>Autobús</i> | 2,5 | 10 | 3 | 12.000 | 330 |
| <i>Camió</i> | 2,1 | 7 | 3,5 | 9.000 | 125 |

Figura 6-13: Taula de paràmetres propis dels vehicles

Cal tenir en compte la proporció de vehicles de cada tipus respecte al total. Dins del fitxer `'Vehicles_and_Driving_behaviour_parameters.m'` s'han establert uns rangs els quals manifesten les probabilitats de que cada cotxe que es crea sigui d'un tipus o d'un altre. En funció de l'estudi que es vulgui fer es pot modificar. Per exemple, hi ha carreteres on vehicles de grans dimensions com són els camions no se'ls permet el pas o carreteres on no hi circula cap línia d'autobús. En aquests casos per fer estudis del trànsit s'hauran de modificar els rangs.

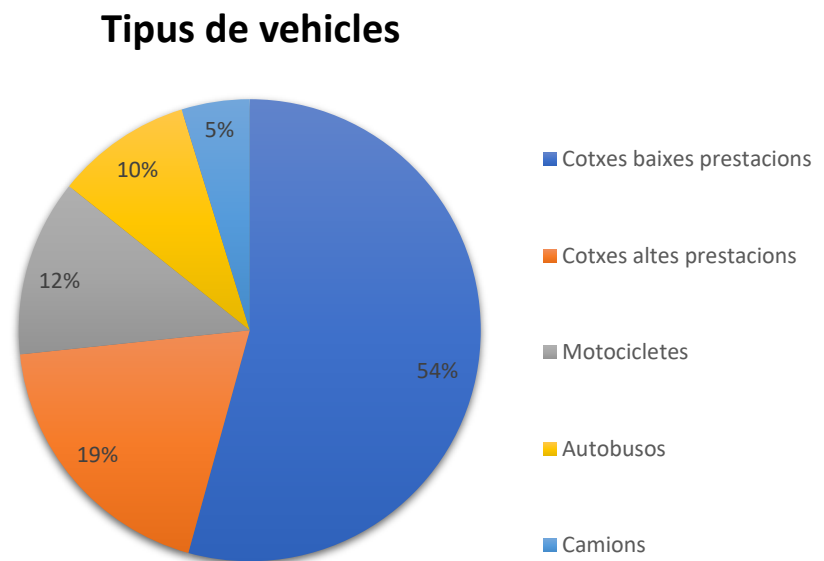


Figura 6-14: Exemple de distribució de vehicles

6.4.2. Paràmetres propis dels perfils de conducció

Amb motiu de recrear el comportament humà s'han afegit una sèrie de paràmetres, individuals per a cada vehicle, que permetran conductes personalitzades.

Per posar un exemple podem fixar-nos en el paràmetre s_0 . Aquest marca la distància que un conductor deixaria entre el seu vehicle i el següent en cas d'estar aturats. Aproximadament els conductors deixen una distància d'uns dos metres, tanmateix, algun conductor pot deixar-ne un i mig i un altre 3. Considerant aquest fet s'ha emprat la distribució normal atès a que és capaç de aproximar la variabilitat de comportaments entre conductors.

Per implementar una distribució normal dins de Matlab s'ha fet ús del mètode `'randn'`. Al invocar aquest mètode s'obté un escalar aleatori extret de la distribució normal estàndard. En aquest per a cada paràmetre s'han considerat dos distribucions normals amb la mateixa

desviació però centrades a punts diferents per distingir dos grups. Al primer grup tindriem els autobusos i els camions i al segon grup la resta de vehicles. Aquesta distinció s'ha aplicat ja que el primer grup són vehicles molt pesants i la manera de conduir-los és diferent.

Si s'agafa una mostra de 1000 vehicles i es fa un gràfica de cada paràmetre s'obtenen valors com els de la Figura 6-15:

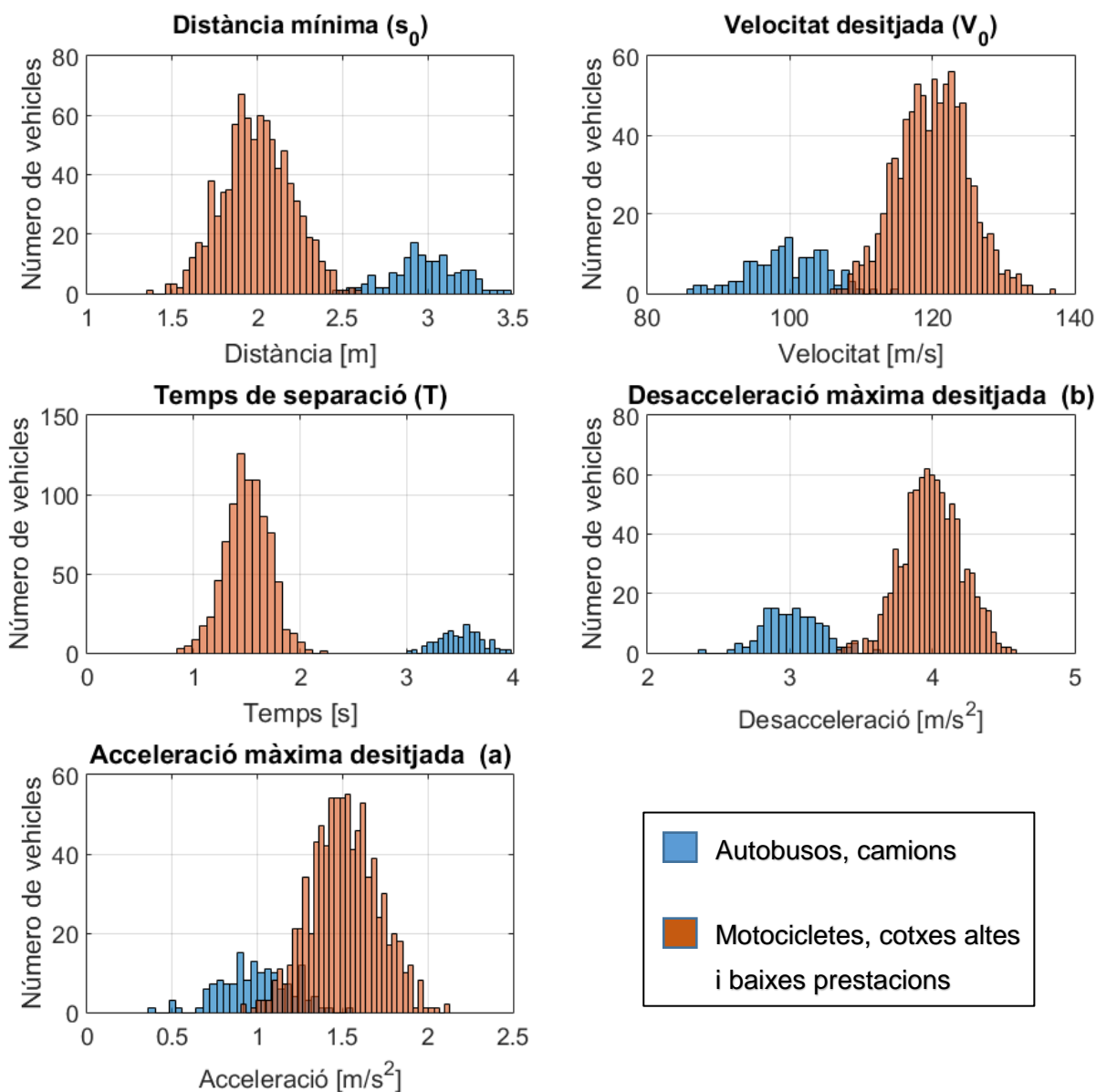


Figura 6-15: Distribució de s_0 , V_0 , T , b i a per a una mostra de 1000 vehicles

7. Simulacions

Un cop s'ha definit el funcionament de la plataforma de simulació passem a la part més pràctica del projecte, és a dir, realitzar les simulacions les quals s'analitzaran i s'extrauran conclusions.

Respecte als dos models de seguiment que s'han tractat teòricament al apartat 4.1., cadascun té un propòsit diferent. Per una banda el model simple és un model que s'ha dissenyat per a poder-se implementar en un cotxe autònom. Aquest no incorpora paràmetres propis que caracteritzen la conducció humana. Per contra, el model IDM es va dissenyar expressament amb la finalitat de simular el comportament humà. El model IDM sí que incorpora un nombre més elevat de paràmetres que intenten agrupar tots aquelles conductes inherents a cada conductor. Cada vehicle disposarà dels seus propis paràmetres i seran similars però alhora diferents. Aquestes petites diferències porten a la creació de congestions originades exclusivament pel comportament individual de cada vehicle sense que la carretera tingui cap coll d'ampolla concret que pugui propiciar la situació de trànsit.

A continuació es fan un seguit de simulacions amb ambdós models per verificar si s'obtenen els resultats esperats.

7.1. Simulació Model IDM

Per visualitzar la reducció de fluïdesa del trànsit per culpa del factor humà cal que la densitat de vehicles sigui elevada.

S'escull un radi de circuit de 40 metres, 20 vehicles en total, velocitat inicial dels vehicles de 5 Km/h i temps total de simulació de 100 segons.

Pel que fa als paràmetres propis del IDM:

```
Average_Speed = 100;           Desviation_Speed = 5;
Average_Minimum_Space = 2;      Desviation_Minimum_Space = 0.5;
Average_Time_Headway = 1.3;     Desviation_Time_Headway = 0.6;
Average_Vehicles_Acceleration = 1;Desviation_Acceleration = 0.1;
Average_Vehicles_Decelaretion = 4;Desviation_Vehicles_Deceleration = 0.2;
```

Per començar observem l'evolució de les acceleracions:

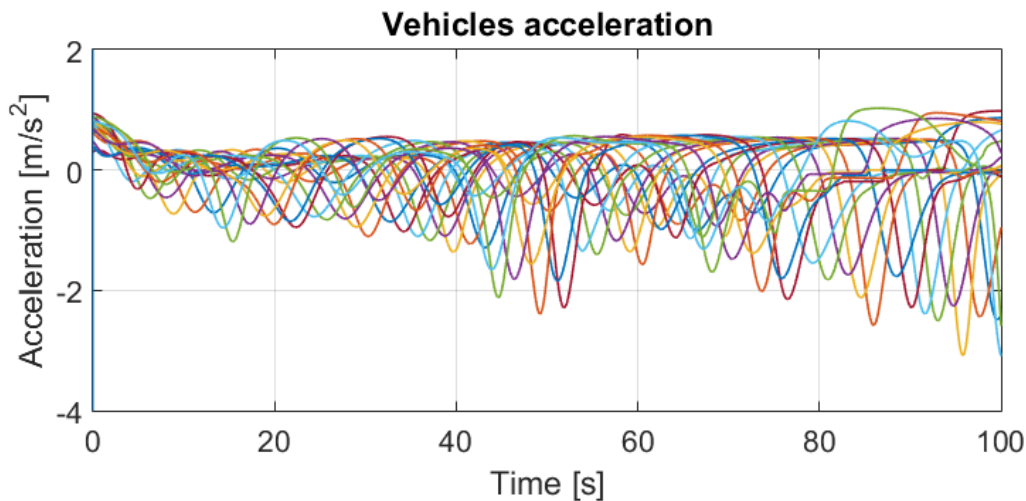


Figura 7-1: Gràfic acceleracions amb Model IDM

Veiem que les acceleracions prenen els valors desitjats, valors no desitjats correspondrien a acceleracions o desacceleracions elevades.

S'han capturat les imatges a l'instant inicial i final de la simulació a la Figura 7-2. Efectivament es comprova que s'arriba a un estat de congestió on els vehicles han de reduir més la velocitat del que caldria.

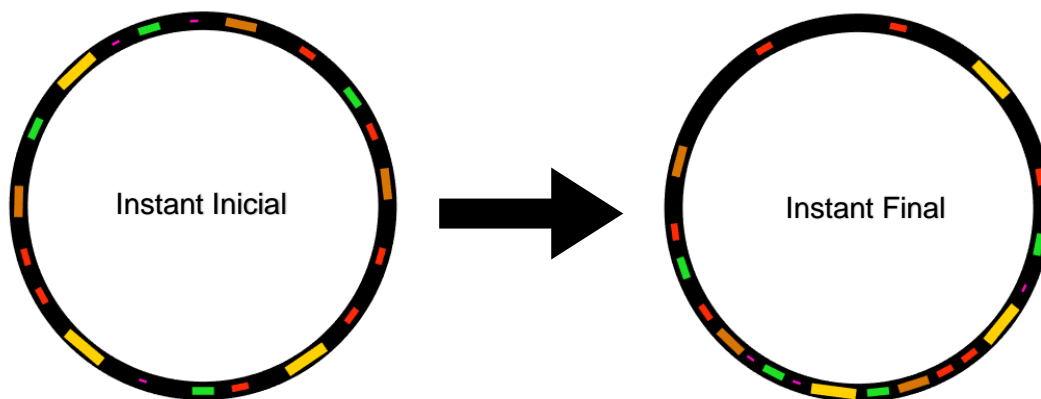


Figura 7-2: Disposició dels vehicles al llarg del temps, Model IDM

Observant l'evolució de les velocitats, a partir de la Figura 7-3, es poden observar dos règims. El primer és el règim transitori on es veu com cada vegada el rang de velocitats es fa més ampli. El segon correspon al règim estacionari, situació on l'embús s'ha desenvolupat i es va propagant a la direcció contrària d'avanç dels vehicles.

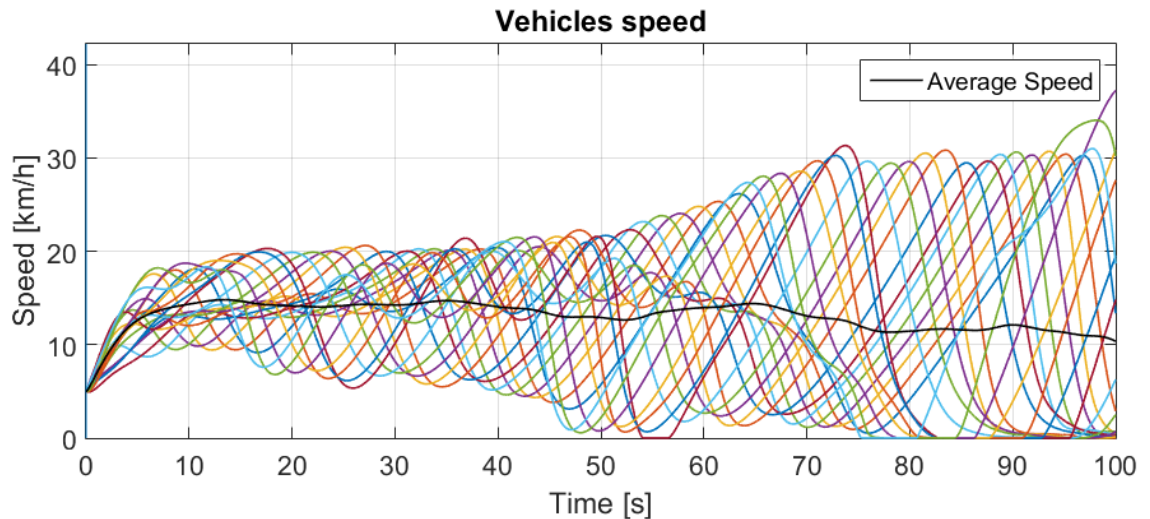


Figura 7-3: Gràfic de velocitats, Model IDM

7.2. Simulació amb el model Simple

Aquesta vegada es farà una simulació fent ús del model Simple.

Els paràmetres i condicions inicials són les següents:

Temps de simulació de 50 segons, radi del circuit de 40 metres, 20 vehicles i els següents valors pels paràmetres que fa servir l'algoritme: $k_p = 1$; $s_o = 1.5$; $i_T = 1$.

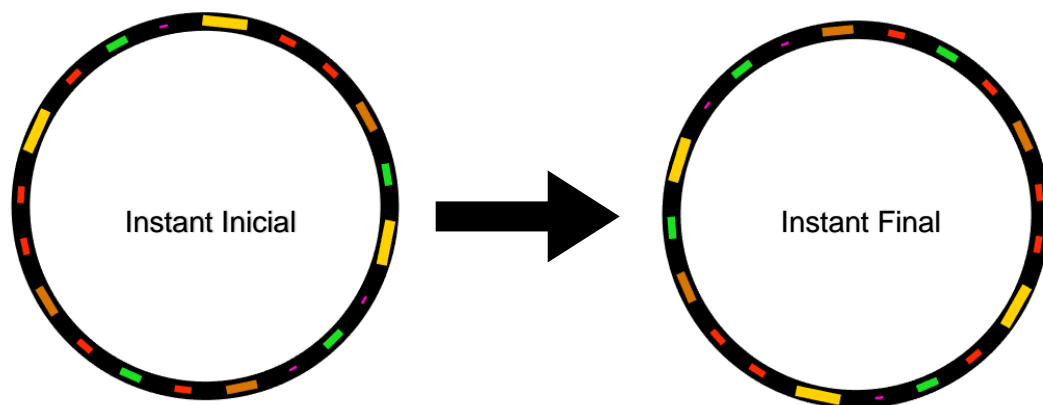


Figura 7-4: Disposició dels vehicles al llarg del temps, Model Simple

Aquest cop tal com s'observa a la Figura 7-4 els vehicles no s'alenteixen i són capaços de seguir el vehicle davant sense problemes. Aquest fet era previsible ja que l'algoritme no incorpora cap paràmetre del comportament humà. A la Figura 7-5, el gràfic de velocitats, es visualitza com tots els vehicles van adequant la seva velocitat fins que sobre el segon 20 ja tots circulen amb la mateixa rapidesa:

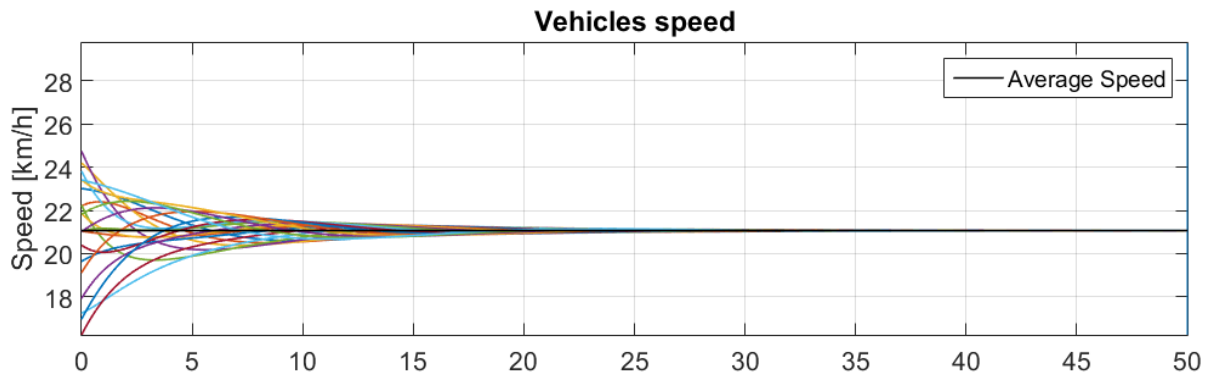


Figura 7-5: Gràfic de velocitats, Model Simple

Aquesta velocitat correspon a la màxima que poden assolir, tenint en compte la densitat de vehicles present, tot preservant la distància de seguretat (aquesta creix al augmentar la velocitat). De fet es pot comprovar analíticament a partir de l'expressió de l'algorisme:

Comprovació de velocitat màxima per a una certa densitat de circulació:

- (1) $v_k = k_p \cdot (S_i - S_{ref}) + v_{k+1}$ *Expressió del control de moviment.*
- (2) $v_k - v_{k+1} = 0 = k_p \cdot (S_i - S_{ref})$ *La diferència de velocitats és nul·la al assolir l'estat estacionari.*
- (3) $S_i = S_{ref} = S_0 + T \cdot v_k$
- (4) $S_i = \frac{2\pi r - \sum_{i=1}^n L_i}{n}$ *Els vehicles equidisten els uns dels altres.*
- (5) $\frac{2\pi r - \sum_{i=1}^n L_i}{n} = S_0 + T \cdot v_k$
- (6) $v_k = \frac{\frac{2\pi r - \sum_{i=1}^n L_i}{n} - S_0}{T} = \frac{\frac{2\pi 40 - 104,3}{20} - 1,5}{1} = 5,85 \text{ m/s} = 21,06 \text{ km/h}$

El resultat coincideix amb el valor exacte que marca la simulació com es pot observar a la Figura 7-5.

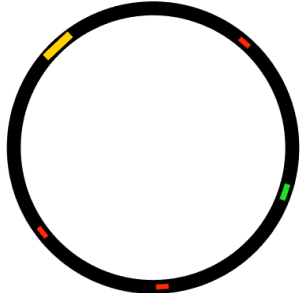
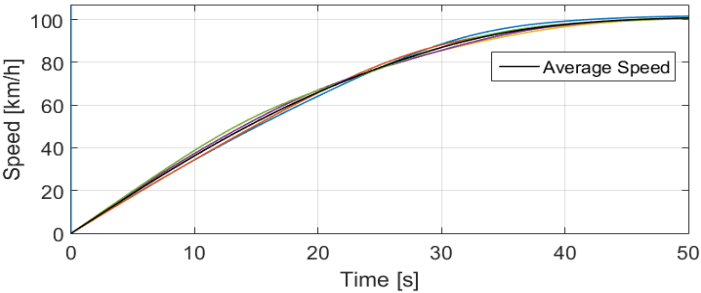
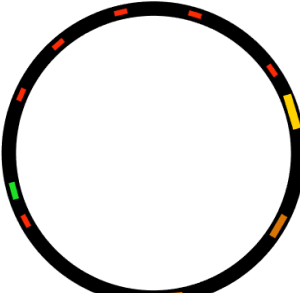
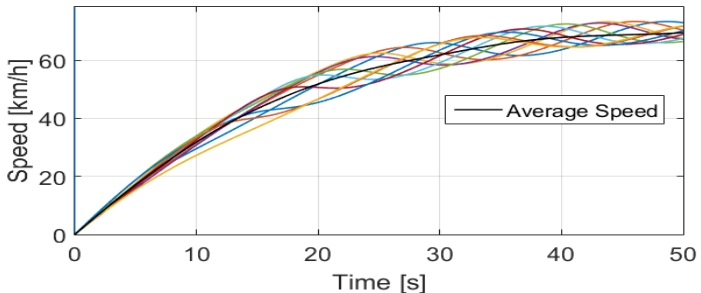
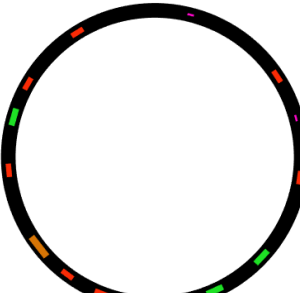
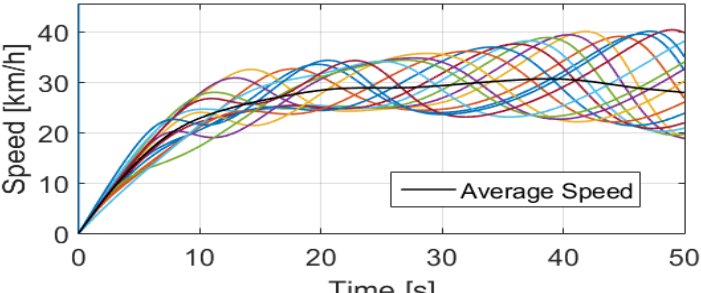
7.3. Simulacions per veure afectació de la densitat de vehicles

La densitat de vehicles k és el factor més determinant alhora de marcar la velocitat mitja u i el flux de cotxes q . Aquestes variables estan relacionades en una expressió :

$$q = k \cdot u \tag{Eq. 7.1.}$$

$$\left[\frac{\text{vehicles}}{\text{hora}} \right] = \left[\frac{\text{vehicles}}{\text{km}} \right] \cdot \left[\frac{\text{km}}{\text{h}} \right]$$

Per observar la afectació de la densitat de vehicles a la fluïdesa del tràfic, a la següent taula es mostren les gràfiques de l'evolució de velocitat per a diferents densitats:

| N_{veh} | Circuit + vehicles | Gràfic velocitats |
|-----------|---|--|
| 5 |  |  |
| 10 |  |  |
| 15 |  |  |

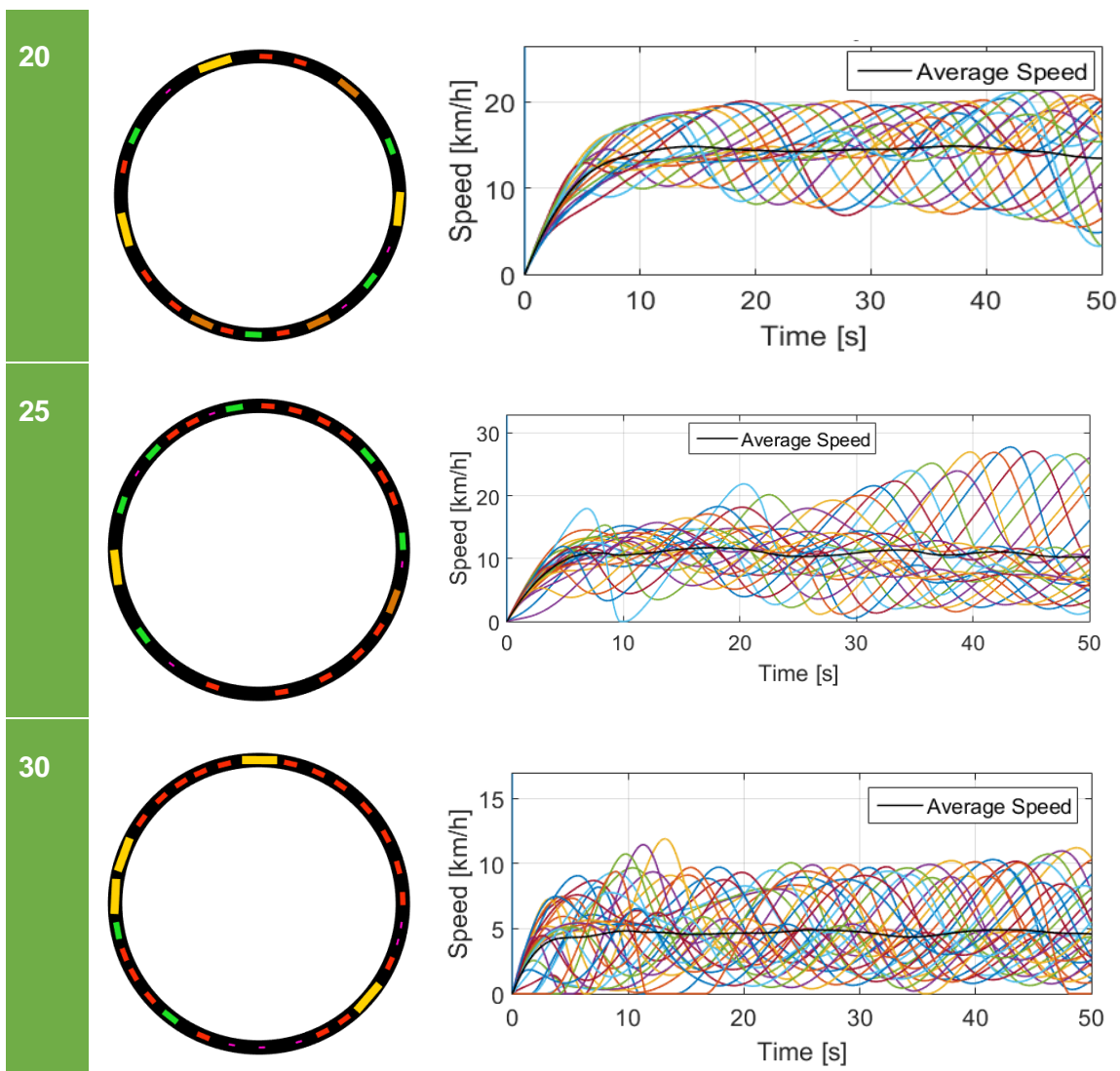


Figura 7-6: Taula simulacions variant el nombre de vehicles

En la Figura 7-6 es posa de manifest com es comporta el trànsit en funció de la densitat de vehicles. En la primera simulació on la densitat és baixa els vehicles gairebé no es veuen afectats per la resta i circulen a la velocitat desitjada. A mesura que anem augmentant el nombre de vehicles es veu com les velocitats són més inestables, rang de valors més ampli, i el seu valor mig va disminuint.

Tornant a l'Eq. 7.1., la representació d'aquesta expressió es coneix amb el nom de 'Diagrama fonamental' [5] i té moltes variants.

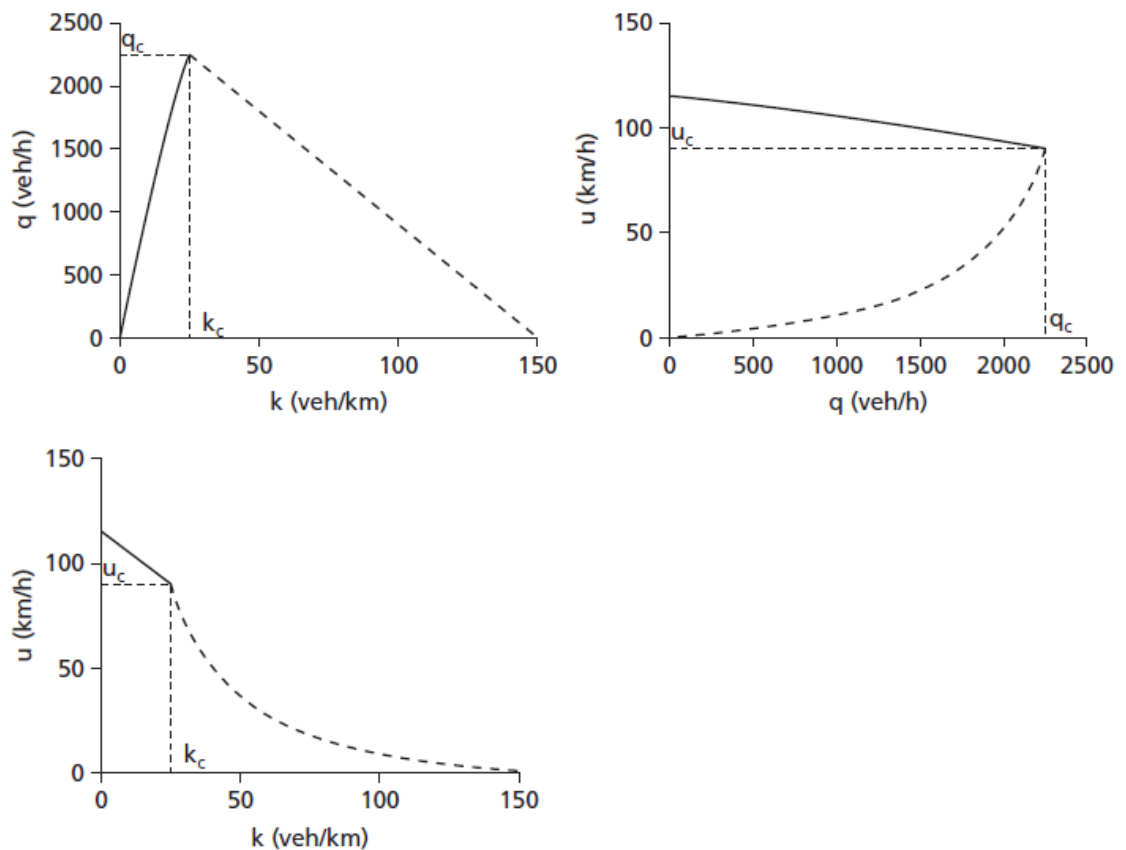


Figura 7-7: Exemples del Diagrama Fonamental

Amb les diferents versions del Diagrama Fonamental es mostren els punts més importants del trànsit. Aquests són la capacitat de la carretera, la densitat/velocitat crítiques a partir de les quals el flux comença a disminuir, densitat a la qual la velocitat arriba a zero, etc.

Amb l'objectiu de recrear el Diagrama Fonamental a partir de la plataforma de simulació dissenyada en aquest Treball Final de Curs, es realitzaran un seguit de simulacions. Aquesta vegada tots els vehicles s'agafaran del mateix tipus per no introduir cap tipus d'alteració. El radi del circuit serà de 200 metres (amb un radi d'aquesta mida es podran obtenir mostres més representatives del que està passant).

Com que fer manualment cada punt del gràfic es convertiria en una feina molt llarga i carregosa s'ha creat un programa, es troba a l'Annex B.13., el qual automàticament executa la simulació variant el nombre de vehicles presents.

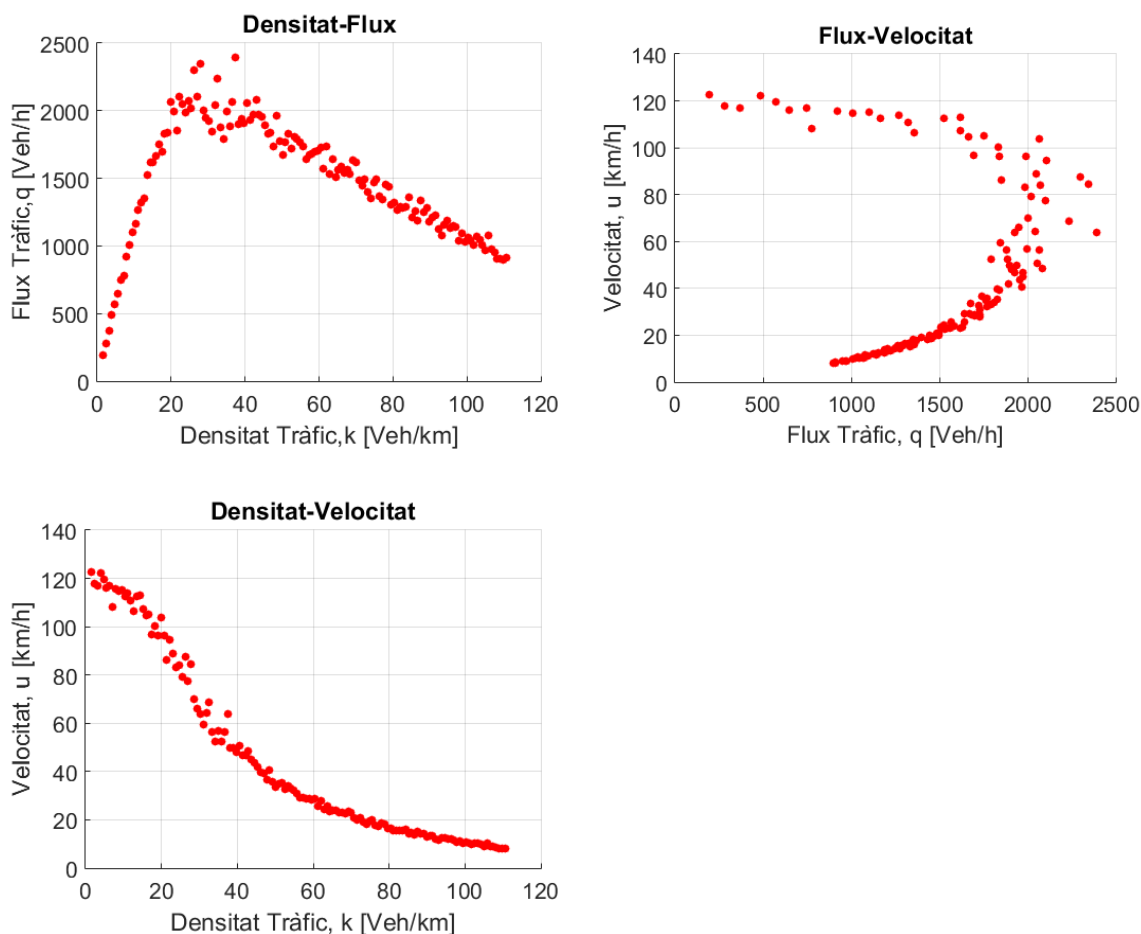


Figura 7-8: Diagramas de Flux Fonamental realitzats amb la plataforma de simulació del projecte

Obtenim els diagrames de la Figura 7-9 que s'aproximen de forma satisfactòria als diagrames teòrics vists anteriorment. S'observa que el flux de vehicles comença a decaure quan la densitat arriba als 30 veh/km o quan la velocitat als 85 km/h. La capacitat màxima del circuit és de d'uns 2.300 veh/h i que si la densitat se situa per sobre dels 140-150 km/h els vehicles estaran aturats:

- $K_c = 30$ veh/km
- $q_c = 2.300$ veh/h
- $k_{jam} = 145$ veh/km ($u = 0$)
- $u_c = 85$ km/h

7.4. Simulacions per observar l'afectació individual dels paràmetres pertanyents als perfils del conductors

En aquest apartat s'observa quins paràmetres del model IDM (el model que simula la fluïdesa del trànsit amb conductors humans) ajuden més a la formació de congestions. Es realitzarà una simulació amb els paràmetres següents que s'agafa com a referència.

| Radi | N_{veh} | T_{sim} | $V_{inicial}$ | s_0 | T | a | b | δ |
|------|-----------|-----------|---------------|-------|-------|--------------------|--------------------|----------|
| 20 | 10 | 200 s | 0 m/s | 2 m | 1,5 s | 1 m/s ² | 4 m/s ² | 4 |

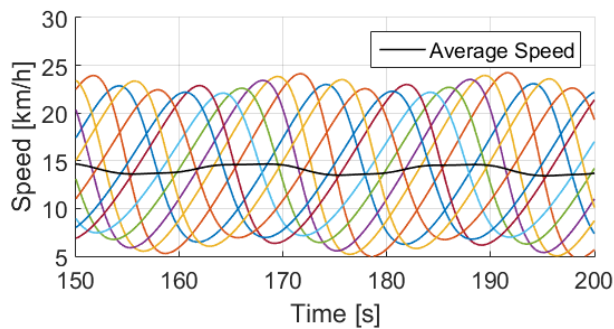
Figura 7-9 Taula amb paràmetres de referència

Seguidament es procedeix a repetir la simulació dues vegades per cada un dels paràmetres s_0 , T , a i b . En el primer cas prenent un valor més baix i en el segon agafant un valor més alt del de referència. Aquesta vegada no s'usarà el model mecànic ja que al per poder apreciar les diferències de comportament per cada variable s'usaran valors poc freqüents que poden requerir acceleracions elevades. El temps de simulació de 200 segons s'ha escollit per tal de donar temps a que les velocitats s'estabilitzin.

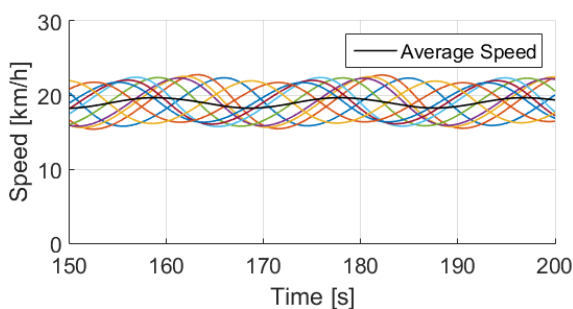
De tota la simulació es selecciona un tram pertanyent al règim estacionari i es fa la taula següent (a l'apartat C de l'annex es mostren les gràfiques de velocitats completes):

Velocitats al assolir règim estacionari:

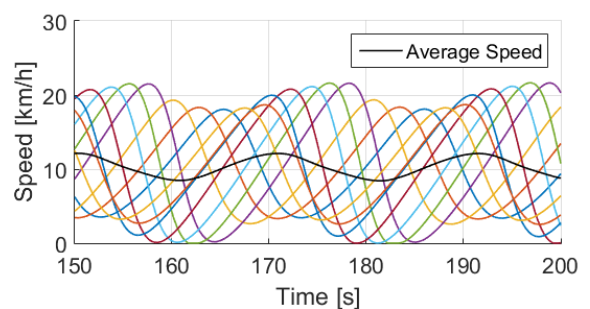
Cas de referència



Cas $s_0 = 0,5$ m



Cas $s_0 = 3,5$ m



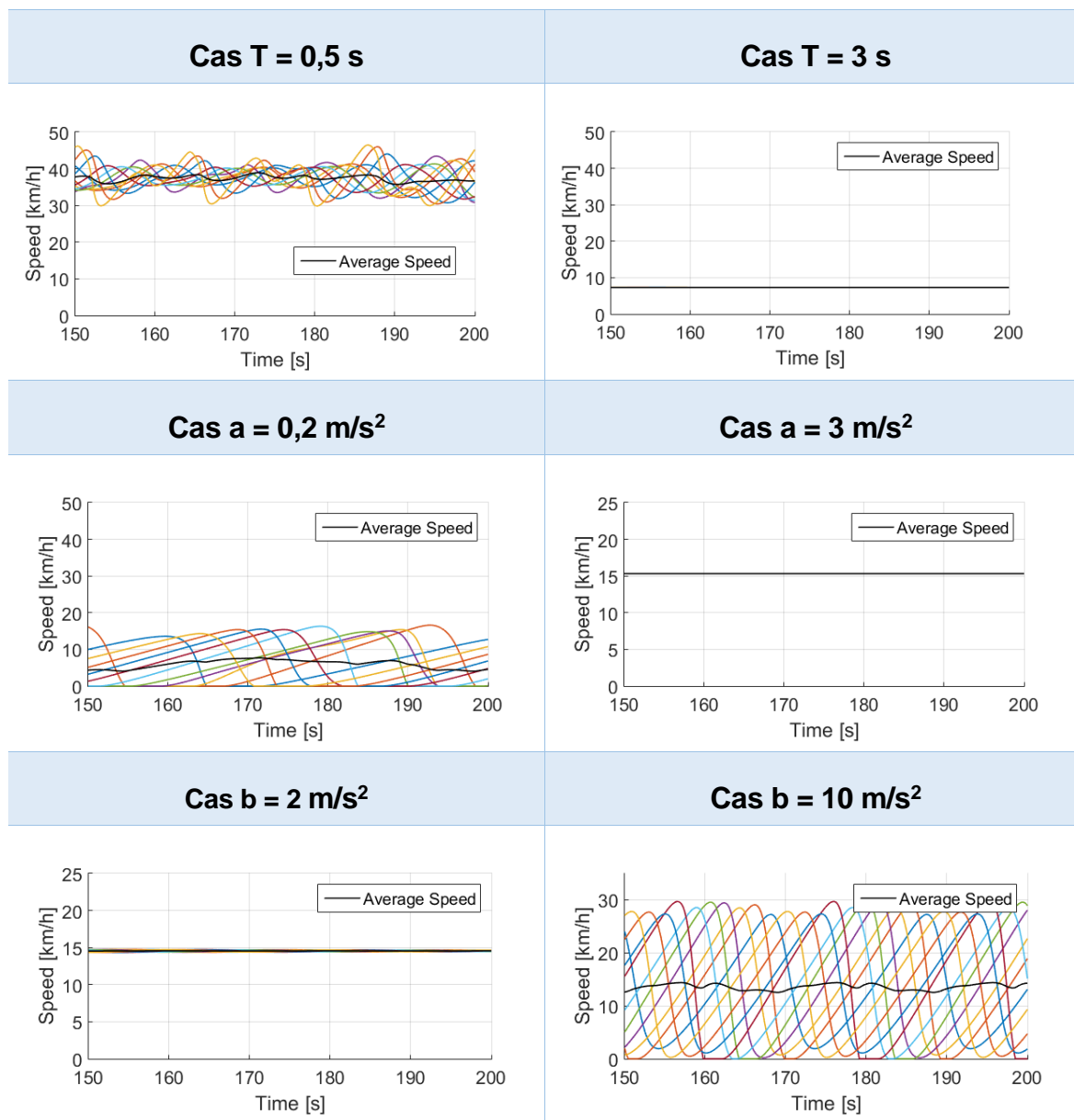


Figura 7-10: Taula de simulacions variant un paràmetre

Les conclusions que se'n deriven de la taula són :

- **s₀**: valors baixos de la distància mínima de seguretat fan augmentar lleugerament la velocitat mitjana dels vehicles i redueix considerablement el rang de velocitats màximes i mínimes per a aquesta densitat de vehicles. En canvi al augmentar les velocitats mitjanes disminueixen una mica i el rang de velocitats no es veu molt afectat.
- **T**: al disminuir el temps que els vehicles deixen al proper la velocitat mitjana augmenta de forma notable, amb aquesta densitat de vehicles s'arriba a duplicar. Altrament al augmentar-lo els vehicles aconseguen homogeneïtzar la seva velocitat amb el

contra argument de reduir la seva velocitat mitjana.

- **a:** al reduir l'acceleració dels vehicles aquests originen el fenomen 'stop and go', és a dir, es crea un embús que els obliga a aturar-se contínuament i la velocitat mitjana disminueix notablement. Per contra al augmentar la velocitat aquests són capaços de seguir-se sense entorpir-se i amb una velocitat mitjana lleugerament superior.
- **b:** al reduir la desacceleració màxima aquests tornen a seguir-se a la mateixa velocitat i amb un valor lleugerament superior al de referència. Al augmentar-la es torna a produir el fenomen 'stop and go', aquest cop amb rangs de velocitats més amplis i velocitat mitjana més alta que en el cas de d'acceleració de valor alt.

D'aquestes observacions s'extreu que l'afectació de l'acceleració i desacceleració màxima actuen de forma totalment oposada. També el fet de que modificant alguns paràmetres es pot aconseguir millorar el seguiment. El problema radica en què el cas de referència és el més representatiu del comportament de la població i seria difícil intentar canviar aquestes conductes.

Per últim s'ha fet una simulació amb les mateixes condicions però aquest cop utilitzant el control que proporciona el Model Simple. Tenint en compte que està pensat per cotxes autònoms i aquests, apart de no dependre dels reflexos d'una persona, són capaços de comunicar-se entre ells i conèixer al moment la acció que realitzarà el cotxe de davant. Per tant, la seva velocitat de resposta d'actuació és molt més elevada i resulta lògic pensar que no necessiten deixar tanta distància de seguretat i per això s'han agafat els paràmetres del Model Simple de $s_0 = 2\text{m}$ i $T = 0,5\text{s}$. S'obté una velocitat de 48 km/h per a tots els vehicles. Com és d'esperar la velocitat es molt superior a la que s'arriba a una situació de trànsit amb conductors humans.

7.5. Simulació amb el model amb alçada

L'objectiu principal d'incloure la possibilitat de fer simulacions en tres dimensions era poder observar com afecta el fet d'incloure variacions de pendents al llarg del circuit.

Per fer la comparativa s'han fet dos simulacions, una en el pla i l'altre amb alçada on l'altura del circuit depèn d'una funció senoïdal amb tres pics al circuit i pendent màxim de 26,6 graus. En ambdues simulacions s'ha fet servir la mateixa matriu 'Par', és a dir, mateixos vehicles amb mateixos paràmetres de comportament. El radi del model en el pla és de 50 metres i en el model amb alçada de 53 de manera que les longituds dels circuits són equivalents ($333,01 \approx 332,96\text{ m}$).

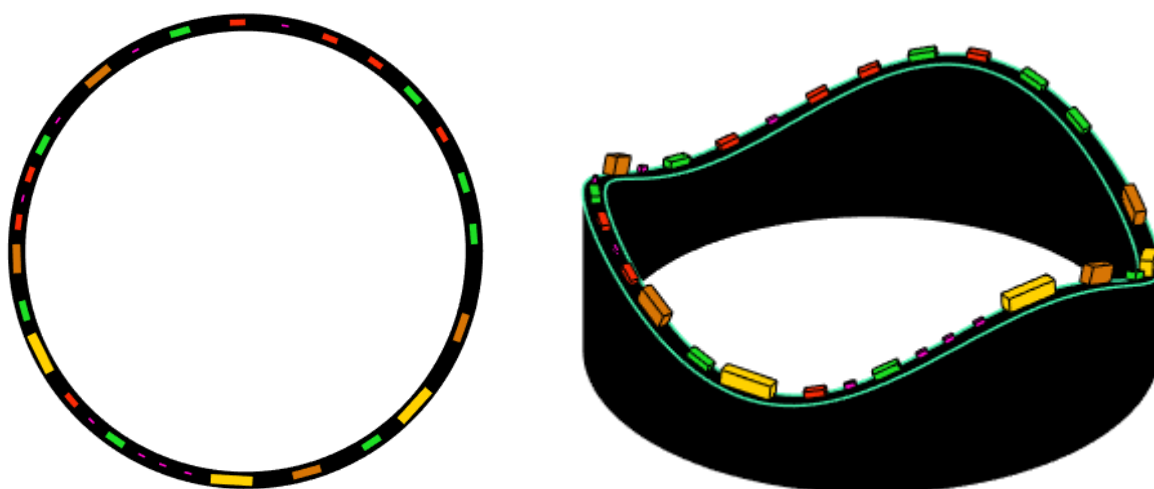


Figura 7-11: Circuits en el pla i amb alçada

Després de fer les simulacions s'obtenen les gràfiques d'acceleració i velocitat següents:

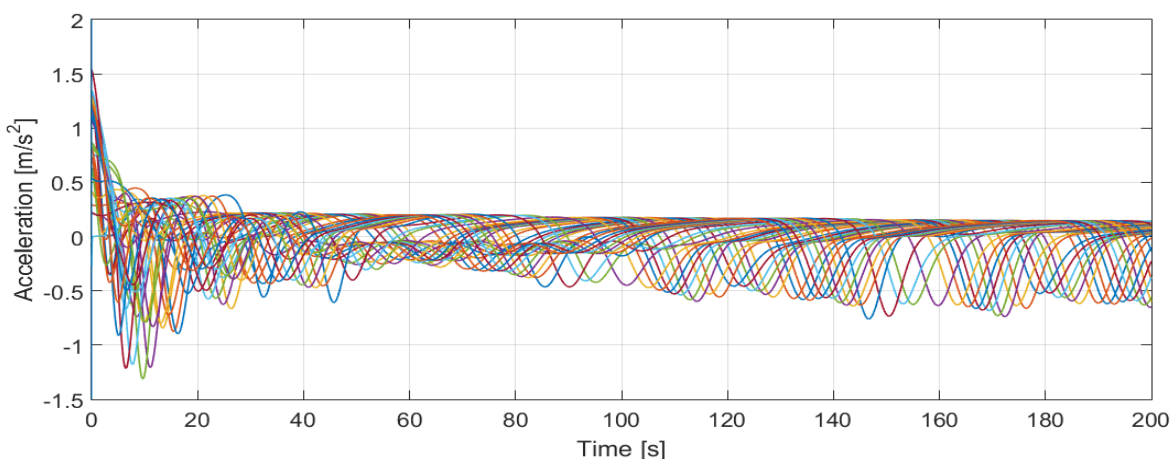


Figura 7-12: Gràfic d'acceleracions del model en el pla

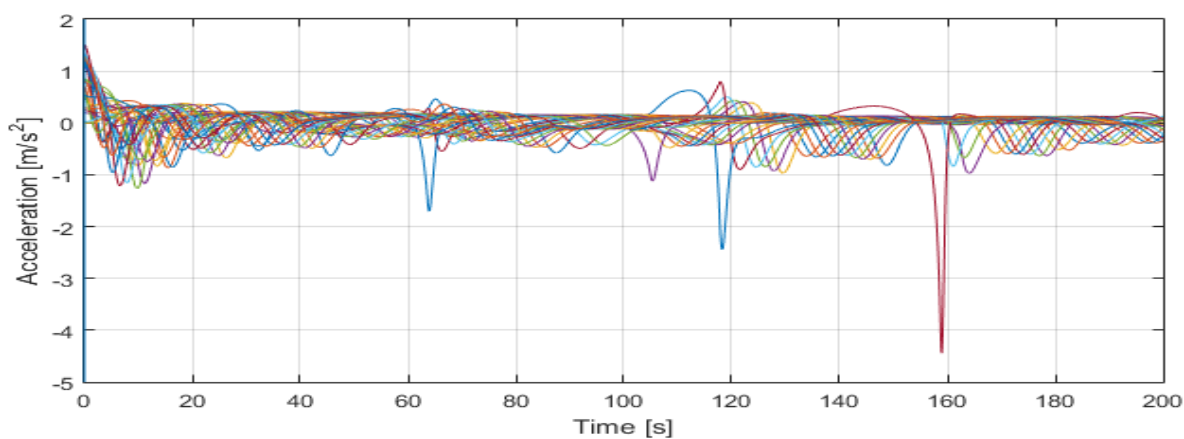


Figura 7-13: Gràfic d'acceleracions del model amb alçada

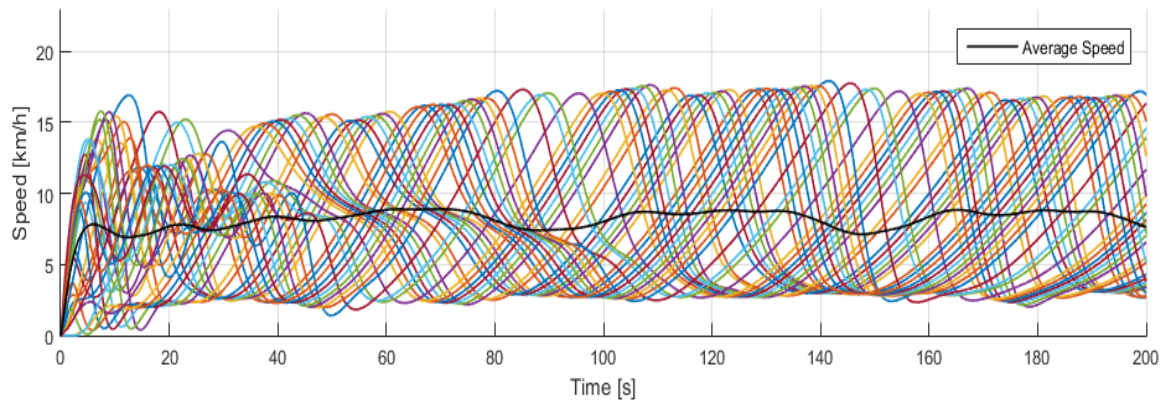


Figura 7-14: Gràfic de velocitats del model en el pla

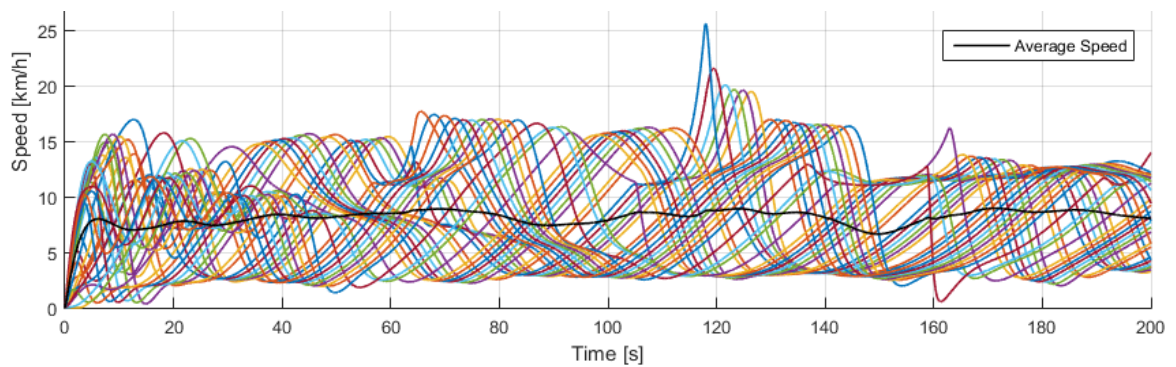


Figura 7-15: Gràfic de velocitats del model amb alçada

En aquests s'observa que la velocitat mitja no es veu molt afectada, en ambdós casos està sobre els 7-8 km/h. Per altra banda es contemplen diverses situacions on al model amb alçada les variacions de pendent originen maniobres més brusques. En el model en el pla les acceleracions més grans, un cop s'arriba al règim estacionari, són de l'ordre de $0,7 \text{ m/s}^2$ i, el model amb alçada són similars però amb pics cada cert temps que arriben als 4 m/s^2 .

8. Pressupost econòmic

Per calcular el pressupost econòmic que suposa desenvolupar aquest projecte, es mostraran les despeses que una empresa hauria de cobrir a un enginyer que efectués les mateixes tasques. A la pràctica el cost econòmic ha estat pràcticament zero ja que les eines o programaris necessaris han estat proporcionats per la entitat universitària.

Material i Software

| Concepte | Quantitat | Cost Unitari (€/Unitat) | Cost Total (€) |
|------------------------------|-----------|-------------------------|----------------|
| Matlab 2016b | 1 | 2.000,00 | 2.000,00 |
| Plataforma Simulink | 1 | 3.000,00 | 3.000,00 |
| Ordinador prestacions mitges | 1 | 600,00 | 600,00 |
| TOTAL (1) | | | 5.600,00€ |

Figura 8-1: Taula Materials i Software utilitzats

Temps de treball

| Temps dedicat | Hores | Cost Unitari (€/hora) | Cost Total (€) |
|------------------|-------|-----------------------|----------------|
| Estudi previ | 10 | 10 | 100,00 |
| Creació del codi | 500 | 10 | 5000,00 |
| Simulacions | 200 | 10 | 2000,00 |
| TOTAL (2) | | | 7.100,00€ |

Figura 8-2: Taula del temps de treball emprat

En total el pressupost econòmic suposa un valor de 12.700 €.

Conclusions

La simulació és una eina molt potent que permet estudiar sistemes complexes i preveure el seu comportament. En aquest projecte s'ha dissenyat una plataforma de simulació destinada a la implementació de models de seguiment. Com s'ha constatat a l'apartat de simulacions, s'ha complert l'objectiu de reproduir situacions reals de trànsit introduint els comportaments d'un conductor amb el model 'Intelligent Driver Model' (IDM). També s'ha analitzat cada variable d'aquest model per esbrinar com millorar la fluïdesa del trànsit. S'ha trobat que les congestions es creen majoritàriament degut a la manera d'accelerar i de frenar dels conductors. La clau resideix en no frenar de manera brusca ni accelerar molt lentament, ara bé, és una tasca molt difícil la de millorar la coordinació entre conductors.

Amb referència a les simulacions amb el model mecànic, s'ha vist com no sempre els vehicles són capaços de realitzar les accions que se li encomanen. Hi ha a vegades que poden apropar-se més de la distància mínima desitjada degut a una forta frenada. És per això que cal emprar paràmetres adients, dins d'uns marges, perquè la simulació sigui útil.

Així com per simular el comportament humà ha calgut utilitzar algorismes complexes amb moltes variables, s'ha vist que amb un algorisme simple es pot obtenir un model de bona qualitat pels vehicles autònoms. Fent comparacions amb situacions d'igualtat els vehicles autònoms són més eficients a l'hora de circular.

Pel que fa a les contribucions realitzades en el treball cal diferenciar entre les idees i models propis i el que s'ha agafat d'altres autors. En primer lloc la idea de fer una simulació en el pla s'ha agafat de l'autor Martin Treiber. A part d'afegir fins a cinc tipus diferents de vehicles, les característiques afegides més rellevants són: incloure un model mecànic individual per a cada vehicle i incorporar una opció que permet realitzar estudis en circuits de tres dimensions on es poden definir circuits amb inclinacions diferents. Amb aquestes dues contribucions s'ha aconseguit apropar-se més a situacions reals. Després la contribució més rellevant del projecte, que dóna validesa al model IDM, ha estat la de reproduir les diferents variants del Diagrama Fonamental amb el que es pot extreure la capacitat d'una carretera, la velocitat crítica o densitat crítica amb les quals el flux de vehicles disminueix.

Per concloure es comenten els treballs futurs que donen continuïtat al projecte. Per començar es poden afegir un carril o més al circuit juntament amb models d'avançament corresponents per completar encara més la plataforma de simulació. De cara als vehicles autònoms, actualment comencen a sortir alguns models al mercat. Aquests encara tenen camí per recórrer abans de introduir-se completament al trànsit, sobretot degut a la complexa tasca de circular en una ciutat. No obstant, hi ha qui creu que algun dia podrien arribar a substituir completament als conductors. Abans però, ens trobaríem en una època de transició on

conviurien vehicles controlats per persones i controlats de manera autònoma. Per tant, es pot fer servir la plataforma de simulació d'aquest treball per estudiar com afectaria la combinació de models autònoms amb models de comportament humà a la fluïdesa del trànsit. Un altre característica per afegir seria la de crear circuits més complexes on per exemple els cotxes segons la curvatura limitessin la seva velocitat.

Per últim, seria interessant fer servir els càlculs que fa la plataforma de simulació per posicionar els vehicles al llarg del temps però, alhora de recrear la simulació gràficament, utilitzar un programa amb millors eines visuals com és Unity 3D per obtenir una simulació totalment realista (tant matemàtica com visualment).

Agraïments

En primer lloc voldria donar les gràcies al propi lector. Seguidament vull posar de manifest la gran satisfacció que m'ha deixat el fet de realitzar aquest projecte. Gràcies a la temàtica innovadora i la utilització de software d'alt nivell he pogut gaudir i aprendre al llarg del projecte. Per suposat dono gràcies per la ajuda del tutor Arnau Dòria Cerezo, primer per haver proposat el treball i després per guiar-me durant la realització. Per últim vull agrair a totes les persones que comparteixen el seu coneixement i experiència de manera lliure i desinteressada. Gràcies a la comunitat online cada cop que m'ha sorgit algun dubte de la vessant de la programació, ja sigui mitjançant fòrums, llibreries o manuals, he pogut resoldre els conflictes i seguir endavant.

Bibliografia

Referències bibliogràfiques

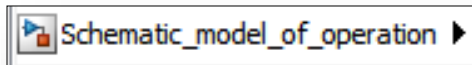
- [1] Plataforma de simulació consultada al setembre de 2016, desenvolupada per l'autor Martin Treiber, enllaç:< <http://www.traffic-simulation.de/>>.
- [2] MARTIN TREIBER AND DIRK HELBING, *Explanation of observed features of self-organization in traffic flow*, University of Stuttgart, arXiv:cond-mat/9901239v1, Jan 2000
- [3] DANIEL MONTESINOS MIRACLE, *Control DCmotor Mobilitat Elèctrica*, Transparències Mobilitat Elèctrica Etseib (Universitat Politècnica de Catalunya), 2016
- [4] RAFAEL RAMÍREZ ROS, *Cálculo vectorial*, Apunts Equacions Diferencials, Departament de Matemàtiques Universitat Politècnica de Catalunya, 2016
- [5] TRAFFIC FLOW THEORY AND MODELING, *Serge Hoogendoorn and Victor Knoop*, Bert van Wee, Jan Anne Annema and David Banister (Eds.) *The Transport System and Transport Policy*, Edward Elgar Publishing, Cheltenham, UK ,p. 16-17

Bibliografia complementària

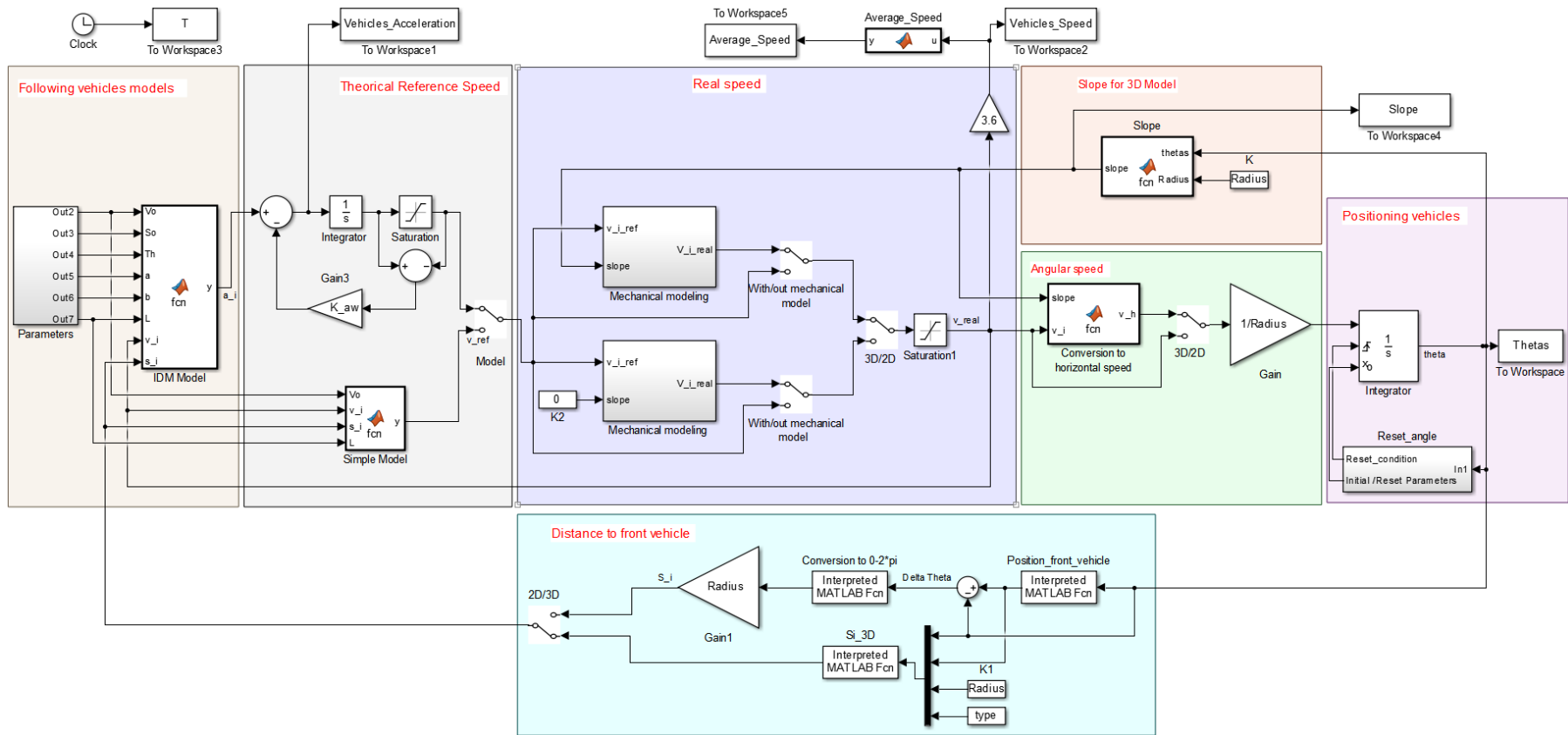
- MARTIN TREIBER, ANSGAR HENNECKE AND DIRK HELBING, *Congested Traffic States in Empirical Observations and Microscopic Simulations*, II. Institute of Theoretical Physics, University of Stuttgart, arXiv:cond-mat/0002177v2 Aug 2000
- JONGSANG SUH, KYONGSUYI, JIYEOLJUNG, KYUNGIUNLEE, HYOKJINCHONG, *Design and evaluation of a model predictive vehicle control algorithm for automated driving using a vehicle traffic Simulator*, School of Mechanical and Aerospace Engineering, Seoul National University, 2016

Annex

A. Esquemes de blocs Simulink

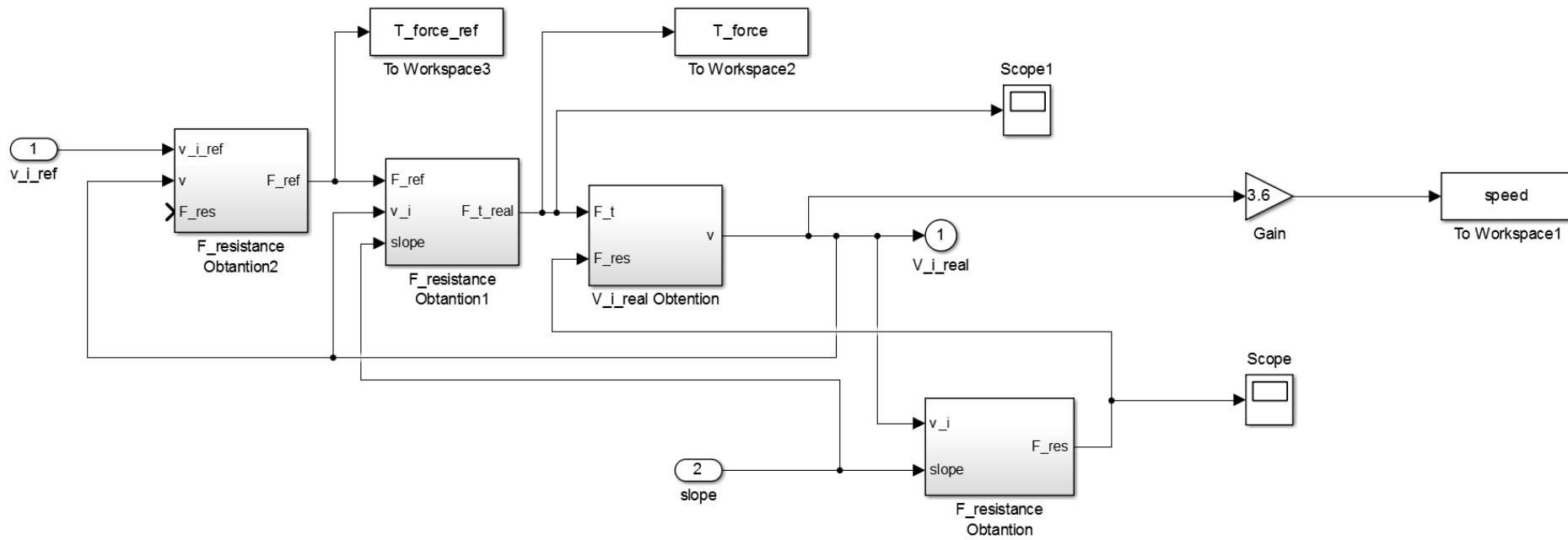


A.1. Esquema de blocs principal



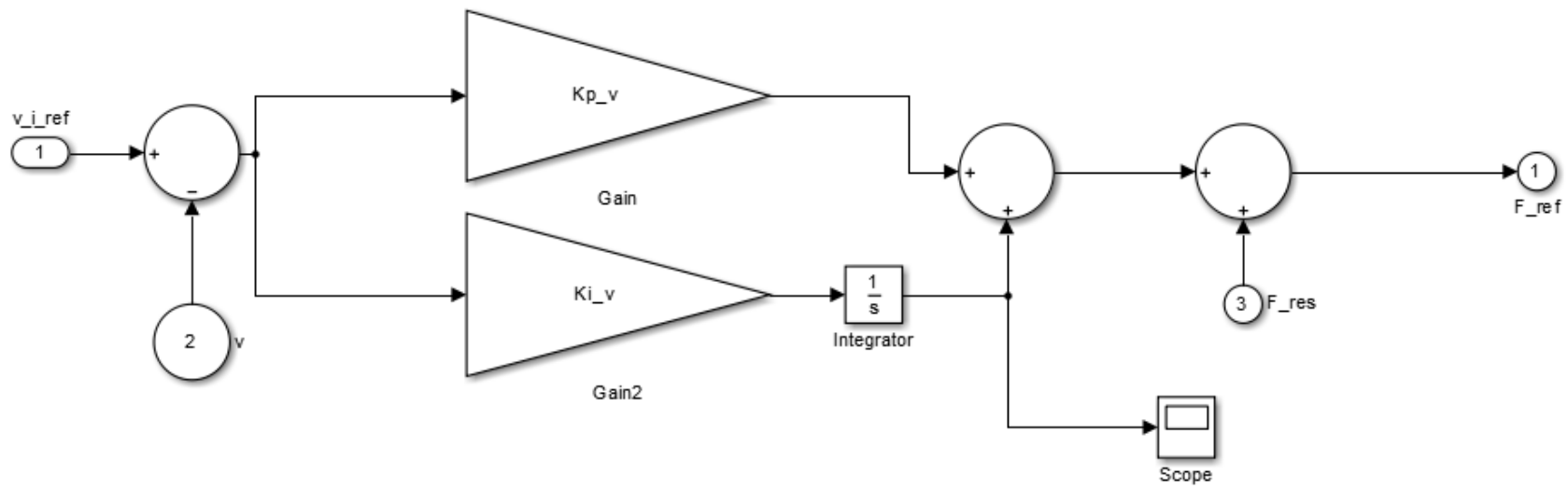
A.2. Esquema dels blocs del model mecànic

Schematic_model_of_operation Mechanical modeling



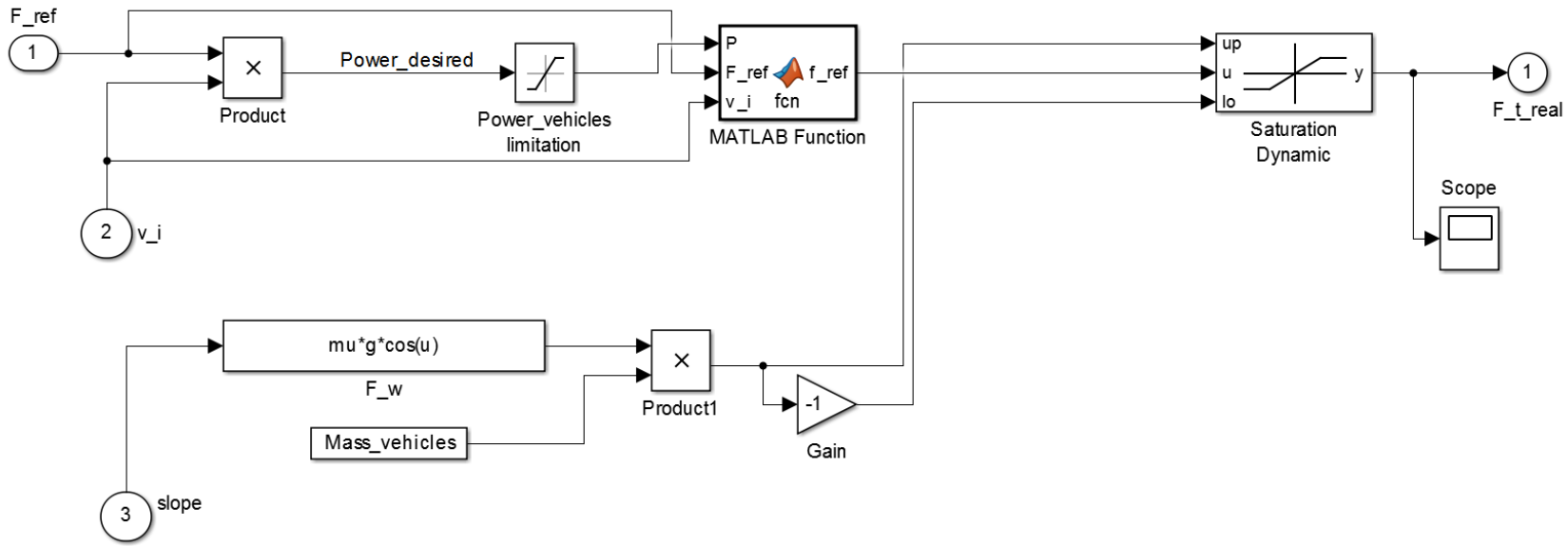
A.2.1. Blocs del càlcul de la força de referència

Schematic_model_of_operation ▶ Mechanical modeling ▶ F_resistance Obtantion2



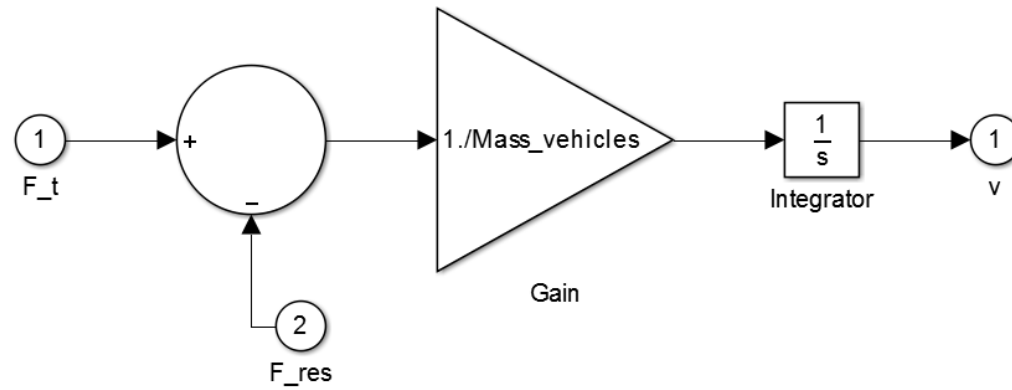
A.2.2. Blocs per obtenir la força real

Schematic_model_of_operation ▶ Mechanical modeling ▶ F_resistance Obtantion1



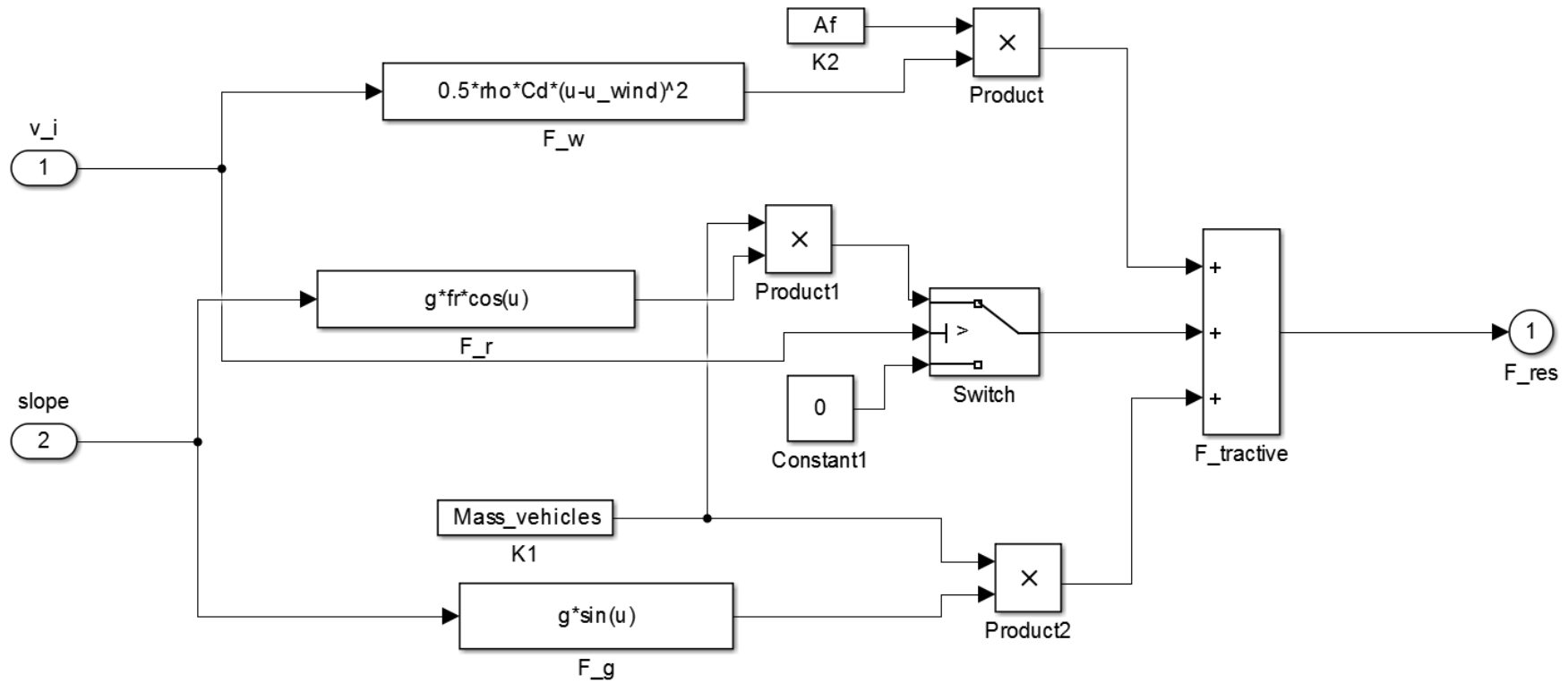
A.2.3. Blocs per obtenir la velocitat real

Schematic_model_of_operation ▶ Mechanical modeling ▶ V_i_real Obtention

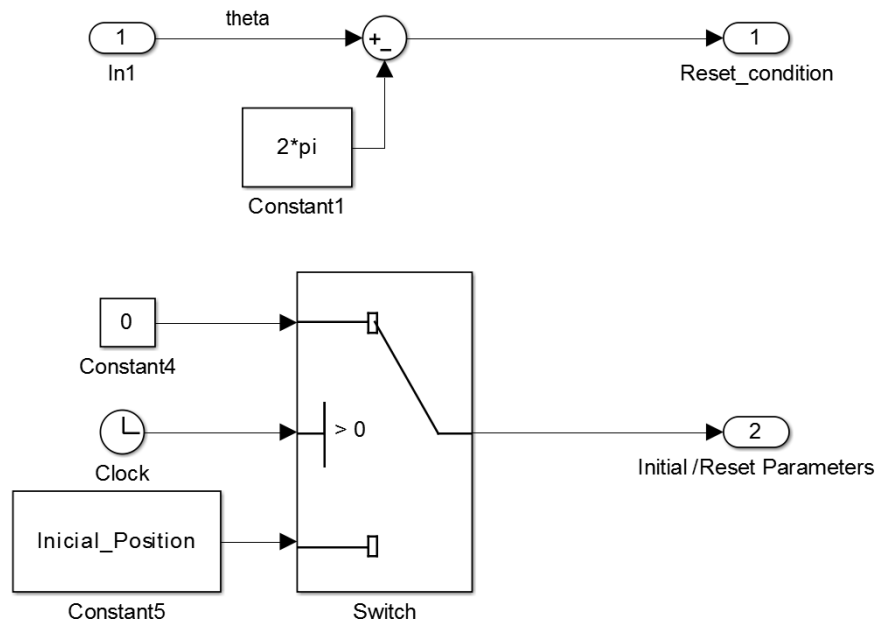
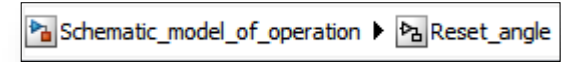


A.2.4. Blocs per obtenir la fora resistant total

Schematic_model_of_operation ▶ Mechanical modeling ▶ F_resistance Obtantion



A.3. Blocs de inicialització i reset del l'integrador velocitat-posició



B. Fitxers Matlab

B.1. Traffic flow simulation 2D

```

close all
clear all
clc

%% Calling parameters in other scripts
Input_Parameters;

%% Line track
landa = linspace(0,2*pi);
%x1 = Radius*cos(landa);y1 = Radius*sin(landa);
x2 = (Radius+2.2)*cos(landa);y2 = (Radius+2.2)*sin(landa);
x3 = (Radius-2.1)*cos(landa);y3 = (Radius-2.1)*sin(landa);

%% Making of animation (Figure 1)
sim('Schematic_model_of_operation');

vid = VideoWriter('sims','MPEG-4');
vid.FrameRate = 30;
myVideo.Quality = 500;

%% Graphics

% figure(1)
plot(T,Vehicles_Acceleration);
t2 = line([0 0], ylim);
grid on;hold on;
title('Vehicles acceleration');
xlabel('Time [s]');ylabel('Acceleration [m/s^2]');

%%
ff=Functions;

figure(2)
set(figure(2), 'Position', [70 150 1800 700]);hold on;
subplot(2,6,[1 2 7 8], 'Color','r')
Text = text(-
Radius,Radius*1.2, strcat('Time:', num2str(T(1),2)), 'FontSize',15);

% Background
r=Radius-2;R=Radius+2;xf=0;Xf=0;yf=0;Yf=0;
t = linspace(0,2*pi,200);
x = xf + r*cos(t);
y = yf + r*sin(t);
X = Xf + R*cos(t);
Y = Yf + R*sin(t);
patch([x X],[y Y], 'k', 'linestyle', 'non');
%plot(x1,y1, '--b');
hold on;plot(x2,y2, 'color', Linecolor, 'linewidth', -0.03*Radius+3.5);hold
on;plot(x3,y3, 'color', Linecolor, 'linewidth', -0.03*Radius+3.5);

```

```

title('Traffic flow simulation','FontSize',20);
axis([-1.2*Radius 1.2*Radius -1.2*Radius 1.2*Radius]);
axis off;hold on;axis equal;
%Inizializing cars
X_1 = [];X_2 = [];X_3 = [];X_4 = [];X_5 = [];
Y_1 = [];Y_2 = [];Y_3 = [];Y_4 = [];Y_5 = [];
for k = 1:n
    [Xk,Yk] = ff.Coord2D(Par(1,k),Par(2,k),Thetas(1,k),Radius);
    if Par(12,k) == 1;
        X_1 = [X_1,Xk];
        Y_1 = [Y_1,Yk];
    elseif Par(12,k) == 2;
        X_2 = [X_2,Xk];
        Y_2 = [Y_2,Yk];
    elseif Par(12,k) == 3;
        X_3 = [X_3,Xk];
        Y_3 = [Y_3,Yk];
    elseif Par(12,k) == 4;
        X_4 = [X_4,Xk];
        Y_4 = [Y_4,Yk];
    elseif Par(12,k) == 5;
        X_5 = [X_5,Xk];
        Y_5 = [Y_5,Yk];
    end
end
p1 = patch(X_1,Y_1,TruckColor);
p2 = patch(X_2,Y_2,MotorcycleColor);
p3 = patch(X_3,Y_3,BusColor);
p4 = patch(X_4,Y_4,HPCarColor);
p5 = patch(X_5,Y_5,LPCarColor);

%%
figure(2)
subplot(2,6,[3 4 5 6])
h = animatedline;
axis([0,15,0,2*pi])
plot(T,Thetas);
t1 = line([0 0], ylim);
grid on;hold on;
title('Vehicles position');
xlabel('Time [s]');ylabel('Position [Rad]');
%%
figure(2)
subplot(2,6,[9 10 11 12])
plot(T,Vehicles_Speed);
t2 = line([0 0], ylim);
grid on;hold on;
title('Vehicles speed');
xlabel('Time [s]');ylabel('Speed [km/h]');

%% Update simulation
Counter = 0;
for i = 1:length(T)
    if T(i) > Counter
        Counter = Counter + DeltaT;
        X_1 = [];X_2 = [];X_3 = [];X_4 = [];X_5 = [];
        Y_1 = [];Y_2 = [];Y_3 = [];Y_4 = [];Y_5 = [];
    end
end

```

```

c=1;
for k = 1:n
    [Xk,Yk] = ff.Coord2D(Par(1,k),Par(2,k),Thetas(i,k),Radius);
    if Par(12,k) == 1;
        X_1 = [X_1,Xk];
        Y_1 = [Y_1,Yk];
    elseif Par(12,k) == 2;
        X_2 = [X_2,Xk];
        Y_2 = [Y_2,Yk];
    elseif Par(12,k) == 3;
        X_3 = [X_3,Xk];
        Y_3 = [Y_3,Yk];
    elseif Par(12,k) == 4;
        X_4 = [X_4,Xk];
        Y_4 = [Y_4,Yk];
    elseif Par(12,k) == 5;
        X_5 = [X_5,Xk];
        Y_5 = [Y_5,Yk];
    end
    c = c + 1;
end
set(p1,'XData',X_1,'YData',Y_1);
set(p2,'XData',X_2,'YData',Y_2);
set(p3,'XData',X_3,'YData',Y_3);
set(p4,'XData',X_4,'YData',Y_4);
set(p5,'XData',X_5,'YData',Y_5);
set(t1,'XData',[T(i) T(i)]);
set(t2,'XData',[T(i) T(i)]);
set(Text,'String',strcat('Time:',num2str(T(i),'% 3.0f'
)));
open(vid);
f = getframe(gcf);
writeVideo(vid,f);
end
end
close(vid);

```

B.2. Traffic flow simulation 3D

```

close all
clear all
clc
%% Calling parameters in other scripts
Input_Parameters;
%% Track
Color=[68 237 169]./255;
r=Radius-2;R=Radius+2;xf=0;Xf=0;yf=0;Yf=0;

t = linspace(0,2*pi,200);
x = xf + r*cos(t);
y = yf + r*sin(t);
X = Xf + R*cos(t);
Y = Yf + R*sin(t);
%z = sin(t)*Radius/4+Radius/2;

```



```

[~,z] = Track(t,Radius,type);
z0= t*0;
%% Simulation

sim('Schematic_model_of_operation');

%% Making of animation (Figure 1)
vid = VideoWriter('sims','MPEG-4');

%% Graphics
figure(1)
plot(T,Vehicles_Acceleration);
t2 = line([0 0], ylim);
grid on;hold on;
title('Vehicles acceleration','LineWidth',1.2);
xlabel('Time [s]');ylabel('Acceleration [m/s^2]');

figure(2)

set(figure(2), 'Position', [70 150 1800 700]);
subplot(2,6,[1 2 7 8],'Color','r')
set(gca,'LooseInset',get(gca,'TightInset'))
Text = text(-Radius,-
Radius,strcat('Time:',num2str(T(1),2)),'FontSize',15);
% Track plot
patch([x X],[y Y],[z z],'k','linestyle','non');
patch([x X],[y Y],[z0 z0],'k','linestyle','non');
patch([X X],[Y Y],[z z0],'k','linestyle','non');
patch([x x],[y y],[z z0],'k','linestyle','non');
L(1) = line(x,y,z,'color',Color,'Linewidth',1);
L(2) = line(X,Y,z,'color',Color,'Linewidth',1);
axis equal
%view(3)
axis([-1.1*Radius 1.1*Radius -1.1*Radius 1.1*Radius -1.2*Radius
1.2*Radius]);
title('Traffic flow simulation','FontSize',20);
hold on;
axis off;hold on;
%Inizializating cars
X_1 = [];X_2 = [];X_3 = [];X_4 = [];X_5 = [];
Y_1 = [];Y_2 = [];Y_3 = [];Y_4 = [];Y_5 = [];
Z_1 = [];Z_2 = [];Z_3 = [];Z_4 = [];Z_5 = [];
for k = 1:n
    [Xk,Yk,Zk] =
ff.Coord3D(Par(1,k),Par(2,k),Par(3,k),Thetas(1,k),Radius,r,R,type);
    Xo = [Xk(2) Xk(3) Xk(7) Xk(6);Xk(1) Xk(2) Xk(3) Xk(4);Xk(1) Xk(4)
Xk(8) Xk(5);Xk(5) Xk(6) Xk(7) Xk(8);Xk(4) Xk(3) Xk(7) Xk(8);Xk(1) Xk(2)
Xk(6) Xk(5)];
    Yo = [Yk(2) Yk(3) Yk(7) Yk(6);Yk(1) Yk(2) Yk(3) Yk(4);Yk(1) Yk(4)
Yk(8) Yk(5);Yk(5) Yk(6) Yk(7) Yk(8);Yk(4) Yk(3) Yk(7) Yk(8);Yk(1) Yk(2)
Yk(6) Yk(5)];
    Zo = [Zk(2) Zk(3) Zk(7) Zk(6);Zk(1) Zk(2) Zk(3) Zk(4);Zk(1) Zk(4)
Zk(8) Zk(5);Zk(5) Zk(6) Zk(7) Zk(8);Zk(4) Zk(3) Zk(7) Zk(8);Zk(1) Zk(2)
Zk(6) Zk(5)];
    if Par(12,k) == 1;
        X_1 = [X_1,Xo];
        Y_1 = [Y_1,Yo];

```

```

        Z_1 = [Z_1,Zo];
elseif Par(12,k) == 2;
        X_2 = [X_2,Xo];
        Y_2 = [Y_2,Yo];
        Z_2 = [Z_2,Zo];
elseif Par(12,k) == 3;
        X_3 = [X_3,Xo];
        Y_3 = [Y_3,Yo];
        Z_3 = [Z_3,Zo];
elseif Par(12,k) == 4;
        X_4 = [X_4,Xo];
        Y_4 = [Y_4,Yo];
        Z_4 = [Z_4,Zo];
elseif Par(12,k) == 5;
        X_5 = [X_5,Xo];
        Y_5 = [Y_5,Yo];
        Z_5 = [Z_5,Zo];
    end
end
p1 = patch(X_1,Y_1,Z_1,TruckColor);
p2 = patch(X_2,Y_2,Z_2,MotorcycleColor);
p3 = patch(X_3,Y_3,Z_3,BusColor);
p4 = patch(X_4,Y_4,Z_4,HPCarColor);
p5 = patch(X_5,Y_5,Z_5,LPCarColor);
%%
figure(2)
subplot(2,6,[3 4 5 6])
h = animatedline;
axis([0,15,0,2*pi])
plot(T,Thetas);
t1 = line([0 0], ylim);
grid on;hold on;
title('Vehicles position');
xlabel('Time [s]');ylabel('Position [Rad]');
%%
figure(2)
subplot(2,6,[9 10 11 12])
plot(T,Vehicles_Speed);
t2 = line([0 0], ylim);
grid on;hold on;
title('Vehicles speed');
xlabel('Time [s]');ylabel('Speed [km/h]');

%% Update simulation
Counter = 0;
for i = 1:length(T)
    if T(i) > Counter
        Counter = Counter + DeltaT;
        X_1 = [];X_2 = [];X_3 = [];X_4 = [];X_5 = [];
        Y_1 = [];Y_2 = [];Y_3 = [];Y_4 = [];Y_5 = [];
        Z_1 = [];Z_2 = [];Z_3 = [];Z_4 = [];Z_5 = [];
        for k = 1:n
            [Xk,Yk,Zk] =
ff.Coord3D(Par(1,k),Par(2,k),Par(3,k),Thetas(i,k),Radius,r,R,type);
            Xo = [Xk(2) Xk(3) Xk(7) Xk(6);Xk(1) Xk(2) Xk(3) Xk(4);Xk(1)
Xk(4) Xk(8) Xk(5);Xk(5) Xk(6) Xk(7) Xk(8);Xk(4) Xk(3) Xk(7) Xk(8);Xk(1)
Xk(2) Xk(6) Xk(5)]';

```

```

        Yo = [Yk(2) Yk(3) Yk(7) Yk(6);Yk(1) Yk(2) Yk(3) Yk(4);Yk(1)
Yk(4) Yk(8) Yk(5);Yk(5) Yk(6) Yk(7) Yk(8);Yk(4) Yk(3) Yk(7) Yk(8);Yk(1)
Yk(2) Yk(6) Yk(5)]';
        Zo = [Zk(2) Zk(3) Zk(7) Zk(6);Zk(1) Zk(2) Zk(3) Zk(4);Zk(1)
Zk(4) Zk(8) Zk(5);Zk(5) Zk(6) Zk(7) Zk(8);Zk(4) Zk(3) Zk(7) Zk(8);Zk(1)
Zk(2) Zk(6) Zk(5)]';
        if Par(12,k) == 1;
            X_1 = [X_1,Xo];
            Y_1 = [Y_1,Yo];
            Z_1 = [Z_1,Zo];
        elseif Par(12,k) == 2;
            X_2 = [X_2,Xo];
            Y_2 = [Y_2,Yo];
            Z_2 = [Z_2,Zo];
        elseif Par(12,k) == 3;
            X_3 = [X_3,Xo];
            Y_3 = [Y_3,Yo];
            Z_3 = [Z_3,Zo];
        elseif Par(12,k) == 4;
            X_4 = [X_4,Xo];
            Y_4 = [Y_4,Yo];
            Z_4 = [Z_4,Zo];
        elseif Par(12,k) == 5;
            X_5 = [X_5,Xo];
            Y_5 = [Y_5,Yo];
            Z_5 = [Z_5,Zo];
        end
    end
    set(p1,'XData',X_1,'YData',Y_1,'ZData',Z_1);
    set(p2,'XData',X_2,'YData',Y_2,'ZData',Z_2);
    set(p3,'XData',X_3,'YData',Y_3,'ZData',Z_3);
    set(p4,'XData',X_4,'YData',Y_4,'ZData',Z_4);
    set(p5,'XData',X_5,'YData',Y_5,'ZData',Z_5);
    set(t1,'XData',[T(i) T(i)]);
    set(t2,'XData',[T(i) T(i)]);
    set(Text,'String',strcat('Time:',num2str(T(i),'% 3.0f'
)));
    open(vid);
    f = getframe(gcf);
    writeVideo(vid,f);
end
end
close(vid);

```

B.3. Vehicles and Driving behaviour paramers

```

function AutoPar=
Vehicles_and_Driving_behaviour_parameters(n_vehicles,Average_Speed,Desvia
tion_Speed,Average_Minimum_Space,Desviation_Minimum_Space,Average_Time_He
adway,Desviation_Time_Headway,Average_Vehicles_Acceleration,Desviation_Ac
celeration,Average_Vehicles_Decelaretion,Desviation_Vehicles_Deceleration
,Radius, P_1,P_2,P_3,P_4)
%% Situate vehicles
Pos_0 = 0;
Delta = 2*pi/n_vehicles;
Pos_actual = Pos_0;
for e = 1:n_vehicles
    Par(6,e) = Pos_actual;
    Pos_actual = Pos_actual + Delta;
end
%% Vehicles_parameters (Percentage distribution)
for e = 1:n_vehicles
    k = rand;
    %% Truck type
    if k < P_1
        % Vehicles_parameters
        Par(1,e) = 2.1;
        Par(2,e) = 7;
        Par(3,e) = 3.5;
        Par(4,e) = 9000;
        Par(5,e) = 125000;
        Par(12,e) = 1;
        % Driving Profile Parameters (with normal distribution)
        Par(7,e) = Desviation_Speed.*randn + Average_Speed - 20;
        Par(8,e) = Desviation_Minimum_Space.*randn +
Average_Minimum_Space + 1;
        Par(9,e) = Desviation_Time_Headway.*randn + Average_Time_Headway
+ 2;
        Par(10,e) = Desviation_Acceleration.*randn +
Average_Vehicles_Acceleration - 0.5;
        Par(11,e) = Desviation_Vehicles_Deceleration.*randn +
Average_Vehicles_Decelaretion - 1;
    %% Motorcycle type
    elseif (k >= P_1) && (k <= P_1+P_2)
        % Vehicles_parameters
        Par(1,e) = 0.8;
        Par(2,e) = 2;
        Par(3,e) = 1.1;
        Par(4,e) = 180;
        Par(5,e) = 70000;
        Par(12,e) = 2;
        % Driving Profile Parameters (with normal distribution)
        Par(7,e) = Desviation_Speed.*randn + Average_Speed;
        Par(8,e) = Desviation_Minimum_Space.*randn +
Average_Minimum_Space;
        Par(9,e) = Desviation_Time_Headway.*randn + Average_Time_Headway;
        Par(10,e) = Desviation_Acceleration.*randn +
Average_Vehicles_Acceleration;
        Par(11,e) = Desviation_Vehicles_Deceleration.*randn +
Average_Vehicles_Decelaretion;
    %% Bus type

```

```

elseif (k >= P_1+P_2) && (k <= P_1+P_2+P_3)
    % Vehicles_parameters
    Par(1,e) = 2.5;
    Par(2,e) = 10;
    Par(3,e) = 3;
    Par(4,e) = 12000;
    Par(5,e) = 330000;
    Par(12,e) = 3;
    %% Driving Profile Parameters (with normal distribution)
    Par(7,e) = Desviation_Speed.*randn + Average_Speed - 20;
    Par(8,e) = Desviation_Minimum_Space.*randn +
Average_Minimum_Space + 1;
    Par(9,e) = Desviation_Time_Headway.*randn + Average_Time_Headway
+ 2;
    Par(10,e) = Desviation_Acceleration.*randn +
Average_Vehicles_Acceleration - 0.5;
    Par(11,e) = Desviation_Vehicles_Decelaretion.*randn +
Average_Vehicles_Decelaretion - 1;
    %% High performance car
elseif (k >= P_1+P_2+P_3) && (k <= P_1+P_2+P_3+P_4)
    % Vehicles_parameters
    Par(1,e) = 1.9;
    Par(2,e) = 5;
    Par(3,e) = 1.5;
    Par(4,e) = 1900;
    Par(5,e) = 200000;
    Par(12,e) = 4;
    % Driving Profile Parameters (with normal distribution)
    Par(7,e) = Desviation_Speed.*randn + Average_Speed;
    Par(8,e) = Desviation_Minimum_Space.*randn +
Average_Minimum_Space;
    Par(9,e) = Desviation_Time_Headway.*randn + Average_Time_Headway;
    Par(10,e) = Desviation_Acceleration.*randn +
Average_Vehicles_Acceleration;
    Par(11,e) = Desviation_Vehicles_Decelaretion.*randn +
Average_Vehicles_Decelaretion;
    %% Low perfomance car
else
    % Vehicles_parameters
    Par(1,e) = 1.7;
    Par(2,e) = 3.9;
    Par(3,e) = 1.4;
    Par(4,e) = 1000;
    Par(5,e) = 70000;
    Par(12,e) = 5;
    % Driving Profile Parameters (with normal distribution)
    Par(7,e) = Desviation_Speed.*randn + Average_Speed;
    Par(8,e) = Desviation_Minimum_Space.*randn +
Average_Minimum_Space;
    Par(9,e) = Desviation_Time_Headway.*randn + Average_Time_Headway;
    Par(10,e) = Desviation_Acceleration.*randn +
Average_Vehicles_Acceleration;
    Par(11,e) = Desviation_Vehicles_Decelaretion.*randn +
Average_Vehicles_Decelaretion;
end
end
AutoPar = Par;
end

```

B.4. Input Parameters

```

%% Parametres of the vehicles
load Par;
%%
Radius = 40;
%%
P_1 = 10/100;P_2 = 12/100;P_3 = 10/100;P_4 = 30/100;
%% Type model 2=2D/3D without slope or 3=3D with slope
type= 3; % type 2 allows to avoid unnecessary calculations of 3D model
%%
% Creation of parameters involving driving profiles and vehicles
% characteristics
Number_of_Vehicles = 100;
Average_Speed = 60;                Desviation_Speed = 5;
Average_Minimum_Space = 2;         Desviation_Minimum_Space = 0.5;
Average_Time_Headway = 2;          Desviation_Time_Headway = 0.6;
Average_Vehicles_Acceleration = 1; Desviation_Acceleration = 0.1;
Average_Vehicles_Decelaretion = 4; Desviation_Vehicles_Deceleration =
0.2;

Par = Vehicles_and_Driving_behaviour_parameters(Number_of_Vehicles,...
Average_Speed,Desviation_Speed,Average_Minimum_Space,...
Desviation_Minimum_Space,Average_Time_Headway,Desviation_Time_Headway,...
Average_Vehicles_Acceleration,Desviation_Acceleration,...
Average_Vehicles_Decelaretion,Desviation_Vehicles_Deceleration,Radius);
save Par;
%%
[m,n]=size(Par); %where n is the number of cars
Width_vehicles = Par(1,:);
Length_vehicles = Par(2,:);
Height_vehicles = Par(3,:);
Mass_vehicles = Par(4,:);
Power_vehicles = Par(5,:);
Inicial_Position = Par(6,:);
Desired_velocity = Par(7,:)./3.6;
Minum_spacing = Par(8,:);
Desired_time_headway = Par(9,:);
Maximum_vehicle_acceleration = Par(10,:);
Comfortable_braking_deceleretion = Par(11,:);

%% Simulation
Initial_speed = ones(n,1)*Average_Speed./3.6/5*0;
tfinal = 10;
DeltaT = 0.4; %0.1;
%% More Vehicle parameters
R_wh = 0.26; % Tyre effective rolling radious (m)
Cd = 0.35; % Drag coeficient
canviar...
Af = Width_vehicles'.*Height_vehicles';
fr = 0.017; % Rolling resistance coeficient
rho = 1.22521; % Air density (kg/m^3)
g = 9.8; % Gravity constant (m/s^2)

```

es podria

```

%% Speed Controller
tr_v = 0.1;
alpha_control_v = log(9)/tr_v;
m = 1000;
b = 0.5;
Kp_v = alpha_control_v * m;
Kp_v = 1000;
Ki_v = 39;%alpha_control_v * b;

% Enviroment parameter
mu = 0.7; % road frictional coefficient
u_wind = 0; % Wind speed (m/s)
%% Colors used
LineColor = [11 247 239]./255;
TruckColor = [215 117 5]./255;%[114 58 35]./255;
BusColor = [255 209 0]./255;
MotorcycleColor = [252 7 202]./255;
LPCarColor = [250 43 6]./255;
HPCarColor = [30 224 37]./255;
Vehiclescolor = [250 43 6]./255;

```

B.5. Functions

```

function fh = Functions
fh.Coord2D = @Coordinates2D;
fh.Coord3D = @Coordinates3D;
fh.Tan3D = @Tangent_Vector;
fh.Norm3D = @Normal_Vector;
end

function [Xk,Yk] = Coordinates2D(h,l,theta,Radius)
%Vertex at (0,0)
V = [-h,l;h,l;h,-l;-h,-l].*0.5;
Matriu_rot = [cos(-theta),sin(-theta);-sin(-theta),cos(-theta)];
%Vertex applying orientation
Vf = (Matriu_rot*V)';
%Moving vertexs to curve
Vf2 = Vf +
Radius.*[cos(theta),sin(theta);cos(theta),sin(theta);cos(theta),sin(theta)
);cos(theta),sin(theta)];
Xk = Vf2(:,1);
Yk = Vf2(:,2);
end

function [Xk,Yk,Zk] = Coordinates3D(h,l,z,theta,Radius,r,R,type)
n = [R*cos(theta)-r*cos(theta),R*sin(theta)-r*sin(theta),0];
n = n./norm(n);
%n = Normal_Vector(theta,Radius); % track normal vector
t = Tangent_Vector(theta,Radius,type); % track tangent vector
p = cross(n,t);
V = 0.5*[-t*l-n*h-p*z;-t*l-n*h+p*z;-t*l+n*h+p*z;-t*l+n*h-p*z;+t*l-n*h-
p*z;+t*l-n*h+p*z;+t*l+n*h+p*z;+t*l+n*h-p*z];% Vehicles orientation at
(0,0,0)

```

```

V = V +
[Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type);Track(theta,Radius,type)];
V(:,3) = V(:,3) + ones(8,1)*0.5*z + 0.75; % Distance from vehicles to track added
Xk = V(:,1);
Yk = V(:,2);
Zk = V(:,3);
end

function y = Tangent_Vector(landa,Radius,type) % Obtaining tangent unit vector in a theta
theta=0;
syms theta;
d=diff(Track(theta,Radius,type));
theta = landa;
y = eval(d)./norm(eval(d));
end

function y = Normal_Vector(landa,Radius) % Obtaining tangent unit vector in a theta
theta=0;
syms theta;
d=diff(Track(theta,Radius),2);
theta = landa;
y = eval(d)./norm(eval(d));
end

```

B.6. Track

```

function [h,z] = Track(theta,Radius,type)
x = Radius*cos(theta);
y = Radius*sin(theta);
z = Radius/2+sin(3*theta)*Radius/6;
if type == 2 % type 2 allows to avoid unnecessary calculations of 3D model
z = Radius/4+sin(theta)*0.00001;
end
h = [x,y,z];
end

```

B.7. Length curve ('Interpreted Matlab Fcn' Simulink block)

```

function S_i_3D = Length_curve(a)
N = (length(a)-2)/2;
Dist=zeros(N,1);
Radius = a(end);
if a(4) == 3
f = @(Thetas)
sqrt(Radius.^2*cos(Thetas).^2+Radius.^2*sin(Thetas).^2+(Radius.^2*cos(3*Thetas).^2)/4);
Total_length = integral(f,0,2*pi);
end

```



```

    for e = 1:N
        Dist(e)= integral(f,a(e),a(e+N));
        if Dist(e) < 0                               %% Singularity because
of the model, a conversion is made
            Dist(e) = Dist(e) + Total_length;
        end
    end
end
S_i_3D=Dist;
end

```

B.8. Total length ('Interpreted Matlab Fcn' Simulink block)

```

function y = Total_Length(Radius)
f = @(Thetas)
sqrt(Radius.^2*cos(Thetas).^2+Radius.^2*sin(Thetas).^2+(Radius.^2*cos(3*Thetas).^2)/4);
y = integral(f,0,2*pi);
end

```

B.9. IDM Model ('Matlab Function' Simulink block)

```

function y = fcn(Vo,So,Th,a,b,L,v_i,s_i)

% Vo = Desired_velocity;So = Minum_spacing;T = Desired_time_headway;a =
Maximum_vehicle_acceleration;b = Comfortable_braking_deceleration

Delta = 10;
% Net distance (distance to the vehicle in front)
L_front = [L(2:length(v_i));L(1)];           % canvi per v_i
de position_i
s_i = s_i - 0.5*(L+L_front);
% Aproching rate ( velocity difference with the car in front,)
v_i_front = [v_i(2:length(v_i));v_i(1)];
Delta_V = v_i-v_i_front;
% Model definition of acceleration
Acceleration = a.*[1-((v_i./Vo).^Delta)-
((So+max(0,v_i.*Th+(v_i.*Delta_V)./(2*sqrt(a.*b)))))./s_i).^2];

for e = 1:length(Acceleration)
    if v_i(e) == 0
        Acceleration(e) = max (0,Acceleration(e));
    end
end
y=Acceleration;
end

```

B.10. Simple Model ('Matlab Function' Simulink block)

```
function y = fcn(Vo,v_i,s_i,L)
% Speed vehicle in front
v_i_front = [v_i(2:length(v_i));v_i(1)];
% Net distance (distance to the vehicle in front)
L_front = [L(2:length(v_i));L(1)];
s_i = s_i - 0.5*(L+L_front);

% Model parameters
T = 0.5;      % Safaty distance time
k = 1;       % Proporcional constant
s_o = 2;     % Minum distance
s_ref = s_o*ones(length(Vo),1) + T.*v_i;
% Velocity control algorithm
v=k*(s_i-s_ref)+v_i_front;
v=min(Vo,v);
y=v;
end
```

B.11. Slope ('Matlab Function' Simulink block)

```
function slope = fcn(thetas,Radius)
% If track is: [ Radius*cos(theta), Radius*sin(theta), Radius/2 +
(Radius*sin(4*theta))/4]
tangent = [-
Radius*sin(thetas),Radius*cos(thetas), (3*Radius*cos(3*thetas))/6];
norms = sqrt(sum(tangent.^2,2));
alpha = asin(tangent(:,3)./norms);
slope = alpha;
end
```

B.12. Conversion to horitzontal speed ('Matlab Function' Simulink block)

```
function v_h = fcn(slope,v_i)
v_h=v_i.*cos(slope);
end
```

B.13. Programa per crear els Diagrames Fonamentals

```
% Fundamental diagram
load Final_Speed;
n_max = 150;
Density = [2:n_max]/(2*200*pi/1000);
Final_Speed = [];
load N_veh;
```

```

for e = 2:n_max
    N_veh = e;
    %% Calling parameters in other scripts
    Input_Parameters;
    sim('Schematic_model_of_operation');
    Final_Speed= [Final_Speed,Average_Speed(end,1)];
end

Flux = Density.*Final_Speed;

figure(1)
scatter(Density,Final_Speed,'Filled','r');
set(gca,'FontSize',16);
title('Velocitat-Densitat');
grid on;
xlabel('Velocitat [km/h]');ylabel('Densitat Tràfic [Veh/km]');

figure(4)
scatter(Flux,Final_Speed,'Filled','r');
set(gca,'FontSize',16);
title('Flux-Velocitat');
grid on;
xlabel('Velocitat [km/h]');ylabel('Flux Tràfic [Veh/h]');

figure(3)
scatter(Density,Flux,'Filled','r');
set(gca,'FontSize',16);
title('Flux-Densitat');
grid on;
xlabel('Densitat Tràfic [Veh/km]');ylabel('Flux Tràfic [Veh/h]');

```

B.14. Programa per mostrar les propietats normals de cada paràmetre

```

%Normal properties parameters
Input_Parameters,
a_1=[];
a_2= [];

for e = 1:length(Par)
    if Par(12,e) == 1 || Par(12,e) == 3
        a_1 = [a_1,Par(8,e)];
    else
        a_2 = [a_2,Par(8,e)];
    end
end

figure(1)
subplot(3,2,1)
h1 = a_1;
h2 = a_2;
histogram(h1,25);hold on;grid on;
histogram(h2,35);
set(gca,'fontsize',14);
title('Distància mínima (s_0)');
xlabel('Distància [m]');ylabel('Número de vehicles');

```

```

%%
v_1= [];
v_2= [];

for e = 1:length(Par)
    if Par(12,e) == 1 || Par(12,e) == 3
        v_1 = [v_1,Par(7,e)];
    else
        v_2 = [v_2,Par(7,e)];
    end
end
figure(1)
subplot(3,2,2)
h1 = v_1;
h2 = v_2;
histogram(h1,30);hold on;grid on;
histogram(h2,40);
set(gca,'fontsize',14);
title('Velocitat desitjada (V_0)');
xlabel('Velocitat [m/s]');ylabel('Número de vehicles');
%%
T_1= [];
v_2= [];

for e = 1:length(Par)
    if Par(12,e) == 1 || Par(12,e) == 3
        T_1 = [T_1,Par(9,e)];
    else
        v_2 = [v_2,Par(9,e)];
    end
end
figure(1)
subplot(3,2,3)
h1 = T_1;
h2 = v_2;
histogram(h1,20);hold on;grid on;
histogram(h2,20);
set(gca,'fontsize',14);
title('Temps de separació (T)');
xlabel('Temps [s]');ylabel('Número de vehicles');
%%
a_1=[];
a_2= [];

for e = 1:length(Par)
    if Par(12,e) == 1 || Par(12,e) == 3
        a_1 = [a_1,Par(11,e)];
    else
        a_2 = [a_2,Par(11,e)];
    end
end

figure(1)
subplot(3,2,4)
h1 = a_1;
h2 = a_2;
histogram(h1,25);hold on;grid on;

```

```
histogram(h2,35);
set(gca,'fontsize',14);
title('Desacceleració màxima desitjada (b)');
xlabel('Desacceleració [m/s^2]');ylabel('Número de vehicles');
%%
a_1=[];
a_2= [];

for e = 1:length(Par)
    if Par(12,e) == 1 || Par(12,e) == 3
        a_1 = [a_1,Par(10,e)];
    else
        a_2 = [a_2,Par(10,e)];
    end
end

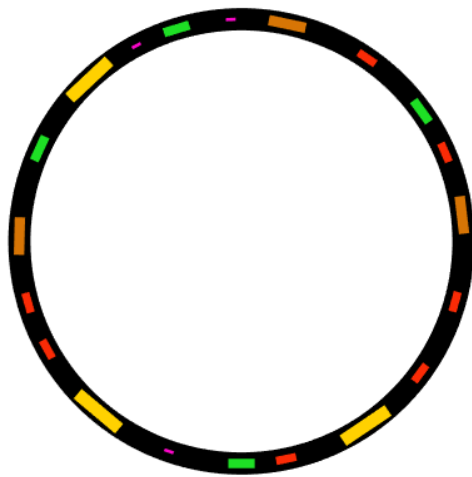
figure(1)
subplot(3,2,5)
h1 = a_1;
h2 = a_2;
histogram(h1,30);hold on;grid on;
histogram(h2,40);
set(gca,'fontsize',14);
title('Acceleració màxima desitjada (a)');
xlabel('Acceleració [m/s^2]');ylabel('Número de vehicles');
```

C. Gràfiques de les simulacions

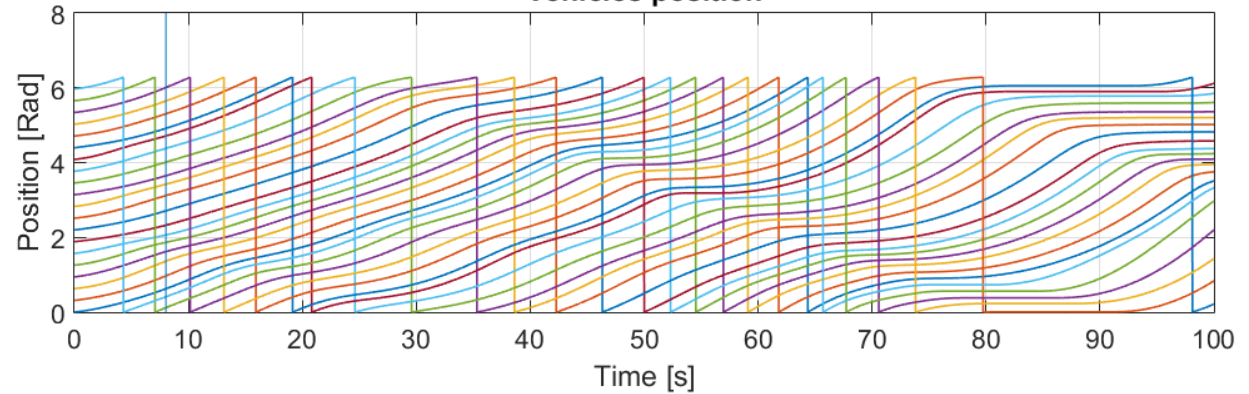
Simulació 7.1, Model IDM:

Traffic flow simulation

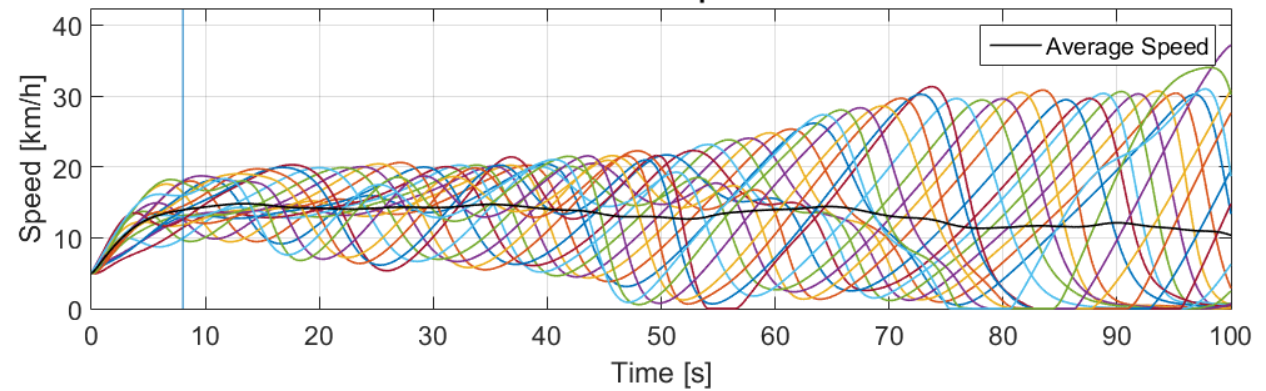
Time:8



Vehicles position



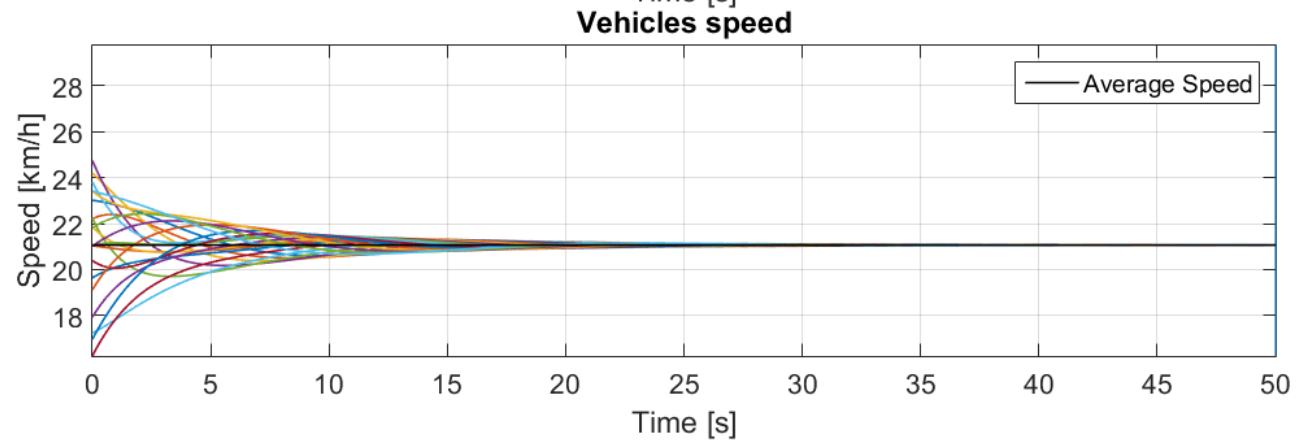
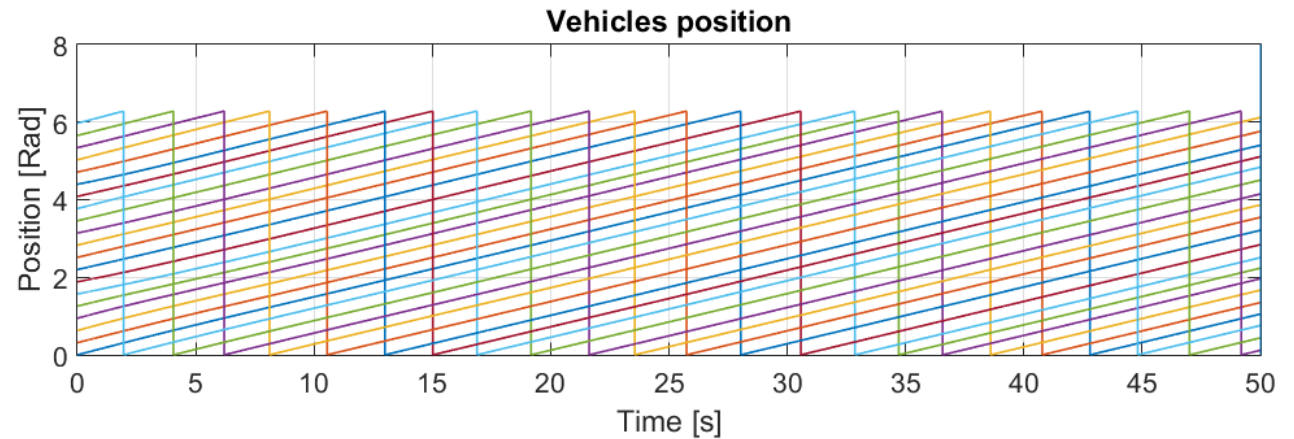
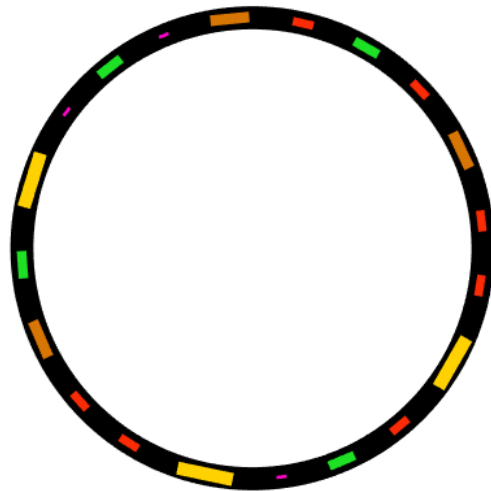
Vehicles speed



Simulació 7.2, Model Simple:

Traffic flow simulation

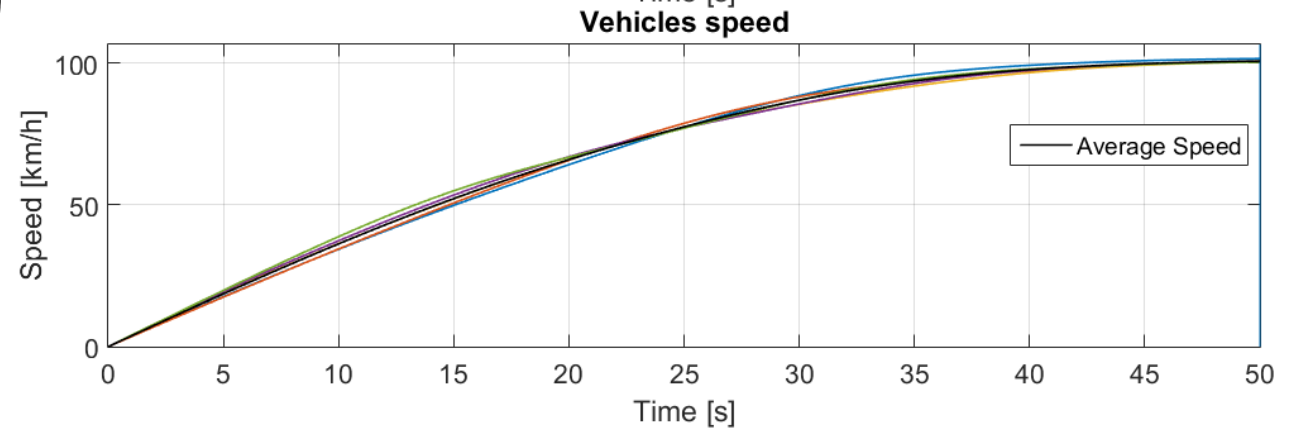
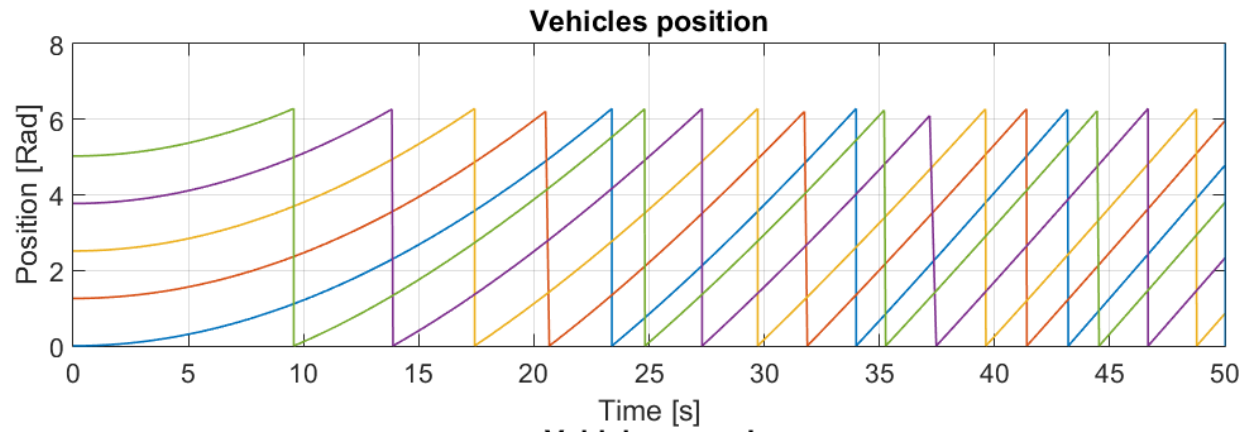
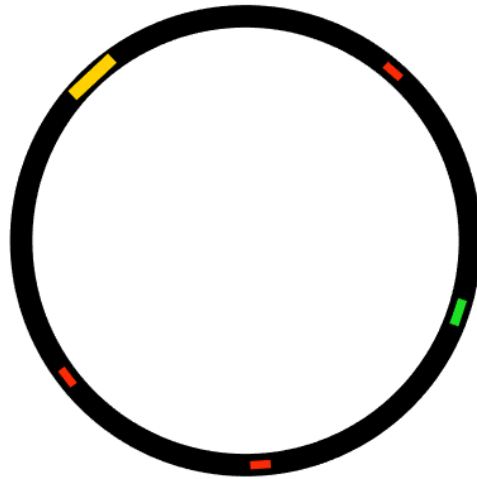
Time:50



Simulació 7.3, Nveh = 5:

Traffic flow simulation

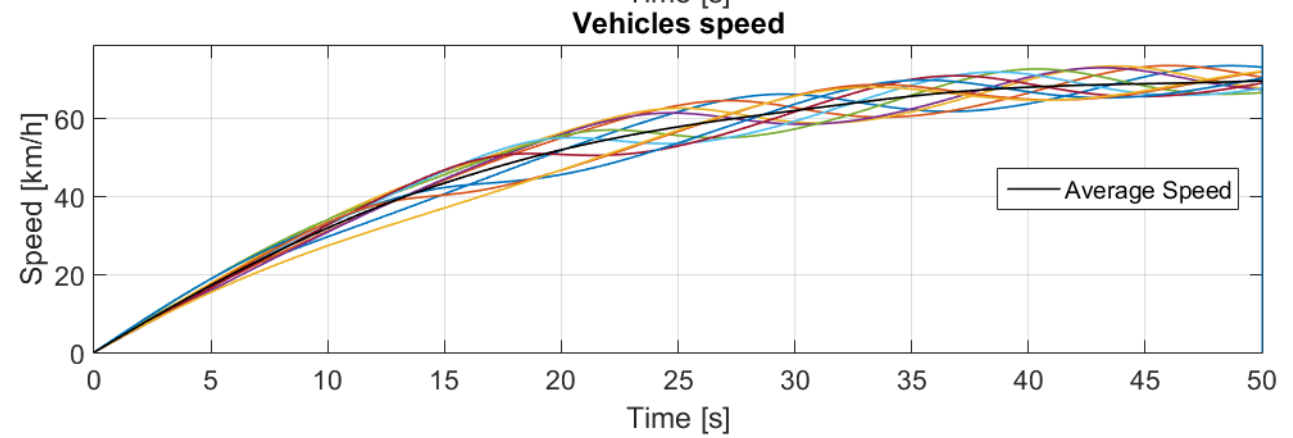
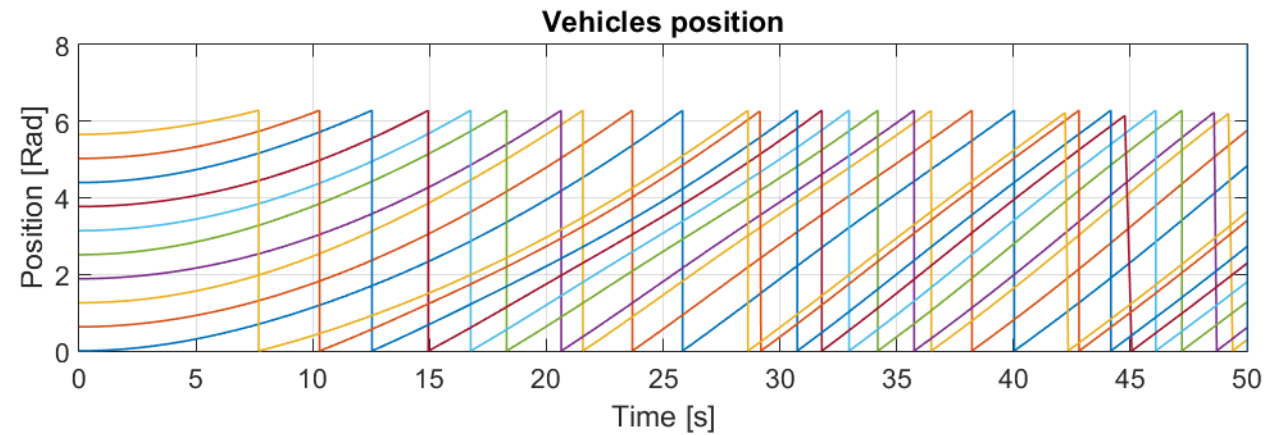
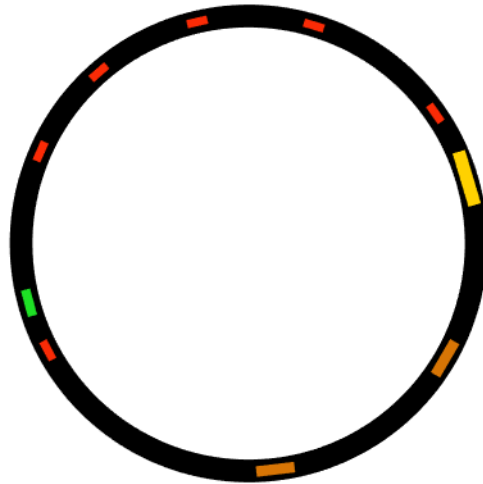
Time:



Simulació 7.3, Nveh = 10:

Traffic flow simulation

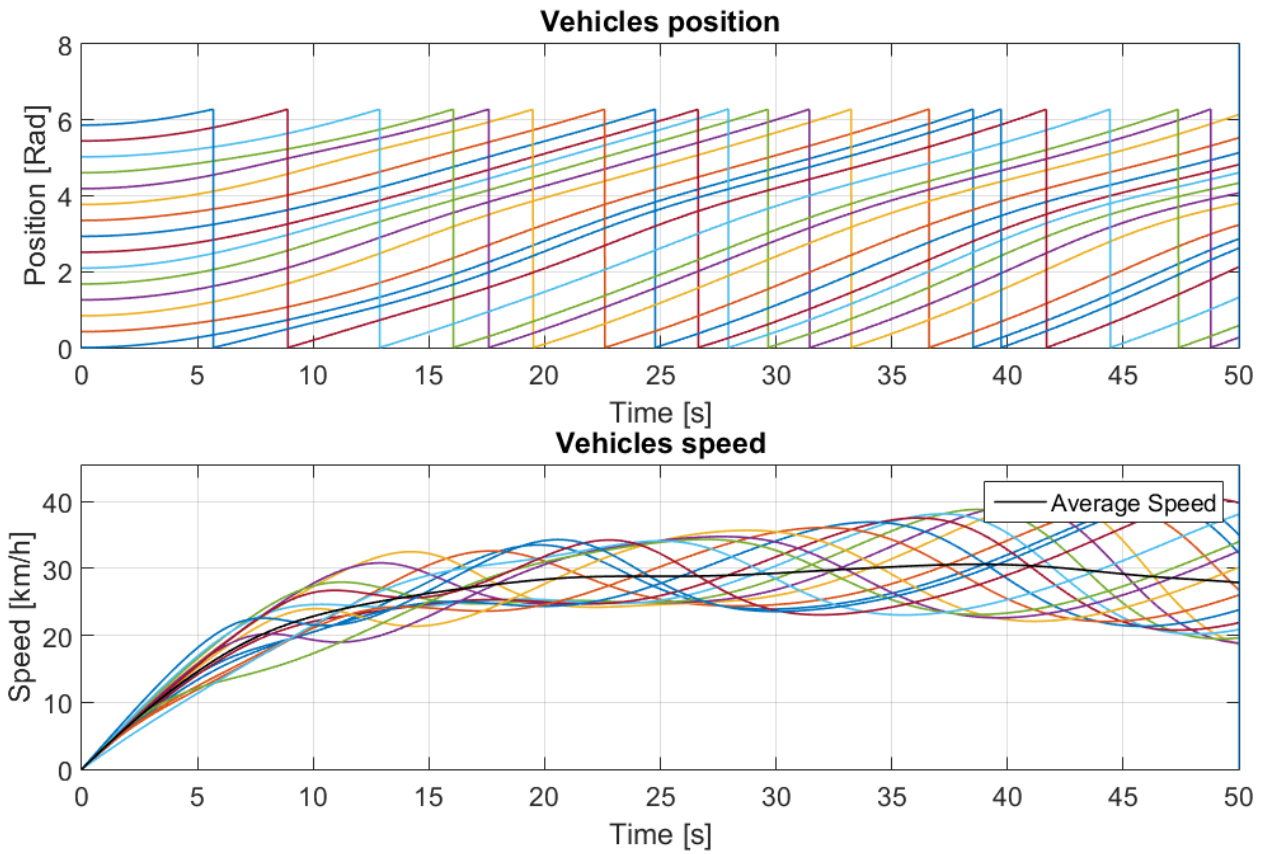
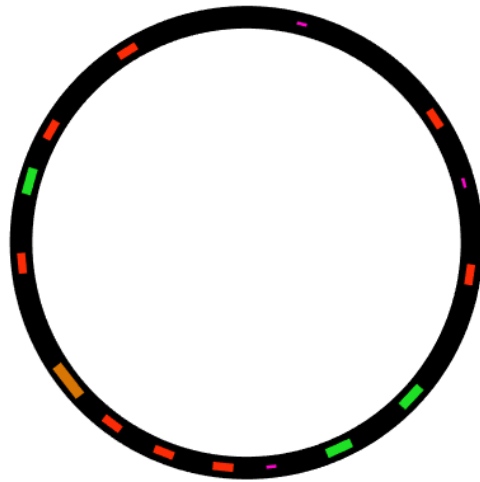
Time:



Simulació 7.3, Nveh = 15:

Traffic flow simulation

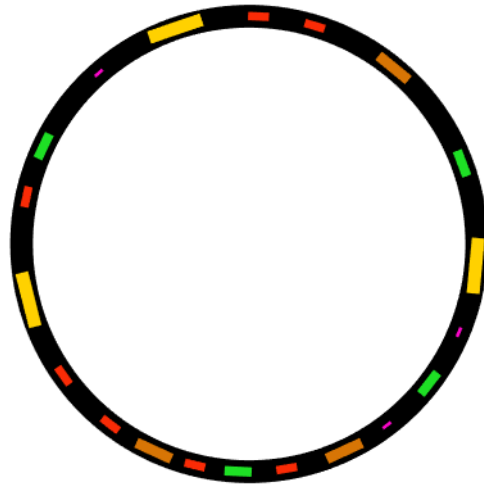
Time:



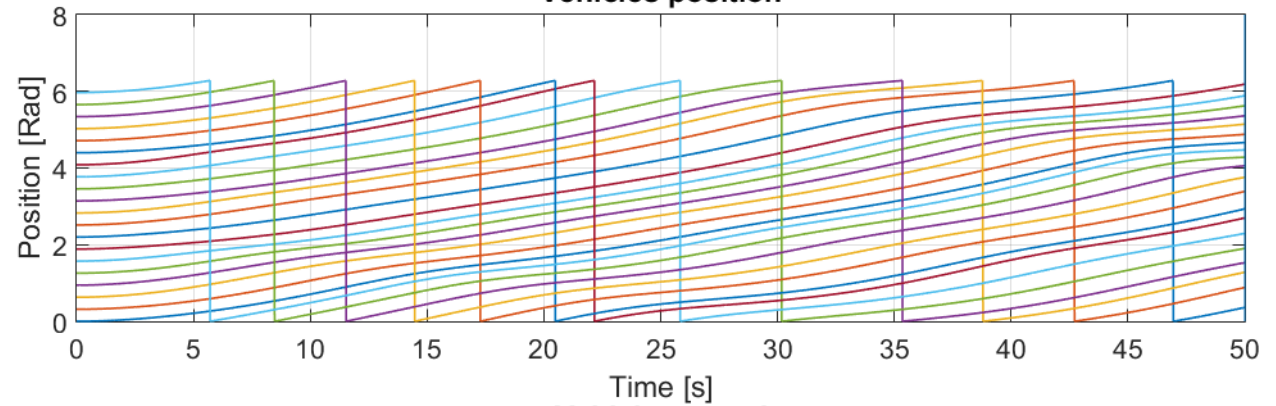
Simulació 7.3, Nveh = 20:

Traffic flow simulation

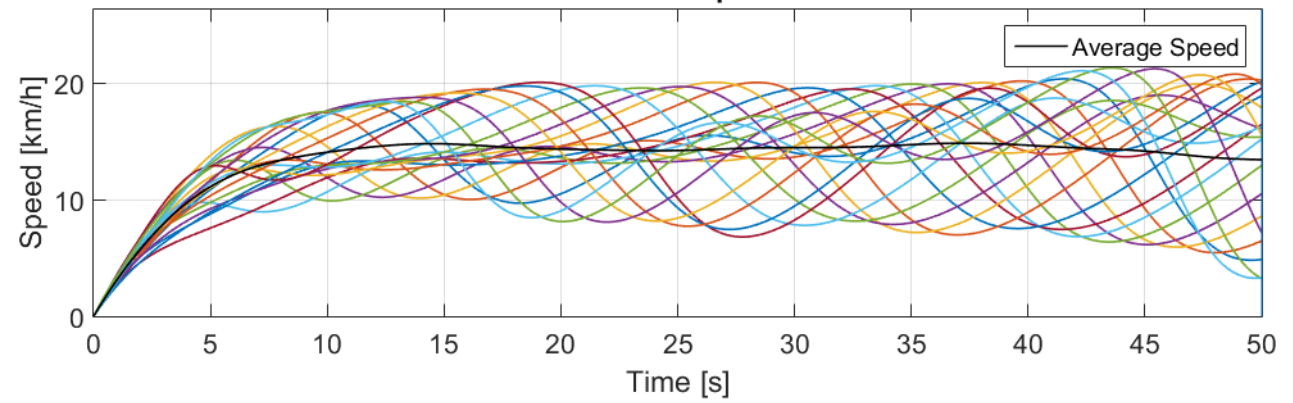
Time:



Vehicles position



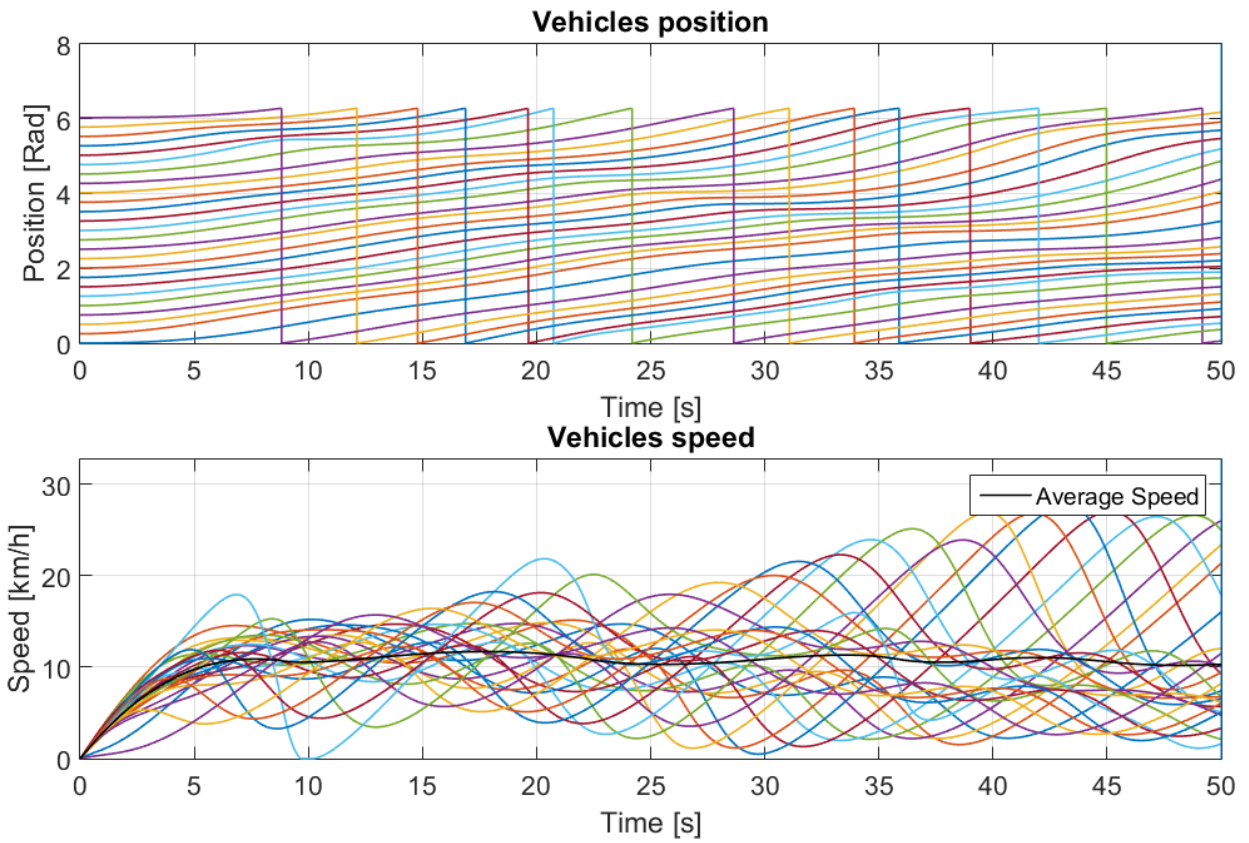
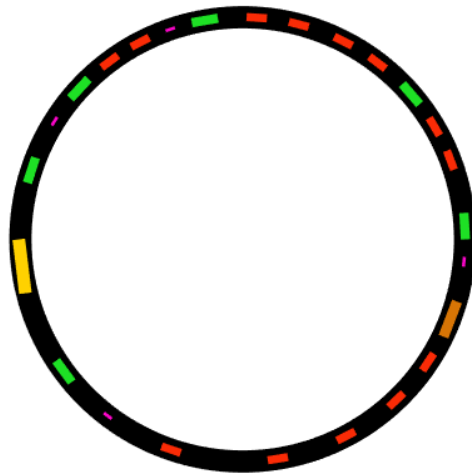
Vehicles speed



Simulació 7.3, Nveh = 25:

Traffic flow simulation

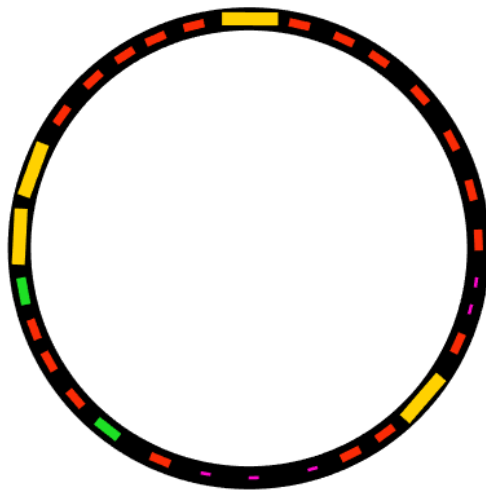
Time:



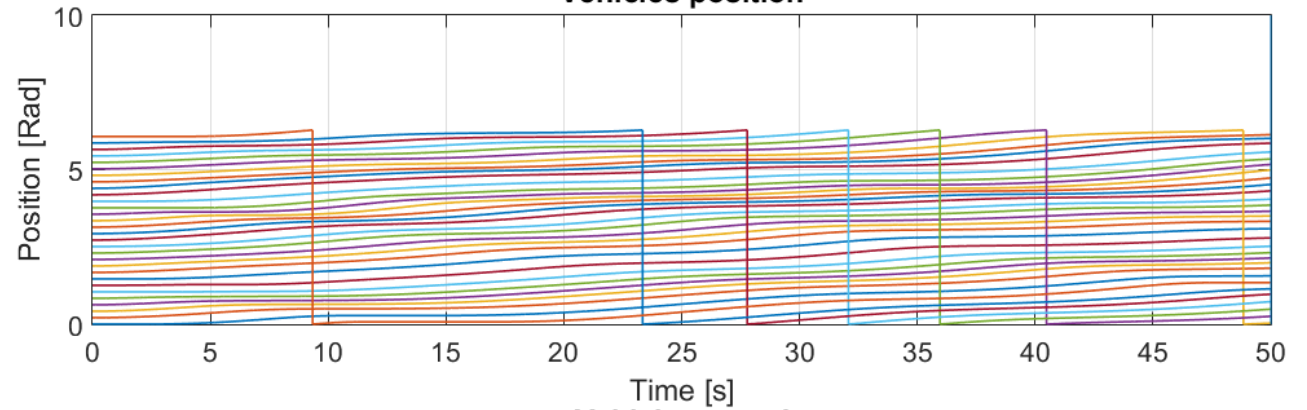
Simulació 7.3, Nveh = 30:

Traffic flow simulation

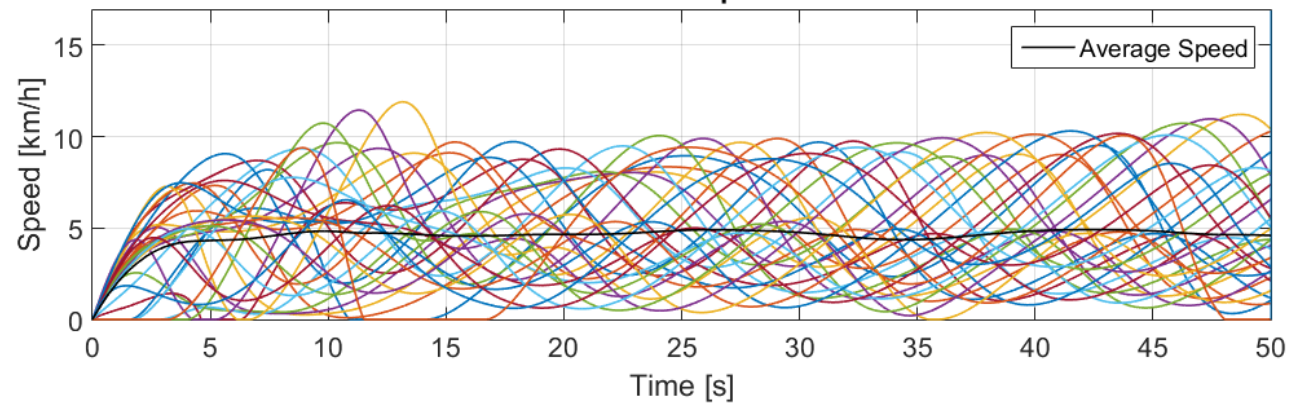
Time:



Vehicles position



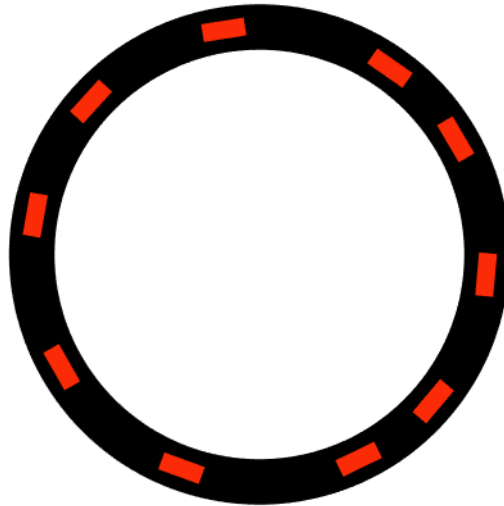
Vehicles speed



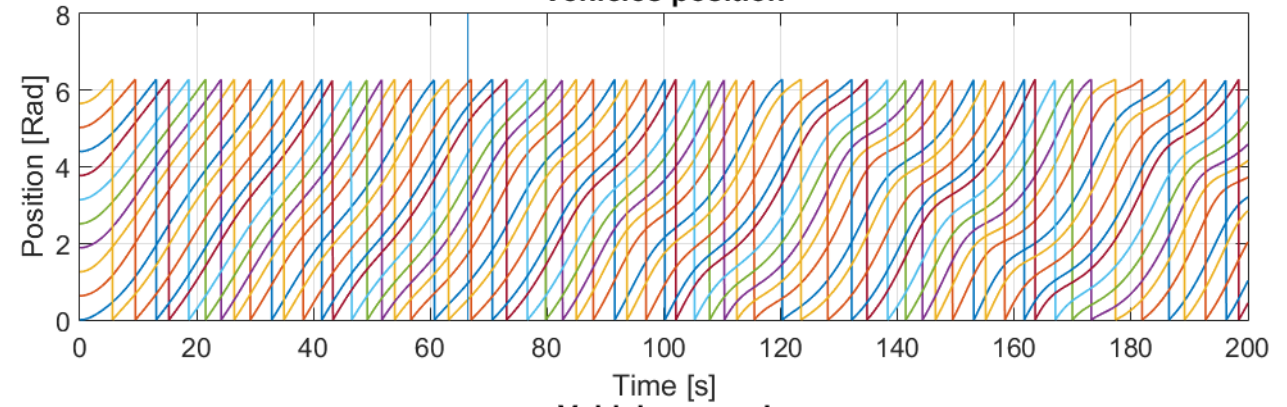
Simulació 7.4, Cas de referència:

Traffic flow simulation

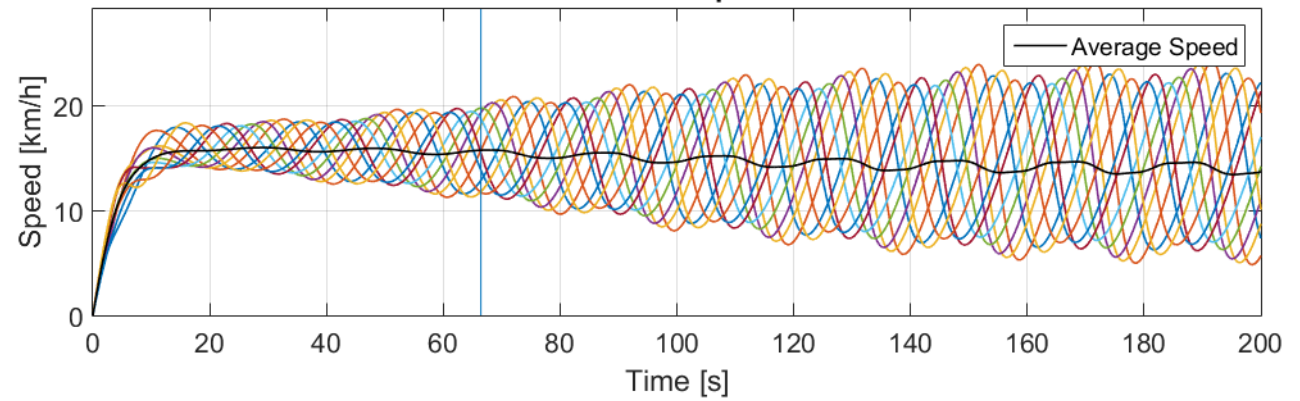
Time:66



Vehicles position



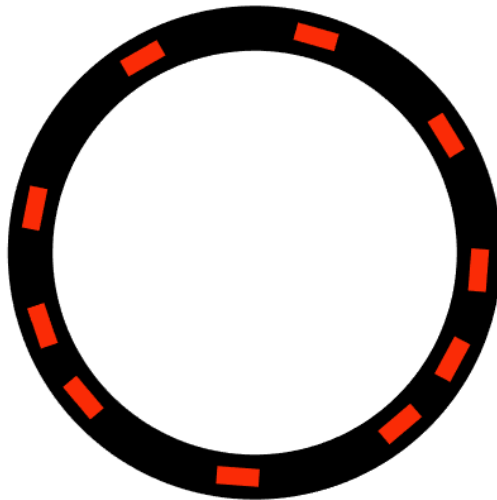
Vehicles speed



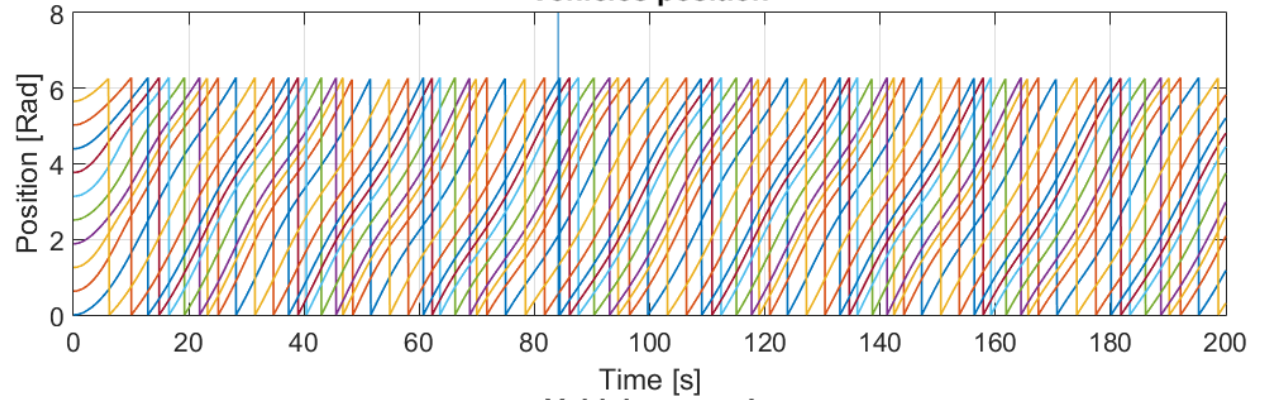
Simulació 7.4, Cas amb $s_0 = 0,5$:

Traffic flow simulation

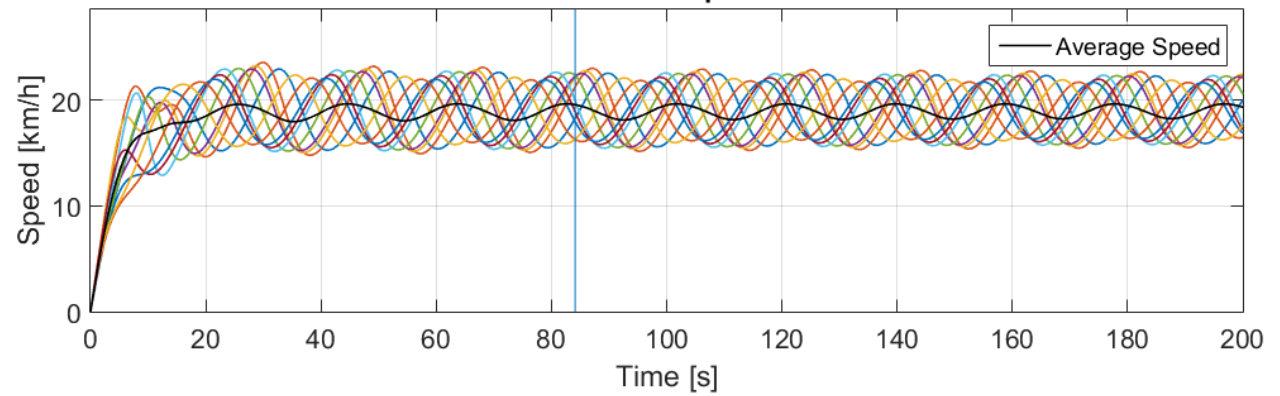
Time:84



Vehicles position



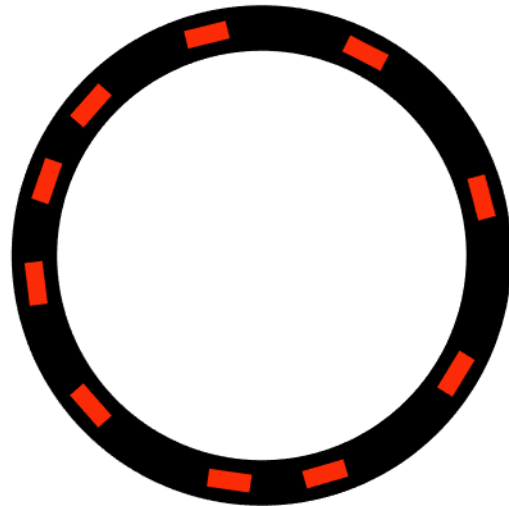
Vehicles speed



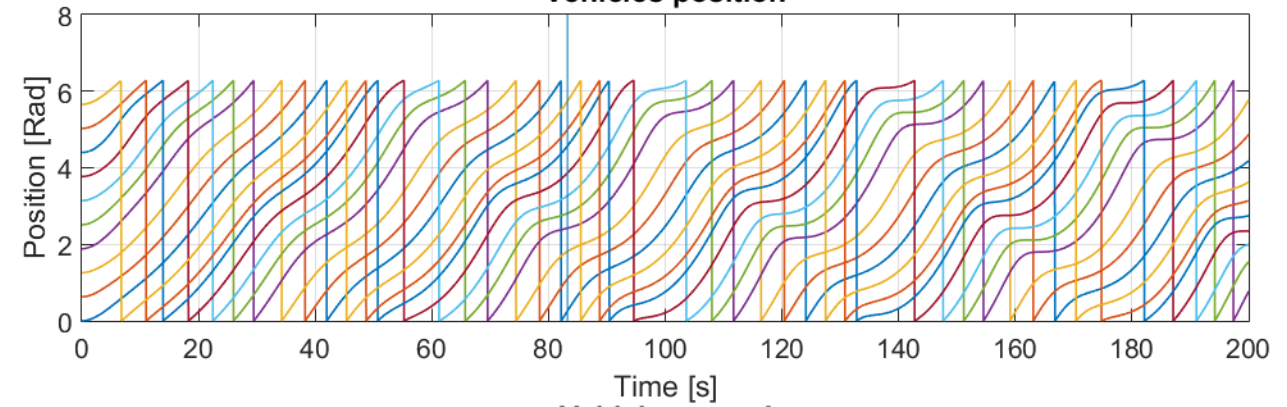
Simulació 7.4, Cas amb $s_0 = 3,5$:

Traffic flow simulation

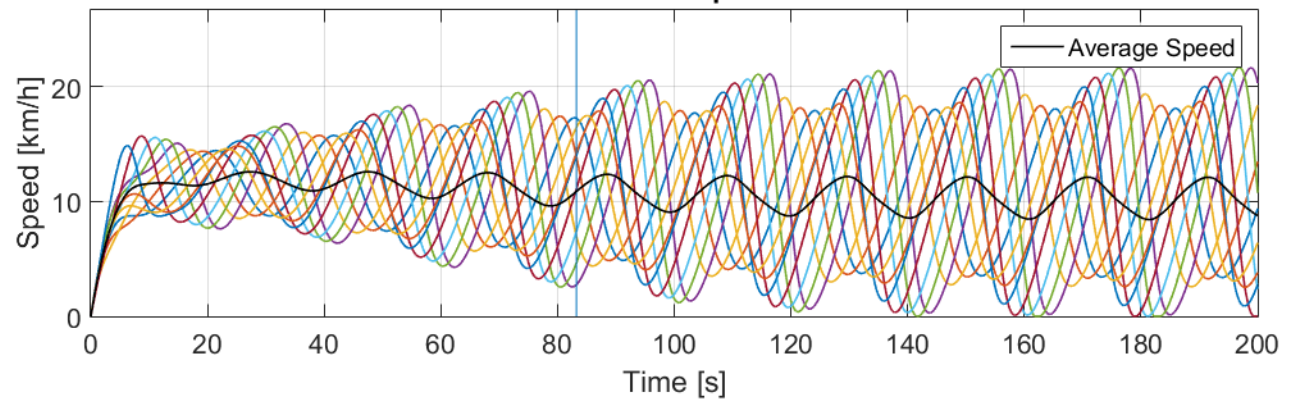
Time:83



Vehicles position



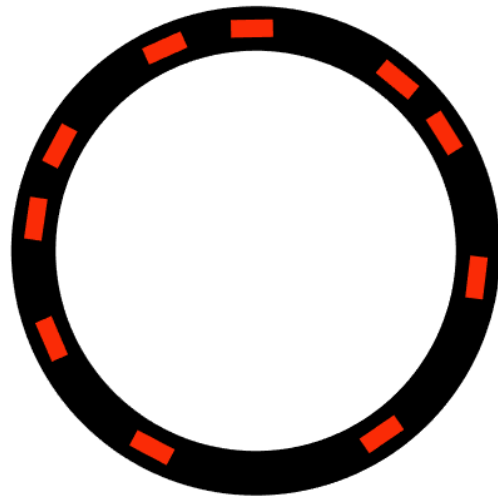
Vehicles speed



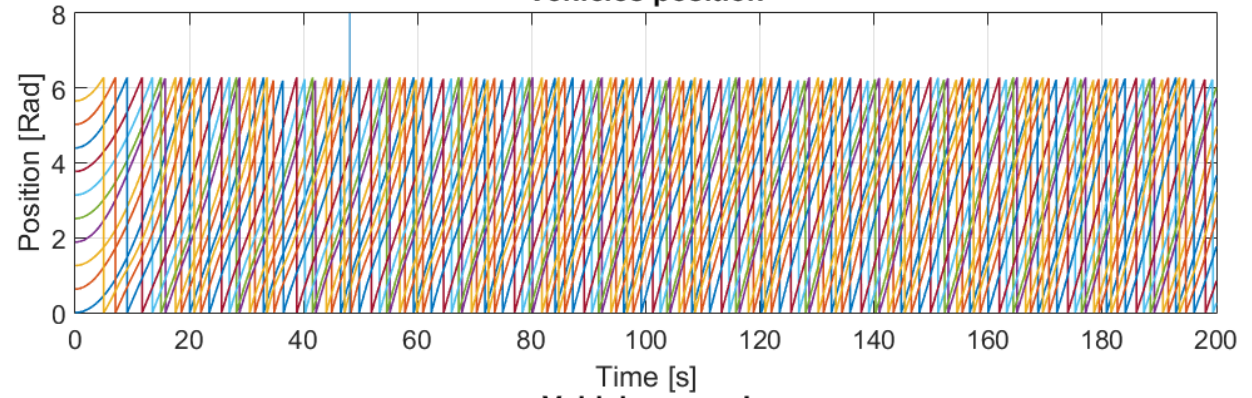
Simulació 7.4, Cas amb $T = 0,5$:

Traffic flow simulation

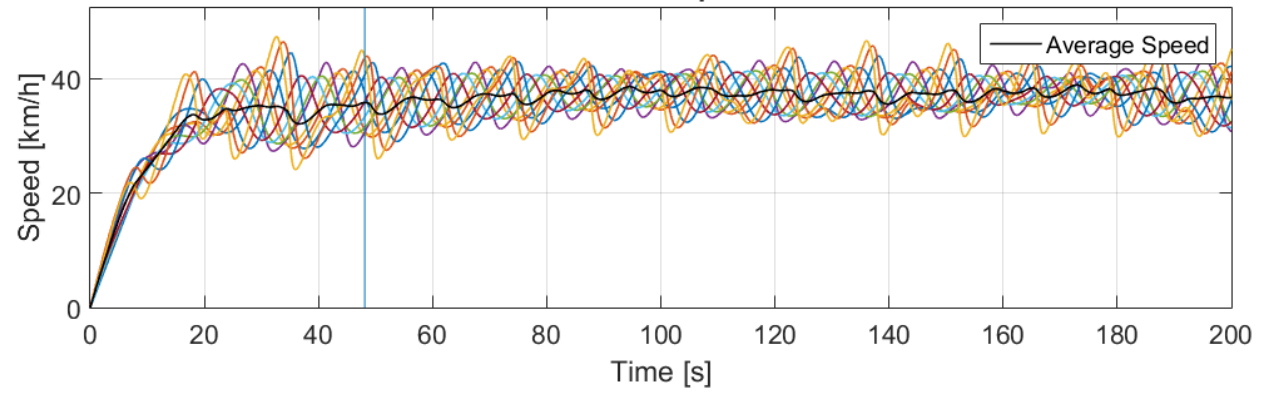
Time:48



Vehicles position



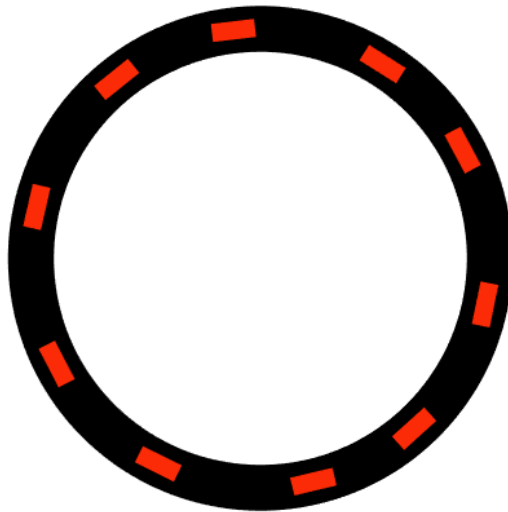
Vehicles speed



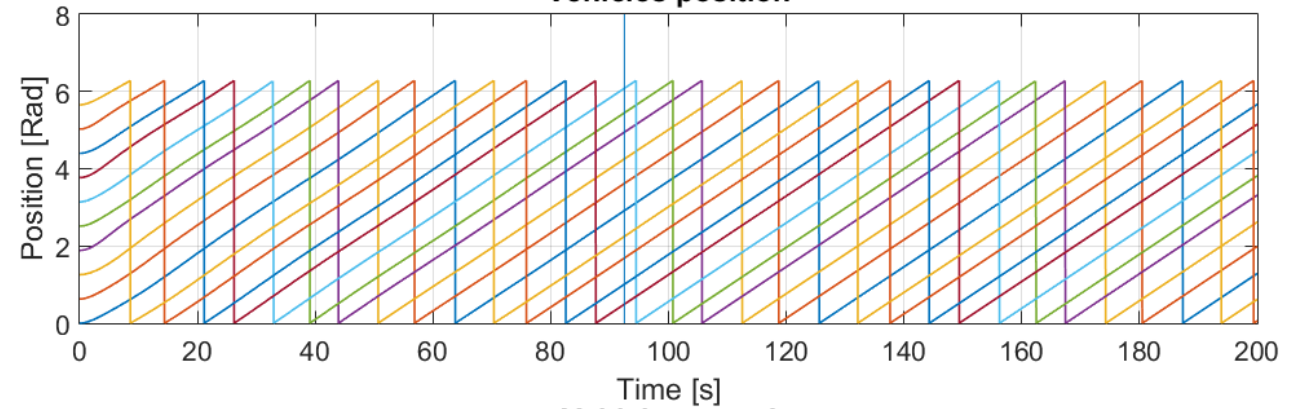
Simulació 7.4, Cas amb $T = 3$:

Traffic flow simulation

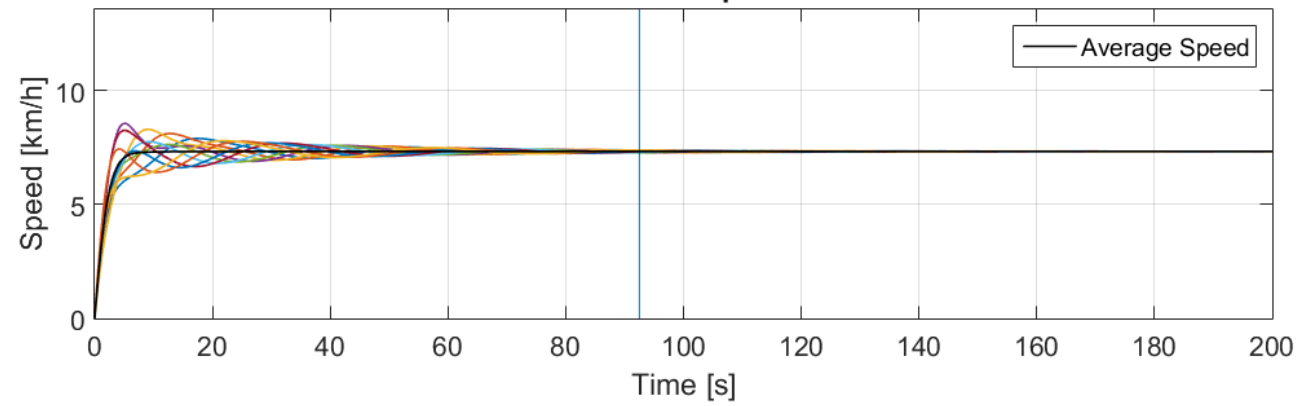
Time:93



Vehicles position



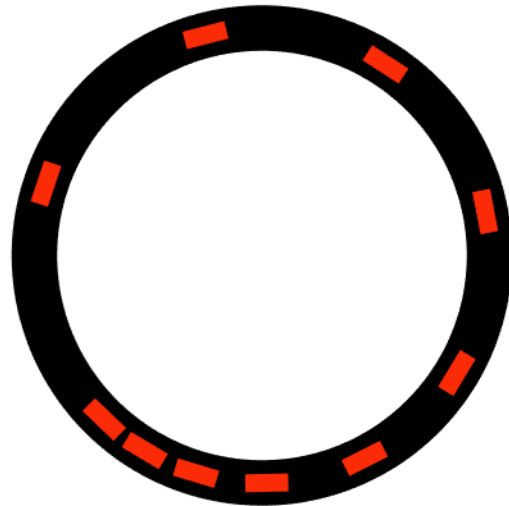
Vehicles speed



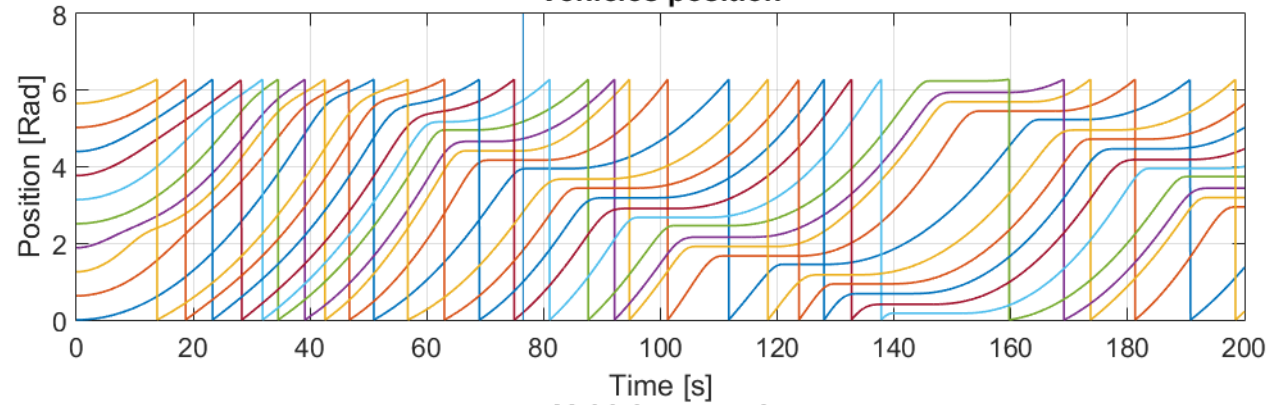
Simulació 7.4, Cas amb $\alpha = 0,2$:

Traffic flow simulation

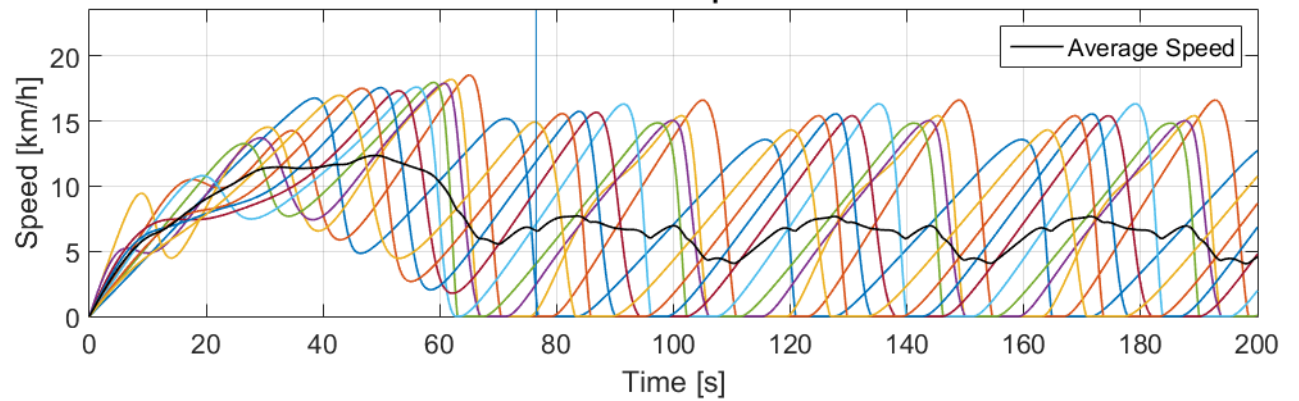
Time:76



Vehicles position



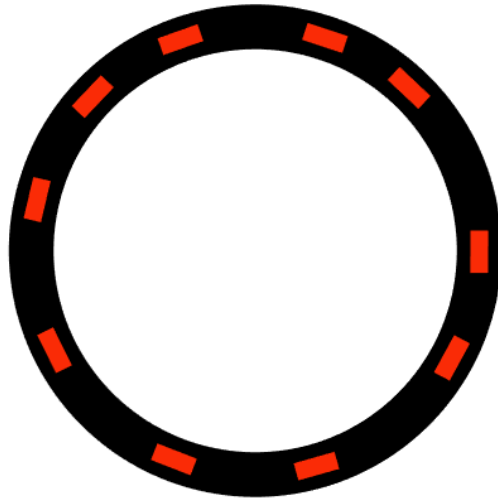
Vehicles speed



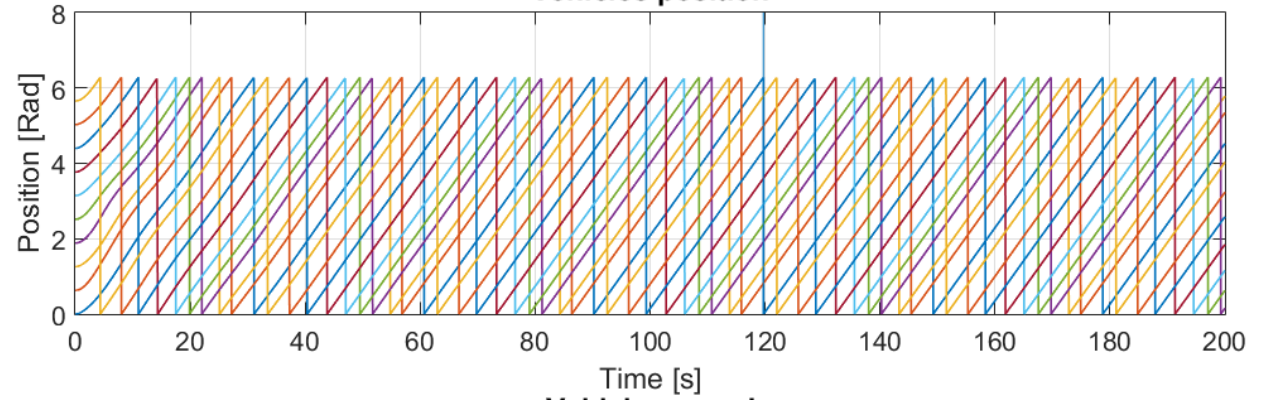
Simulació 7.4, Cas amb $\alpha = 3$:

Traffic flow simulation

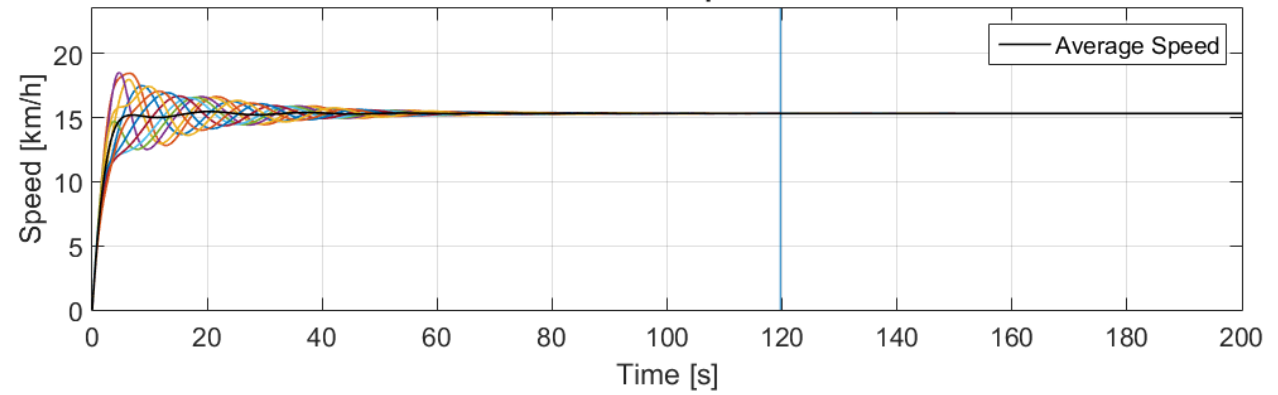
Time: 120



Vehicles position



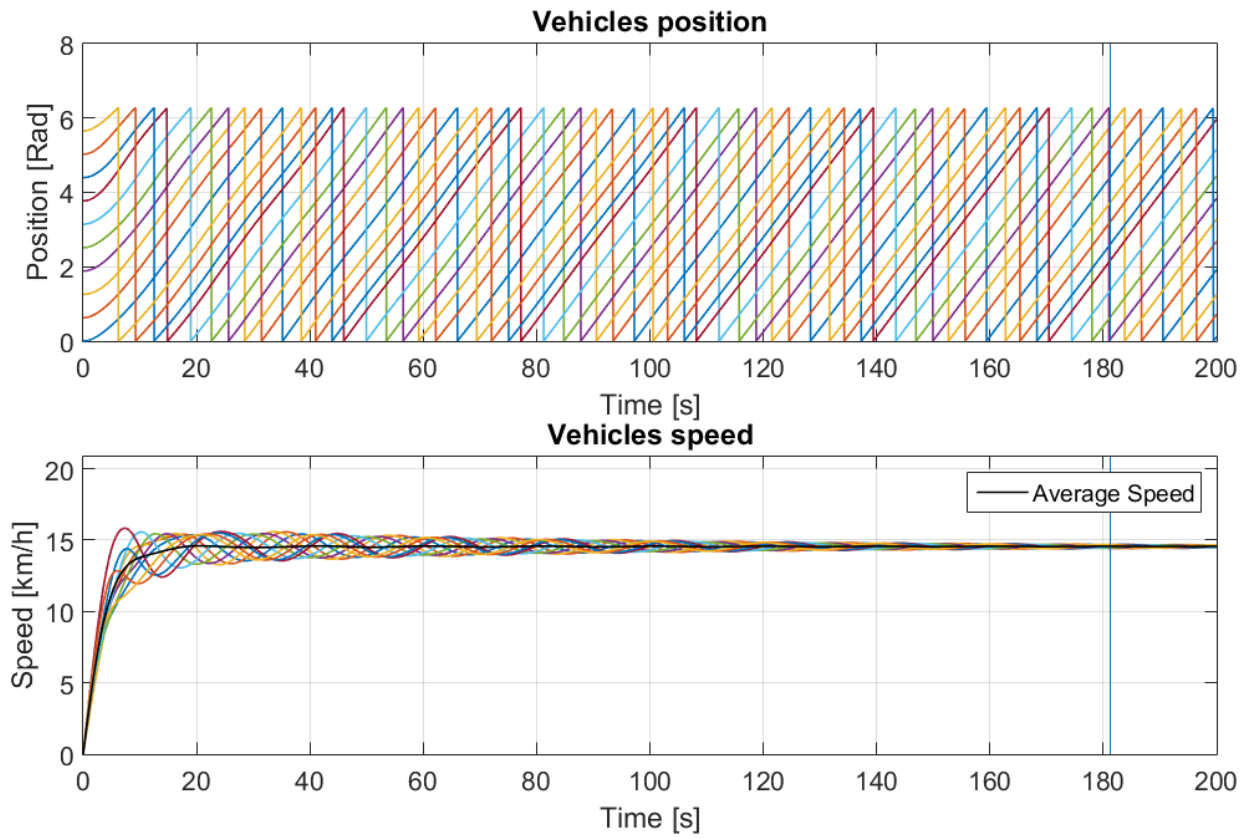
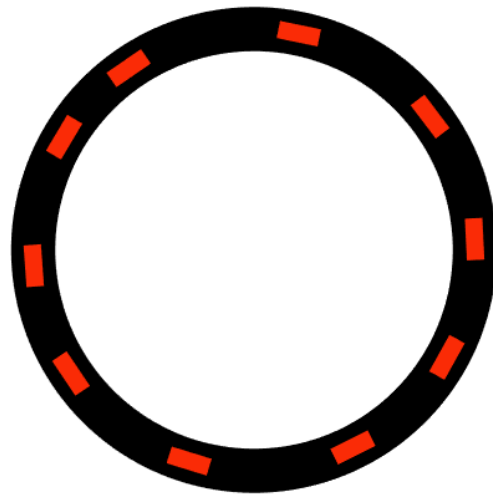
Vehicles speed



Simulació 7.4, Cas amb $b = 2$:

Traffic flow simulation

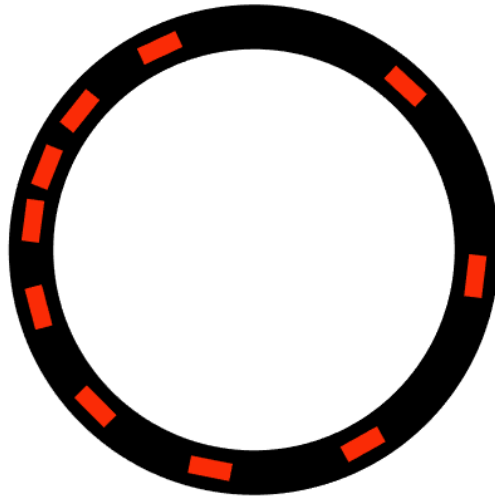
Time:181



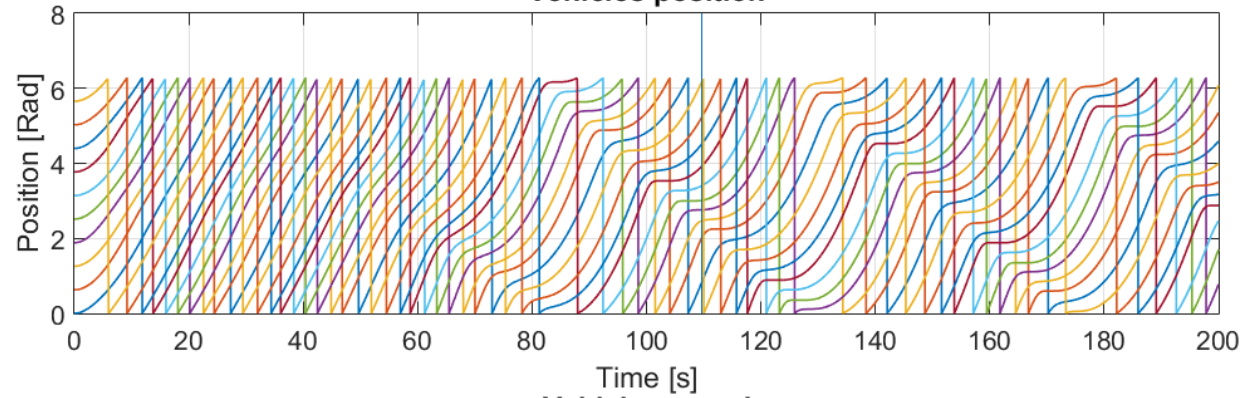
Simulació 7.4, Cas amb $b = 10$:

Traffic flow simulation

Time: 110



Vehicles position



Vehicles speed

