



School of Production Engineering and Management

Dynamic Systems and Simulation Laboratory

Analysis of Cruise Control Systems for Autonomous Vehicles on Lane-Free Roads with Sampled-Data and Saturated Control Inputs

Ανάλυση συστημάτων ελέγχου πορείας για αυτόνομα οχήματα σε δρόμους απαλλαγμένους από λωρίδες με διακριτή δειγματοληψία και κορεσμένες εισόδους

DIPLOMA THESIS

by

Filippos Tzortzoglou

Supervisor:

Prof. Ioannis Papamichail

Committee Members:

Prof. Dimitrios Ipsakis, Prof. Eleftherios Doitsidis

Chania, Greece

Summer 2022

Acknowledgements

I am so thankful to my labmate, Dr. Dionysios Theodosis, for his participation throughout my journey of this thesis. The substantive comments and his real contribution constituted a reference point for me. In addition, he was always available at any time to respond immediately to any issue. The results of Figure 5.1 and the proofs of the Lemmas 1 and 2 have been provided by him. Besides Dionysios, I wish to extend my gratitude to my supervisor Prof. Ioannis Papamichail for his contribution and comments and for motivating me to continue in doctoral studies. Thereby, this thesis is dedicated both to Dionysios and Ioannis. Also, I want to thank Prof. Markos Papageorgiou for giving me the opportunity to be part of the leading DSSL laboratory. Finally, I want to thank my family, my friends and the Unboxholics team for their support.

A part of this diploma thesis has been published at the 30th Mediterranean Conference on Control and Automation:

Theodosis, D., Tzortzoglou, F.N., Karafyllis, I., Papamichail, I., Papageorgiou, M.: Sampled-data controllers for autonomous vehicles on lane-free roads. *Mediterranean Conference on Control and Automation (MED 2022)*, Athens, Greece, June 28 - July 1, 2022, pp. 103-108.

Εκτεταμένη περίληψη στα Ελληνικά | Extensive abstract in Greek

Ο αυτοματισμός των οχημάτων έχει σημειώσει τεράστια πρόοδο τις τελευταίες δεκαετίες και η πορεία προς την πλήρη αυτοματοποίηση φαίνεται αναπόφευκτη. Ένα αρχικό στάδιο αυτοματισμού του οχήματος είναι το σύστημα ελέγχου ταχύτητας που διατηρεί την ταχύτητα του οχήματος στην επιθυμητή τιμή για να βοηθήσει τον οδηγό. Έπειτα αναπτύχθηκαν διάφορες καινοτομίες κατά τις οποίες η ταχύτητα εξαρτάται από την απόσταση του οχήματος από το προπορευόμενο ή το επόμενο όχημα. Ακόμη, πρόσφατες έρευνες έχουν αναδείξει τα πλεονεκτήματα της χρήσης συνδεδεμένων οχημάτων όπου τα αυτοκίνητα έχουν την δυνατότητα να επικοινωνούν μεταξύ τους μειώνοντας την κυκλοφοριακή συμφόρηση και αυξάνοντας παράλληλα την χωρητικότητα των δρόμων.

Οι περισσότερες εκ των νέων αυτών μεθοδολογιών υποθέτουν ότι τα οχήματα οφείλουν να εναρμονίζονται με την οριζόντια σήμανση στους δρόμους όσον αφορά τις λωρίδες. Η αλλαγή λωρίδας όμως αποτελεί σύμφωνα με μελέτες μια επικίνδυνη διαδικασία στην οποία αποδίδεται ένα σημαντικό ποσοστό των ατυχημάτων στις μέρες μας.

Παρόλα αυτά, τα τελευταία χρόνια, υπήρξαν ακόμη πιο καινοτόμες έρευνες οι οποίες βασίστηκαν σε δρόμους απαλλαγμένους από λωρίδες. Ως εκ τούτου, τα οχήματα σε αυτήν την περίπτωση δεν οφείλουν να υπακούνε σε κανένα περιορισμό όσον αφορά τις λωρίδες επεκτείνοντας έτσι τις δυνατότητες τους. Όμως, οι προαναφερθείσες έρευνες, δεν εξασφαλίζουν 1) αποφυγή συγκρούσεων μεταξύ των οχημάτων ή με το όριο του δρόμου, 2) ότι οι ταχύτητες παραμένουν πάντα θετικές και 3) ότι οι ταχύτητες παραμένουν στα πλαίσια των επιτρεπόμενων ορίων. Πέρα όμως από την μη ύπαρξη λωρίδων, μία ακόμη καινοτομία η οποία μπορεί να αυξήσει την κυκλοφοριακή ροή και την κυκλοφορία στους δρόμους είναι αυτή της “εικονικής ώθησης” κατά την οποία τα οχήματα έχουν την δυνατότητα να ασκούν μια εικονική δύναμη προς τα υπόλοιπα.

Σε αυτή τη διπλωματική εργασία μελετήθηκε ένα σύστημα ελέγχου ταχύτητας για αυτόνομα οχήματα σε δρόμους απαλλαγμένους από λωρίδες υιοθετώντας παράλληλα την αρχή της εικονικής ώθησης.

Αρχικά, στα προκαταρκτικά στο Κεφάλαιο 2 γίνεται αναφορά στο συνεχές ποδηλατικό μοντέλο και αποδεικνύεται ο τρόπος με τον οποίο εξάγονται οι μη γραμμικές διαφορικές εξισώσεις οι οποίες το προσδιορίζουν, όταν λαμβάνεται ως σημείο αναφοράς το πίσω μέρος του οχήματος. Ακόμα γίνεται αναφορά, μέσω ενός γενικού παραδείγματος, στον τρόπο μετατροπής ενός συνεχούς μοντέλου σε διακριτό, με την χρήση δειγματοληψίας.

Στο Κεφάλαιο 3 γίνεται αναπαράσταση του συνεχούς ποδηλατικού μοντέλου και των εξισώσεων που το περιγράφουν. Ταυτόχρονα, εισάγονται σημαντικά στοιχεία προκειμένου να οριστεί το πρόβλημα διακριτού χρόνου, όπως ο τρόπος υπολογισμού των αποστάσεων μεταξύ των οχημάτων, οι περιορισμοί που υπάρχουν και το σύνολο όλων των δυνατών καταστάσεων του συστήματος.

Επιπλέον, παρουσιάζονται τα χαρακτηριστικά των συναρτήσεων οι οποίες συνδράμουν στο φαινόμενο της εικονικής ώθησης το οποίο λαμβάνει χώρα μεταξύ των αυτοκινήτων αυτών καθ' αυτών αλλά και μεταξύ των αυτοκινήτων με το σύνορο του δρόμου. Επίσης αναλύονται τα χαρακτηριστικά της συνάρτησης η οποία καθορίζει την βαρύτητα που θα έχει στο σύστημα η προσμέτρηση της ταχύτητας των γειτονικών αυτοκινήτων. Συνεχίζοντας, στο Κεφάλαιο 3 παρουσιάζεται η συνάρτηση η οποία προσδιορίζει την ενέργεια του συστήματος από την οποία εξάγονται οι ελεγχτές σύμφωνα με την εργασία [9]. Η ενέργεια του συστήματος προσδιορίζει πόσο κοντά βρίσκεται το σύστημα σε σύγκλιση. Ως εκ τούτου, μεγάλες τιμές τις αρχικής ενέργειας εκπροσωπούν αρχικές συνθήκες με σημαντικές διαφορές από τις τιμές σύγκλισης του συστήματος, ενώ όταν παρατηρούνται μετριασμένες αποκλίσεις από τις τιμές σύγκλισης, τότε το σύστημα προσδιορίζεται από μικρές τιμές αρχικής ενέργειας. Οι ελεγχτές είναι σχεδιασμένοι έτσι ώστε οι τιμές της συνάρτησης ενέργειας να πλησιάζουν το μηδέν όσο εκτελείτε η εκάστοτε προσομοίωση. Αποτέλεσμα τούτου, είναι η συνάρτηση ενέργειας να αποτελεί μία γνησίως φθίνουσα συνάρτηση. Τέλος, στο Κεφάλαιο 3 παρουσιάζονται η κατάσταση του προβλήματος και το Θεώρημα 1 μέσα από το οποίο επιβεβαιώνεται η ύπαρξη λύσης στο πρόβλημα κάτω από συγκεκριμένες συνθήκες.

Στο Κεφάλαιο 4 πραγματοποιήθηκε η μετατροπή του μοντέλου συνεχούς χρόνου σε ένα σύστημα διακριτού χρόνου με το επιπλέον χαρακτηριστικό ότι οι τιμές των σημάτων εισόδου είναι διακριτοποιημένες μέσω δειγματοληψίας. Πιο συγκεκριμένα κατά την φάση της ολοκλήρωσης του συνεχούς μοντέλου σε ένα αυθαίρετά επιλεγμένο διάστημα $[0, T]$, υιοθετήθηκε η αρχή ότι οι εισοδοί του συστήματος είναι σταθερές. Τα αποτελέσματα αυτής της διακριτοποίησης παρουσιάζονται στις σχέσεις (4.2) και (4.3), ενώ ο τρόπος με τον οποίο έλαβε χώρα παρουσιάζεται στο παράρτημα. Στο ίδιο Κεφάλαιο φαίνονται συγκεκριμένα αποτελέσματα προσομοιώσεων όπου αναδεικνύεται η σημαντικότητα της επιλογής της περιόδου δειγματοληψίας (η οποία συμβολίζεται με T) και πως αυτή μπορεί να επηρεάσει το σύστημα οδηγώντας το σε σύγκρουση.

Σε δεύτερη φάση χρησιμοποιούνται αποκεντρωμένοι ελεγχτές από την εργασία [9] τροποποιημένοι με τέτοιο τρόπο ώστε να έχουν την δυνατότητα να προσμετρούν την ταχύτητα των γειτονικών αμαξιών. Έχοντας πλέον το διακριτό μοντέλο και του ελεγχτές, είμαστε σε θέση να εκτελέσουμε προσομοιώσεις υιοθετώντας κατάλληλες συναρτήσεις οι οποίες χρησιμοποιούνται για την λειτουργία του φαινομένου εικονικής ώθησης, βλέπε (5.1), (5.2), (5.3), (5.4). Πέρα από τις συναρτήσεις αυτές, παρουσιάζονται όλες εκείνες οι παράμετροι και οι μεταβλητές οι οποίες επιλέχτηκαν προκειμένου να εξαχθούν αποτελέσματα από τις προσομοιώσεις.

Ένα σημαντικό χαρακτηριστικό όμως το οποίο εξαρτάται άμεσα από τις εκάστοτε αρχικές συνθήκες και καθ' επέκταση από την αρχική ενέργεια του συστήματος είναι η κατάλληλη επιλογή της περιόδου δειγματοληψίας T . Στο Κεφάλαιο 5, ερευνάται η μέγιστη επιτρεπτή χρονική περίοδος T ανά αρχική ενέργεια, προκειμένου ο χρήστης να είναι σε θέση να επιλέξει με ασφάλεια το T ανάλογα με τις αρχικές συνθήκες. Τα αποτελέσματα φαίνονται στο γράφημα 5.1 και παρατηρείται ότι όταν η αρχική ενέργεια είναι μεγαλύτερη από 40, το T αναγκάζεται να πάρει τιμές εξαιρετικά κοντά στο μηδέν.

Έχοντας τα δεδομένα του ανωτέρω διαγράμματος είμαστε σε θέση να εκτελούμε προσομοιώσεις διαλέγοντας κατάλληλα και με ασφάλεια το T . Παρόλα αυτά, ένα σταθερό T κοινό για όλα τα αμάξια κατά την διάρκεια της εκάστοτε προσομοίωσης, αποτελεί ένα μη ρεαλιστικό χαρακτηριστικό καθώς τα οχήματα θα πρέπει να δέχονται αλληλεπίδραση από τους ελεγχτές ταυτόχρονα. Επιπρόσθετα, ένα σταθερό T αναγκάζει τον υπολογιστή να καταναλώσει ίσως παραπάνω χρόνο στην προσπάθειά του να ολοκληρώσει την προσομοίωση. Αυτό διότι το κατάλληλο T σύμφωνα με το γράφημα 5.1, επιλέγεται σύμφωνα με την αρχική ενέργεια του συστήματος. Όμως τις περισσότερες φορές υπάρχουν αμάξια που δεν συμβάλουν σε μεγάλο ποσοστό στην τιμή της αρχικής ενέργειας.

Άρα, αναλύοντας αυτά τα οχήματα μεμονωμένα, ίσως μπορούσαν να υιοθετήσουν σε πρώτο χρόνο ένα μικρότερο T .

Για τον λόγο αυτό στο Κεφάλαιο 6, παρουσιάζονται ικανές συνθήκες τέτοιες ώστε το εκάστοτε αμάξι να μπορεί να υπολογίζει αυτόνομα το T , σύμφωνα με την κατάσταση του συστήματος εκείνη την χρονική στιγμή. Πιο συγκεκριμένα, το Λήμμα 6.1 παρέχει τις συνθήκες κάτω από τις οποίες ένα αυθαίρετο όχημα i μπορεί να πραγματοποιήσει τον προαναφερθέντα υπολογισμό. Παράλληλα, το Λήμμα 6.2 εξηγεί ότι ο υπολογισμός του T μπορεί να εφαρμοστεί με την βοήθεια του Λήμματος 6.1, έτσι ώστε σε κάθε δειγματοληψία να επιλέγεται T ανάλογα με την κατάσταση του συστήματος. Για παράδειγμα σε μια αυθαίρετη δειγματοληψία, αν όλα τα οχήματα έχουν το ίδιο T και ταυτόχρονα ικανοποιούνται οι προϋποθέσεις του Λήμματος 6.1 για όλα τα οχήματα για το συγκεκριμένο T , τότε, μέχρι την επόμενη δειγματοληψία το σύστημα δεν πρόκειται να έχει καμία σύγκρουση.

Παίρνοντας ως δεδομένα τα δύο προηγούμενα Λήμματα, είμαστε σε θέση να σχεδιάσουμε 3 ξεχωριστά σενάρια διαφορετικού υπολογισμού του T , τα οποία παρουσιάζονται στο Κεφάλαιο 7. Το πρώτο σενάριο υποθέτει ότι το T είναι αμετάβλητο καθ' όλη την διάρκεια της προσομοίωσης. Το δεύτερο σενάριο αντιπροσωπεύει την περίπτωση όπου το T αλλάζει μη περιοδικά αλλά είναι ίδιο για όλα τα οχήματα σε κάθε δειγματοληψία σύμφωνα με το Λήμμα 6.2. Τέλος, το σενάριο 3 αναπαριστά την περίπτωση όπου κάθε όχημα υπολογίζει αυτόνομα και ξεχωριστά το δικό του T ανάλογα με την εκάστοτε κατάσταση του συστήματος. Τα αποτελέσματα από τις προσομοιώσεις, υπέδειξαν ότι τα σενάρια 2 και 3 βοήθησαν σημαντικά στην μείωση του χρόνου εκτέλεσης των προσομοιώσεων σε σύγκριση με το σενάριο 1. Πιο συγκεκριμένα στο σενάριο 2 παρουσιάστηκε μικρότερος χρόνος εκτέλεσης των προσομοιώσεων με μικρή διαφορά από το σενάριο 3. Παρότι το αναμενόμενο θα ήταν το σενάριο 3 να προσδιορίζεται από μικρότερους χρόνους εκτέλεσης, αυτό δεν συνέβη καθώς το κάθε αμάξι όφειλε να υπολογίζει τις προσωρινές αποστάσεις των υπολοίπων αυτοκινήτων τα οποία δεν διακριτοποιούνται ταυτόχρονα όπως στο σενάριο 2, προσθέτοντας σημαντικό υπολογιστικό κόστος στον επεξεραστή. Η παραπάνω παρατήρηση φαίνεται περισσότερο αξιοσημείωτη αναλογιζόμενοι ότι το μέσο T στο σενάριο 3 παρουσίασε σημαντικά χαμηλότερη τιμή από το μέσο T στο σενάριο 2, όταν όλοι οι υπόλοιποι παράμετροι ήταν κοινοί, βλέπε Πίνακες 8.1, 8.2.

Επιπρόσθετα, κατά την διάρκεια των προηγούμενων προσομοιώσεων εξετάστηκε η περίπτωση κατά την οποία τα οχήματα λαμβάνουν υπόψιν τις ταχύτητες των γειτονικών αυτοκινήτων. Σε αυτήν την περίπτωση παρατηρήθηκε η ικανότητα διατήρησης των επιταχύνσεων σε μικρότερα επίπεδα κάτι το οποίο κρίνεται σημαντικό, καθώς βελτιώνει ενθαρρυντικά την εμπειρία οδήγησης. Πιο συγκεκριμένα στο γράφημα 7.12α παρουσιάζεται ένα παράδειγμα όπου συγκρίνονται οι δύο περιπτώσεις. Η πρώτη αφορά την περίπτωση όταν τα οχήματα προσμετρώνε τις ταχύτητες των γειτονικών αμαξιών και η δεύτερη όταν τις αγνοούν. Φαίνεται ξεκάθαρα ότι το σύστημα τείνει να απορροφάει την μεγάλη αύξηση των τιμών της επιτάχυνσης όταν οι ταχύτητες των γειτονικών αυτοκινήτων λαμβάνονται υπόψη. Επιπλέον παρατηρήθηκε γρηγορότερη σύγκλιση των οχημάτων στις επιθυμητές τιμές.

Στα προηγούμενα Κεφάλαια φάνηκε, ότι όταν η αρχική ενέργεια λαμβάνει τιμές μεγαλύτερες από το 80, τα οχήματα τείνουν να αποκτούν κάτω από οποιοδήποτε συνθήκες, σημαντικά μεγάλες επιταχύνσεις που συχνά κρίνονται μη ρεαλιστικές. Για τον λόγο αυτό θεωρήθηκε απαραίτητη η υιοθέτηση ορίων που ανταποκρίνονται σε φυσικά όρια όσον αφορά τις επιταχύνσεις. Στο Κεφάλαιο 8 παρουσιάζονται προσομοιώσεις χρησιμοποιώντας άνω και κάτω φράγματα στις επιταχύνσεις και επιβραδύνσεις αντίστοιχα, τα οποία δεν αφήνουν τα οχήματα να κινούνται με απότομες μη ρεαλιστικές μεταβολές ταχύτητας. Σε πρώτη φάση τα αποτελέσματα φαίνονται άκρως ενθαρρυντικά καθώς η συρρίκνωση των επιταχύνσεων στα δοσμένα πλαίσια δεν οδήγησε σε καμία σύγκρουση.

Παρόλα αυτά, όπως είναι αναμενόμενο, όταν η διαφορά ανάμεσα στην δοθείσα από τους ελεγκτές επιτάχυνση και στο υιοθετημένο όριο επιτάχυνσης είναι αρκετά μεγάλη, υπάρχει πιθανότητα συγκρούσεων.

Ορμώμενοι από την προαναφερθείσα περίπτωση, προκειμένου να αποφευχθεί η περίπτωση κατά την οποία οι αρχικές συνθήκες οδηγούν σε άμεση σύγκρουση κατά τις πρώτες στιγμές των προσομοιώσεων (δηλαδή η αρχική ενέργεια κατέχει πολύ μεγάλες τιμές οδηγώντας τους ελεγκτές να παράγουν μη ρεαλιστικές επιταχύνσεις), δημιουργήθηκε ένας αλγόριθμος ο οποίος μας τροφοδοτεί με ρεαλιστικές αρχικές συνθήκες. Πιο συγκεκριμένα, ο εν λόγω αλγόριθμος αποτρέπει την δημιουργία αρχικών συνθηκών κάτω από τις οποίες 1) τα οχήματα έχουν επικίνδυνη πορεία προς το όριο του δρόμου και 2) τα οχήματα τείνουν να συγκρουστούν μεταξύ τους. Ο αλγόριθμος παρουσιάζεται στο Κεφάλαιο 9.

Τέλος, στο [παράρτημα](#), εκτός από την διαδικασία μετατροπής του συνεχούς μοντέλου σε διακριτό, παρουσιάζεται ένα παράδειγμα το οποίο υποδεικνύει τον τρόπο με τον οποίο λειτουργεί το σενάριο 3, το οποίο αναφέρθηκε νωρίτερα.

Abstract

Vehicle automation has made tremendous advances in recent decades and the path to full automation of vehicles seems inevitable. In this dissertation, cruise control systems for autonomous vehicles on lane-free roads will be studied. The model depicting the motion of such vehicles is described by a large number of nonlinear continuous-time differential equations. The application of such models in real systems requires conversion of the continuous-time system into a discrete-time system with the additional feature that the values of the input signals are sampled. The modeling and conversion of the continuous-time system into a sampled-data control system will be studied first in this diploma thesis. Next, appropriate design of controllers with discrete sampling will be considered. The design of controllers for discrete sampling systems is performed using the simulation approach in which we first design a controller in the continuous-time domain and then discretize the controller and apply it digitally. Since the digital controller approach can only be successful if the sampling period is sufficiently short, the selection of the sampling period that guarantees the stability of the system to the proposed controllers by discrete sampling will be studied through simulations. At the same time, using appropriate proven Lemmas, it will be examined the occasion when the sampling period changes non- periodically. Then a study will be made through simulations of the discrete-time model when there are saturation effects on the input of the system. Finally, an algorithm will be presented that creates realistic initial positions in order to avoid high initial energies which can lead to a collision course.

Contents

1	Introduction	10
2	Preliminaries	13
2.1	Sampled-data systems	13
2.2	Bicycle kinematic model	14
3	Continuous-time model	16
3.1	Problem statement	18
3.2	Controllers	19
3.3	Theorem 1	20
4	Sampled-data model	22
4.1	Exact discrete model	22
4.2	Influence of sampling period	23
4.3	Influence of initial energy	25
5	Numerical investigation of Maximum Allowable Sampling Period (MASP)	28
5.1	Selection of potential functions, parameters and gains	28
5.2	MASP per initial energy	29
6	State-dependent sampling period	30
6.1	Lemma 1	30
6.2	Lemma 2	31
7	Applications of state-dependent sampling-period	32
7.1	Scenario 1	32
7.2	Scenario 2	37
7.3	Scenario 3	40
8	Saturated control input	44
9	Initial positions generator	49
9.1	Examination if each vehicle is moving dangerously towards the boundary of the road.	49
9.2	Examination if each vehicle has in front of it or behind it, another vehicle at a dangerously close distance.	50
9.3	Examination if a vehicle has an adjacent vehicle with opposite orientation.	52
10	Conclusions	55
	Appendix	56

Chapter 1

Introduction

In the past few years, vehicle automation has been introduced into our lives, and it's easily observed that the way to full automation in the following decades seems possible. Firstly, the companies related to vehicle automation dealt with developing systems like the cruise control system, which maintains the vehicle speed to a standard point, assisting the driver. Following that process, the industry techniques mentioned above were improved, so that vehicles could take into account the distance from a front vehicle to maintain the desired speed (Adaptive Cruise Control (ACC) systems). Further to this, recent research has indicated the advantages of Cooperative ACC systems (CACC) where communication technologies have been established lest vehicles are wirelessly connected and can communicate, maintaining shorter inter-vehicle distances and increasing the capacity of the roads. Consequently, the safety has increased while the accidents and the congestion have reduced, improving the traffic flow on highways, ([4],[12],[18]). Both ACC and CACC systems have been extensively studied in the literature (see for instance [12], [8], [10], [16], [20], [22]).

A significant segment of research deals with vehicles that abide to a lane discipline, increasing safety since manual driving is simplified. Notably, the control strategies mentioned earlier (ACC, CACC) are developed based on information from the vehicle directly in front or behind (only for CACC). In addition, it should be underlined that drivers in lane-based traffic models must be accustomed to lane-changing, which is a riskier manoeuvre since the driver has to estimate other vehicles' speeds to find a gap to complete the lane-changing process. Modelling lane changes and two-dimensional movement on lane-based roads is a complicated problem, and various approaches have been considered, since lane changing is known to affect both capacity and safety (see for instance [3],[15],[23]).

However, new research material has emerged in the last years that concerns vehicles which operate on lane-free roads ([14],[1]). Thus, vehicles don't have to obey to any restrictions related to lane changing as in conventional traffic since the road has no lanes and the vehicles connect wirelessly or via sensors. Thus, connected and automated vehicles are suitable and more efficient in a lane-free environment where they can use their capabilities to their full extent. These approaches are based on linear systems theory and traditional longitudinal car-following models, which, however, do not guarantee: (i) collision avoidance with other vehicles or with the boundary of the road, (ii) positivity of speeds, and (iii) speeds within road speed limits. Apart from the lane-free concept, the capacity and the traffic flow could increase via the "nudging" effect. The term "nudging" describes a virtual force that can exert a vehicle to another one, and it's proved that nudging effect could increase the flow on a ring road and have a strong stabilizing effect; see [11] and references therein.

Considering identical autonomous vehicles described by the bicycle kinematic model and using a family of decentralized controllers (see [9]) for the safe operation of the vehicles on lane-free roads, we obtain the following features:

1. The proposed controllers are fully decentralized, and each vehicle only has access to the distance from the boundaries of the road and the relative distances and relative speeds from adjacent cars.
2. The vehicles may also use relative speeds from adjacent vehicles
3. The vehicles do not collide with each other or with the boundary of the road;
4. The speeds of all vehicles are always positive and remain below a given speed limit;
5. All vehicle speeds converge to a given longitudinal speed set-point;
6. The accelerations, lateral speeds, and orientations of all vehicles tend to zero.

The bicycle kinematic model is selected because it can capture the non-holonomic constraints of the actual vehicle (see [16]).

When calculating the distance between two vehicles, elliptic metrics are adopted, which effectively approach the actual dimensions of the vehicles. Consequently, in comparison to the Euclidean distance, the capacity on the road can increase even more since the elliptic distance assists in the determination of the minimum safety distance between vehicles to avoid collisions. It is vital we select an appropriate “elliptical” distance as we can regulate both the safety distance and the number of vehicles that can be placed laterally on the road. Finally, potential functions have been employed to avoid collisions between vehicles and the road boundary. Potential functions have also been used in certain lane-changing and lane-keeping problems in traffic control (, [7],[21]).

Simulation-wise, several issues emerge when trying to simulate a continuous-time system of ordinary differential equations. Typical methods to approximate the solution of such systems follow from numerical analysis and are known as Runge-Kutta methods (see [2]). While for numerical simulations, the solution of a control system can be approximated by the aforementioned methods, in practical applications, a control system acts as a hybrid system, where the model is in continuous-time and the control inputs are discretized and applied in periodic instances (sampled-data systems).

In this diploma thesis, firstly, we discretized the continuous-time bicycle model, which is described in Chapter 3, using sampled-data inputs. This process is necessary since it allows us to develop a code in a programming environment to run numerical simulations using piecewise constant inputs. Following that, we investigate the Maximum Allowable Sampling Period (MASP) under various conditions of the system. Sampling period is specified as the time between two sampling instants and the higher it is, the less time each simulation take. It is noteworthy, that when we discuss about MASP, we assume that the sampling period is constant throughout the simulation. The usage of a constant sampling period for all the vehicles during a simulation seems to be a bit unrealistic and cost prohibitive in terms of execution time. For these reasons, considering appropriate proved conditions, we managed to find efficient ways to calculate the sampling period non-periodically during the simulations in order to reduce the CPU effort and consequently, the execution time. Particularly, we adopt two new methods in which the sampling period can take higher values according to the positions of the other vehicles and the distance from the road boundary. The first method indicates that the

sampling period changes non-periodically, but it is the same for all the vehicles per sampling. This approach is of substantial importance since it speeds-up the execution time considerably. The second method allows the system to asynchronously calculate the sampling period of each vehicle during the episode. In simple words, each vehicle can calculate its own sampling period autonomously. The importance of the second method stems from the fact that in practice, the internal clocks of vehicles are not synchronized and each vehicle updates its control input independently from other vehicles. Execution time-wise the former methods has shown many advantages and evidently they allow the system to increase the average sampling period. Simultaneously, we studied certain properties when the controller measures the speeds of the adjacent vehicles. It was observed that from many perspectives, there exist substantial advantages when the speed of the neighbouring vehicles is measured. Finally, an algorithm that generates initial positions with specific features is presented, which is used since we want to take realistic initial positions when we run simulations. At every stage, simulations and results are displayed.

Notation

Throughout this thesis, we adopt the following notation.

- * $\mathbb{R}_+ := [0, +\infty)$ denotes the set of non-negative real numbers.
- * By $|x|$ we denote both the Euclidean norm of a vector $x \in \mathfrak{R}^n$ and the absolute value of a scalar $x \in \mathfrak{R}$. By x' we denote the transpose of a vector $x \in \mathfrak{R}^n$. By $|x|_\infty = \max\{|x_i|, i = 1, \dots, n\}$ we denote the infinity norm of a vector $x = (x_1, x_2, \dots, x_n)' \in \mathfrak{R}^n$.
- * Let $A \subseteq \mathfrak{R}^n$ be an open set. By $C^0(A, \Omega)$, we denote the class of continuous functions on $A \subseteq \mathfrak{R}^n$, which take values in $\Omega \subseteq \mathfrak{R}^m$. By $C^k(A; \Omega)$, where $k \geq 1$ is an integer, we denote the class of functions on $A \subseteq \mathfrak{R}^n$ with continuous derivatives of order k , which take values in $\Omega \subseteq \mathfrak{R}^m$. When $\Omega = \mathfrak{R}$ then we write $C^0(A)$ or $C^k(A)$. For a function $V \in C^1(A; \mathfrak{R})$, the gradient of V at $x \in A \subseteq \mathfrak{R}^n$, denoted by $\nabla V(x)$, is the row vector $\left[\frac{\partial V}{\partial x_1}(x) \cdots \frac{\partial V}{\partial x_n}(x) \right]$. By $\nabla^2 V(x)$ we denote the Hessian matrix at $x \in A \subseteq \mathfrak{R}^n$ of a function $V \in C^2(A; \mathfrak{R})$.

Chapter 2

Preliminaries

2.1 Sampled-data systems

Most models of real-life control processes are described by differential equations in continuous-time $\dot{x} = f(x, v)$. However, due to digitalization, the control input cannot be applied in continuous-time and thus, a sampled version of the input is applied through a conversion known as Zero-Order-Hold (see [5]).

The idea of sampling consists of defining an exact discrete-time system such that the values of this discrete time system and the continuous time system coincide at the sampling times, *e.i.* $t_0 < t_1 < t_2 < \dots < t_N$.

Hereinafter, we use the symbol v for the control function in order to emphasize the difference between the continuous time control function $v(\bullet)$ and the discrete time control sequence $u(\bullet)$. To illustrate the idea of sampled-data systems we consider the following simple continuous time control system

$$\dot{x}(t) = v(t) \tag{2.1}$$

It is obvious that $v(t)$ constitutes a continuous input function. Let us transform the continuous relation above into an exact discrete-time model. By letting $x_0 = x(0)$ denote the initial condition, it is easily verified that the solutions of the system are given by

$$x(t) = x_0 + \int_0^t v(\tau) d\tau \tag{2.2}$$

By considering the interval $[0, T]$, we obtain:

$$x(T) = x_0 + \int_0^T u(\tau) d\tau \tag{2.3}$$

If we restrict ourselves to constant control functions $u(t) \equiv u \in R$ on the time interval $[0, T]$ (for ease of notation we use the same symbol u for the function and for its constant value), we get

$$x(T) = x_0 + Tu \tag{2.4}$$

This is the sampled-data model of the continuous time system, where the input remains constant between time intervals $[kT, (k+1)T)$, $k = 0, 1, 2, \dots$

In the next pages we will see that we use sampled data feedback for the actual implementation of our controllers. The whole integration of our continuous time model is presented in Appendix while the results are shown in the Chapter 4.

2.2 Bicycle kinematic model

The bicycle kinematic model has long been used as a suitable control-oriented model for representing vehicles because of its simplicity and adherence to the nonholonomic constraints of a car. In the next sentences we describe how it has been developed. The bicycle model we deal with is called as “the front wheel steering model” since the front wheel orientation can be controlled relative to the heading of the vehicle. We assume that the vehicle operates on a 2D plane denoted by the inertial frame FI . The front wheel represents the front right and left wheels of the car and the rear wheel represents the rear right and left wheels of the car.

To analyze the kinematics of the bicycle model, we have to select a reference point X, Y on the vehicle which can be placed at the center of the rear axle (see Figure 2.1), the center of the front axle, or at the center of gravity (cg). The selection of the reference point changes the kinematic equations that result, which in turn change the controller designs we use. In this diploma thesis we assume that the reference point is placed at the center of the rear axle. Following we analyse this assumption.

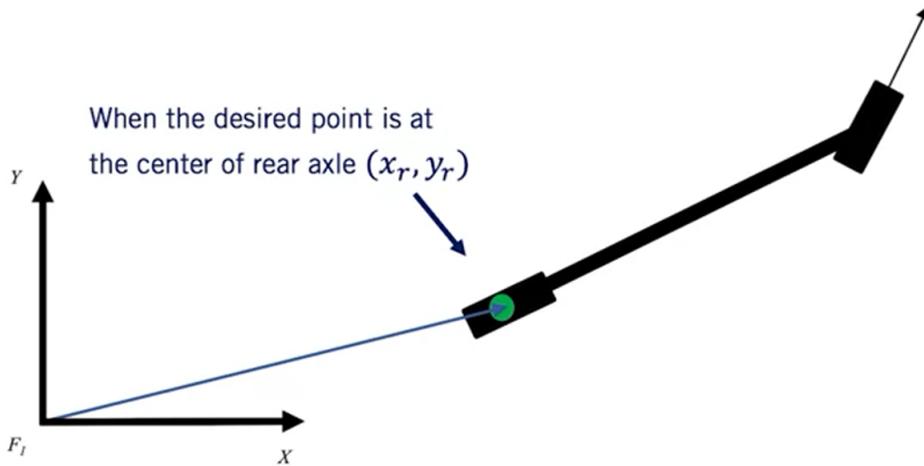


Figure 2.1: The reference point when considering the rear axle reference point model.

Rear axle reference point model

We denote the location of the rear axle reference point as X_r, Y_r and the heading of the bicycle as θ . We use σ for the length of the bicycle, measured between the two-wheel axes. The former features constitute our main model states. In addition, we denote the steering angle δ which is measured relative to the forward direction of the bicycle. Finally the velocity is denoted v and points in the same direction as each wheel, see Figure 2.2. This is an assumption referred to as the no slip condition, which requires that our wheel cannot move laterally or slip longitudinally either.

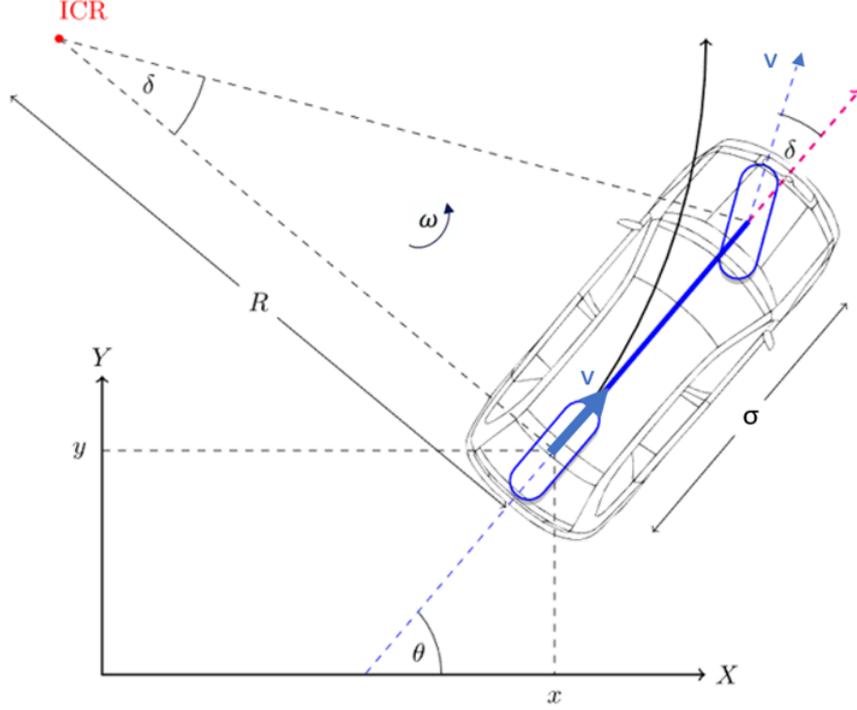


Figure 2.2: Analysis of the bicycle kinematic model.

Because of the no slip condition, we have that ω (e.g. the rotation rate of the bicycle $\dot{\theta}$), is equal to the velocity over the instantaneous center of rotation radius R . More specifically,

$$\dot{\theta} = \omega = \frac{v}{R} \quad (2.5)$$

From the similar triangles formed by L and R and v and δ we get,

$$\tan(\delta) = \frac{\sigma}{R} \quad (2.6)$$

Finally considering (2.5) and (2.6) we get that,

$$\dot{\theta} = \omega = \frac{v}{R} = \frac{v}{\frac{\sigma}{\tan(\delta)}} = \frac{v \tan(\delta)}{\sigma} \quad (2.7)$$

In simple words,

$$\dot{\theta} = v \tan(\delta) \sigma^{-1} \quad (2.8)$$

We can now form the complete kinematic bicycle model for the rear axle reference point. Based on this model configuration, the velocity components of the reference point in the X and Y direction are equal to the forward velocity $v \cos(\theta)$ and $v \sin(\theta)$, respectively. These two relations are combined with the equation for rotation rate to form the rear axle bicycle model where,

$$\begin{aligned} \dot{x} &= v \cos(\theta) \\ \dot{y} &= v \sin(\theta) \\ \dot{\theta}_i &= \sigma^{-1} v \tan(\delta) \end{aligned} \quad (2.9)$$

Here, the speed can be considered as a continuous time signal. It is also possible to consider a differential equation for the evolution of speed $\dot{v} = F$ where F is the acceleration.

Chapter 3

Continuous-time model

Consider n vehicles on a lane-free road of width $2a > 0$, where the bicycle kinematic model describes the movement of each vehicle

$$\begin{aligned} \dot{x}_i &= v_i \cos(\theta_i) \\ \dot{y}_i &= v_i \sin(\theta_i) \\ \dot{\theta}_i &= \sigma^{-1} v_i \tan(\delta_i) \\ \dot{v}_i &= F_i \end{aligned} \tag{3.1}$$

for $i = 1, \dots, n$, where $\sigma > 0$ is the length of each vehicle. Where,

1. $(x_i, y_i) \in \mathfrak{R} \times (-a, a)$ is the reference point of the i_{th} vehicle in an inertial frame with Cartesian coordinates (X, Y) , with $i \in \{1, \dots, n\}$ and is placed at the midpoint of the rear axle of the vehicle.
2. with $x_i \in \mathfrak{R}$ being the longitudinal position and $y_i \in (-a, a)$ being the lateral position of the vehicle.
3. v_i is the speed of the i_{th} vehicle at the point (x_i, y_i) .
4. $\theta_i \in (-\frac{\pi}{2}, \frac{\pi}{2})$ is the angular orientation of the i_{th} vehicle.
5. δ_i is the steering angle of the front wheels relative to the orientation θ_i of the i_{th} vehicle.
6. F_i is the acceleration of the i_{th} vehicle.

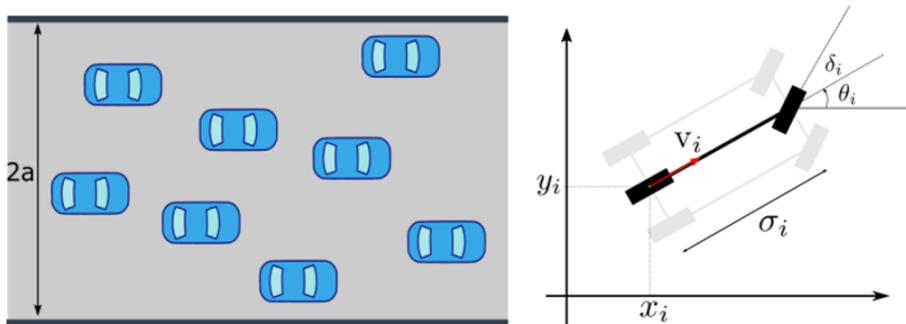


Figure 3.1: Lane-free road of width $2a > a$ (left). Each vehicle is modelled by the bicycle kinematic model (right).

To simplify the subsequent analysis, we use the preliminary feedback:

$$u_i = \sigma^{-1} v_i \tan(\delta_i), \quad i = 1, \dots, n \quad (3.2)$$

Thus, the model (3.1) can be written in the form:

$$\begin{aligned} \dot{x}_i &= v_i \cos(\theta_i) \\ \dot{y}_i &= v_i \sin(\theta_i) \\ \dot{\theta}_i &= u_i \\ \dot{v}_i &= F_i \end{aligned} \quad (3.3)$$

Let $v^* \in (0, v_{\max})$ be given (i.e. the speed set-point) and define the set:

$$S = \mathfrak{R}^n \times (-a, a)^n \times (-\varphi, \varphi)^n \times (0, v_{\max})^n \quad (3.4)$$

where $\varphi \in (0, \frac{\pi}{2})$ is an angle that satisfies,

$$\cos(\varphi) > \frac{v^*}{v_{\max}} \quad (3.5)$$

The set S describes all possible states of the system of n vehicles. More specifically, each vehicle should stay within the road, i.e., $(x_i, y_i) \in \mathbb{R} \times (-\alpha, \alpha)$ for $i = 1, \dots, n$, the vehicles should not be able to turn perpendicular to the road, i.e., $\theta_i \in (-\frac{\pi}{2}, \frac{\pi}{2})$, for $i = 1, \dots, n$ and the speeds of all vehicles should always be positive, i.e., no vehicle moves backwards; and respect the road speed limits. The constant φ is a safety constraint, to restrict the movement of a vehicle, in terms of orientation.

In what follows, we assume that the distance between vehicles is defined by

$$d_{i,j} := \sqrt{(x_i - x_j)^2 + p(y_i - y_j)^2} \quad \text{for } i, j = 1, \dots, n \quad (3.6)$$

where $p > 0$ is a weighting factor. Note that for $p = 1$ we obtain the standard Euclidean distance, while for larger values of $p > 1$, we have an ‘‘elliptical’’ metric which will allow to approximate more accurately the dimensions of a vehicle.

Let the following notation:

$$w = (x_1, \dots, x_n, y_1, \dots, y_n, \theta_1, \dots, \theta_n, v_1, \dots, v_n)' \in \mathfrak{R}^{4n} \quad (3.7)$$

Due to the various constraints that were explained above, the state space of the model is

$$\Omega := \{w \in S : d_{i,j} > L \quad i, j = 1, \dots, n, j \neq i\} \quad (3.8)$$

where L is the minimum acceptable inter-vehicle distance.

As mentioned above, a virtual force called ‘‘nudging’’ is exerted from vehicles to other vehicles to avoid collisions. We consider the potential functions $V : (L, +\infty) \rightarrow \mathbb{R}_+$ and $U : (-a, a) \rightarrow \mathbb{R}_+$ that are both C^2 functions and satisfy the following properties:

$$\lim_{x \rightarrow L^+} (V(d)) = +\infty \quad (3.9)$$

$$V(d) = 0, \text{ for all } d \geq \lambda \quad (3.10)$$

$$\lim_{x \rightarrow (-a)^+} (U(y)) = +\infty, \lim_{x \rightarrow (a)^-} (U(y)) = +\infty \quad (3.11)$$

$$U(0) = 0 \quad (3.12)$$

where λ is a constant that indicates the distance when two vehicles are not influenced by nudging. The potential function V is designed to exert a repulsive force when two vehicles have distance $d \in (L, \lambda)$, and this repulsive force increases as the distance between vehicle decreases. The potential function U deals with the repulsive force when a vehicle approaches the boundary of the road. Also, a function κ will be used, to assist the derived controllers to take into account the speed from the adjacent vehicles. The function has the following property:

$$\kappa(d) = 0, \text{ for all } d \geq \lambda \quad (3.13)$$

which implies that when the distance between 2 vehicles is higher than λ , the function $\kappa(d)$ does not exert any influence on the vehicles.

3.1 Problem statement

For n vehicles operating on a lane free road of width 2α , a Lyapunov function methodology is applied to design decentralized feedback for u_i and F_i which ensure the following properties:

- P1 For $w(0) \in \Omega$, then $w(t) \in \Omega$ for all $t \geq 0$. This implies that throughout a simulation all the values $x_i, y_i, v_i, \theta_i \forall i$ are a part of the state space of the model. Consequently, there are not any collisions among the vehicles or with the boundary of the road and simultaneously all the speeds remain positive and below the given speed limit. Finally, the orientation of each vehicle is bounded by the given value $\varphi \in (0, \frac{\pi}{2})$.
- P2 The orientation of each vehicle will converge to zero and the speed of each vehicle will converge to the given speed set point.
- P3 The accelerations, angular speeds and lateral speed of all vehicles tend to zero.

More specifically,

1. $d_{i,j}(t) > L$ for $t \geq 0$ $i, j = 1, \dots, n, j \neq i$
2. $y_i(t) \in (-a, a)$ for $t \geq 0$
3. $v_i(t) \in (0, v_{\max})$ for all $t \geq 0$
4. $\theta_i(t) \in (-\varphi, \varphi)$ for $t \geq 0$,
5. $\lim_{t \rightarrow \infty} (\theta_i(t)) = 0, i = 1, \dots, n$
6. $\lim_{t \rightarrow \infty} (v_i(t)) = v^*, i = 1, \dots, n$
7. $\lim_{t \rightarrow \infty} (F_i(t)) = 0, i = 1, \dots, n$
8. $\lim_{t \rightarrow \infty} (\dot{y}_i(t)) = 0, i = 1, \dots, n$

We define the Control Lyapunov Function where the feedback laws are appropriately selected to render the derivative of a Lyapunov function negative semi-definite, for all $w \in \Omega$,

$$H(w) := \frac{1}{2} \sum_{i=1}^n (v_i \cos(\theta_i) - v^*)^2 + \frac{1}{2} \sum_{i=1}^n v_i^2 \sin^2(\theta_i) + \sum_{i=1}^n U(y_i) + \frac{1}{2} \sum_{i=1}^n \sum_{i \neq j} V(d_{i,j}) + A \sum_{i=1}^n \left(\frac{1}{\cos(\theta_i) - \cos(\varphi)} - \frac{1}{1 - \cos \varphi} \right) \quad (3.15)$$

where $A > 0$ is a parameter of the controller and the Lyapunov function and $v^* \in (0, v_{\max})$ is a given longitudinal speed set-point.

The function H indicates the total energy of the system of n vehicles and allows to be exploited certain properties of the state space Ω . The first two terms ($\frac{1}{2}(\sum_{i=1}^n (v_i \cos(\theta_i) - v^*)^2 + \frac{1}{2} \sum_{i=1}^n v_i^2 \sin^2(\theta_i))$) project the kinetic energy of the system penalizing the deviation of the longitudinal and lateral speed from their desired values v^* and zero, respectively. The terms ($\sum_{i=1}^n U(y_i) + \frac{1}{2} \sum_{i=1}^n \sum_{i \neq j} V(d_{i,j})$) constitute the potential energy of the system which is produced from the functions V and U . Finally, the term $\left(A \sum_{i=1}^n \left(\frac{1}{\cos(\theta_i) - \cos(\varphi)} - \frac{1}{1 - \cos(\varphi)} \right) \right)$ is a penalty term that blows up when $\theta_i \rightarrow \pm\varphi$.

3.2 Controllers

The feedback laws for each vehicle $i = 1, \dots, n$ can be designed using the H function in terms of their own speed and orientation and the gradient of the potential functions V and U mentioned earlier.

$$u_i = \left(G_i(w) - U'(y_i) - p \sum_{j \neq i} V'(d_{i,j}) \frac{(y_i - y_j)}{d_{i,j}} - \sin(\theta_i) F_i \right) \left(v^* + \frac{A}{v_i (\cos(\theta_i) - \cos(\varphi))^2} \right)^{-1} \quad (3.16)$$

$$F_i = -\frac{1}{\cos(\theta_i)} (k_i(w)(v_i \cos(\theta_i) - v^*) + \Lambda_i(w)) \quad (3.17)$$

where,

$$G_i(w) = -\mu_1 v_i \sin(\theta_i) + \sum_{j \neq i} \kappa(d_{i,j}) (v_j \sin(\theta_j) - v_i \sin(\theta_i)) \quad (3.18)$$

and

$$\Lambda_i(w) = \sum_{j \neq i} V'(d_{i,j}) \frac{(x_i - x_j)}{d_{i,j}} - \sum_{j \neq i} \kappa(d_{i,j}) (v_i \cos(\theta_j) - v_i \cos(\theta_i)) \quad (3.19)$$

and

$$k_i(w) = \mu_2 + \frac{\Lambda_i(w)}{v^*} + \frac{v_{\max} \cos(\theta_i)}{v^* (v_{\max} \cos(\theta_i) - v^*)} f(-\Lambda_i(w)) \quad (3.20)$$

where,

$f \in C^1(\mathbb{R})$ is any function that satisfies :

$$\max(0, x) \leq f(x) \text{ for all } x \in \mathbb{R} \quad (3.21)$$

The term $k_i(w)$ in the acceleration $F_i(t)$ given by (3.17), is a state-dependent controller gain which guarantees that the speed of each vehicle will remain positive and less than the speed limit. The first term in $\Lambda_i(w)$ given by (3.19), is the summation of the repulsive forces from vehicles that are within the nudging area of vehicle i and the second term constitutes the summation of relative longitudinal speeds from vehicles that are in close proximity to vehicle i . If V in (3.9), (3.10) is decreasing, then, the first term of (3.19) is negative if vehicle j is behind vehicle i , i.e., $(x_i - x_j) > 0$. Indeed, in this case, we have that $-V'(d_{i,j})(x_i - x_j)d_{i,j}^{-1} > 0$, and this term represents the effect of nudging, since vehicles that are close and behind vehicle i will also exert a “pushing” force towards it that will increase its acceleration.

Remark 1:

1. Properties (3.9) and (3.13) guarantee that the feedback laws (3.16) and (3.17) depend only on information from adjacent vehicles, namely from vehicles that are located at a distance less than $\lambda > 0$. Notice also that the control inputs (3.16), (3.17), (3.20) only require the distance and the relative speeds from neighbouring vehicles and not additional information, such as relative orientations $(\theta_i - \theta_j)$ $i, j = 1, \dots, n, i \neq j$.
2. Any function $f \in C^1(\mathfrak{R})$ that satisfies (3.21) can be used in (3.20). For instance, the function $f(x) = \frac{\varepsilon}{2} + \frac{1}{2\varepsilon}x^2$ for every $\varepsilon > 0$ satisfies (3.21), since $\max(x, 0) \leq |x| \leq \frac{\varepsilon}{2} + \frac{1}{2\varepsilon}x^2$ for all $x \in \mathfrak{R}$. Another function that satisfies (3.21) is the function

$$f(x) = \frac{1}{2\varepsilon} \begin{cases} 0 & , \text{if } x \leq -\varepsilon \\ (x + \varepsilon)^2 & , \text{if } -\varepsilon < x < 0 \\ \varepsilon^2 + 2\varepsilon x & , \text{if } x \geq 0 \end{cases} \quad (3.22)$$

for every $\varepsilon > 0$.

This generic design for all the function f will allow to regulate the longitudinal acceleration as desired. For instance, in the first example above, f exhibits quadratic growth while in (3.22) only linear growth for $x \geq 0$.

3.3 Theorem 1

Suppose that there exist constants $a > 0, \lambda > L > 0, p \geq 1$ and C^2 functions $V : (L, +\infty) \rightarrow \mathbb{R}_+, U : (-\alpha, \alpha) \rightarrow \mathbb{R}_+$, that satisfy (3.9), (3.10) and (3.11), (3.12), respectively. Moreover, for given constants $v_{\max} > 0, v^* \in (0, v_{\max}),$ and $\varphi \in (0, \frac{\pi}{2})$ that satisfies (3.5), define function $H : \Omega \rightarrow \mathbb{R}_+$ by means of (3.15) where Ω is given by (3.8). Then, for every $w_0 \in \Omega$ there exists a unique solution $w(t) \in \Omega$ of the initial-value problem (3.3), (3.16), (3.17), (3.20) with initial condition $w(0) = w_0$. The solution $w(t) \in \Omega$ is defined for all $t \geq 0$ and satisfies for $i = 1, \dots, n$

$$\lim_{t \rightarrow \infty} (v_i(t)) = v^*, \quad \lim_{t \rightarrow \infty} (\theta_i(t)) = 0 \quad \lim_{t \rightarrow \infty} (u_i(t)) = 0, \quad \lim_{t \rightarrow \infty} (F_i(t)) = 0 \quad (3.23)$$

In addition, there exists a non-decreasing function $Q_k : \mathbb{R}_+ \rightarrow \mathbb{R}_+ (k = 1, 2)$ such that $|F_i(t)| \leq Q_1(H(w(0))), |u_i(t)| \leq Q_2(H(w(0))),$ for all $t \geq 0, i = 1, \dots, n$ and for every solution $w(t) \in \Omega$ of (3.3), (3.16) and (3.17).

Remark 2:

1. The results of Theorem 1 hold true globally, i.e., for any initial condition $w_0 \in \Omega$.
2. It is important to notice that due to technical constraints, an inequality of the form $|F_i(t)| \leq K$ must be satisfied for all $t \geq 0$, where $K > 0$ is a constant that depends on the technical characteristics of the vehicles and the road. Inequality $|F_i(t)| \leq Q_1(H(w(0)))$ allows us to determine the set of initial conditions $w_0 \in \Omega$ for which the inequality $|F_i(t)| \leq K$ holds: it includes the set of all $w_0 \in \Omega$ with $Q_1(H(w_0))v_{\max} \leq K$.
3. Although we cannot predict the “ultimate” arrangement of the vehicles on the road (and we cannot even show that a final configuration of the vehicles on the road is attained; see remark below), the limits (3.23) and definitions (3.17) allow us to predict that
$$\lim_{t \rightarrow +\infty} \left(\sum_{j \neq i} V'(d_{i,j}(t)) \frac{(x_i(t) - x_j(t))}{d_{i,j}(t)} \right) = \lim_{t \rightarrow +\infty} \left(U'(y_i(t)) + p \sum_{j \neq i} V'(d_{i,j}(t)) \frac{(y_i(t) - y_j(t))}{d_{i,j}(t)} \right) = 0$$
 for $i = 1, \dots, n$. Consequently, the “ultimate” arrangement of the vehicles in the road (if such a thing exists) must satisfy the following equation $\sum_{j \neq i} V'(d_{i,j}) \frac{(x_i - x_j)}{d_{i,j}} = U'(y_i) + p \sum_{j \neq i} V'(d_{i,j}) \frac{(y_i - y_j)}{d_{i,j}} = 0$ for $i = 1, \dots, n$ as well as the constraints $|y_i| < a$ and $d_{i,j} > L$ for $i, j = 1, \dots, n, j \neq i$. Despite the fact that the constrained system of $2n$ equations has infinite solutions, not every arrangement of vehicles satisfies the aforementioned constrained system.
4. The proof of the Theorem 1 is presented in [9].

Chapter 4

Sampled-data model

For the actual implementation of the controllers, we used sampled-data feedback. Initially, we designed the continuous-time controller that satisfies certain closed-loop control specifications and then we transformed it into a digital controller. This technique is called “emulation” and it has been widely used in the literature, see [13]. More specifically, the control inputs F_i and δ_i are digitalized and produce a sequence of control values $F_i(t_k)$, $\delta_i(t_k)$ using the sampled version of the system’s state $w(t_k)$ at each sampling instant $t_{k+1} = t_k + T$, $k = 0, 1, 2, \dots, T > 0$, t_0 being the initial time.

Figure 4.1 shows how we have to consider the values of the input F_i when we integrate the continuous model. As mentioned before, the input F_i is digitalized and therefore maintains its value between two sampling time. As a result, when doing the relative integration, we assume that the input is a constant.

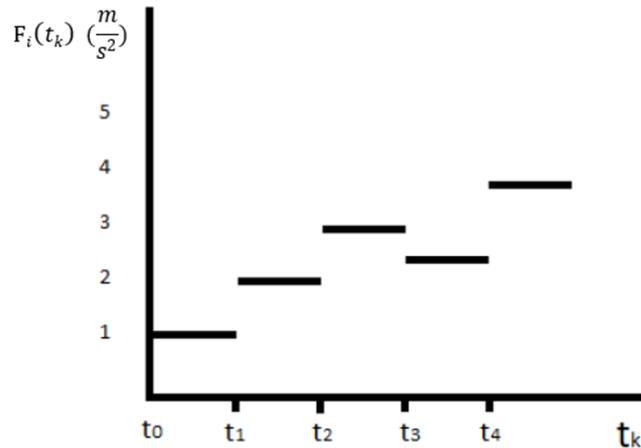


Figure 4.1: The way we consider the values of the input F_i when integrating the continuous model.

4.1 Exact discrete model

For brevity and conciseness, let

$$\omega_i := \sigma^{-1} \tan(\delta_i), \quad \text{for } i = 1, \dots, n \quad (4.1)$$

The exact discrete model of (3.3) under constant inputs F_i and u_i is given by

If $\omega_i = 0$

$$\begin{aligned}x_i(t + s) &= x_i(t) + v_i(t) \cos(\theta_i(t))s + F_i \cos(\theta_i(t))\frac{s^2}{2} \\y_i(t + s) &= y_i(t) + v_i(t) \sin(\theta_i(t))s + F_i \sin(\theta_i(t))\frac{s^2}{2} \\ \theta_i(t + s) &= \theta_i(t) \\ v_i(t + s) &= v_i(t) + sF_i\end{aligned}\tag{4.2}$$

for all $s \in [0, T]$; and if $\omega_i \neq 0$

$$\begin{aligned}x_i(t + s) &= x_i(t) + \frac{1}{\omega_i} \left(\sin \left(\theta_i(t) + \omega_i \left(v_i(t) + \frac{1}{2}sF_i \right) s \right) - \sin(\theta_i(t)) \right) \\y_i(t + s) &= y_i(t) + \frac{1}{\omega_i} \left(\cos(\theta_i(t)) - \cos \left(\theta_i(t) + \omega_i \left(v_i(t) + \frac{1}{2}sF_i \right) s \right) \right) \\ \theta_i(t + s) &= \theta_i(t) + \omega_i \left(v_i(t) + \frac{1}{2}sF_i \right) s \\ v_i(t + s) &= v_i(t) + sF_i\end{aligned}\tag{4.3}$$

for all $s \in [0, T]$.

In Appendix, there exists the whole procedure of integration with which we get the exact discrete model mentioned above.

4.2 Influence of sampling period

The exact discrete model will be used in the numerical simulations of the following Chapters. While the parameters and gains (e.g., A , p , μ_1 , μ_2 , etc.) greatly influence the behaviour of our system (such as convergence rate and the magnitude of the acceleration) a decisive factor is the sampling period T . As depicted in the following pages, the sampling period constitutes the most significant value from the perspective of the simulation time. However, an arbitrarily high value of the sampling period will lead to collisions since the vehicles will not have time to renew their control input according to the feedback laws. This feature illustrated in the following examples. The first example illustrates a case where a crash was avoided thanks to the small sampling period whereas in the second example the car collided with the boundary of the road, due to the high sampling period.

As we can see in Figure 4.2, there exists a small enough sampling period ($T=0.1$), and therefore at the 3rd sampling, the vehicle starts to turn as the feedback laws prevent a crash with the boundary of the road. It should be underlined that the small value of the sampling period plays a significant role here since the vehicle has enough time to renew its values.

In Figure 4.3, the sampling period was too high ($T=0.5$ sec), and consequently, the vehicle had no time to re-evaluate its control input according to the feedback laws. Between the two sampling times, the vehicle went towards the road boundary, and the subsequent sampling did not take place as the vehicle crashed. It is easily realized that although high values of the sampling period lead to faster simulation times, sampling period selection is vital since collisions can occur. In the following pages, we will see different and alternative ways to calculate the sampling period efficiently with the guarantee that there will not be any collisions.

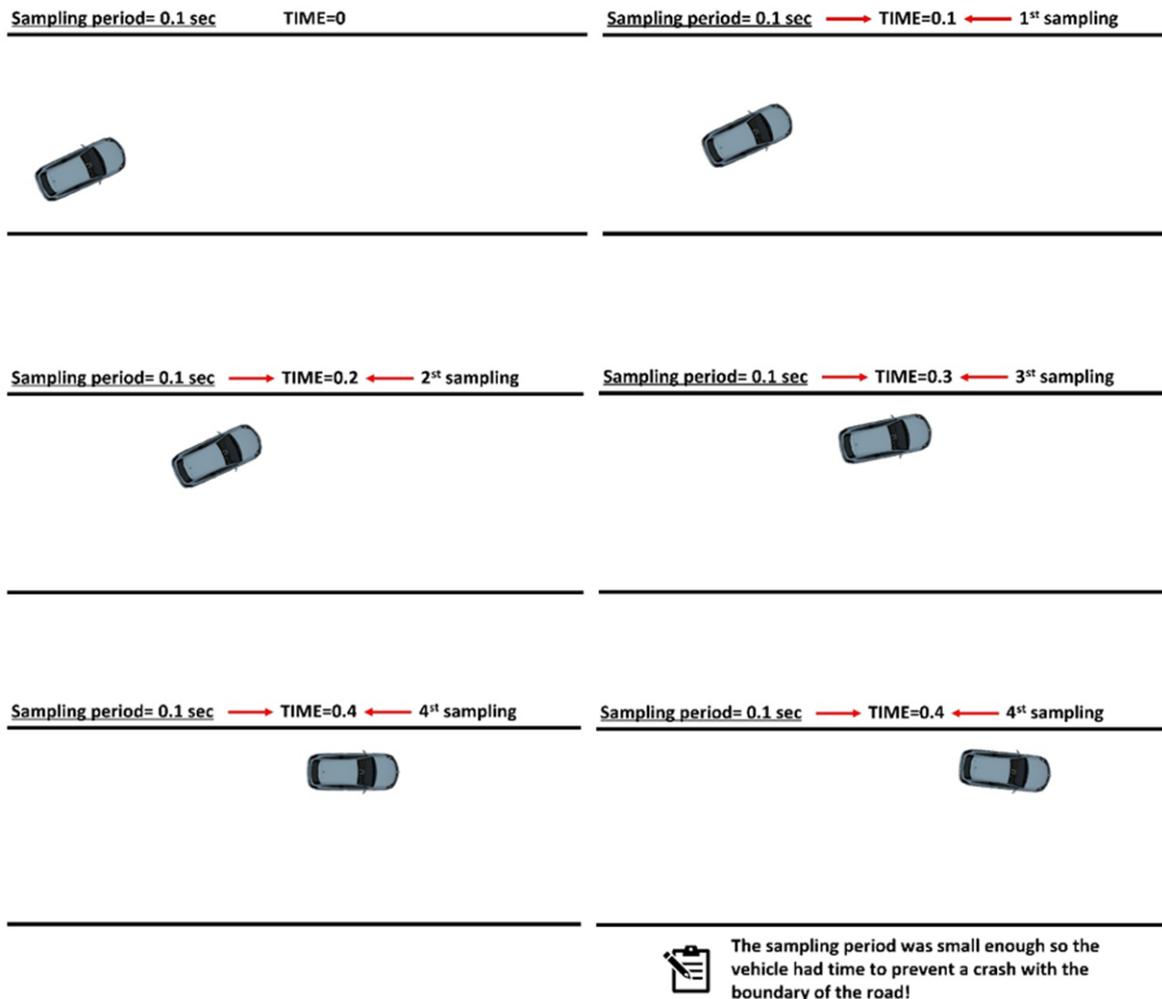


Figure 4.2: An example where a crash is detected thanks to the small sampling period.

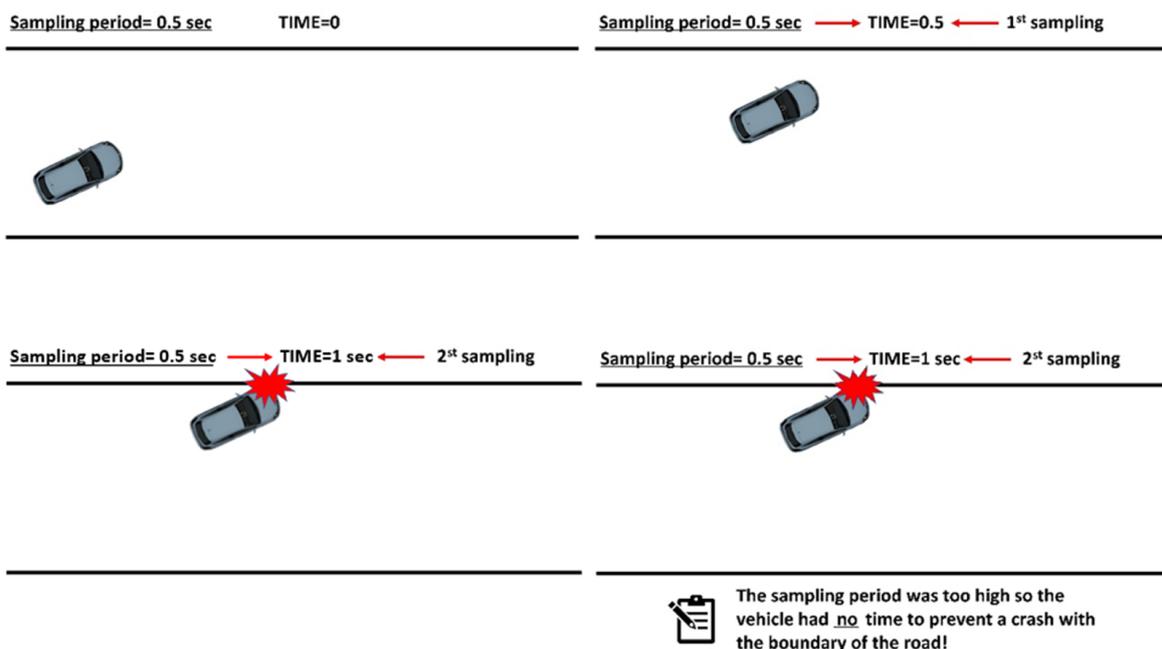


Figure 4.3: An example where a crash took place due to the high sampling period.

4.3 Influence of initial energy

While the sampling period constitutes a decisive factor which can lead to collisions, the initial energy of the system can influence it, accordingly. Initial energy is the energy produced by the Lyapunov function H in (3.15) that obtains the system from the initial positions. High initial energy means that the system is far away from convergence. Conversely, small initial energy indicates that the system may converge faster. Next, we will study two examples in which the sampling periods are the same and the only value that changes is the initial energy (e.g. the initial positions, speed, and orientation).

In the snapshots in Figure 4.4, where the initial energy $H(0) = 18.32$ and the sampling period $T = 0.1$, there is not any collision neither among vehicles nor with the boundary of the road.

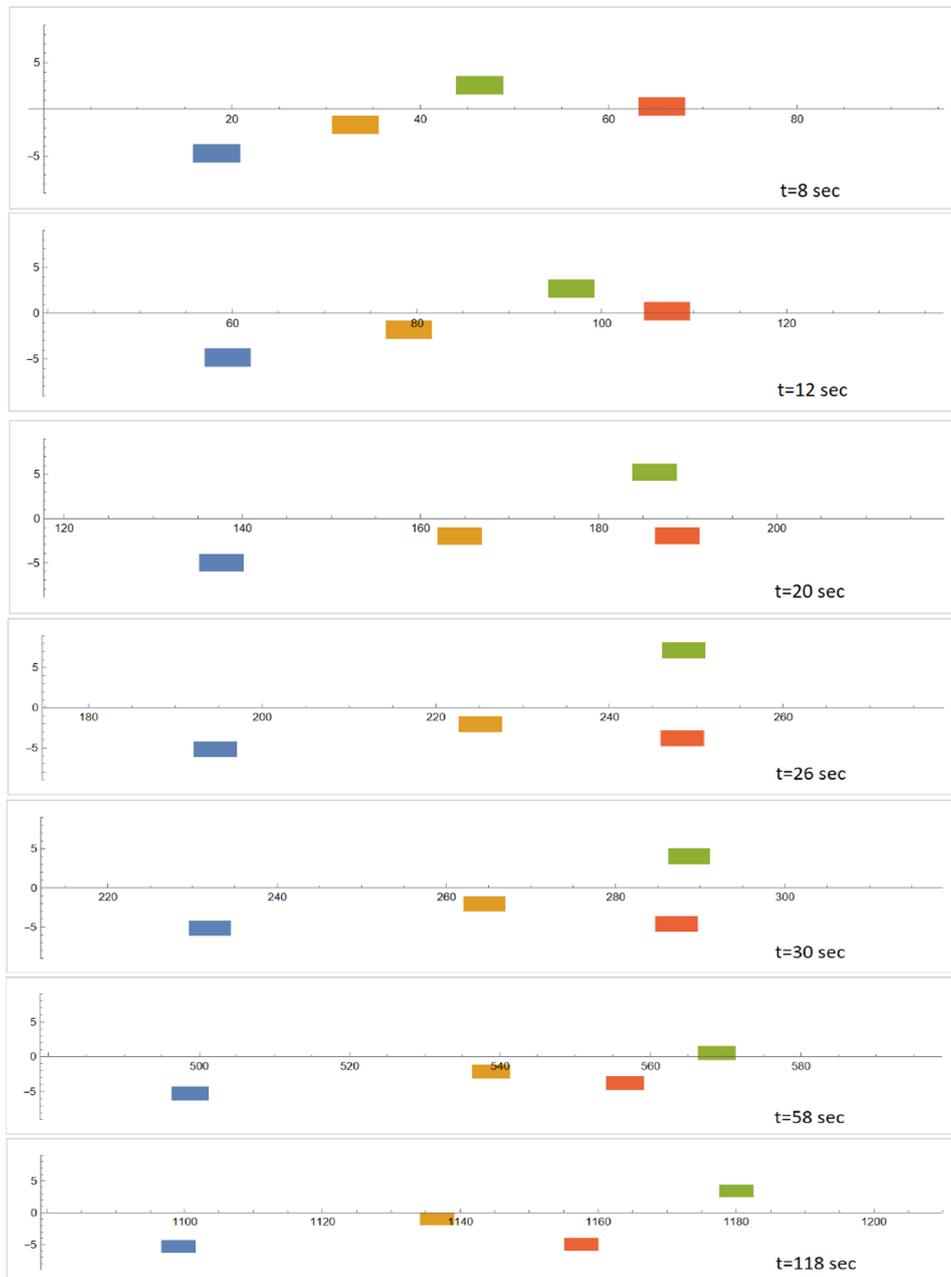


Figure 4.4: A real example with initial energy $H(0)=18.32$ and sampling period $T=0.1$.

On the contrary, when observing the snapshots in Figure 4.5, where the initial energy increased to 39.10, a collision between two vehicles is presented. It is noteworthy that although the sampling period did not change, a crash took place.

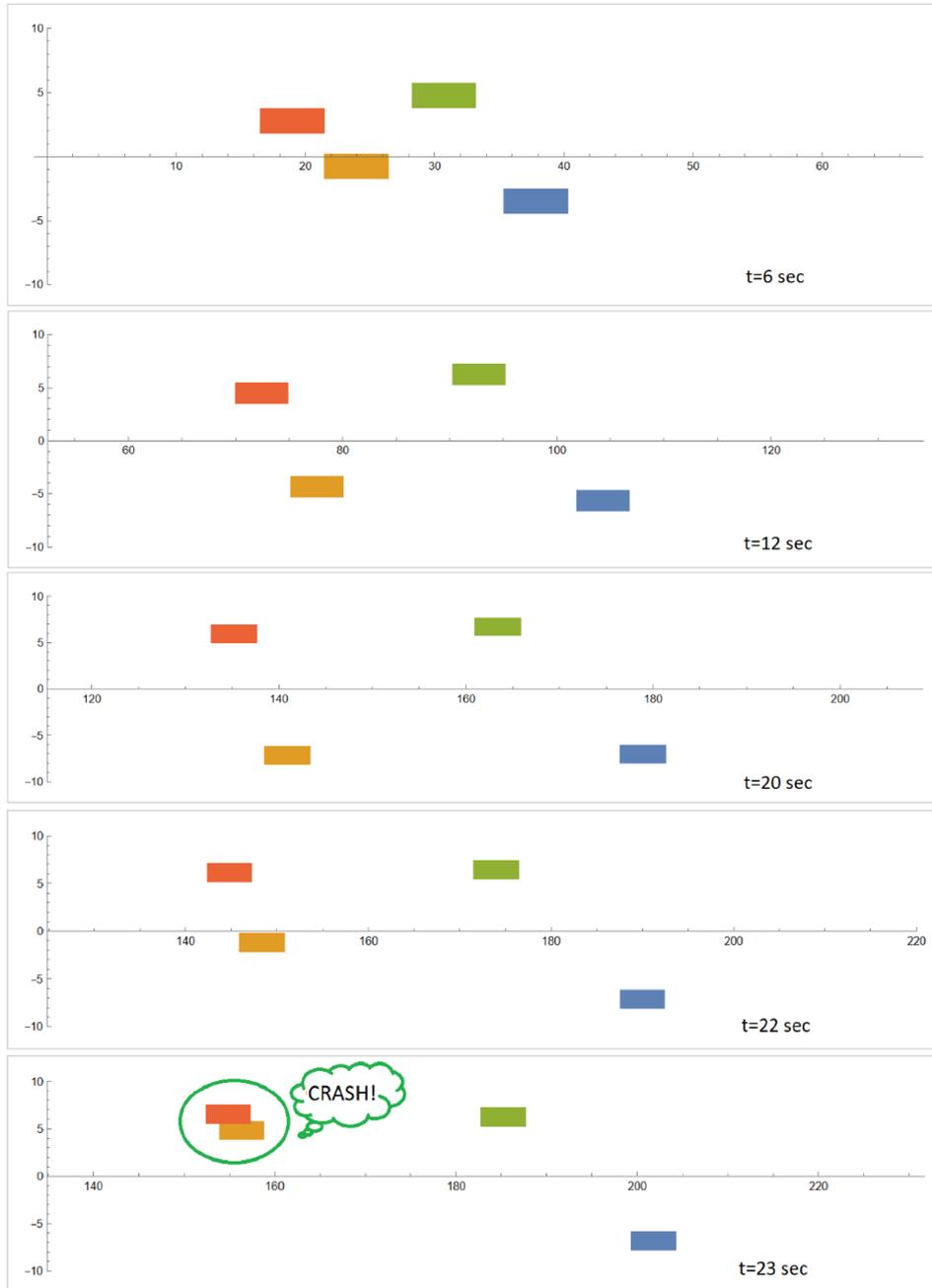


Figure 4.5: A real example with initial energy $H(0)=39.10$ and sampling period $T=0.1$.

We can realize that for specific potential functions, parameters, gains and constants, the only factors that can lead to a collision, are the initial energy and the sampling period. In Figure 4.6, we will see the same example as in Figure 4.5 adopting a smaller sampling period $T = 0.05$. We can observe that thanks to the smaller sampling period T , the orange vehicle managed to avoid the red one, so any collision took place. All parameters, gains, constants and potential function we used are presented in section 5.1.

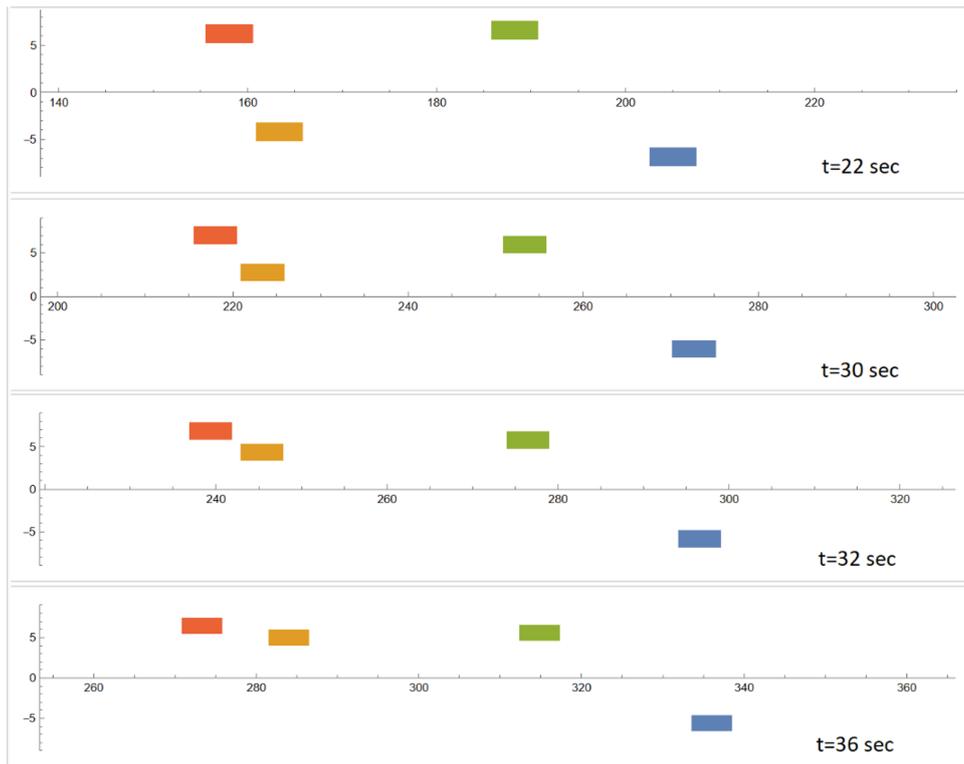


Figure 4.6: A real example with initial energy $H(0)=39.10$ and sampling period $T=0.05$.

Chapter 5

Numerical investigation of Maximum Allowable Sampling Period (MASP)

5.1 Selection of potential functions, parameters and gains

Taking the occasions above as an example, we can see in the following pages how the initial energy of the system can influence the Maximum Allowable Sampling Period (MASP). Namely, it is investigated the maximum $T > 0$ for which $w(t) \in \Omega$, $t \geq 0$ for all $w(0) \in \Omega$. In all simulations in this thesis, unless otherwise stated, the following functions will be used,

$$V(d) = \begin{cases} q_1 \frac{(\lambda-d)^3}{d-L} & L < d \leq \lambda \\ 0 & d > \lambda \end{cases} \quad (5.1)$$

$$U(y) = \begin{cases} \left(\frac{1}{\alpha^2 - y^2} - \frac{c}{\alpha^2} \right)^4 & -\alpha < y < -\frac{\alpha\sqrt{c-1}}{\sqrt{c}} \text{ or } \frac{\alpha\sqrt{c-1}}{\sqrt{c}} < y < \alpha \\ 0 & -\frac{\alpha\sqrt{c-1}}{\sqrt{c}} < y < \frac{\alpha\sqrt{c-1}}{\sqrt{c}} \end{cases} \quad (5.2)$$

$$\kappa(d) = \begin{cases} q_2(\lambda - d)^2 & L < d \leq \lambda \\ 0 & d > \lambda \end{cases} \quad (5.3)$$

$$f(x) = \frac{1}{2\varepsilon} \begin{cases} 0 & x \leq -\varepsilon \\ (x + \varepsilon)^2 & -\varepsilon < x < 0 \\ \varepsilon^2 + 2\varepsilon x & x \geq 0 \end{cases} \quad (5.4)$$

where $c \geq 1, q > 0$ are design parameters.

Notice that V and U above, satisfy (3.9), (3.10) and (3.11), (3.12), respectively. Simultaneously, κ and f satisfy (3.13) and (3.21), respectively. By appropriately selecting the constant q_1 , we can adjust the repulsion force of the potential V in (3.9) and, consequently, the magnitude of the acceleration F_i , see (3.17). In particular, for small values of q_1 , the values of V (and consequently the acceleration F_i) will be smaller away from L , but will increase more sharply as d approaches close to L . The constant $c \geq 1$ affects the final configuration of the vehicles relative to the boundary of the road. More specifically, for $c = 1$ we have that $U(y) = 0$ if $y = 0$, which will force the vehicles to form a single platoon in the middle of the road. For $c > 1$, we have that $U(y) = 0$ in a neighborhood around $y = 0$, and the vehicles' converged lateral positions in this case will be within the strip $-\frac{\alpha\sqrt{c-1}}{\sqrt{c}} \leq y \leq \frac{\alpha\sqrt{c-1}}{\sqrt{c}}$. More details can be found in [9].

In table 5.1, all the parameters and gains that we use in the simulations are presented.

n	10 vehicles
σ	5m
v_{\max}	35m/s
α	7.2m
v^*	30m/s
φ	0.25 rad
p	5.11
L	5.59m
ε	0.2
μ_1	0.5
μ_2	0.1
q_1	$3 \cdot 10^{-3}$
λ	25m
A	1
c	1.5

Table 5.1: The parameters and gains for all the simulations

The selection of these values was based on [9], which also includes a general description on the selection of appropriate gains.

5.2 MASP per initial energy

To study the MASP of the model numerically, we consider initial conditions $w(0) \in \Omega$ which satisfy $H(w(0)) = l \in \mathbb{R}_+$ for specific values of $l > 0$, where H is defined in (3.15). Finally, it is investigated both the case where the vehicles only measure the distance from neighbouring vehicles ($q_2 = 0$) and the case where the vehicles take into account the relative speeds of adjacent vehicles ($q_2 \neq 0$). When the parameter $q_2 \neq 0$, the function $\kappa(d)$ contributes to our system and consequently, the vehicles measure the neighbouring speeds apart from the distances.

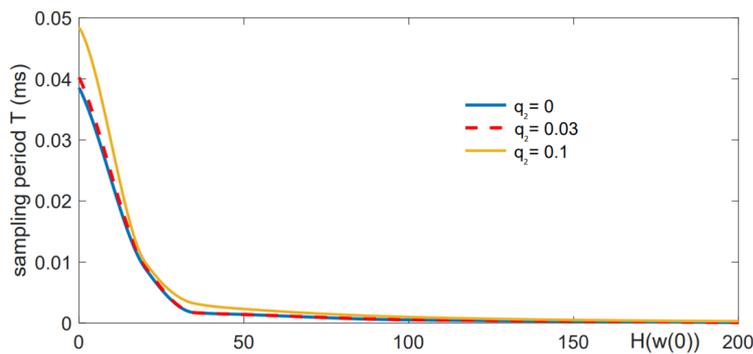


Figure 5.1: The sampling times T for increasing values of H, for different values of q_2 .

Observing Figure 5.1, we realize that while the initial energy increases, the maximum allowable sampling period tends to have minimal values. However, it is noteworthy that as the parameter q_2 takes greater values, the MASP is higher in comparison to smaller values of q_2 . More specifically, it is presented that for small values of the initial energy, the MASP is approximately 0.01 higher when $q_2 = 0.1$. This feature seems to be a substantial advantage on the controller since we can run simulations in less time. Finally, it should be noted that the approximation of the MASP above corresponds to the specific selection of the potential functions V, U, and the various constants associated with them. For a different selection, it is possible to obtain higher values for the MASP.

Chapter 6

State-dependent sampling period

In the previous Chapters, we considered that all the vehicles shared the same sampling period which is constant during the whole simulation. So, at each simulation we had to select a sampling period accordingly. Safety wise, the only guarantee we had when we selected the sampling period was the results of the Figure 5.1 that indicates the MASP per initial energy. Therefore, before the selection of the sampling period, we ought to find the initial energy related to the initial positions as a way to exploit the former results in terms of safety. However, while a constant sampling period for all the vehicles does not seem realistic, the computer needs more effort to complete the simulations. For these reasons, we are interested in alternative ways of sampling period calculation. The subsequent lemmas allow every vehicle to obtain its own sampling period according to:

- Its distance from the other vehicles.
- Its distance from the boundary of the road.
- Its orientation.

6.1 Lemma 1

Let $T > 0$ and consider model (4.2), (4.3) with $F_i(t) \equiv F_i$, $\omega_i(t) \equiv \omega_i$ for all, $t \in [0, T]$, $k = 1, \dots, n$. Let $w(0) \in \Omega$ be given, where Ω is defined by means of (3.8). Let also an arbitrary index $i \in \{1, \dots, n\}$ be given and suppose that for each $j = 1, \dots, n$ $j \neq i$ it holds that $v_j(T) \in (0, v_{\max})$ and $\theta_j(T) \in (-\varphi, \varphi)$. Moreover, assume that the following inequalities hold

$$T < \min \left\{ \frac{a - |y_i(0)|}{v_{\max} \sin(\varphi)}, \frac{\min_{j \neq i} \{d_{i,j}(0) - L\}}{\delta} \right\} \quad (6.1)$$

$$-\frac{v_i(0)}{T} < F_i(0) < \frac{v_{\max} - v_i(0)}{T} \quad (6.2)$$

$$\frac{-\varphi - \theta_i(0)}{v_i(0)T + \frac{1}{2}F_iT^2} < \omega_i(0) < \frac{\varphi - \theta_i(0)}{v_i(0)T + \frac{1}{2}F_iT^2} \quad (6.3)$$

where $\delta = \sqrt{2}v_{\max}\sqrt{1 + (2p - 1)\sin^2(\varphi)}$. Then, $v_i(t) \in (0, v_{\max})$, $\theta_i(t) \in (-\varphi, \varphi)$, $y_i(t) \in (-\alpha, \alpha)$ and $d_{i,j}(t) > L$, for all $t \in [0, T]$, $j = 1, \dots, n$ $j \neq i$. So, it is confirmed that when all the conditions mentioned above hold, then for the vehicle i , there is not any crash neither with

other vehicles nor with the boundaries of the road. **This holds only until the very next sampling.** Lemma 1 provides an upper bound on the sampling period for a vehicle i to satisfy property **P1**. In practice, each vehicle has its own internal clock and may also have its own sampling period, even in the case of identical vehicles. So, lemma 1 can be exploited for each vehicle autonomously. In simple words, each vehicle will be able to choose its own sampling period based on the distance of the other vehicles and the road boundaries. In the next pages, simulations will be projected according to this method.

6.2 Lemma 2

Let $w(0) \in \Omega$ be given, where Ω is defined by means of (3.8) and consider model (4.2), (4.3) with $F_i(t) \equiv F_i$, $\omega_i(t) \equiv \omega_i \in \mathbb{R}$, $i = 1, \dots, n$ being constant for all $t \in [0, T]$. Suppose also that (6.1), (6.2) and (6.3) hold for all $i = 1, \dots, n$. Then $w(t) \in \Omega$ for all $t \in [0, T]$.

Lemma 2 suggests that if T is the same for all vehicles and satisfies (6.1), (6.2) and (6.3) for all $i = 1, \dots, n$ then the following implication holds: $w(0) \in \Omega \Rightarrow w(t) \in \Omega$ for all $t \in [0, T]$. So, the lemma 2 can be exploited for non-periodic sampling with all vehicles having the same sampling period T . More specifically, at each sampling every vehicle calculates its own sampling period. After this calculation all the vehicles adopt the smallest sampling period so the lemma 2 can be applied.

The proofs of the Lemma 1 and Lemma 2 are presented in [17].

Remark 3: Classical sampled-data control is based on performing sensing and actuation periodically rather than when the system needs attention, see Chapter 2.1. Self-triggered control, on the other hand, is proactive and computes the next sampling or actuation instance ahead of time. The term self-triggered control was coined by [19] in the context of real-time systems. A self-triggered implementation of the feedback control laws (3.16), (3.17) has for objective the computation of the actuator values as well as the computation of the next instant of time at which the control law should be recomputed. The state-dependent bounds of Lemma 1, essentially imply that the considered approach is based on self-triggered control. This is illustrated on Chapter 7.

Chapter 7

Applications of state-dependent sampling-period

From now on, we can exploit the lemmas of the previous Chapter to create alternative methods to calculate the sampling period(s) during the simulations. More specifically, these lemmas allow non-periodic and state-dependent sampling periods to be calculated, reducing the simulation time significantly. In the following pages, we investigate three scenarios to calculate the sampling period, observing the simulation time contextually. In addition, we examine some noteworthy features in both cases when $q_2 = 0$ and $q_2 \neq 0$.

7.1 Scenario 1

In this scenario, the sampling period T does not change during the simulation. This implies that all the vehicles have the same constant T . Particularly, in Figure 7.1 we can see a flow chart which describes in short how the algorithm operates in Scenario 1. As we can see, the sampling period T is constant until the end of the simulation. The episode ends when a crash occurs or when the simulation time is complete.

We run simulations with the same design parameters, constants, gains and potential functions as in Chapter 5 adopting the initial positions which are presented in table 7.1:

vehicles	1	2	3	4	5	6	7	8	9	10
x	17.8254	39.7647	55.2647	32.0363	70.9238	57.1023	100.0467	0.4	12.7664	87.7138
y	-0.6695	-3.3109	0.5742	5.8221	4.3965	-6.8737	2.4528	-4.2688	-6.1721	6.9079
θ	-0.0069	-0.0084	0.0252	-0.0017	-0.0292	-0.0391	0.015	0.0356	-0.0244	-0.0255
v	24.0623	34.7188	34.211	28.1151	33.5243	22.7974	32.2472	25.0238	31.8568	30.4904

Table 7.1: Initial positions for Scenarios 1, 2 and 3.

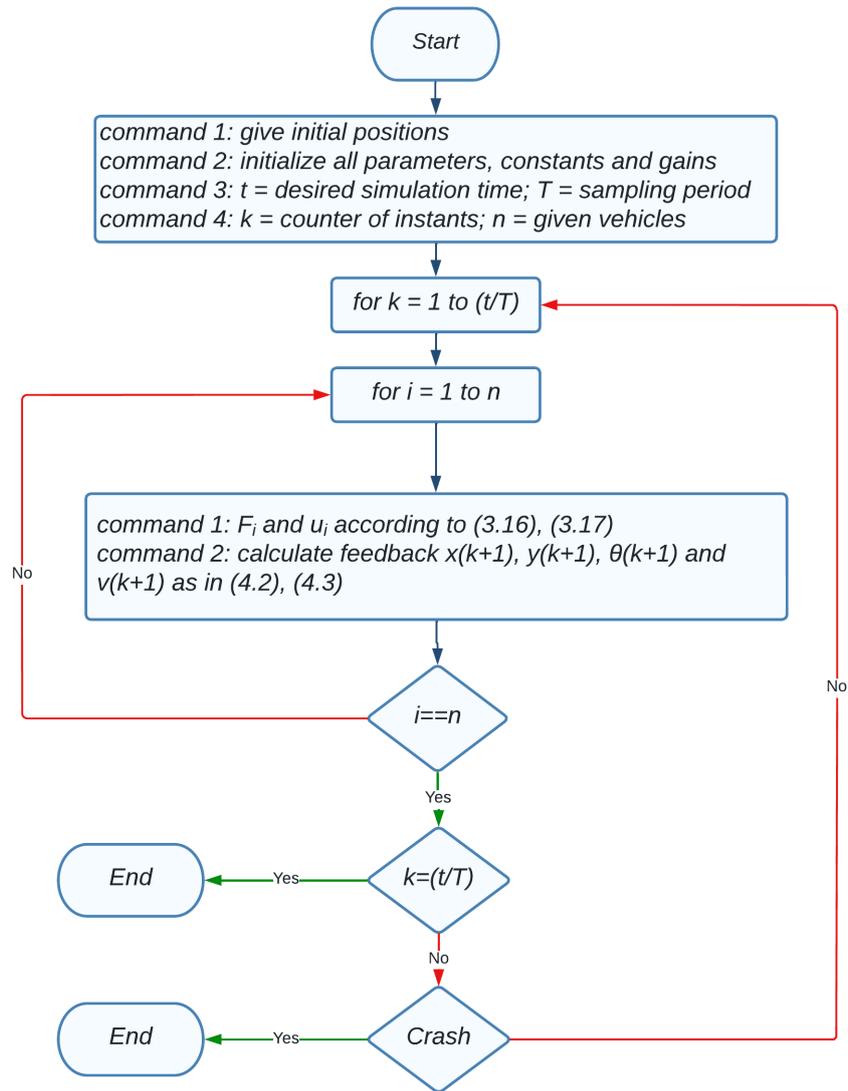


Figure 7.1: Description of scenario 1 calculation method.

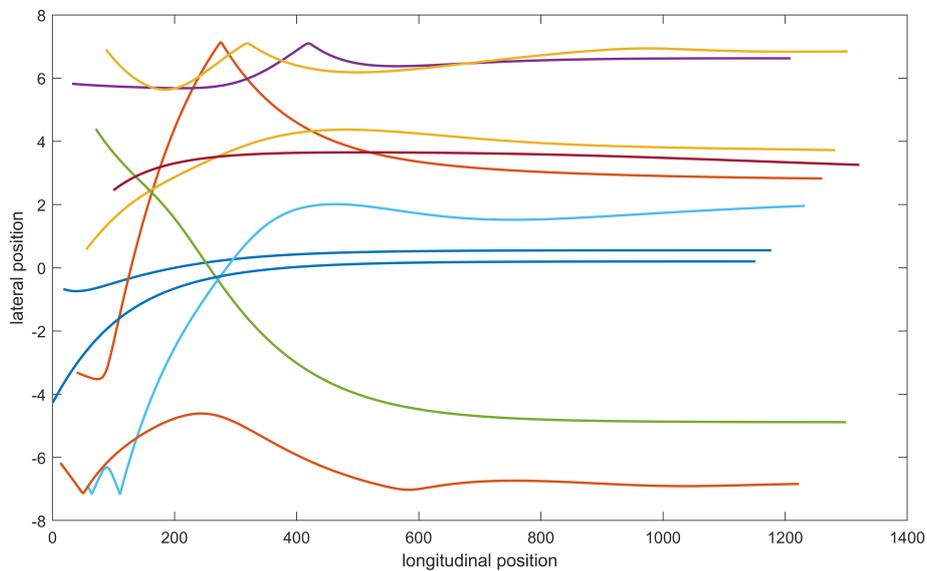


Figure 7.2: Vehicles' trajectories for $q_2=0$ and $T=0.01$.

Figure 7.2 shows the trajectories of the vehicles when $q_2 = 0$ and $T = 0.01$. It is verified that the vehicles remain within the boundaries of the road. Notice that the intersection of the trajectories does not imply collision between vehicle, since the minimum inter-vehicle distance is greater than L , see Figure 7.3. Properties like the nudging effect and the lane free principle are easily observed. For clarity, the trajectories of 2 vehicles are not shown in the Figure.

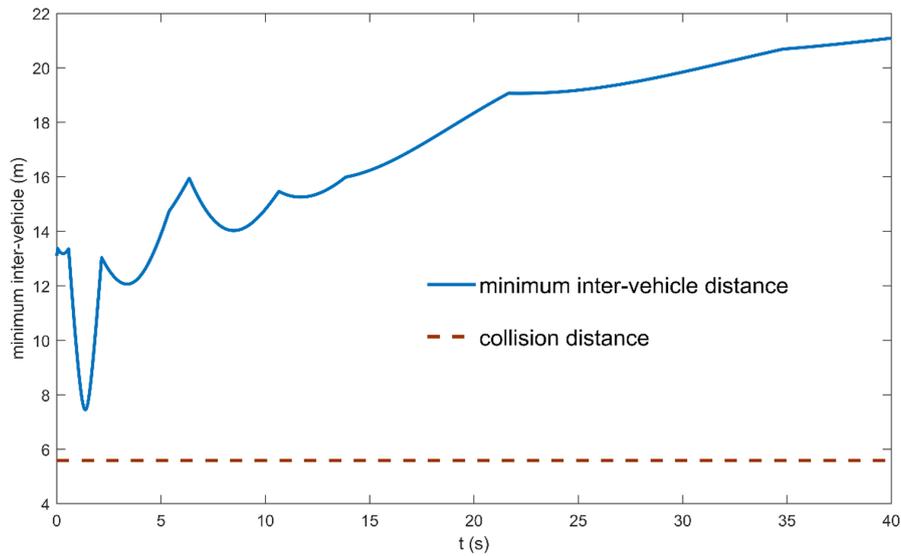


Figure 7.3: Minimum inter-vehicle distance for $q_2=0$ and $T=0.01$.

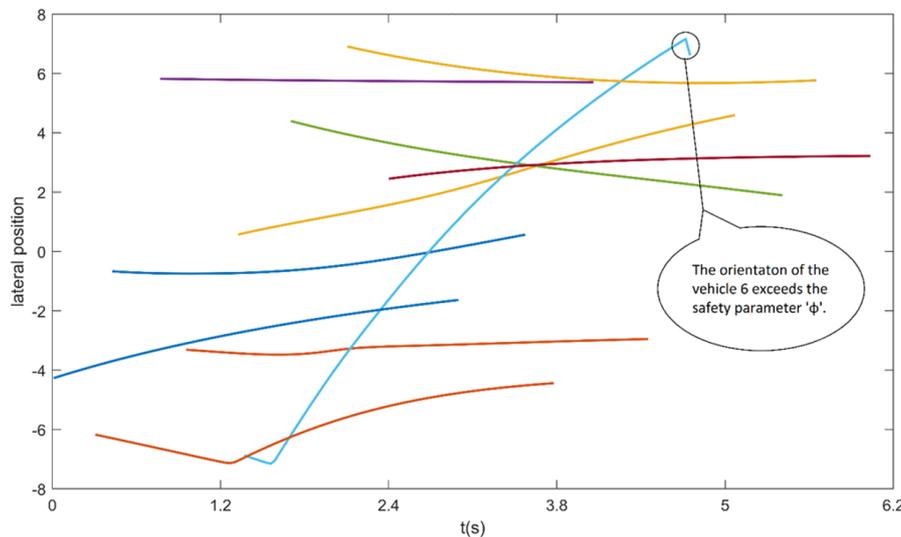


Figure 7.4: Vehicles' trajectories for $q_2=0$ and $T=0.03$.

Figure 7.4 shows the occasion when $q_2 = 0$ and $T = 0.03$. We can see that the 6th vehicle was forced to turn dangerously, taking a perpendicular orientation higher than the safety limit. This condition can more than likely lead to a crash either on other vehicles or on the road boundary. On this occasion, a collision with the road boundary occurred after some seconds. We can realize that the high sampling period $T = 0.03$ was the main reason this occurred since the 6th vehicle was late to do its subsequent sampling, and consequently, it was forced to obtain high values in terms of orientation while approaching the road. The same effect is mentioned in the example in Chapter 4.

In Figure 7.5, we can see how the orientation θ changes in both cases when $T = 0.1$ and $T = 0.03$. When $T = 0.1$ the blue line is smoother because more samplings take place per second. When $T = 0.3$ the line changes more abruptly because the sampling period lasts more. As we discussed before, this is the reason why the orientation took extremely high values, leading the vehicle to crash.

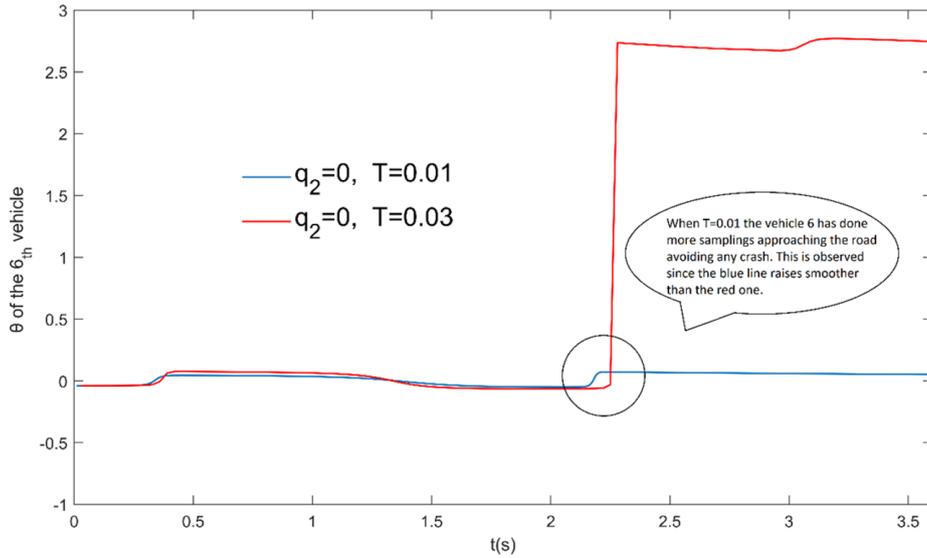


Figure 7.5: Orientation of the 6_{th} vehicle for $q_2=0$.

Let us investigate the occasion when $q_2 \neq 0$ assuming $q_2 = 0.001$. In Figure 7.6, where $T = 0.044$, we can realize that there is not any collision with the road boundary. Moreover, there is not any crash nor among the vehicles, too. Consequently we realize that while q_2 became higher, the sampling period T (for the same conditions as the previous example for $q_2 = 0$) had margin to be higher, as well. In addition, the vehicles' trajectories are smoother compared to Figure 7.2. This feature illustrates a substantial advantage when the controller measures the speed of the adjacent vehicles. Apart from the sampling period, in the next pages we will see that when $q_2 \neq 0$ we take more efficient results in terms of vehicles' acceleration.

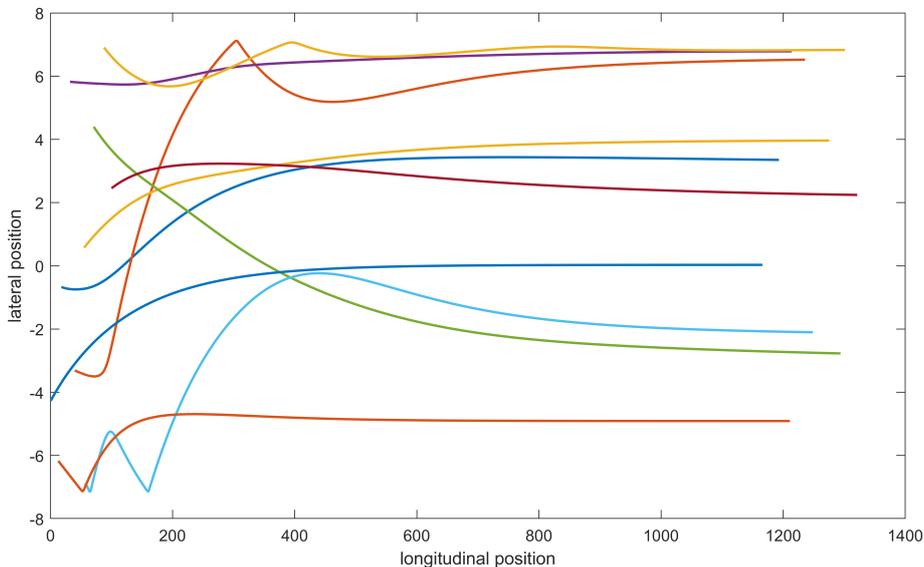


Figure 7.6: Vehicles' trajectories for $q_2=0.001$ and $T=0.044$.

In Figure 7.7, when $q_2 = 0.001$ and $T = 0.048$, it is projected the same issue as in Figure 7.4. The sampling period was too high, so the 6th vehicle was forced to obtain a dangerous orientation, more extensive than the safety limit. In this case, a crash on the road boundary occurred some seconds later, like before.

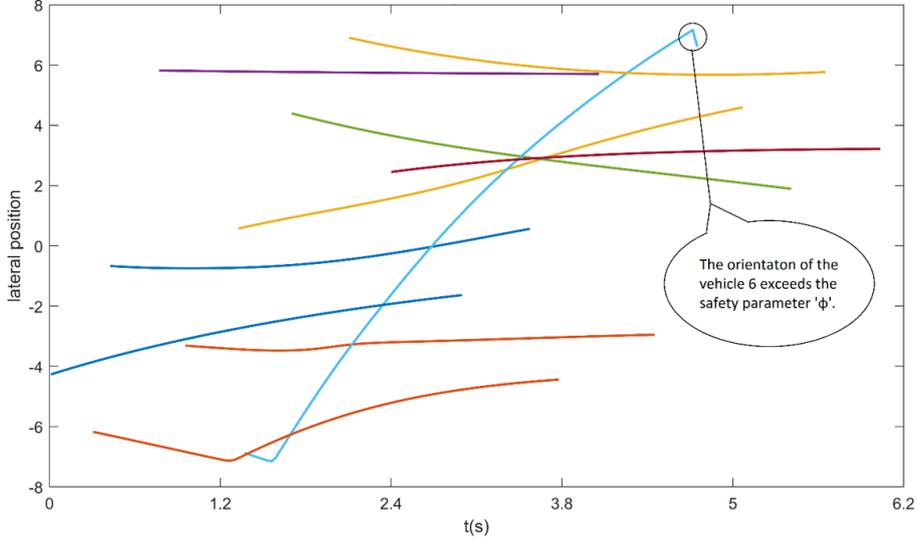
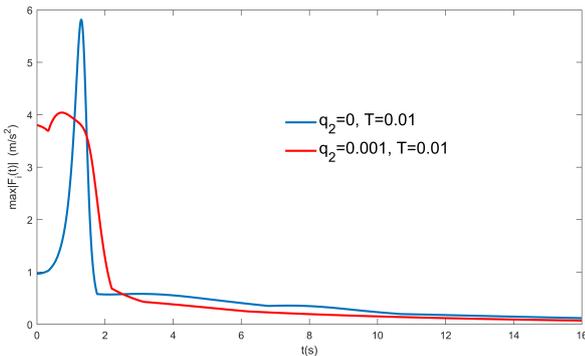
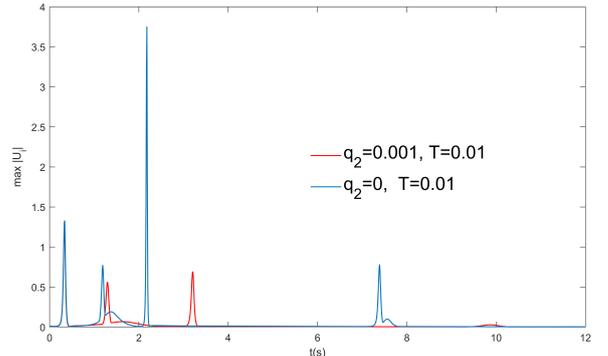


Figure 7.7: Vehicles' trajectories for $q_2=0.001$ and $T=0.048$.

To understand better the properties and the differences when $q_2 = 0$ and $q_2 \neq 0$ we run a simulation where the sampling period is the same, letting only the parameter q_2 to do the difference. Assume $T = 0.01$. As we can see in Figure 7.8a, the existence of the value q_2 efficiently helps the systems to maintain small accelerations. Although the initial maximum acceleration is larger when $q_2=0.001$, by taking into account the speeds of neighboring vehicles, the system can maintain its maximum acceleration at lower values, in general. This feature seems extremely important since it improves the driving experience. Evidently, the former feature would influence the value of the u_i . In Figure 7.8b it is presented that the values of u_i take much lower values when $q_2 \neq 0$. Particularly, when $q_2 \neq 0$, the maximum $u_i = 1.3144$, whereas when $q_2 = 0$, the maximum $u_i = 3.7539$. This means that the orientation of each vehicle converges to the desired value smoother when the system measures the speeds of the adjacent vehicles. that the values of u_i take much lower values when $q_2 \neq 0$. Particularly, when $q_2 \neq 0$, the maximum $u_i = 1.3144$, whereas when $q_2 = 0$, the maximum $u_i = 3.7539$. This means that the orientation of each vehicle converges to the desired value smoother when the system measures the speeds of the adjacent vehicles.



(a) Vehicles' max $|F_i|$ for $T=0.01$.



(b) Vehicles' max $|u_i|$ for $T=0.01$.

Figure 7.8: Max $|F_i|$ and $|u_i|$ for $T=0.01$.

7.2 Scenario 2

In this Scenario, the sampling period T is not constant as in Scenario 1 but is dynamically calculated on the basis of Lemma 2. Lemma 2 illustrates that if all the vehicles have the same sampling period T , and (6.1), (6.2), (6.3) are applied for all the vehicles, then until the very next sampling there is not any collision to take place neither among vehicles nor with the boundary of the road. More specifically, T is calculated according to the formula:

$$T_k = \max \left\{ \alpha, \beta \min_{i=1, \dots, n} \left\{ \frac{a - |y_i(t_k)|}{v_{\max} \sin(\varphi)}, \frac{\min_{j \neq i} \{d_{i,j}(t_k) - L\}}{\delta} \right\} \right\}$$

where $t_{k+1} = t_k + T_k$, $k = 0, 1, \dots$, $\alpha > 0$, $\beta \in (0, 1)$. The constant $\alpha > 0$ provides a lower bound on the sampling period, and β adjusts the sampling period in order for (6.1) to hold. We set α as a value which is slightly smaller than the relative MASP and $\beta = 0.9$. We use the constant α since we want to avoid computational errors. Let's describe an example how the algorithm works. Initially, the formula calculates, the value $\min \left\{ \frac{a - |y_i(t_k)|}{v_{\max} \sin(\varphi)}, \frac{\min_{j \neq i} \{d_{i,j}(t_k) - L\}}{\delta} \right\}$, for each vehicle, per sampling. Then, it keeps the lowest of all values among the vehicles and multiplies it with the parameter β . Finally, compares this product with the constant α and adopt the highest of these two values. Then, the algorithm has found the sampling period until the next sampling. In the next sampling the same process will take place. In the following snapshots the former procedure is presented.

S1 We assume 4 vehicles with random initial positions.

S2 We consider the value $m(i) = \min \left\{ \frac{a - |y_i(t_k)|}{v_{\max} \sin(\varphi)}, \frac{\min_{j \neq i} \{d_{i,j}(t_k) - L\}}{\delta} \right\}$ for $i = 1, \dots, n$.

S3 We multiply the smallest value of vector m with the parameter β and we get the sampling period T for all the vehicles.

S4 We go to the next sampling.

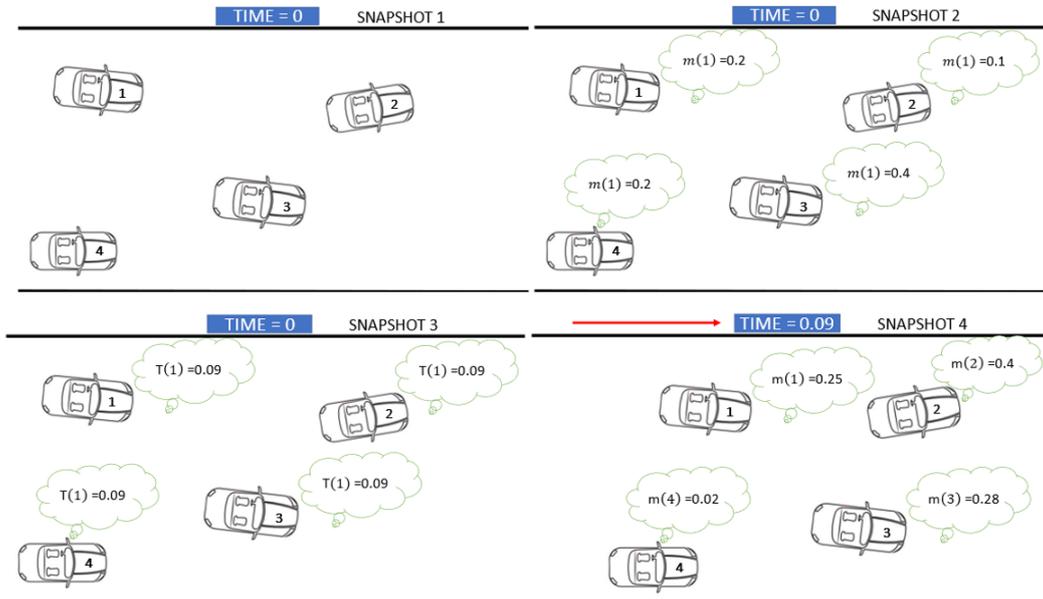


Figure 7.9: Sampling period calculation in Scenario 2.

Flow chart in Figure 7.10 describes how Scenario 2 works in general. It should be underlined that the algorithm terminates with a different method in comparison to Scenario 1 since the system is not aware of how many loops are needed. Now, using the same system features as in Scenario 1 and the same initial positions, we are in position to present the following results.

Observing Figure 7.11, we can realize how the sampling period changes throughout the simulation. Initially, we see that the dotted lines concern the occasion when the sampling period is calculated periodically. The red dotted line describes the case where we do not measure the speed of the neighbouring vehicles ($q_2 = 0$), whereas the blue one indicates the case where we take into account the speeds of the adjacent vehicles ($q_2 \neq 0$). Conversely, the continuous lines depict the occasion when the sampling period changes non-periodically. The red line concerns the case where $q_2 = 0$ and the blue line the occasion where $q_2 \neq 0$. In the first 40 seconds, we observe that when $q_2 \neq 0$, the sampling period tends to note intense fluctuations. This occurs since the energy of the system obtains its highest values at the beginning of the simulation. While the episode is taking place, the energy approaches minimal values and the system tends to convergence. So, the distances among the vehicles and from the road boundary do not have substantial changes. So according to (6.1), we expected this phenomenon as the sampling period in Scenario 2 is influenced by these distances. Besides that, a quite noteworthy feature is that in both cases when $q_2 \neq 0$ and $q_2 = 0$ the sampling periods exceed the relative values of Scenario 1. Moreover, when $q_2 \neq 0$ the sampling periods tend to higher values.

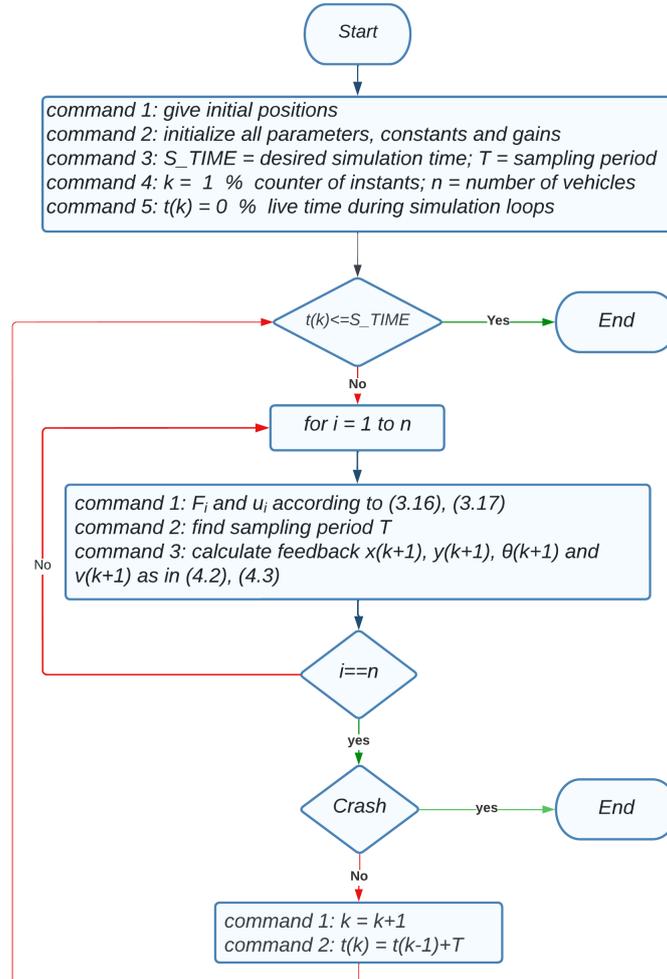


Figure 7.10: Description of scenario 2 calculation method.

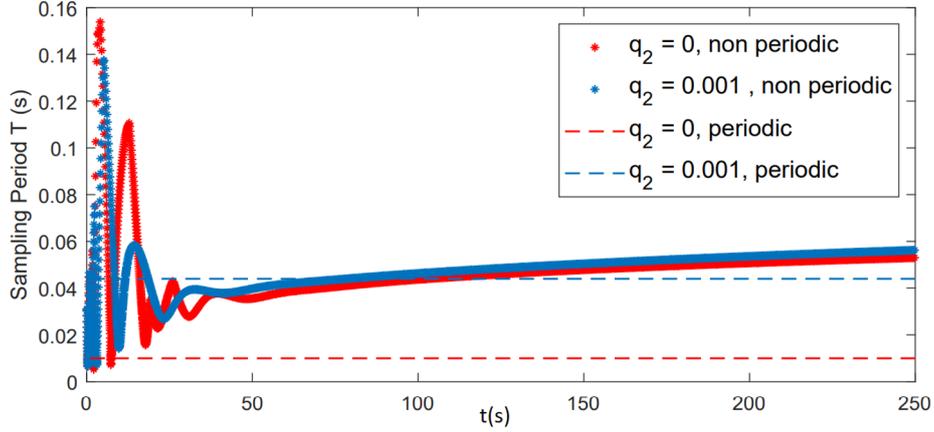


Figure 7.11: The sampling periods during simulations in both scenarios 1 and 2 when $q_2 = 0$ and $q_2 = 0.001$.

Table 7.2, shows that for the time-interval $[0, 300s]$, the execution's runtime for Scenario 2 was 47s for $q_2 = 0$ and 41s for $q_2 \neq 0$. Particularly, as execution time, we mean the real time that the computer needs to run an episode, whereas simulation time is a variable which indicates how long the simulation is, in the programming environment. It is evident that for higher simulation times, we obtain higher execution times when all the system features are the same. We used a machine powered by Intel Core i5-6200, 2.30 Ghz, with Matlab. For Scenario 1, the execution runtime was 204s for $q_2 = 0$ and 80s for $q_2 \neq 0$. These results depict how the non-periodic sampling assisted the computer to run simulations in much less time. However, note that the computation of T_k calls for data from all vehicles, hence this handling of sample time may be valuable to speed up the simulation, but it is not for use by real vehicles. In Scenario 3, we consider state-dependent sample times, which are different for each vehicle and can be computed based on decentralized data.

Simulation Time 300s	$q_2 = 0$	$q_2 = 0.001$
Scenario 1	204s	80s
Scenario 2	47s	41s

Table 7.2: Execution times when the simulation time equals to 300s.

Figure 7.12a shows that the existence of q_2 can assist the system to maintain lower and more realistic accelerations. We observed the same feature in the Figure 7.8a in Scenario 1. More specifically, when $q_2 = 0$, the maximum acceleration tends to obtain values close to $6 m/s^2$ whereas when $q_2 \neq 0$, the maximum acceleration marginally exceeds $4 m/s^2$. In addition, it is observed that the system converges faster when $q_2 \neq 0$. It should be mentioned that this characteristic sometimes can be considered as a disadvantage since the vehicles will probably have sharper movements in order to reach a faster convergence. At the same time, faster convergence of the system in terms of velocity is presented when measuring the speed of the neighbouring vehicles, see Figure 7.12b. Particularly, when $q_2 \neq 0$, we can see that the absolute values of the quantity $|v_i(t) - v^*|$ tend to zero more quickly, during the episode.

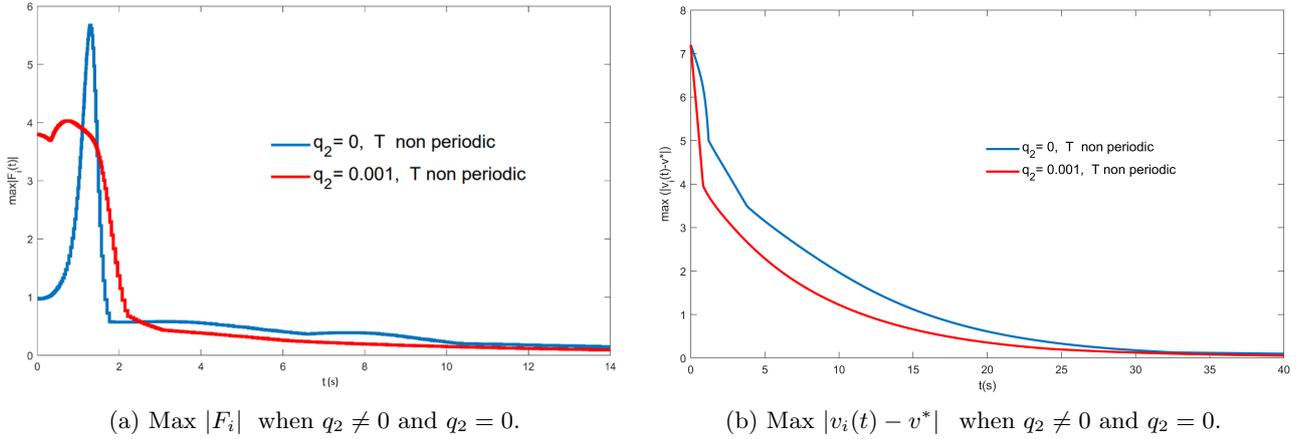


Figure 7.12: Max $|F_i|$ and $|v_i(t) - v^*|$ when $q_2 \neq 0$ and $q_2 = 0$.

7.3 Scenario 3

In this scenario, we assume that each vehicle has its own sampling period T_i which changes dynamically at each sampling instant. Specifically, each sampling period is computed according to Lemma 1 and depends only on the distance from the neighbouring vehicles and the distance from the boundary of the road. We can see the formula how each sampling period is calculated below:

$$T_k^i = \max \left\{ \alpha, \beta \min \left\{ \frac{a - |y_i(t_k^i)|}{v_{\max} \sin(\varphi)}, \frac{\min_{j \neq i} \{d_{i,j}(t_k^i) - L\}}{\delta} \right\} \right\}$$

where,

$$t_{k+1}^i = t_k^i + T_k^i, k = 0, 1, \dots, \alpha > 0, \beta \in (0, 1)$$

The parameters α and β exist for the same reasons as in scenario 2. Since the sampling period of each vehicle depends on the distance from adjacent vehicles, it is evident that each vehicle has to estimate the position of the other vehicles. However, the vehicles have separate sampling periods and consequently different sum of samplings. In the former scenarios, the sampling period of all the vehicles was calculated simultaneously and it was easy for each vehicle to compute the other vehicles' position. Now, when a vehicle calculates its sampling period, the other vehicles don't probably calculate their own ones. As a result, each vehicle has to do extra calculations to compute the positions of the other vehicles. Flow chart in Figure 7.13 presents the method how this estimation takes place in short. Besides, an specific example how this procedure takes place is presented in Appendix.

Following the this algorithm, we run simulations with the same features as in the previous scenarios in both cases when $q_2 \neq 0$ and $q_2 = 0$. Initially, in Figures 7.14 and 7.18, we can observe the trajectory of each vehicle and the inter-vehicle minimum distance, respectively. As it is obvious any collision takes place with neither with the boundary of the road nor among the vehicles. Figures 7.15a and 7.15b show that when the vehicles take into account the speeds of the adjacent vehicles, the maximum acceleration reduces and the convergence of the system succeeds faster, as well. When realizing that we noticed exactly the same feature in the previous scenarios, we are in position to say that the method of sampling period calculation cannot change the properties of the existence of q_2 , in terms of acceleration and convergence time.

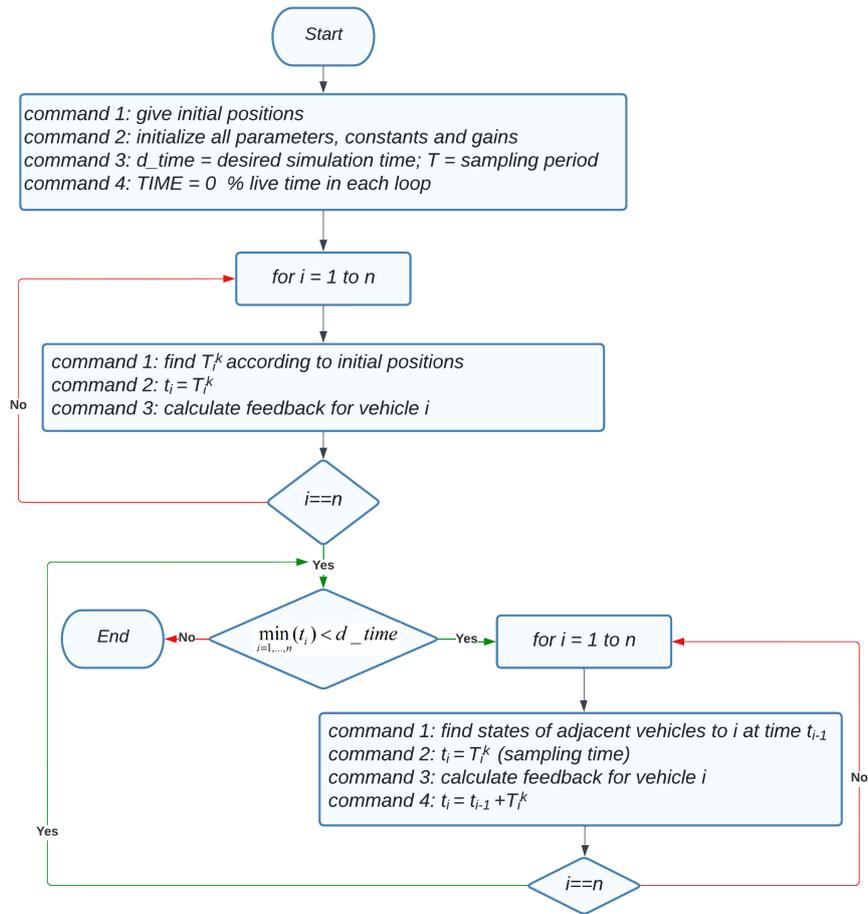


Figure 7.13: Description of scenario 3 calculation method.

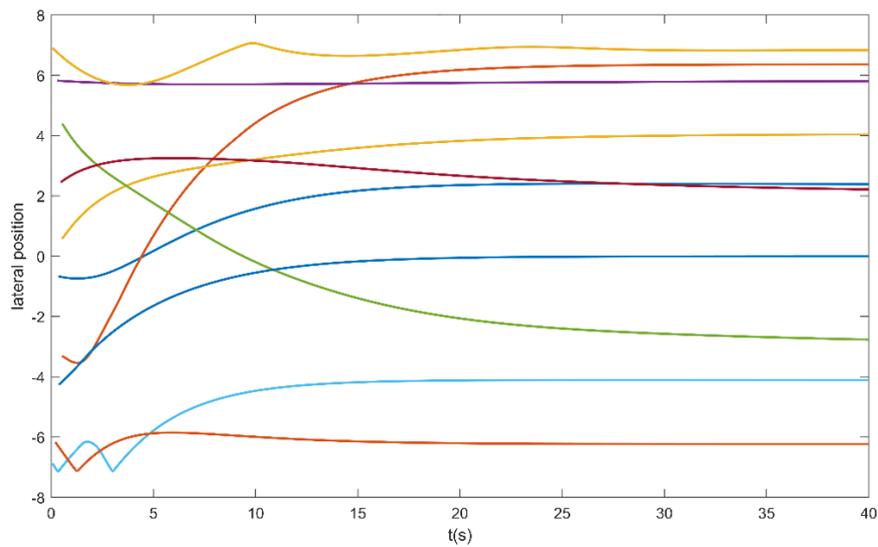
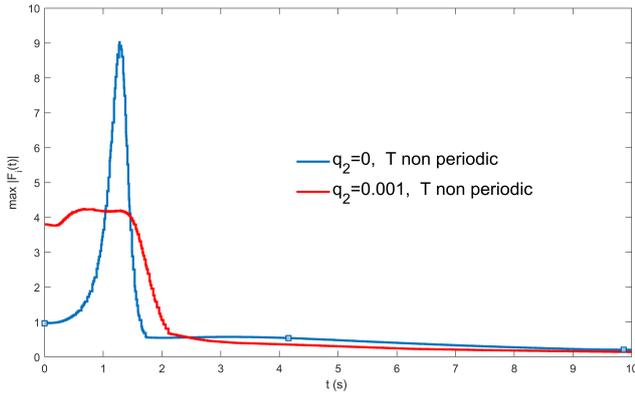
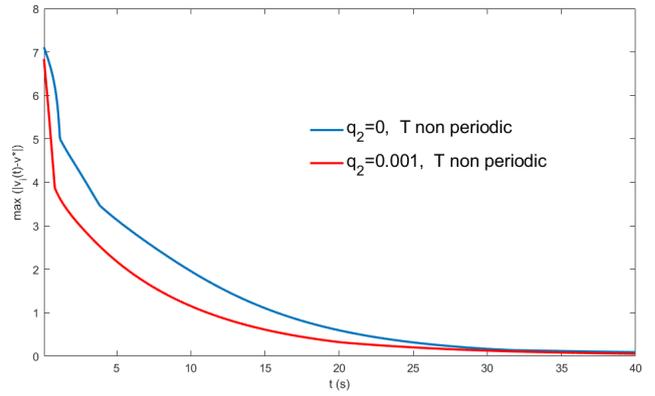


Figure 7.14: Vehicles' trajectories when $q_2=0.001$.



(a) Vehicles' max $|F_i|$ for $T=0.01$ when $q_2 \neq 0$ and $q_2 = 0$.



(b) Max $|v_i(t) - v^*|$ when $q_2 \neq 0$ and $q_2 = 0$.

Figure 7.15: Max $|F_i|$ and $|v_i(t) - v^*|$ when $q_2 \neq 0$ and $q_2 = 0$.

Figures 7.16 and 7.17 depict the sampling period of 4 vehicles during the simulation when $q_2 = 0$ and $q_2 \neq 0$, respectively. In these Figures, we can see that each vehicle autonomously and separately calculates its sampling period. It is noteworthy to focus on the sampling periods of the 10th vehicle. As we can see, the sampling periods do not exceed 0.05 since the vehicle is forced to be near to the road boundary since another vehicle is moving closely to it. As we observed in (6.1) the sampling period of each vehicle is exclusively influenced from its distance from the other vehicles and from the boundary of the road, see Figure 7.14.

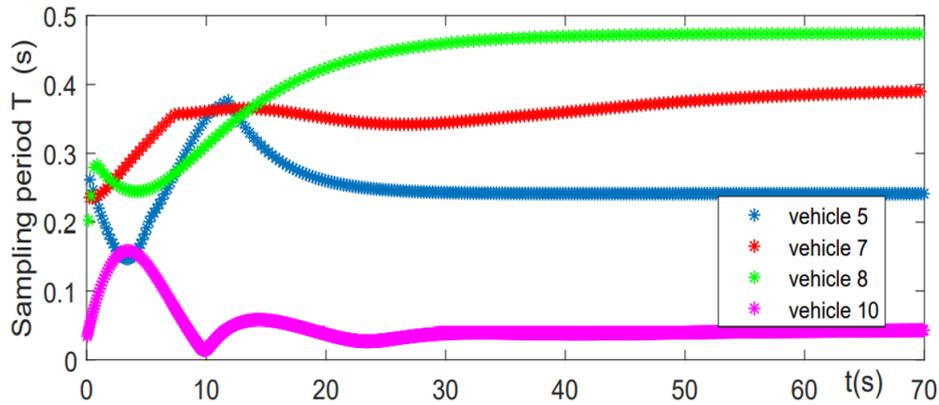


Figure 7.16: The sampling periods of 4 of the vehicles during the episode for $q_2 = 0$.

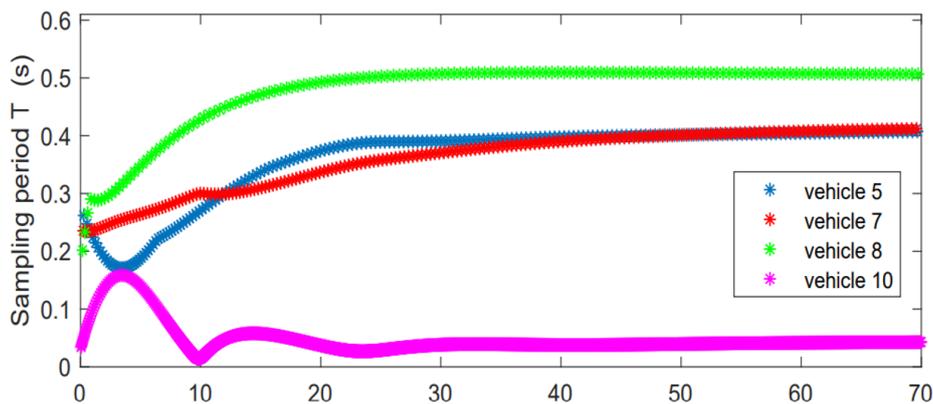


Figure 7.17: The sampling periods of 4 of the vehicles during the episode for $q_2 = 0.001$.

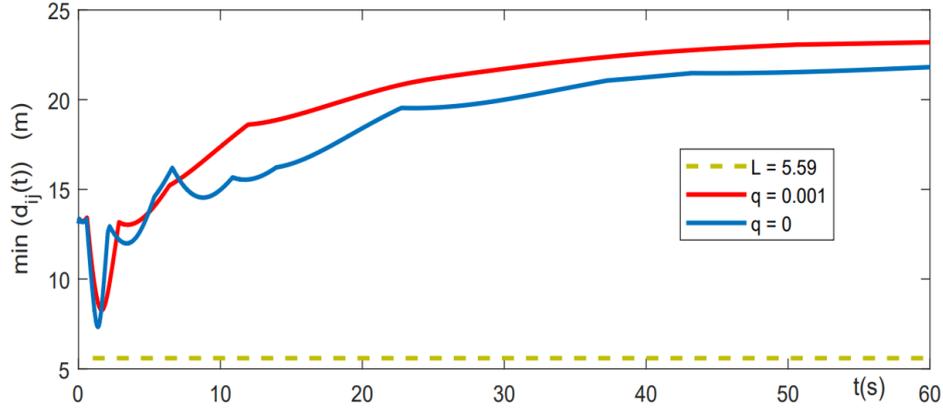


Figure 7.18: The minimum inter-vehicle distance when $q_2 \neq 0$ and $q_2 = 0$.

Finally in tables 7.3 and 7.4 we can compare the execution times and the average sampling periods for all the scenarios when the simulation time was 300s. We can observe that the $T_{average}$ for the 2nd scenario increased from 0.051 when $q_2 = 0$ to 0.054 when $q_2 \neq 0$. This is the reason why the execution time is lower when $q_2 \neq 0$, as we observed in the 2nd scenario. The most remarkable feature is that despite the 3rd Scenario obtains the highest $T_{average}$ by far, the execution time is not the shortest. This phenomenon occurs because in Scenario 3, in each sampling, the computer must calculate the temporary position of the other vehicles. This procedure forces the computer to spend more time to complete the episode. Consequently, although the 3rd scenario seems the most realistic, it obtains lower execution times in comparison to the scenario 2.

Execution times	$q_2 = 0$	$q_2 = 0.001$
Scenario 1	204s	80s
<u>Scenario 2</u>	<u>47s</u>	<u>41s</u>
Scenario 3	69s	60s

Table 7.3: Execution times when the simulation time equals to 300s.

$T_{average}$	$q_2 = 0$	$q_2 = 0.001$
Scenario 1	fixed	fixed
Scenario 2	0.051	0.054
<u>Scenario 3</u>	<u>0.149</u>	<u>0.159</u>

Table 7.4: Average sampling periods when the simulation time=300s.

Chapter 8

Saturated control input

Saturation effects occur when any part of a feedback control system reaches a natural limit since every physical actuator is subject to saturation, see [6]. In such cases, the performance of the closed-loop system designed without considering actuator saturation may deteriorate or even lose its stability properties. In control systems, every physical actuator or sensor is subject to saturation owing to its maximum and minimum limits. A saturation function $sat : \mathbb{R} \rightarrow [a, b] \subset \mathbb{R}$ can be symmetrical or asymmetric as below

$$sat(x) = \begin{cases} b & x \geq b \\ x & x \in (a, b) \\ a & a \leq x \end{cases} \quad (8.1)$$

where $a, b \in \mathbb{R}$.

As already discussed in the previous Chapters, the initial energy of the system $H(w(0))$ plays a crucial role in the values of the acceleration (see Remark 2) and also in the sampling period T . High initial energy implies that the initial speed of the vehicles are away from the desired speed set-point, or that the distances between vehicles or between vehicles and the boundary of the road is very small. In such cases, the acceleration F_i may take high values for vehicles to avoid collision and remain within the road. In practical applications however, there are physical limitations. We study below the performance of the system due to saturations on the control inputs of the system.

In the following examples, we examine two occasions. The first example illustrates a case in which the system adopts saturations and the initial energy of the system equals to 81.6431. Conversely, in the second occasion we changed the initial positions and the initial energy increased to 89.7691. The initial positions are shown in tables 8.1 and 8.2 respectively:

vehicles	1	2	3	4	5	6	7	8	9	10
x	17.8254	33.7647	55.2647	32.0363	70.9238	57.1023	100.0467	0.4	12.7664	87.7138
y	-0.6695	-0.109	0.5742	5.8221	4.3965	-6.8737	2.4528	-4.2688	-6.1721	6.9079
θ	-0.0069	-0.0084	0.0252	-0.0017	-0.0292	-0.0391	0.015	0.0356	0.0244	-0.0255
v	33.0623	25.7188	34.211	28.1151	33.5243	22.7974	32.2472	25.0238	31.8568	30.4904

Table 8.1: Initial positions with initial energy $H(0)=81.6431$.

vehicles	1	2	3	4	5	6	7	8	9	10
x	17.8254	33.7647	55.2647	32.0363	70.9238	57.1023	100.0467	0.4	12.7664	87.7138
y	-0.6695	-0.109	0.5742	5.8221	4.3965	-6.8737	2.4528	-4.2688	-6.1721	6.9079
θ	-0.0069	-0.0084	0.0252	-0.0017	-0.0292	-0.0391	0.015	0.0356	0.0244	-0.0255
v	35.0623	25.7188	34.211	28.1151	33.5243	22.7974	32.2472	25.0238	31.8568	30.4904

Table 8.2: Initial positions with initial energy $H(0)=89.7691$.

To run the following simulations, we used periodic sampling with $q_2 = 0$ and $T = 0.01$. We used $q_2 = 0$ since higher accelerations can take place and the effects of saturation usage can be shown easier. However we can use saturations either when the sampling is periodic or non-periodic. Hereinafter, we use the saturation function $sat : \mathfrak{R} \rightarrow [-6, 3.5] \subset \mathfrak{R}$:

$$sat(x) = \begin{cases} 3.5 & x \geq 3.5 \\ x & x \in (-6, 3.5) \\ -6 & x \leq -6 \end{cases} \quad (8.2)$$

We will see that when the acceleration takes values that exceed 3.5 m/s^2 , then the relative vehicle will maintain its acceleration to 3.5 m/s^2 . On the other hand, when the acceleration will be lower than -6 m/s^2 , the relative vehicle will keep this value.

In Figure 8.1 is shown that when $H(w(0)) = 81.6431$ the vehicles obey the boundaries related to acceleration as mentioned before. Especially two vehicles tend to reach the saturation limits but the system does not allow them to exceed the relative values, as discussed before.

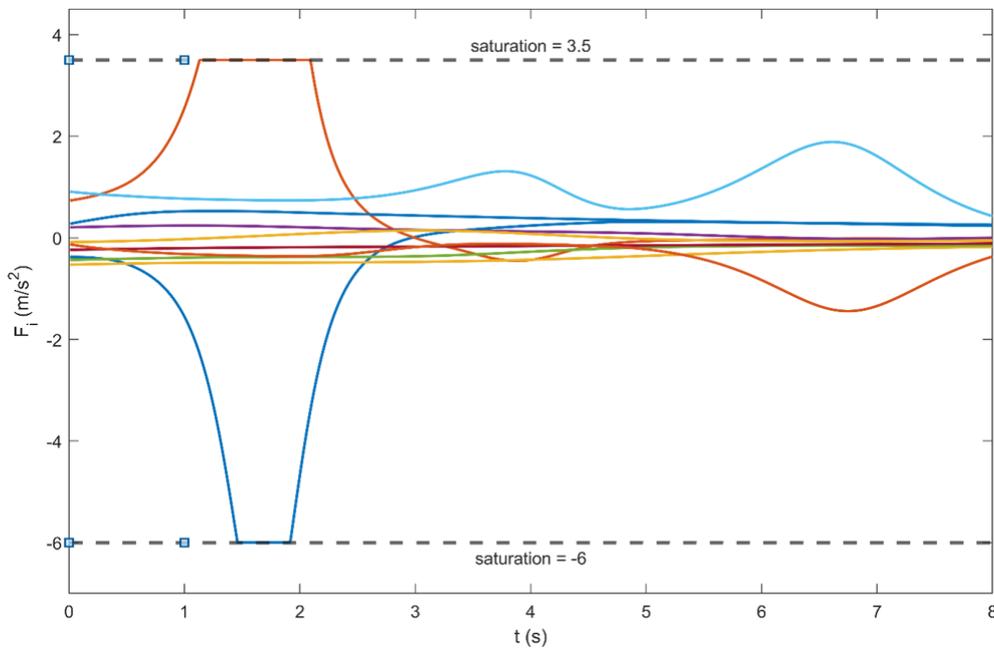


Figure 8.1: Accelerations adopting saturations when initial energy $H(0)=81.6431$.

At the same time, in Figures 8.2 and 8.3, we can see that no collision took place either with boundary of the road or among the vehicles, respectively. In Figure 8.4 we can see that there no issues occurred from the perspective of orientation, too.

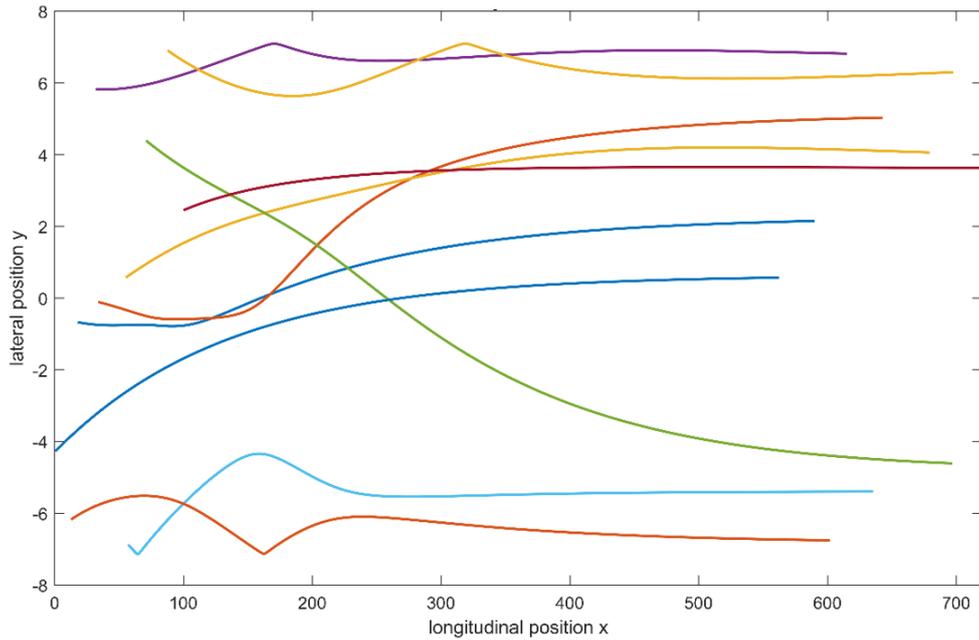


Figure 8.2: Trajectories with initial energy $H(0)=81.6431$.

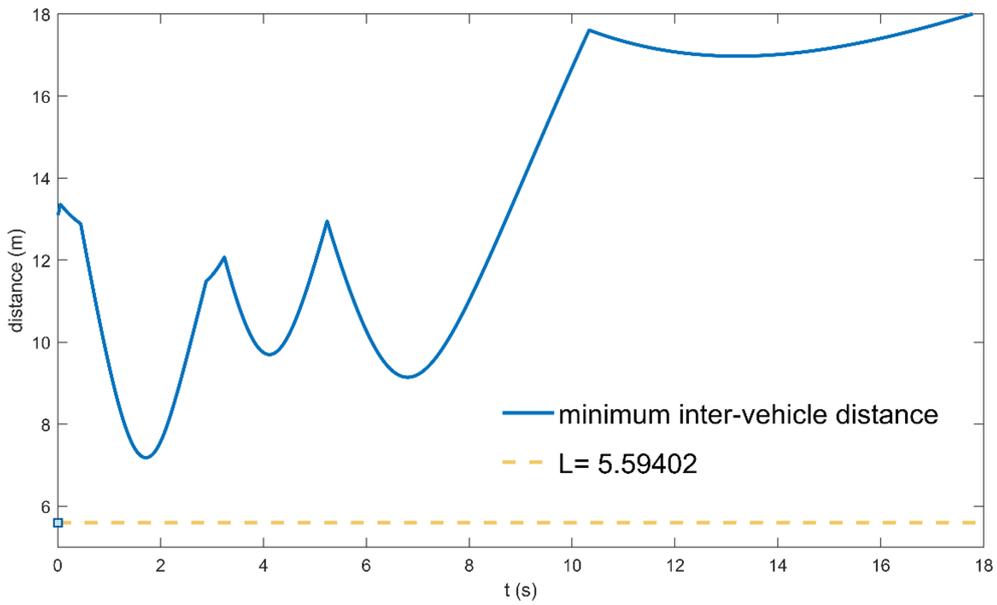


Figure 8.3: Minimum inter-vehicle distance with initial energy $H(0)=81.6431$.

However, in the next example, where the initial positions has changed and simultaneously the initial energy has increased, we can observe a crash. This phenomenon was expected since the system needed even higher acceleration values. In Figure 8.5 is observed that the controllers, obviously do not allow the vehicles to obtain accelerations higher or lower than the saturation limits. In this occasion collision avoidance did not succeed. In Figure 8.6 we can see that a crash between two vehicles took place in the 4th second.

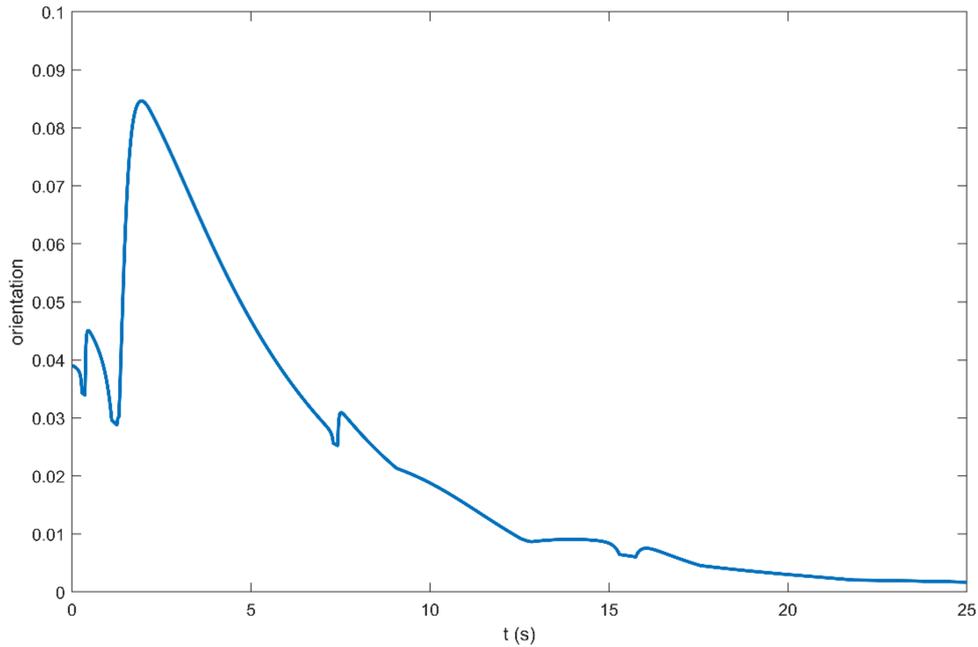


Figure 8.4: Max orientation $|\theta_i|$ with initial energy $H(0)=81.6431$.

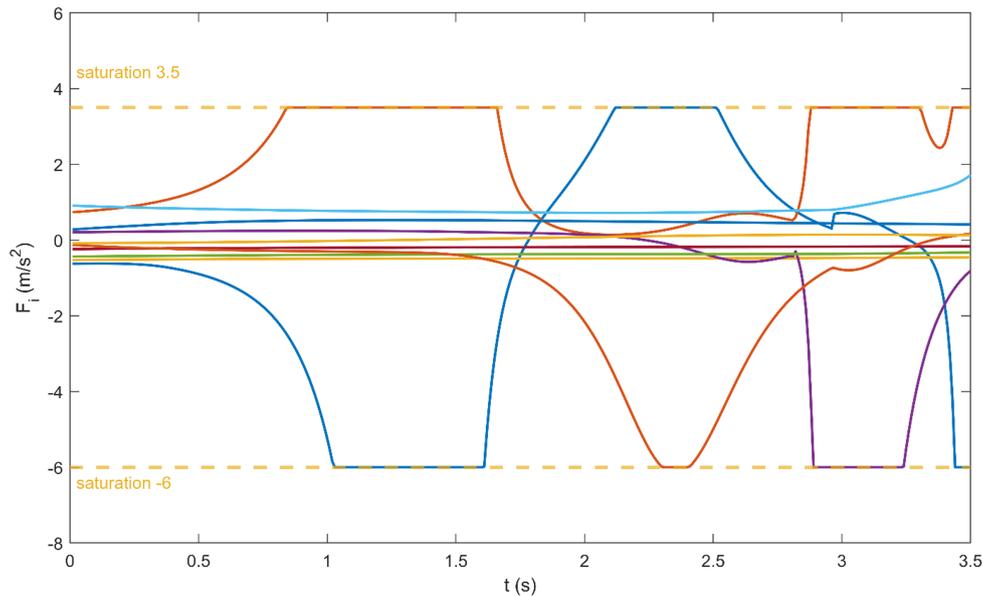


Figure 8.5: Accelerations adopting saturations when initial energy $H(0)=89.7691$.

This example makes more understandable that the existence of saturations sometimes can lead to collisions. Especially when the initial energy obtains high values. This means that vehicles are initialized in a collision course, with very small distances between them or with the boundary of the road. In practice, since the controllers already guarantee that such case will not occur, it is expected that there would be no collisions. It should be noticed that in the next Chapter we have included an algorithm that allow us to obtain realistic initial conditions, which is not the case in this example. In Figures 8.7a and 8.7b, we can see the behaviour of our vehicles in terms of acceleration without saturations when initial energy equals to 81.6431 and 89.7691, respectively. In the second case the vehicles developed twice the values of acceleration, so it was evident that the systems had more possibilities for a crash when the saturations were adopted. The dotted lines show the saturations we used earlier in order to be realized better the differences in both cases.

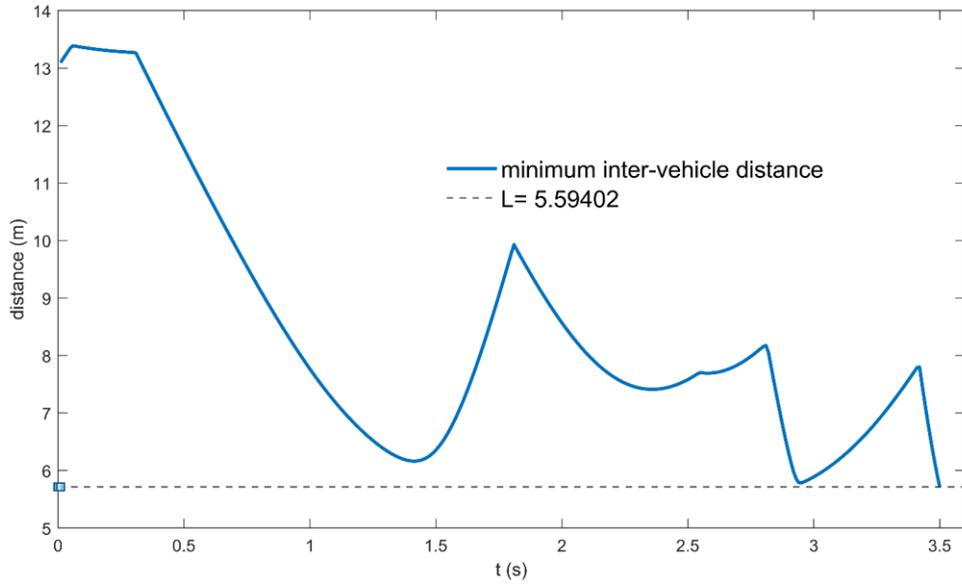
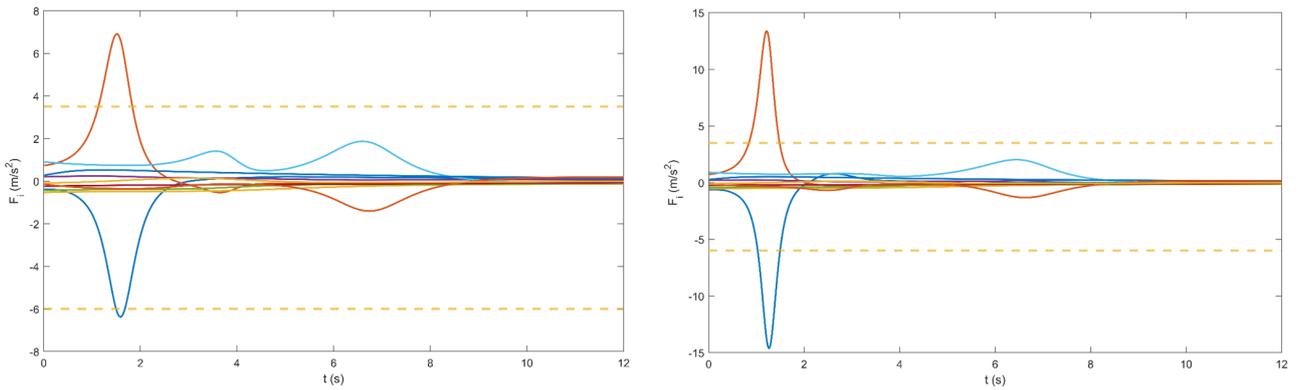


Figure 8.6: Minimum-inter vehicle distance when initial energy $H(0)=89.7691$.



(a) Accelerations without saturations when initial energy $H(0)=81.6431$.

(b) Accelerations without saturations when initial energy $H(0)=89.7691$.

Figure 8.7: Accelerations without saturations.

More specifically, if we examine the 8.1 and 8.2, we will understand that in both cases the initial positions of the vehicles did not change in terms of x, y and θ . Only the speed of the 1st vehicle increased from 33.0623 m/s to 35.0623 m/s. Meanwhile, the 2nd vehicle was only 15.93 m in front of the 1st vehicle with speed 25.7188 m/s. Consequently, the change in the 1st car in terms of speed played a significant role in the initial energy. This is the reason why the two vehicles obtained high accelerations that exceeded 10 m/s^2 . In simple words, the initial positions in the second occasion were unrealistic since the vehicles were forced to crash in the very next second.

Chapter 9

Initial positions generator

The initial conditions (position, speed, orientation) heavily influence the transient behavior of the system. As it has been observed in the previous Chapters, through the initial conditions we obtain the initial energy of the system which directly affects the acceleration of the vehicles, the MASP as well as the sampling period T . Besides that, we observed in the latter Chapter that sometimes the initial positions can be unrealistic, as vehicles may be initialized in a collision course which however will not happen in practice since the proposed controllers guarantee safety. For these reasons in this Chapter an algorithm is presented that generates initial positions with several features that are presented below. Initially the system generates arbitrary initial positions and next checks whether these features are satisfied. To simplify exposition, we divide the algorithm into three parts.

9.1 Examination if each vehicle is moving dangerously towards the boundary of the road.

In the event of a vehicle is moving towards the boundary of the road, the algorithm calculates the distance between the vehicle and the boundary. If this distance is smaller than $d_b m$ (e.g. $d_b = 2.5m$), the algorithm computes the lateral speed of the vehicle. If the lateral speed of the vehicle ($v_i \sin(\theta_i)$) is higher than $l_s m/s^2$ (e.g. $l_s = 1 m/s^2$), then the initial positions are rejected. This implies that the vehicle has direction towards the boundary of the road, with high lateral speed. The constants d_b and l_s are chosen from the operator. The flow chart in Figure 9.1 that describes the method.

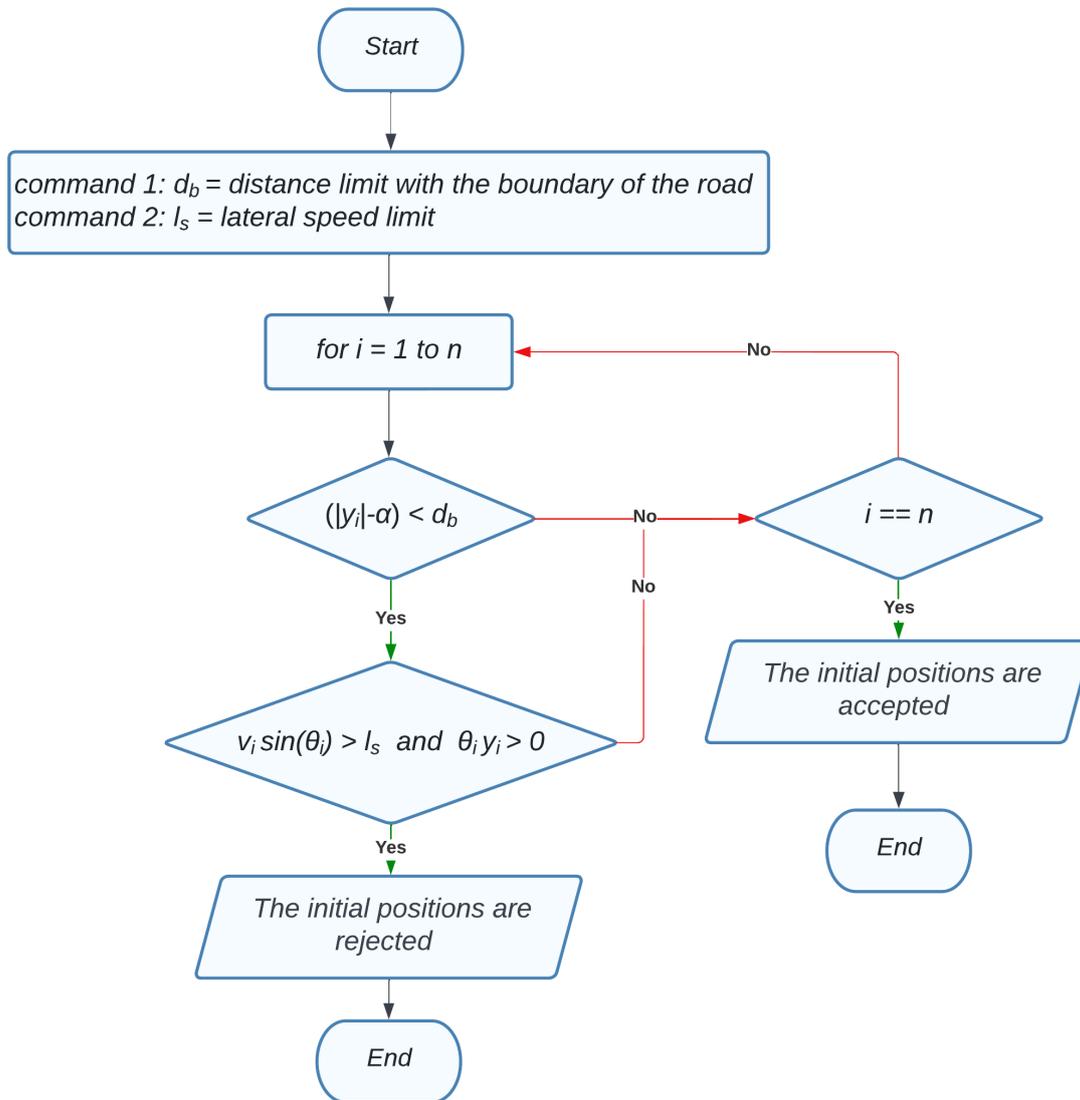


Figure 9.1: Examination if each vehicle is moving dangerously towards the boundary of the road.

9.2 Examination if each vehicle has in front of it or behind it, another vehicle at a dangerously close distance.

Consider two vehicles, one behind the other, and let the constants $S > 0$ and $T_h > 0$. The constant S indicates a safe distance when the vehicles are stationary, whereas the constant T_h constitutes the time-headway that concerns the time when the two vehicles cross the same point on the road. In addition, we assume V is the speed of the rear vehicle. When the distance between these two vehicles is smaller than the value $S + V * T_h$, the initial positions are rejected.

We can determine how safe the system is by appropriately choosing the constants S and T_h values. It must be mentioned that the method above should apply even when two vehicles are not exactly the one behind the other. Notably, a wider tolerance margin could offer a better drive experience since high accelerations or crashes will be avoided. To make it more understandable, we can observe Figure 9.2.

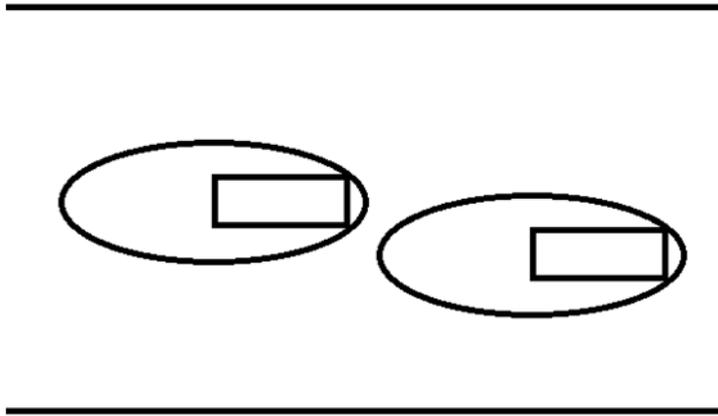


Figure 9.2: : The two vehicles are in close proximity and at the same time they are not one behind the other.

The distance between the vehicles is small enough and at the same time the two vehicles are not one behind the other. However, a possible overtaking due to nudging forces could lead to negative results since the vehicles will obtain extremely high accelerations. To avoid that, we adopt a constant *width* that demonstrates a lateral margin.

Finally, when observing Figure 9.3, it is easily understandable that each vehicle according to its speed, obtains a virtual safety parallelogram (taking into account the margin *width*) in which no other vehicle must be included.

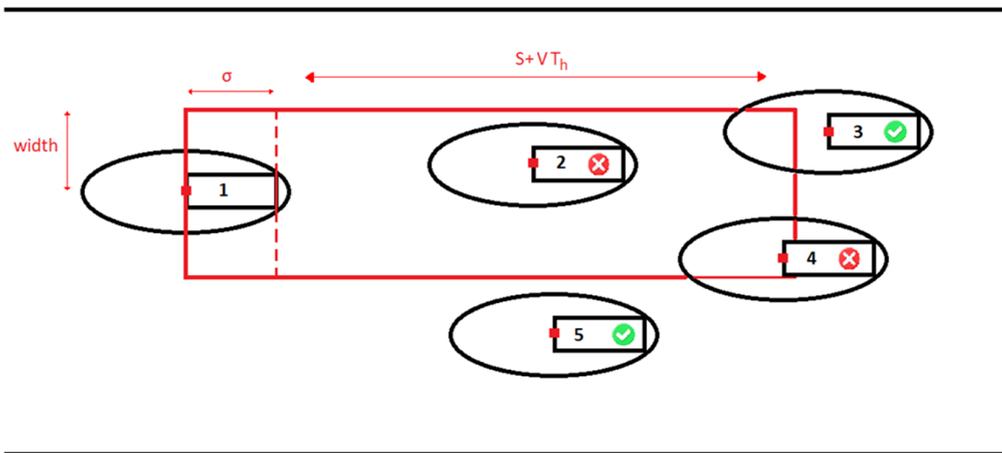


Figure 9.3: The safety parallelogram according to the value $S+VT_h$.

As we can see, the vehicle 1 has a virtual space that is designed according to its speed at each moment. Vehicles 2 and 4 are rejected since they are included in this space. The flow chart in Figure 9.4 describes the method of this algorithm.

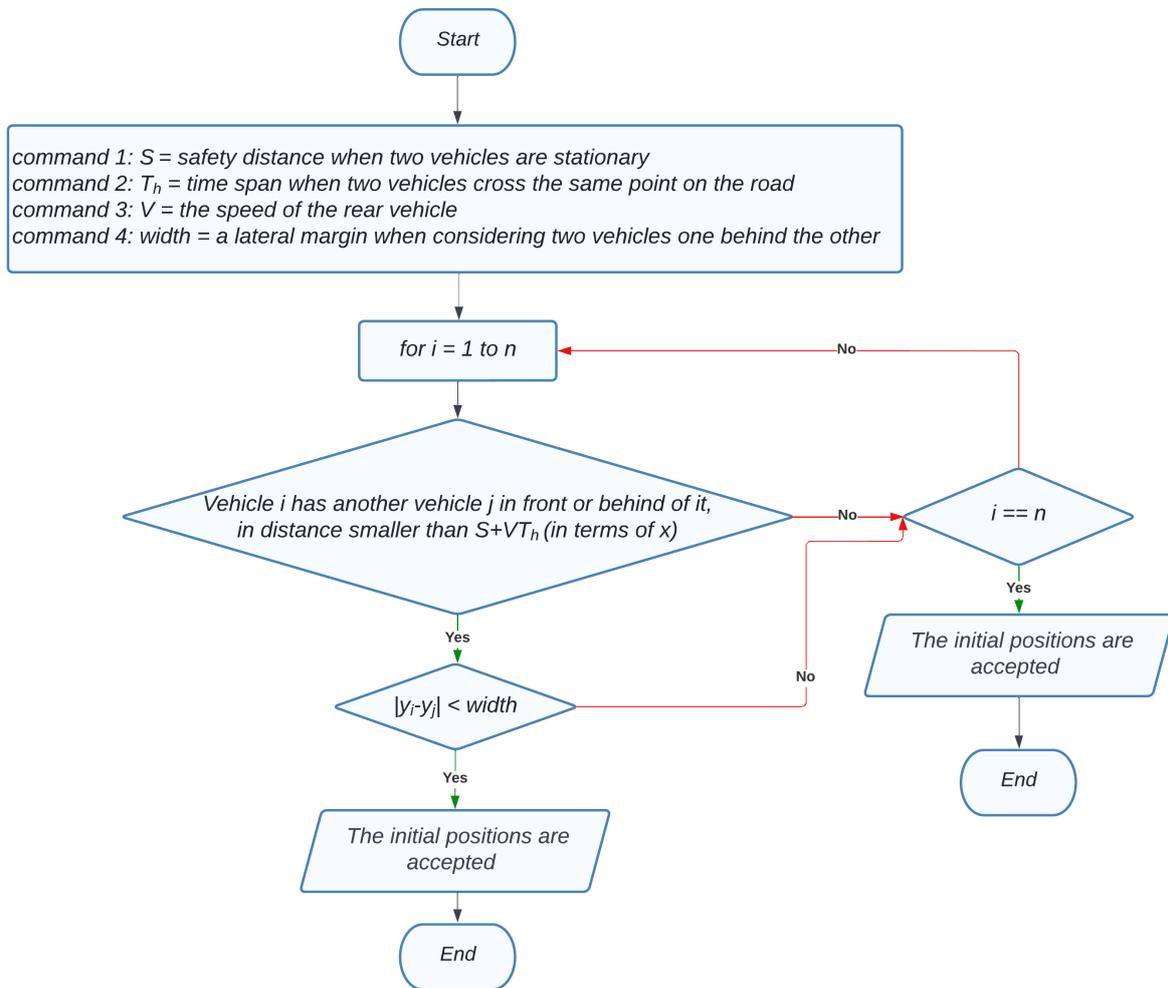


Figure 9.4: Examination whether each vehicle has in front of it or behind it, another vehicle at a dangerously close distance.

9.3 Examination if a vehicle has an adjacent vehicle with opposite orientation.

Sometimes, although all the conditions mentioned above are satisfied, the orientation of two vehicles can lead to negative effects. Let's observe Figure 9.5:

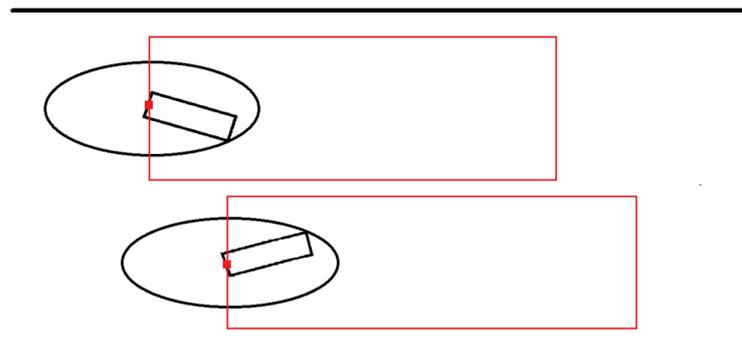


Figure 9.5: Two nearby vehicles with opposite orientation.

The vehicles neither approach the boundary of the road nor are included in any safety parallelogram. However, it seems that in the next seconds extremely high accelerations or crash are going to take place. To avoid that, we use the following methodology:

1. For each vehicle, it is checked if there is any adjacent vehicle. To characterize two vehicles “adjacent” we use the area bounded by the largest ellipse which indicates the area where the nudging effect occurs.
2. To inspect if a vehicle i is within the area bounded by the ellipse of a vehicle j we check if their distance is smaller than λ :

Then the algorithm checks whether two vehicles are adjacent. If they are, it is calculated:

1. The multiplication of the angles (in terms of θ) of the two vehicles. If this value is negative (especially, smaller than a parameter lm_{theta}), then we assume that the two vehicles have dangerously opposite orientation.
2. The lateral speeds of the two vehicles. If the relative speed of the two vehicles (absolute value of the sum of each lateral speed) is higher than a constant lm_v , then we assume that each vehicle dangerously approaches the other one.

If the values of the former calculations exceed the given limits then the initial positions of the vehicles are rejected. Let’s see Figure 9.6 to realize better how the ellipses work if the orientation and the lateral speeds exceed the limits.

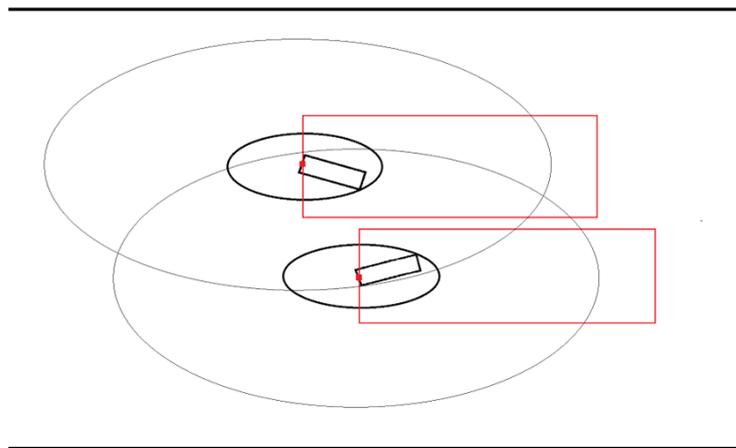


Figure 9.6: Two vehicles which are concerned adjacent according to their big ellipses.

We observe that one vehicle is within the area bounded from the ellipse of the other vehicle. So the vehicles are adjacent. Since the vehicles have the same length, it is obvious that each vehicle will be inside the other’s vehicle ellipse. Finally, it should be mentioned, that this specific condition is checked only if the first two conditions do not hold. For example, two vehicles may be the one behind the other (and simultaneously they may be adjacent). However, this condition has been checked earlier, so the algorithm will never check if the vehicles are adjacent. Following, we can see in Figure 9.7 a flow chart that describes the method.

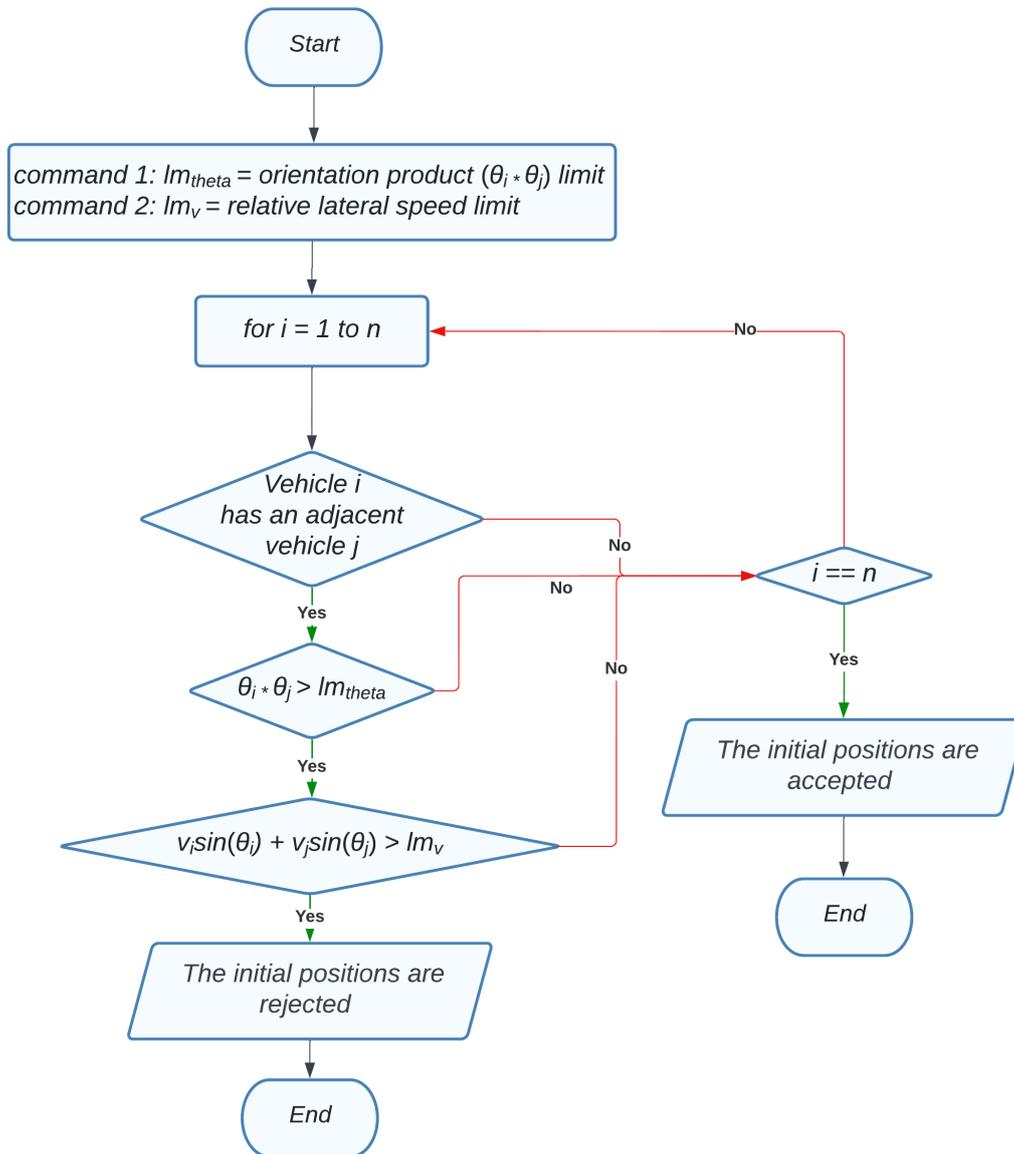


Figure 9.7: Examination whether a vehicle has an adjacent vehicle with opposite orientation.

As discussed before the initial positions generator must satisfy all the aforementioned conditions in order to create appropriate data. A flow chart which contains all the three parts was avoided since the comprehension of the information would be quite difficult and confusing.

Chapter 10

Conclusions

In this diploma thesis, we examined several issues from the perspectives of sampling-data controllers, sampling period, saturated inputs, and initial positions. Initially, we analyzed the continuous model provided by the bicycle kinematic model which we transformed into a discrete one using sampling data. We considered that the inputs were fixed between two sampling times during this transformation. Following this process, we found actual results for the Maximum Allowable Sampling Period per initial energy to select appropriate constant and periodical sampling periods. More specifically, we observed that while the initial energy related to the contextual initial positions increases and exceeds 40, the MASP has to take values that tend to zero. Since a constant sampling period throughout the simulations does not seem realistic, we found alternative ways to calculate the sampling period during the simulations; non-periodically and state-dependently according to the distance of each vehicle from the other vehicles and the boundary of the road. Firstly, we noticed that the execution time was significantly reduced by adopting a non-periodical sampling period (same for all the vehicles per sampling). However, when adopting a state-dependent sampling period, the execution time was reduced too but it did not take smaller values compared to the non-periodical sampling period. As discussed, this occurred because each vehicle has to calculate the temporary positions of the other vehicles while calculating the relative state since the samplings do not take place synchronously. At the same time, we examined the case in which the controllers take into account the speeds of the adjacent vehicles. We saw that when the vehicles measure the speeds of the neighboring vehicles, their trajectories seem smoother, and their maximum acceleration is retained in more natural values. Besides that, a faster convergence was observed. After extracting these results, we noticed that when the initial energy tends to obtain high values, the maximum acceleration of the vehicles sometimes develops unrealistic values. For these reasons, saturations were adopted in order to retain the accelerations in natural values. Evidently, if the difference between the maximum acceleration produced from the controllers and the adopted saturation is high, there exist possibilities for collisions. It should be mentioned that the former condition is importantly influenced by the initial energy of the system. Finally, in order to avoid unrealistic initial positions with extremely high values of acceleration, which probably lead the vehicles to a collision course, we created an algorithm that creates initial positions. These initial positions guarantee that the vehicles 1) will not go towards the road boundary dangerously, and 2) will not have a collision with each other at the very first seconds.

Appendix

Integration of continuous model (3.3)

We consider first the case $\omega_i = 0$.

Using (3.1) and since the input F_i is constant on the interval $[0, T]$, i.e., $F_i(t) = F_i(0) = F_i$, we get

$$\dot{v}_i = F_i \Rightarrow \int_0^T \dot{v}_i ds = \int_0^T F_i ds \Rightarrow v_i(T) = v_i(0) + TF_i \quad (\text{A1})$$

Since $\omega_i = 0$, it follows that $\dot{\theta}_i = 0$, and therefore, $\theta_i(t) = \theta_i(0)$ for $t \in [0, T]$.

Using again (3.1) model and (A1) we get

$$\dot{x}_i(s) = v_i(s) \cos(\theta_i(s)) \Rightarrow \int_0^T \dot{x}_i(s) dt = \int_0^T v_i(s) \cos(\theta_i(s)) dt \Rightarrow$$

$$\int_0^T \dot{x}_i(s) ds = \int_0^T (v_i(0) + sF_i) \cos(\theta_i(s)) ds$$

The latter implies that

$$x_i(T) = x_i(0) + Tv_i(0) \cos(\theta_i(0)) + \frac{T^2}{2} F_i \cos(\theta_i(0))$$

With similar arguments, we finally get

$$\dot{y}_i = v_i \sin(\theta_i) \Rightarrow \int_0^T \dot{y}_i ds = \int_0^T v_i \sin(\theta_i) ds \Rightarrow$$

$$\int_0^T \dot{y}_i ds = \int_0^T (v_i(0) + sF_i) \sin(\theta_i) ds \Rightarrow y_i(T) = y_i(0) + Tv_i(0) \sin(\theta_i(0)) + \frac{T^2}{2} F_i \sin(\theta_i(0))$$

We consider now the case $\omega_i \neq 0$

$$v_i(T) = v_i(0) + TF_i \quad \text{by (A1)}$$

Since $\omega_i \neq 0$, it follows that $\dot{\theta}_i \neq 0$,

In addition we assume that ω_i is constant on the time interval $[0, T]$ i.e., $\omega_i(t) = \omega_i(0) = \omega_i$.

Since $\omega_i = \sigma^{-1} \tan(\delta_i)$, we get that δ_i is constant on the time interval $[0, T]$, too. Namely, $\delta_i(t) = \delta_i(0) = \delta_i$.

Using (3.1) model and (A1) we get

$$\begin{aligned}\dot{\theta}_i(s) &= \sigma^{-1}(v_i(s) \tan(\delta_i)) \Rightarrow \\ \int_0^T \dot{\theta}_i(s) ds &= \int_0^T \sigma^{-1}(v_i(s) \tan(\delta_i)) ds \Rightarrow \\ \int_0^T \dot{\theta}_i(s) ds &= \int_0^T \sigma^{-1}((v_i(0) + sF_i) \tan(\delta_i)) ds \Rightarrow \\ \int_0^T \dot{\theta}_i(s) ds &= \sigma^{-1} \int_0^T v_i(0) \tan(\delta_i) + sF_i \tan(\delta_i) ds \Rightarrow\end{aligned}$$

The latter implies that

$$\theta_i(T) = \theta_i(0) + \sigma^{-1} \left(T v_i(0) \tan(\delta_i) + \frac{T^2}{2} F_i \tan(\delta_i) \right)$$

Using again (3.1) model and (A1) we get

$$\begin{aligned}\dot{x}_i(s) &= v_i(s) \cos(\theta_i(s)) \Rightarrow \\ \int_0^T \dot{x}_i(s) ds &= \int_0^T v_i(s) \cos(\theta_i(s)) ds \Rightarrow \\ \int_0^T \dot{x}_i(s) ds &= \int_0^T (v_i(0) + sF_i) \cos \left(\theta_i(0) + \sigma^{-1} \left(s v_i(0) \tan(\delta_i) + \frac{s^2}{2} F_i \tan(\delta_i) \right) \right) ds \Rightarrow \\ \int_0^T \dot{x}_i(s) ds &= \frac{\sigma}{\tan(\delta_i)} \int_0^T \frac{\tan(\delta_i)}{\sigma} (v_i(0) + sF_i) \cos \left(\theta_i(0) + \sigma^{-1} \left(s v_i(0) \tan(\delta_i) + \frac{s^2}{2} F_i \tan(\delta_i) \right) \right) ds\end{aligned}$$

Which implies that

$$x_i(T) = x_i(0) + \frac{1}{\omega_i} \left(\sin \left(\theta_i(0) + \omega_i \left(v_i(0) + \frac{T}{2} F_i \right) T \right) - \sin(\theta_i(0)) \right)$$

Following the same process using (3.1) model and (A1), we get

$$y_i(T) = y_i(0) + \frac{1}{\omega_i} \left(\cos(\theta_i(0)) - \cos \left(\theta_i(0) + \omega_i \left(v_i(0) + \frac{T}{2} F_i \right) T \right) \right)$$

This completes the integration procedure.

Scenario 3 | An example

Let us assume an arbitrary vehicle i . Using,

1. the ODEs of model (3.1),
2. the sampling time of the vehicle i ,
3. the latter sampling times of the other vehicles
4. the values x, y, θ , of the other vehicles at their latter sampling time,

the vehicle i can estimate the temporary positions of the other vehicles in order to calculate its sampling period.

Firstly, all the vehicles have to initialize their values related to sampling period. Let us assume:

1. $T_last(i)$ = the time when the last sampling of the vehicle i took place.
2. $T(i)$ = the exact sampling period of the vehicle i .
3. $T_next(i)$ = the time when the next sampling of the vehicle i takes place.
4. $TIME$ = the time we are in each loop.

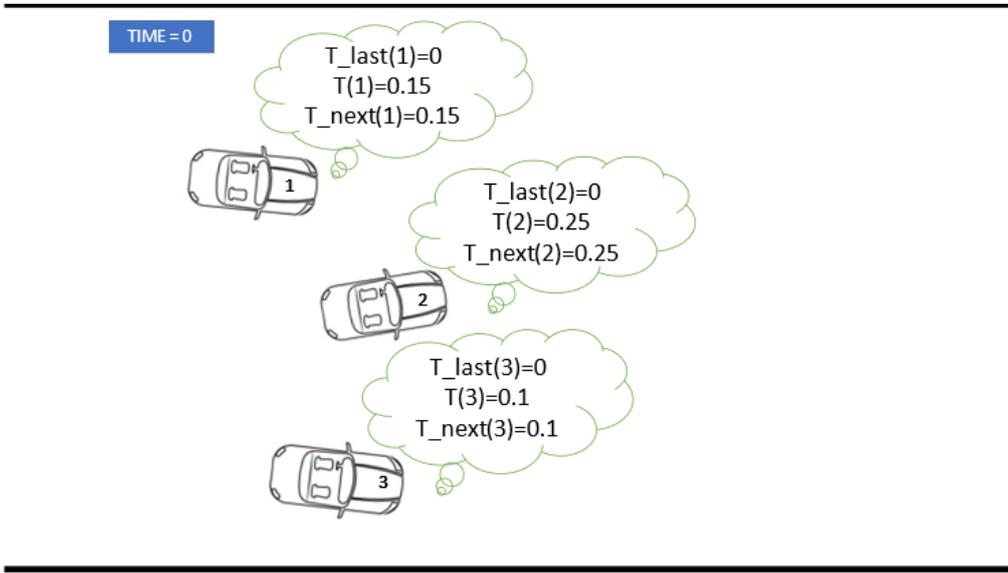


Figure 1: Vehicles' initial values related sampling period.

At the first loop in Figure 1, it is obvious that the variables T_next and T are the same for all the vehicles since the variable $TIME=0$. Only for the first loop applies that $T_next=T+TIME$ for all the vehicles. Now, the system has to find the vehicle which is going to be sampled sooner. Namely, it has to find the smallest value of the vector T_next and the vehicle which possesses this value. As we can see, it is the 3rd vehicle with $T_next=0.1$. At this time, observing Figure 2, the system changes the variable $TIME$ to 0.1 and starts the sampling process of the 3rd vehicle. As the other vehicles have not been sampled yet, the estimation of their position will take place via the following Euler's formula:

$$x_{temporary}(j) = x_j + v_j \cos(\theta_j)(TIME - T_{last})$$

$$y_{temporary}(j) = y_j + v_j \sin(\theta_j)(TIME - T_{last})$$

Actually, this formula considering known the position of the other vehicle at its latter sampling, estimates its temporary position using the ODEs of model (3.1). Particularly, it calculates the difference between the variables TIME and T_{last} and multiplies this difference with the relative ODE of model (3.1). The product of the multiplication is added to the last known position. The previous process is done *for* $j = 1, \dots, n, \quad i \neq j$.

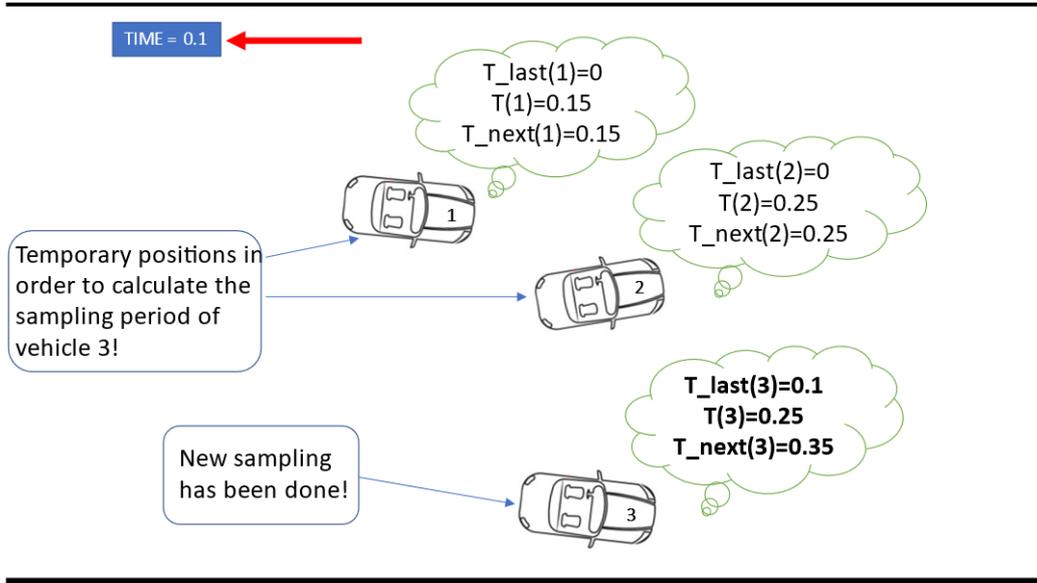


Figure 2: Autonomous sampling.

NOTE! The vehicles 1 and 2 have not been sampled yet! Their temporary positions have been estimated just to be calculated the sampling period of the 3rd vehicle. Besides that, the vehicle 3 needed the positions of the other vehicles to compute the nudging forces that are exerted to it. Now the system has to find again the vehicle which is going to be sampled sooner. It is the 1st vehicle with $T_{next}=0.15$. The system will do the same process as before. One noteworthy feature at the 2nd loop is that the variable T_{last} is not equal to zero as the vehicle 3 has been sampled.

Bibliography

- [1] K. Chavoshi and A. Kouvelas, “Cooperative distributed control for lane-less and direction-less movement of autonomous vehicles on highway networks,” *9th Symposium of the European Association for Research in Transportation (hEART 2020)*, 2020.
- [2] C. Clapham, J. Nicholson, and J. R. Nicholson, “The concise oxford dictionary of mathematics,” *Oxford University Press*, 2014.
- [3] R. Delpiano, J. C. Herrera, J. Laval, and J. E. Coeymans, “A two-dimensional car-following model for two-dimensional traffic flow problems,” *Transportation Research Part C: Emerging Technologies*, vol. 114, pp. 504–516, 2020.
- [4] C. Diakaki, M. Papageorgiou, I. Papamichail, and I. Nikolos, “Overview and analysis of vehicle automation and communication systems from a motorway traffic management perspective,” *Transportation Research Part A: Policy and Practice*, vol. 75, pp. 147–165, 2015.
- [5] L. Grüne and J. Pannek, “Discrete time and sampled data systems,” *Nonlinear Model Predictive Control*, pp. 13–43, 2017.
- [6] T. Hu and Z. Lin, “Control systems with actuator saturation: analysis and design,” *Springer Science & Business Media*, 2001.
- [7] L. Iftekhar and R. Olfati-Saber, “Autonomous driving for vehicular networks with nonlinear dynamics,” *IEEE Intelligent Vehicles Symposium*, pp. 723–729, 2012.
- [8] P. A. Ioannou and C.-C. Chien, “Autonomous intelligent cruise control,” *IEEE Transactions on Vehicular technology*, vol. 42, no. 4, pp. 657–672, 1993.
- [9] I. Karafyllis, D. Theodosis, and M. Papageorgiou, “M.: Lyapunov-based two-dimensional cruise control of autonomous vehicles on lane-free roads,” *Automatica*, *accepted*, doi = 10.48550/arxiv.2103.12205, 2021.
- [10] —, “Nonlinear adaptive cruise control of vehicular platoons,” *International Journal of Control*, pp. 1–23, 2021.
- [11] —, “Analysis and control of a non-local pde traffic flow model,” *International Journal of Control*, vol. 95, no. 3, pp. 660–678, 2022.
- [12] Y. Liu and B. Xu, “Improved protocols and stability analysis for multivehicle cooperative autonomous systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2700–2710, 2015.
- [13] D. Nesic, A. R. Teel, and D. Carnevale, “Explicit computation of the sampling period in emulation of controllers for nonlinear sampled-data systems,” *IEEE transactions on Automatic Control*, vol. 54, no. 3, pp. 619–624, 2009.

- [14] M. Papageorgiou, K.-S. Mountakis, I. Karafyllis, I. Papamichail, and Y. Wang, “Lane-free artificial-fluid concept for vehicular traffic,” *Proceedings of the IEEE*, vol. 109, no. 2, pp. 114–121, 2021.
- [15] M. Rahman, M. Chowdhury, Y. Xie, and Y. He, “Review of microscopic lane-changing models and future research opportunities,” *IEEE transactions on intelligent transportation systems*, vol. 14, no. 4, pp. 1942–1956, 2013.
- [16] S. Taghvaeeyan and R. Rajamani, “Portable roadside sensors for vehicle counting, classification, and speed measurement,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 73–83, 2013.
- [17] D. Theodosis, F. N. Tzortzoglou, I. Karafyllis, I. Papamichail, and M. Papageorgiou, “Sampled-data controllers for autonomous vehicles on lane-free roads,” *Mediterranean Conference on Control and Automation (MED 2022)*, pp. 103–108, 2022.
- [18] P. Tientrakool, Y.-C. Ho, and N. F. Maxemchuk, “Highway capacity benefits from using vehicle-to-vehicle communication and sensors for collision avoidance,” *IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1–5, 2011.
- [19] M. Velasco, J. Fuertes, and P. Marti, “The self triggered task model for real-time control systems,” *Work-in-Progress Session of the 24th IEEE Real-Time Systems Symposium (RTSS03)*, vol. 384, pp. 67–70, 2003.
- [20] Z. Wang, G. Wu, and M. J. Barth, “A review on cooperative adaptive cruise control (cacc) systems: Architectures, controls, and applications,” *21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2884–2891, 2018.
- [21] P. Wieland and F. Allgöwer, “Constructive safety using control barrier functions,” *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007.
- [22] X. Xu, J. W. Grizzle, P. Tabuada, and A. D. Ames, “Correctness guarantees for the composition of lane keeping and adaptive cruise control,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1216–1229, 2017.
- [23] Z. Zheng, “Recent developments and research needs in modeling lane changing,” *Transportation research part B: methodological*, vol. 60, pp. 16–32, 2014.