Stability analysis for controller switching in autonomous vehicles

Francisco Navas Matos

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à MINES ParisTech

Analyse de stabilité pour la reconfiguration de contrôleurs dans de véhicules autonomes

Soutenue par

Francisco NAVAS

Le 28 Novembre 2018

Ecole doctorale n° 432

Sciences des Métiers de l'Ingénieur (SMI)

Spécialité

Mathématiques, Informatique temps-réel, robotique

Composition du jury :

Michel, BASSET Univ. d'Haute Alsace	*Président*
Mariana, NETTO IFFSTAR	*Rapporteur*
Youcef, MEZOUAR Univ. Clermont Auvergne	*Rapporteur*
Lydie, NOUVELIERE Univ, d'Evry	*Examinateur*
Jorge, VILLAGRÁ CSIC (Espagne)	*Examinateur*
Fawzi, NASHASHIBI INRIA	*Directeur de thèse*
Vicente, MILANÉS Renault	*Encadrant de thèse*

MINES ParisTech | PSL

Acknowledgments

I would like to express my sincere gratitude to my supervisors Vicente Milanés and Fawzi Nashashibi for the continuous support of my Ph.D. thesis and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having better supervisors for my Ph.D. thesis.

I thank my fellow labmates in the Robotics and Intelligent Transportation Systems team at INRIA for the stimulating discussions, the help during experimental tests, and for all the fun we have had in the last three years.

Last but not the least, I must express my gratitude to my familiy and friends, Ales, Cris, David, Jared, Jorge, Julia, Pietro, Raoul and Raquel for their continued support and encouragement, and for their patience experiencing all the ups and downs of my research. I thank Pietro for his willingness to read countless pages of meaningless equations.

Contents

Chapter 1

Introduction

Autonomous vehicles are getting more and more attention because of their potential to both significantly reduce the number of road fatalities and improve drivers' daily lives. Driverless cars research field has been very active in recent years, and significant advances have been achieved. However, there are still some significant gaps before having fully automated vehicles on public roads.

The research on the last years has been focused on the development of multi-sensor systems able to perceive the environment in which the vehicle is driving in, permitting to create a comprehensive map of the traffic situation. These multi-sensor perception systems are significantly increasing the complexity when it comes to autonomously control the vehicle. Different control systems are activated according to a multi-target decision making system. Each of these systems follows performance and stability criteria for its design, but they all have to work together, providing stability guarantees and being able to handle unexpected situations as unpredicted uncertainties or even fully outages from sensors. With these premises, the goal of this Ph.D. work is to further investigate intelligent advanced control systems to provide stable responses for autonomous vehicles under different circumstances.

This thesis has been developed within the Robotics and Intelligent Transportation Systems (RITS) team/project at the French National Institute for Research in Computer Science and Control (INRIA, from french: Institut National de Recherche en Informatique et en Automatique). In the following, the author explains motivation, objectives, manuscript organization and main contributions of the presented work.

1.1 Motivation

Autonomous driving aims to improve traffic flow, reduce accidents and fuel consumption, and make possible personal car travel for everyone regardless of their abilities or conditions. An autonomous vehicle is built by combining a set-of-sensors and actuators together with sophisticated algorithms. These algorithms perform different functions, taking the information coming from the sensors to make the vehicle react to different traffic situations through the actuators [Luettel et al., 2012]. A general architecture for autonomous driving is in Fig. 1.1. Details about each of the blocks that form this autonomous vehicle's architecture are found below:

- Acquisition. It is the process on charge of getting information from the installed sensors in the vehicle. Global Position Systems (GPS), Inertial Measurement Unit (IMU) and odometry are used for vehicle location in a coordinate framework; Light Detection And Ranging

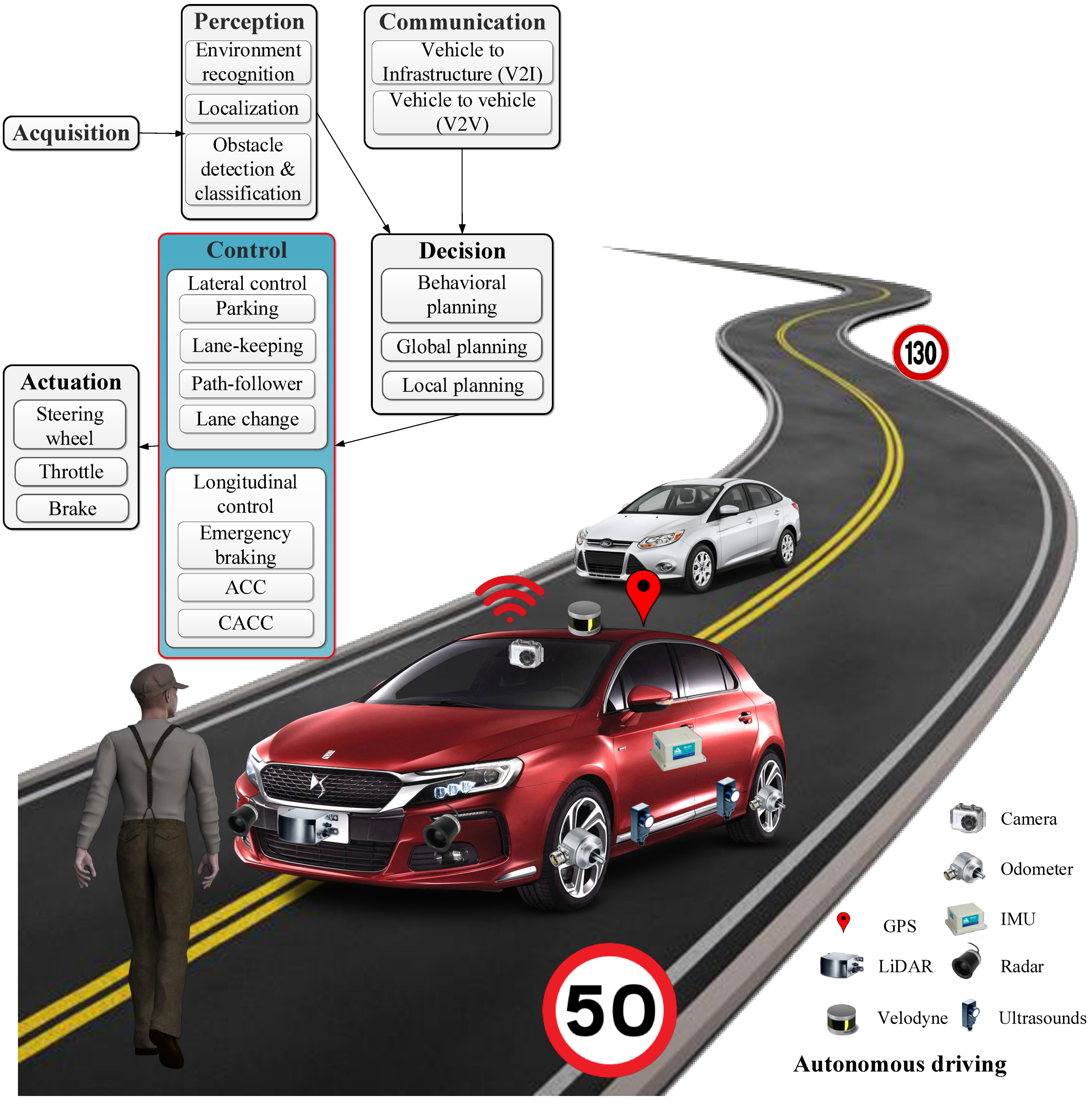


Figure 1.1: Autonomous vehicle architecture in changing environment.

(LiDAR) sensors, radar, ultrasounds, and cameras are employed for having a 360º view of the environment.

- Communication. Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) wireless communications are used to be able to communicate with other vehicles and road infrastructure.
- Perception. This block uses the information coming from the acquisition stage, in order to understand and model the environment around the vehicle, being aware of its state in such environment. Obstacles detection in the surroundings (pedestrians, vehicles...), ego-vehicle's localization and detection of lane marks on the road, are some of the tasks linked to this block.
- Decision. It manages the data processed in the perception stage for a dynamic behavior of the vehicle. It is able to react and interact with unexpected situations that usually affect the predefined driving such as: pedestrians, road works, obstacles, Human-Machine-Interface (HMI) request, etc.
- Control. It is responsible for the reference path tracking and driving safety requirements provided by the decision stage. Control variables as steering angle and longitudinal velocity are obtained in order to correctly follow orders given by the decision stage.
- Actuation. The control output is sent to the different vehicle actuators: The steering wheel for the lateral control; and throttle, brake and gear shift for longitudinal control.

This PhD work is focused in the control block design. According to different scenarios (i.e. road layout, other traffic agents interaction), different control systems are required. Activation is commanded by the decision system based on the information provided by the perception block. Control systems can be divided in classical, optimal, robust, adaptive and Fault Tolerant Control (FTC):

- Classical control is based on the use of linear differential equations describing system dynamics. Control mission is to make the error between reference input and feedback sensor state zero.
- Optimal control, on the other hand, is an extension of classical control in which you answer the question: How do I design my controller to ensure that I optimize a performance index? It assumes a perfect model of the system.
- Robust control, on the contrary, assumes that the model is imperfect, seeking for stability and quality of the controller given external disturbances or uncertainty in the system model.
- Adaptive control is required in scenarios where large changes occur. Controller parameters change with time and tracks the changes in the plant, with the goal of designing a system which, at all instants, performs in accordance with the design constraints.
- FTC aims to increase plant availability and reduce the risk of safety hazards. Its goal is to prevent that simple faults develop into serious failure.

Vehicle dynamic control can be divided in lateral and longitudinal controllers. The former allows to automatically steer according to a planned trajectory. The latter acts on the throttle and brake for folllowing a reference speed, playing a key role to ensure safety and comfort of passengers.

Specifically related to automated longitudinal control, Cruise control and platooning tasks are mainly developed. Cruise control permits to set a maximum speed at which someone desires to travel, acting over throttle and break pedals in order to maintain the speed of the vehicle even on up and down hills. By adding a forward radar, vehicle gains environmental information – intervehicle distance – to adapt its velocity according to the preceding's one. This is called Adaptive Cruise Control (ACC). ACC reduces congestion in highways by making formation of vehicles. V2V communication is added to the existing ACC system, improving traffic flow through the formation of a tighter string of vehicles– so-called Cooperative Adaptive Cruise Control (CACC). While ACC/CACC are comfort systems to help the driver and reduce traffic congestion problem, they do not have a way to prevent a crash with a forward vehicle or pedestrian. These kind of systems are not full range, and another controller will be needed if preceding vehicle brakes suddenly or a pedestrian passes between two vehicles. Emergency brake control needs to be developed. Once both cruise and emergency brake control are designed, an optimization process could be carried out in order to make maneuvers optimal. This involves completing maneuvers, such as join, split or change lane in the minimum possible time, while maintaining as high a speed and as small a distance from the preceding vehicle as practicable and safe. Optimal acceleration/deceleration could be also treated in the sense of minimizing fuel consumption. All these solutions together significantly enhance road safety and improve highway utility. However, various uncertainties and disturbances present in the real world should be considered to have not only a optimal solution, but robust. Uncertainties or external disturbances are for example dynamics of different vehicles, variant delay communication between vehicles, wind gust or road slopes. When dynamics difference are large, and adaptive control with identification process would be needed. Finally, FTC is also employed in order to deal with communication link availability or in-wheel motor faults in electrical vehicles, among others. This shows the need to employ different control systems in order to have a complete solution dealing with different traffic situations, dynamics of ego and surrounding vehicles, sensor/actuators availability, pedestrians or even driver preferences.

Lateral control is in charge to turn the steering wheel for applying path corrections to reduce or remove errors between actual and intended paths. The intended path can change in order to avoid obstacles or pedestrians. It is clear that GPS, camara, odometry and LiDAR is needed in order to localize the vehicle with respect to the intended path. A classical solution should be able to navigate soft turns and straight lines at specific velocity. It is logical that the steering wheel of our car does not turn in the same way to take a curve, if this is done at $10km/h$ or $100km/h$. Different controllers would be needed depending on the longitudinal velocity of the vehicle. At the same time, those controllers could be modified to cover a wide range of roads/situations, including sharp turns, roundabouts, lane change and so on. Adaptive control could offer a good adaptability to different road types and velocities. Robustness is also important in lateral control. Uncertainties are mainly in parameters as tire cornering stiffness, vehicle longitudinal velocity and yaw rate. A typical external disturbance is the road surface friction condition, which is uncertain and can change extremely quickly. It is not the same to drive in icy than in rough asphalt roads. To improve control, the road surface friction can be treated as a robust solution for a specific range, or even as an adaptive solution if an estimation of this parameter exists. Finally, FTC solutions would be also important to ensure accurate path tracking in the presence of faults. Faults could go from sensors fails to wheel lock. Consequently, lateral control needs different control solutions depending on the road form and situation, vehicle dynamics and sensor/actuators availability. Driver preferences could be also important in order to adapt driving style.

In short, both longitudinal and lateral control solutions have many different solutions depending on the nature of the problem, and a control/supervision structure would be necessary to deal with all the types of changes that may come. In this thesis, Youla-Kucera (YK) parameterization is analysed as a methodology that could improve the security of autonomous driving systems by providing a framework managing different sensor/actuator setups, dynamics and traffic situations with stability guarantees.

1.2 Objectives

The objective of this Ph.D. thesis is to further investigate the YK parameterization to provide stable responses for autonomous vehicles when dynamics or environmental changes occur. This thesis explores the use of the YK parameterization in dynamics systems such as vehicles, with special emphasis on stability when some dynamic change or the traffic situation demands controller reconfiguration.

In order to meet with the idea of general control framework handling those changes into the vehicle, different steps should be followed: First, controller reconfiguration due to different traffic situations is explored. Then, dynamics of ego or surrounding vehicles could be important in order to improve vehicle performance/stability. Thus, identification of unmodeled vehicle dynamics is analysed. Finally, both controller reconfiguration and dynamics identification should be used together following some performance/stability criteria.

Focus is in obtaining simulation and experimental results related to the use of the YK parameterization in the longitudinal control of an autonomous vehicle. CACC applications are targetted, with the aim not only of using for the very first time YK parameterization in the Intelligent Transportation Systems (ITS) domain, but improving CACC state-of-the art by providing stable controller reconfiguration results when non-available communication link with the preceding vehicle, cut-in/out maneuvers or surrounding vehicles with different dynamics.

With the present results, the author aims to prove adaptability, stability and real implementation of the YK parameterization as control framework for secure responses in autonomous driving.

1.3 Manuscript organization

The present Ph.D. work is organized in a total of six chapters. An overview of remaining chapters is given below:

Chapter 2. State of the art. This chapter presents a review of the YK parameterization related to classical, optimal, adaptive, robust and FTC. The origins of this mathematical framework are explained. Important groups worldwide are reviewed, focusing on the different types of control applications developed, allowing the understanding of open challenges and future research work.

Chapter 3. Youla-Jabr-Bongiorno-Kucera parameterization. YK parameterization provides all stabilizing controllers for a given plant, and this is used for performing stable controller reconfiguration. YK mathematical basis is provided with emphasis in stability proof. Different control structures for stable switching are derived from this parameterization, dealing with problems such order complexity, plant disconnection or matrix inversability. Different numerical examples are given for the better understading of the stable controller reconfiguration and transient behavior depending on the chosen YK-based control structure.

Chapter 4. Dual Youla-Jabr-Bongiorno-Kucera parameterization. Dual YK parameterization provides all plants stabilized by a given controller, and this is used to perform controller design in the presence of system variations or Closed-Loop (CL) identification. The basis of this parameterization is also explained. CL stabilization in the presence of system variations is analysed. The dual YK parameterization properties are used for obtaining a Multi Model Adaptive Control (MMAC) approach, and CL identification algorithms.

Chapter 5. Applications. This chapter explores the uses of YK and dual YK parameterization in autonomous driving; specifically, CACC applications are considered in the presence of traffic or dynamics changes. YK-based stable controller reconfiguration is used to deal with the problem of non-available communication link with the preceding vehicle; and vehicles joining/leaving the string. Then, as vehicles in the string could be different, dual YK parameterization is employed to perform CL longitudinal dynamics identification. Finally, both YK and dual YK parameterization are used in a MMAC approach to deal with vehicles heterogeneity in CACC string of vehicles. Simulation and experimental results with different type of controllers and structures prove adaptability, stability and real implementation of the YK parameterization.

Chapter 6. Conclusions. Conclusions and most important remarks, with respect to the problems addressed in the present Ph.D. work, are given in this chapter. Also future research lines are presented and discussed.

1.4 Contributions

In the present dissertation, YK parameterization is used as control framework able to deal with controller reconfiguration, dynamics identification and adaptive control approaches. Contributions are detailed below:

1. YK parameterization provides all stabilizing controllers for a given plant. This is used in order to perform stable controller reconfiguration. Different YK-based control structures are obtained for dealing with problems such order complexity, plant disconnection or matrix inversability. Stability properties are preserved even if different structures are employed, but transient behavior between controllers changes depending on the employed YK-based structure. One of the structures presents the best transient behavior without oscillations, a lower order controller complexity and no need to disconnect the initial controller, which would be important if the system shutdown is very expensive, or the initial controller is part of a safety circuit. This structure is used together with CACC applications improving CACC state-of-the-art. A hybrid behavior between two CACC controllers with different time gaps is explored by means of the YK parameterization, in order to avoid ACC degradation when communication link with preceding vehicle is lost. The proposed system uses YK parameterization and communication with a vehicle ahead (different from the preceding one) providing stable responses and, more interestingly, reducing intervehicle distances in comparison with an ACC degradation. A similar idea of hybrid behavior between CACC controller with different time gap is developed for entering/exiting vehicles in the string. In that case, YK parameterization is able to ensure stability of these merging/splitting maneuvers.

2. Dual YK parameterization provides all the plants stabilized by a controller. This is employed for solving CL identification problems, or adaptive control solutions, which integrate identification and controller reconfiguration processes. YK-based CL identification uses classical OL identification algorithms, providing better results than if it is used alone. Results in a CACC-equipped vehicle prove how CL nature of the data affects a classical OL identification algorithm, and how dual YK parameterization helps to mitigate these effects. Finally, an adaptive control application is developed by using MMAC. Longitudinal dynamics of two vehicles in a CACC string are estimated within a model set, so the good CACC sytem can be chosen even if a heterogeneous string of vehicles is considered. Dynamics estimation results much faster than other estimation processes in the literature.

3. Different type of controllers and structures are used throughout this thesis, proving the

adaptability of the YK parameterization to any type of controller. Simulation and experimental results demonstrate real implementation of stable controller reconfiguration, CL identification and adaptive control solutions dealing with dynamics changes or different traffic situations. The author thinks that YK is a suitable control framework able to ensure responses in autonomous driving.

1.5 Publications

As results from the work in the development of the Ph.D. thesis, the author cites the following publications:

1.5.1 Journal articles

Title: Youla-Kucera based Advanced Adaptive Cruise Control.
Authors: F. Navas, V. Milanés and F. Nashashibi.
Journal: IEEE Transactions on Vehicular Technology.
Status: Second revision submitted October 2018.

Title: Multi Model Adaptive Control for CACC applications.
Authors: F. Navas, V. Milanés, C. Flores and F. Nashashibi.
Journal: Control Engineering Practice.
Status: Second revision submmited September 2018.

Title: Youla-Kucera based Fractional Controller for Stable Cut-in/Cut-out Transitions in Cooperative Adaptive Cruise Control Systems.
Authors: F. Navas, R. de Charette, C. Flores, V. Milanés and F. Nashashibi.
Journal: IEEE Transactions on Intelligent Transportation Systems.
Status: Submitted September 2018.

Title: A Cooperative Car-Following/Emergency Braking System With Prediction-Based Pedestrian Avoidance Capabilities
Authors: C. Flores, P. Merdrignac, R. de Charette, F. Navas, V. Milanés and F. Nashashibi.
Journal: IEEE Transactions on Intelligent Transportation Systems
Number: 99 **Pages:** 1-10 **Year:** 2018.

1.5.2 Conference papers

Title: Youla-Kucera based lateral controller for autonomous Vehicle.
Authors: I. Mahtout, F. Navas, D. González, V. Milanés and F. Nashashibi.
Proceedings: 21st IEEE International Conference on Intelligent Transportation Systems
Place: Hawaii, USA **Date:** November 2018.

Title: Youla-Kucera control structures for switching.
Authors: F. Navas, I. Mahtout, V. Milanés and F. Nashashibi.

Proceedings: 2nd IEEE Conference on Control Technology and Applications.
Place: Copenhagen, Denmark **Date:** August 2018.

Title: Youla-Kucera based online closed-loop identification for longitudinal vehicle dynamics.
Authors: F. Navas, V. Milanés and F. Nashashibi.
Proceedings: 21st IEEE International Conference on System Theory, Control and Computing.
Place: Sinaia, Romania **Date:** October 2017.

Title: Using Plug&Play Control for stable ACC-CACC system transitions.
Authors: F. Navas, V. Milanés and F. Nashashibi.
Proceedings: 2016 IEEE Intelligent Vehicles Symposium.
Place: Gothenburg, Sweden **Date:** June 2016.

Chapitre. État de l'art

Below is a French summary of the following chapter "State of the art".

Au sein de l'automatisme, l'ingénierie du contrôle est considérée comme une technologie mature et cela de différentes manières. Elle s'inclut quasiment dans chaque type d'application dans le monde de l'industrie. La littérature regorge d'algorithmes de contrôle de systèmes, aussi complexes que soient les situations. Cependant, en pratique, de nombreux problémes apparaissent au moment de l'implémentation et plus particuliérement quand le système inclut des changements dynamiques, structuraux ou environnementaux. Pour résumer, beaucoup d'outils existent pour concevoir des contrôleurs appliqués aux systèmes dont la structure est connue. Toutefois, ces derniers ne fournissent pas les mêmes performances quand la structure du système contrôlé change au fil du temps. Le probléme de concevoir un contrôleur capable de traiter ces changements n'est pas nouveau. On peut citer le "Fault Tolerant Control" (FTC) spécialisé dans le cas des composants tombant en panne. Cependant, le travail dans le domaine est généralement limité par un nombre de pannes spécifiques auxquelles s'ajoute le probléme de l'apparition de nouveaux composants. Un autre domaine qui considère le changement au sein des systèmes est le contrôle adaptatif qui permet de suivre des changements pouvant être définis par la variation des paramètres du système controlé. Les changements nécessaires à la reconfiguration du système doivent être identifiés d'une manière ou d'une autre. Lorsque ces changements sont modélisés par des variations paramétriques prédéfinis, les changements structuraux ou l'introduction d'une nouvelle dynamique ne le sont pas. On retrouve aussi le contrôle robuste qui considère, quant à lui, un système dont les caractéristiques changent avec incertitudes. Quand ces incertitudes sont limitées, ce contrôleur fixe, garantit un comportement acceptable. Cependant, ce dernier n'est pas concevable dans les scénarios où ces changements sont trop importants. Les structures hiérarchiques gérant les changements des structures, à un instant précis, ont aussi été largement étudiées dans les domaines du contrôle décentralisé, distribué, hiérarchique et réseau. Traiter ces changements structuraux, implique aussi des transitions d'un système à un autre. Le domaine de "bumpless transfer" étudie le comportement des contrôleurs pendant la transition entre différent systèmes. Pour conclure, il y a différentes solutions dépendantes de la nature du problème. De plus, une structure de contrôle et de supervision, doit être appliquée lorsque différents types de changements interviennent.

La paramétrisation de Youla-Jabr-Bongiorno-Kucera (YK) est un outil de contrôle qui est apparu simultanément dans [Kučera, 1975,Youla et al., 1976a,Youla et al., 1976b]. La paramétrisation YK fournit tous les contrôleurs stabilisant un système donné. Ces derniers sont paramétrisés par la fonction de transfert appelée paramètre YK Q et donc : $K(Q)$. Il peut être utilisé pour réaliser la reconfiguration stable du contrôleur quand différents changements interviennent. Le contrôleur en question peut être classique, adaptatif, optimal ou robuste. Mélanger différents types de contrôleur est permis dans cette reconfiguration. La théorie duale de la paramétrisation YK, fournit tous les systèmes stabilisés par un contrôleur donné. L'ensemble de tous les systèmes stabilisés par un contrôleur dépend de la fonction de transfert appelée paramètre YK dual S, ce qui donne: $G(S)$. Ce paramètre peut représenter la variation paramétrique, les incertitudes de modélisation, changement de point d'opération .... Ce dernier est employé pour les identifications dynamiques et/ou l'identification de nouveaux capteurs/actionneurs connectés au système. Finalement, les deux paramètres YK peuvent être utilisés ensemble et ainsi une structure de contrôle qui change, basé sur des identifications dynamiques, est obtenue. Comme les capteurs/actionneurs sont identifiés, le contrôle hiérarchique et le FTC sont ainsi réalisable par la paramétrisation de YK. Le contrôleur est modifié en fonction des nouvelles dynamiques, avec différents critères de performance et de stabilité. Avec ces hypothèses, YK est capable de répondre à toutes les solutions proposées au dèbut de ce chapitre avec la même structure théorique, tout en garantissant la stabilité. Par conséquent, il peut servir d'outil général de contrôle pour les systèmes exposés aux changements dynamiques, structuraux ou environnementaux.

Ce chapitre donne un aperçu du domaine de recherche de la paramétrisation YK. Les origines de cette technique sont expliquées. Les travaux principaux à travers le monde sont étudiés, en se concentrant principalement sur les différents domaines d'application du contrôle basé sur cet outil mathématique. Les applications sont classées en fonction de l'utilisation de Q, de S ou bien des deux.

Chapter 2

State of the art

Control engineering is considered as a mature technology in many different ways, being able of dealing with almost any kind of application in the industrial context. The literature is rich in algorithms to design control systems, even highly complex control problems. But, in practice, several problems appear for its implementation, especially when the system is exposed to dynamics, instrumental or environmental changes. In short, a lot of tools exists to design feedback controllers for a system with a known structure, but they are not providing proper responses when the structure of the system to be controlled changes over time. The problem of designing a controller able to deal with these changes is not new: Fault Tolerant Control (FTC) specializes in the case of components that fail. The work in this area, however, is usually limited to a prespecified amount of faults, and the problem of handling new components is not addressed. Another field that considers changing systems, it is the adaptive control area, which allows tracking changes that can be defined as parameters in the controlled system. Changes for controller reconfiguration need to be identified somehow. Since those changes are already set as predefined parameters, structural changes or new dynamics introduction are not considered either. On the other hand, robust control considers a system that changes their characteristics over time through uncertainties. These changes are somewhat bounded, so a fixed controller can be designed, guaranteeing an acceptable behavior. However, robust control design is not possible in scenarios where changes in the system are large. Hierarchical structures to deal with running structural changes have been also widely studied in the areas of decentralized, distributed, hierarchical or networked controls. Handling these structural changes also involves dealing with the transients when changing from one system to the other. Considerations about transient behavior when doing controller reconfiguration can be found on the bumpless transfer control area. In short, there are many different solutions depending on the nature of the problem, and a control/supervision structure would be necessary to deal with all the types of changes that may come.

Youla-Jabr-Bongiorno-Kucera (YK) parameterization is a control framework that appeared simultaneously in [Kučera, 1975, Youla et al., 1976a, Youla et al., 1976b]. YK parameterization provides all stabilizing controllers for a given system. All stablizing controllers are parameterized based on the transfer function called YK parameter Q, so $K(Q)$. It can be used to perform stable controller reconfiguration when some change occurs. The type of controller could be any– classical, adaptive, optimal or robust control. Mixing different types of controller is allowed in this controller reconfiguration. The dual theory, dual YK parameterization, provides all the plants stabilized for a given controller. The class of all the plant stabilized by a controller depends on the transfer function called dual YK parameter S, so $G(S)$. This parameter could represent any plant variations, uncertainties, parameter variations, change of operation point, etc. This is employed for dynamics identification and/or identification of new sensors/actuators connected to a system. Finally, both can be used together, so a control structure that changes based on identified dynamics is obtained; as sensors/actuators are identified, hierarchical and fault tolerant control structures are also supported by YK. Controller is changed depending on new dynamics with some performance/stability criteria. With these premises, YK is able to encompass all the solutions proposed at the beginning of this chapter within the same theoretical framework and with stability guarantees. Thus, it could serve as a general control framework to deal with systems exposed to dynamics, instrumental or environmental changes.

The present chapter gives an overview of the YK parameterization research field. The origins of this technique are explained. Important groups worldwide are reviewed, focusing on the different type of control applications by using this mathematical framework. Applications are divided depending on whether Q or S are used, or both.

2.1 Origins

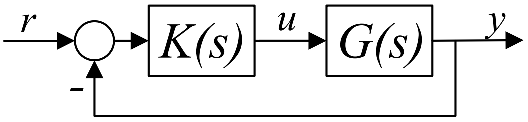


Figure 2.1: Negative feedback loop.

The origin of YK is in [Newton et al., 1957]. Given a Single-Input-Single-Output (SISO) stable plant, they found a way to parameterize all the controllers that stabilize it. Let's assume a feedback loop as in Fig. 2.1. If a stable controller $K(s)$ is connected to the plant $G(s)$ in a negative feedback loop, the transfer function of the control input u from the reference signal r yields:

$$Q(s) = \frac{U(s)}{R(s)} = \frac{K(s)}{1 + K(s)G(s)} \tag{2.1}$$

where if $Q(s)$ and $G(s)$ are known, the controller transfer function $K(s)$ can be recovered as follows:

$$K(s) = \frac{Q(s)}{1 - G(s)Q(s)} \tag{2.2}$$

From Eq. 2.2, it is clear that if $K(s)$ is a stabilizing controller, $Q(s)$ is stable and proper; thus any stable and proper transfer function $Q(s)$ represents a stabilizing controller for $G(s)$. The class of all stabilizing controllers for a plant is obtained. This could seem useless, but they observed that the nonlinear transfer function from reference r to output y in $K(s)$ becomes linear in $Q(s)$ (see Eqs. 2.3 and 2.4 respectively). Thus, the design of $Q(s)$ to achieve a desired performance is linear, obtaining $K(s)$ by back-substitution.

$$\frac{Y(s)}{R(s)}(K(s)) = \frac{K(s)G(s)}{1 + K(s)G(s)} \tag{2.3}$$

$$\frac{Y(s)}{R(s)}(Q(s)) = Q(s)G(s) \tag{2.4}$$

This idea of reparameterizing a set plant-controller in order to obtain linearity reappeared in [Zames, 1981]; and is well known as internal model control in chemical control process [Morari and Zafiriou, 1989]. But, it was not able to be applied for Multi-Input-Multi-Output (MIMO) systems.

[Kučera, 1975] and [Youla et al., 1976a,Youla et al., 1976b] proposed simultaneusly discrete and continous solutions to deal with MIMO unstable plants– so-called Youla-Kucera parameterization. There are two key points in the solutions: First, an initial stabilizing controller is considered; and second, plants are described using stable polynomial fractional transformations. Its use permited to see the plant as the combination of two stable transfer functions– e.g. an unstable plant $G(s) = 1/(s-5)$ is represented by $X(s)Y(s)^{-1}$ with $X(s) = 1/(s+1)$ and $Y(s) = (s-5)/(s+1)$. These factors were employed in order to obtain an equivalent to $Q(s)$ in Eq. 2.1. This new $Q(s)$, called YK parameter, characterizes the class of all stabilizing controllers depending on stable polynomial fractional factors for $G(s)$ and an initial $K(s)$. Linearity was preserved even if stable polynomial fractional factors were used.

This approach was updated with coprime factors in order to avoid algebraic difficulties as noticed by [Desoer et al., 1980] and [Vidyasagar, 1985] for SISO and MIMO systems. An efficent method for obtaining these factors is based on a state-space representation [Nett et al., 1984]. As those coprime factors are the basis for obtaining the class of all stabilizing controllers, this state-space representation is preserved in almost every future application.

The linearity of Q within the Closed-Loop (CL) function facilitates optimization over the class of all stabilizing controllers. Every single controller could be augmented with Q. This Q is seen as a stable filter that can be optimize offline or online in order to improve system's performace. An adaptive Q technique could be no longer useful when systems variations or uncertainites are large. A controller solution for such situations is provided by the dual YK parameterization.

Coprime factors of an inital plant connected to a stabilizing controllers are used in order to obtain the class of all plants stabilized by a controller. The connection between the dual YK and YK parameterizations was first developed by [Tay et al., 1989a], giving robust stability results. This dual YK parameter was used to suppress CL identification difficulties in [Hansen et al., 1989, Schrama, 1991]. The identification of a plant in the presence of a feeedback loop could be complex due to the noise. Given an initial model and controller, by identifying the dual YK S instead of $G(s)$, the CL problem is transformed into an Open-Loop (OL) like problem. This is called in the literature Hansen scheme. The resulting S is used to carefully redesign the filter Q such that a better performance is achieved without loosing the stability of the system.

Performance enhancement techniques working with an adaptive Q are seen particularly in the work of the Australian National University. It is also there that the dual YK parameterization and robust stability results were first developed. Later, the Technical University of Denmark analysed Q as a fixed filter, focusing in stable controller reconfiguration properties. Stable controller reconfiguration through Q is combined with a fault-detection system through the dual YK parameter S, obtaining a Fault-Tolerant-Control (FTC) solution. Finally, some results are in proceeding with structural changes through S at the University of Aalborg. A more detailed review of the YK design methodologies at the Australian National University, in the Technical University of Denmark and at the Aalborg University is in the following three sections.

2.2 Australian National University

The department of System Engineering, Research School of Information Sciences and Engineering at the Australian National University was the pioneer in using YK parameterization, to get what they called high performance control. The concept of high performance control is to use the tools of classical, optimal, robust and adaptive control in order to deal with complexity, uncertainty and variability of the real world. They aimed to find a mathematical framework able to join performance and robustness.

First steps in this direction were made by John Moore at 1970s, on a high order NASA flexible wing aircraft model with flutter mode uncertaintiess. Least square identification was used in order to have an adaptive loop based on linear quadratic optimal control able to achieve robustness to these uncertainties. However, the blending betweeen adaptive and robust control lacked a mathematical framework. A collaboration with Keith Glover at Cambridge University allowed them to discover the interpretation of the YK parameterization as a general solution to optimal control problem provided by [Doyle, 1983]. Doyle characterized the class of all stabilizing controller as an initial Linear-Quadratic-Gaussian (LQG) controller with a stable filter Q, what fits the adaptive filter that they put in their solution for the aircraft model. A graduate student, Teng Tiow Tay started working on how to use that theory, obtaining really encouraging results. Initial point of his thesis with Moore as supervisor [Tay et al., 1989b]. Different applications related to YK formed a new field of research. We refer to book [Tay et al., 1997] and article [Anderson, 1998] as principal referees for undestanding how to use YK parameterization towards high performance control. Different techniques and applications related to that are detailed below:

2.2.1 Q offline control design

An offline optimization of Q was carried out in order to achieve various performance objectives. The idea is to design a controller in the class of all stabilizing controllers instead of over the class of all possible controllers (which includes destabilizing controllers). Different control performance objectives can be set in order to optimize the YK filter Q. Performance requirements can be described in time or frequency domain. System norms in the frequency domain is directly related to optimal control. $\mathcal{H}_\infty$ is concerned primarily with the peaks in the frequency response, while $\mathcal{H}_2$ is related to the overall response of the system. The idea is simple, once a transfer function between different signal of interest is determined, $\mathcal{H}_\infty$ control designs a stabilizing controller that ensures that the peaks in the transfer function are knocked down; on the contrary, $\mathcal{H}_2$ or LQG control designs a stabilizing controller that reduces the $\mathcal{H}_2$ of the transfer function as much as possible. Penalization of the energy of the tracking error and control energy are examples of LQG control; while penalizing the maximum tracking error subject to control limits is an example of $\mathcal{H}_\infty$ control.

The design of an LQG controller with loop transfer recovery was analysed. This LQG controller uses a state estimator with the aim of estimating the non-accesible states of the plant. Its aim is to minimize error tracking and control effort. The controller will be optimal if a good model of the plant has been considered, otherwise the performance could be poor. Loop transfer recovery refers to the idea of reconfiguring the initial LQG controller to achieve full or partial loop transfer recovery of the original feedback loop. This is usually done through a scalar parameter as a trade-off between performance and robustness. In [Moore and Tay, 1989], loop recovery was achieved by augmenting the original LQG controller with the additional YK filter Q. They showed how full or partial loop recovery may be obtained depending if minimum or non-minimum phase plants are considered. The technique was ilustrated for the case of minimum and non-minimum phase plants through simulation. Improvements over standard loop recovery techniques were obtained.

The CL transfer function including Q from a disturbance input to a tracking error with a $\mathcal{H}_\infty$ norm is in chapter 4 of [Tay et al., 1997]. This equation is useful to keep the tracking error within a given tolerance. However, minimizing the tracking error could lead sometimes to large control efforts, what would be unnaceptable. As weighting factors between tracking error and control effort in a $\mathcal{H}_\infty$ setup is not allowed [Dahleh and Pearson, 1986], a l^1 equivalent was proposed in [Teo and Tay, 1995]. This algorithm allows to choose the correct weighting factors in a l^1 manner. A curve with all the possible solutions is generated for a simulation example, analysing the limitations and choosing the best weighting factors. This strategy is used in a hard disk servo system to minimize the maximum position error signal [Teo and Tay, 1996], which is the deviation of the read/write head from the center of the track.

2.2.2 Direct adaptive Q-control

In the previous section, an offline optimization of the YK parameter Q has been explained. [Wang et al., 1991] presented the first results in online optimization of Q without an identification process. The method is valid when the uncertainty is limited but unknown, and the plant-model mistmatch is not important. The optimization process is based on root-mean-square signals measures. A state-space relationship between a nominal plant with disturbances and a observer-based feedback controller $K(Q)$ is obtained. The order of Q should be fixed depending on the application. A steepest descent algorithm is used to obtain the parameter values of the predefined YK parameter Q, so the error is minimized from the disturbances on the system. Simulation results of the direct adaptive-Q controller were presented in [Tay and Moore, 1991] to ilustrate their performance enhancement capabilities when disturbances appear on the system. Part of these results were previouysly validated in a 55th order aircraft model with a controller design via LQG with Q augmentations for achieving resonance suppression in [Moore et al., 1989].

In chapter 6 of [Tay et al., 1997], this method was analyzed to discover its limitations. First, a perfect plant model with disturbances was considered, achieving again without problems an optimal control. Then, the model-plant mistmach case was analysed, seeing how the adaptive mechanism breaks down under severe model-plant mismatch. An identification algorithm would be needed when a large model-plant mistmach is present. Section below presents an extension of the first YK-based CL identification algorithm– Hansen scheme.

2.2.3 CL identification

CL identification provided by [Hansen et al., 1989] is extended when connected to a controller with the YK filter Q in chapter 5 of [Tay et al., 1997]. Robust stabilization results in [Tay et al., 1989a] connecting $K(Q)$ and $G(S)$ are used to obtained an unbiased identification of S when a YK parameter Q is applied. A time-invariance property of Q is considered in the result. The unbiased CL identification of S is done through the identification of $\hat{S} = S(I - QS)^{-1}$, which includes Q. In order to obtain the real value of S, Q needs to be known. This CL identification method is the basis of the iterated (Q, S) control design shown below.

On the other hand, the original Hansen scheme is also extended with a non-linear initial model $G(s)$ connected to a stabilizing controller $K(s)$ in [Linard and Anderson, 1996, Linard and Anderson, 1997]; and [De Bruyne et al., 1998] presented a modification able to tune the order of the resulting model given by the Hansen scheme.

2.2.4 Iterated/Nested (Q, S) control design

This section considers solutions with unmodeled dynamics in the nominal model of the plant. $K(Q)$ is seen as a controller where Q is changed online, as well as $G(S)$ is seen as a nominal plant with an augmentation related to unmodeled dynamics. The process is the following: First, a nominal controller $K(s)$ is designed for a nominal plant $G(s)$. Plant-model mistmatch is identified through the dual YK parameter S, and then the augmented controller Q is designed to optimally control S to some performance criteria. The performance is usually the same that the one used for the initial controller. These ideas are based on the robust stabilization concept in [Tay et al., 1989a]. Iterative and nested solutions are present in chapter 5 of [Tay et al., 1997].

For iterative control (Q, S) design, an initial stabilizing controller is designed for the nominal plant $G(s)$. Then, unmodeled dynamics represented by $\hat{S}$ are identified by using the CL identification method proposed in [Tay et al., 1997]. It avoids bias problems in the identification process. $\hat{S}$ is used in an iterative manner for finding the Q that improves the performance criteria. Iteration is needed as the value of $\hat{S}$ won't be reliable enough at the very beginning or due to new deficiencies in the model. In each iteration the order of the controller increases as $\hat{S}$ includes the applied Q, followed by a control update step. The sucess of this method relies on the use low-order approximations of S. Controller reduction will be also crucial in practical implementation. Simulation results related to Iterated Pole-Placement, Linear-Quadratic (LQ) control design and $\mathcal{H}_\infty$ control are in [Tay et al., 1997].

For nested control (Q, S) design, succesives S are identified on the residual mistmatch between model and plant. An external signal needs to be injected in order to identify the new S. In each step the model of the system is updated. This new model is then taken into consideration for obtaining a new Q, until the performance criteria is fulfilled. This kind of structure is practical when a plant is described by m recursive fractional forms. It could be the case of a complex model composed by low order models, so the controller could be broken down into a sequence of low order controller designs for a sequence of low-order models. Thus, a S is identified for each fractional form of the model, and a Q affined in the nested control structure.

2.2.5 Indirect adaptive (Q, S)-control

Sometimes, the major limit is the little knowledge about the plant. In such cases, iterated and nested control designs have been proposed. But, these algorithms are limited to a time-invariance Q property. In order to deal with a time variance Q, an adaptive version of nested control was proposed in [Tay et al., 1989b]. In any adaptive algorithm a fixed structure of Q is created. Parameters in Q are the ones changing depending on the model-plant mistmach identified by S. Notice how the unbiased identification provided in previous sections is no longer available as Q varies with time. External excitation signals are needed in order to identify S, and this could compromise the control performance. Two methods were proposed to solve that:

The use of two different time scales in the adaptive algorithm, a faster one for the identification of S, and a much slower for the adaptation of Q. This idea was proposed in the PhD thesis [Wang, 1991].

In chapter 7 of [Tay et al., 1997] a different method is explored. They considered that the model-plant mistmatch is significant in a frequency range above the passband of the nominal control loop. In fact, if Q is present on the system, the identification of S is frustrated. In order to solve that, the idea was to augment Q with a filter in the frequency of the excitation signals needed for the identification of S. The filtered excitation algorithm provides a suitable method for including external signal in a particular frequency range to identify S, without compromising the control performance.

A 55th order aircraft model was used in the literature for obtaining results that validates indirect (Q, S)- control. Adaptive LQG and pole-placements solutions were presented in [Chakravarty and Moore, 1986] and [Chakravarty et al., 1986] to suppress wing flutter.

2.3 Technical University of Denmark

The potential of YK were further explored in recent years. There is a large amount of work developed by Professor Hans Henrik Niemman in the Electrical Engineering Department of the Technical University of Denmark. Control solutions considering Q, S and both are again developed.

Related to the class of all stabilizing controller for a given plant parameterized by Q, the concern is in stable controller reconfiguration. [Niemann and Stoustrup, 1999] showed how it is possible to change between multivariable controllers online in a smooth way, guaranteeing CL stability. The focus is not in the design of Q for obtaining a desired performance, but in the use of Q as a stable transition between an initial and a final controller. Switching between two or more controllers is considered. The stability proof is extended with a numerical example in [Niemann et al., 2004]. It is shown how linear switching between two controllers results unstable, while the use of YK turns stable the same problem. Finally, structural changes are considered in connection with YK in [Niemann, 2006a]. It is demonstrated how it is possible to introduce new sensors/actuators into the system, and use them in the YK parameterization. The stability of the CL system is still affine in Q even if new sensors or actuators are added. This work will be the basis of the Plug&Play project presented in next section.

YK stable controller reconfiguration is based in the absence of uncertainties in the plant; otherwise, the dual YK parameterization needs to be used. [Niemann and Stoustrup, 1999] presented a relation between S and system variations, with robust stabilization results similar to those in [Tay et al., 1989a]. The general model-plant mistmatch represented by S, and used by the Australian National University, is reformulated by Niemman in [Niemann, 1999]. A connection between a nominal plant model with a uncertainty block Δ is done through a Linear-Fractional-Transformation (LFT). The dual YK parameter S is in function of the block uncertainty Δ, yielding $S(\Delta)$, which is closer to a robust control design. CL transfer function is also analysed depending on $S(\Delta)$. Eight types of system descriptions S in function of Δ are in [Niemann, 2003]. The method is constraint to the exact knowledge of Δ. A literature review shows different applications in funcion of this $S(\Delta)$ as:

- H_∞ control design with partial uncertainty description. There is a problem when trying to design a H_∞ controller if uncertainty description is not full complex. An iterative process as the one in [Lin et al., 1993] could be used, but problems as increasing controller order or non-optimal solutions will be faced up. A transformation betweem Δ and S allows to a have a full complex uncertain block, avoiding these problems [Niemann, 1999].

- Model validation with partial uncertainty description. The idea of model validation is to detect/estimate variations in the system that are not described in the model. Variations could be several: uncertainties, parameter variations, change of the operation point, etc. Model validation can be done offline or online depending on the desired control application. FTC is an example where an online methodology is needed. [Savkin and Petersen, 1995] presented an online model validation algorithm based on Integral Quadratic Constraint (IQC) for full complex uncertainty. Dual YK parameterization serves again as a conversor from partial to full uncertainty description.

- Performance validation. [Niemann, 2003] proposed to make a connection between $S(\Delta)$ and CL performance criteria, so an upper bound of S with respect to the uncertainty can be used for validation. An estimation of Δ based on S could be also obtained if the system description with respect to uncertainty is known.

- Parameter estimation for gain scheduling. Gain scheduling techniques are motivated by the large number of control applications that have significant nonlinearities which can not always be handled well by linear control design techniques. [Niemann and Stroustrup, 1999] considered the case where a gain scheduling controller is needed, but the scheduling parameter vector cannot be directly measured. An estimation of the same is done using the dual YK parameterization. Dual YK parameterization in connection with the paremeter gives a validation method, which gives very precise parameter estimation. Then, this parameter can be directly employed on a gain scheduling controller.

- Modified Hansen scheme. A modification of the Hansen scheme is carried out in [Sekunda et al., 2015]. It gets rid of signals that are not directly measurable. Some a priori knowledge and numerical accuracies are reduced. Basis of this method is in the CL modification already carried out in [Tay et al., 1997].

Finally, this dual YK parameter description is connected to the class of all stabilizing controllers $K(Q)$. Optimal and FTC solutions are proposed:

- Optimal control design. There is a connection between S with Δ and Q. The idea is to find the optimal value of Q that minimized the value of S, so the nominal performance can be preserved [Niemann and Stoustrup, 2000]. Algorithms in high performance control are proposed as solution, but with $S(\Delta)$.

- Fault tolerant control. Dual YK parameterization is used in the design of FTC systems. When a fault appears in a system, a nonzero S results. If S is unstable, the fault makes the CL system unstable. Then, controller reconfiguration needs to be carried out to recover stability. This reconfiguration is done through the YK parameter Q. YK allows fault diagnosis and the corresponding controller reconfiguration in the same approach [Niemann and Stoustrup, 2002]. A connection with different additive and parametric faults is in [Niemann and Stoustrup, 2005]. The system with additive faults is directly change with Q, without consideration of S, as it shouldn't affect the CL stability. When it comes to parametric faults S plays a key role, obtaining the value of Q that makes stable $1/(1-QS)$. An example with a servo is given [Niemann and Stoustrup, 2005]. Deviations on the value of tacho gain make the system unstable. This is seen as an unstable S. Different Q's are obtained depending on the deviation value. The optimization of Q is done offline, so only fault diagnosis will be needed in order to choose the proper value of Q. This method is restricted to CL system with one fault. The fault diagnois method based on dual YK is extended in [Niemann, 2006b] to deal with OL systems and CL systems with a feedback controller different from the nominal one. The latter is important in FTC, as fault diagnosis should be running after the first fault has been detected and the controller has been reconfigured.

2.4 Aalborg University

The concept of dealing with structural changes ensuring stability of the system as in [Niemann, 2006a] has been also used at the University of Aalborg (Denmark). A collaboration exists between Professor Niemann and Professor Jakob Stoustrup (check some of the references in previous section). Professors Jan Dimon Bendsten and Klaus Trangbaek are other co-workers in the main project related to YK: Plug and Play Process control (P&P). The project was developed by the Department of Electronics Systems of Aalborg University from 2006 to 2011.

The idea was to investigate control problems for complex systems with a modular structure. Because of that, the fundamental aspect was to understand how to detect the addition of components to the system, reconfiguring the controller to maintain the system stable and improve performance. The addition of subsystems could be any kind of sensors or actuators. The general theory is explained in papers [Bendtsen et al., 2013] and [Stoustrup, 2009]. In fact, three different scenarios where sensors are included were the referenced ideas to carry out.

- Imagine a stable where the pigs are not comfortable. The farmer plugs a new sensor in a vacant socket in that part of the stable to stabilize the indoor climate in the proximity of the sensor. The stable ventilation system needs to automatically register the new component and reconfigures the control law.

- Imagine the situation where biomass is being added to the fuel of a power plant causing large thermal stresses to the boiler. Instead of shutting down the plant, the operator of the plant sticks on a few sensors in the stressed areas. Thanks to the P&P control, after few minutes the controller is reconfigured and the thermal stresses are within permissible bounds.

- Imagine a grocer who buys a new refrigerated display for his shop. He plugs it in himself. His compressor rack and the condensators on the roof start sounding slightly different, and after a couple of hours the new display case as well as all the old ones work correctly. The eco-meter in his backstore room displays optimal power consumption for all of them.

A total of five companies participated in the P&P control research program; Danfoss, Grundfos, Skov, DONG Energy and FLSmidth Automation, each providing different case studies. A literature review related to the use of YK in P&P control has been carried out:

- Reconfiguration of existing controllers whenever structural changes are introduced in the system being controlled. The class of all stabiling controllers provided by YK is used to have stable controller reconfiguration when some change happens. The focus is in the correct integration of sensors/actuators and corresponding controller reconfiguration trough Q. Extended results in [Niemann, 2006a] are crucial. First results related to a buffer tank model are in [Trangbaek et al., 2008] and [Trangbaek and Bendtsen, 2009]. A manual valve is replaced for an automatic one, augmenting the original controller through Q in order to improve the general performance of the system. Experimental results are in [Bendtsen et al., 2013] for laboratory-scale model of a district heating system and a livestock stable climate system: In the district heating system, as consumers are not happy with the variable supply rate, differential pressure sensors are added to examine the problem. That revealed a performance problem, so control capabilites are added to another pump, improving the initial LQG controller through the corresponding augmented Q; a real livestock stable is also considered, which is not completely airtight due to crack in the walls. The climate system is initially with a single temperature sensor; but the farmer detects another area in the stable with an extra draft. The sensor does not reach this area, so YK is used in order to integrate a second temperature sensor, making the livestock stable temperature homogeneous.

- CL identification. [Bendtsen et al., 2008] modified the Hansen scheme proposed in [Hansen et al., 1989] in order to deal with new measurements that become available during online operation. New dynamics related to new sensors are simply identified by the dual YK parameter S. On the other hand, this Hansen scheme is extended to Linear-Parameter-Varying (LPV) systems in [Bendtsen and Trangbaek, 2014]. LPV system is a linear state-space representation whose dynamics vary as function of certain time-varying parameters called scheduling parameters. Interesting results are obtained in terms of stability, and doubly coprime factors based on these scheduling parameters. Simulation results of coupled dynamics identification in heat distribution systems are in [Trangbaek and Bendtsen, 2010].

- Automatic control reconfiguration to achieve optimal performance together with identification. Here, the CL identification provided by dual YK is used in order to improve controller performance. Controller reconfiguration is carried out in a simulation district heating system, once couple dynamics are identified through S [Trangbaek, 2009]. Strong coupling in the network is due to the addition of a second pump demanded by a consumption increment. Other contribution is in the area of Multi Model Adaptive Control (MMAC). MMAC is a supervisor who chooses the proper controller among pre-designed candidates controllers once more information is known about the plant. Controllers are designed based on a predefined set of linear models. Once the closer model in the set is known, the switching is direct. Results in [Anderson et al., 2001] and [Baldi et al., 2011] are improved in [Bendtsen and Trangbaek, 2012], as noise correlation problem in CL is supressed by employing the dual YK parameterization. A LPV simulation example with a total of five predefined Linear-Quadratic-Regulator (LQR) controllers is provided; the closer model in the set to the real system is chosen, switching to the corresponding controller through the correct Q. Finally, as in MMAC the switching is based on the closer model in a predefined set, nobody assures that the switching of the controller with the real plant results in a stable CL. This situation is analysed in [Trangbaek, 2011].

2.5 Discussion

This chapter reviewed the YK control framework state-of-the art. Origins of the class of all stabilizing controllers for a plant $K(Q)$, and its dual version, the class of all plants stabilized by a controller $G(S)$ are explained. Robust stabilization results between Q and S are fundamental for the YK-based applications in the field of optimal, robust, adaptive and fault-tolerant control. These applications are mainly developed in three different institutions: Australian National University, Technical University of Denmark and Aalborg University. A state-of-the-art classification is in Table 2.1. YK applications timeline from the origin to the most recent work is in Fig. 2.2.

The Australian National University was the first to use YK parameterization as a control tool able to use classical control, optimal control, robust control and adaptive control theories together. High performance control goes beyond all of them by blending the strengths of each to obtain the best performance possible in a real world subject to uncertainties and system variations. Offline methods related to optimal LQG control and robust $\mathcal{H}_\infty$ control depending on Q are introduced to achieve various performance objective. An adaptive version of the same is also proposed for cases where disturbances and uncertainties are not fully known. The dual YK parameterization plays a key role when unmodeled dynamics are present on the system. Iterative, nested and adaptive solutions consider the identified dynamics provided by S in order to optimize the YK filter Q. Theoretical basis present in [Tay et al., 1997] is strong and exemplified through simulation results for a hard disk servo system and a 55th order aircraft model. Experimental results with real applications are missing in the literature, especially for iterated/nested solutions. This is due to the degree explosion of the solution; with each iteration the order of the resulting parameters S and Q increase. Related also to the order, it looks complicated to get a simple representation of S even if the model-plant mistmath is simple. Order reduction techniques and model simplication should be carried out to make these solutions viables. There is neither an explicit parameterization when considering decentralized control.

On the other hand, at the Technical University of Denmark, Professor Niemann extended the YK parameterization of all stabilzing controllers with additional sensors/actuators. This would be useful to solve decentralized and FTC problems. Research interests are dual YK parameter S description based on block uncertainty Δ. Several applications related to control optimization, performance and model validation are derived, but no simulation or experimental results are present in the literature. A YK-based fault tolerant control solution is also proposed, integrating controller reconfiguration, fault diagnosis and isolation in the same approach. The most advance FTC control architecture is in [Niemann, 2012]. Start-up or safe mode coexists with normal, full performance, reduced performance and closed-down modes. Fault detection based on dual YK parameterization determines which mode is applied through the corresponding Q. Safe mode is activated during start-up and fault isolation. Closed-down is set when the loop becomes irremediably unstable after a fault. Again, experimental real cases are missing.

University of Aalborg, through its project called Plug&Play control developed a novel concept for process distributed control, which allows the control system to self reconfigure once an instrumental change is introduced. The idea is similar to Niemann's; in fact, an active collaboration exists between both universities. Extended version of YK parameterization is crucial. While Niemann's idea is more in the field of faults (a sensor or an actuator fails), here the objective is the opposite: a sensor or an actuator is plugged in, and the controller is reconfigured to enhance performance. An active collaboration between both should be set up in order to get a control system with full capabilities. Closed-down mode could be avoided if the correct sensor/actuator is plugged in. Simulation and experimental results well exemplified the application of the theory. It is by far, the part of the literature that presents more detailed and clear examples.

Once the state of the art of YK control framework has been carried out, current challenges are associated to the non-linear extension of the YK parameterization; integration of intelligent control system as fuzzy control, model predictive control, genetic algorithm or neuronal networks; transition analysis for the different YK-based control structures for switching in the literature; analysis of a scalar factor regulating the action between controllers through Q in order to improve the performance of the system; and extension of YK-based FTC and P&P to a more general control structure (they are all build with an observer-based feedback controller).

Different vehicles dynamics depending on longitudinal speed, emergency maneuvers in the stability limit of pneumatic systems, or vibrations in chassis control are some examples of what a unique plant, as a vehicle, needs to handle through different control solutions. YK represents a suitable technique for its application in ITS. But, almost all the studied cases are mainly focused in system with very low dynamics, except from some simulation with an aircraft model in high performance control. There is neither a faster dynamics case, which in the case of ITS, sensor/actuators fails could result in a traffic accident. The application of YK in autonomous driving will not only serves as a tool, but as extension to real fast dynamics experimental cases.

Table 2.1: Summary table state-of-the-art Youla-Kucera.

YK	Technique	Technique description	Implemeted in
Q	An LQG design	Offline $\mathcal{H}_2$ control design. Loop recovery in LQG control through YK parameter Q.	[Moore and Tay, 1989]
	An l^1 design approach	Offline $\mathcal{H}_\infty$ control design with a l^1 modification for error and control weighting.	[Dahleh and Pearson, 1986] [Vidyasagar, 1991] [Teo and Tay, 1995] [Teo and Tay, 1996]
	Direct adaptive Q-control	Online optimization of Q based on averaging techniques when disturbances are present on the system.	[Tay and Moore, 1991] [Wang et al., 1991] [Tay et al., 1997]
S	CL identification tunable order	The order of the resulting Hansen scheme model is tunable.	[De Bruyne et al., 1998]
	CL identification non-linear	Hansen scheme is extended with a nonlinear initial model.	[Linard and Anderson, 1996] [Linard and Anderson, 1997]
(Q, S)	Iterative/Nested (Q, S) design	Q is modified depending on the identified S following a given performance criteria. Q remains invariant through time. S represents the model-plant mistmatch. Difference between iterated and nested is in the identification process of S.	[Tay et al., 1997]
	Indirect adaptive-Q control	Adaptive version of nested (Q, S) control design. Q has a fixed order and is time variant; its parameters are changed depending on S identification.	[Tay et al., 1989a] [Yan and Moore, 1992] [Yan and Moore, 1996]

	YK	Technique	Technique description	Implemeted in
Technical University of Denmark	Q	Stable controller reconfiguration	Stable controller reconfiguration through Q given two or several controllers.	[Niemann and Stoustrup, 1999] [Niemann et al., 2004]
		System structure changes	Sensors and actuators are included in the class of all stabilizing controllers provided by YK.	[Niemann, 2006a]
	S	Relation with uncertainty Δ	$S(\Delta)$ provides a full complex uncertainty description that solves H_∞ control and online model validation problems.	[Niemann, 1999] [Niemann, 2003]
		Performance validation	A connection between $S(\Delta)$ and CL is done, so an upper bound of S is used for performance validation.	[Niemann, 2003]
		Parameter estimation for gain scheduling	Dual YK is used in order to get a more precise estimation of non-measurable gain scheduling parameters.	[Niemann and Stroustrup, 1999]
		CL identification	Hansen scheme is modified to get rid of signals that are not directly measurable.	[Sekunda et al., 2015]
	(Q,S)	Fault tolerant control	Connection between fault diagnosis and controller reconfiguration with YK and dual YK parameterizations.	[Niemann and Stoustrup, 2002] [Niemann and Stoustrup, 2005] [Niemann, 2006b] [Niemann and Poulsen, 2009b] [Niemann, 2012]
		Optimal controller design	Connection between S(Δ) and Q. The idea is to find the optimal value of Q that minimized the value of S.	[Niemann and Stoustrup, 2000]

	YK	Technique	Technique description	Implemeted in
University of Aalborg	Q	Controller reconfiguration with new sensors/actuators	Integration of new sensors/actuators and consequence controller reconfiguration. YK-based practical examples.	[Trangbaek et al., 2008] [Trangbaek and Bendtsen, 2009] [Bendtsen et al., 2013]
	S	LPV CL Identification	Hansen scheme is extended to LPV systems.	[Trangbaek and Bendtsen, 2010] [Bendtsen and Trangbaek, 2014]
		CL identification with new sensors	New dynamics related to new sensors are identified in a modified Hansen scheme.	[Bendtsen et al., 2008]
	(Q, S)	Optimal control	YK-based controller reconfiguration and CL identification are employed to improve system performance.	[Trangbaek, 2009]
		Multi model adaptive control	A YK-based MMAC is used to avoid noise correlation problems.	[Bendtsen and Trangbaek, 2012] [Trangbaek, 2011]

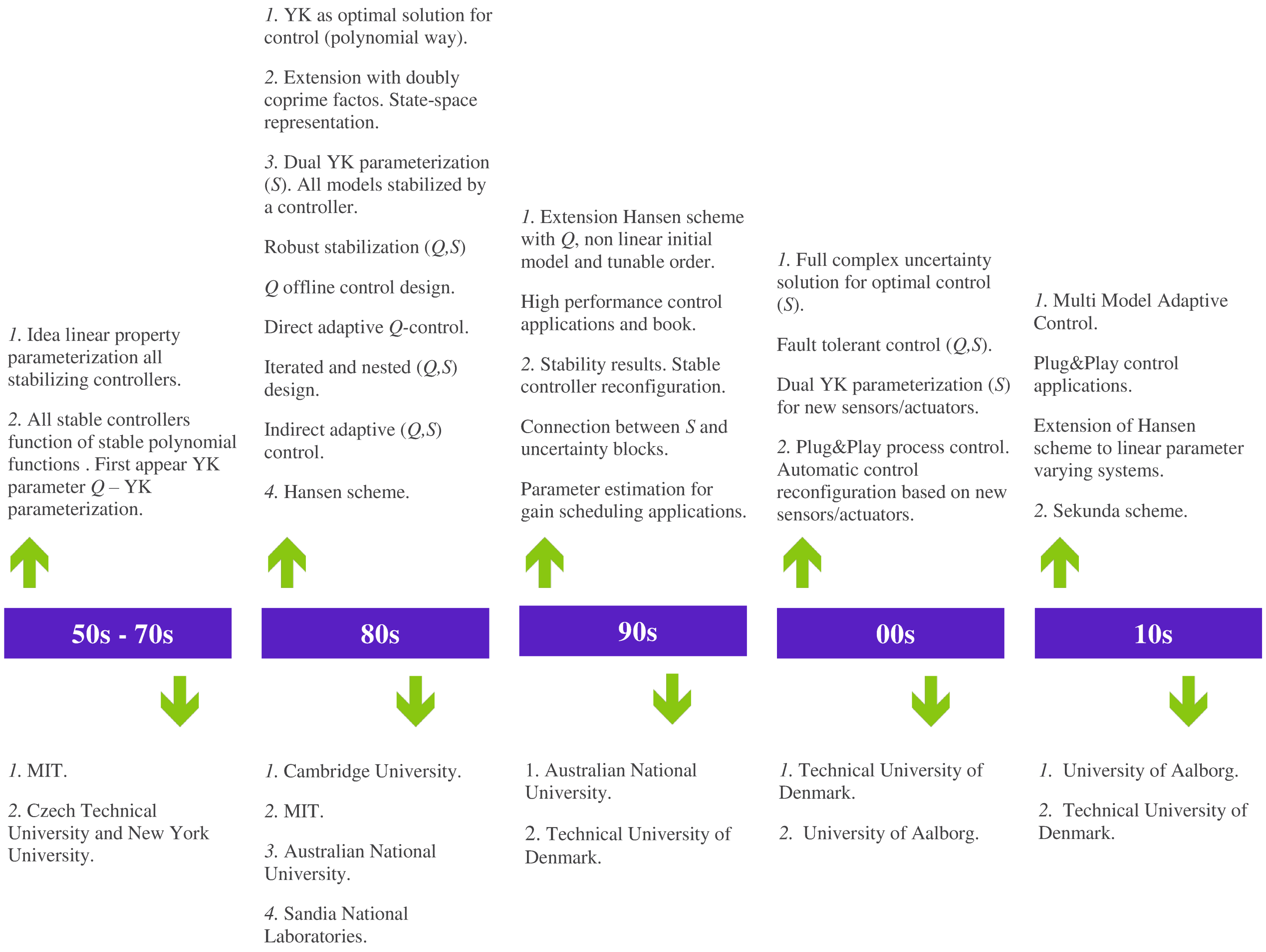

50s - 70s
1. Idea linear property parameterization all stabilizing controllers.
2. All stable controllers function of stable polynomial functions . First appear YK parameter Q – YK parameterization.
1. MIT.
2. Czech Technical University and New York University.
80s
1. YK as optimal solution for control (polynomial way).
2. Extension with doubly coprime factos. State-space representation.
3. Dual YK parameterization (S). All models stabilized by a controller.
Robust stabilization (Q,S)
Q offline control design.
Direct adaptive Q-control.
Iterated and nested (Q,S) design.
Indirect adaptive (Q,S) control.
4. Hansen scheme.
1. Cambridge University.
2. MIT.
3. Australian National University.
4. Sandia National Laboratories.
90s
1. Extension Hansen scheme with Q, non linear initial model and tunable order.
High performance control applications and book.
2. Stability results. Stable controller reconfiguration.
Connection between S and uncertainty blocks.
Parameter estimation for gain scheduling applications.
1. Australian National University.
2. Technical University of Denmark.
00s
1. Full complex uncertainty solution for optimal control (S).
Fault tolerant control (Q,S).
Dual YK parameterization (S) for new sensors/actuators.
2. Plug&Play process control. Automatic control reconfiguration based on new sensors/actuators.
1. Technical University of Denmark.
2. University of Aalborg.
10s
1. Multi Model Adaptive Control.
Plug&Play control applications.
Extension of Hansen scheme to linear parameter varying systems.
2. Sekunda scheme.
1. University of Aalborg.
2. Technical University of Denmark.

Figure 2.2: Timeline Youla-Kucera parameterization.

Chapitre. Paramétrisation Youla-Jabr-Bongiorno-Kucera

Below is a French summary of the following chapter "Youla-Jabr-Bongiorno-Kucera parameterization".

Ce chapitre présente les bases mathématiques pour lesquelles la paramétrisation de Youla-Jabr-Bongiorno-Kucera (YK) permet de réaliser une reconfiguration stable des contrôleurs. La paramétrisation YK décrit l'ensemble de tous les contrôleurs qui stabilisent un modèle de système Linéaire Invariant dans le Temps (LIT). Cette paramétrisation est basée sur la double factorisation coprime. Cet ensemble est fonction du paramètre YK Q, qui est une fonction de transfert stable. La reconfiguration stable des contrôleurs est réalisée entre différents contrôleurs, en choisissant différents Q's. Une description générale d'un modèle de système LIT est donnée en même temps que les critères nécessaires à la conception d'un contrôleur stable. Cette description générale est employée dans le but d'obtenir les doubles coprime facteurs nécessaires. Ces facteurs permettent la génération de la classe des contrôleurs stabilisant le modèle. Des détails sur l'utilisation des ces facteurs pour la reconfiguration des contrôleurs ainsi que la preuve de la stabilité sont donnés dans ce chapitre. Le chapitre est structuré comme suit: La section 3.1 décrit un modèle ainsi qu'un contrôleur général. Un contrôleur stabilisant est aussi fourni vérifiant le concept de stabilité en Boucle Fermée (BF). Dans la section 3.2, la double factorisation coprime et l'identité de Bézout associée, sont analysées. La section 3.3 décrit la classe de tous les contrôleurs stabilisants. Elle décrit aussi comment obtenir une reconfiguration stable des contrôleurs entre deux ou plusieurs contrôleurs. La preuve est également donnée dans cette section. La section 3.4 montre différents exemples numériques, pour une analyse détaillée des propriétés de YK. Enfin, quelques commentaires sont donnés en conclusion.

Chapter 3

Youla-Jabr-Bongiorno-Kucera parameterization

This chapter presents the mathematical basis in which Youla-Jabr-Bongiorno-Kucera (YK) parameterization relies on for making stable controller reconfiguration. YK parameterization describes the collection of all controllers that stabilize a Linear-Time-Invariant (LTI) plant model. This parameterization is based on the doubly coprime factorization. This collection is a function of the YK parameter Q, that can be any stable transfer function. Stable controller reconfiguration is carried out between different controllers by choosing different Q's.

A general description of a LTI plant model is given together with criteria for stable control design. This general description is employed for obtaining the needed doubly coprime factors. These factors permit the generation of the class of all the stabilizing controllers. Details on how to use them to perform controller reconfiguration and stability proof are in the present chapter.

The chapter is structured as follows: Section 3.1 describes a general model/controller description, introducing a general plant model. A stabilizing controller is also provided, reviewing the Closed-Loop (CL) stability concept. In section 3.2, doubly coprime factorization and the associated Bézout identity are analysed. Section 3.3 describes the class of all stabilizing controllers, and how to obtain stable controller reconfiguration between two or several controllers. Stability proof can be found within the section. Section 3.4 shows some numerical examples, analysing in detail YK properties. Finally, some concluding remarks are given.

3.1 System description

This section describes some basic notation, which will be used extensively in the sequel. A general description of a nominal plant model and a stabilizing controller is given. This notation can be found in many books and papers; readers are referred to [Zhou et al., 1996] and [Ogata, 2013].

3.1.1 The nominal plant model

The control design begins with the modeling of a physical system. A model is a mathematical representation of the system dynamics. Models allow to reason about a system and make predictions about how the system will behave. In this text, the interest is in models describing the input/output behavior of physical systems. Roughly speaking, a dynamical system is one in which the effects of actions do not occur instantly. For example, the velocity of a vehicle does not change at the same moment that the gas pedal is pushed; or a headache does not vanish right after an aspirin is taken. Input/output behavior can be really different: from the simpliest one with a gain and a rising time, to the more complex one with a higher order response, delays, dead time and so on. The number of inputs and outputs can also vary, having different response models between them. In an industrial context, the number of inputs and outputs depend on the number of sensors and actuators: Single-Input-Single-Output (SISO) or Multi-Input-Multi-Output (MIMO) systems. Input-output behavior is merged with some performance and external disturbances relations in a more general plant:

$$P = \begin{bmatrix} e \\ y \end{bmatrix} = \begin{bmatrix} G_{ew} & G_{eu} \\ G_{yw} & G_{yu} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} G_{ew} & G_{eu} \\ G_{yw} & G \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \tag{3.1}$$

where u is the variable subject to control, so-called control input, sent to the system actuators; w is the disturbance vector, also called exogeneous input or auxiliary input; y is the measurement vector coming from the sensors; and e is the controlled external output signal, which includes signals of interest in system's performance. e often coincides with the measurement signal y. G_{ew}, G_{eu} and G_{yw} represent external disturbance effects and performance requirements, while G_{yu} is the real plant. Note that $G_{yu} = G$ represents the input/output behavior or internal dynamics of the physical system to control. This model includes the modeling of all sensors and actuators for the plant. In the case of LTI continous systems, the state space representation of G yields:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned}, \quad G = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \tag{3.2}$$

where t indicates time, $x(t)$ is the state vector, $\dot{x}(t) = dx/dt$ is the evolution over time of the state vector, $y(t)$ the measurement vector and $u(t)$ the control vector. Coefficients A, B, C and D are constant matrices. Transfer function may be found directly from a state-space representation through the Laplace transform s:

$$G(s) = C(sI - A)^{-1}B + D = \frac{Y(s)}{U(s)} = \frac{b_m s^m + b_{m-1}s^{m-1} + ... + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + ... + a_1 s + a_0} \tag{3.3}$$

where $Y(s)$ denotes the s-transform of the filter output signal $y(t)$ and $U(s)$ denotes the s transform of the filter input signal $u(t)$. m is the numerator order, while n is the transfer function order. $U(s)$ and $Y(s)$ allow to find poles and zeros of the transfer function $G(s)$.

Definition 3.1. *Zeros are the complex angular frequencies that make the overall gain of the filter transfer function zero: The values of s where $Y(s) = 0$.*

Definition 3.2. *Poles are the complex angular frequencies that make the overall gain of the filter transfer function infinite: The values of s where $U(s) = 0$.*

Once both poles and zeros are found, they can be plotted onto the s-plane. The s-plane is the complex plane on which Laplace transforms are graphed. Analysing the complex roots of an s-plane equation and plotting them reveal information about the frequency response and stability of the system.

Stability of G is directly related to poles position. If a small perturbation arises in the system inputs or initial conditions, a stable system will present small modifications in its perturbed response. In an unstable system, any perturbation, no matter how small, will make states or outputs grow unbounded until the system saturates or stops working.

The concept of stability refers to stable response to bounded inputs, assuming zero initial conditions; and stable response to different initial conditions, assuming inputs zero. A system

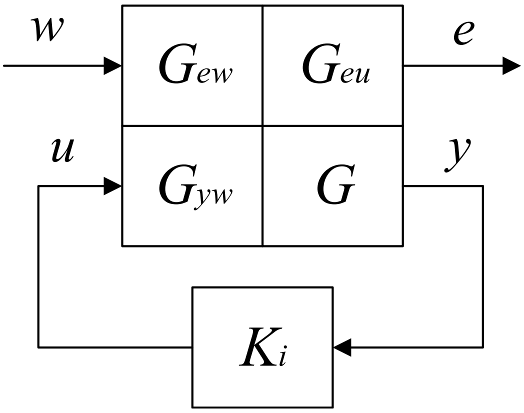


Figure 3.1: CL map of a general plant P with a controller K_i.

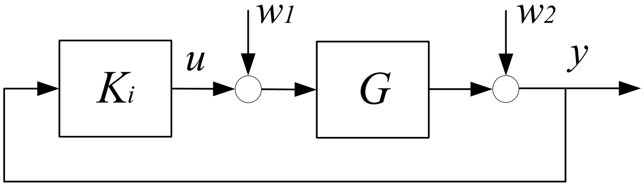


Figure 3.2: Feedback controller connection.

is asymptotically stable if its state response converges to the origin for any initial condition or bounded input. Here, for LTI continuous systems, stability is defined as:

Theorem 3.1. *A LTI continous system with a transfer function $G(s)$ is asymptotically stable if and only if every pole is in the left half-plane: $z \in \mathbb{C} : Re(z) < 0$.*

A LTI continous system with state space representation G is stable if and only if all the eigenvalues of A have a negative real part: $\mid A - \lambda I \mid = 0 \quad with \quad Re(\lambda) < 0$.

Stability condition is equivalent to belonging to the subspace $\mathcal{RH}_\infty$. $\mathcal{RH}_\infty$ is a real rational subset of $\mathcal{H}_\infty$ with all proper and real rational stable transfer functions/matrices.

Once the plant model has been described, the objetive of a controller will be to change the position of zeros/poles to make the system behaves according to some stability/performace criteria.

3.1.2 The stabilizing controller

In this subsection, the general plant model in Eq. 3.1 is connected to a general controller K_i. Subindex i refers to different controllers $i \in [0, p]$, where p is the number of controllers (useful notation for subsequent controller reconfiguration). CL stability conditions for feedback control and feedforward/feedback control are revisited. The CL map of P with a controller K_i is shown in Fig. 3.1. Equivalently to Eq. 3.2, a LTI continous state-space representation of a general controller is given below:

$$\begin{aligned} \dot{x}(t) &= A_i^c x(t) + B_i^c y(t) \\ u(t) &= C_i^c x(t) + D_i^c y(t) \end{aligned}, \quad K_i = \left[\begin{array}{c|c} A_i^c & B_i^c \\ \hline C_i^c & D_i^c \end{array} \right] \tag{3.4}$$

The transfer function is unique from state-space representation:

$$K_i(s) = C_i^c(sI - A_i^c)^{-1}B_i^c + D_i^c = \frac{U_i(s)}{Y_i(s)} = \frac{b_{m_{K_i},i}s^{m_{K_i}} + b_{m_{K_i}-1,i}s^{m_{K_i}-1} + ... + b_{1,i}s + b_{0,i}}{s^{n_{K_i}} + a_{n_{K_i}-1,i}s^{n_{K_i}-1} + ... + a_{1,i}s + a_{0,i}} \tag{3.5}$$

where m_{K_i} and n_{K_i} are numerator and denominator orders corresponding to K_i.

Feedback control, also known as regulator problem, manipulates the system input to counteract the effect of disturbances. Connection details are found in Fig. 3.2. In other words, the objective is to regulate the output variables to zero in the presence of disturbances. The general controller K_i takes as input the measurement signals y to deliver a control input u, such $u = K_i y$.

Let's see the stability conditions when K_i is connected to G. The stability conditions will be later extended to the general plant P, being defined as LTI continous systems.

Theorem 3.2. *A necessary and sufficient condition to ensure stability of a feedback control loop with G is:*

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty \tag{3.6}$$

equivalently, all the poles corresponding to $(Is - K_iG)^{-1}$*,* $K_i(Is - GK_i)^{-1}$*,* $G(Is - K_iG)^{-1}$ *and* $(Is - GK_i)^{-1}$ *are in the left half-plane:* $s \in \mathbb{C} : Re(s) < 0$.

CL stability when K_i is connected to the general plant P is seen as bounded disturbances response e in function of the bounded disturbances input w:

$$e = f(P, K_i)w, \quad f(P, K_i) = G_{ew} + G_{eu}K_i(zI - GK_i)^{-1}G_{yw} \tag{3.7}$$

where $f(P, K)$ is a Linear-Fractional-Transformation (LFT) with respect to K_i.

Theorem 3.3. *A necessary and sufficient condition to ensure stability of a feedback controller* K_i *connected to a general plant* P *is:*

$$\begin{bmatrix} I & -\begin{bmatrix} 0 & 0 \\ 0 & K_i \end{bmatrix} \\ -P & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty \tag{3.8}$$

which is equivalent to:

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty, \quad \textit{with} \quad G_{ew}, \quad G_{eu} \quad \textit{and} \quad G_{yw} \in \mathcal{RH}_\infty \tag{3.9}$$

Notice how stability of the the general plant P is ensured by stability of G with K_i, and stable representations of external disturbances and performance requirements G_{ew}, G_{eu} and G_{yw}.

Let's consider the more general tracking case, which objective is to minimize the difference between output and a reference trajectory in the presence of disturbances. A feedforward/feedback controller as the one in Fig. 3.3 is used to determine stability conditions. r is the reference signal and K_i^f is the feedforward controller.

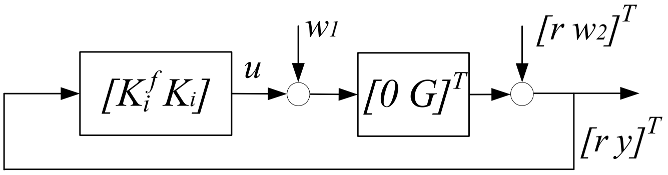


Figure 3.3: Feedforward/feedback controller connection.

Theorem 3.4. *A necessary and sufficient condition to ensure stability of a feedforward/feedback control loop with G is:*

$$\begin{bmatrix} I & -\begin{bmatrix} K_i^f & K_i \end{bmatrix} \\ -\begin{bmatrix} 0 \\ G \end{bmatrix} & \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \end{bmatrix}^{-1} \in RH_\infty \tag{3.10}$$

which is equivalent to: all the poles corresponding to $(Is - K_iG)^{-1}$, $(Is - K_iG)^{-1}K_i^f$, $K_i(Is - GK_i)^{-1}$, $G(Is - K_iG)^{-1}$, $G(Is - K_iG)^{-1}K_i^f$ and $(Is - GK_i)^{-1}$ are in the left half-plane: $z \in \mathbb{C}$: $Re(z) < 0$.

This can be also expressed:

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix} \in RH_\infty, \quad \begin{bmatrix} I \\ G \end{bmatrix} (I - K_iG)^{-1}K_i^f \in RH_\infty \tag{3.11}$$

From this Theorem, stability condition for feedforward/feedback control is equivalent to feedback control condition when K_f^i is stable. On the contrary, unstability related to K_i^f should be included in K_i. Stability condition for a general plant P connected to $[K_i^f \quad K_i]$ remains the same than in Theorem 3.3:

Theorem 3.5. *A necessary and sufficient condition to ensure stability of a feedforward/feedback controller $[K_i^f \quad K_i]$ connected to a general plant P is:*

$$\begin{bmatrix} I & -\begin{bmatrix} 0 & 0 \\ 0 & [K_i^f \quad K_i] \end{bmatrix} \\ -P & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty \tag{3.12}$$

which is equivalent to:

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty, \quad \textit{with} \quad K_i^f, \quad G_{ew}, \quad G_{eu} \quad \textit{and} \quad G_{yw} \in \mathcal{RH}_\infty \tag{3.13}$$

In short, if LTI continuous systems are considered and K_i^f, G_{ew}, G_{eu} and $G_{yw} \in \mathcal{RH}_\infty$, CL stability of P depends directly on the poles position of the feedback connection between K_i and G. Thus, feedback connection is considered as basis connection for doubly coprime factorization, YK controller reconfiguration and stability proof.

3.2 Doubly coprime factorization

A mandatory step towards the class of all stabilizing controllers for a given plant model is the doubly coprime factorization. In this section, guidelines are given to obtain doubly coprime factors for a specific plant model G connected to a stabilizing controller K_i.

For a LTI SISO continuous system, factorization leads to plant and controller being represented as the product of two scalar transfer functions. It could be also the product of two state-space matrix partitions as in Eqs. 3.2 or 3.4. Coprimeness refers to the absence of common zeros in the right half-plane: $z \in \mathbb{C} : Re(z) > 0$. For a LTI MIMO continuous system, factorization is represented as the ratio between a transfer function matrix with another transfer function matrix inversely stable. State-space matrix partition can be also used. Coprimeness is expressed as full rank condition in the right half-plane. When dealing with matrices, inversability could be right or left, and equivalently right and left coprime factorizations exist. Explicit definitions for a MIMO system [Nett et al., 1984], also applicable to SISO, are given below together with a numerical example.

Definition 3.3. *Two different matrices M_i and N_i are right coprimes over $\mathcal{RH}_\infty$ if they have the same number of columns and if matrices $X_{r,i}$ and $Y_{r,i}$ exist such that:*

$$\begin{bmatrix} X_{r,i} & Y_{r,i} \end{bmatrix} \begin{bmatrix} M_i \\ N_i \end{bmatrix} = X_{r,i}M_i + Y_{r,i}N_i = I \tag{3.14}$$

Definition 3.4. *Two different matrices $\tilde{M}_i$ and $\tilde{N}_i$ are left coprimes over $\mathcal{RH}_\infty$ if they have the same number of rows and if matrices $X_{l,i}$ and $Y_{l,i}$ exist such that:*

$$\begin{bmatrix} \tilde{M}_i & \tilde{N}_i \end{bmatrix} \begin{bmatrix} X_{l,i} \\ Y_{l,i} \end{bmatrix} = \tilde{M}_i X_{l,i} + \tilde{N}_i Y_{l,i} = I \tag{3.15}$$

Example 3.1. *For a LTI SISO continous system $\frac{2.5}{(s+2.5)}$, a coprime factorization could be $N_i = \frac{2.5}{(s+1)}$, $M_i = \frac{(s+2.5)}{(s+1)}$, $\tilde{N}_i = \frac{2.5(s+0.5)}{(s+2.5)(s+1)}$ and $\tilde{M}_i = \frac{(s+0.5)}{(s+0.1)}$, as functions $X_{r,i} = \frac{s}{(s+2.5)}$, $Y_{r,i} = 0.4$, $X_{l,i} = \frac{s}{(s+0.5)}$ and $Y_{l,i} = \frac{0.4(s+2.5)}{(s+0.5)}$ exist such Eqs. 3.14 and 3.15 are fulfilled.*

In order to obtain the class of all stabilizing controllers for G, these coprime factors should represent G and K_i such that:

$$\begin{aligned} G &= N_i M_i^{-1} = \tilde{M}_i^{-1} \tilde{N}_i \\ K_i &= U_i V_i^{-1} = \tilde{V}_i^{-1} \tilde{U}_i \end{aligned} \tag{3.16}$$

These coprime factors should be stables U_i, $\tilde{U}_i$, V_i, $\tilde{V}_i$, N_i, $\tilde{N}_i$, M_i, $\tilde{M}_i \in \mathcal{RH}_\infty$, and satisfy the double Bézout's identity [Pommaret and Quadrat, 1998]:

$$\begin{bmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{bmatrix} \begin{bmatrix} M_i & U_i \\ N_i & V_i \end{bmatrix} = \begin{bmatrix} M_i & U_i \\ N_i & V_i \end{bmatrix} \begin{bmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{3.17}$$

Double Bézout's identity is a relation between doubly coprime factors and stability condition for a feedback controller:

Theorem 3.6. *A necessary and sufficient condition to ensure stability of a feedback control loop between $K_i = U_i V_i^{-1} = \tilde{V}_i^{-1} \tilde{U}_i$ and $G = N_i M_i^{-1} = \tilde{M}_i^{-1} \tilde{N}_i$ with U_i, $\tilde{U}_i$, V_i, $\tilde{V}_i$, N_i, $\tilde{N}_i$, M_i, $\tilde{M}_i \in \mathcal{RH}_\infty$ is:*

$$\begin{aligned} \tilde{V}_i M_i - \tilde{U}_i N_i &= I \\ \tilde{M}_i V_i - \tilde{N}_i U_i &= I \end{aligned} \tag{3.18}$$

which is equivalent to the double Bézout's identity in Eq. 3.17.

Proof. [Tay et al., 1997] by substituting Eq. 3.16 in Eq. 3.6 with $K_i = \tilde{V_i}^{-1}\tilde{U_i}$ and $G = N_i M_i^{-1}$:

$$\begin{aligned}
(I - K_iG)^{-1} &= (I - \tilde{V_i}^{-1}\tilde{U_i}N_iM_i^{-1})^{-1} = M_i(\tilde{V_i}M_i - \tilde{U_i}N_i)^{-1}\tilde{V_i}\\
K_i(I - GK_i)^{-1} &= \tilde{V_i}^{-1}\tilde{U_i}(I - N_iM_i^{-1}\tilde{V_i}^{-1}\tilde{U_i})^{-1} = M_i(\tilde{V_i}M_i - \tilde{U_i}N_i)^{-1}\tilde{U_i}\\
G(I - K_iG)^{-1} &= N_iM_i^{-1}(I - \tilde{V_i}^{-1}\tilde{U_i}N_iM_i^{-1})^{-1} = N_i(\tilde{V_i}M_i - \tilde{U_i}N_i)^{-1}\tilde{V_i}\\
(I - GK_i)^{-1} = I + GK_i(I - GK_i)^{-1} &= I + N_iM_i^{-1}\tilde{V_i}^{-1}\tilde{U_i}(I - N_iM_i^{-1}\tilde{V_i}^{-1}\tilde{U_i})^{-1} =\\
&= I + N_i(\tilde{V_i}M_i - \tilde{U_i}N_i)^{-1}\tilde{U_i}
\end{aligned} \tag{3.19}$$

Thus:

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix}^{-1} = \begin{bmatrix} M_i \\ N_i \end{bmatrix} (\tilde{V_i}M_i - \tilde{U_i}N_i)^{-1} \begin{bmatrix} \tilde{V_i} & \tilde{U_i} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} \tag{3.20}$$

And by substituting Eq. 3.16 in Eq. 3.6 with $K_i = U_iV_i^{-1}$ and $G = \tilde{M_i}^{-1}\tilde{N_i}$:

$$\begin{aligned}
(I - K_iG)^{-1} &= (I - U_iV_i^{-1}\tilde{M_i}^{-1}\tilde{N_i})^{-1} = V_i(\tilde{M_i}V_i - \tilde{N_i}U_i)^{-1}\tilde{M_i}\\
K_i(I - GK_i)^{-1} &= U_iV_i^{-1}(I - \tilde{M_i}^{-1}\tilde{N_i}U_iV_i^{-1})^{-1} = U_i(\tilde{M_i}V_i - \tilde{N_i}U_i)^{-1}\tilde{M_i}\\
G(I - K_iG)^{-1} &= \tilde{M_i}^{-1}\tilde{N_i}(I - U_iV_i^{-1}\tilde{M_i}^{-1}\tilde{N_i})^{-1} = V_i(\tilde{M_i}V_i - \tilde{N_i}U_i)^{-1}\tilde{N_i}\\
(I - GK_i)^{-1} = I + GK_i(I - GK_i)^{-1} &= I + \tilde{M_i}^{-1}\tilde{N_i}U_iV_i^{-1}(I - \tilde{M_i}^{-1}\tilde{N_i}U_iV_i^{-1})^{-1} =\\
&= I + U_i(\tilde{M_i}V_i - \tilde{N_i}U_i)^{-1}\tilde{N_i}
\end{aligned} \tag{3.21}$$

Thus:

$$\begin{bmatrix} I & -K_i \\ -G & I \end{bmatrix}^{-1} = \begin{bmatrix} V_i \\ U_i \end{bmatrix} (\tilde{M_i}V_i - \tilde{N_i}U_i)^{-1} \begin{bmatrix} \tilde{M_i} & \tilde{N_i} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} \tag{3.22}$$

□

Doubly coprime factors extraction from a model G connected to a controller K_i has been studied in the literature. [Nett et al., 1984] presented a doubly coprime factorization for a LTI plant connected to a controller with a full observer-feedback form. From this work, doubly coprime factors related to reduced-order observer-based controllers were derived in [Telford and Moore, 1989], [Hippe, 1989] and [Fujimori, 1993]. In [Fujimori, 1993] some of the factors did not result stables, while [Telford and Moore, 1989] and [Hippe, 1989] provided stable parameters for a reduced-order observer-based controller. Finally, [Ishihara and Sales, 1999] and [Tay et al., 1997] extended the doubly coprime factorization to any stabilizing controller in state-space form. This is the doubly coprime factors extraction here explained.

Theorem 3.7. *Consider a plant in state space representation G as in Eq. 3.2 with A, B, C, D stabilizable and detectable, and a stabilizing controller K_i as in Eq. 3.4. F_i, F_i^c should be chosen such that $A + BF_i$ and $A_i^c + B_i^cF_i^c \in \mathcal{RH}_\infty$. Then doubly coprime factors are given by:*

$$\begin{bmatrix} M_i & U_i \\ N_i & V_i \end{bmatrix} = \left[\begin{array}{cc|cc} A + BF_i & 0 & B & 0 \\ 0 & A_i^c + B_i^cF_i^c & 0 & B_i^c \\ \hline F_i & C_i^c + D_i^cF_i^c & I & D_i^c \\ C + DF_i & F_i^c & D & I \end{array}\right] \tag{3.23}$$

$$\begin{bmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{bmatrix} = \left[\begin{array}{cc|cc} A + BY_iD_i^cC & BY_iC_i^c & -BY_i & BY_iD_i^c \\ B_i^cZ_iC & A_i^c + B_i^cZ_iDC_i^c & -B_i^cZ_iD & B_i^cZ_i \\ \hline F_i - Y_iD_i^cC & -C_i^c & I & -D_i^c \\ C & -F_i^c & 0 & I \end{array}\right] \tag{3.24}$$
$$\textit{with} \quad Y_i = (I - D_i^cD)^{-1} \quad \textit{and} \quad Z_i = (I - DD_i^c)^{-1}$$

Proof. For proof see [Ishihara and Sales, 1999]. □

In brief, the steps required to obtain the doubly coprime factors of a given feedback control loop are: 1) construct a model-controller state-space form realization, 2) solve a pole-assignment problem such $A + BF_i$ and $A_i^c + B_i^cF_i^c \in \mathbb{R}H_\infty$, and 3) perform some algebraic manipulations according to Eqs. 3.23 and 3.24.

3.3 All stabilizing controllers/Controller reconfiguration

Youla-Jabr-Bongiorno-Kucera parameterization provides all stabilizing controllers for a given plant G, by interconnecting a controller K with a parameter Q, called YK parameter, which can be any stable system with appropriate dimensions:

Theorem 3.8. *Consider a fixed plant G connected to a stabilizing controller K described by their coprime factors $G = NM^{-1} = \tilde{M}^{-1}\tilde{N}$ and $K = UV^{-1} = \tilde{V}^{-1}\tilde{U}$. Then, the set of all stabilizing controllers for G is described by:*

$$\begin{aligned} K(Q) = (U + MQ)(V + NQ)^{-1} = (\tilde{V} + Q\tilde{N})^{-1}(\tilde{U} + Q\tilde{M}) = \\ = K + \tilde{V}^{-1}Q(I + V^{-1}NQ)^{-1}V^{-1}, \quad Q \in \mathcal{RH}_\infty \end{aligned} \tag{3.25}$$

Proof. Let's see how this $K(Q)$ stabilizes the plant G, representing the class of all stabilizing controllers:

$$\begin{bmatrix} I & -K(Q) \\ -G & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty \tag{3.26}$$

$$\begin{aligned} \begin{bmatrix} I & -K(Q) \\ -G & I \end{bmatrix}^{-1} &= \begin{bmatrix} I & -(\tilde{V} + Q\tilde{N})^{-1}(\tilde{U} + Q\tilde{M}) \\ -\tilde{M}^{-1}\tilde{N} & I \end{bmatrix}^{-1} = \\ &= \left(\begin{bmatrix} (\tilde{V} + Q\tilde{N})^{-1} & 0 \\ 0 & \tilde{M}^{-1} \end{bmatrix} \begin{bmatrix} (\tilde{V} + Q\tilde{N}) & -(\tilde{U} + Q\tilde{M}) \\ -\tilde{N} & \tilde{M} \end{bmatrix} \right)^{-1} = \\ &= \begin{bmatrix} M & (U + MQ) \\ N & (V + NQ) \end{bmatrix} \begin{bmatrix} (\tilde{V} + Q\tilde{N}) & 0 \\ 0 & \tilde{M} \end{bmatrix} = \\ &= \left(\begin{bmatrix} M & U \\ N & V \end{bmatrix} + \begin{bmatrix} 0 & MQ \\ 0 & NQ \end{bmatrix} \right) \left(\begin{bmatrix} \tilde{V} & 0 \\ 0 & \tilde{M} \end{bmatrix} + \begin{bmatrix} Q\tilde{N} & 0 \\ 0 & 0 \end{bmatrix} \right) = \\ &= \begin{bmatrix} M & U \\ N & V \end{bmatrix} \begin{bmatrix} \tilde{V} & 0 \\ 0 & \tilde{M} \end{bmatrix} + \begin{bmatrix} MQ\tilde{N} & 0 \\ NQ\tilde{N} & 0 \end{bmatrix} + \begin{bmatrix} 0 & MQ\tilde{M} \\ 0 & NQ\tilde{M} \end{bmatrix} = \\ &= \begin{bmatrix} I & -K \\ -G & I \end{bmatrix}^{-1} + \begin{bmatrix} M \\ N \end{bmatrix} Q \begin{bmatrix} \tilde{N} & \tilde{M} \end{bmatrix} \end{aligned} \tag{3.27}$$

As coprime factors are stable by definition, CL stability between G and K is guaranteed, it is clear that any controller parameterized by $Q \in \mathcal{RH}_\infty$ will stabilize G, representing the class of all stabilizing controllers. As Q varies all over the stable space, all possible proper stabilizing controllers are provided by $K(Q)$. □

By using the class of all stabilizing controllers, it is possible to perform stable controller reconfiguration as mentioned in [Niemann and Stoustrup, 1999]. The main goal is to guarantee system stability whereas switching between controllers occurs. The reasons for the change could be numerous. For instance, new sensors or actuators have been added, or a better knowledge of the system has been obtained. Transitions between controllers are carried out through a scalar factor γ that affects the YK parameter Q.

How to obtain the YK parameter Q is detailed in the following subsections. Different Q's are considered depending on the number of controllers to be implemented. Stability is studied as a function of γ to carry out controller reconfiguration from an initial controller to a final one, or to several ones. The different mathematical expressions derive in different YK control structures for stable controller reconfiguration. These structures are shown and analysed in detail in section 3.3.3. The mathematical stability proof is given for each structure.

3.3.1 From a initial stabilizing controller to a final stabilizing controller

Consider that G is connected to an initial controller K_0. An arbitrary final controller K_1 is also chosen. YK makes possible stable controller reconfiguration betweem K_0 and K_1 online, just by choosing the appropriate Q [Niemann and Stoustrup, 1999]:

Theorem 3.9. *Let $G = N_0M_0^{-1} = \tilde{M_0}^{-1}\tilde{N_0} = N_1M_1^{-1} = \tilde{M_1}^{-1}\tilde{N_1}$ be a coprime factorization of the plant G and $K_0 = U_0V_0^{-1} = \tilde{V_0}^{-1}\tilde{U_0}$ an initial stabilizing controller represented by its coprime factors. A second controller is given by $K_1 = U_1V_1^{-1} = \tilde{V_1}^{-1}\tilde{U_1}$. Then, K_1 can be implemented in a stable way by calculating Q as follows:*

$$Q = X_1(\tilde{U_1}V_0 - \tilde{V_1}U_0) = X_1(\tilde{V_1}(K_1 - K_0)V_0) \tag{3.28}$$

with $X_1 = M_0^{-1}M_1$.

Proof. A controller K_1 is implemented as a stable Q parameter based on a another stabilizing controller K_0 by using Eq. 3.28 in Eq. 3.25 together with Bézout's identity in Eq. 3.17:

$$\begin{aligned}
K_0(Q) &= K_0 + \tilde{V_0}^{-1}Q(I + V_0^{-1}N_0Q)^{-1}V_0^{-1} = \\
&= K_0 + \tilde{V_0}^{-1}X_1\tilde{V_1}(K_1 - K_0)V_0(I + V_0^{-1}N_0X_1\tilde{V_1}(K_1 - K_0)V_0)^{-1}V_0^{-1} \\
K_0(Q) &= K_0 + \tilde{V_0}^{-1}X_1\tilde{V_1}(K_1 - K_0)V_0(V_0 + N_0X_1\tilde{V_1}(K_1 - K_0)V_0)^{-1} = \\
&= K_0 + \tilde{V_0}^{-1}X_1\tilde{V_1}(K_1 - K_0)(I + N_0X_1\tilde{V_1}(K_1 - K_0))^{-1} \\
K_0(Q) &= K_0 + \tilde{V_0}^{-1}X_1\tilde{V_1}(I + N_0\tilde{V_1}X_1(K_1 - K_0))^{-1}(K_1 - K_0) = \\
&= K_0 + \tilde{V_0}^{-1}X_1(\tilde{V_1}^{-1} + N_0X_1(K_1 - K_0))^{-1}(K_1 - K_0) \\
K_0(Q) &= K_0 + \tilde{V_0}^{-1}X_1(\tilde{V_1}^{-1} + N_0X_1\tilde{V_1}^{-1}\tilde{U_1} - N_0X_1\tilde{V_0}^{-1}\tilde{U_0})^{-1}(K_1 - K_0) = \\
&= K_0 + \tilde{V_0}^{-1}X_1(\tilde{V_1}^{-1}(I + \tilde{U_1}N_1) - \tilde{V_0}^{-1}\tilde{U_0}N_1)^{-1}(K_1 - K_0) \\
K_0(Q) &= K_0 + \tilde{V_0}^{-1}X_1(M_1 - \tilde{V_0}^{-1}\tilde{U_0}N_1)^{-1}(K_1 - K_0) = \\
&K_0 + (\tilde{V_0}M_0 - U_0N_0)^{-1}(K_1 - K_0) = K_0 + K_1 - K_0 = K_1
\end{aligned} \tag{3.29}$$

□

When switching from K_0 to K_1, the YK parameter Q can be scaled from 0 to 1 to smooth the switching between controllers. This scalar factor γ is included in Eq. 3.25 resulting in:

$$K_0(Q) = (U_0 + M_0\gamma Q)(V_0 + N_0\gamma Q)^{-1} = K_0 + \tilde{V_0}^{-1}\gamma Q(I + V_0^{-1}N_0\gamma Q)V_0^{-1} \tag{3.30}$$

A complete description of the controller $K_0(Q)$ depending on γ is given below.

$$\begin{aligned}
K_0(Q) = K_0 + \tilde{V_0}^{-1}\gamma Q(I + V_0^{-1}N_0\gamma Q)^{-1}V_0^{-1} = \\
= K_0 + \tilde{V_0}^{-1}\gamma X_1\tilde{V_1}(K_1 - K_0)V_0(I + V_0^{-1}N_0\gamma X_1\tilde{V_1}(K_1 - K_0)V_0)^{-1}V_0^{-1} \\
K_0(Q) = K_0 + \tilde{V_0}^{-1}\gamma X_1\tilde{V_1}(K_1 - K_0)V_0(V_0 + N_0\gamma X_1 V_1(K_1 - K_0)V_0)^{-1} = \\
= K_0 + \tilde{V_0}^{-1}\gamma X_1\tilde{V_1}(K_1 - K_0)(I + N_0\gamma X_1\tilde{V_1}(K_1 - K_0))^{-1} \\
K_0(Q) = K_0 + \tilde{V_0}^{-1}(I + N_0\gamma M_1 M_0^{-1}\tilde{V_1}(K_1 - K_0))^{-1}(\gamma M_1 M_0^{-1}\tilde{V_1}(K_1 - K_0)) = \\
= K_0 + \tilde{V_0}^{-1}(M_0 + N_0\gamma M_1\tilde{V_1}(K_1 - K_0))^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) = \\
= K_0 + (M_0\tilde{V_0} + N_0\gamma M_1\tilde{V_1}\tilde{V_0}(K_1 - K_0))^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) \\
K_0(Q) = K_0 + (M_0\tilde{V_0} + \gamma M_1(\tilde{U_1}N_0\tilde{V_0} - \tilde{U_0}N_0\tilde{V_1}))^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) = \\
= K_0 + ((I - \gamma M_1\tilde{V_1})M_0\tilde{V_0} + \gamma M_1\tilde{V_1} + \gamma M_1\tilde{U_1}N_0\tilde{V_0})^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) \\
K_0(Q) = K_0 + ((I - \gamma M_1\tilde{V_1})M_0\tilde{V_0} + \gamma M_1\tilde{V_1} + \gamma M_1\tilde{U_1}N_0\tilde{V_0})^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) = \\
= K_0 + ((I - \gamma M_1\tilde{V_1})M_0\tilde{V_0} + \gamma M_1\tilde{V_1} + \gamma\tilde{U_1}N_1M_0\tilde{V_0})^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) = \\
= K_0 + ((I - \gamma M_1\tilde{V_1} + \gamma\tilde{U_1}N_1)M_0\tilde{V_0} + \gamma M_1\tilde{V_1})^{-1}(\gamma M_1\tilde{V_1}(K_1 - K_0)) = \\
= \tilde{V_0}^{-1}\tilde{U_0} + ((I - \gamma)M_0\tilde{V_0} + \gamma M_1\tilde{V_1})^{-1}(\gamma M_1\tilde{V_1}(\tilde{V_1}^{-1}\tilde{U_1} - \tilde{V_0}^{-1}\tilde{U_0})) = \\
= ((I - \gamma)M_0\tilde{V_0} + \gamma M_1\tilde{V_1})^{-1}((1 - \gamma)M_0\tilde{U_0} + \gamma M_1\tilde{U_1})
\end{aligned} \tag{3.31}$$

$K_0(Q)$ description in Eq. 3.30 is connected to the general plant in Eq. 3.1, so CL stability from w to e can be studied [Tay et al., 1997]:

A necessary and sufficient condition to ensure stability is given in Theorem 3.3:

$$\begin{bmatrix} I & -\begin{bmatrix} 0 & 0 \\ 0 & K_0(Q) \end{bmatrix} \\ -P & I \end{bmatrix}^{-1} = G_{ew} + G_{eu}K_0(Q)(I - GK_0(Q))^{-1}G_{yw} \in \mathcal{RH}_\infty \tag{3.32}$$

Eq. 3.27 together with the CL description in Eq. 3.32 are used, yielding:

$$T_{ew} = G_{ew} + G_{eu}(M_0\tilde{U_0} + M_0\gamma Q\tilde{M_0})G_{yw} \tag{3.33}$$

where Q is given by Theorem 3.9. Bézout's identity is employed below, resulting:

$$
\begin{aligned}
T_{ew} = G_{ew} + G_{eu}(M_0\tilde{U}_0 + M_0\gamma M_0^{-1}M_1(\tilde{U}_1V_0 - \tilde{V}_1U_0)\tilde{M}_0)G_{yw} = \\
= G_{ew} + G_{eu}(M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1V_0\tilde{M}_0 - \gamma M_1\tilde{V}_1M_0\tilde{U}_0)G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma M_1\tilde{V}_1)M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1V_0\tilde{M}_0)G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma(I+\tilde{U}_1N_1))M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1V_0\tilde{M}_0)G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 - \gamma\tilde{U}_1N_1M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1V_0\tilde{M}_0)G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 - \gamma\tilde{U}_1(N_1M_0\tilde{U}_0 - M_1V_0\tilde{M}_0))G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 - \gamma\tilde{U}_1(N_0M_1\tilde{U}_0 - M_1V_0\tilde{M}_0))G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 - \gamma\tilde{U}_1(M_1(N_0\tilde{U}_0 - V_0\tilde{M}_0)))G_{yw} = \\
= G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1)G_{yw}
\end{aligned}
\tag{3.34}
$$

This proves that CL stability of a general plant connected to $K_0(Q)$ depends on coprime factors M_0, $\tilde{U}_0$, M_1 and $\tilde{U}_1$. These coprime factors are stable by definition, so CL stability is ensured for every value of γ.

Theorem 3.10 (Stable controller reconfiguration between K_0 and K_1)**.** *CL transfer function of a general plant connected to a feedback controller is in Eq. 3.7. Consider a controller $K_0(Q)$ given by:*

$$K_0(Q) = (U_0 + M_0\gamma Q)(V_0 + N_0\gamma Q)^{-1} = (\tilde{V}_0 + \gamma Q\tilde{N}_0)^{-1}(\tilde{U}_0 + \gamma Q\tilde{M}_0) \tag{3.35}$$

with $\gamma \in [0,1]$. Then, the CL transfer function T_{ew} depending on γ is given by:

$$T_{ew} = G_{ew} + G_{eu}((1-\gamma)M_0\tilde{U}_0 + \gamma M_1\tilde{U}_1)G_{yw} \tag{3.36}$$

CL stability is ensured for every value of γ if and only if G_{ew}, G_{eu}, G_{yw}, M_0, $\tilde{U}_0$, M_1 and $\tilde{U}_1 \in \mathcal{RH}_\infty$. As they are stable by definition, stability when doing controller reconfiguration between K_0 and K_1 is ensured.

Proof. Proof is above. □

The different time-variations of γ result in different transients when switching between controllers. CL stability analysis is extended to any time-variation of γ in [Hespanha and Morse, 2002]. It is stated that the switching signal γ between both controllers could be simpleminded, because stability is ensured in every moment. As example, two controllers are designed for the control of the roll angle of an aircraft: the first, slow but with good noise rejection; and the second, fast but sensitive to noise. By switching between them is possible to achieve good noise rejection when noise is large, and fast response when noise is small. Stability is ensured regardless the algorithm used to obtain the switching signal. Thus, with the YK parameterization, one can use a simpleminded algorithm to switch between controllers whereas keeping stability.

3.3.2 From a initial stabilizing controller to several stabilizing controllers

Former sections considered the switching between two possible controllers. However, applications where more controllers are required could exist: K_i with $i \in [1,p]$. K_0 is the initial controller. Further, each of these controllers K_i should be implemented with the appropriate Q_i. Theorem 3.9 is rewritten in a more general way:

Theorem 3.11. *Let $G = N_i M_i^{-1} = \tilde{M_i}^{-1}\tilde{N_i}$ be a coprime factorization of the plant G and $K_0 = U_0 V_0^{-1} = \tilde{V_0}^{-1}\tilde{U_0}$ an initial stabilizing controller represented by its coprime factors. p stabilizing controllers are given by $K_i = U_i V_i^{-1} = \tilde{V_i}^{-1}\tilde{U_i}$ with $i \in [1, p]$. Then, K_i can be implemented in a stable way by calculating Q_i as follows:*

$$Q_i = X_i(\tilde{U_i}V_0 - \tilde{V_i}U_0) = X_i(\tilde{V_i}(K_i - K_0)V_0) \tag{3.37}$$

with $X_i = M_0^{-1}M_i$.

Proof. Proof is equivalent to the one in Theorem 3.9. Replace subindex $_1$ by $_i$. □

Different scalar factors γ_i are employed for providing smoothness when doing controller reconfiguration from K_0 to K_i. Moreover, they can be also used to provide a combination of different stabilizing controllers such Q yields:

$$Q = \sum_{i=1}^{p} \gamma_i Q_i \quad with \quad \sum_{i=1}^{p} \gamma_i = 1 \tag{3.38}$$

The number of controllers is $p = 2$, both different from the initial controller K_0. This simplifies the controller description without loss of generality:

$$\begin{aligned} Q = \gamma_1 Q_1 + \gamma_2 Q_2 = \gamma_1 X_1(\tilde{U_1}V_0 - \tilde{V_1}U_0) + \gamma_2 X_2(\tilde{U_2}V_0 - \tilde{V_2}U_0) = \\ = \gamma_1 X_1 \tilde{V_1}(K_1 - K_0)V_0 + \gamma_2 X_2 \tilde{V_2}(K_2 - K_0)V_0 \quad with \quad \gamma_1 + \gamma_2 = 1 \end{aligned} \tag{3.39}$$

This new Q is included into Eq. 3.25 for having a complete description of $K(Q)$ able to switch between several controllers:

$$\begin{aligned}
K(Q) = K_0 + \tilde{V_0}^{-1} Q(I + V_0^{-1} N_0 Q)^{-1} V_0^{-1} = \\
= K_0 + \tilde{V_0}^{-1}(\gamma_1 X_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 X_2 \tilde{V}_2(K_2 - K_0)) \\
(I + N_0(\gamma_1 X_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 X_2 \tilde{V}_2(K_2 - K_0)))^{-1} = \\
= K_0 + \tilde{V_0}^{-1}(I + (\gamma_1 X_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 X_2 \tilde{V}_2(K_2 - K_0))N_0)^{-1} \\
(\gamma_1 X_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 X_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + \tilde{V_0}^{-1}(M_0 + (\gamma_1 M_1(\tilde{U}_1 - \tilde{V}_1 K_0) + \gamma_2 M_2(\tilde{U}_2 - \tilde{V}_2 K_0))N_0)^{-1} \\
(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + (M_0 \tilde{V}_0 + (\gamma_1 M_1(\tilde{U}_1 - \tilde{V}_1 K_0) + \gamma_2 M_2(\tilde{U}_2 - \tilde{V}_2 K_0))N_0 \tilde{V}_0)^{-1} \\
(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + (M_0 \tilde{V}_0 + \gamma_1 M_1(\tilde{U}_1 N_0 \tilde{V}_0 - \tilde{V}_1 U_0 \tilde{N}_0) + \gamma_2 M_2(\tilde{U}_2 N_0 \tilde{V}_0 - \tilde{V}_2 U_0 \tilde{N}_0))^{-1} \\
(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + (\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2 + (I - \gamma_1 M_1 \tilde{V}_1 - \gamma_2 M_2 \tilde{V}_2) M_0 \tilde{V}_0 + (\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2) N_0 \tilde{V}_0)^{-1} \\
(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + (\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2 + \gamma_1 \tilde{U}_1(\tilde{M}_1 N_0 - \tilde{N}_1 M_0)\tilde{V}_0 + \gamma_2 \tilde{U}_2(\tilde{M}_2 N_0 - \tilde{N}_2 M_0)\tilde{V}_0)^{-1} \\
(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= K_0 + (\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2)^{-1}(\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)) = \\
= (K_0(\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2) + (\gamma_1 M_1 \tilde{V}_1(K_1 - K_0) + \gamma_2 M_2 \tilde{V}_2(K_2 - K_0)))(\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2)^{-1} = \\
= (\gamma_1 M_1 \tilde{V}_1 + \gamma_2 M_2 \tilde{V}_2)^{-1}(\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2)
\end{aligned} \tag{3.40}$$

Without loss of generality, the solution for $p = 2$ can be extended to p controllers [Niemann et al., 2004]:

$$K(Q) = \left(\sum_{i=1}^{p} \gamma_i M_i \tilde{V}_i \right)^{-1} \sum_{i=1}^{p} \gamma_i M_i \tilde{U}_i \quad \mathit{with} \quad \sum_{i=1}^{p} \gamma_i = 1 \tag{3.41}$$

Notice how $K(Q)$ is independent from K_0. The reason is that the scaling parameters γ_i satisfy $\sum_{i=1}^{p} \gamma_i = 1$. However, if the sum is not equal to one, the final controller will be also function of the initial controller K_0.

Once a general description of $K(Q)$ is given, CL stability from w to e is analysed. T_{ew} for $p = 2$ is equivalent to:

$$\begin{aligned}
T_{ew} = G_{ew} + G_{eu}(M_0 \tilde{U}_0 + \gamma_1 M_1(\tilde{U}_1 V_0 - \tilde{V}_1 U_0)\tilde{M}_0 + \gamma_2 M_2(\tilde{U}_2 V_0 - \tilde{V}_2 U_0)\tilde{M}_0) G_{yw} = \\
= G_{ew} + G_{eu}((1 - \gamma_1 M_1 \tilde{V}_1 - \gamma_2 M_2 \tilde{V}_2) M_0 \tilde{U}_0 + \gamma_1 M_1 \tilde{U}_1 V_0 \tilde{M}_0 + \gamma_2 M_2 \tilde{U}_2 V_0 \tilde{M}_0) G_{yw} = \\
= G_{ew} + G_{eu}((\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2) + (\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2) N_0 \tilde{U}_0 - (\gamma_1 U_1 \tilde{N}_1 + \gamma_2 U_2 \tilde{N}_2) M_0 \tilde{U}_0) G_{yw} = \\
= G_{ew} + G_{eu}((\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2) + \gamma_1 U_1(\tilde{M}_1 N_0 - \tilde{N}_1 M_0)\tilde{U}_0 + \gamma_2 U_2(\tilde{M}_2 N_0 - \tilde{N}_2 M_0)\tilde{U}_0) G_{yw} = \\
= G_{ew} + G_{eu}((\gamma_1 M_1 \tilde{U}_1 + \gamma_2 M_2 \tilde{U}_2)) G_{yw}
\end{aligned} \tag{3.42}$$

This is again extended to p controllers without loss of generality:

$$T_{ew} = G_{ew} + G_{eu}\left(\sum_{i=1}^{p}\gamma_i M_i\tilde{U_i}\right)G_{yw} \quad with \quad \sum_{i=1}^{p}\gamma_i = 1 \tag{3.43}$$

Therefore, CL stability depends directly on the coprime factors M_i and $\tilde{U_i}$, which are stable by definition. The information above is summarized in the following Theorem:

Theorem 3.12 (Stable controller reconfiguration between several controllers)**.** *CL transfer function of a general plant connected to a feedback controller is in Eq. 3.7. Consider a controller $K(Q)$ given by:*

$$K(Q) = (U_0 + M_0\sum_{i=1}^{p}\gamma_i Q_i)(V_0 + N_0\sum_{i=1}^{p}\gamma_i Q_i)^{-1} = (\tilde{V_0} + \sum_{i=1}^{p}\gamma_i Q_i\tilde{N_0})^{-1}(\tilde{U_0} + \sum_{i=1}^{p}\gamma_i Q_i\tilde{M_0}) \tag{3.44}$$

with $\sum_{i=1}^{p}\gamma_i = 1$. Then, the CL transfer function T_{ew} depending on γ_i is given by:

$$T_{ew} = G_{ew} + G_{eu}\left(\sum_{i=1}^{p}\gamma_i M_i\tilde{U_i}\right)G_{yw} \tag{3.45}$$

CL stability is ensured for every value of γ_i if and only if G_{ew}, G_{eu}, G_{yw}, M_i and $\tilde{U_i} \in \mathcal{RH}_\infty$. As they are stable by definition, stability when doing controller reconfiguration between several controllers K_i is ensured.

Proof. Proof is above. □

3.3.3 Controller structures

This subsection explores the different control structures in connection with YK parameterization to provide stable controller reconfiguration. Some drawbacks come up when using the YK architecture: complexity of the resulting controller, matrix inversability or disconnection of the plant for implementation. The standard structure in [Moore et al., 1990] has been modified in the literature to avoid matrix inversability [Niemann and Poulsen, 2009a], plant disconnection [Trangbaek et al., 2008] [Trangbaek and Bendtsen, 2009], or to reduce the complexity of the resulting controller [Niemann and Poulsen, 2009a]. These structures are summarized, highlighting the solved problems, and providing mathematical proof that stability remains. For the sake of simplicity, control structures shown in this subsection are for controller reconfiguration between K_0 and K_1. It could be extended to several stabilizing controllers K_i without losing generality.

3.3.3.a Structures 1 and 2

From Theorem 3.10, the standard control structures for stable controller reconfiguration are derived. Figure 3.4 shows the control structure for switching based on right coprime factors. On the other hand, the structure in Fig. 3.5 depends on left coprime factors. Standard structures refer to the first YK control structures that appeared in [Moore et al., 1990]. The complexity of these control structures is defined as the order/state dimension of the switched controller $K_0(Q)$: A non-minimal realization of structures 1 and 2 may yield a $K_0(Q)$ with $7(n + n_{K_0}) + 3(n + n_{K_1})$ states.

When doing controller transitions, the scalar factor γ plays a key role. It regulates the different levels of activation of the YK parameter Q. γ may vary from 0 to 1, being 0 a 100% contribution of K_0 and 1 a 100% contribution of K_1. In short, γ is the switching signal between K_0 and K_1.

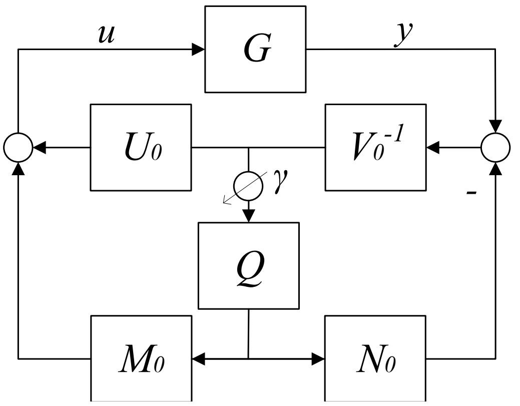


Figure 3.4: Structure 1.

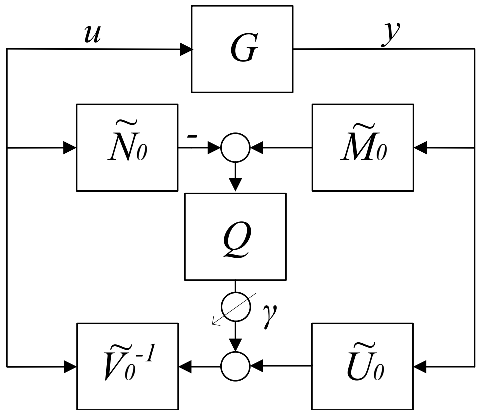


Figure 3.5: Structure 2.

Both implementations present some drawbacks such high order complexity of resulting controller, matrix inversability or controller design from scratch. These structures are modified below, so the associated problems can be suppressed.

3.3.3.b Structures 3 and 4

Controller reconfiguration using structures 1 and 2 requires the initial controller K_0 to be divided in its coprime factors U_0, V_0 or $\tilde{U}_0$ and $\tilde{V}_0$. Even if the system is already operational with an initial controller K_0, this one should be disconnected. This is unfeasible if the system shutdown is very expensive, or the initial controller is part of a safety circuit. Remaining the original controller would be useful for not replicating supervisory logic already installed in the initial controller, or to come back to the previous control system when failures occur. In several cases, the inside of the controller could not be accessible.

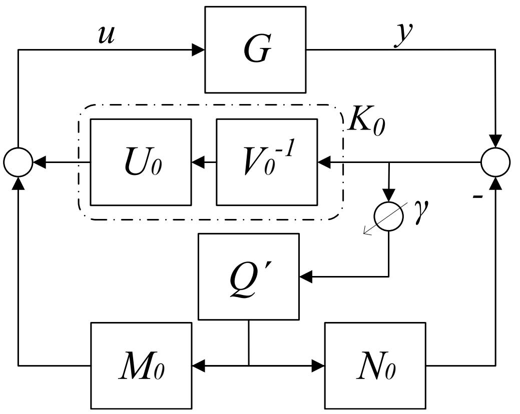


Figure 3.6: Structure 3.

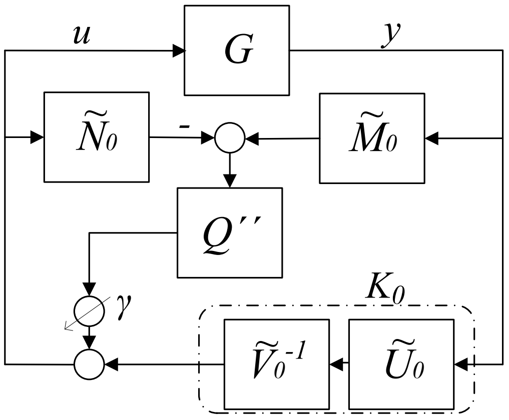


Figure 3.7: Structure 4.

[Trangbaek and Bendtsen, 2009] keep the initial controller in place, accessing at its terminal to carry out the YK controller reconfiguration. New control dynamics can be added online, without removing the original controller. This allows to return to the original controller in case of problems with the new one. Control structures for right and left coprime factorizations are shown in Figs. 3.6 and 3.7 respectively. As K_0 is not decomposed in coprime factors, inversion of matrices is no longer needed; and the complexity of the resulting $K_0(Q)$ is lower: $5(n + n_{K_0}) + 3(n + n_{K_1}) + n_{K_0}$ for structure 3 and $6(n + n_{K_0}) + 3(n + n_{K_1}) + n_{K_0}$ for structure 4. Notice that calculations of the YK parameters Q' and Q'' differ from the standard one in Eq. 3.28:

Theorem 3.13. *Let* $G = N_0M_0^{-1} = \tilde{M}_0^{-1}\tilde{N}_0 = N_1M_1^{-1} = \tilde{M}_1^{-1}\tilde{N}_1$ *be a coprime factorization of the plant* G *and* $K_0 = U_0V_0^{-1} = \tilde{V}_0^{-1}\tilde{U}_0$ *an initial stabilizing controller that can not be disconnected. A second controller is given by* $K_1 = U_1V_1^{-1} = \tilde{V}_1^{-1}\tilde{U}_1$*. In order to use structures 3 and 4:*

When using right coprime factors M_0 *and* N_0*,* Q' *is calculated as:*

$$Q' = QV_0^{-1} = X_1(\tilde{U}_1 - \tilde{V}_1\tilde{U}_0\tilde{V}_0^{-1}) \tag{3.46}$$

When using left coprime factors $\tilde{M}_0$ *and* $\tilde{N}_0$*,* Q'' *is calculated as:*

$$Q'' = Q\tilde{V}_0^{-1} = X_1(\tilde{V}_0^{-1}(\tilde{U}_1V_0 - \tilde{V}_1U_0)) \tag{3.47}$$

with $X_1 = M_0^{-1}M_1$*.*

Proof. Proof is equivalent to stability proof below. □

Once structures 3 and 4 are defined, the YK property of stable controller reconfiguration is verified. Stability has already been demonstrated in Theorem 3.10. To prove that stability is still preserved in these new structures, it is only necessary to check that $K_0(Q)$ remains as in Eq. 3.35.

According to the block diagram of structure 3 (Fig. 3.6), $K_0(Q)$ yields:

$$\begin{aligned} u &= (K_0 + M_0\gamma Q')(1 + N_0\gamma Q')^{-1}y \\ u &= (U_0V_0^{-1} + M_0\gamma Q')(1 + N_0\gamma Q')^{-1}y \\ u &= (U_0V_0^{-1} + M_0\gamma QV_0^{-1})(1 + N_0\gamma QV_0^{-1})^{-1}y \\ u &= (U_0 + M_0\gamma Q)(V_0 + N_0\gamma Q)^{-1}y \end{aligned} \tag{3.48}$$

which is equivalent to $K_0(Q)$ description in Eq. 3.35 for right coprime factors.

According to the block diagram of structure 4 (Fig. 3.7), $K_0(Q)$ yields:

$$\begin{aligned} u &= (K_0 + \gamma Q''\tilde{M}_0)(1 + \gamma Q''\tilde{N}_0)^{-1}y \\ u &= (\tilde{V}_0^{-1}\tilde{U}_0 + \gamma Q''\tilde{M}_0)(1 + \gamma Q''\tilde{N}_0)^{-1}y \\ u &= (\tilde{V}_0^{-1}\tilde{U}_0 + \gamma Q\tilde{V}_0^{-1}\tilde{M}_0)(1 + \gamma Q''\tilde{N}_0)^{-1}y \\ u &= (\tilde{U}_0 + \gamma Q\tilde{M}_0)(\tilde{V}_0 + \gamma Q\tilde{N}_0)^{-1}y \end{aligned} \tag{3.49}$$

which is equivalent to $K(Q)$ description in Eq. 3.35 for left coprime factors.

Stable controller reconfiguration property is ensured even if different structures are employed.

3.3.3.c Structures 5 and 6

Another critical point in the implementation of structures 1 and 2 is the inversion of coprime factors V_0 and $\tilde{V}_0$. As a solution, two new structures related to loop transfer recovery were proposed in [Niemann et al., 1991] and [Niemann and Poulsen, 2009a]. Structures so-called 5 and 6 do not present matrix inversion. Block diagrams of both structures are depicted in Figs. 3.8 and 3.9 respectively. Notice that unlike structures 3 and 4 the calculation of the YK parameter Q remains as in Eq. 3.28, and the initial controller K_0 should be disconnected for structure implementation. Besides, the resulting controller order is the same as structures 1 and 2: $7(n + n_{K_0}) + 3(n + n_{K_1})$.

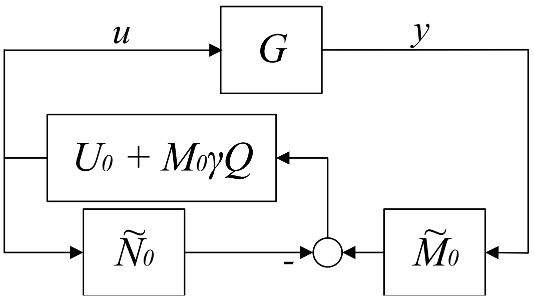


Figure 3.8: Structure 5.

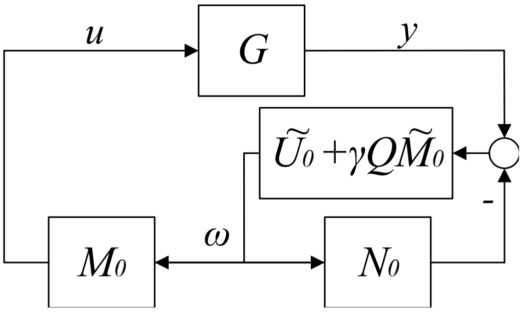


Figure 3.9: Structure 6.

Again stable controller reconfiguration property is checked with Eq. 3.35. According to the block diagram of structure 5 (Fig. 3.8), $K_0(Q)$ yields:

$$\begin{aligned} u &= (U_0 + M_0\gamma Q)(\tilde{M}_0 y - \tilde{N}_0 u) = \\ u &= (I + (U_0 + M_0\gamma Q)\tilde{N}_0)^{-1}(U_0 + M_0\gamma Q)\tilde{M}_0 y \\ u &= (\tilde{M}_0 V_0 + \tilde{M}_0 N_0\gamma Q)^{-1}(U_0 + M_0\gamma Q)\tilde{M}_0 y \\ u &= (V_0 + N_0\gamma Q)^{-1}(U_0 + M_0\gamma Q)y \end{aligned} \tag{3.50}$$

which is equivalent to $K(Q)$ description in Eq. 3.35 for right coprime factors.

According to the block diagram of structure 6 (Fig. 3.9), $K_0(Q)$ yields:

$$\begin{aligned} u &= M_0\omega \\ u &= M_0(\tilde{U}_0 + \gamma Q\tilde{M}_0)(1 + (\tilde{U}_0 + \gamma Q\tilde{M}_0)N_0)^{-1}y \\ u &= M_0(\tilde{U}_0 + \gamma Q\tilde{M}_0)(\tilde{V}_0 M_0 + \gamma Q\tilde{N}_0 M_0)^{-1}y \\ u &= (\tilde{U}_0 + \gamma Q\tilde{M}_0)(\tilde{V}_0 + \gamma Q\tilde{N}_0)^{-1}y \end{aligned} \tag{3.51}$$

which is equivalent to $K_0(Q)$ description in Eq. 3.35 for left coprime factors.

3.3.3.d Structures 7 and 8

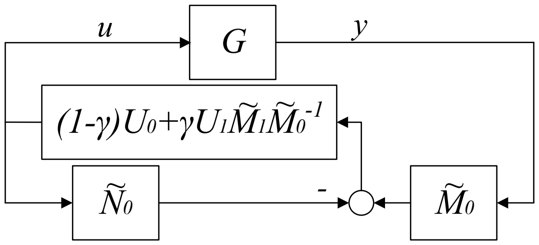


Figure 3.10: Structure 7

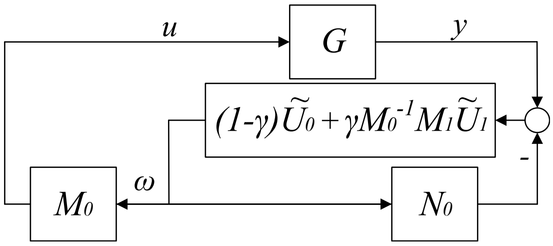


Figure 3.11: Structure 8

Finally, the work in [Niemann and Poulsen, 2009a] deals with the reduction of the implementation complexity of the YK parameter Q. Notice that this parameter is derived from Eq. 3.28, which depends on six coprime factors. Structures 7 and 8 make the YK controller reconfiguration independent of Q. Both control structures are shown in Figs. 3.10 and 3.11 respectively. The equivalent complexity order results in $4(n+n_{K_0})+2(n+n_{K_1})$ states. Again structure implementation requires the initial controller to be disconnected. Notice how for the special case where $X_1 = I$, the complexity order gets even lower.

Stable controller reconfiguration property is still preserved even if Q is no longer in the structure for controller reconfiguration. Mathematical proof is shown below:

According to the block diagram of structure 7 (Fig. 3.10), $K_0(Q)$ yields:

$$\begin{aligned}
u &= ((1-\gamma)U_0 + \gamma U_1 \tilde{M}_1 \tilde{M_0}^{-1})(\tilde{M}_0 y - \tilde{N}_0)\\
u &= ((1-\gamma)U_0 + \gamma U_1 \tilde{M}_1(-GU_0 + V_0))(\tilde{M}_0 y - \tilde{N}_0)\\
u &= ((1-\gamma)U_0 + \gamma(-U_1\tilde{N}_1 U_0 + U_1\tilde{M}_1 V_0))(\tilde{M}_0 y - \tilde{N}_0)\\
u &= ((1-\gamma)U_0 + \gamma((I - M_1\tilde{V}_1)U_1 + M_1\tilde{U}_1 V_0))(\tilde{M}_0 y - \tilde{N}_0)\\
u &= (U_0 + \gamma M_1(\tilde{U}_1 V_0 - \tilde{V}_1 U_0))(\tilde{M}_0 y - \tilde{N}_0)\\
u &= (U_0 + M_0\gamma Q)(\tilde{M}_0 y - \tilde{N}_0)\\
u &= (I + (U_0 + M_0\gamma Q)\tilde{N}_0)^{-1}(U_0 + M_0\gamma Q)\tilde{M}_0 y\\
u &= (\tilde{M}_0 V_0 + \tilde{M}_0 N_0\gamma Q)^{-1}(U_0 + M_0\gamma Q)\tilde{M}_0 y\\
u &= (V_0 + N_0\gamma Q)^{-1}(U_0 + M_0\gamma Q)y
\end{aligned} \tag{3.52}$$

which is equivalent to $K_0(Q)$ description in Eq. 3.35 for right coprime factors.

According to the block diagram of structure 8 (Fig. 3.11), $K_0(Q)$ yields:

$$\begin{aligned}
u &= M_0\omega\\
u &= M_0((1-\gamma)\tilde{U}_0 + \gamma M_0^{-1}M_1\tilde{U}_1)\\
&(1 + ((1-\gamma)\tilde{U}_0 + \gamma M_0^{-1}M_1\tilde{U}_1)N_0)^{-1}y\\
u &= M_0(\tilde{U}_0 + \gamma Q\tilde{M}_0)(1 + (\tilde{U}_0 + \gamma Q\tilde{M}_0)N_0)^{-1}y\\
u &= M_0(\tilde{U}_0 + \gamma Q\tilde{M}_0)(\tilde{V}_0 M_0 + \gamma Q\tilde{N}_0 M_0)^{-1}y\\
u &= (\tilde{U}_0 + \gamma Q\tilde{M}_0)(\tilde{V}_0 + \gamma Q\tilde{N}_0)^{-1}y
\end{aligned} \tag{3.53}$$

which is equivalent to $K(Q)$ description in Eq. 3.35 for left coprime factors.

On the other hand, it has been assumed that there is no variation in the dynamic system represented by G; when doing controller switching in a system with variations, dual YK formulation needs to be employed. Further details are found in Chapter 4. All the plants stabilized by a given controller are represented by $G(S)$, where S is the dual YK parameter [Niemann, 2003]. In that case, CL stability involves both Q and S: $(I - QS)^{-1} \in \mathbb{R}H_\infty$ [Niemann and Stoustrup, 1999]. Thus, control structures 7 and 8 cannot be used when system variations are present, as Q is no longer in the structure.

3.4 Numerical examples

In this section, different numerical examples are given for the better understading of the stable controller reconfiguration provided by the YK parameterization. In particular, it is shown how YK is able to solve a transition problem where a direct linear change between two controllers could result in a stability problem. Root locus representation during the transition phase between initial and final first-order controllers connected to a first-order system is also provided. It allows to make a connection between YK stability property and zeros/poles position when doing controller reconfiguration. Finally, transient behavior depending on the control structure is also analysed. A comparative study is carried out among all the structures, determining if the use of some YK control structure improves transient performance when doing controller switching.

3.4.1 Stable transition

A linear combination between two controllers K_0 and K_1 could results in an unstable system. In [Niemann et al., 2004], a theoretical example shows how the linear combination of two stabilizing controllers does not result in a stabilizing controller of the system. The example is here reused to show the stabilizing capabilities of the YK parameterization.

Consider the following state-space representation of a system G:

$$G = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{ccc|c} 7.0 & 0.0 & 0.0 & 1.0 \\ 1.0 & -7.0 & -2.4495 & 0.0 \\ 0.0 & 2.4495 & 0.0 & 0.0 \\ \hline 1.0 & -5.0 & 253.1139 & 0.0 \end{array}\right] \tag{3.54}$$

The system is unstable, but it is stabilized by an initial controller K_0:

$$K_0 = \left[\begin{array}{c|c} A_0^c & B_0^c \\ \hline C_0^c & D_0^c \end{array}\right] = \left[\begin{array}{c|c} 0.0 & 0.0 \\ \hline 0.0 & 1000 \end{array}\right] \tag{3.55}$$

such that the following stable CL poles remain:

$$poles_{CL(G,K_0)} = \begin{bmatrix} -998.67 \\ -0.6660 \pm 25.027i \end{bmatrix} \tag{3.56}$$

Later, the controller is replaced by a more advanced one. It could be the case where the initial controller is applied in the start-up process of the system, but later replaced with a more advanced controller. An observer-based feedback controller K_1 is considered:

$$K_1 = \left[\begin{array}{c|c} A_1^c & B_1^c \\ \hline C_1^c & D_1^c \end{array}\right] = \left[\begin{array}{ccc|c} -15.070 & 45.992 & -2309.7 & 9.1283 \\ 0.3537 & -3.7679 & -166.07 & 0.64643 \\ -0.13121 & 3.1056 & -33.212 & 0.13121 \\ \hline -12.941 & 0.35054 & 0.85619 & 0.0 \end{array}\right] \tag{3.57}$$

which CL poles result:

$$poles_{CL(G,K_1)} = \begin{bmatrix} -25.1218 \\ -0.9022 \\ -7.7082 \pm 1.01005i \\ -5.3047 \pm 1.1643i \end{bmatrix} \tag{3.58}$$

A direct linear change between controllers K_0 and K_1 is considered. Let the linear combination between both controllers be as follows:

$$K_\alpha = (1-\alpha)K_0 + \alpha K_1, \quad with \quad \alpha \in [0,1] \tag{3.59}$$

It results that CL stability is not ensured for every value of $\alpha \in [0,1]$. CL poles are shown for different values of α in Table 3.1. Notice how for $\alpha \in [0.7, 0.9]$ there are poles in the right half-plane, and therefore the response is unstable. A direct linear change between two controllers

Table 3.1: CL poles (G, K_α). Direct linear change between K_0 and K_1.

α	CL poles
$\alpha = 0.0$	$[-998.67, -0.6660 \pm 25.027i]$
$\alpha = 0.1$	$[-8.7288 \pm 14.0935i, -0.6288 \pm 25.0287i, -34.6057, -898.7287]$
$\alpha = 0.2$	$[-8.7279 \pm 14.0891i, -0.5825 \pm 25.0304i, -34.6244, -798.8044]$
$\alpha = 0.3$	$[-8.7268 \pm 14.0835i, -0.5229 \pm 25.0320i, -34.6486, -698.9015]$
$\alpha = 0.4$	$[-8.7254 \pm 14.0760i, -0.4436 \pm 25.0332i, -34.6813, -599.0303]$
$\alpha = 0.5$	$[-8.7234 \pm 14.0655i, -0.3328 \pm 25.0328i, -34.7276, -499.2095]$
$\alpha = 0.6$	$[-8.7205 \pm 14.0500i, -0.1673 \pm 25.0278i, -34.7986, -399.4754]$
$\alpha = 0.7$	$[-8.7157 \pm 14.0245i, 0.1058 \pm 25.0083i, -34.9211, -299.9088]$
$\alpha = 0.8$	$[-8.7065 \pm 13.9748i, 0.6380 \pm 24.9291i, -35.1835, -200.7292]$
$\alpha = 0.9$	$[-8.6818 \pm 13.8348i, 2.0578 \pm 24.4344i, -36.1582, -102.6435]$
$\alpha = 1.0$	$[-25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$

could result in a stability problem. This can be more critical in the case that one wants to change between more than two controllers.

When using any of the control structures in subsection 3.3.3 stable transition is ensured between controllers K_0 and K_1. G, K_0 and K_1 are defined as in Eqs. 3.54, 3.55 and 3.56 and used within Theorem 3.7 for obtaining doubly coprime factors N_0, M_0, N_1, M_1, $\tilde{N}_0$, $\tilde{M}_0$, $\tilde{N}_1$, $\tilde{M}_1$, U_0, V_0, U_1, V_1, $\tilde{U}_0$, $\tilde{V}_0$, $\tilde{U}_1$, $\tilde{V}_1$. These factors are used for the calculation of Q so the stable YK controller reconfiguration can be implemented (see Theorem 3.9). CL poles are shown for different values of γ in Table 3.2. YK paramaterization is able to stabilize a transition between controllers where direct linear change results unstable. Independently of the value of γ, CL poles during the transition are the combination of CL poles of (G, K_0) and (G, K_1) [Niemann and Stoustrup, 1999]. Poles do not change during the controller transition. This property is detailed in the subsection below.

3.4.2 Root locus evaluation

In this subsection, a root locus comparison between direct linear change between controllers and YK controller reconfiguration is given. For the shake of clarity, a first order plant is used for the stability analysis.

Consider the following state space representation of a system:

$$G = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{c|c} -2.5 & 2.0 \\ \hline 1.25 & 0.0 \end{array}\right] \tag{3.60}$$

A controller K_0 was designed to make the system follow a step reference:

$$K_0 = \left[\begin{array}{c|c} A_0^c & B_0^c \\ \hline C_0^c & D_0^c \end{array}\right] = \left[\begin{array}{c|c} 0.0 & 1.0 \\ \hline -1.0 & -0.4 \end{array}\right] \tag{3.61}$$

such that the following stable CL pole remains:

$$poles_{CL(G,K_0)} = \left[-1\right] \tag{3.62}$$

Table 3.2: CL poles $(G, K_0(Q))$. YK controller reconfiguration between K_0 and K_1.

γ	CL poles
$\gamma = 0.0$	$[-998.67, -0.6660 \pm 25.027i]$
$\gamma = 0.1$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.2$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.3$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.4$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.5$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.6$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.7$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.8$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 0.9$	$[-998.67, -0.6660 \pm 25.027i, -25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$
$\gamma = 1.0$	$[-25.1218, -0.9022, -7.7082 \pm 1.01005i, -5.3047 \pm 1.1643i]$

Later, the controller is replaced by a slower one:

$$K_1 = \left[\begin{array}{c|c} A_1^c & B_1^c \\ \hline C_1^c & D_1^c \end{array}\right] = \left[\begin{array}{c|c} 0.0 & 0.5 \\ \hline -1.0 & -0.2 \end{array}\right] \tag{3.63}$$

which CL pole is closer to the origin:

$$poles_{CL(G,K_1)} = \Big[-0.5\Big] \tag{3.64}$$

Once G, K_0 and K_1 are defined as in Eqs. 3.60, 3.61 and 3.62, doubly coprime factors N_0, M_0, N_1, M_1, $\tilde{N_0}$, $\tilde{M_0}$, $\tilde{N_1}$, $\tilde{M_1}$, U_0, V_0, U_1, V_1, $\tilde{U_0}$, $\tilde{V_0}$, $\tilde{U_1}$, $\tilde{V_1}$ are obtained through Theorem 3.7. These factors are used for the calculation of Q, so the root locus comparison is carried out.

Figure 3.12 shows the root locus representation corresponding to different values of α in Eq. 3.59. The CL pole of the system moves from its position with K_0 to its position with K_1 as α increases. On the contrary, Fig. 3.13 plots the root locus representation corresponding to the different values of γ in the YK stable controller reconfiguration. YK controller reconfiguration includes zeros in the vicinity of the pole whose effect should cancel, instead of moving directly the pole position. At any stage on the transition process, the CL poles of the system are the poles of (G, K_0) and (G, K_1) compensated with a zero depending on the value of γ. For $\gamma \in [0, 1]$, poles position remains in $\{poles_{CL(G,K_0)} \wedge poles_{CL(G,K_1)}\}$, so there is not posibility that an unstable pole appears. On the contrary, for $\alpha \in [0, 1]$ in the direct linear change, CL poles move with the value of α, leaving the possibility that they go to the unstable area.

3.4.3 Transient behavior

By exploiting the YK parameterization it is possible to change controllers without losing stability, no matter what of the described control structures are used. As already mentioned, when doing controller transition, γ plays the key role as switching signal between K_0 and K_1. But even if stability is ensured, a step change from one controller to another can lead to unnaceptable

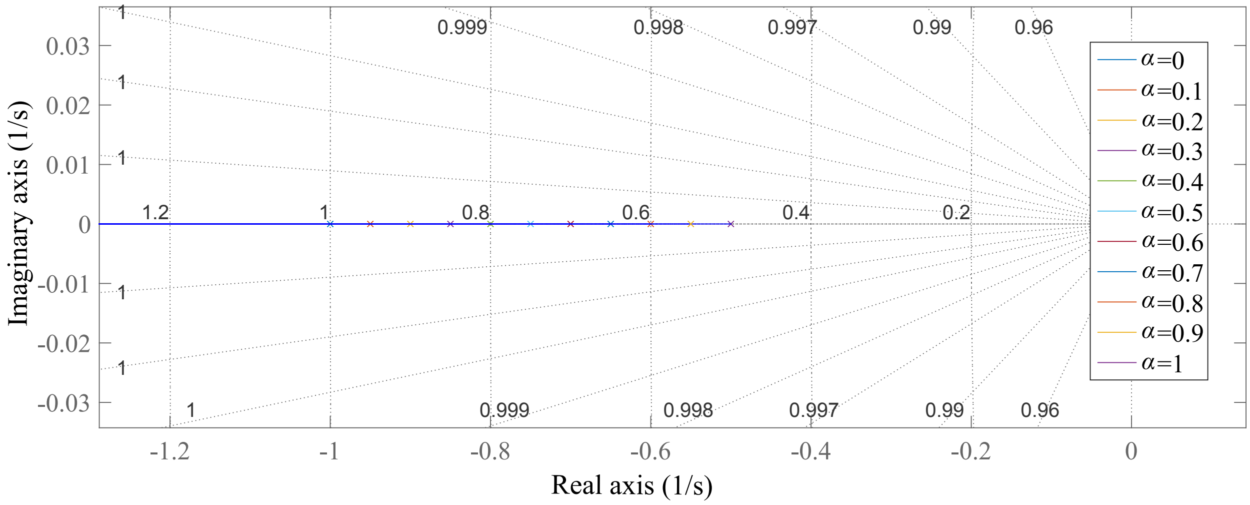


Figure 3.12: Root locus representation direct linear change between K_0 and K_1.

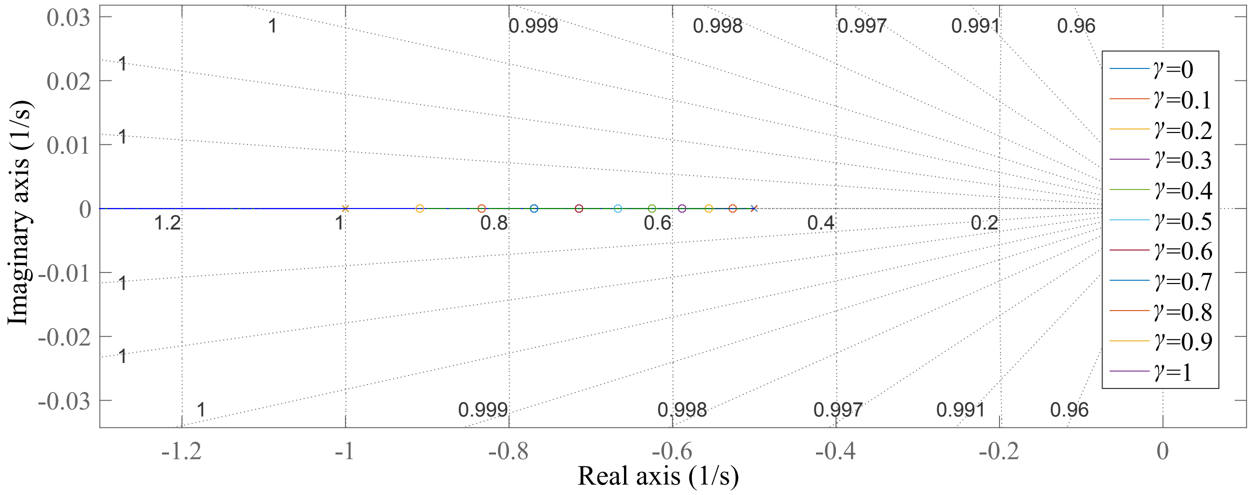


Figure 3.13: Root locus representation YK controller reconfiguration between K_0 and K_1.

transitions. It would cause spikes and oscillations in the output, which are not acceptable. The rate of change of the switching signal γ could be any, without affecting the CL stability of the system [Hespanha and Morse, 2002]. This subsection studies transient responses when using the different YK control structures, determining if the use of some structure improves the transient behavior when doing controller reconfiguration.

Notice that the rate of change of γ could be any, but a numerical example with the fastest γ rate is given such that transient differences are more remarkable. G, K_0 and K_1 are defined as in Eqs. 3.60, 3.61 and 3.63 respectively. Responses of both controllers are shown as black and blue dotted lines in the middle graph of Fig. 3.14. Doubly coprime factors N_0, M_0, N_1, M_1, $\tilde{N}_0$, $\tilde{M}_0$, $\tilde{N}_1$, $\tilde{M}_1$, U_0, V_0, U_1, V_1, $\tilde{U}_0$, $\tilde{V}_0$, $\tilde{U}_1$, $\tilde{V}_1$ are obtained through Theorem 3.7. These factors are used for the calculation of Q, Q' and Q'', so all the YK control structures can be implemented and compared in terms of transient performance.

Figure 3.14 depicts the transient behavior of each of the structures. The top graph represents how γ is modified to carry out the switching from K_0 to K_1. System responses when doing the

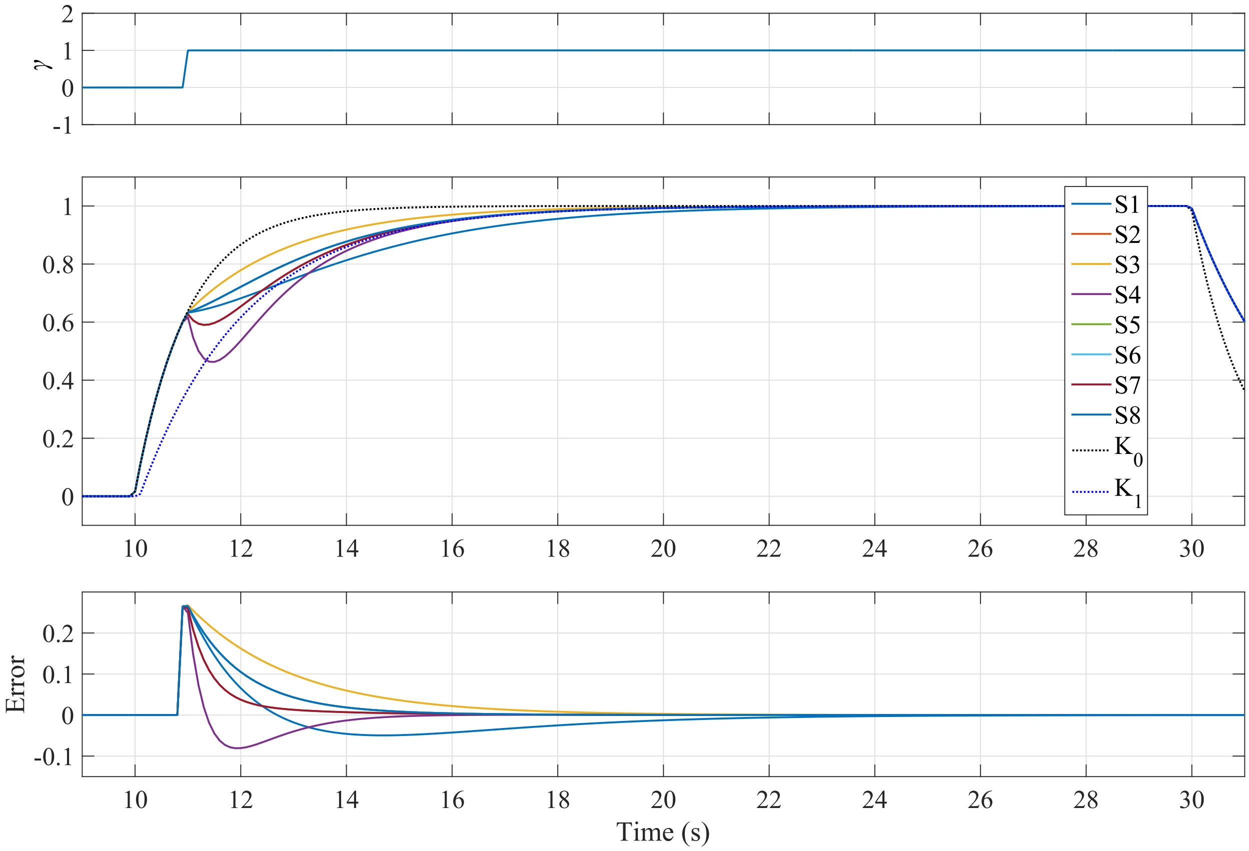


Figure 3.14: Comparison of transient behavior between different YK control structures. Step from 0 to 1.

transition are shown for the structures 1 to 8 at the middle graph. The bottom graph plots the error to K_1 response once the YK structure is activated. In all cases, the initial controller K_0 (black dotted line) is working until γ becomes 1 at $11s$. Then, the transient to reach K_1 response (blue dotted line) is different for each of the structures: Structures 5 and 7 presents the same response, the time to reach the desired behavior is the fastest $2s$, but with an oscillation in the response; structures 2, 6 and 8 takes twice the time but without overshoot; Structure 3 is even slower, about 8 s, but consequently transition is smoother; finally, structure 1 and 4 present oscillations over the desired response, with times reaching $12s$ and $4s$ respectively.

When doing controller reconfiguration, one looks for a transition without overshoot between K_0 and K_1 responses. This is the case of structures 2, 3, 6 and 8.

Table 3.3 gathers benefits and drawbacks of each structure seen in previous sections, as well as transition behavior characterized by rising time and the presence of oscillations.

Table 3.3: Summary Table.

	K_0 disconnection	Complexity	Iversability	YK parameter	System variations (Q,S)	Time	Oscillations
Structure 1	Yes	$7(n+n_{K_0})+3(n+n_{K_1})$	Yes	Q	Yes	$12s$	Yes
Structure 2	Yes	$7(n+n_{K_0})+3(n+n_{K_1})$	Yes	Q	Yes	$4s$	No
Structure 3	No	$5(n+n_{K_0})+3(n+n_{K_1})+n_{K_0}$	No	Q'	Yes	$8s$	No
Structure 4	No	$6(n+n_{K_0})+3(n+n_{K_1})+n_{K_0}$	No	Q''	Yes	$4s$	Yes
Structure 5	Yes	$7(n+n_{K_0})+3(n+n_{K_1})$	No	Q	Yes	$2s$	Yes
Structure 6	Yes	$7(n+n_{K_0})+3(n+n_{K_1})$	No	Q	Yes	$4s$	No
Structure 7	Yes	$4(n+n_{K_0})+2(n+n_{K_1})$	Yes	-	No	$2s$	Yes
Structure 8	Yes	$4(n+n_{K_0})+2(n+n_{K_1})$	Yes	-	No	$4s$	No

3.5 Conclusions

In this chapter, YK parameterization examines the use of doubly coprime factors to parameterize the class of all stabilizing controllers for a plant in terms of an initial stabilizing controller and a stable filter Q, called YK parameter.

The basis for the rest of the thesis has been defined. Derivation of a plant model, with transfer function and state space representation is provided. Stabilizing properties of feedback and feed-forward/feedback controllers for such plant model are revisited. The process of obtaining stable doubly coprime factors for a plant model G connected to any stabilizing controller K_i is explained; with special emphasis in the relation between coprime representations and stability properties, called Bézout's identity. The entire class of stabilizing controllers for the a plant model is then parameterized in terms of a stable filter Q using these stable coprime factors. It turns out that this approach gives rise to CL transfer functions that are affine in the stable filter Q. By using a scalar factor with the stable YK parameter Q, it is possible to switch between controllers in a stable way. If a direct linear change between controllers is performed, there is no guarantee that transition is stable. This lack of CL stability during transition is removed by using the YK parameterization. A numerical example is given, proving how YK parameterization is able to stabilize a transition between two controllers where direct linear change results unstable. At any point of the switching process, the CL poles of the system are the poles of the plant model with initial and final controllers compensated by zeros. Zeros position depends on the value of the scalar factor multiplying Q.

Finally, the YK mathematical basis is applied to derive the standard YK structures for controller switching. Different modifications are proposed to deal with problems such as order complexity, plant disconnection or matrix inversability. Eight structures are obtained, which controller complexity results in decreasing order: structures 1, 2, 5 and 6; structure 4; structure 3; and structures 7 and 8. Stability property is still preserved despite modifications in the structure, which means that CL poles during transition are the same even if different structures are used. Even if stability and CL poles are maintained during transition, transient behavior of each structure is investigated through a numerical example. Transient responses result different depending on the applied YK controller structure. Structures 1, 4, 5 and 7 exhibit an oscillating response to be avoided. The rest presents acceptable behaviors, but one or the other should be chosen depending on the desired time to reach the final response, controller complexity, presence of system variations or initial controller disconnection. Structures 2, 6 and 8 are the fastest ones without oscillation, while structure 3 take twice the time (so smoother) with a lower controller complexity, no need to disconnect the initial controller, and system variations included. Structure 3 will be the chosen one throughout this thesis.

Chapitre. Paramétrisation Youla-Jabr-Bongiorno-Kucera Duale

Below is a French summary of the following chapter "Dual Youla-Jabr-Bongiorno-Kucera parameterization".

Un des problèmes fréquents du contrôle est la conception d'un modèle Linéaire Invariant dans le Temps (LIT) pour un point d'opération donné, qui couvre rarement toute la gamme d'opération du système. Même si le système n'a qu'un unique point d'opération, les bruits externes ou les incertitudes peuvent exister, modifiant encore le comportement du systéme. C'est au court de ces situations que l'identification du système et le contrôle adaptatif jouent un rôle clé.

Ce chapitre décrit les bases mathématiques de l'application de la paramétrisation duale de Youla-Jabr-Bongiorno-Kucera (YK) dans les domaine de l'identification en boucle fermée (BF) et le contrôle adaptatif. La paramétrisation YK duale décrit l'ensemble de tous les systèmes stabilisés par un contrôleur LIT. Cette paramétrisation est aussi basée sur la double factorisation coprime. Cet ensemble est fonction du paramètre YK dual S.

Une description des variations du système dans cas réel est exprimé par des incertitudes sur le modèle ou par une représentation Linéaire à Paramétres Variants (LPV). Ceci peut être exprimé en fonction du paramètre YK dual S. S peut être interprété comme la différence entre le système réel et son modèle, rendant possible la connexion avec la paramétrisation YK. Ceci, dans le but de concevoir le paramètre YK Q capable de satisfaire des critères de stabilité/performance. D'autre part, le paramètre YK dual S peut être utilisé pour résoudre le probléme d'identification en BF et le transformer en identification en boucle ouverte (BO).

Le chapitre est structuré comme suit: la section 4.1 décrit un système général avec ses variations. La section 4.2 reformule la double factorisation coprime pour la paramétrisation YK duale. Dans la section 4.3, la classe de tous les systèmes stabilisés par un contrôleur donné est développée. La section 4.4 décrit comment cette classe et la classe de tous les contrôleurs stabilisant un système donné, peuvent être utilisées ensemble en analysant sa stabilité en BF et sa connexion avec la conception du contrôle adaptatif. Dans la section 4.5 la paramétrisation YK duale est utilisée pour transformer un problème d'identification BF en problème d'identification BO. Des détails à propos des deux différentes solutions sont donnés. Enfin, différentes remarques sont fournies en conclusion dans la section 4.6.

Chapter 4

Dual Youla-Jabr-Bongiorno-Kucera parameterization

One of the common problems in controllers is the design for a Linear-Time-Invariant (LTI) model in a given operation point, which rarely covers all the operation range of the system. Even if the plant has an unique operation point, external disturbances or uncertainties could exist, modifying again the system behavior. It is under these situations that system's identification and adaptive control play a key role.

This chapter describes the mathematical basis in which the dual Youla-Jabr-Bongiorno-Kucera (YK) parameterization relies on for making Closed-Loop (CL) identification and adaptive control. Dual YK parameterization describes the collection of all plants stabilized by a LTI controller. This parameterization is also based on the doubly coprime factorization. This collection is function of the dual YK parameter S.

A description of system variations for a real plant is given as uncertainties or Linear-Parameter-Varying (LPV) representation. These could be expressed in function of the dual YK parameter S. S can be interpreted as the difference between the real system and its model, being possible to make a connection with the YK parameterization in order to design the YK parameter Q able to fulfill some stability/performance criteria. On the other hand, the dual YK parameter S can be identified for solving CL identification problems as Open-Loop-like (OL-like) problems.

The chapter is structured as follows: Section 4.1 describes a general system with variations. Section 4.2 reformulates the doubly coprime factorization for the dual YK parameterization. In section 4.3, the class of all plants stabilized by a given controller is derived. Section 4.4 describes how this class can be used together with the class of all stabilizing controllers for a given plant, analysing its CL stability and connection with adaptive control design. In section 4.5, the dual YK parameterization is used to transform a CL identification problem into a OL-like identification problem. Details about two different schemes are given. Finally, some concluding remarks are in section 4.6.

4.1 System variations

Models are an aproximation of the real physical system. Usually, LTI models are used to represent physical systems, which are really time variant and nonlinear. As example, consider a LTI model for describing the yaw control of a vehicle. A complete description of this model is non-linear, and depends on variables such velocity, mass, cornering stiffness, road friction coefficient, and so on. A simple LTI model can at best capture only the essential behavior in the neighborhood of an operation point.

Every model-plant mistmatch is called uncertainty in the plant model. Generally speaking, the sources of uncertainty can be several: from plant aging, unmodeled dynamics (because the system is too complex, no easy model), neglected dynamics (accuracy is too expensive), operation over a large range of operation points, non-repeatable dynamics behaviours, inaccurate description of component characteristics, shifting of operation points, time delay, parasitic coupling, hysteresis, etc. The discrepancies between a system and its mathematical representation may lead to violation of some performance specifications, or even to the CL instability, and thus they should be modeled for a robust control design.

A typical procedure to represent uncertainty in models is the use of Linear Fractional Transformations (LFT) [Doyle et al., 1991]. It basically separates what is known from what is unknown in a feedback-like connection, bounding the possible values of the unknown elements. The basic principle is to eliminate all the uncertaintites which can appear in a plant and combine them in one uncertainty block Δ. This uncertainty block is used in robust control design, to achieve a performance/stability level in the present of uncertainties. Uncertainty modeling could not be simple, as in real systems all sources of uncertainty are mixed.

On the other hand, LPV representation [Bruzelius, 2004] offers a systematic way to obtain a nonlinear model suitable for control. LPV system is a state-space representation whose dynamics vary as function of certain time-varying parameters called scheduling parameters. It can represent practical/real systems subject to uncertainties as parameters variations, unmodeled dynamics and operation point shifting.

A LPV system can be expressed as:

$$\begin{aligned} \dot{x}(t) &= A(p(t))x(t) + B(p(t))u(t) \\ y(t) &= C(p(t))x(t) + D(p(t))u(t) \end{aligned} \tag{4.1}$$

where $p(t)$ is a vector of time-varying parameters assumed to be bounded $\forall t$. These time-varying parameters are called scheduling parameters. A LPV system is a state-space representation where $A(p)$, $B(p)$, $C(p)$ and $D(p)$ are the state space matrices parametrized by the scheduling parameter p. Scheduling parameters are exogenous if they are external variables. On the contrary, they are endogenous if they are function of the state variable $p = p(x(t))$, and, in that case, the LPV system is referred as a quasi-LPV system. Different system variations are considered depending on the nature of the scheduling parameter. Slow or fast scheduling parameter can be considered. Both related to plants which dynamics change with time, due for example the actual operation point of the system. Other representation is considered if p is piecewise-constant, or varies in a finite set of elements. This is a representation mainly used for switching systems.

It does not matter what type of representation is used; S describes any plant-model mistmath, and this can be identified in order to improve controller performance.

4.2 Doubly coprime factorization

A mandatory step towards the class of all systems stabilized by a controller K is the doubly coprime factorization. In order to obtain the class of all stabilizing controller for K, these coprime factors should represent G_i and K such that:

$$\begin{aligned} G_i &= N_i M_i^{-1} = \tilde{M_i}^{-1} \tilde{N}_i \\ K &= U_i V_i^{-1} = \tilde{V_i}^{-1} \tilde{U}_i \end{aligned} \tag{4.2}$$

These coprime factors should be stables U_i, $\tilde{U}_i$, V_i, $\tilde{V}_i$, N_i, $\tilde{N}_i$, M_i, $\tilde{M}_i \in \mathcal{RH}_\infty$, and satisfy the double Bézout's identity in Eq. 3.17. Notice how coprime factors change in comparison with Eq. 3.16, as a system that varies G_i is connected to a fixed controller K:

Theorem 4.1. *Consider a plant in state space representation G_i with A_i, B_i, C_i, D_i stabilizable and detectable, and a stabilizing controller K. F_i, F_i^c should be chosen such that $A_i + B_iF_i$ and $A^c + B^cF_i^c \in \mathcal{RH}_\infty$. Then doubly coprime factors are given by:*

$$\begin{bmatrix} M_i & U_i \\ N_i & V_i \end{bmatrix} = \left[\begin{array}{cc|cc} A_i + B_iF_i & 0 & B_i & 0 \\ 0 & A^c + B^cF_i^c & 0 & B^c \\ \hline F_i & C^c + D^cF_i^c & I & D^c \\ C_i + D_iF_i & F^c & D_i & I \end{array}\right] \tag{4.3}$$

$$\begin{bmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{bmatrix} = \left[\begin{array}{cc|cc} A_i + B_iY_iD^cC_i & B_iY_iC^c & -B_iY_i & B_iY_iD^c \\ B^cZ_iC_i & A^c + B^cZ_iD_iC^c & -B^cZ_iD_i & B^cZ_i \\ \hline F_i - Y_iD^cC_i & -C^c & I & -D^c \\ C_i & -F_i^c & 0 & I \end{array}\right] \tag{4.4}$$

$$\textit{with} \quad Y_i = (I - D^cD_i)^{-1} \quad \textit{and} \quad Z_i = (I - D_iD^c)^{-1}$$

Proof. For proof see [Ishihara and Sales, 1999]. □

In short, to obtain those elements, 1) construct a model-controller in state-space form, 2) solve a pole-assignment problem such $A_i + B_iF_i$ and $A^c + B^cF_i^c \in \mathbb{R}H_\infty$, and 3) perform algebraic manipulations according to Eqs. 4.3 and 4.4.

4.3 All systems stabilized by a controller

Once double coprime factors are obtained, it is possible to derive the dual theory related to Theorem 3.8. This theory allows to know the class of all the systems stabilized by a given controller [Tay et al., 1997].

Theorem 4.2. *Consider an initial plant G connected to a stabilizing fixed controller K described by their coprime factors $G = NM^{-1} = \tilde{M}^{-1}\tilde{N}$ and $K = UV^{-1} = \tilde{V}^{-1}\tilde{U}$. Then the set of all plants stabilized by the controller K is described by:*

$$\begin{aligned} G(S) &= (N + VS)(M + US)^{-1} = \\ &= (\tilde{M} + S\tilde{U})^{-1}(\tilde{N} + S\tilde{V}), \quad S \in \mathcal{RH}_\infty \end{aligned} \tag{4.5}$$

where S is a stable transfer function of appropiate dimensions called dual YK parameter.

Proof. Let's see how this $G(S)$ represents all the plants stabilized by K. CL stability between $G(S)$ and K is analysed (see Eq. 3.6 in the previous chapter):

$$\begin{bmatrix} I & -K \\ -G(S) & I \end{bmatrix}^{-1} = \begin{bmatrix} I & -G(S) \\ -K & I \end{bmatrix}^{-1} \in \mathcal{RH}_\infty \tag{4.6}$$

$$\begin{aligned}
\begin{bmatrix} I & -G(S) \\ -K & I \end{bmatrix}^{-1} = \begin{bmatrix} I & -(\tilde{M}+S\tilde{U})^{-1}(\tilde{N}+S\tilde{V}) \\ -\tilde{V}^{-1}\tilde{U} & I \end{bmatrix}^{-1} = \\
= \left(\begin{bmatrix} (\tilde{M}+S\tilde{U})^{-1} & 0 \\ 0 & \tilde{V}^{-1} \end{bmatrix} \begin{bmatrix} (\tilde{M}+S\tilde{U}) & -(\tilde{N}+S\tilde{V}) \\ -\tilde{U} & \tilde{V} \end{bmatrix} \right)^{-1} = \\
= \begin{bmatrix} V & (N+VS) \\ U & (M+US) \end{bmatrix} \begin{bmatrix} (\tilde{M}+S\tilde{U}) & 0 \\ 0 & \tilde{V} \end{bmatrix} = \\
= \left(\begin{bmatrix} V & N \\ U & M \end{bmatrix} + \begin{bmatrix} 0 & VS \\ 0 & US \end{bmatrix} \right) \left(\begin{bmatrix} \tilde{M} & 0 \\ 0 & \tilde{V} \end{bmatrix} + \begin{bmatrix} S\tilde{U} & 0 \\ 0 & 0 \end{bmatrix} \right) = \\
= \begin{bmatrix} V & N \\ U & M \end{bmatrix} \begin{bmatrix} \tilde{M} & 0 \\ 0 & \tilde{V} \end{bmatrix} + \begin{bmatrix} VS\tilde{U} & 0 \\ US\tilde{U} & 0 \end{bmatrix} + \begin{bmatrix} 0 & VS\tilde{V} \\ 0 & US\tilde{V} \end{bmatrix} = \\
= \begin{bmatrix} I & -G \\ -K & I \end{bmatrix}^{-1} + \begin{bmatrix} V \\ U \end{bmatrix} S \begin{bmatrix} \tilde{U} & \tilde{V} \end{bmatrix}
\end{aligned} \tag{4.7}$$

As coprime factors are stable by definition, CL stability between G and K is guaranteed; and then it is clear that any system parameterized by $S \in \mathcal{RH}_\infty$ is stabilized by K. □

The dual YK parameter S can represent any plant-model mistmatch present in a real system, either as a uncertainty Δ or as LPV representation. If the resulting S is stable, it means that uncertainty/plant-variation does not destabilize the loop. Nevertheless, the performance could be affected.

Equivalent to the previous chapter, below it is shown the parameterization in function of S_i from a nominal/initial LTI model G_0 to the one which reflects the real behavior of the system G_i. This real model could include fixed uncertainties or linear parameter variations in function of a scheduling parameter $p(t)$. The mistmatch source will determine if S_i is an LTI or LPV function. S_i is LTI or LPV depending on the real system G_i.

4.3.1 From a nominal model to a real model

Consider that a nominal LTI model G_0 is connected to an initial controller K_0. An arbitrary final LTI or LPV model G_i represents the real system. Then, the dual YK parameterization allows to represent G_i in function of S_i as follows:

Theorem 4.3. *Consider an initial plant model G_0 connected to a fixed controller K_0 described by their coprime factors $G_0 = N_0 M_0^{-1} = \tilde{M}_0^{-1}\tilde{N}_0$ and $K_0 = U_0 V_0^{-1} = \tilde{V}_0^{-1}\tilde{U}_0$. The real system connected to K_0 has dynamics corresponding to a model G_i, which coprime factors are $G_i = N_i M_i^{-1} = \tilde{M}_i^{-1}\tilde{N}_i$. Then, the real system G_i is represented in function of S_i as:*

$$\begin{aligned}
G_i = G_0(S_i) = (N_0 + V_0 S_i)(M_0 + U_0 S_i)^{-1} = (\tilde{M}_0 + S_i\tilde{U}_0)^{-1}(\tilde{N}_0 + S_i\tilde{V}_0) = \\
= G_0 + \tilde{M}_0^{-1} S_i (I + M_0^{-1} U_0 S_i) M_0^{-1}
\end{aligned} \tag{4.8}$$

where S_i is equivalent to:

$$S_i = X_i \tilde{M}_i (G_i - G_0) M_0 = X_i(\tilde{N}_i M_0 - \tilde{M}_i N_0) \tag{4.9}$$

with $X_i = V_0^{-1} V_i$.

Proof. Below, it is demonstrated how is possible to represent the plant model G_i as a stable S_i based on a initial model G_0 connected to a stabilizing controller K_0:

$$\begin{aligned}
G_0(S_i) &= G_0 + \tilde{M_0}^{-1} S_i (I + M_0^{-1} U_0 S_i)^{-1} M_0^{-1} = \\
&= G_0 + \tilde{M_0}^{-1} X_i \tilde{M}_i (G_i - G_0) M_0 (I + M_0^{-1} U_0 X_i \tilde{M}_i (G_i - G_0) M_0)^{-1} M_0^{-1} \\
G_0(S_i) &= G_0 + \tilde{M_0}^{-1} X_i \tilde{M}_i (G_i - G_0) M_0 (M_0 + U_0 X_i \tilde{M}_i (G_i - G_0) M_0)^{-1} = \\
&= G_0 + \tilde{M_0}^{-1} X_i \tilde{M}_i (G_i - G_0)(I + U_0 X_i \tilde{M}_i (G_i - G_0))^{-1} \\
G_0(S_i) &= G_0 + \tilde{M_0}^{-1} X_i \tilde{M}_i (I + U_0 X_i \tilde{M}_i (G_i - G_0))^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} X_i (\tilde{M_i}^{-1} + U_0 X_i (G_i - G_0))^{-1} (G_i - G_0) \\
G_0(S_i) &= G_0 + \tilde{M_0}^{-1} X_i (\tilde{M_i}^{-1} + U_0 X_i \tilde{M_i}^{-1} \tilde{N}_i - U_0 X_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} X_i (\tilde{M_i}^{-1} + U_0 V_0^{-1} V_i \tilde{M_i}^{-1} \tilde{N}_i - U_0 V_0^{-1} V_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} X_i (\tilde{M_i}^{-1} + U_i V_i^{-1} V_i \tilde{M_i}^{-1} \tilde{N}_i - U_i V_i^{-1} V_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} X_i (\tilde{M_i}^{-1} + U_i \tilde{M_i}^{-1} \tilde{N}_i - U_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) \\
G_0(S_i) &= G_0 + \tilde{M_0}^{-1} X_i ((I + U_i \tilde{N}_i) \tilde{M_i}^{-1} - U_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} X_i (V_i - U_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) = \\
&= G_0 + \tilde{M_0}^{-1} V_0^{-1} V_i (V_i - U_i \tilde{M_0}^{-1} \tilde{N}_0)^{-1} (G_i - G_0) \\
G_0(S_i) &= G_0 + (\tilde{M}_0 V_0 - U_0 \tilde{N}_0)^{-1} (G_i - G_0) = G_0 + (G_i - G_0) = G_i
\end{aligned} \tag{4.10}$$

□

Notice how in the LPV case, the real model G_i could vary with some scheduling parameter $p(t)$ or a time-varying uncertainty $\Delta(t)$, and therefore the corresponding dual YK parameter S_i. Specific representations in function of a scheduling parameter $p(t)$ or an uncertainty model Δ are found in [Bendtsen and Trangbaek, 2014] and [Niemann and Stoustrup, 1999] respectively. For the sake of simplicity, in the rest of the chapter, a LTI S_i is considered.

From the general description of any plant stabilized by a controller K_0 in Theorem 4.3, the structure shown in Fig. 4.1 is obtained (This is the one based on right coprime factors). The structure based on left coprime factors is depicted in Fig. 4.2. In both, n' represents the measurement noise n relocated to affect the output of S_i: $n' = (\tilde{M}_0 + \tilde{S}_i \tilde{U}_0) n$. r_1 and r_2, as in the previous chapter, could be the reference input and the feedforward output respectively; and finally, ζ_0 and z_0 represent S_i input and output.

4.4 Adaptive control design

The dual YK parameterization is used in connection with controller design. Any system variation can be expressed in function of the dual YK parameter S. It is then possible to use the dual YK parameterization in connection with the YK parameterization, so the changes related to S induce some change in Q with some stability or performance criteria. Different controller tuning techniques based on YK are named in this section, with special emphasis in Multi Model Adaptive Control (MMAC). But first, a connection between both classes is carried out, analysing CL stability for a system formed by $G_0(S_i)$ (Theorem 4.3) and $K_0(Q_i)$ (Theorem 3.11).

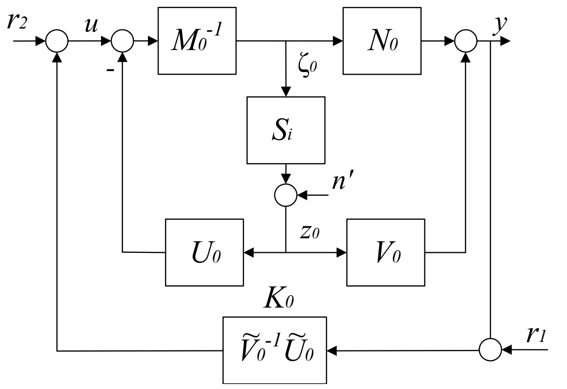


Figure 4.1: Dual YK parameterization structure based on right coprime factors.

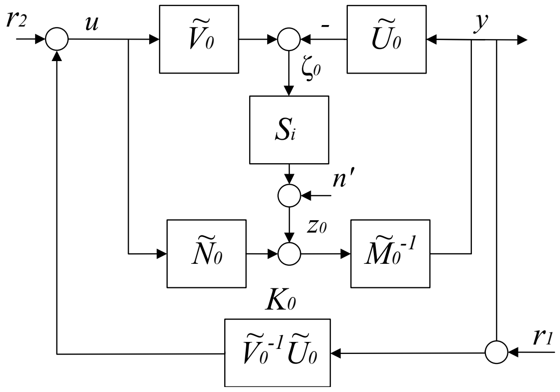


Figure 4.2: Dual YK parameterization structure based on left coprime factors.

Theorem 4.4. *Let (G_0, K_0) be a stabilizing LTI plant controller pair defined by its coprime factors $G_0 = N_0 M_0^{-1} = \tilde{M_0}^{-1}\tilde{N}_0$ and $K_0 = U_0 V_0^{-1} = \tilde{V_0}^{-1}\tilde{U}_0$. Consider the class of all stabilizing controllers for G_0 as:*

$$K_0(Q_i) = (U_0 + M_0 Q_i)(V_0 + N_0 Q_i)^{-1} = (\tilde{V}_0 + Q_i\tilde{N}_0)^{-1}(\tilde{U}_0 + Q_i\tilde{M}_0) \tag{4.11}$$

and the class of all the plants stabilized by K_0 as:

$$G_0(S_i) = (N_0 + V_0 S_i)(M_0 + U_0 S_i)^{-1} = (\tilde{M}_0 + S_i\tilde{U}_0)^{-1}(\tilde{N}_0 + S_i\tilde{V}_0) \tag{4.12}$$

with $Q_i, S_i \in \mathcal{RH}_\infty$. Then the pair $(G_0(S_i), K_0(Q_i))$ is CL stable if and only if the pair (Q_i, S_i) is stable:

$$\left(\begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}^{-1} - I\right) \in \mathcal{RH}_\infty \tag{4.13}$$

Proof. Consider CL stability condition in Theorem 3.2 together with descriptions in Eqs. 4.11 and 4.12:

$$\begin{aligned}
\begin{bmatrix} I & -K_0(Q_i) \\ -G_0(S_i) & I \end{bmatrix}^{-1} &= \begin{bmatrix} I & -(\tilde{V}_0 + Q_i\tilde{N}_0)^{-1}(\tilde{U}_0 + Q_i\tilde{M}_0) \\ -(\tilde{M}_0 + S_i\tilde{U}_0)^{-1}(\tilde{N}_0 + S_i\tilde{V}_0) & I \end{bmatrix}^{-1} = \\
&= \left(\begin{bmatrix} (\tilde{V}_0 + Q_i\tilde{N}_0)^{-1} & 0 \\ 0 & (\tilde{M}_0 + S_i\tilde{U}_0)^{-1} \end{bmatrix}\begin{bmatrix} (\tilde{V}_0 + Q_i\tilde{N}_0) & -(\tilde{U}_0 + Q_i\tilde{M}_0) \\ -(\tilde{N}_0 + S_i\tilde{V}_0) & (\tilde{M}_0 + S_i\tilde{U}_0) \end{bmatrix}\right)^{-1} = \\
&= \left(\begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}\begin{bmatrix} \tilde{V}_0 & -\tilde{U}_0 \\ -\tilde{N}_0 & \tilde{M}_0 \end{bmatrix}\right)^{-1}\begin{bmatrix} (\tilde{V}_0 + Q_i\tilde{N}_0) & 0 \\ 0 & (\tilde{M}_0 + S_i\tilde{U}_0) \end{bmatrix} = \\
&= \begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}^{-1}\begin{bmatrix} \tilde{V}_0 & -\tilde{U}_0 \\ -\tilde{N}_0 & \tilde{M}_0 \end{bmatrix}^{-1}\left(\begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}\begin{bmatrix} \tilde{V}_0 & 0 \\ 0 & \tilde{M}_0 \end{bmatrix} + \begin{bmatrix} Q_i\tilde{N}_0 & Q_i\tilde{M}_0 \\ S_i\tilde{V}_0 & S_i\tilde{U}_0 \end{bmatrix}\right) = \\
&= \begin{bmatrix} M_0 & U_0 \\ N_0 & V_0 \end{bmatrix}\begin{bmatrix} \tilde{V}_0 & 0 \\ 0 & \tilde{M}_0 \end{bmatrix} + \begin{bmatrix} M_0 & U_0 \\ N_0 & V_0 \end{bmatrix}\begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}^{-1}\begin{bmatrix} 0 & Q_i \\ S_i & 0 \end{bmatrix}\begin{bmatrix} \tilde{V}_0 & \tilde{U}_0 \\ \tilde{N}_0 & \tilde{M}_0 \end{bmatrix}
\end{aligned} \tag{4.14}$$

where:

$$\begin{bmatrix} M_0 & U_0 \\ N_0 & V_0 \end{bmatrix}\begin{bmatrix} \tilde{V}_0 & 0 \\ 0 & \tilde{M}_0 \end{bmatrix} = \begin{bmatrix} I & -K_0 \\ -G_0 & I \end{bmatrix}^{-1} \tag{4.15}$$

and

$$\begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}^{-1} \begin{bmatrix} 0 & Q_i \\ S_i & 0 \end{bmatrix} = \begin{bmatrix} I & -Q_i \\ -S_i & I \end{bmatrix}^{-1} - I \tag{4.16}$$

From these expresions is clear that the stability of the set $(G_0(S_i), K_0(Q_i))$ depends on an initial set (G_0, K_0) stable and stability of the pair (Q_i, S_i). □

This theorem is the heart of different control applications already seen in the state-of-the-art:

The iterative solution presented in [Tay et al., 1997]. An initial controller K_0 stabilizing both model G_0 and $G_0(S_i)$ can be updated/retuned by identifying S_i: This is done selecting a Q_i stabilizing S_i and fulfilling some performance criteria. The design of Q_i is based on the knowdlege of S_i. A simple transformation between Q_i and Q_i' could be done in order to use the structure three– $Q_i' = Q_i V_0^{-1}$. Notice how low order approximations for S_i should be used in order to keep a low order controller. Even with that, for practical implementation, controller order reduction may be required, as controller order increases with each iteration.

A direct relation between S_i and uncertainty Δ is adressed in [Niemann and Stoustrup, 1999]. This relation is used in the literature in H_∞ control problems, in order to design optimal controllers when uncertainty description is not full complex [Niemann, 1999]. A transformation between Δ and S_i permits to a have a full complex uncertain block, avoiding problems such increasing order controller and non-optimal solution. Once a connection between Δ and S_i is obtained, the idea is to find the optimal value of Q_i that minimizes the value of S_i.

Fault tolerant control applications are also found in the literature [Niemann and Stoustrup, 2005] [Niemann and Stoustrup, 2002] [Niemann and Stoustrup, 2004]. In that case, the use of the YK parameterization allows to have both fault recognition/detection (Dual YK parameterization) and controller reconfiguration (YK parameterization) in the same mathematical framework. An optimization algorithm is again needed in order to obtain the Q_i able to reduce the effect of the fault in the system. If S_i is unstable, the faulty system is unstable and a reconfiguration/retuning of the controller needs to be done through Q_i for stabilising the system. Plug&Play control extends the idea to connection/disconnection of sensors and actuators [Stoustrup, 2009].

All these previous solutions cover a great dynamic range as identification and optimization processes are carried out. With the idea of avoiding processes that could slow down the control loop, a set of linear plants that describes a wide range of system dynamics could be defined. Controller reconfiguration depending on a set of linear plants it is called in the literature MMAC [Lourenco and Lemos, 2006]. MMAC is a supervisor who chooses the proper controller, among pre-designed candidates controllers, once more information is known about the plant. Pre-designed controllers are conceived with the set of linear plants. MMAC is able to determine the closer plant in the set, switching to its corresponding controller to maintain a desired performance. [Anderson et al., 2001] proposed an indirect adaptive control based on identification of linear plants by using the ν-gap metric. As the metric is difficult to obtain in real time, [Baldi et al., 2011] proposed a similar approach but using model unfalsification: If a model and a controller are unable to reproduce the observed behavior in CL, then the set plant-controller is not the correct representation of the system. However, noise correlation problems are affecting system performance. The noise correlation problem is later solved in [Bendtsen and Trangbaek, 2012] by using the dual YK parameterization. The latter is extended below for a general set of nominal plants and predesigned controllers.

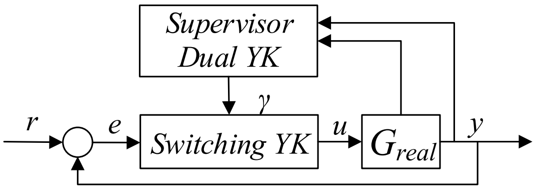


Figure 4.3: Outline structure used in MMAC.

4.4.1 Multi model adaptive control

Consider a set of nominal plants represented as $\{G\} = \{G_0..., G_i..., G_p\}$, describing different dynamics of a system. G_i denotes the i^{th} LTI plant mapping input signals u in output signals y. For each of these plants a LTI feedback controller K_i mapping error signals e in input signals u is designed such that the CL behavior of the system is the desired one. Thus, a set of candidate controllers is defined based on the nominal set of plants $\{K\} = \{K_0..., K_i, ..., K_p\}$. Time variations in the real plant G_{real} are considered slow compared to input-output dynamics. Nominal plants are defined, such that once a candidate controller is selected, it remains unchanged–i.e. the variations in G_{real} are smaller than those needed to change from one nominal plant to other within the set.

YK parameterization provides all stabilizing controllers for a given plant G_i within the set $\{G\}$ by interconnecting an initial controller K_0 with Q_i', called YK parameter. The initial controller could be any in the set $\{K\}$.

Thus, different controllers K_i can be implemented just by getting the YK parameter Q_i'. Different Q_i''s are obtained for each controller in the set $\{K\}$; so the YK parameter set is $\{Q'\} = \left\{Q_0', ..., Q_i', ..., Q_p'\right\}$ (see Theorem 3.13 for Q_i''s calculation). As the initial controller is K_0, its corresponding $Q_0' = 0$.

When doing controller transitions, the scalar factor γ_i is the switching signal. If the set of controllers is greater than 2, $p > 2$, a linear combination of all the controllers could be implemented as seen in Eq. 3.38. In a MMAC approach, just one of the candidate controllers is activated at the same time. Controller reconfiguration depends on the supervisor. It considers that the real plant G_{real} belongs to a set of nominal plants $\{G\} = \{G_0..., G_i..., G_p\}$, or at least is close to one of them. Each of the nominal plants is associated to a controller to give a desired performance $\{K\} = \{K_0..., K_i, ..., K_p\}$. An outline of the structure used in MMAC is shown in Fig. 4.3. The supervisor is at higher level, specifying which is the switching sequence γ that makes the system converge to the best controller for the unknown real plant G_{real}. If G_{real} coincides with one of the nominal plants in the set $\{G\}$, a good candidate controller K_i is straightforward. Otherwise, the closest nominal plant in the set should be chosen, switching to the corresponding controller.

The switching sequence $\{\gamma\} = \{\gamma_0..., \gamma_i, ..., \gamma_p\}$ is specified by the supervisor. Supervisor is based on the dual YK parameterization. The goal is to figure out which plant in the set $\{G\}$ is the closest to the real plant G_{real}. As outlined in [Bendtsen and Trangbaek, 2012], the dual YK parameter S_i does not need to be directly identified to know the closest plant in the set. If G_{real} coincides with the initial plant G_0, z_0 should be zero for any value of u and y (see structure in Fig. 4.2). By choosing different coprime factors $\tilde{M}_i$ and $\tilde{N}_i$ for every nominal plant in the set $\{G\}$, $z_i = \tilde{M}_i y - \tilde{N}_i u$ gives the closeness to these plants. The smallest truncated 2-norm $J_i = (\|z_i\|_2)^2$ will activate the signal γ_i corresponding to the controller K_i able to fulfill performance requirements.

The MMAC algorithm for a real plant G_{real} with a set of stabilizing controllers $\{K\}$ designed for a set of nominal plants $\{G\}$ is described in Algo. 1. h should be $h > 0$, and expresses the mandatory difference between two norms for controller change. In order to clarify the needed

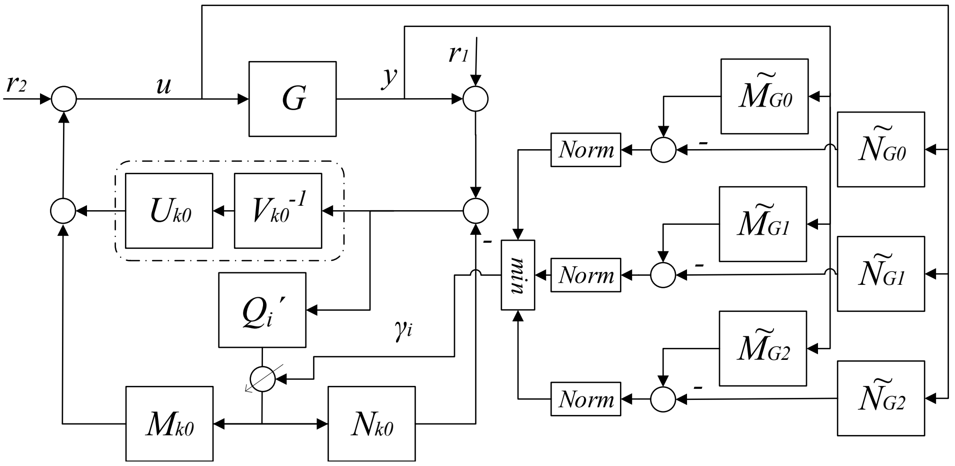


Figure 4.4: MMAC solution for Example 4.1.

coprime factors when using the YK and the dual YK parameterization in a MMAC approach, an example is given:

Example 4.1. *Consider a set composed by three models* $\{G\} = \{G_0, G_1, G_2\}$*. A controller is designed for each model following some performace criteria; and thus the resulting controller set is* $\{K\} = \{K_0, K_1, K_2\}$*.*

First, the class of all stabilizing controllers for a given plant is applied, so controller reconfiguration can be obtained. The controller K_0 *is considered as the initial controller, and the plant model is supposed to be* G_0*. Three different pairs plant-controller are considered:* (G_0, K_0)*,* (G_0, K_1)*,* (G_0, K_2)*. Coprime factors are obtained by applying Theorem 3.7:*

The pair (G_0, K_0) *is factorized in coprime factors* U_{k0}*,* $\tilde{U}_{k0}$*,* V_{k0}*,* $\tilde{V}_{k0}$*,* N_{k0}*,* $\tilde{N}_{k0}$*,* M_{k0}*,* $\tilde{M}_{k0} \in \mathcal{RH}_\infty$*.*

The pair (G_0, K_1) *is factorized in coprime factors* U_{k1}*,* $\tilde{U}_{k1}$*,* V_{k1}*,* $\tilde{V}_{k1}$*,* N_{k1}*,* $\tilde{N}_{k1}$*,* M_{k1}*,* $\tilde{M}_{k1} \in \mathcal{RH}_\infty$*.*

The pair (G_0, K_2) *is factorized in coprime factors* U_{k2}*,* $\tilde{U}_{k2}$*,* V_{k2}*,* $\tilde{V}_{k2}$*,* N_{k2}*,* $\tilde{N}_{k2}$*,* M_{k2}*,* $\tilde{M}_{k2} \in \mathcal{RH}_\infty$*.*

So, the set of YK parameters $\{Q'\} = \{Q'_0, Q'_1, Q'_2\}$ *results:*

$$Q'_0 = M_{k0}^{-1} M_{k0} (\tilde{U}_{k0} - \tilde{V}_{k0} \tilde{U}_{k0} \tilde{V}_{k0}^{-1}) = 0 \tag{4.17}$$

$$Q'_1 = M_{k0}^{-1} M_{k1} (\tilde{U}_{k1} - \tilde{V}_{k1} \tilde{U}_{k0} \tilde{V}_{k0}^{-1}) \tag{4.18}$$

$$Q'_2 = M_{k0}^{-1} M_{k2} (\tilde{U}_{k2} - \tilde{V}_{k2} \tilde{U}_{k0} \tilde{V}_{k0}^{-1}) \tag{4.19}$$

Once this is done, a controller structure as the one in Fig. 3.6 can be used to perform controller switching, which is the one with the best transient behavior.

Second, the class of all the models stabilized by a given controller is used to estimate the closest model in the set. The given controller is K_0*, but the real model could be any in the set* $\{G\}$*. Three different pairs plant-controllers are considered:* (G_0, K_0)*,* (G_1, K_0) *and* (G_2, K_0)*, so coprime factors result (Theorem 4.1):*

The pair (G_0, K_0) *is factorized in coprime factors* U_{G0}*,* $\tilde{U}_{G0}$*,* V_{G0}*,* $\tilde{V}_{G0}$*,* N_{G0}*,* $\tilde{N}_{G0}$*,* M_{G0}*,* $\tilde{M}_{G0} \in \mathcal{RH}_\infty$*. Notice how these factors are equivalent to* U_{k0}*,* $\tilde{U}_{k0}$*,* V_{k0}*,* $\tilde{V}_{k0}$*,* N_{k0}*,* $\tilde{N}_{k0}$*,* M_{k0} *and* $\tilde{M}_{k0}$*.*

Algorithm 1 Multi Model Adaptive Control

```
***1. Initialization***
$\gamma[p] = [0]$                                      ▷ Switching sequence initialization
$K[p] = [K_0, ..., K_i, ..., K_p]$                     ▷ Candidate controllers
$\tilde{M}[p] = [\tilde{M}_0, ..., \tilde{M}_i, ..., \tilde{M}_p]$   ▷ Left coprime factor $M$ for $G_i$
$\tilde{N}[p] = [\tilde{N}_0, ..., \tilde{N}_i, ..., \tilde{N}_p]$   ▷ Left coprime factor $N$ for $G_i$
$z[p] = [0]$                                           ▷ $S_i$ output initialization
$J[p] = [0]$                                           ▷ Truncated 2-norm initialization
**loop**
***2. YK Controller reconfiguration***
UpdateController($\gamma$)                             ▷ Apply controller $K_i$, with $i = \gamma$
Get($u$,$y$)                                           ▷ Obtain measurements $u$ and $y$
***3. Supervisor***
***3.1 Closeness to plants in set***
**for** $i$ in $(0, n)$ **do**
    $z[i] = \tilde{M}[i]y - \tilde{N}[i]u$             ▷ Output of $S_i$
    $J[i] = (norm2(z[i]))^2$                           ▷ Compute truncated 2-norm/ Closeness to nominal plants
**end for**
 $i_{min} = argmin_i\{i \in n \mid J[i]\}$             ▷ The smallest norm corresponds to the closer plant
***3.2 Evaluate switching sequence***
**if** $(J[\gamma[i] == 1] \leq J[i_{min}] + h)$ **then**
    $\gamma = \gamma$                                  ▷ Previous controller remains
**else**
    $\gamma[i_{min}] = 1$                              ▷ Controller changes
    $\gamma[\forall i \quad except \quad i = i_{min}] = 0$
**end if**
**end loop**
```

The pair (G_1, K_0) is factorized in coprime factors U_{G1}, $\tilde{U}_{G1}$, V_{G1}, $\tilde{V}_{G1}$, N_{G1}, $\tilde{N}_{G1}$, M_{G1}, $\tilde{M}_{G1} \in \mathcal{RH}_\infty$. Those coprime factors differs from the ones for the pair (G_0, K_1).

The pair (G_2, K_0) is factorized in coprime factors U_{G2}, $\tilde{U}_{G2}$, V_{G2}, $\tilde{V}_{G2}$, N_{G2}, $\tilde{N}_{G2}$, M_{G2}, $\tilde{M}_{G2} \in \mathcal{RH}_\infty$. Those coprime factors differs from the ones for the pair (G_0, K_2).

Notice how the coprime factors needed for closeness ($z_i = \tilde{M}_i y - \tilde{N}_i u$) are those corresponding to sets (G_1, K_0) and (G_2, K_0), and not the ones used for controller reconfiguration. Coprime factors X_{ki} are only used for the calculation of Q_i. A detailed block diagram of MMAC with coprime factors is shown in Fig. 4.4.

4.5 Dynamics identification

In this section, before explaining the dual YK-based CL identification, a general OL identification setup is compared to a general CL identification setup, so the advantages of using the second one can be highlighted.

4.5.1 Open-loop identification

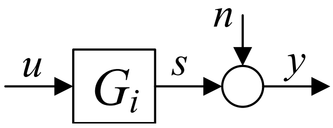


Figure 4.5: Setup for OL identification.

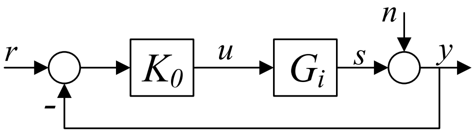


Figure 4.6: Setup for CL identification.

Let's consider an OL identification case as the one described in Fig. 4.5. Control input u and measurement noise n are assumed to be uncorrelated. Some control input u can be applied to the system G_i, obtaining the corresponding output y with noise n:

$$y = G_i u + n \tag{4.20}$$

If measurements u and y are available, many OL identification schemes (Auto Regressive model with eXternal inputs (ARX) [Karaboyas and Kalouptsidis, 1991], PBSIDopt [van Wingerden, 2012] ...) can be used to find cross-correlation with u, and estimate G_i

$$\Phi_{yu} = G_i \Phi_{uu} + \Phi_{nu} \tag{4.21}$$

where $\Phi_{nu} = 0$ as n and u are independent.

4.5.2 Closed-loop identification

Now let's consider the case where the loop is closed with a controller K_0 (see Fig. 4.6), where reference signal r and output noise n are uncorrelated. Equation 4.21 remains, but Φ_{nu} is not zero, as n is feedback through the controller K_0 affecting the control signal u. Cross-correlation expression results:

$$\Phi_{yu} = G_i \Phi_{uu} - (1 + K_0^* G_i^*)^{-1} \Phi_{nn} \tag{4.22}$$

where the superscript * denotes complex conjugation on the $j\omega$ axis.

It is obvious that the identification process becomes complex. Even if Φ_{nn} is really small, Eq. 4.22 denotes that G_i should be stable, what could not be the case. As a solution, one could seek to estimate the CL transfer function from the reference signal r to y: $P_i = K_0G_i(1 + K_0G_i)^{-1}$. Once an estimation of the CL function is obtained $\hat{P}_i$, an estimation of the real system $\hat{G}_i$ would be:

$$\hat{G}_i = \frac{\hat{P}_i}{K_0(1 - \hat{P}_i)} \tag{4.23}$$

but problems could occur if K_0 has some unstables poles/zeros; the estimation could result again unstable.

It is then logical to disconnect the plant in order to carry out an OL identification. But there will be cases in which this is not possible: The plant is unstable, disconnecting the plant supposes a great economic cost, the feedback controller is embedded in the system, or an online estimation of the system is needed for controller improvement, being necessary CL identification. Among OL and CL identification methods, it is well-known that for model-based control design, CL identification gives better performance [Hjalmarsson et al., 1996]; but you need to deal with its associated difficulties: Linear matrix inequality (LMI) feasibility [Sznaier and Mazzaro, 2003], linear fractional dependence with respect to measured variables [Salcedo and Martinez, 2008] or linear-deterministic subspace selection [Santos et al., 2007] are some examples picturing these difficulties.

A clever solution to suppress CL identification difficulties was given by [Hansen et al., 1989]. This solution is mentioned in the literature as Hansen scheme. Given a LTI initial model and a controller, the key idea is to identify the dual YK S_i instead of G_i. Interestingly, the identification of S_i is a standard OL identification problem, so a CL problem is transformed into an OL-like problem.

Modifications and extensions of the Theorem have been carried in the literature. A non-linear initial model G_0 connected to a stabilizing controller K_0 is considered in [Linard and Anderson, 1996] [Linard and Anderson, 1997]. [De Bruyne et al., 1998] presented a modification able to tune the order of the resulting model given by the Hansen scheme. Several analysis have later demonstrated how the obtained model with the Hansen scheme is superior than an OL identification solution for subsequent control design [Gevers et al., 2001] [Douma et al., 2003]. The Hansen scheme has been also extended to LPV systems [Bendtsen and Trangbaek, 2014] [Trangbaek and Bendtsen, 2010], and identification of actuators/sensors connected to a system [Knudsen et al., 2008]. The most recent work in the literature dates from the year 2015, and presents a modification of the Hansen scheme able to get rid of signals that are not directly measurable, eliminating the need of some a priori knowdledge and reducing numerical accuracies [Sekunda et al., 2015]. In the following section, Hansen scheme and its most recent modification are introduced.

4.5.2.a Hansen scheme

From the general description of any plant stabilized by a controller K_0 provided by the structure in Fig. 4.1, the following relations are derived:

$$u - U_0z_0 = M_0\zeta_0 \tag{4.24}$$

$$y - V_0z_0 = N_0\zeta_0 \tag{4.25}$$

Applying the coprime factors $\tilde{N}_0$ and $\tilde{M}_0$ in Eqs. 4.24 and 4.25 respectively, yields:

$$\tilde{N}_0(u - U_0z_0) = \tilde{N}_0M_0\zeta_0 \tag{4.26}$$

$$\tilde{M_0}(y - V_0 z_0) = \tilde{M_0} N_0 \zeta_0 \tag{4.27}$$

subtracting one from the other and applying the Bézout's identity in Eq. 3.17, results:

$$\begin{aligned} \tilde{M_0}(y - V_0 z_0) &= \tilde{N_0}(u - U_0 z_0) \\ \tilde{M_0} y - \tilde{N_0} u &= (V_0 \tilde{M_0} - U_0 \tilde{N_0}) z_0 \\ z_0 &= \tilde{M_0} y - \tilde{N_0} u \end{aligned} \tag{4.28}$$

Again from the block diagram in Fig. 4.1, the following expressions result:

$$\begin{aligned} N_0 \zeta_0 + V_0 (S_i \zeta_0 + n') &= y \\ (N_0 + V_0 S_i) \zeta_0 &= y - V_0 n' \end{aligned} \tag{4.29}$$

$$\begin{aligned} u - U_0 S_i \zeta_0 - U_0 n' &= M_0 \zeta_0 \\ (M_0 + U_0 S_i) \zeta_0 &= u - U_0 n' \\ (M_0 + U_0 S_i) \zeta_0 &= r_2 + \tilde{V_0}^{-1} \tilde{U_0} (y + r_1) - U_0 n' \end{aligned} \tag{4.30}$$

Applying coprime factors $\tilde{U_0}$ and $\tilde{V_0}$ in Eqs. 4.29 and 4.30, the relations are:

$$\tilde{U_0}(N_0 + V_0 S_i) \zeta_0 = \tilde{U_0} y - \tilde{U_0} V_0 n' \tag{4.31}$$

$$\tilde{V_0}(M_0 + U_0 S_i) \zeta_0 = \tilde{V_0} r_2 + \tilde{U_0}(y + r_1) - \tilde{V_0} U_0 n' \tag{4.32}$$

substituying Eq. 4.31 in Eq. 4.32, yields:

$$\begin{aligned} \tilde{V_0}(M_0 + U_0 S_i) \zeta_0 &= \tilde{U_0} r_1 + \tilde{V_0} r_2 + \tilde{U_0}(N_0 + V_0 S_i) \zeta_0 \\ (\tilde{V_0} M_0 + \tilde{V_0} U_0 S_i - \tilde{U_0} N_0 - \tilde{U_0} V_0 S_i) \zeta_0 &= \tilde{U_0} r_1 + \tilde{V_0} r_2 \\ (\tilde{V_0} M_0 - \tilde{U_0} N_0) \zeta_0 &= \tilde{U_0} r_1 + \tilde{V_0} r_2 \\ \zeta_0 &= \tilde{U_0} r_1 + \tilde{V_0} r_2 \end{aligned} \tag{4.33}$$

Assuming that the output noise n is not correlated to r_1 and r_2, then ζ_0 is also independent of n'. Thus, although u and y are measured in CL, the identification of the dual YK parameter S_i is OL by using the signals ζ_0 and z_0. OL identification algorithms like ARX [Karaboyas and Kalouptsidis, 1991] or PBSIDopt [van Wingerden, 2012] can be used for obtaining S_i. By identifying S_i in OL, advantages of CL identification are preserved with a simpler method.

Theorem 4.5. *Given an initial LTI model G_0 and a stabilizing LTI controller K_0. A CL identification of the real system G_i connected to K_0 is possible through the OL identification of the associated dual YK parameter S_i. Filtered signals ζ_0 and z_0 are obtained in order to estimate $\hat{S}_i$ through any OL identification algorithm:*

$$\begin{aligned} \zeta_0 &= \tilde{U_0} r_1 + \tilde{V_0} r_2 \\ z_0 &= \tilde{M_0} y - \tilde{N_0} u \end{aligned} \tag{4.34}$$

where r_1 and r_2 are external excitation signals; and u and y are control input and output measurement respectively. Then, the equivalent CL model results:

$$\hat{G_i} = (N_0 + V_0 \hat{S_i})(M_0 + U_0 \hat{S_i})^{-1} \tag{4.35}$$

Proof. Proof is above. □

From this Theorem, it is clear that it is impossible to obtain direct measurements of the internal signals ζ_0 and z_0. These signals result from filtered information in the system. [Sekunda et al., 2015] stated that this approach might lead to some numerical problems, as excitation signals r_1 and r_2 are imposed for the OL identification of S_i. Sekunda scheme studies how to directly apply a signal equivalent to ζ_0 and how to measure a signal equivalent to z_0.

4.5.2.b Sekunda scheme

Sekunda scheme is based on the relation between the class of all the plants stabilized by a controller $G_0(S_i)$, and the class of all the controllers stabilizing a given plant $K_0(Q_i)$. This relation is first seen in [Tay et al., 1997], as the problem of identifying S_i when a controller $K_0(Q_i)$ has been implemented. Both parameterizations can be seen as LFT depending on S_i and Q_i:

$$K_0(Q_i) = F_l(J_{K_0}, Q_i) \quad with \quad J_{K_0} = \begin{bmatrix} K_0 & \tilde{V}_0^{-1} \\ V_0^{-1} & -V_0^{-1}N_0 \end{bmatrix} \tag{4.36}$$

$$G_0(S_i) = F_u(J_{G_0}, S_i) \quad with \quad J_{G_0} = \begin{bmatrix} -M_0^{-1}U_0 & M_0^{-1} \\ \tilde{M}_0^{-1} & G_0 \end{bmatrix} \tag{4.37}$$

A connection between Eqs. 4.36 and 4.37 is shown in Fig. 4.7. The cross coupling between J_{G_0} and J_{K_0} is solved through the Redheffer star product [Redheffer, 1960] of two matrices:

$$J_{G_0} * J_{K_0} = \begin{bmatrix} F_l(J_{G_0}, K_0) & M_0^{-1}(I - K_0G_0)^{-1}\tilde{V}_0^{-1} \\ V_0^{-1}(I - K_0G_0)^{-1}\tilde{V}_0^{-1} & F_u(J_{K_0}, G_0) \end{bmatrix} \tag{4.38}$$

where:

$$\begin{aligned} F_l(J_{G_0}, K_0) &= -M_0^{-1}U_0 + M_0^{-1}K_0(I - G_0K_0)^{-1}\tilde{M}_0^{-1} = \\ &= -M_0^{-1}U_0 + M_0^{-1}U_0V_0^{-1}(I - \tilde{M}_0^{-1}\tilde{N}_0U_0V_0^{-1})^{-1}\tilde{M}_0^{-1} = \\ = -M_0^{-1}U_0 + M_0^{-1}U_0V_0^{-1}V_0(\tilde{M}_0V_0 - \tilde{N}_0U_0)^{-1}\tilde{M}_0\tilde{M}_0^{-1} &= -M_0^{-1}U_0 + M_0^{-1}U_0 = 0 \end{aligned} \tag{4.39}$$

$$\begin{aligned} F_u(J_{K_0}, G_0) &= -V_0^{-1}N_0 + V_0^{-1}G_0(I - K_0G_0)^{-1}\tilde{V}_0^{-1} = \\ &= -V_0^{-1}N_0 + V_0^{-1}N_0M_0^{-1}(I - \tilde{V}_0^{-1}\tilde{U}_0N_0M_0^{-1})^{-1}\tilde{V}_0^{-1} = \\ = -V_0^{-1}N_0 + V_0^{-1}N_0M_0^{-1}M_0(\tilde{V}_0M_0 - \tilde{U}_0N_0)^{-1}\tilde{V}_0\tilde{V}_0^{-1} &= -V_0^{-1}N_0 + V_0^{-1}N_0 = 0 \end{aligned} \tag{4.40}$$

$$\begin{aligned} M_0^{-1}(I - K_0G_0)^{-1}\tilde{V}_0^{-1} &= M_0^{-1}(I - \tilde{V}_0^{-1}\tilde{U}_0N_0M_0^{-1})^{-1}\tilde{V}_0^{-1} = \\ = M_0^{-1}M_0(\tilde{V}_0M_0 - \tilde{U}_0N_0)^{-1}\tilde{V}_0\tilde{V}_0^{-1} &= (\tilde{V}_0M_0 - \tilde{U}_0N_0)^{-1} = I \end{aligned} \tag{4.41}$$

$$\begin{aligned} V_0^{-1}(I - G_0K_0)^{-1}\tilde{M}_0^{-1} &= V_0^{-1}(I - \tilde{M}_0^{-1}\tilde{N}_0U_0V_0^{-1})^{-1}\tilde{M}_0^{-1} = \\ = V_0^{-1}V_0(\tilde{M}_0V_0 - \tilde{N}_0U_0)^{-1}\tilde{M}_0\tilde{M}_0^{-1} &= (\tilde{M}_0V_0 - \tilde{N}_0U_0)^{-1} = I \end{aligned} \tag{4.42}$$

And then it is proved that the output of Q_i is the input of S_i, and the input of Q_i is the output of S_i, being possible to measure the signals ζ_0 and z_0 from s_0 and r_0. Connection between S_i and Q_i is in Fig. 4.8.

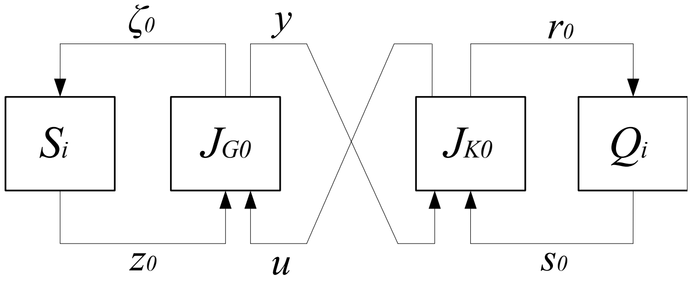


Figure 4.7: Block representation of the connection between $G_0(S_i)$ and $K_0(S_i)$.

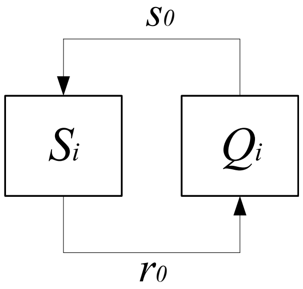


Figure 4.8: Connection between Q_i and S_i.

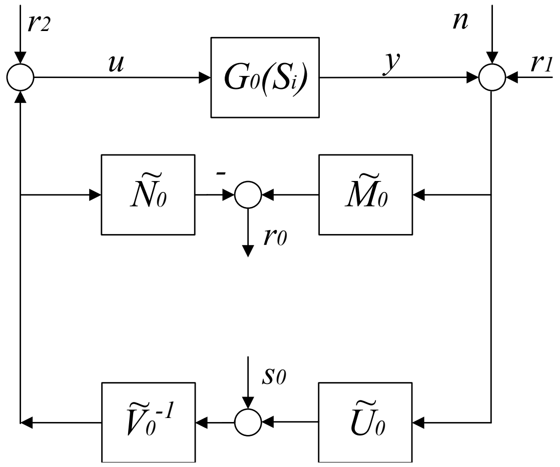


Figure 4.9: General control scheme using YK structure 2 and $G_0(S_i)$.

Connection of s_0 and r_0 with r_1, r_2, u and y is analysed. For doing so, a YK control structure as the one in Fig. 3.5 is connected to $G_0(S_i)$ with $Q_i = 0$. The resulting CL structure is shown in Fig. 4.9. The transfer function relations between all the signals are reagrupped and summarized in a matricial form as follows:

$$\begin{bmatrix} y \\ u \\ r_0 \end{bmatrix} = \begin{bmatrix} (N_0 + V_0 S_i)\tilde{U}_0 & (N_0 + V_0 S_i)\tilde{U}_0 & (N_0 + V_0 S_i)\tilde{V}_0 & (N_0 + V_0 S_i) \\ (M_0 + U_0 S_i)\tilde{U}_0 & (M_0 + U_0 S_i)\tilde{U}_0 & (M_0 + U_0 S_i)\tilde{V}_0 & (M_0 + U_0 S_i) \\ (\tilde{M}_0 + S_i \tilde{U}_0) & (\tilde{M}_0 + S_i \tilde{U}_0) & (\tilde{N}_0 + S_i \tilde{V}_0) & S_i \end{bmatrix} \begin{bmatrix} n \\ r_1 \\ r_2 \\ s_0 \end{bmatrix} \tag{4.43}$$

where r_1, r_2 and r_0 are uncorrelated. By applying an excitation signals s_0 it is possible to measure r_0. This method is superior as it is possible to get r_0 independently of the external excitation signals r_1 and r_2.

In the absence of external excitation signals r_1 and r_2, z_0 and ζ_0 yield:

$$\begin{aligned} z_0 = r_0 &= \tilde{M}_0 y - \tilde{N}_0 u \\ \zeta_0 = s_0 &= \tilde{V}_0 u - \tilde{U}_0 y \end{aligned} \tag{4.44}$$

4.6 Conclusions

In this chapter, the dual YK parameterization is examined to parameterize the class of all plants stabilized by a given controller in terms of an initial nominal model and a stable filter S, called dual YK parameter. The use of the dual YK parameterization is justified when there is variation in the dynamics system to control. Variations as uncertainty Δ or LPV representations are briefly introduced.

The process of obtaining stable doubly coprime factors for several plant models G_i connected to a stabilizing controller K is formulated; then the dual question of describing all the plants stabilized by a given controller is considered. As in a feedback loop, controller and plant can be interchanged without affecting the CL stability, the dual version of Theorem 3.8 is obtained in function of the dual YK parameter S. When a nominal initial model is compared with the real system, the function S results to be the mistmatch between nominal plant and real system. The mistmatch representation of S could be in function of a LPV scheduling parameter, an uncertainty Δ, or in a more general way as a LPV tranfer function. If the resulting S is stable, it means that the mistmatch does not destabilize the loop. Nevertheless, the performance of the system will be affected.

Once a explicit relation between system changes and the dual YK parameter is derived, the dual YK parameterization can be applied in connection with the design of a controller. A connection with the class of all stabilizing controllers introduced in the previous chapter is carried out, resulting that the dual YK parameter S is the OL transfer function between input and output of the YK parameter Q. Then, stability of the CL formed by $G(S)$ and $K(Q)$ requires stability of the nominal CL system (G, K) and stability of the dual YK parameter S with the YK parameter Q. Given S, control design should be in the direction of getting the optimal value of Q that fulfils some stability/performance criteria; but it will require a high computational cost, that wouldn't be affordable for some industrial applications. So, a YK-based MMAC algorithm based on a set of nominal plants and predefined controllers is proposed. MMAC acts as a supevisor determining the closer plant to the real system in the set, switching to the predesigned controller that will give the best performance possible. As identification and/or optimization processes are not needed, the control-loop is not slowed down.

Finally, the dual YK parameterization is used to perform CL identification of system dynamics. The CL identification provided by the Hansen scheme is based in the OL identification of the dual YK parameter S. Mathematical proof is given of how an OL identification of S is equivalent to a CL identification of the real system connected to a controller. Identification using Hansen scheme was conducted using indirect excitation signals r_1 and r_2 for identification. This approach makes difficult to determine the frequency response of the excitation signal, as r_1 and r_2 could be already reference input and feedforward output. In order to solve so, Sekunda scheme proposed a modification, letting to impose any desired excitation signal for the identification of S.

Chapitre. Applications

Below is a French summary of the following chapter "Applications".

Ce chapitre explore l'utilisation des deux paramétrisations Youla-Kucera (YK) et YK duale dans les véhicules autonomes, en insistant spécialement sur la stabilité quand différents changements dynamiques ou situations de trafic, nécessitent de reconfigurer le contrôleur. Des simulations et résultats expérimentaux sont obtenus par différentes applications sur des Systèmes de Transports Intelligents (STI). La paramétrisation YK est utilisée dans l'application "Cooperative Adaptive Cruise Control" (CACC) pour résoudre différentes situations qui n'ont pas encore été résolues dans l'état de l'art. Spécifiquement, un comportement hybride entre deux contrôleurs CACC avec différents temps d'écart en utilisant la reconfiguration stable de YK est étudié. Ce comportement hybride utilise les propriétés de YK pour éviter les dégradations ACC quand la communication avec le véhicule précédent n'est plus disponible; et pour assurer la stabilité quand les véhicules sont entrant/sortant de la chaîne de véhicules. Les deux applications sont développées pour des véhicules ayant la même dynamique. Cependant, il est clair qu'en situation de trafic réel, la dynamique changera. En conséquence, l'identification en Boucle-Fermée (BF) basée sur YK est utilisée dans le but d'identifier les dynamiques du véhicule connecté à un système CACC. Finalement, la reconfiguration du contrôleur basé sur YK et l'identification BF sont tout les deux utilisés, ensemble, dans le but d'obtenir une approche adaptative, capable de gérer l'hétérogénéité dynamique dans une chaîne de véhicules CACC.

Le chapitre est structuré comme suit : la plateforme expérimentale et les modèles de simulation utilisés pour développer les différentes applications CACC sont décrits dans la section 5.1. La section 5.2 traite de la reconfiguration du contrôleur quand des changements spécifiés apparaissent. Une chaine de véhicules équipée avec un système CACC est considérée. La paramétrisation YK est utilisée pour améliorer la circulation quand la communication Vehicule-to-Vehicule (V2V) échoue ou quand les véhicules sont entrant/sortant de la chaîne. La section 5.3 considère l'dentification en BF de la dynamique longitudinale du véhicule . Une comparaison entre l'identification en Boucle-Ouverte (BO) et l'identification en BF fournie par la paramétrisation YK duale est faite durant l'utilisation du système CACC. La section 5.4 utilise la reconfiguration du contrôleur et l'identification, ensemble, dans un Contrôle Adaptatif Multi Model (CAMM). L'idée est de pouvoir gérer l'hétérogénéité de la chaîne de véhicules équipée du CACC. Enfin, différentes remarques sont fournies en conclusion dans la section 5.5, déterminant si la paramétrisation YK peut être utile comme outil général de contrôle pour sécuriser la réponse du véhicule autonome.

Chapter 5

Applications

This chapter explores the use of both Youla-Kucera (YK) and dual YK parameterizations in automated vehicles, with special emphasis on stability when some dynamics changes or the traffic situation demand controller reconfiguration. Both simulation and experimental results are obtained for different Intelligent Transportation Systems (ITS) applications.

YK parameterization is used with Cooperative Adaptive Cruise Control (CACC) application for solving cases that have not been addressed in the state-of-the-art. Specifically, an hybrid behaviour between two CACC controllers with different time gaps is explored by means of the YK-based stable controller reconfiguration. This hybrid behaviour uses YK properties for avoiding ACC degradation when communication link with preceding vehicle is not available; and for ensuring stability when vehicles are entering/exiting the string of vehicles. Both applications are developed for vehicles with same dynamics, but it is clear that in a real traffic situation, dynamics will change. Consequently, YK-based Closed-Loop (CL) identification is used in order to identify the dynamics of a vehicle connected to a CACC system. Finally, both YK-based controller reconfiguration and CL identification are applied together in order to obtain an adaptive approach able to deal with vehicle heterogeneity in CACC string of vehicles.

The chapter is structured as follows: Experimental platform and simulation models used to develop the different CACC applications are described in section 5.1. Section 5.2 deals with controller reconfiguration when a prespecified change occurs. A string of vehicles equipped with a CACC system is considered. YK parameterization is used to enhance traffic flow when Vehicle-to-Vehicle (V2V) communication fails or vehicles are entering/exiting the string. Section 5.3 considers CL identification of longitudinal vehicle dynamics. A comparison between Open-Loop (OL) and CL identification provided by dual YK is carried out when using a CACC system. Section 5.4 uses both controller reconfiguration and identification in a Multi Model Adaptive Control (MMAC) approach. The idea is to deal with vehicles heterogeneity in CACC-equipped string. Finally, some concluding remarks are given in section 5.5, determining if YK parameterization could serve as general control approach for secure responses in autonomous driving.

5.1 Experimental platform and simulation models

Different simulation models are used throughout the chapter for proving controller design in stable YK-based controller reconfiguration, dual YK-based CL identification and YK-based MMAC approach. For simulation purpose, the system identification tool provided by MATLAB has been used for emulating low-speed INRIA experimental platform. A Linear Time Invariant (LTI) model describing longitudinal dynamics is obtained for subsequent controller design. With the aim of providing results closer to reality on highways, a high-speed model is also introduced. The LTI identified high-speed Nissan Infinity model at California PATH in [Milanés et al., 2014] is presented, and used in subsequent control applications. Both low and high speed results are important in order to cover every possible situation a real vehicle would experience. Graphical programming tool MATLAB/Simulink is employed for simulating and analysing dinamical behaviour of the LTI model with designed controllers. Real tests have been performed using INRIA's low-speed electric vehicles called Cybabs. All data processing and control are experimentally implemented in C++ using RTMaps[1] prototyping software.

Section 5.1.1 gives details about the INRIA low-speed electric vehicle (Cycab), as well as all its equipment and the conceived perception system for autonomous driving. Identified LTI models for low and high speed applications are presented in sections 5.1.1.b and 5.1.2 respectively.

5.1.1 Cycab

Real tests are carried out with a low-speed electric vehicle called cycab (Fig. 5.1 shows a real test with three cycabs). Cycab motion is powered by four electric motors, each of them coupled to a wheel. From a kynematic point of view, cycabs can turn their rear wheels as a linear function of the steering angle of the front wheels [Sekhavat and Hermosillo, 2000]. Their speed is controlled by a low-level controller in function of the speed error. Speed is limited to $4m/s$.

Figure 5.1: Three cycabs during an experiment.

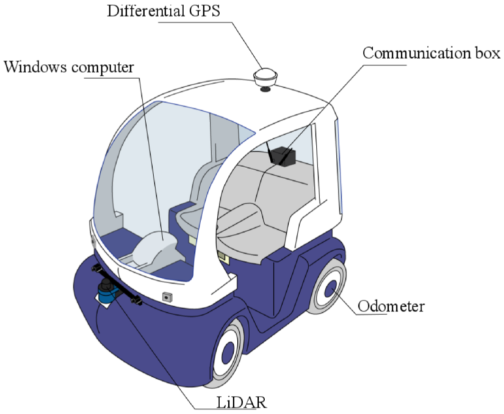
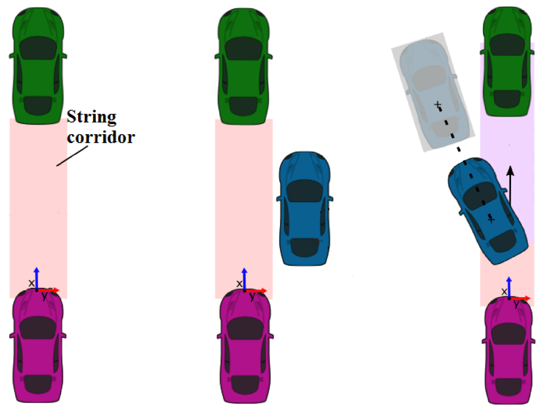


Figure 5.2: Cycab with equipment.

The mentioned platform is equipped with an odometer, an Ashtech Z-Xtrm RTK GPS, an Ibeo LiDAR, and a Yawarra ALiz-3 communication box. Odometer and Global Position Systems (GPS) measurements are both filtered with an Extended Kalman Filter (EKF), which improves localization robustness. A front Ibeo Light Detection And Ranging (LiDAR) sensor provides a 110° Field of view with an angular resolution of 0.125°. Even though the LiDAR provides several layers, only the most parallel to the ground is used to simplify the processing while remaining close to the industrial setup. For further details about the equipment position, see Fig. 5.2.

CACC applications are performed thanks to V2V communication and the perception system installed in the vehicle. For a tight string, all vehicles broadcast control velocity, output velocity and cut-in request using the 802.11g IEEE standard. Cut-in request refers to the communication protocol between vehicles in order to carry out gap opening. Merging maneuver based on communication protocol is an idea applied during the Grand Cooperative Driving Challenge (GCDC)

[1] https://intempora.com/



(a) String (b) Cut-in (c) Cut-out

Figure 5.3: Schematic view of the regular string scenario (a) and the detection schemes for cut-in (b) and cut-out (c) intentions.

2016 [Englund et al., 2016]. As gap opening only depends on a communication protocol, any vehicle with communication capabilities would be able to increase the gap without really being in a position to perform a cut-in maneuver. INRIA's perception system is extended with cut-in/out intention avoiding unnecessary or early gap opening. The gap will be opened only if the cut-in request has been received and the perception system has detected vehicle's intention to enter the string. Details about the perception system are found in the section below. Finally, longitudinal cycab dynamics identification is also presented.

5.1.1.a Perception system

Perception system is out of the scope of this Ph.D. thesis, but this section briefly describes the employed algorithm for detection of preceding, cut-in and cut-out vehicles in CACC applications. Preceding vehicle tracking is really important for obtaining the needed inter-vehicle distance in a classical CACC application. The proposed approach processes 2D LiDAR point clouds in real time to detect and classify objects in the environment as well as to detect the cut-in/out intentions of vehicles. In the literature similar setups are often used in urban scenarios [Wang et al., 2015] and car-following applications [Hsu et al., 2012].

To detect and classify objects in the vehicle coordinate frame the point clouds are first clustered in segments. Segments are grouped assuming an adjacent property, and objects are classified using geometrical heuristics (size, orientation, etc.). For temporal consistency, objects are tracked using a Kalman filter. Even if INRIA's perception system is conceived for more demanding applications, here only vehicle classification is required.

To identify the preceding vehicle, the polygon formed by the ego vehicle is projected along its trajectory and the closest vehicle intersecting this polygon is target locked.

To detect the intention of surrounding cars, there is a need to analyze their temporal behaviour, as it cannot be determined from a single snapshot. A schematic view of the string strategy and cut-in/out detection is provided in Fig. 5.3. A cut-in intention is detected if a vehicle - laterally close to the string - is seen stationary for a short duration denoted δ_t, as depicted in Fig. 5.3b. On the contrary, the preceding vehicle is considered to cut-out if its current trajectory deviates too much from the current string trajectory, as depicted in Fig. 5.3c.

5.1.1.b Identified model

Friction force between tire and road is the main reason why any vehicle moves [Pacejka, 2005]. It converts the motor torque provided by the electrical machine to longitudinal force, making possible forward vehicle movement. As physical parameters about the real experimental platform are missing, vehicle dynamics identification from experimental data seems a viable solution for obtaining a vehicle's model. Cycab's model is identified based on the response of the experimental platform to different speed changes applied to the low-level controller. This low-level controller is in charge of obtaining the needed electrical torque for making speed error zero. Real response depending on this reference velocity is plotted in Fig. 5.4. This experimental data is employed within MATLAB system identification tool, obtaining a LTI second order model as a compromise between simplicity and goodness. Cycab LTI model response corresponding to Eq. 5.1 is also in Fig. 5.4.

$$G(s) = \frac{1}{0.8768s^2 + 1.252s + 1} \tag{5.1}$$

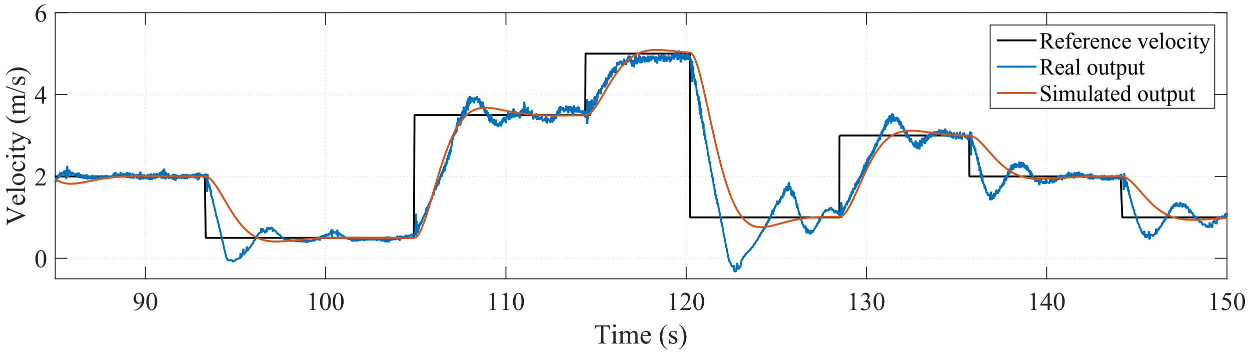


Figure 5.4: Velocity cycab response compared with identified model.

5.1.2 Nissan Infinity M56

Cycab is designed as a transportation system solution for the last mile in dense urban environments, so reaching high speeds is not necessary, reason why its velocity is limited. As vehicle control solutions should be tested in the whole velocity range (including high-speed solutions), a high-speed LTI model has been chosen from the literature.

For high-speed simulation purposes, the vehicle model introduced in [Milanés et al., 2014] is used. It is a second-order response with a time delay identification from a Nissan Infiniti M56. Vehicle LTI model is identified based on its response to different speed changes, resulting:

$$G(s) = \frac{1.136}{s^2 + 1.067s + 1.1385} \tag{5.2}$$

5.2 YK controller reconfiguration

In this section, two applications related to CACC are developed by using the YK stable controller reconfiguration presented in chapter 3. As concluded in section 3.4.3, among all YK-based structures for controller reconfiguration, structure 3 is employed because of its lower complexity and better transient behaviour.

In order to highlight the scientific contribution of these applications, section 5.2.1 summarizes the state of the art about CACC, and gives details about the control structure and the concept of string stability. CACC controller reconfiguration is carried out when some specific change occurs. Section 5.2.2 deals with the problem of non-available communication link with the preceding vehicle, while section 5.2.3 considers traffic perturbations as vehicles joining/leaving the string. Different CACC controller types are used in each application, so YK adaptability can be tested. In both applications, two different CACC controllers are designed. Those controllers correspond to two extreme situations, focusing in how to change the scalar factor γ in the YK-based controller reconfiguration in order to improve system performance.

5.2.1 Cooperative adaptive cruise control

Recently, the International Council on Clean Transportation has estimated that over the next two decades vehicle ownership is expected to increase 7 million only in the European Union (EU) [ICCT, 2016]. Therefore, road transport will have to deal with these figures. Among the associated problems, drivers will find more congested roads, resulting in an enormous waste of fuel and productivity together with health problems. According to the European Commission, congestion costs are equivalent to 1% of Gross Domestic Product (GDP)–in other words more than the EU budget [Commision, 2012].

Since building new infrastructure is no longer an appropriate solution, more intelligent and efficient options result of ITS. Specifically, related to traffic congestion, intelligent longitudinal speed control is a suitable system to improve congestion in highways, through homogeneous speed on the part of the driver and shorter intervehicle distances. Adaptive Cruise Control (ACC) [Marsden et al., 2001] is a commercial system already implemented in production vehicles. An ACC system can track the preceding vehicle, measuring the actual distance and the ego-vehicle velocity. These inputs allow the system to maintain a selected time gap, calculating the required acceleration or deceleration to reach the desired velocity or to prevent a collision. Recently, research focuses on the cooperative version of the system, so-called Cooperative ACC (CACC) [Ploeg et al., 2011a]. V2V communication is added to the existing ACC system, improving traffic flow through the formation of a tighter string of vehicles.

CACC research has received a lot of attention in last years [Dey et al., 2016]. Related to real vehicle implementations, first european results can be found in the Connect Drive Dutch project where a fleet of six Toyota Prius were equipped with CACC capabilities [Ploeg et al., 2011b] using a Proportional-Derivative (PD) feedback/feedforward controller [Ploeg et al., 2011a]. Later, the first Grand Cooperative Driving Challenge (GCDC) held in The Netherlands in 2011 showed for the very first time several vehicles from different institutions performing a two-lane CACC string. Control algorithms were based either in the already used PD feedback/feedforward structure [Lidstrom et al., 2012] or Model Predictive Control (MPC) algorithm [Kianfar et al., 2012]. In United States, the California PATH carried out a four vehicle string demonstration on California highways showing the benefits of CACC against ACC in real traffic using a PD feedback/feedforward structure [Milanés et al., 2014]. The latter is selected for this work due to its simplicity and proven experimental implementation; making it ideal for embedded systems. Details are in section 5.2.1.a.

Traffic flow improvement is directly related to string stability. When tighter car-following policies are implemented–i.e. CACC–, string stability fulfilment plays a key role to ensure a proper response of the vehicle string [Swaroop, 1997]. On string stable car-following, the impact of traffic perturbations is attenuated upstream, improving traffic flow and reducing traffic jams. String stability is linked with the implemented car-following policy and vehicle dynamics. An homogeneus string of vehicles is considered along section 5.2. Most algorithms are based on the so-called constant time gap policy [Swaroop and Rajagopal, 2001], which references the ideal spacing (proportional to ego-vehicle speed) between vehicles, improving string stability with respect to constant spacing algorithms.

Traffic flow benefits in function of the ACC-equipped vehicles market penetration have been widely studied. [Kesting et al., 2008] [Kesting et al., 2007] stated that a 25% of market penetration could remove congestion. Recent studies [Shladover et al., 2012] proposes a different ACC model that provides no traffic flow benefits even with a market penetration of 100%. This model is based on the field test of ACC driven by 16 drivers from the general public in [Nowakowski et al., 2010b]. These drivers were encouraged to select the time gap setting that they preferred, resulting in a time-gap close to manual driving. As quantitative example, the highway capacity with every vehicle driven manually is 2050 veh/h, while with ACC-equipped vehicles the capacity increases up to 2200 veh/h [Werf et al., 2002].

In contrast, the CACC system on the traffic flow characteristics presents much more optimistic results. [Arem et al., 2006] concluded that CACC improves the highway capacity when the penetration rate is greater than 60%, obtaining better results on high traffic volume because of more vehicles participate in the string. The CACC model in [Shladover et al., 2012] validates these results, showing a maximum lane capacity of 4000 veh/h under 100% CACC-equipped vehicles condition. This model is again based on the chosen time-gap by the general public in the field test in [Nowakowski et al., 2010b]. CACC can double the highway capacity in the ideal situation.

It is clear that the market penetration of CACC systems would progressively occur, making necessary to work with mixing traffic situations. As the highway capacity is sensitive to the percentage of CACC-equipped vehicles, it is important to preserve the CACC behaviour when there is no communication with the preceding vehicle, but with another vehicle further on. For instance, when a vehicle loses communication capability into a string previously formed by CACC-equipped vehicles. As the communication with the preceding vehicle is no longer available, existing literature on the field (see [Milanés and Shladover, 2014] and [Ploeg et al., 2013] for details) degrades the system to a conventional ACC, removing the strong impact of CACC systems on highway capacity.

Finally, CACC systems capability of improving string stability has been widely demonstrated, but the system stability when the string structure is modified (either a new vehicle is entering or leaving the string) is still an open research field. First efforts on this direction using real vehicles were carried out by California PATH, where an algorithm able to handle vehicles cutting in the string was implemented [Milanés and Shladover, 2016]. However, the stability during time gap transition was not demonstrated.

YK-based controller reconfiguration is proposed as solution for covering the two present gaps in CACC state-of-the-art. Section 5.2.2 avoids ACC degradation, by employing the V2V communication with a vehicle ahead (different from the preceding one). γ is designed in order to improve traffic flow and vehicle response. Section 5.2.3 explains relation bewteen γ and time gap to guarantee stability when vehicles are entering/exiting the string.

5.2.1.a Control structure and string stability

A string of w vehicles driving in the same lane is considered. j determines the order of a vehicle inside of the string $j \in [1, w]$. Vehicle j denotes ego-vehicle, vehicle $j-1$ preceding vehicle and $j = 1$ leader vehicle. A schematic representation of a CACC-equipped string of vehicles is in Fig. 5.5, where d^j is the distance between vehicle j and $j-1$ seen as the difference between absolute positions x^j and x^{j-1}; and v^j is the velocity of the vehicle j. Control velocities v_c are sent through communication between vehicles. A reference distance d_r^j is followed for each vehicle.

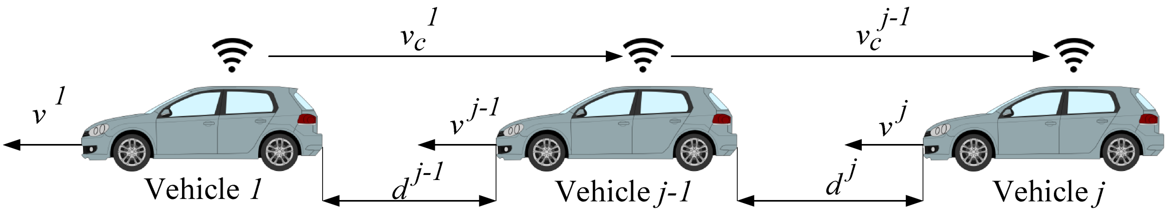


Figure 5.5: Schematic representation of a CACC-equipped string of vehicles.

This reference distance comes from a constant time gap spacing policy [Swaroop and Rajagopal, 2001]:

$$d_r^j = d_{std}^j + h_i^j v^j \tag{5.3}$$

where d_{std}^j is the distance at standstill, and h_i^j the constant time gap. Time gap can be seen as the time for vehicle j to reach vehicle $j-1$ at constant velocity. Notice that subindex i, as in chapter 3, refers to different controller or time gap values.

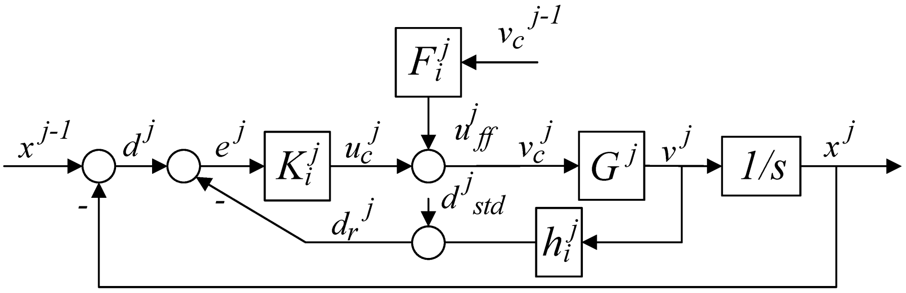


Figure 5.6: Classical PD-based feedback/feedforward CACC control structure.

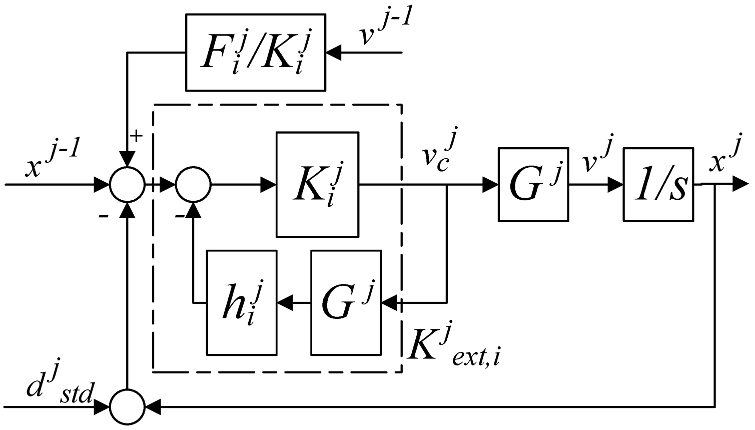


Figure 5.7: Modified CACC control structure.

Figure 5.6 shows a PD feedback/feedforward structure for CACC systems (see [Ploeg et al., 2011a] for more details), where G^j represents the vehicle model from control velocity v_c^j to output velocity v^j; K_i^j a stabilizing PD controller able to regulate the distance error $e^j = d^j - d_r^j$ to zero (see control law in Eq. 5.4); and v_c^{j-1} the control velocity of vehicle $j-1$ received and processed by means of a feedforward filter F_i^j in order to get a tighter string of vehicles. Feedforward transfer function is in Eq. 5.5. PD's output u_c^j and feedforward's output u_{ff}^j addition results in the control velocity v_c^j.

$$K_i^j(s) = \frac{U_c^j}{E^j} = k_{p,i}^j + k_{d,i}^j s \tag{5.4}$$

$$F_i^j(s) = \frac{1}{1 + h_i^j s} \tag{5.5}$$

Highway capacity improves when tighter gap policies can be adopted whereas keeping string stability. String stability is defined as the attenuation of disturbances along the string of vehicles. A sufficient condition for string stability is given in [Rajamani, 2011], which means that the absolute position of each vehicle must not increase as it propagates through the string. Condition is equivalent to the following equation:

$$\|X^j/X^{j-1}\| \leq 1 \quad for \quad j > 1 \tag{5.6}$$

According to [Naus et al., 2010], for a PD-based feedback/feedforward CACC system, string stability function results:

$$\frac{X^j}{X^{j-1}} = \frac{D^j + (1 + h_i^j s)G^j K_i^j}{(1 + h_i^j s)(1 + (1 + h_i^j s)G^j K_i^j)}, \quad for \quad i > 1 \tag{5.7}$$

Under the ideal situation where the communication delay is null ($D^j = 1$), string stability yields:

$$\frac{X^j}{X^{j-1}} = \frac{1}{(1 + h_i^j s)}, \quad for \quad i > 1 \tag{5.8}$$

Therefore, string stability is guaranteed for any $h_i^j > 0$. The existence of communication with preceding vehicle makes vehicle response faster, allowing any time gaps without compromising the string stability. On the contrary, as ACC system has not communication, the achievable stable time gap is longer, even close to the time gap of a manual driver. It explains why even with a high market penetration of ACC-equipped vehicles the traffic flow improvement effects are not visible.

Finally, the classical feedback/feedforward CACC system is modified in Fig. 5.7 to include the time gap h_i^j into the controller. Thereby, time gap is changed when doing YK controller reconfiguration. The extended controller is shown in Eq. (5.9). Feedforward filter is changed to F_i^j/K_i^j.

$$K_{ext,i}^j(s) = \frac{K_i^j}{1 + G^j h_i^j K_i^j} \tag{5.9}$$

5.2.2 Advanced cooperative adaptive cruise control

CACC provides significant traffic flow improvements when a V2V communication link exists with the preceding vehicle (upper plot of Fig. 5.8 depicts the situation), but it degrades to ACC when this communication link is no longer available [Milanés and Shladover, 2014] [Ploeg et al., 2013] (middle plot of Fig. 5.8). As quantitative example, [Ploeg et al., 2011a] showed how the minimum time gap string stable increases from $0.7s$ to $3.16s$ when communication is not available. This

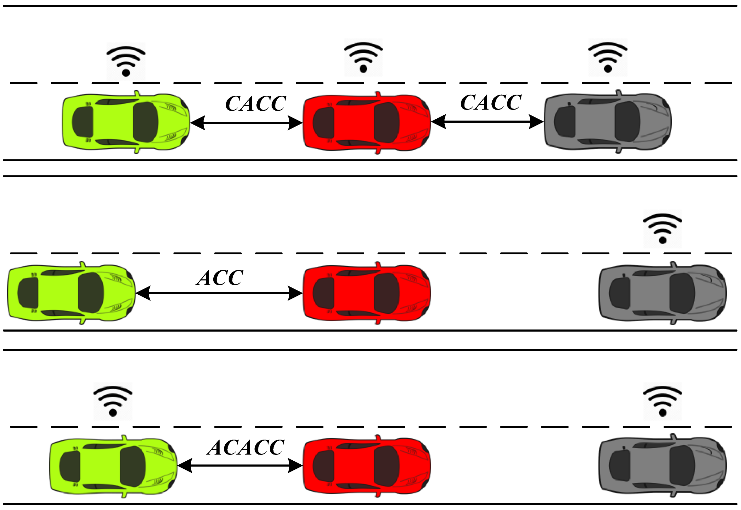


Figure 5.8: ACACC scenario.

degradation occurs even if the information from another V2V-equipped vehicle ahead (different to the preceding one) is still available. This application presents a novel car-following control system–Advanced Cooperative Adaptive Cruise Control (ACACC)–that benefits from the existing communication with this vehicle ahead in the string, reducing intervehicle gap whereas keeping stability.

Since the time gap policy has a significant influence on the highway capacity, the proposed control structure provides a hybrid behaviour between two CACC controllers with different time gaps. YK parameterization is used to obtain stable transitions between them. How to change the switching signal γ is analysed in order to provide traffic flow improvement while using the existing communication link with the closer ahead communicated vehicle. As the system is not degraded to ACC, the highway capacity previously accomplished by the CACC string of vehicles is better preserved, as well as the shockwave effect is better mitigated. Bottom part of Fig. 5.8 shows an ACACC scenario; where the red vehicle could be either a conventional driver or an ACC-equipped vehicle, both without communication capabilities.

The proposed system has been simulated for highway velocities by using the vehicle model in subsection 5.1.2. ACACC has been tested under different situations, being the preceding vehicle either a conventional or an ACC-equipped one. For all cases, the system exhibits a good performance, achieving tighter string of vehicles. These simulations results are later validated on the INRIA experimental platform described in subsection 5.1.1. ACACC improves traffic flow capacity.

The application is structured as follows: Section 5.2.2.a introduces the problem formulation. The YK-based control algorithm able to deal with the ACACC scenario is detailed in section 5.2.2.b. Section 5.2.2.c shows simulation results. The control algorithm is validated on the low-velocity experimental platform in section 5.2.2.d. Finally, some concluding remarks are given in section 5.2.2.e.

5.2.2.a Problem formulation

A string of three vehicles $w = 3$ is considered. Problem formulation focuses on ego-vehicle situation according to V2V communication availability with vehicle $j - 1$ and vehicle 1 in the string. V2V communication between vehicle j and vehicle $j - 1$ is defined as C^{j-1}. Communication between vehicle j and vehicle 1 is defined as C^1.

In the literature, communication availability with vehicle $j - 1$ determines the use of ACC or CACC controllers in the vehicle j, and therefore its time gap h_i^j:

- When C^{j-1} exists, a regular CACC controller can be used. Faster responses can be achieved, allowing tighter string of vehicles by employing a time gap h_{CACC}–i.e.
  $\exists C^{j-1} \rightarrow CACC \quad where \quad h_i^j = h_{CACC}$.
- If C^{j-1} is no longer available: the system degrades to a conventional ACC algorithm [Navas et al., 2016]. Consequently, benefits of CACC systems on highway capacity are removed because a longer time gap $h_{ACC} > h_{CACC}$ needs to be set to ensure string stability– i.e. $\nexists C^{j-1} \rightarrow ACC \quad where \quad h_i^j = h_{ACC} > h_{CACC}$.

These situations have been extensively studied (see [Marsden et al., 2001] for ACC and [Ploeg et al., 2011a] for CACC), but potential benefits of using C^1 when C^{j-1} is not available have not been further investigated. Speed oscillations on the string (i.e. the difference between v^1 and v^{j-1}) is limited to $5m/s$ according to [Milanés and Shladover, 2014]. Experimental results demonstrated that over this value, drivers disengage ACC system due to its degraded performance.

The objective of the ACACC controller is to enhance traffic flow when there is no communication with vehicle $j - 1$, by taking the information from the closer V2V-equipped vehicle ahead, in that case, vehicle 1. Since V2V communication is always available (no matter from which vehicle comes from), a CACC control structure is proposed. ACACC is composed by two CACC controllers with different time gaps: A CACC controller with a short time gap $h_i^j = h_{CACC}$ so-called $CACC_{STG}$ and a CACC controller with a longer time gap $h_i^j = h_{ACC}$ so-called $CACC_{LTG}$. The proposed ACACC algorithm benefits from the YK parameterization to provide a hybrid response between both controllers. The regulation between the effect of each controller through γ in the control command is based on a correlation between the speed received via V2V communication and the one detected by the on-board ego-vehicle sensor (i.e. radar). The correct tuning of γ with a maximum speed oscillation of $5m/s$, assures stability and improves traffic flow.

In brief, the application here described works when C^{j-1} is lost but C^1 is available, providing a control structure of the form:

$$\begin{gathered} if \quad \nexists C^{j-1} \& \exists C^1 \\ ACACC = CACC_{STG}(1-\gamma) + CACC_{LTG}\gamma; \quad \gamma \in [0,1] \end{gathered} \tag{5.10}$$

where $h_i^j \in [h_{CACC}, h_{ACC}]$. Notice how the time gap would be lower than degrading the system to a regular ACC.

5.2.2.b Control algorithm

ACACC is composed by two different CACC controllers: $CACC_{STG}$ and $CACC_{LTG}$. Time gap values h_{CACC} and h_{ACC} are chosen according to the general public accepted time gaps for CACC and ACC systems [Nowakowski et al., 2010b]. For CACC, the shortest gap is set at $0.6s$, while for ACC is set to $1.5s$.

For each of these controllers, the modified CACC controller presented in Fig. 5.7 is used to allow stable transitions between controllers with different h_i^j, $k_{p,i}^j$ and $k_{d,i}^j$. $CACC_{STG}$ has the index $i = 0$, while $CACC_{LTG}$ has the index $i = 1$. The information related to both controllers is summarized in Table 5.1. Notice how the values of $k_{p,0}^j$, $k_{p,1}^j$, $k_{d,0}^j$ and $k_{d,1}^j$ will depend on the accepted time gaps by drivers and the vehicle model G^j.

Once $CACC_{STG}$ and $CACC_{LTG}$ are defined, doubly coprime factors for pairs $(K_{ext,0}^j, G^j/s)$ and $(K_{ext,1}^j, G^j/s)$ are obtained following guidelines in Theorem 3.7. Then, it is possible to change from one to the other, without losing stability by obtaining the parameter Q' through Theorem 3.13. Figure 5.9 modifies the YK control structure 3 in Fig. 3.6 to allow CACC controller modification. Please notice how the structure only differs in adding the filtered communication link with the closest V2V-equipped vehicle, in that case, communication link C^1.

Once the YK control structure for CACC is obtained, a different percentage of each of the controllers is applied depending on the traffic situation. The percentage is chosen through the YK gamma γ. Gamma tuning is chosen in order to improve traffic flow. A simple decision-making system is used when C^{j-1} does not exist, but there is communication with vehicle 1, C^1. γ is modified according to the following equation:

$$\gamma = 0.033(v_s^{j-1} - v^1) + 0.5 \quad \text{if} \mid v^1 - v_s^{j-1} \mid < 5m/s$$

where v_s^{j-1} is the velocity of the vehicle $j-1$ obtained through the on-board sensor systems, and v^1 is the velocity of vehicle 1 received through C_1. A maximum speed oscillation of $\mid v^1 - v_s^{j-1} \mid = 5m/s$ is considered as the operation range for the present application.

Table 5.1: CACC parameters.

	$K_{ext,u}^j$	$k_{p,i}^j$	$k_{d,i}^j$	h_i^j
$CACC_{STG}$	$K_{ext,0}^j$	$k_{p,0}^j$	$k_{d,0}^j$	$h_0^j = h_{CACC} = 0.6s$
$CACC_{LTG}$	$K_{ext,1}^j$	$k_{p,1}^j$	$k_{d,1}^j$	$h_1^j = h_{ACC} = 1.5s$

Different traffic situations are pictured for understanding the tuning of γ:

When vehicle $j-1$ has a behaviour similar to vehicle 1, γ is modified to response faster to speed changes.

- If vehicle 1 accelerates, v^1 will be higher than v_s^{j-1}. So, γ decreases, making the gap time shorter.
- If vehicle 1 brakes, v^1 will be lower than v_s^{j-1}. So, γ increases, making the gap time longer.
- When both vehicle 1 and vehicle $j-1$ have similar velocities, γ results in an intermediate value between both controllers.

Advantages of using this γ are analysed in sections below through simulation and experimental results.

5.2.2.c Simulation results

This section presents the ACACC performance at high speeds (i.e. highway scenario). For simulation purposes, the vehicle model introduced in section 5.1.2 is used. The extended control structure in section 5.2.1.a is employed, obtaining two different CACC controllers: $K_{ext,0}^j$ and $K_{ext,1}^j$. The

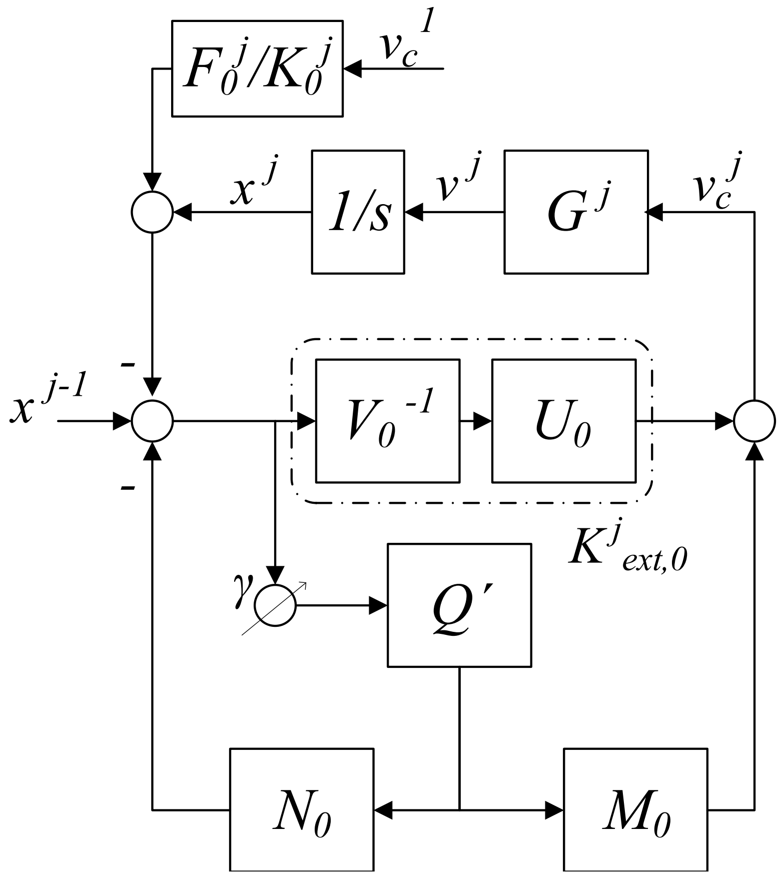


Figure 5.9: YK control structure for modifying CACC controllers online. ACACC.

Table 5.2: Nissan Infiniti M56 CACC parameters. ACACC.

	$K^j_{ext,i}$	$k^j_{p,i}$	$k^j_{d,i}$	h^j_i	$d^j_{std,i}$
$CACC_{STG}$	$K^j_{ext,0}$	$k^j_{p,0} = 0.45$	$k^j_{d,0} = 0.25$	$h^j_0 = h_{CACC} = 0.6s$	$d^j_{std,0} = 5m$
$CACC_{LTG}$	$K^j_{ext,1}$	$k^j_{p,1} = 0.45$	$k^j_{d,1} = 0.25$	$h^j_1 = h_{ACC} = 1.5s$	$d^j_{std,1} = 5m$

controller gains $k^j_{p,i}$ and $k^j_{d,i}$ correspond to those already designed for a Nissan Infinity M56 in [Milanés et al., 2014]. $CACC_{STG}$ and $CACC_{LTG}$ information is summarized in Table 5.2. Once both controllers are defined, stability depending on γ is studied in the corresponding section. Finally, the algorithm performance is tested by using as vehicle $j-1$ a non V2V-equipped vehicle. It is modeled as a human driver–using the Intelligent Driver Model (IDM) [Kesting et al., 2010]–or equipped with an ACC system.

Stability CL stability for the designed ACACC is studied in function of γ. A necessary and sufficent condition to ensure stability for a feedback/feedforward control loop as the one in Fig.

5.9 is in Theorem 3.4. As feedforward filter is stable, CL stability condition yields:

$$\begin{gathered}\left[\begin{array}{cc} I & -K_0(Q') \\ -G^j/s & I \end{array}\right]^{-1} \in \mathcal{RH}_\infty = \\ = \frac{K_0(Q')G^j/s}{1+K_0(Q')G^j/s} = \frac{\frac{K^j_{ext,0}+M_0\gamma Q'}{1+N_0\gamma Q'}G^j/s}{1+\frac{K^j_{ext,0}+M_0\gamma Q'}{1+N_0\gamma Q'}G^j/s} \in \mathcal{RH}_\infty\end{gathered} \tag{5.11}$$

CL poles are shown for different values of γ in Table 5.3. As in section 3.4.1, CL poles during the transition are the combination of CL poles of $(G^j/s, K^j_{ext,0})$ and $(G^j/s, K^j_{ext,1})$. CL stability is ensured for every value of γ. String stability is not studied as there is not communication with the preceding vehicle.

Table 5.3: CL poles $(G^j/s, K_0(Q'))$. YK controller reconfiguration between $K^j_{ext,0}$ and $K^j_{ext,1}$. Nissan Infinity M56.

γ	CL poles
$\gamma = 0.0$	$[-6.283e^7, 0, -0.6185 \pm 1.0308i]$
$\gamma = 0.1$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.2$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.3$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.4$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.5$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.6$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.7$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.8$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 0.9$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7465 \pm 1.1609i, -0.6185 \pm 1.0308i]$
$\gamma = 1.0$	$[-6.283e^7, 0, -0.7465 \pm 1.1609i]$

Using IDM as preceding non V2V-equipped vehicle First simulations consider a human driven vehicle (modeled by the IDM) as vehicle $j-1$ without communication. IDM is a well-known car-following model in the traffic flow simulation literature [Kesting et al., 2010]. It defines an acceleration a_{IDM} as a continous function incorporating different driving modes for all velocities in freeway and urban traffic. The acceleration profile depends on desired v_0 and actual v velocities, free acceleration exponent δ, mimimum spacing s_0, actual distance s, desired time gap T, maximum acceleration value a, and desired deceleration b (see Eq. 5.12). By choosing different desired velocities v_0, impatient or relaxed drivers can be emulated. The parameters in [Milanés and Shladover, 2014] are used as reference values, and they are shown in Table 5.4.

$$a_{IDM} = a\left[1-\left(\frac{v}{v_0}\right)^\delta - \left(\frac{s_0+max(0, vT+\frac{v\Delta v}{2\sqrt{ab}})}{s}\right)\right] \tag{5.12}$$

Table 5.4: IDM parameters.

Parameter	Value
desired velocity	v_0
δ	4
T	$1.1s$
s_0	$0m$
a	$1m/s^2$
b	$2m/s^2$

Figure 5.10 depicts the performance of the ACACC controller in comparison with ACC and CACC (assuming V2V capabilities in the IDM vehicle, only in the CACC case) controllers when the IDM desired velocity is $v_0 = 33.33m/s$. The comparison is between the perfect CACC situation– V2V communication with preceding vehicle exists; classical ACC degradation and the proposed ACACC system (both without communication with preceding vehicle). The top graph plots the vehicles' speeds. The second graph plots the vehicles' accelerations during the simulation. The third graph shows the relative distance between vehicles. The bottom graph represents how γ is modified together with the time gap. For notation, vehicle 1 (solid magenta line) is a V2V-equipped vehicle and the first vehicle during the whole simulation; vehicle $j-1$ (solid cyan line) is the one that starts the string in the second position as IDM; and, the vehicle j or follower is either an ACC-equipped vehicle (solid red line), a CACC-equipped vehicle (solid green line, assuming than the IDM is V2V-equipped for comparison purposes) or an ACACC-equipped vehicle (solid blue line).

The value of γ is always around 0.5, changing with acceleration or braking phases. The performance of the ACACC system is a hybrid response between $CACC_{STG}$ and $CACC_{LTG}$. Its response comes earlier to changes in leader speed v^1 than the ACC/CACC systems. For instance, when vehicle 1 is braking at $300s$, the ACACC system brakes practically at the same time, while ACC and CACC react $5s$ later. Besides, the system is more damped while speed changes take place, better preserving string stability. Finally, the main objective of the present application is also fulfilled, instead of degrading to an ACC system, ACACC takes C^1 for making the string of vehicles tighter, improving highway capacity. The relative distance corresponds to a spacing policy between $CACC_{STG}$ and $CACC_{LTG}$ responses.

Figure 5.11 shows a second simulation using IDM. The desired velocity parameter v_0 from [Milanés and Shladover, 2014] is changed from $33.33m/s$ to $28m/s$, emulating a very slow vehicle dynamic on the non V2V-equipped vehicle $j-1$ when tracking changes in v^1. Since the IDM model is not correctly following vehicle 1 response, the ACACC system has a closer behaviour to $CACC_{LTG}$. This behaviour is appreciated on the lower plot of Fig. 5.11.

Using an ACC-equipped vehicle as preceding non V2V-equipped vehicle [Milanés and Shladover, 2014] compared the car following performance among ACC and CACC systems for a Nissan Infiniti M56s (section 5.1.2). Experimental results are used to obtain car-following models for representing the production ACC and the new designed CACC controller, highlighting how the production ACC results string unstable, amplifying the speed changes of the preceding vehicle. A simulation using the ACC/CACC car-following models presented in [Milanés and Shladover,

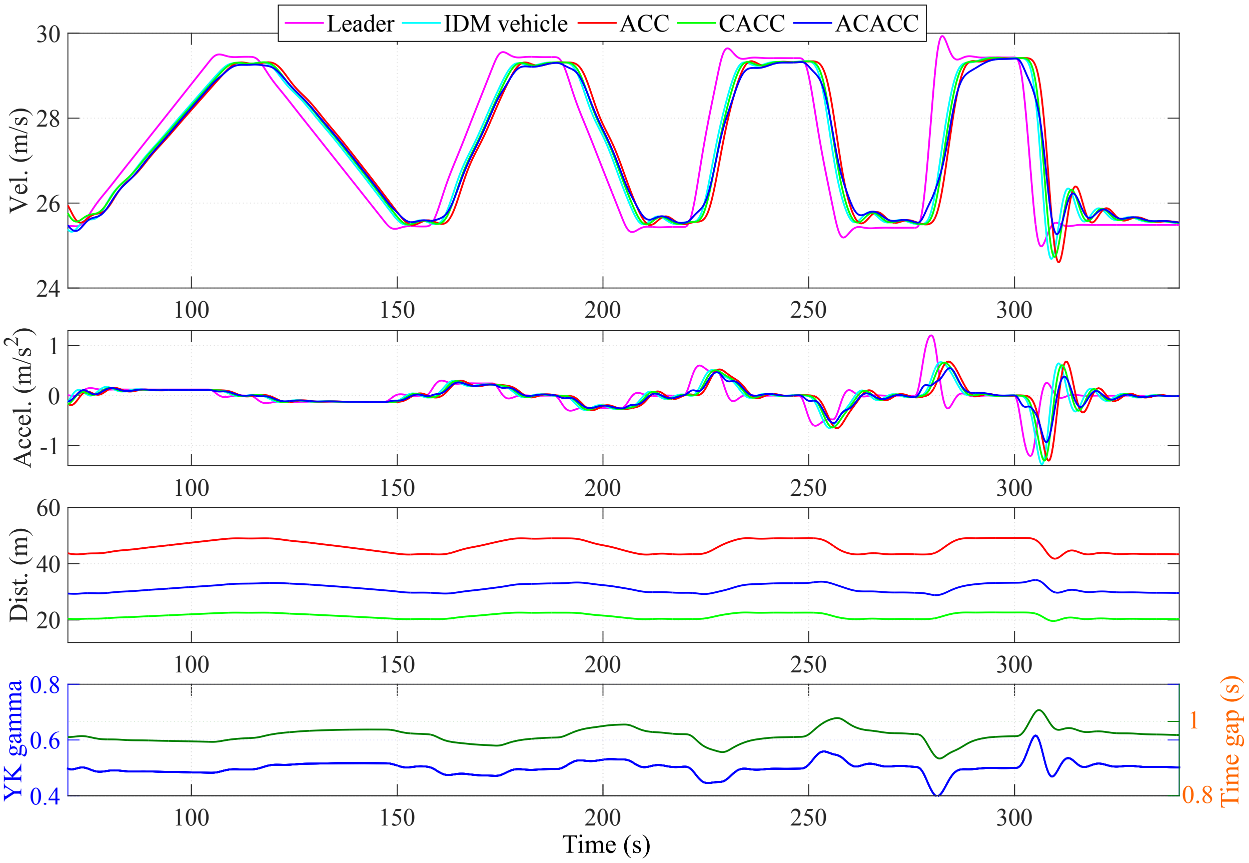


Figure 5.10: Simulation results comparison among car-following policies using CACC, ACC and ACACC controllers when a non V2V-equipped vehicle is in front. The non V2V-equipped is a conventional driver emulated by a IDM with the desired velocity of $33.33m/s$.

2014] is carried out. Specifically, the more the ACC-equipped vehicles in the string, the bigger the amplification. Here, instead of degrading the system to another ACC, the system is changed to ACACC with C^1.

Figure 5.12 depicts the performance of the ACACC controller in comparison with ACC and CACC controllers when an ACC-equipped vehicle is the vehicle $j-1$. For notation, vehicle 1 (solid magenta line) is a CACC-equipped vehicle following a speed reference; vehicle $j-1$ (solid cyan line) is the one that starts the string in the second position with an ACC as controller; and, vehicle j or follower is either an ACC-equipped vehicle (solid red line), a CACC-equipped vehicle (solid green line, only possible if vehicle $j-1$ is V2V-equipped) or an ACACC-equipped vehicle (solid blue line).

ACACC system significantly reduces speed oscillation introduced by the ACC-equipped vehicle, providing string stability but also increasing traffic flow by reducing intervehicle distances. This simulation represents the closer behaviour to real traffic environment with a high penetration of ACC-equipped and CACC-equipped vehicles, giving an insight into the potential benefits of ACACC.

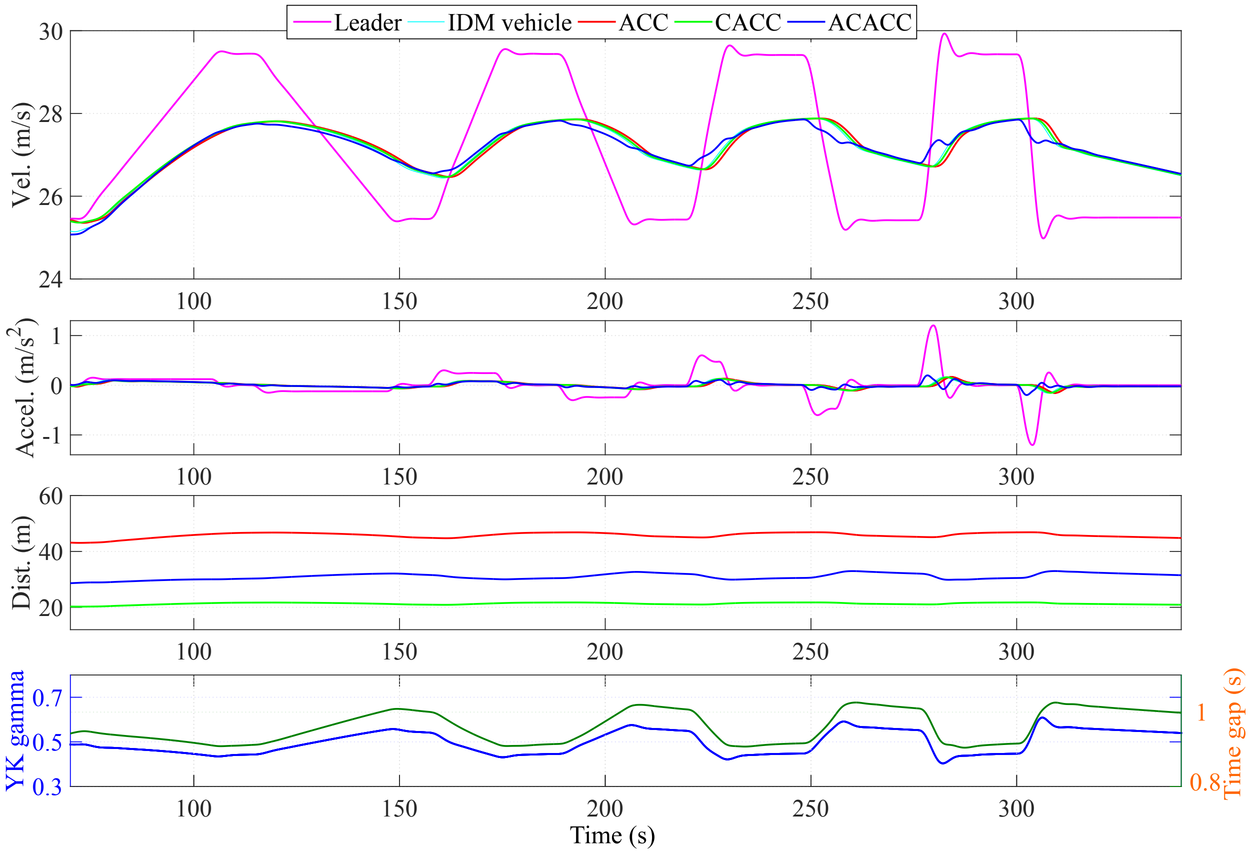


Figure 5.11: Simulation results comparison among car-following policies using CACC, ACC and ACACC controllers when a non V2V-equipped vehicle is in front. The non V2V-equipped is a conventional driver emulated by an IDM with the desired velocity of $28m/s$.

Table 5.5: Cycab CACC parameters. ACACC.

	$K^j_{ext,i}$	$k^j_{p,i}$	$k^j_{d,i}$	h^j_i	$d^j_{std,i}$
$CACC_{STG}$	$K^j_{ext,0}$	$k^j_{p,0} = 1.5$	$k^j_{d,0} = 0.2$	$h^j_0 = h_{CACC} = 0.6s$	$d^j_{std,0} = 4m$
$CACC_{LTG}$	$K^j_{ext,1}$	$k^j_{p,1} = 1.5$	$k^j_{d,1} = 0.2$	$h^j_1 = h_{ACC} = 1.5s$	$d^j_{std,1} = 4m$

5.2.2.d Experimental results

Three cycabs are used as experimental vehicles. As ACACC has been designed for high-velocities, the proposed system is adapted accordingly to these low-speed platforms. The vehicle dynamic model for evaluating the performance of ACACC is presented in Eq. 5.1. The extended control structure in section 5.2.1.a is employed, obtaining again two different CACC controllers: $K^j_{ext,0}$ and $K^j_{ext,1}$. Control gains were firstly designed using tuning tools from MATLAB/Simulink and guidelines in [Naus et al., 2010]. The objective is to maintain the desired time gap to the preceding vehicle with smoothness and accuracy. A final tuning was carried out in the experimental platform in order to obtain the best quality riding from the driver point of view. The resulting controller gains are shown in Table 5.5.

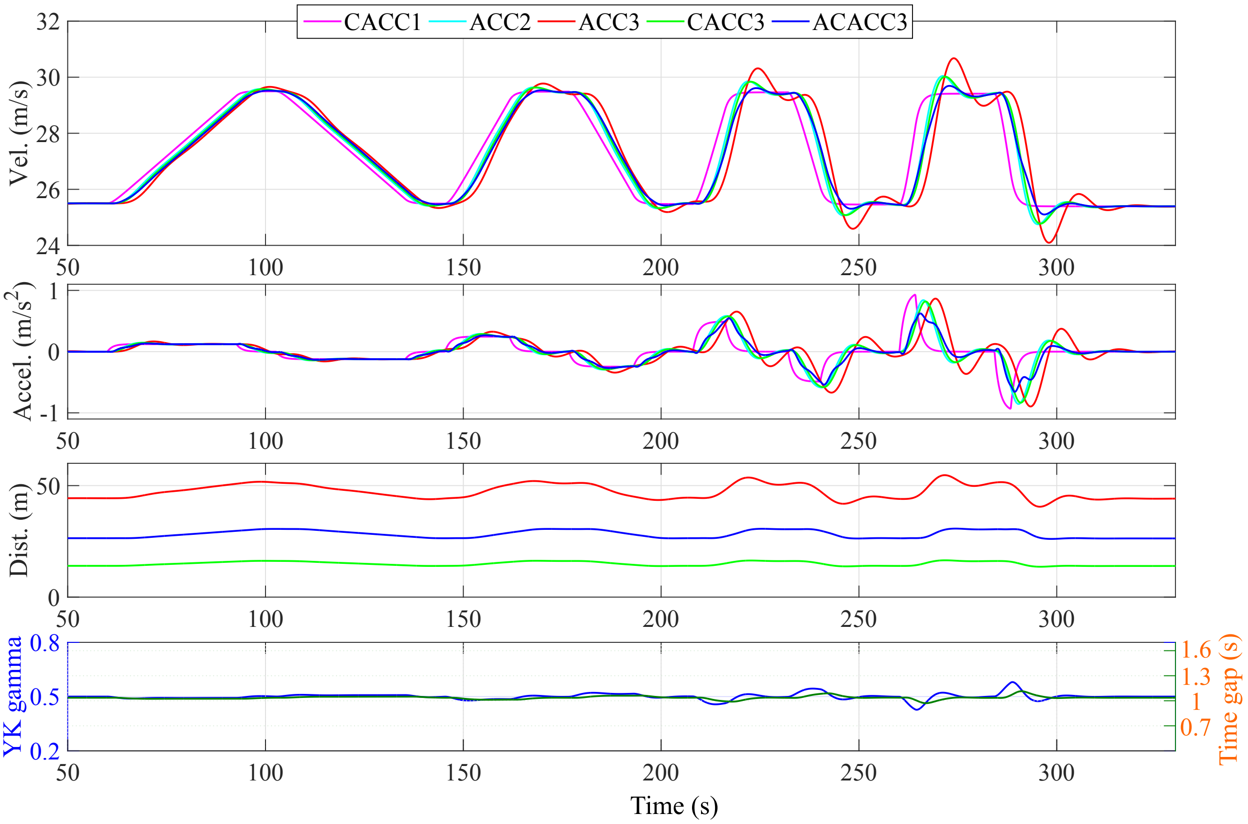


Figure 5.12: Simulation results comparison among car-following policies using CACC, ACC and ACACC controllers when a non V2V-equipped vehicle is in front. The non V2V-equipped is an ACC-equipped vehicle.

Stability CL stability for the designed ACACC is studied in function of γ with Eq. 5.11. Controller gains in Table 5.5 and model in Eq. 5.1 are considered in order to obtain the CL poles shown in Table 5.6. CL stability is preserved for every value of γ.

Using IDM as preceding non V2V-equipped vehicle The non V2V-equipped vehicle $j-1$ is a cycab with an IDM controller with the desired velocity of $33.33m/s$. As IDM controllers depend on the ego-velocity and the velocity of its preceding vehicle, both signals are scaled with a factor of 10 for obtaining a low-velocity behaviour. In the same way, the velocities used in the tuning of γ are scaled for proving the correct performance of the approach.

Figure 5.13 depicts the performance of the scaled ACACC controller in comparison with ACC and CACC controllers when an IDM-equipped cycab is the vehicle $j-1$. The top graph plots the velocity responses of the three cycabs in the string. The second depicts the different accelerations of these vehicles during the test. The third graph shows the relative distances to the preceding vehicle. And the bottom graph represents how γ is changed over time, together with the time gap.

For notation, vehicle 1 (solid magenta line) is a cycab with a speed profile that goes from $1m/s$ to $3m/s$ or viceversa with increasing steps of acceleration/deceleration; vehicle $j-1$ (solid cyan line) is the second cycab in the string with an IDM controller; and, vehicle j or follower in third position is either ACC (solid red line), CACC (solid green line) or ACACC (solid blue line).

The performance of the ACACC is well-proved. The hybrid behaviour between $CACC_{STG}$

Table 5.6: CL poles $(G^j/s, K_0(Q'))$. YK controller reconfiguration between $K^j_{ext,0}$ and $K^j_{ext,1}$. Cycab.

γ	CL poles
$\gamma = 0.0$	$[-6.283e^7, 0, -0.7825 \pm 1.2469]$
$\gamma = 0.1$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.2$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.3$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.4$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.5$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.6$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.7$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.8$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 0.9$	$[-6.283e^7, -6.283e^7, 0, 0, -0.7825 \pm 1.2469, -0.8850 \pm 1.7099i]$
$\gamma = 1.0$	$[-6.283e^7, 0, -0.8850 \pm 1.7099i]$

and $CACC_{LTG}$ responses of the ACACC controller is preserved. Its response comes earlier when vehicle 1 is accelerating or braking. And the relative distance to vehicle $j-1$ is shorter than with the classical degradation to ACC, improving the capacity of the road.

Nevertheless, it is noteworthy to mention how at low-velocities the adapted time gap by the system is more sensitive to any velocity change, as well to the standstill distance. It explains why the variation of the real time gap time through γ becomes more complex for the low-velocity test (see bottom part in Fig. 5.13).

Using an ACC-equipped vehicle as preceding non V2V-equipped vehicle A scenario where vehicle $j-1$ is an ACC-equipped cycab is carried out. Please notice that the ACC shown here is not string unstable. The performance of the ACACC controller can be accomplished with a stable ACC without endangering the experimental platform.

Figure 5.14 depicts the performance of the ACACC controller compared to ACC and CACC when the non V2V-equipped vehicle is an ACC-equipped cycab. For notation, vehicle 1 (solid magenta line) is a cycab with the speed profile showed in the previous section; vehicle $j-1$ (solid cyan line) is the second cycab with an ACC controller; and vehicle j or follower in third position is either ACC (solid red line), CACC (solid green line) or ACACC (solid blue line).

These low-speed tests demonstrate the good performance of the proposed system. ACACC controller responses faster to speed changes, while preserving better the string stability and improving the highway capacity with a shorter distance to vehicle $j-1$.

5.2.2.e Conclusions

This application explores the benefit of using V2V-equipped vehicle information from a vehicle ahead in the string when the preceding vehicle is a non-equipped one. The proposed system provides stable responses and, more interestingly, reduce intervehicle distances, providing tighter

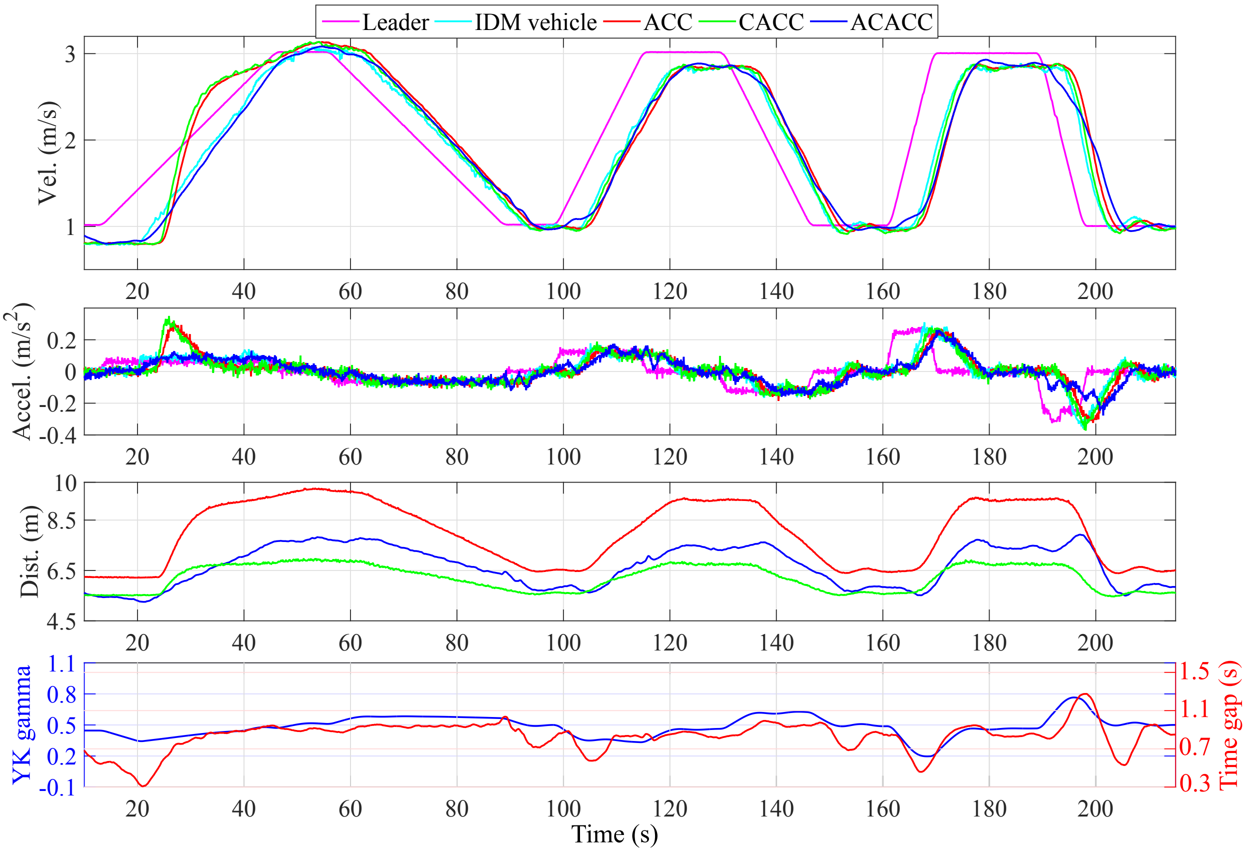


Figure 5.13: Experimental results comparison among car-following policies using CACC, ACC and ACACC controllers when a non V2V-equipped cycab is in front. The non V2V–equipped is a cycab with an IDM with the desired velocity of $33.33 m/s$.

strings. System performance is evaluated using either a human driven vehicle (emulated by the IDM) or an ACC-equipped vehicle as preceding one.

Stable transition between CACC controllers with different time gaps is ensured thanks to the use of the YK parameterization. The switching signal γ is correctly tuned in order to provide a tighter string of vehicles, improving highway capacity.

A comparative analysis using [Milanés and Shladover, 2014] models for ACC and CACC vehicles at high speeds and the IDM model as human driver vehicle is carried out. ACACC system exhibits a good performance, improving ACC degradation by providing a tighter string of vehicles, a faster response to speed changes and a better preservation of the string stability. These results are scaled to the low-velocity experimental platform. The performance is well-proved, improving relative distance to the preceding vehicle and obtaining faster responses.

5.2.3 Cut-in/cut-out transitions in CACC systems

Once communication availability problem has been solved by using the YK-based stable controller reconfiguration, this section proposes a different gamma design with fractional calculus based CACC controllers. These results validate YK adaptability to any type of controller, and provide stable cut-in/out maneuvers.

CACC systems have already shown their benefits when it comes to improve the traffic flow with

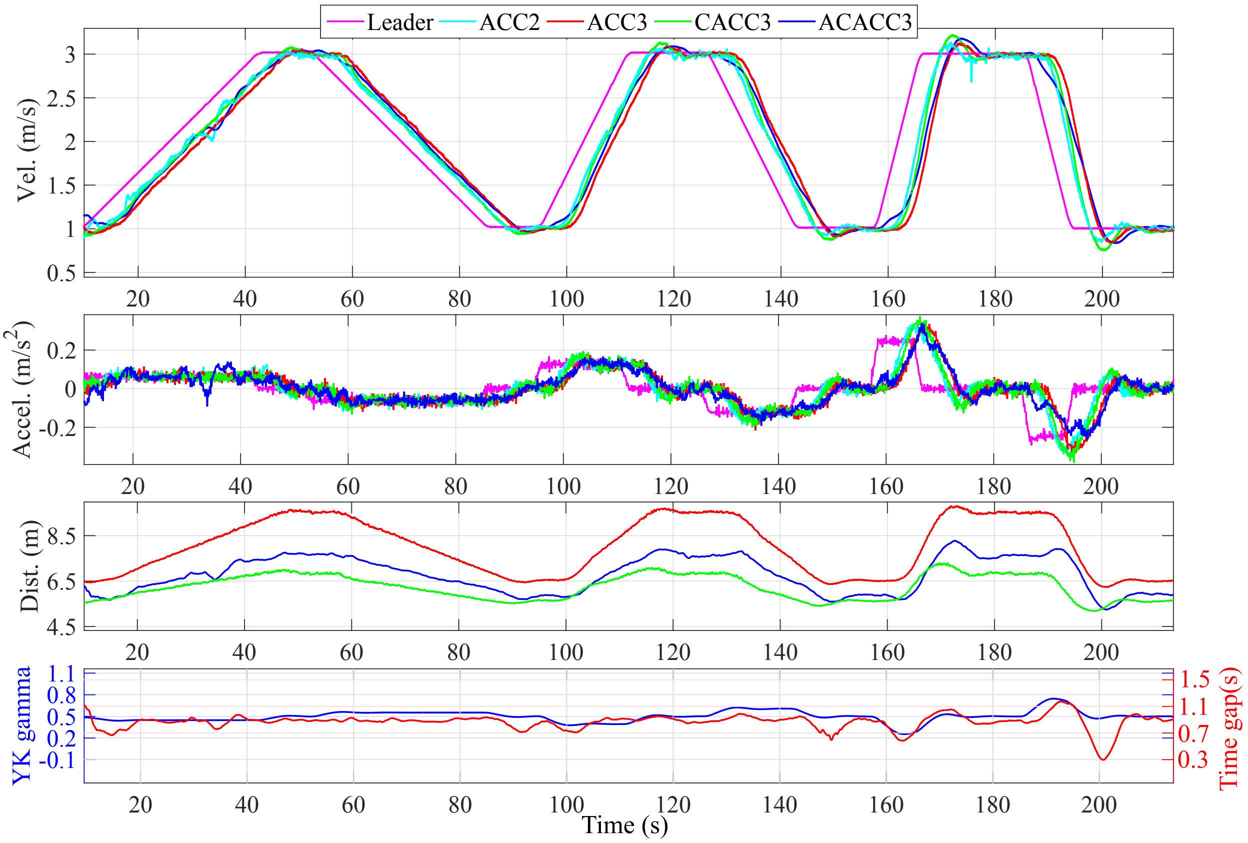


Figure 5.14: Experimental results comparison among car-following policies using CACC, ACC and ACACC controllers when a non V2V-equipped cycab is in front. The non V2V-equipped is an ACC-equipped cycab.

respect to the commercially available ACC system. However, their interaction with real traffic (i.e. their ability to manage vehicles entering and exiting the string) whereas keeping the stability has not been fully explored. This application presents the design of a CACC algorithm using fractional calculus for providing string stable responses to traffic perturbations. Then, the YK parameterization of all controllers is applied to assure stable transitions when other CACC-equipped vehicles are joining/leaving the string. The proposed control algorithm has been simulated for highway velocities by using the vehicle model in section 5.1.2. A LiDAR-based detection system as the one in section 5.1.1 is used for both tracking the preceding vehicle in the string and detecting the cut-in/out intention of a vehicle. This perception system is integrated with communication, so the gap is opened only if the vehicle has sent a request and the LiDAR-based detection system confirms it. Communication, perception and control algorithms have been implemented on the low-speed INRIA experimental platforms, providing encouraging results.

The application is structured as follows: Section 5.2.3.a introduces the problem formulation. The YK-based control algorithm able to provide stable cut-in/out maneuvers is detailed in section 5.2.3.b. High speed simulation results are provided in section 5.2.3.c for control validation. In section 5.2.3.d, the whole system is implemented on INRIA's low-speed real platform, validating the integration of communication and perception into the designed control algorithm. Finally, some concluding remarks are given in section 5.2.3.e.

5.2.3.a Problem formulation

Let define a string of w vehicles driving in the same lane. Problem formulation focuses again on vehicle j; it will be the one equipped with a CACC system that handles cut-in/out maneuvers. These always occur in front of the ego-vehicle. A homogeneus string of vehicles is considered and all-vehicles are V2V-equipped (including cut-in/out vehicles).

Merging maneuvers are carried out depending on the output of the LiDAR-based algorithm and the communication protocol. Here, the algorithm described in section 5.1.1.a is used to provide cut-in/out intentions. Cut-in intention is detected when a vehicle stands in the adjacent lane for a specific time. Thus, when a vehicle sends its request to cut-in, and stands in the LiDAR field of view a specific time, the gap is opened. Cut-out intention is also given when the trajectory of the preceding vehicle diverges from the string trajectory.

In a cut-in scenario, initially ego-vehicle is following its preceding vehicle. Later, a vehicle sends its request to go inside of the string. Then, the intention of a vehicle in the adjacent lane is detected, so ego-vehicle slows down, making space for the cut-in vehicle. Once inside, it becomes the new preceding vehicle, reaching again the desired time gap. In [Milanés and Shladover, 2016], cut-in situations were solved by defining a variable time gap policy. It permits smooth transitions, but system stability is not guaranteed.

In a cut-out scenario, the preceding vehicle already inside of the string decides to leave, so the ego-vehicle has to cover the remaining gap. In the present work, this is considered as a preceding vehicle change. Cut-out intention could be used in the future to decouple the lateral control when the preceding vehicle has the intention to leave. In [Bu et al., 2010], time gap is changed depending on a trapezoidal trajectory of the relative speed with the intervehicle distance to avoid controller saturation and to ensure passengers' comfort. Again performance is good, but stability is not guaranteed in the transition. The goal is to guarantee stability when cut-in/out occurs by using the YK parameterization.

Next steps are followed for system design:

1. A new YK structure based on fractional calculcus is proposed with the aim of improving internal and string stability. The design is carried out with the minimum time gap accepted by the general public for CACC applications [Nowakowski et al., 2010a], so-called h_{min}. It is the CACC controller to use in a normal CACC operation.

2. Once the nominal CACC controller is defined, another CACC controller with the same structure, but with a longer time gap is designed, h_{max}. The value of h_{max} is chosen so there is space for another vehicle. This value differs greatly depending on whether it is a high or low speed application.

3. By defining these controllers, limits in the time gap variation are established. Every cut-in/out maneuvers could be seen as a combination of both controllers, or in other words, as a time gap between $[h_{min}, h_{max}]$. YK parameterization is used to obtain stable transitions between these two fractional CACC controllers. Thus, stability is proved for any time gap variation related to cut-in/out maneuvers. The regulation between the action of each controller is based on the detection of the entering/exiting vehicles and intervehicle distance.

5.2.3.b Control algorithm

Control algorithm focuses on the longitudinal vehicle dynamics. The control algorithm implemented in the ego-vehicle must perform the following functions:

- Regulate the spacing gap with respect to preceding vehicle while ensuring string stability.

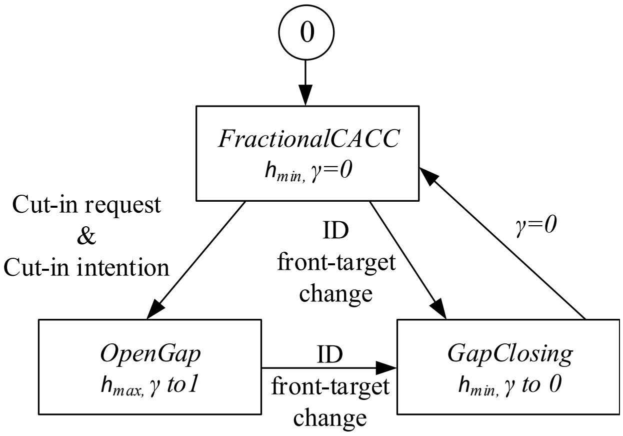


Figure 5.15: State machine for cut-in/out maneuvers depending on γ.

- Open the gap when another vehicle wants to cut in the string, while ensuring stability once the vehicle joins the formation.
- Reduce the inter-distance with respect to the remaining forward vehicle once the preceding is gone, ensuring also stability in the maneuver.

To manage the different needs, a state machine is designed (see Fig. 5.15). The state machine mainly consists in three states: *FractionalCACC*, *OpenGap* and *GapClosing*. The initial state is *FractionalCACC*. Ego-vehicle follows its preceding vehicle while keeping the distance corresponding to a minimum time gap h_{min}. *OpenGap* is activated when a vehicle sends a cut-in request and the perception algorithm confirms it. Thus, the gap is opened by increasing the time gap until a maximum value h_{max}. After, the cut-in vehicle goes inside, coming back to *FractionalCACC* passing through *GapClosing*. *GapClosing* is also activated when the preceding vehicle goes away, recovering the distance corresponding to h_{min} in a smooth and comfortable way. Notice that entering/exiting vehicles are seen as ID preceding vehicle changes.

The YK class of all stabilizing controllers is used with the two designed fractional CACC systems for ensuring stability when doing transitions between them. Doubly coprime factors for pairs $(K^j_{ext,0}, G^j/s)$ and $(K^j_{ext,1}, G^j/s)$ are obtained following guidelines in Theorem 3.7. By obtaining the parameter Q' (Theorem 3.13), stability in the transition is ensured, and then the time gap transition associated to cut-in/out maneuvers is also stable. Time gap modification is done through γ, as shown in Fig. 5.16. Note as unlike the previous application, two feedforward filters F^j_0 and F^j_1 with h_{min} and h_{max} are included. The effect of each of them is also regulated through γ.

General guidelines for fractional calculus design for h_{min} and h_{max}, and remaining states *OpenGap* and *GapClosing* are explained below. Details about how γ is changed to perform cut-in/out maneuvers are given.

Fractional CACC This section describes the initial mode *FractionalCACC*, which is the active state before and after cut-in/out maneuvers are carried out. Guidelines for the design of a feedforward Fractional Order PD (FOPD) CACC system are given with the aim of improving the internal and string stability of the system. The minimum time gap accepted for the general public for CACC applications $h_{min} = 0.6s$ is considered [Nowakowski et al., 2010a]. Please notice that the design of a FOPD CACC system with a time gap h_{max} is also given. Depending on whether it

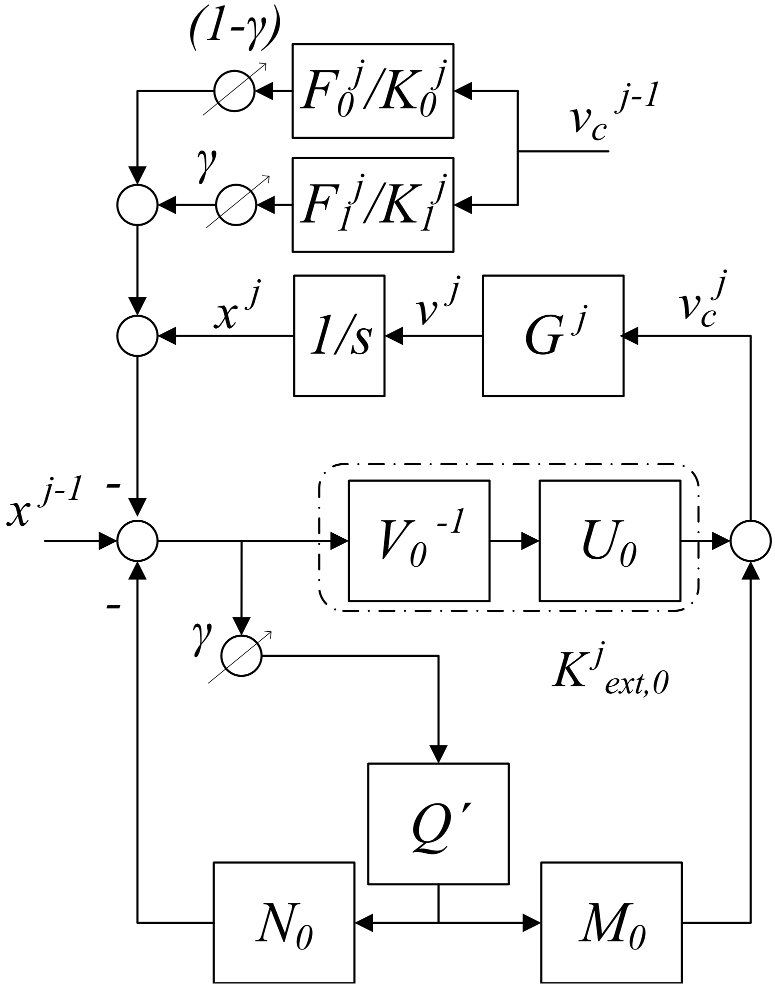


Figure 5.16: YK control structure for stable cut-in/out maneuvers.

is a high or low speed application, the value of h_{max} changes. Even if the latter is not part of the initial state, similar design guidelines are followed to obtain the needed controller to make room for a cut-in vehicle. Thus, both controllers are defined for applying the YK class of all stabilizing controllers to ensure stable time gap modification.

This work proposes a FOPD controller to keep the performance but increasing flexibility design by moving from integer to fractional order differentiation. Notice how the law control that regulates the distance error e^j changes (see Eq. 5.4). This type of controller [Li et al., 2010] is mathematically expressed in the Laplace domain as:

$$K_i^j(s) = \frac{U_c^j}{E_j} = K_{p,i}^j + K_{d,i}^j \cdot s^{\alpha_i^j} \tag{5.13}$$

Since there are three design parameters available for tuning, three control requirements are targeted seeking a more performing frequency-domain response. Given that lower time gaps require a more demanding control performance, for a time gap $h_{min} = 0.6s$ is proposed to employ a design procedure that not only aims to improve stability for the CACC transfer function loop–i.e. $1 + G^j(s)K_i^j(s)(1 + h_i^j s)$–, but also to enhance the string stability of the system. In other words, the controller corresponding to the initial state will maintain the desired phase margin and ensure the string stability even if communication delays affect the feedforward link:

- Phase margin guarantee of the system loop $\phi_m \geqslant \frac{\pi}{3}$.
- Set the desired loop bandwidth ω_{gc}.
- Increase the system string stability.

Table 5.7: FOPD CACC parameters. Cut-in/out.

$K^j_{ext,i}$	$k^j_{p,i}$	$k^j_{d,i}$	α^j_i	h^j_i
$K^j_{ext,0}$	$k^j_{p,0}$	$k^j_{d,0}$	α^j_0	$h^j_0 = h_{min} = 0.6s$
$K^j_{ext,1}$	$k^j_{p,1}$	$k^j_{d,1}$	α^j_1	$h^j_1 = h_{max}$

On the other hand, for cut-in maneuvers, a longer time gap is considered: h_{max}. The fractional order controller DC gain could be increased to compensate the slower CL dynamics that a longer time gap produces in the gap-regulation task. Even if the controller is faster, the phase margin criteria should be still fulfilled. The extended controller in Eq. 5.9 is used to include the time gap h^j_i into the controller. The resulting equation for the extended controller changes as there is a new design parameter α^j_i into the controller $K^j_i(s)$. Details about both controllers are summarized in Table 5.7.

Open gap Vehicles cutting in is a frequent maneuver when driving. Here, the gap between two vehicles is opened when cut-in intention of a vehicle in the adjacent lane is positive, and it has sent its cut-in request. Normally, when a vehicle performs a cut-in in front of a vehicle, both vehicles are driving at the same speed. Once ego-vehicle perceives that the other wants to make the maneuver, it slows down, making space for the cut-in vehicle. With the aim of ensuring stability in this transition, the gap is enlarged to the distance corresponding to h_{max} by setting γ to one.

The transition from the normal CACC operation (state *FractionalCACC*) to *OpenGap* is done by modifying γ from 0 to 1. The change from 0 to 1 is chosen as a trade-off between smoothness and reactivity from driver's point of view:

$$\gamma = \gamma_{min} + \frac{1}{t_{transition}}(\gamma_{max} - \gamma_{min})t \tag{5.14}$$

where $\gamma_{min} = 0$, $\gamma_{max} = 1$ and $t_{transition} = 5s$.

The cut-in vehicle could enter in the string once the gap is totally opened or before. Independently, the entrance of the cut-in vehicle is represented by ID preceding vehicle change, passing to state *GapClosing*.

Gap Closing The aim of the state *GapClosing* is to reduce the distance error when the preceding vehicle changes. ID preceding vehicle change determines communication switch between different vehicles. By minimizing the error, controller saturation is avoided and passengers' comfort achieved. Again transitions are done through γ, so stability is ensured. Three situations are considered:

- The gap has been opened and the cut-in vehicle goes inside the string.

- Cut-out maneuver when the preceding vehicle goes away.
- Once the preceding vehicle is gone, there is no vehicle to follow.

In the first two cases, there is a mismatch between the desired distance corresponding to h_{min} and the actual one. The error is set to zero by obtaining γ_{change} at the moment of the ID change. γ_{change} depends on the intervehicle distance d^j, the ego-vehicle velocity v^j ans standstill distance d^j_{std}. To come back to the state $FractionalCACC$, the transition is done from γ_{change} to 0. Again the transition between both values is set as a trade-off with smoothness and reactivity:

$$\gamma = \gamma_{change} - \frac{1}{t_{transition}}(\gamma_{change} - \gamma_{min})t, \quad \gamma \in [0,1] \tag{5.15}$$

where $t_{transition} = 5s$ and $\gamma_{min} = 0$. Notice that the value of γ will be always between 0 and 1; if intervehicle distance is out of the range $[h_{min}, h_{max}]$, γ saturates.

Finally, the third situation is considered. If the preceding vehicle perfoms a cut-out and there is no other vehicle to follow; ego-vehicle becomes the leader of the string with a specific reference velocity.

5.2.3.c Simulation results

This section presents the YK cut-in/out performance at high speeds (i.e. highway scenario). For simulation purposes, the vehicle model introduced in section 5.1.2 is used.

This longitufinal model and $h_{min} = 0.6s$ are considered for the design of the controller corresponding to the initial state, $K^j_{ext,0}$. The guidelines given in the previous section are followed. Thus, the resulting values are $K^j_{p,0} = 0.45$, $K^j_{d,0} = 0.25$ and $\alpha^j_0 = 0.3847$.

On the other hand, since for cut-in maneuvers a larger gap is needed, a longer time gap is considered: $h_{max} = 1.1s$. This time gap is longer enough to make room for another vehicle under high-speed condition. As there is not much difference between h_{min} and h_{max}, controller variables remain: $K^j_{p,1} = 0.45$, $K^j_{d,1} = 0.25$ and $\alpha^j_0 = 0.3847$. Information related to both controllers is summarized in Table 5.8.

Table 5.8: Nissan Infinity FOPD CACC parameters. Cut-in/out.

$K^j_{ext,i}$	$k^j_{p,i}$	$k^j_{d,i}$	α^j_i	h^j_i	$d^j_{std,i}$
$K^j_{ext,0}$	$k^j_{p,0} = 0.45$	$k^j_{d,0} = 0.25$	$\alpha^j_0 = 0.3847$	$h^j_0 = h_{min} = 0.6s$	$d^j_{std,0} = 2m$
$K^j_{ext,1}$	$k^j_{p,1} = 0.45$	$k^j_{d,1} = 0.25$	$\alpha^j_1 = 0.3847$	$h^j_1 = h_{max} = 1.1s$	$d^j_{std,1} = 2m$

String stability String stability is studied in function of every value of γ. Fig. 5.17 represents the string stability surface in function of γ. Surface results from the string stable transfer function in Eq. 5.16. This equation is obtained from the control structure in Fig. 5.16. As controller reconfiguration depends on γ, stability when switching between controllers is demonstrated. In addition, as this amplitude is always below the unity, string stability is also proved. By proving stability when doing controller transition, stable time gap modification is also guaranteed under cut-in/out maneuvers.

$$\frac{X^j}{X^{j-1}}(s) = G^j K_0(Q')\frac{1 + (1-\gamma)F^j_0/K^j_0 s/G^j + \gamma F^j_1/K^j_1 s/G^j}{s + G^j K_0(Q')} \tag{5.16}$$

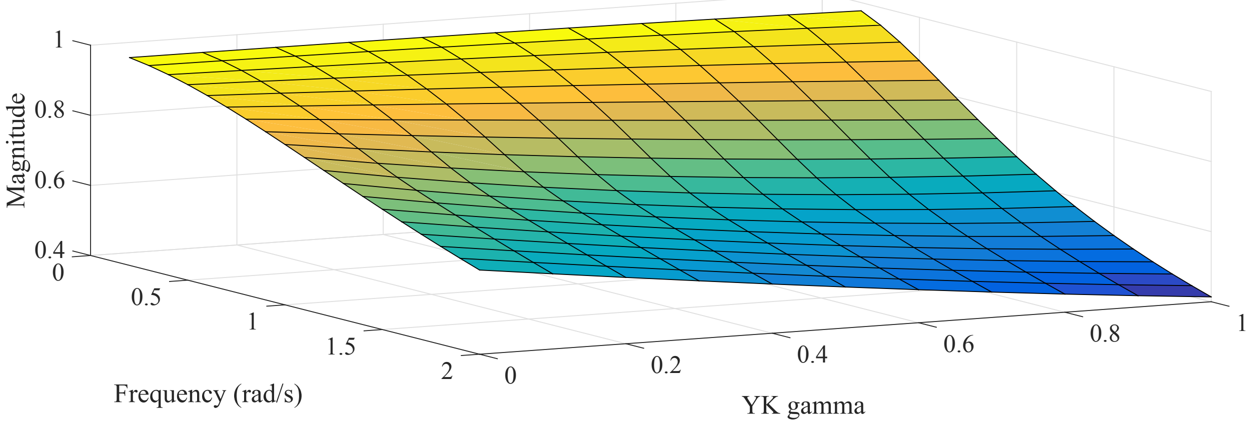


Figure 5.17: String stability's surface depending on γ. Nissan infinity M56.

Cut-in/out results The presented control algorithm has been implemented in MATLAB/Simulink for its validation. Perception system or communication are not integrated at this stage. Single flags are used instead for gap opening and ID preceding vehicle change.

Figure 5.18 depicts the performance of the control algorithm under cut-in situations. Starting from the same initial configuration, the cut-in situation occurs at different moments, ranging from a sudden cut-in up to fully waiting to the open gap. For notation, the top graph plots the vehicles' speeds. The second graph plots distance to preceding vehicle during the simulation. The third graph shows how γ is changed depending on the current state. And the bottom graph represents the time gap associated to the change in γ.

A total of three vehicles are considered; initially there is a two-vehicles CACC string: between old preceding vehicle (solid black line) and ego-vehicle. Later, new preceding vehicle (dotted yellow line) approaches in the adjacent lane. Both new and old preceding vehicles drive at $30m/s$. A flag is activated at $45s$ to simulate the cut-in intention and request; therefore the gap is enlarged by increasing γ to one until this new vehicle enters. Once the flag is received, the cut-in occurs at seconds 47 (solid magenta line), 53 (solid red line), 57 (solid blue line) and 65 (solid green line) for the ego-vehicle 1, 2, 3 and 4 respectively. The ID preceding vehicle change is equivalent to the different abrupt changes in the distance. Once it is inside, γ is modified again to come back to the normal CACC operation. Responses from 1 to 4 represent from the most impatient driver, who performs a cut-in without waiting for the gap opening, to the most conservative one, who even waits few seconds once the gap is fully opened.

Figure 5.19 depicts the performance of the control algorithm under cut-out situations. A CACC three-vehicles string is considered: Old preceding (solid black line) - new preceding (dotted yellow line) - Ego. As in the cut-in situation, different starting distances with the new preceding vehicle when the former one left are shown. The different responses at the ego-vehicle show how the control algorithm behaves when new preceding goes away, leaving different distances to old preceding vehicle. Again the ID preceding vehicle change is equivalent to an abrupt change in distance. Responses from 1 to 4 represent from the shorter to the larger remaining distances.

These results proves how cut-in/out maneuvers are carried out through the modification of γ while ensuring string stability by using the YK class of all stabilizing controllers. Different cases are tested: From impatient to conservative drivers in cut-in situations; and from shorter to larger distance to recover in cut-out situations. Inter-vehicle distances out of the range $[h_{min}, h_{max}]$ are also considered, seeing how gamma saturates, without effects in stability or driver comfort.

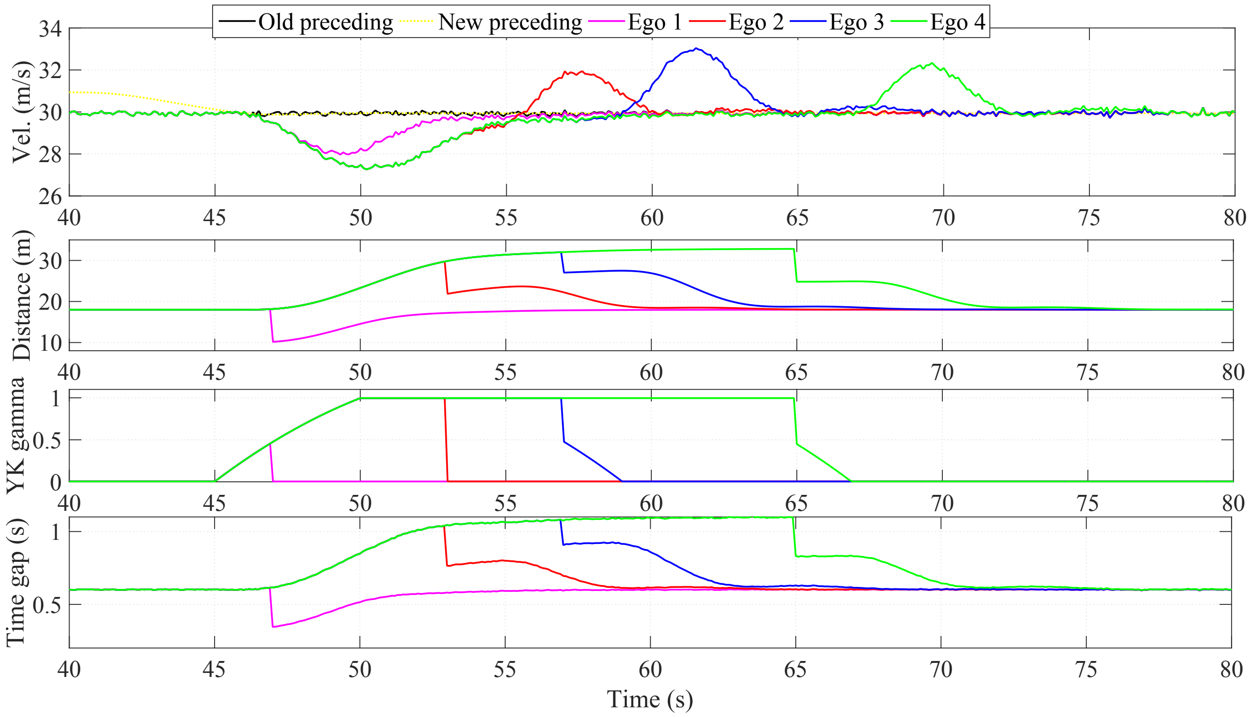


Figure 5.18: High speed simulation results cut-in maneuvers. Nissan Infiniti M56.

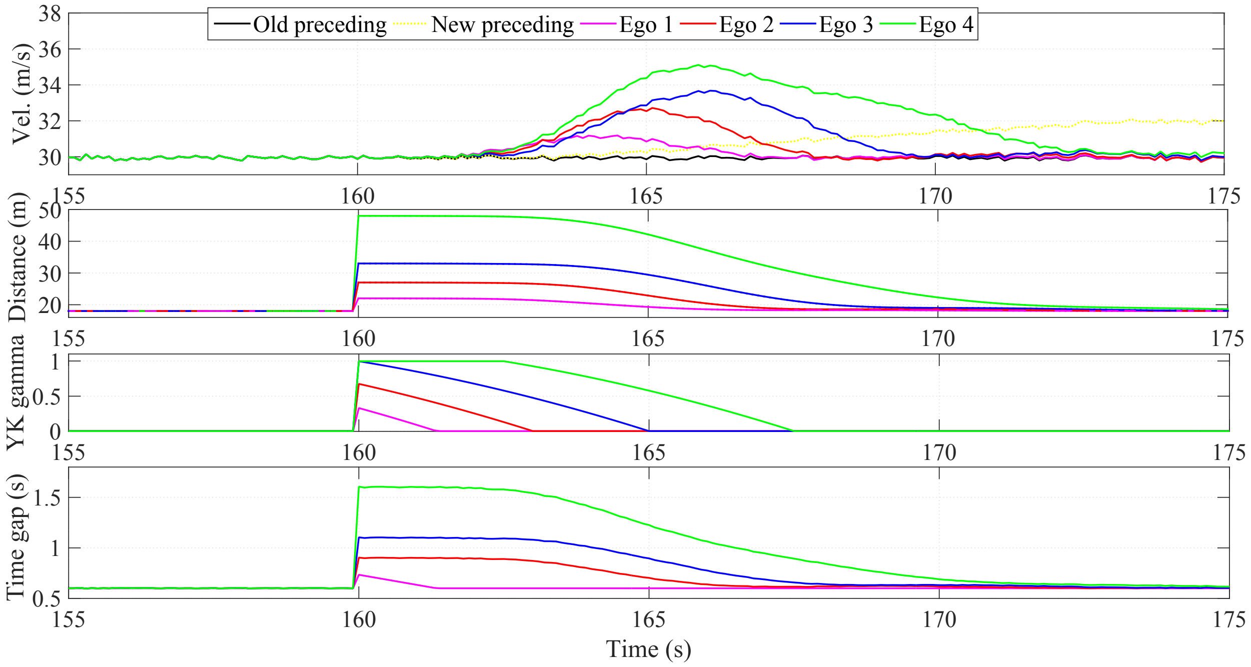


Figure 5.19: High speed simulation results cut-out maneuvers. Nissan Infiniti M56.

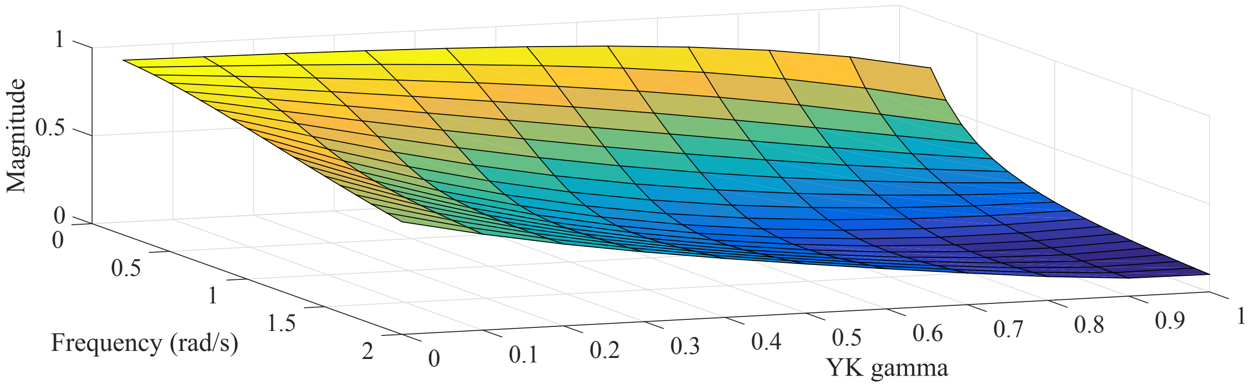


Figure 5.20: String stability's surface depending on γ. Cycab.

5.2.3.d Experimental results

This section presents the experimental validation on the low-speed platform described in section 5.1.1. Longitudinal model in Eq. 5.1 and $h_{min} = 0.6s$ are considered for the design of K^0_{ext}. Guidelines in section 5.2.3.b are followed. The resulting controller parameters are: $K^j_{p,0} = 0.505$, $K^j_{d,0} = 0.225$ and $\alpha^j_0 = 0.3847$.

Again, since the gap between two vehicles needs to be enlarged to let room for another vehicle, a longer time gap is considered: $h_{max} = 5s$. This time gap is longer enough to make room for vehicles than can drive up to $4m/s$. In this case, as the difference with h_{min} is larger, the DC gain of the controller is increased to compensate the slower dynamics that a longer time gap introduces: $K^j_{p,1} = 0.905$, $K^j_{d,1} = 0.225$ and $\alpha^j_1 = 0.3847$. Information related to both controllers is summarized in Table 5.9.

Table 5.9: Cycab FOPD CACC parameters. Cut-in/out.

$K^j_{ext,i}$	$k^j_{p,i}$	$k^j_{d,i}$	α^j_i	h^j_i	$d^j_{std,i}$
$K^j_{ext,0}$	$k^j_{p,0} = 0.505$	$k^j_{d,0} = 0.225$	$\alpha^j_0 = 0.3847$	$h^j_0 = h_{min} = 0.6s$	$d^j_{std,0} = 1.5m$
$K^j_{ext,1}$	$k^j_{p,1} = 0.905$	$k^j_{d,1} = 0.225$	$\alpha^j_1 = 0.3847$	$h^j_1 = h_{max} = 5s$	$d^j_{std,0} = 1.5m$

String stability Fig. 5.20 represents the string stability surface in function of γ. Equation 5.16 is used together with the model in Eq. 5.1 and controller parameters in Table 5.9. The magnitude of the Bode's diagram is always finite independently of γ, so stable controller reconfiguration is proved. Notice also that its value is always below the unity, so string stability is also ensured. As time gap modification is done through controller reconfiguration, stable time gap modification is guaranteed.

Cut-in/out results A complete scenario has been set to evaluate the transitions between the three states that composes the control algorithm.

Figure 5.21 depicts the performance of the control algorithm together with communication and perception algorithm. For notation, the top graph plots the vehicles' speeds. The second graph plots the distance to the vehicle in front during the test. The third graph shows how γ is

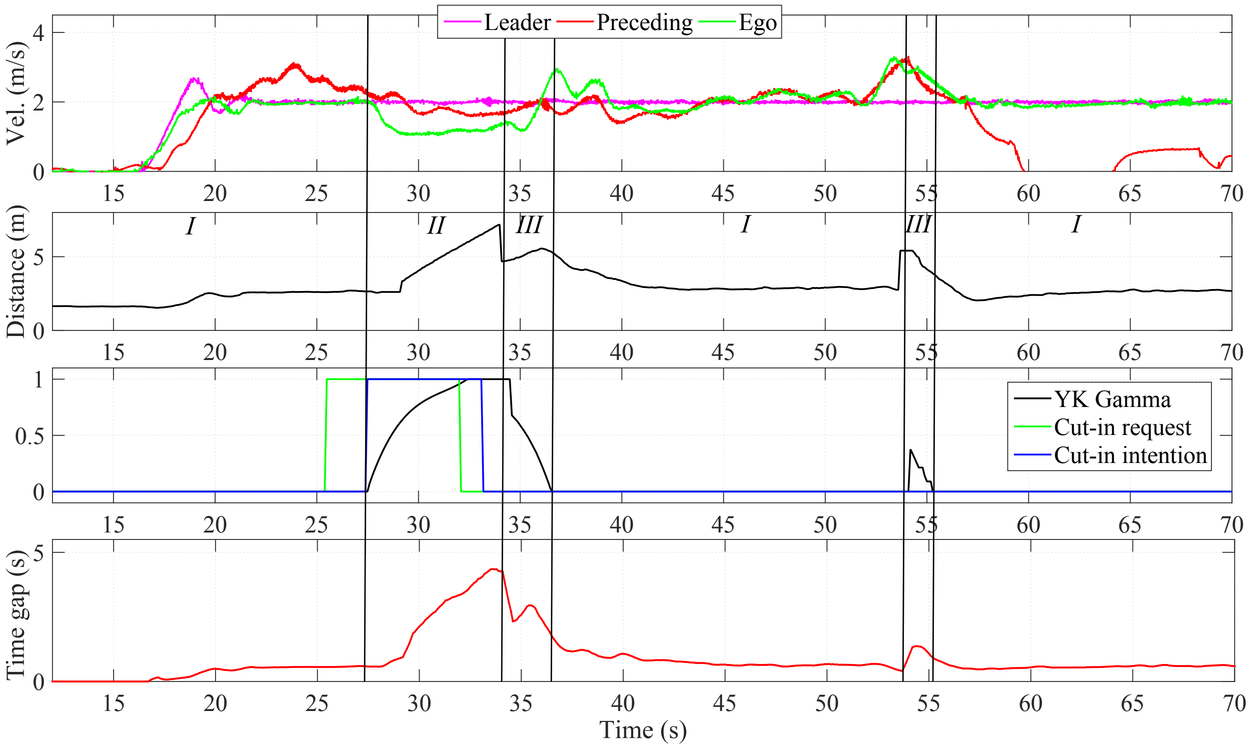


Figure 5.21: Low-speed experimental results cut-in/out maneuvers. State machine integration: $FractionalCACC(I)$, $OpenGap(II)$ and $GapClosing$ (III).

changed depending on cut-in intention and request. The bottom graph represents the real time gap associated to the change in γ.

A total of three cycabs is considered: 1) Leader (solid blue line) follows a velocity profile that changes from 0 to $2m/s$; 2) Preceding (solid red line) is manually driven, being the one entering/exiting the string; and 3) ego-vehicle (solid green line) with control and perception algorithms installed. Initially there is a CACC two-cycabs string: between leader and ego, performing the state $FractionalCACC$. Then, preceding sends its request to go inside the string at $25s$ (solid green line 3th graph), being detected by the perception system $2s$ later (blue solid line third graph). It is at this moment that the system considers that preceding wants to cut-in, passing to state $OpenGap$, opening the gap by setting γ to one (solid black line third graph). The gap remains open until preceding goes inside, fact detected by the perception system as a ID preceding vehicle change. The gap is not fully open when predecing goes inside. Once it is inside, γ is modified to zero in the state $GapClosing$, recovering the distance to come back to the initial state $FractionalCACC$. From $36s$ to $54s$ a normal CACC operation is performed, seeing how ego-vehicle is able to follow the preceding manual velocity. Finally, preceding goes away at $53s$, recovering the distance to leader by modiffying γ to zero. The normal CACC operation of the system is again recovered.

These results demonstrate the correct integration of communication, perception and control systems. The test shows good performance for both perception system and the designed state machine for the control algorithm. These results also validate the stable cut-in/out maneuvers carried out through γ in simulation results.

5.2.3.e Conclusions

This application presents the design of a control algorithm able to ensure stable cut-in/out maneuvers in a CACC string of vehicles. In a cut-in scenario, ego-vehicle slows down, making space for the vehicle willing to enter; while in a cut-out scenario, ego-vehicle accelerates to cover the remaining gap once the preceding vehicle is gone. Both merging/splitting maneuvers are seen as time gap variations in a CACC control structure. By defining fractional CACC systems with different time gaps h_{min} and h_{max}, and using the YK parameterization for doing stable controller reconfiguration between them, one ensures string stable responses to these traffic perturbations.

Control algorithm is first validated through simulation results. Vehicles driving at $30m/s$ are considered. It is proved how the system is able to leave space for another vehicle, perform cut-in maneuvers with impatient or conservative drivers entering the string, and recover different distances once the preceding vehicle is gone; all that while ensuring string stability by using the YK class of all stabilizing controllers.

Finally, a LiDAR-based perception system is used for both tracking the preceding vehicle and detecting cut-in/out intention of a vehicle in the adjacent lane. This cut-in intention is used together with a communication request for opening the gap in the previous control algorithm. Perception, communication and control are employed in the low-speed INRIA experimental platform, proving its correct implementation and validating simulation results.

5.3 Closed-loop identification

Once YK-based stable controller reconfiguration has been used with CACC applications, YK-based CL identification is reviewed. Applications in previous sections are developed for vehicles with same dynamics; but it is clear, that in a real traffic situation vehicle dynamics can change. Thus, YK-based CL identification is used in order to identify the dynamics of a vehicle connected to a CACC system.

In this section, the dual YK parameterization is used to perform CL identification of longitudinal dynamics of a cycab. The CL identification provided by the Hansen scheme (see section 4.5.2.a) is based in the OL identification of the dual YK parameter S_i. OL and CL identification algortihms are compared when employed in a CACC-equipped string of vehicles.

5.3.1 Online closed-loop identification for longitudinal vehicle dynamics

This application deals with the identification of longitudinal dynamics of a cycab for subsequent control performance's improvement. Low-speed vehicle dynamics is used as physical model which non-linear behaviour is complex to properly identify. Among several identification techniques, CL identification gives better performance in a model-based control design. Here, the Hansen scheme presented in section 4.5.2.a is used to transform a CL identification problem in an OL-like. The algorihtm is tested in a string of two cycabs equipped with a PD-based CACC system, showing how the resulting model is improved in comparison with an OL identification algorithm–Auto Regressive model with eXternal inputs (ARX) model.

The application is structured as follows: Section 5.3.1.a introduces the problem formulation. The control algorithm where Hansen scheme is applied is described in section 5.3.1.b. Section 5.3.1.c shows simulation results where a classical OL algorithm and Hansen scheme are compared. Results are validated on the INRIA low-speed experimental platform in section 5.3.1.d. Finally, some concluding remarks are given in section 5.3.1.e.

5.3.1.a Problem formulation

Vehicle dynamics identification is crucial for subsequent controller design. From the ego-vehicle point of view, a single LTI model in a given operation point of the system won't be sufficient for an application that is supposed to work in the whole velocity range. From the string point of view, a single LTI model won't be a representation of all existing vehicles in the market. In this section, INRIA's low-speed experimental platform characterization in section 5.1.1.b continues.

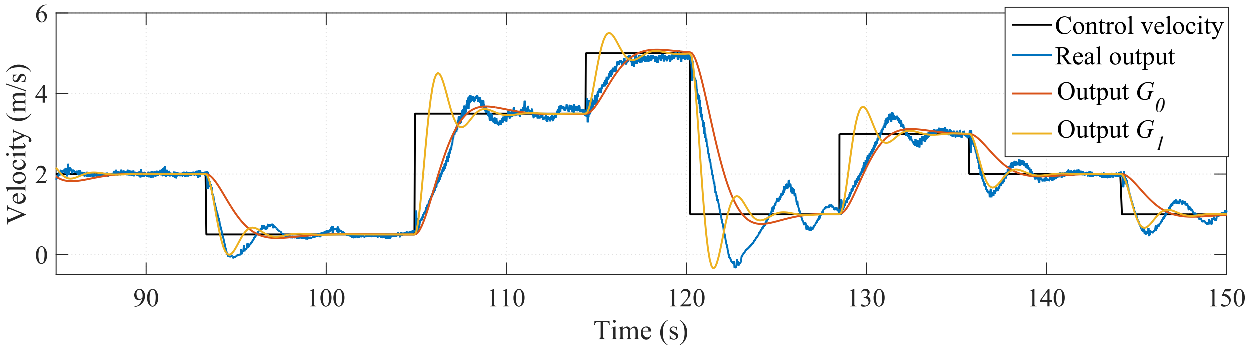


Figure 5.22: Nonlinear velocity cycab response compared with models.

When doing a characterization of the real platform described in section 5.1.1.b (see Fig. 5.22), a non-linear longitudinal behaviour is detected with different time responses and damping factors depending on the control velocity v_c^j (solid blue line). The LTI model identified in Eq. 5.1 (solid red line) is not able to cover the different dynamics in acceleration and braking phases. A non-linear model suitable for non-linear control is needed to satisfy good performance in the different operation points.

In this section, a string of two vehicles, $w = 2$, driving in the same lane is considered. The CL identification algorithm provided by the dual YK parameterization (so-called Hansen scheme) is implemented in the ego-vehicle for identification of longitudinal cycab dynamics. The same is compared with a well-known OL identification algorithm–ARX [Karaboyas and Kalouptsidis, 1991]. Then, advantages of dual YK CL identification can be highlighted at the same time that a more complete cycab model is obtained.

5.3.1.b Control algorithm

In this section, the control algorithm present in a cycab is explained. The vehicle is controlled in CL to compare OL and CL identification algorithms. The signals needed for both identification processes are clarified.

Here, a PD-based CACC as the one in section 5.2.1.a is employed: K_i^j in Eq. 5.4. First, tuning tools in MATLAB/Simulink and guidelines in [Naus et al., 2010] are used to obtain k_p^j and k_d^j. A final tuning is carried out during the experimental set-up in order to obtain smoothness and comfort. These controller gains are shown in Table 5.10. Notice how controller parameters change in comparison with those in Table 5.5. This is due to the use of K_i^j instead of $K_{ext,i}^j$. The transfer function in Eq. 5.1 serves as initial model G_0 for controller design. Doubly coprime factors for the pair (K, G_0) are obtained following guidelines in Theorem 4.1.

First, an OL model of the vehicle is obtained by using v_c^j and v^j directly in the ARX identification algorithm (see [Karaboyas and Kalouptsidis, 1991]). Its basis is equivalent to the following

Table 5.10: PD-based CACC parameters. CL identification.

K_i^j	$k_{p,i}^j$	$k_{d,i}^j$	h_i^j	$d_{std,i}^j$
K	$k_{p,0}^j = 0.5$	$k_{d,0}^j = 0.15$	$h_0^j = 1s$	$d_{std,0}^j = 5m$

equation:

$$y_{k+1} = Z\theta_k + v_k \tag{5.17}$$

where $y = v^j$ is the output vector, Z the matrix with old inputs and output values, θ_k the model parameters to identify and v_k white noise. θ_k is found by minimizing the least squared prediction error between real and estimated output.

Then, a CL identification is carried out applying Theorem 4.5. Note that $(d^j - d_{std}^j)$ and u_{ff}^j serve as external excitation signals r_1 and r_2 in the CL identification (see Figs. 4.2 and 5.6 for comparison). Instead of using v_c^j and v^j directly in the ARX model, filtered signals ζ_0 and z_0 are obtained in Eq. 4.34, and employed to estimate the dual YK parameter S_i. Once $\hat{S}_i$ is known, the CL vehicle model is updated by using Eq. 4.35.

In order to evaluate the performance between both algorithms, the metric called Vinnicombe ν-gap in [Vinnicombe, 2000] is used. ν-gap goes from 0 to 1 and expresses the difference between two models in reference to similarity in its CL operation. The closer its value to 0, the better the performance. It is defined for LTI system; its use into LPV systems cames with the definition of a set of LTI systems that composes the LPV model. The ν-gap metric is used during the simulation step with two different LTI models of a cycab. Experimental results regarding real, OL estimated and CL estimated outputs are provided to support the simulation results.

5.3.1.c Simulation results

As mentioned aboved, two different LTI models are considered for identification comparison. The initial LTI model G_0 already described. The PD-based CACC controller K is designed under consideration of this LTI model. A different LTI model G_1 is also considered, corresponding to the braking phase between $3m/s$ and $2m/s$ (see solid green line in Fig. 5.22). The model G_1 is used together with the controller K. Transfer function of G_1 is in Eq. 5.18. White noise with a signal-noise-ratio (SNR) of $42.42dB$ is added to the output. The modification from G_0 to G_1 emulates the case in which an erroneous model has been identified, being the real one G_1. As the real model is known, ν-gap metric comparison results reliable among OL and CL identified models.

$$G_1(s) = \frac{1}{0.1514s^2 + 0.2551s + 1} \tag{5.18}$$

An OL identification has been carried out for obtaining the model G_1. Signals v_c^j and v^j shown in Fig. 5.23 are used into an ARX model of third order, with one sample delay. The identification is performed using a sliding window approach of 400 samples with a sample time of $0.1s$. The resulting model is compared with G_1 by calculating the corresponding ν-gap.

The proposed identification system computes ζ_0 and z_0 by using Eq. 4.34 with signals $r_1 = d^j - d_{std}^j$, $r_2 = u_{ff}^j$, $u = v_c^j$ and $y = v^j$ in Fig. 5.23. Notice that these signals are now input and output of the same ARX model. Same order and delay are chosen, so advantages of CL identification can be studied under same conditions. Results of the OL identification of the dual YK parameter S_i is provided in Fig. 5.24. The resulting S_i is LTI as G_1 is LTI. The identified $\hat{S}_i$

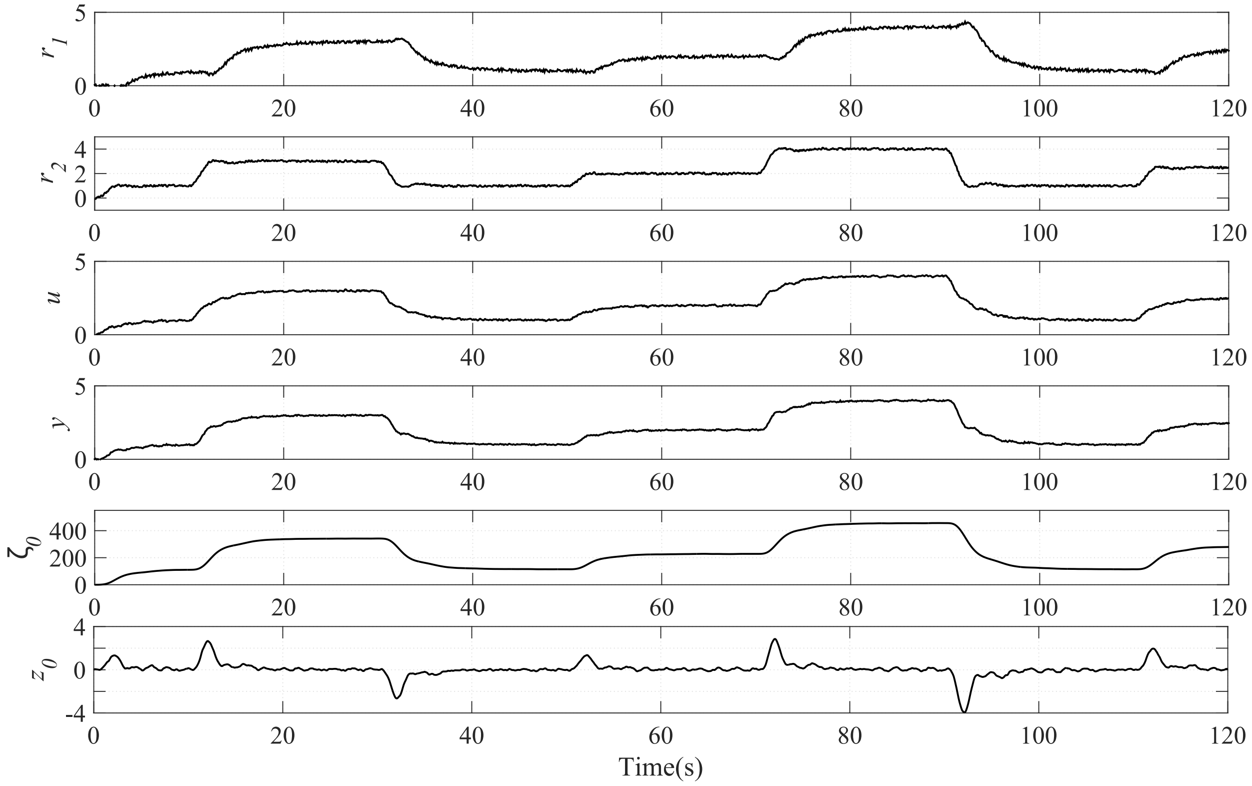


Figure 5.23: Signals for identification. Simulation results.

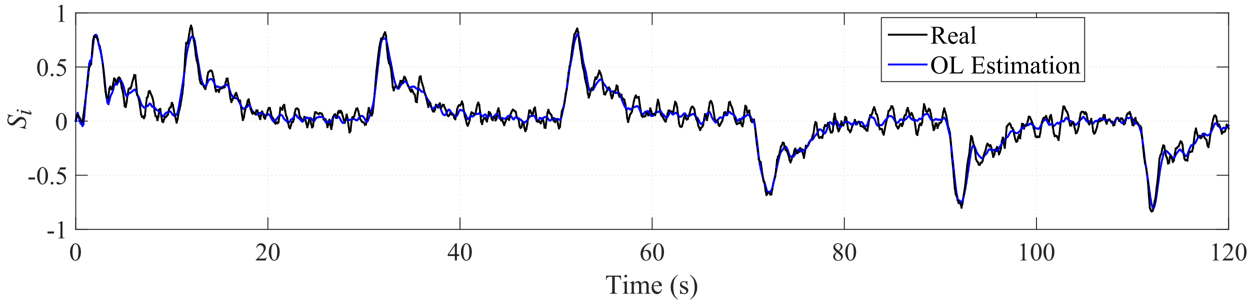


Figure 5.24: OL identification of dual Youla-Kucera parameter S_i. Simulation results.

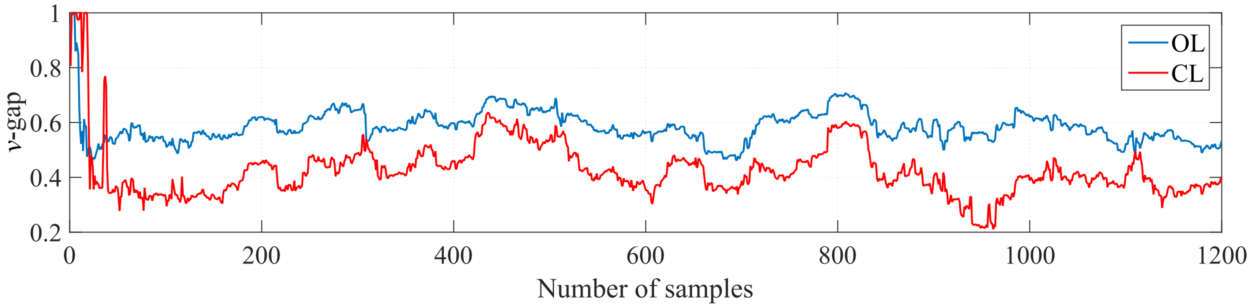


Figure 5.25: CL/OL comparison through ν-gap metric.

is used in Eq. 4.35, for obtaining the CL model of the vehicle. The identified model is compared with G_1 through the ν-gap metric.

ν-gap results from OL and CL identification are compared in Fig. 5.25. A better model is obtained when using ARX model together with the Hansen scheme. Under same conditions, ν-gap results closer to zero. CL nature of the data affects the ARX model, and the Hansen scheme helps to mitigate these effects; obtaining a model closer to the real one.

5.3.1.d Experimental results

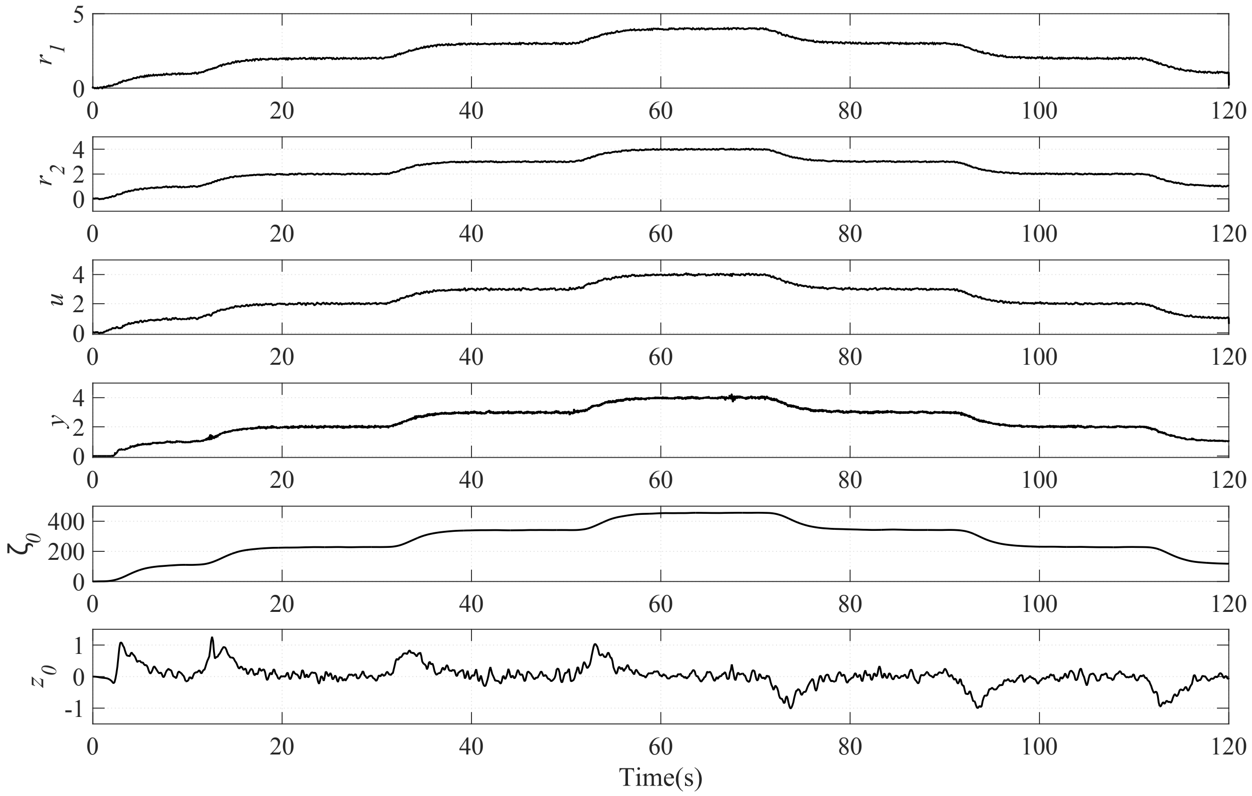


Figure 5.26: Signals for identification. Experimental results with soft speed profile.

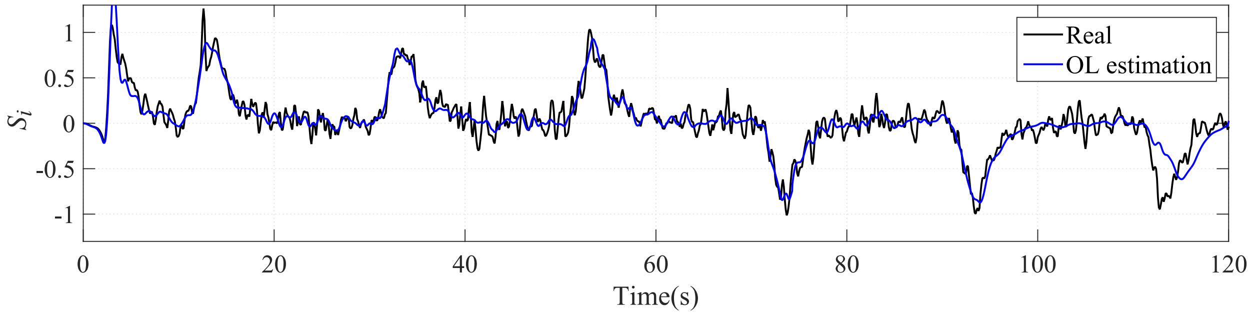


Figure 5.27: OL identification of dual Youla-Kucera parameter S_i. Experimental results with soft speed profile.

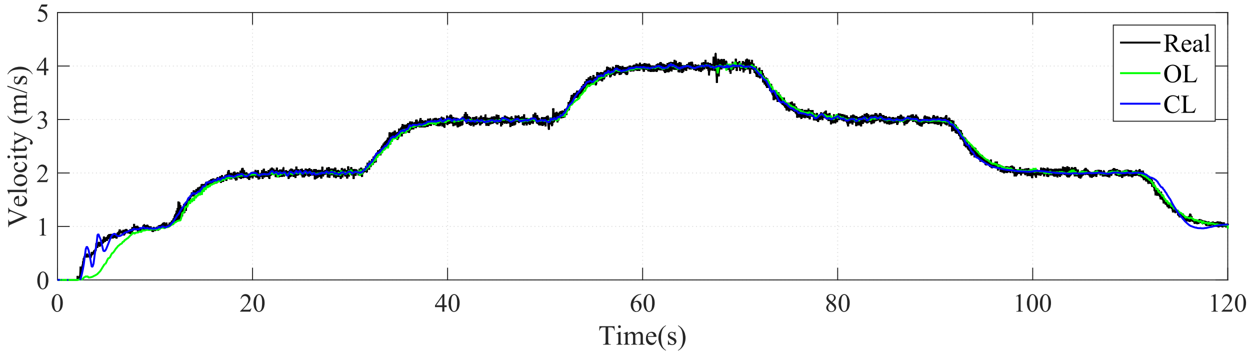


Figure 5.28: CL/OL comparison through estimated output. Experimental results with soft speed profile.

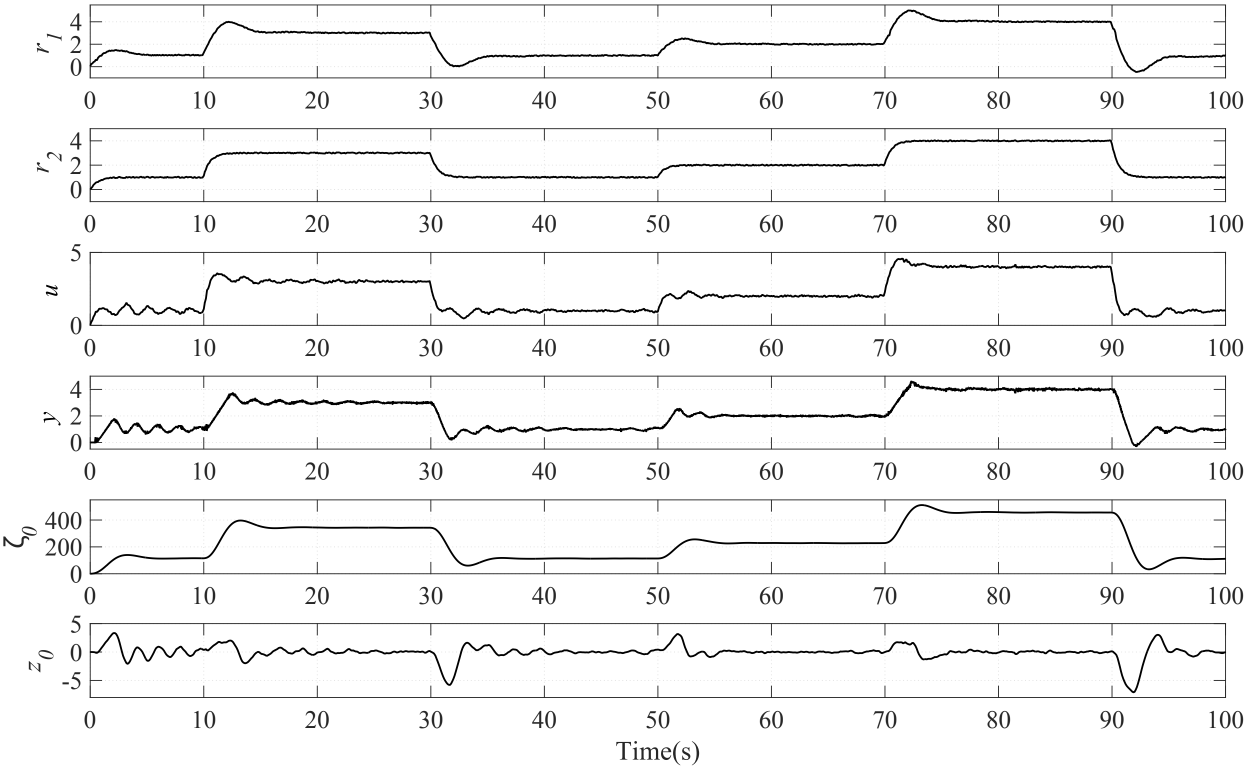


Figure 5.29: Signals for identification. Experimental results with abrupt speed profile.

ARX model, Hansen scheme and CACC controller are implemented in a string of two cycabs. Different speeds profile are applied to the leader of the string, so the algorithm can be tested with slower and faster velocity changes.

The signals from Fig. 5.26 correspond to the slower velocity changes case. An OL identification through ARX model is carried out with v_c^j and v^j; while the same is also used with ζ_0 ans z_0, for the OL identification of S_i. Its good performance is proved through results in Fig. 5.27, where real and identified output of S_i are compared. Note how the resulting $\hat{S}_i$ is LPV, as a single LTI transfer function is not able to recreate cycab dynamics. $\hat{S}_i$ and coprime factors from the pair (K, G_0) are used in Eq. 4.35 to get the CL model of a cycab with a CACC controller. As a priori knowledge of the real model is missing, ν-gap metric is not employed. Instead, estimated output

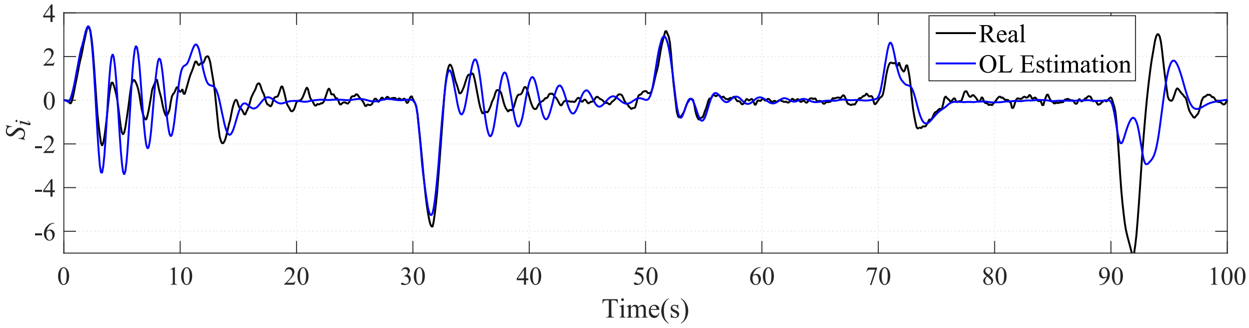


Figure 5.30: OL identification of dual Youla-Kucera parameter S_i. Experimental results with abrupt speed profile.

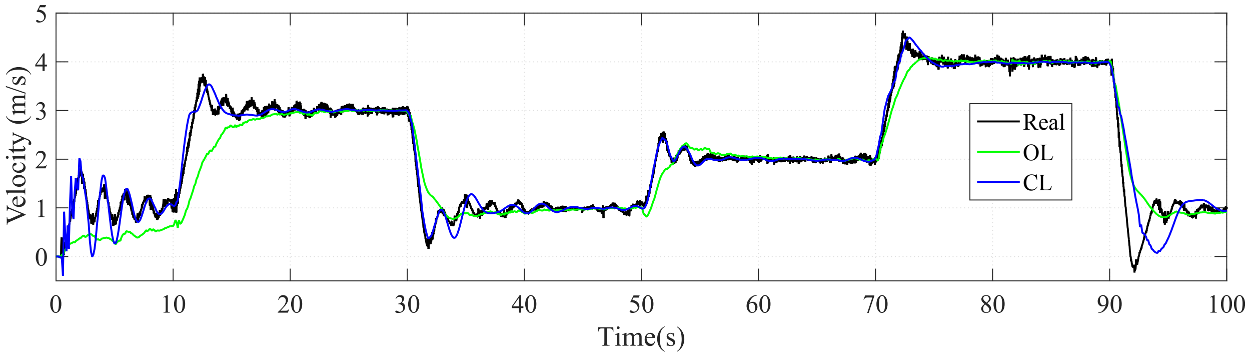


Figure 5.31: CL/OL comparison through estimated output. Experimental results with abrupt speed profile.

from the model obtained by Eq. 4.35, OL model and real output are compared in Fig. 5.28.

On the other hand, Fig. 5.29 shows the signals used in the identification for the faster velocity changes case. The procces is exactly the same than in previous cases. It is important to note that peformance of the OL identification for S_i in Fig. 5.30 is poorer than in the case of Fig. 5.27; this is due to the faster signal changes. Even with that, the CL identified model provided by the Hansen scheme is able to follow non-linearities, while the direct ARX model is not. Comparison of identified outputs is shown in Fig. 5.31. Differences between OL and CL algorithm are visible when faster changes are carried out; proving the good performance of the algorithm previously tested in simulation.

5.3.1.e Conclusions

In this application, Hansen scheme is used for CL identification of longitudinal dynamics of a cycab. The good performance of the algorithm is tested when connected to a PD-based CACC controller. The CL identification provided by the Hansen scheme is based in the OL identification of the dual YK parameter S_i. Thus, the same OL algorithm–ARX model– is used for performance comparison of CL and OL identifications. Simulation and experimental results are conducted to validate the proposed identification algorithm. Results show the feasiblity of the proposed scheme, improving ARX model identification. Dual YK property of identification is tested and a more complete longitudinal cycab model is obtained.

5.4 Adaptive control design

In this section, the dual YK parameterization is used in connection with adaptive control. A MMAC approach, as the one in section 4.4.1, is employed to deal with dynamics heterogeneity in CACC-equipped string of vehicles. The contribution of this application is clarified below.

5.4.1 Multi model adaptive control for cooperative adaptive cruise control applications

This application proposes a MMAC algorithm based on YK theory to deal with heterogeneity in CACC systems. The main idea of MMAC is to choose the plant in a predefined set that best approximates the system dynamics, applying the corresponding predesigned controller. A set of linear plants describing different vehicle dynamics is defined. Different CACC controllers are designed depending on these linear plants. Simulation and experimental results prove how MMAC determines the closest plant in the set, choosing the CACC system able to ensure string stability.

Next sections detail the system design and implementation. Section 5.4.1.a gives some background for the good understanding of the application. The YK-based MMAC approach is modified in section 5.4.1.b. for its application to CACC systems. Simulation and experimental results are analysed in sections 5.4.1.c and 5.4.1.d. Finally, some concluding remarks are given in section 5.4.1.e.

5.4.1.a Problem formulation

A string of w vehicles driving in the same lane is considered. j determines the order of a vehicle inside the string, $j \in [1, w]$. The solution here focuses on ego-vehicle (vehicle j) situation depending on preceding's (vehicle $j-1$) dynamics. From now on, G_{real} is equivalent to ego-vehicle G^j.

When vehicles within the string have identical dynamics (i.e. homogeneous string: $G^j = G^{j-1}$), disturbances attenuate/amplify uniformly along the string. String stability for homogeneous string of vehicles has been widely studied [Hedrick and Swaroop, 1994] [Swaroop and Rajagopal, 2001] [Seiler et al., 2004] [Khatir and Davison, 2004] [Barooah and Hespanha, 2005]. String stability is ensured for any positive time gap value if no delay is introduced in the communication link between vehicles. The statement is validated through experimental results: [Naus, 2010] carried out a string stable CACC with two citroen C4s; [Ploeg et al., 2011a] used a test fleet of six Toyota Prius III Executive, proving how a string-stable behaviour is achieved; a CACC system has been implemented on four production Infiniti M56s, guaranteeing also string stability (see [Milanés et al., 2014] for further details); three standard Scania tractor-trailer are also used in [Alam et al., 2015], forming a string stable CACC with three trucks.

On the contrary, when vehicles in the string have not identical dynamics (i.e. heterogeneous string: $G^j \neq G^{j-1}$), disturbances do not attenuate/amplify in the same way downstream; e.g. a vehicle with slower dynamics will have difficulties to follow a vehicle with faster dynamics. In the Grand GCDC competition in 2011, nine vehicles from different research institutes in Europe performed a CACC. The ATeam remarked how string stability is explicitly affected by preceding vehicle's dynamics–the minimum time gap achievable goes from $0.6s$ to $1.5s$ [Nieuwenhuijze et al., 2012]. Guidelines are given in [Shaw and Hedrick, 2007a] [Shaw and Hedrick, 2007b] about how to change variables of a CACC controller depending on whether slower or faster vehicle dynamics is in front of the ego-vehicle. Although the heterogeneous situation is more realistic, there is no real implementations in the literature.

String stability for heterogeneous string of vehicles has been addressed in two different ways: Robust control– a fixed controller able to work under bounded uncertainties in the plant. Dynamics difference between ego and preceding vehicle is represented as uncertainty in the platoon model; Adaptive control– controller changes with the unknown uncertainties in the system, e.g. control is modified depending on ego and preceding dynamics. Adaptive control is needed when robust control is not able to offer a decent behaviour due to all the parameters changes that can occur in the system.

Related to robust control, [Lestas and Vinnicombe, 2007] added some local conditions for heterogeneity to a robust controller designed for a homogeneous string of vehicles. Dynamics of the different agents within the strings need to be known to validate if the required local conditions are fulfilled. [Guo and Yue, 2011] proposed a controller design based on Linear Matrix Inequality (LMI) to ensure individual stability, while an H_∞ control algorithm considers disturbances that affects string stability. Again some conditions based on the components of the string should be validated. A H_∞ robust control design was presented in [Gao et al., 2015] to overcome uncertainty between model connections. The range of model parameters needs to be known apriori. Finally, [Gao et al., 2016] presented a H_∞ control method which can guarantee both individual and string stabilities to parameters uncertainties and communication delay. Heterogeneity is reflected by different vehicle mass and time lag of powertrain dynamics. Wind gust and road slopes are also considered as external disturbances.

Related to adaptive control, all the existing work takes as baseline controller a PD feedback/feedforward CACC (as the one in section 5.2.1.a). This type of controller has been extensively used due to its simplicity and performance. [Wang and Nijmeijer, 2015] designed a new feedforward controller for a heterogeneous string of vehicles to minimize spacing error and make independent string stability from preceding vehicles' dynamics. Models of ego and preceding vehicles are included in the feedforward, so string stability remains as in a homogeneous string of vehicles. However, there is no mention of how to obtain the dynamics of both vehicles in order to adapt the controller. More recently, a one vehicle look-ahead topology with model reference adaptive control is considered in [Harfouch et al., 2017]. They augmented a normal feedforward/feedback CACC system working in the homogeneous case with an adaptive term compensating unknown driveline vehicle dynamics. The idea is the same, but dynamics estimation for subsequent adaptation is included. Simulation results are given, where all estimates converge to true values in $31s$.

Adaptive control covers a greater dynamic range than robust control, but it is necessary to carry out online identification together with controller reconfiguration. With the idea of avoiding an identification process that could slow down the control loop, a set of linear plants that describes a wide range of vehicle dynamics can be defined.

Here the MMAC algorithm based on YK in section 4.4.1 is used together with the FOPD feedfback/feedforward CACC. A set of linear plants describing different vehicle dynamics is defined representing heterogeneous strings for MMAC validation. MMAC should estimate the closest plant in this set to ego and preceding vehicles, so the predesigned CACC controller able to ensure string stability is chosen. Encouraging results are obtained both in simulation and in real tests.

5.4.1.b Control algorithm

This section describes how MMAC can be used together with a CACC feedback/feedforward structure to address the problem of heterogeneity in CACC string of vehicles.

This work proposes a FOPD feedback/feedforward CACC as the one in section 5.2.3.b. This FOPD CACC controller is extended with the feedforward filter in [Wang and Nijmeijer, 2015] to

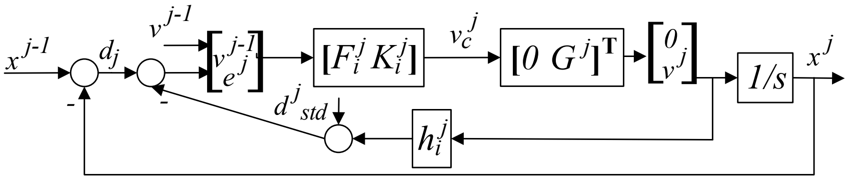


Figure 5.32: CACC MIMO structure.

ensure string stability even if different dynamics are present in the string. A model of each of the vehicles in the string would be necessary.

In order to avoid any identificacion process that could slow down the control loop, MMAC is used as a supervisor able to choose the proper controller among a pre-designed CACC controller set $\{K\}$. The task of the supervisor will be to estimate the closest plant to the ego and preceding vehicles, so the controller ensuring string stability will be chosen. Details how Algo. 1 is modified for a CACC application are given below.

CACC control structure When ideal communication is considered (no delays are present in the communication link), the choice of a feedforward filter as the one in Eq. 5.5 ensures string stability for any positive value of h_i^j [Öncü et al., 2011]. The statement above is no longer valid when a heterogeneous string of vehicles is considered. [Wang and Nijmeijer, 2015] included dynamics of ego and preceding vehicles in F_i^j (see Eq. 5.19), proving string stability even when dynamics are not the same.

$$F_i^j(s) = \frac{1}{h_i^j s + 1} \frac{G^{j-1}}{G^j} \tag{5.19}$$

Both ego G^j and preceding vehicle G^{j-1} need to be known to ensure string stability. When $G_j = G_{j-1}$, the feedforward controller reduces to the homogeneous case.

Figure 5.6 shows the classical SISO structure for CACC systems. Notice how the control signal v_c^j is the addition of u_c^j and u_{ff}^j. Here the controller structure is modified to MIMO, so FOPD controller K_j and feedforward filter F_j can be changed at once by using the YK parameterization. Results coming from this structure will serve as an adaptability proof of YK parameterization to MIMO systems. The modified MIMO structure is shown in Fig. 5.32.

Notice that there are as many controllers as there are possible dynamics combinations between ego and preceding vehicles.

MMAC modification for CACC Let's consider p number of nominal plants in the set $\{G\} = \{G_0..., G_i..., G_p\}$. Preceding G^{j-1} and ego G^j vehicles can be described by any of these nominal plants; or have a close behaviour to one of them. Depending on ego and preceding dynamics combination, a set of $(p+1)^2$ CACC controllers is created $\{K\} = \{K_{xr}\}$, where x is the closest plant in $\{G\}$ to the preceding vehicle G^{j-1}, and r the closest to the ego-vehicle G^j. Similarly, γ is the switching sequence $\{\gamma\} = \{\gamma_{xr}\}$, specifying which controller in $\{K\}$ is activated by modifying to 1 the corresponding γ_{xr} at the YK controller reconfiguration structure. Only one γ_{xr} can be set to 1 at the same time.

The switching sequence γ is specified by the supervisor. The goal is to determine which of the plants in the set $\{G\}$ is the closest to the real one G_{real}. As already outline, this is done through the signal z_i. The smallest its truncated 2-norm J_i the closer to the plant G_i. The same kind of system is installed in the preceding vehicle, sending to ego-vehicle which is the closest plant to G^{j-1}. Once x and r are known, the corresponding switching signal γ_{xr} is activated, so string stability is ensured.

The general MMAC algorithm in Algo. 1 is modified in Algo. 2 for CACC applications. It remains the same, except for a switching sequence γ that depends on two different indexes, a communication link added to get the value of x, and an extra condition when ego-vehicle is exactly between two plants in the set $\{G\}$. Even if unlikely, the fastest model is chosen in order to provide the most string stable controller in the most critical situation. The mandatory difference between two norms h is set to 0.4.

Algorithm 2 Multi Model Adaptive Control for CACC

```
1. Initialization
$\gamma[(p+1)^2] = 0$                                   ▷ Switching sequence initialization
$K[(p+1)^2] = [K_0, ..., K_i, ..., K_{(p+1)^2}]$          ▷ Candidate controllers
$\tilde{M}[p] = [\tilde{M}_0, ..., \tilde{M}_i, ..., \tilde{M}_p]$   ▷ Left coprime factor $M$ for $G_i$
$\tilde{N}[p] = [\tilde{N}_0, ..., \tilde{N}_i, ..., \tilde{N}_p]$   ▷ Left coprime factor $N$ for $G_i$
$\zeta[p] = [0]$                                        ▷ $S_i$ output initialization
$J[p] = [0]$                                            ▷ Truncated 2-norm initialization
loop
2. Controller reconfiguration
UpdateController($K[\gamma]$)                           ▷ Apply controller $K_i$, with $i = \gamma$
Get ($x$)                                               ▷ Index preceding vehicle through communication
Get($u$,$y$)                                            ▷ Obtain measurements $u$ and $y$
3. Supervisor
3.1 Identification
for $i$ in $(0,p)$ do
    $z[i] = \tilde{M}[i]y - \tilde{N}[i]u$              ▷ Output of $S_i$
    $J[i] = (norm2(\zeta[i]))^2$                        ▷ Compute truncated 2-norm/ Closeness to nominal models
end for
if $J[i] == J[\forall \quad except \quad i]$ then
    $i_{min} = fastest$                                 ▷ The fastest plant is considered when norms are identical
else
    $i_{min} = argmin_i\{i \in n \mid J[i]\}$           ▷ The smallest norm corresponds to the closer model
end if
3.2 Evaluate switching sequence
if $(J[\gamma] \leq J[i_{min}] + h)$ then
    $\gamma = \gamma$                                   ▷ Previous controller remains
else
    $\gamma = \gamma_{xi_{min}}$                        ▷ Controller changes
end if
end loop
```

5.4.1.c Simulation results

This section presents the MMAC CACC performance when the string of vehicles is heterogeneous. In the following, the sets of nominal plants and corresponding CACC controllers are introduced. These sets are chosen according to requirements of convergence and stability. Different simulations have been carried out. The first of them (Matching case), dynamics of both vehicles, preceding and ego, coincide exactly with one of the plants in the set $\{G\}$. The second and last (Non-matching plant), considers that preceding and ego-vehicle have dynamics close to one of the plants in the

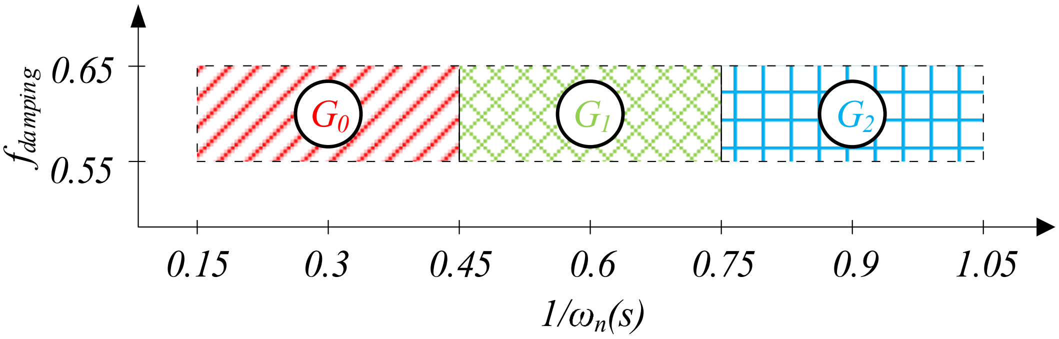


Figure 5.33: Uncertainty range string stable for pairs (G_0, K_0), (G_1, K_1) and (G_2, K_2).

set $\{G\}$.

Plant and controller sets Convergence and stability for MMAC algorithms is assured by assuming that the true plant is sufficiently close to the identified model in the set [Anderson et al., 2000] [Stefanovic et al., 2004]. Thus, determination of the correct set of nominal plants ensures convergence and stability [Anderson et al., 2001]. The idea is to determine stability criterion for each pair plant/model, analysing the maximum uncertainty around the plant without affecting the performance/stability of the system. In that way, one can define the required number of plants for the desired application range.

Vehicles with fast or slow dynamics, and with a certain overshoot are considered through second-order transfer functions: They are LTI plants mapping control velocity signals in output velocity signals:

$$G_i = \frac{w_{n,i}^2}{s^2 + 2f_{damping,i}w_{n,i}s + w_{n,i}^2} \tag{5.20}$$

where $f_{damping,i}$ is the damping factor and $w_{n,i}$ is the natural frequency in rad/s. Application range is defined by limits in $f_{damping,i}$ and $w_{n,i}$: $f_{damping,i} \in [0.55, 0.65]$ and $w_{n,i} \in [0.9524, 6.667]$. Notice that these limits could be extended in order to deal with every possible vehicle in the market.

Once the application range is defined, string stability is analysed in order to get the minimum number of nominal plants in $\{G\}$ able to cover the whole application range with string stability guarantee. String stability of each pair plant-controller is analysed with uncertainties in $w_{n,i}$ and $f_{damping,i}$. As a CACC system, uncertainties are in both preceding and ego vehicles, analysing the range where the homogeneous CACC controller is string stable. Three models in the set $\{G\}$ are enough to cover the application range. String stable areas for pairs (G_0, K_0), (G_1, K_1) and (G_2, K_2) are depicted in Fig. 5.33.

Values for nominal plants in the set are presented in Table 5.11. G_0, G_1 and G_2 go from faster to slower dynamics (see the Bode diagram in Fig. 5.34). Plants G_{x1} and G_{x2} are also present in the table, and they will be the non-matching cases in the corresponding subsection.

For every plant in the set $\{G\} = \{G_0, G_1, G_2\}$ a FOPD controller is designed with the objective of being robust to uncertainties. Guidelines in section 5.2.3.b are followed, resulting in a controller set $\{K\} = \{K_0, K_1, K_2\}$. Controller parameters are shown in Table 5.12. This set is extended including the feedforward filter; Eq. (5.19) depends on possible combinations between plants in $\{G\}$, yielding $\{K_{xr}\} = \{K_{00}, K_{01}, K_{02}, K_{10}, K_{11}, K_{12}, K_{20}, K_{21}, K_{22}\}$, where x is the plant in the preceding vehicle, and r the plant in the ego-vehicle.

Once nominal plants and candidate controllers are defined, Theorem 3.7 is applied to get

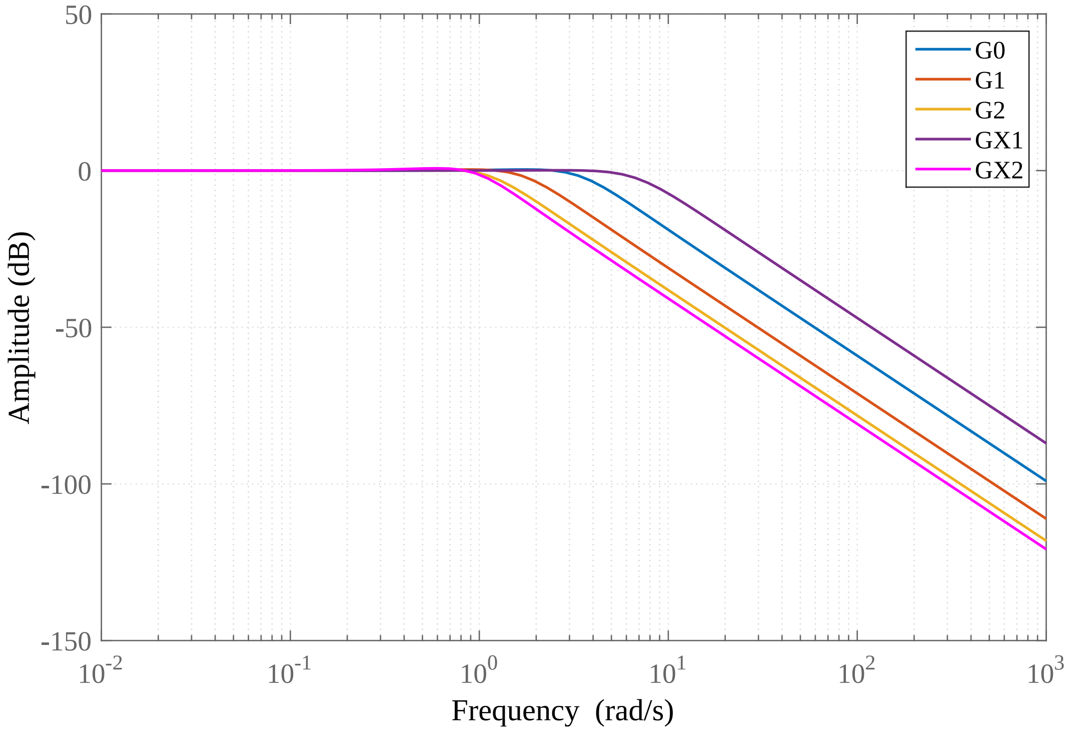


Figure 5.34: Bode candidate plants.

Table 5.11: Plants and parameters. MMAC.

G_i	$f_{damping,i}$	$w_{n,i}$
G_0	$f_{damping,0} = 0.6$	$w_{n,0} = 3.3333$
G_1	$f_{damping,1} = 0.6$	$w_{n,1} = 1.6667$
G_2	$f_{damping,2} = 0.6$	$w_{n,2} = 1.1111$
G_{x1}	$f_{damping,x1} = 0.65$	$w_{n,x1} = 6.6667$
G_{x2}	$f_{damping,x2} = 0.55$	$w_{n,x2} = 0.9524$

Table 5.12: FOPD controller parameters. MMAC.

	$K^j_{p,i}$	$K^j_{d,i}$	α^j_i	h^j_i
K_0	$K^j_{p,0} = 0.35$	$K^j_{d,0} = 0.15$	$\alpha^j_0 = 0.3847$	$h^j_0 = 1s$
K_1	$K^j_{p,1} = 0.5$	$K^j_{d,1} = 0.225$	$\alpha^j_1 = 0.3847$	$h^j_1 = 1s$
K_2	$K^j_{p,2} = 0.6$	$K^j_{d,2} = 0.3$	$\alpha^j_2 = 0.3847$	$h^j_2 = 1s$

the coprime factors needed in Theorem 3.13, so the set of YK parameters is obtained as $\{Q'_{xr}\} = \{Q'_{00}, Q'_{01}, Q'_{02}, Q'_{10}, Q'_{11}, Q'_{12}, Q'_{20}, Q'_{21}, Q'_{22}\}$. Each of them permits controller reconfiguration from an initial controller to a controller in the set $\{K_{xr}\}$. This set is reduced by choosing an initial controller K_{00}. Thus, the new set of YK parameters is $\{Q'_{xr}\} = \{Q_{01}, Q'_{02}, Q'_{10}, Q'_{11}, Q'_{12}, Q'_{20}, Q'_{21}, Q'_{22}\}$. Controller transition is done through the scalar factor γ_{xr} associated to Q'_{xr}. In the switching sequence γ only one γ_{xr} can be set to 1 at the same time. If any of them is activated, the inital controller K_{00} will be applied. Table 5.13 gathers the information related to the controller set and the corresponding YK parameter.

Table 5.13: Controller set and YK parameter. MMAC.

K_{xr}	G^{j-1}	G^j	K^i	F^j	Q'_{xr}	γ_{xr}
K_{00}	G_0	G_0	K_0	$\frac{1}{0.4s+1}$	–	–
K_{01}	G_0	G_1	K_1	$\frac{1}{0.4s+1}\frac{G_0}{G_1}$	Q'_{01}	γ'_{01}
K_{02}	G_0	G_2	K_2	$\frac{1}{0.4s+1}\frac{G_0}{G_2}$	Q'_{02}	γ'_{02}
K_{10}	G_1	G_0	K_0	$\frac{1}{0.4s+1}\frac{G_1}{G_0}$	Q'_{10}	γ_{10}
K_{11}	G_1	G_1	K_1	$\frac{1}{0.4s+1}$	Q'_{11}	γ_{11}
K_{12}	G_1	G_2	K_2	$\frac{1}{0.4s+1}\frac{G_1}{G_2}$	Q'_{12}	γ_{12}
K_{20}	G_2	G_0	K_0	$\frac{1}{0.4s+1}\frac{G_2}{G_0}$	Q'_{20}	γ_{20}
K_{21}	G_2	G_1	K_1	$\frac{1}{0.4s+1}\frac{G_2}{G_1}$	Q'_{21}	γ_{21}
K_{22}	G_2	G_2	K_2	$\frac{1}{0.4s+1}$	Q'_{22}	γ_{22}

Matching case The matching case considers that ego and preceding vehicles have dynamics that coincide exactly with one of the plants in the set $\{G\}$. Specifically, preceding vehicle has dynamics corresponding to plant $G^{j-1} = G_0$, while ego-vehicle has slower dynamics like $G^j = G_2$.

The dual YK parameterization is used in the supervisor, so identification algorithms are not needed to determine which is the closest plant in the set. z_i is used instead. By obtaining the signal value for every plant in the set, one can easily determine closeness. The top graph in Fig. 5.35 shows the evolution of these signals through time for the ego-vehicle G^j, while the bottom graph depicts the truncated 2-norm J_i related to z_i. Different system activation times are considered to see how the system reacts to different initial conditions. Solid-line indicates the case where MMAC is activated at $0s$, while dotted line shows the situation where MMAC is activated at $29s$. In both cases, z_2 and J_2 are always zero, as $G^j = G_2$. This plant index r is used together with the received index x to specify the switching sequence γ.

Figure 5.36 depicts the performance of MMAC CACC algorithm when preceding vehicle is $G^{j-1} = G_0$ and ego-vehicle $G^j = G_2$. A comparison is made between an ego-vehicle with an erroneous controller and with the controller that makes the system string stable, observing the transition from one to another when using MMAC. The initial and erroneus controller in the YK controller reconfiguration is K_{00}. The top graph plots vehicles' speeds when using K_{00} (dotted red line), K_{02} (dotted blue line), MMAC activated at $0s$ (solid green line) and MMAC activated at $29s$ (solid pink line). Preceding vehicle speed is also shown (black solid line). The second graph plots the switching sequence γ given by the supervisor. The switching sequence is determined by the signals in Fig. 5.35. Dotted line indicates the case where MMAC is activated at $29s$. Only γ_{02} is shown to have a lighter graph; the rest are zero even when MMAC is activated at $29s$. Finally, the bottom graph represents the distance error for each of the controllers.

From these results, one observes the importance of using different controllers depending on

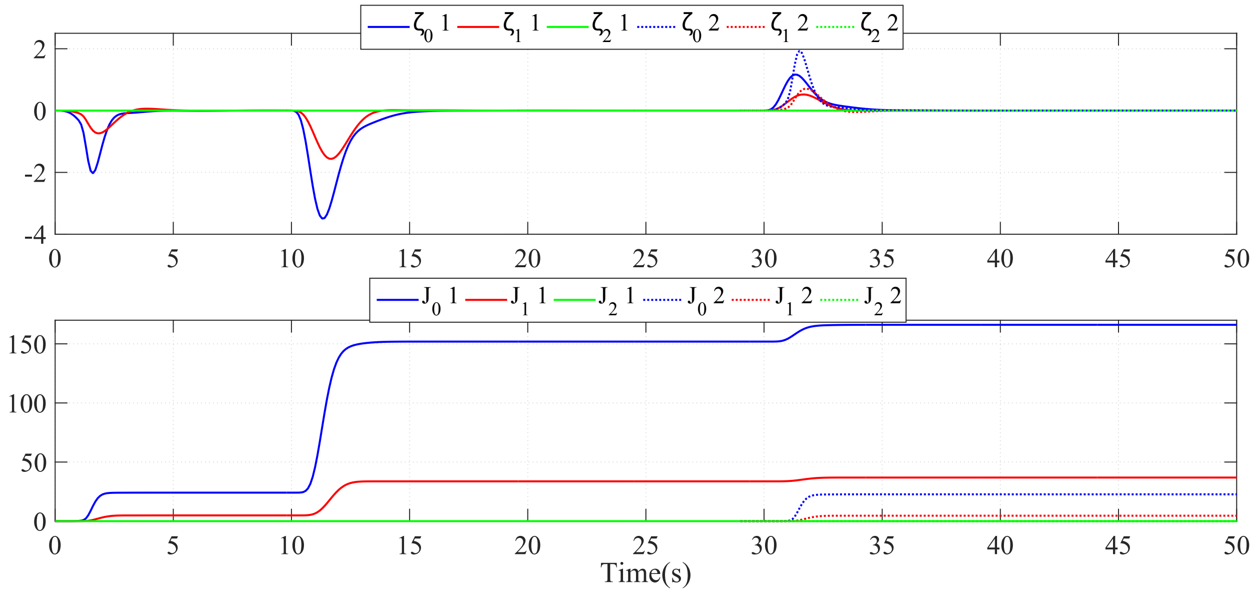


Figure 5.35: ζ_i and J_i comparison. $G_{j-1} = G_0$ and $G_j = G_2$. Solid-line indicates the case where the system is activated at $0s$, while dotted line shows the case where the system is activated at $29s$.

vehicle dynamics. When using K_{00} the string of vehicles results unstable. Ego-vehicle speed is amplified in comparison with the preceding one, and distance error tracking is large. No matter the activation time, MMAC is able to specify in a few seconds the correct switching sequence that makes the string stable. Distance error tends to zero and the transition between controllers is really soft. To the best of authors' knowledge, the only work mixing controller reconfiguration and dynamics estimation for heterogeneous string of vehicles is the one in [Harfouch et al., 2017]. Parameters estimation takes $31s$, which is much slower than the present work.

Not matching case The non-matching case considers that ego-vehicle and/or preceding vehicle are not one of the plants in the set $\{G\}$, but within the application range: G_{x1} and G_{x2}. The Vinnicombe ν-gap is employed to figure out which is the closest plant in the set, so later can be verified if the proposed MMAC algorithm is properly working. ν-gap goes from 0 to 1, and expresses the difference between two plants; the closer to zero the more the two plants look alike. In the case of G_{x1}, the lower value of ν-gap is for G_0 (ν-gap(G_{x1}, G_0)= 0.5073), so it is the closest plant to G_{x1} in the set. In the case of G_{x2}, the closest plant results G_2 with ν-gap(G_{x2}, G_2)= 0.1449.

Figure 5.37 shows the evolution through time of ζ_i and J_i when $G_{j-1} = G_1$ and ego-vehicle $G_j = G_{x1}$. Again different system activation times are considered, $0s$ and $20s$ in solid and dotted lines respectively. In both cases, one can observe how the minimum value corresponds to ζ_0. Even if G_{x1} is not a candidate plant in the set $\{G\}$, dual YK parameterization gives a good insight of the closest plant. Notice that ζ_0 is not zero in these cases, but it is much smaller than ζ_1 and ζ_2.

Figure 5.38 depicts the performance of MMAC CACC algorithm when preceding vehicle is $G^{j-1} = G_1$ and ego-vehicle $G^j = G_{x1}$. Again incorrect (K_{00}, dotted red line) and correct controller (K_{10}, dotted blue line) are compared with MMAC activated at $0s$ (solid green line) and MMAC at $20s$ (solid pink line). Indexes are identified in preceding and ego vehicles, so the corresponding γ_{10} is activated in few seconds no matter the activation time, making the distance error zero in a soft way.

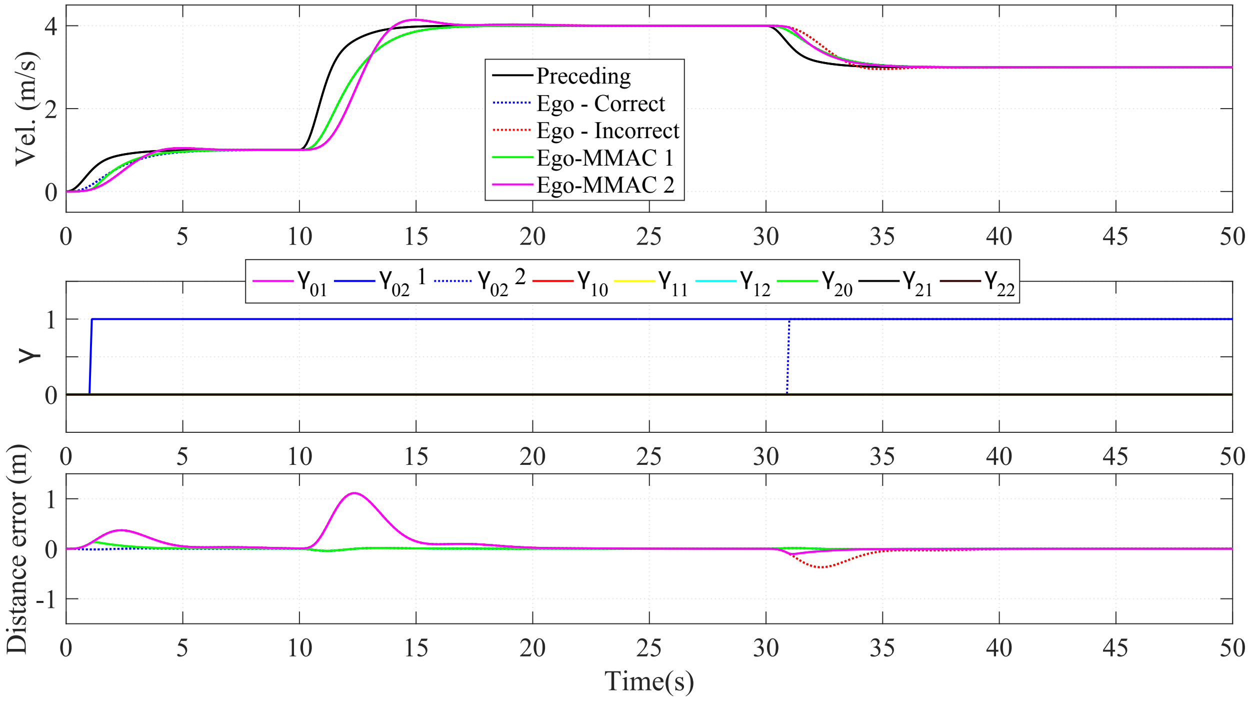


Figure 5.36: Simulation results MMAC CACC. $G_{j-1} = G_0$ and $G_j = G_2$.

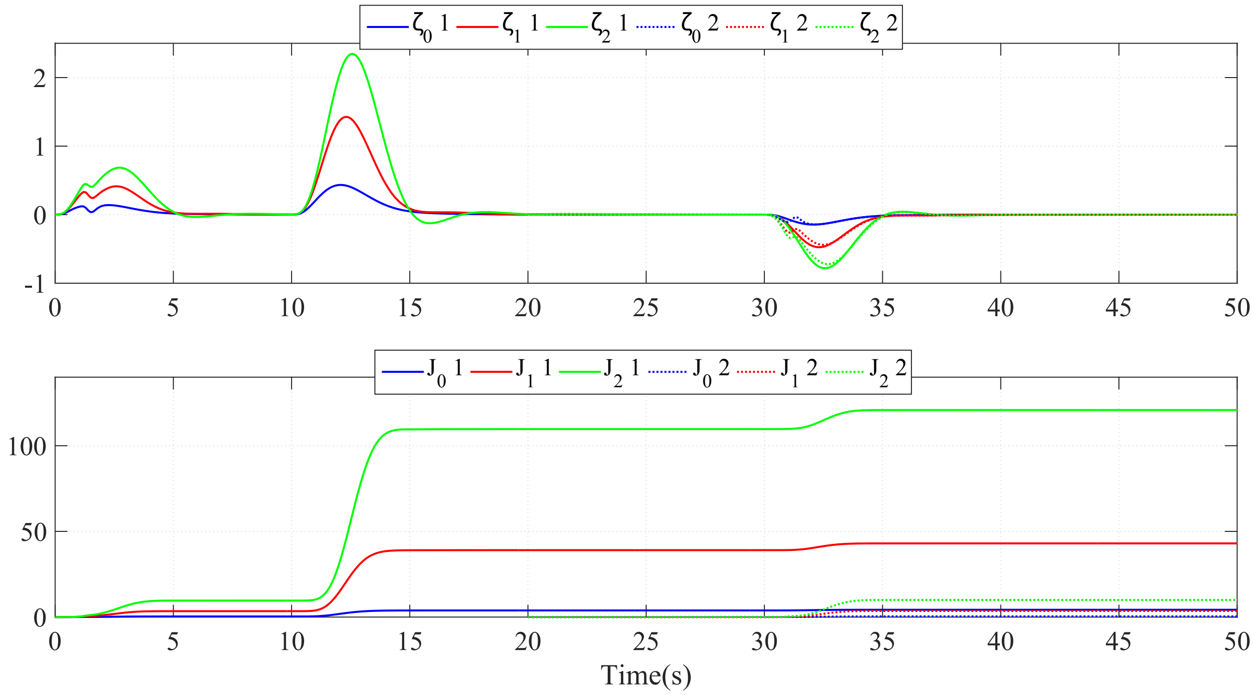


Figure 5.37: ζ_i and J_i comparison. $G_{j-1} = G_1$ and $G_j = G_{x1}$. Solid-line indicates the case where the system is activated at $0s$, while dotted line shows the case where the system is activated at $20s$.

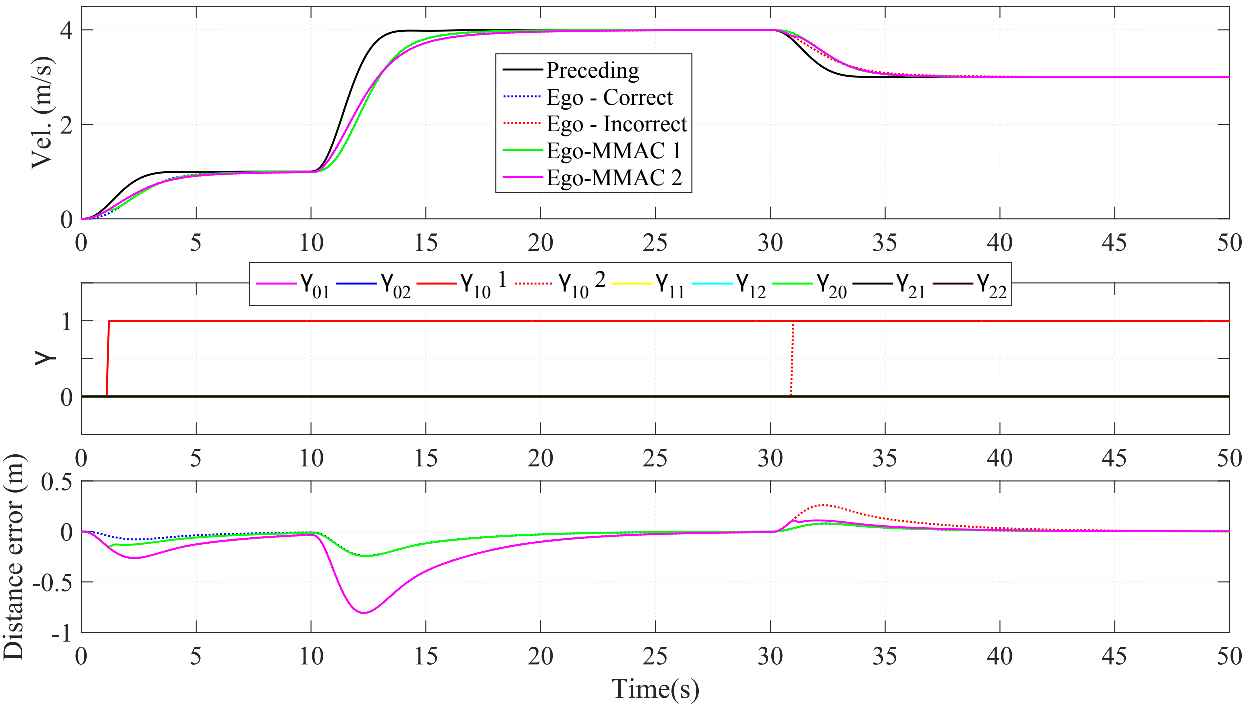


Figure 5.38: Simulation results MMAC CACC. $G_{j-1} = G_1$ and $G_j = G_{x1}$.

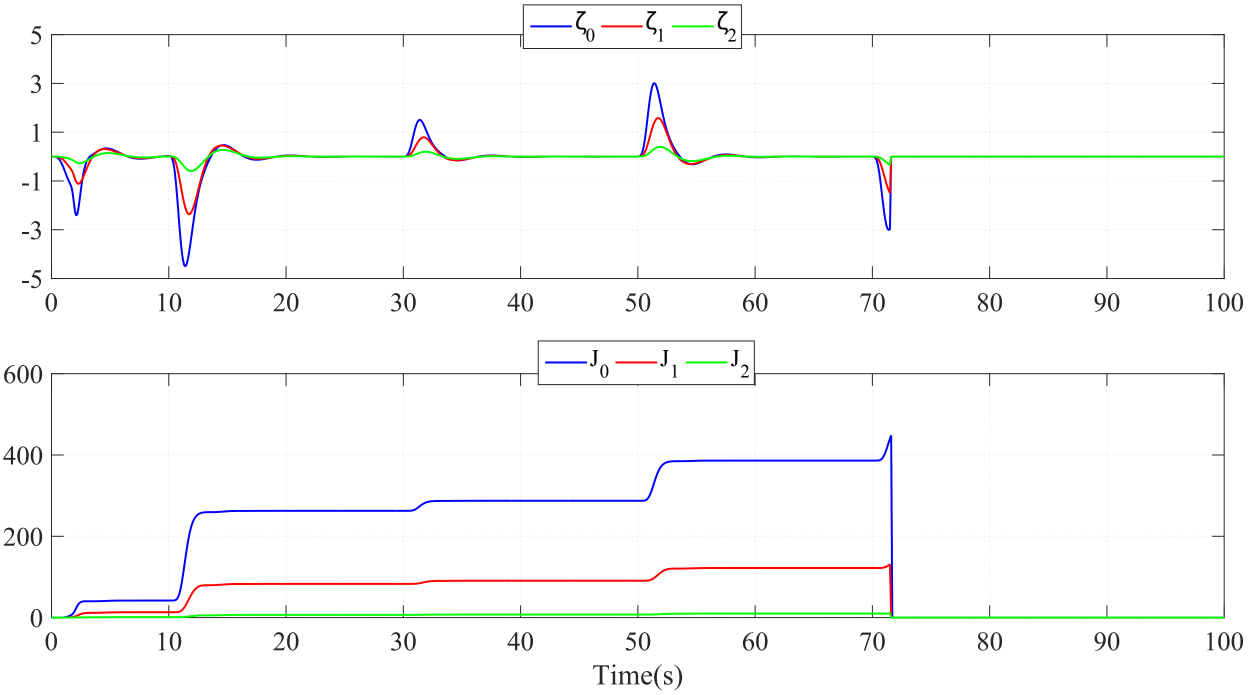


Figure 5.39: ζ_i and J_i comparison. $G_{j-1} = G_{x1}$ and $G_j = G_{x2}$. System shutdown at $71.5s$.

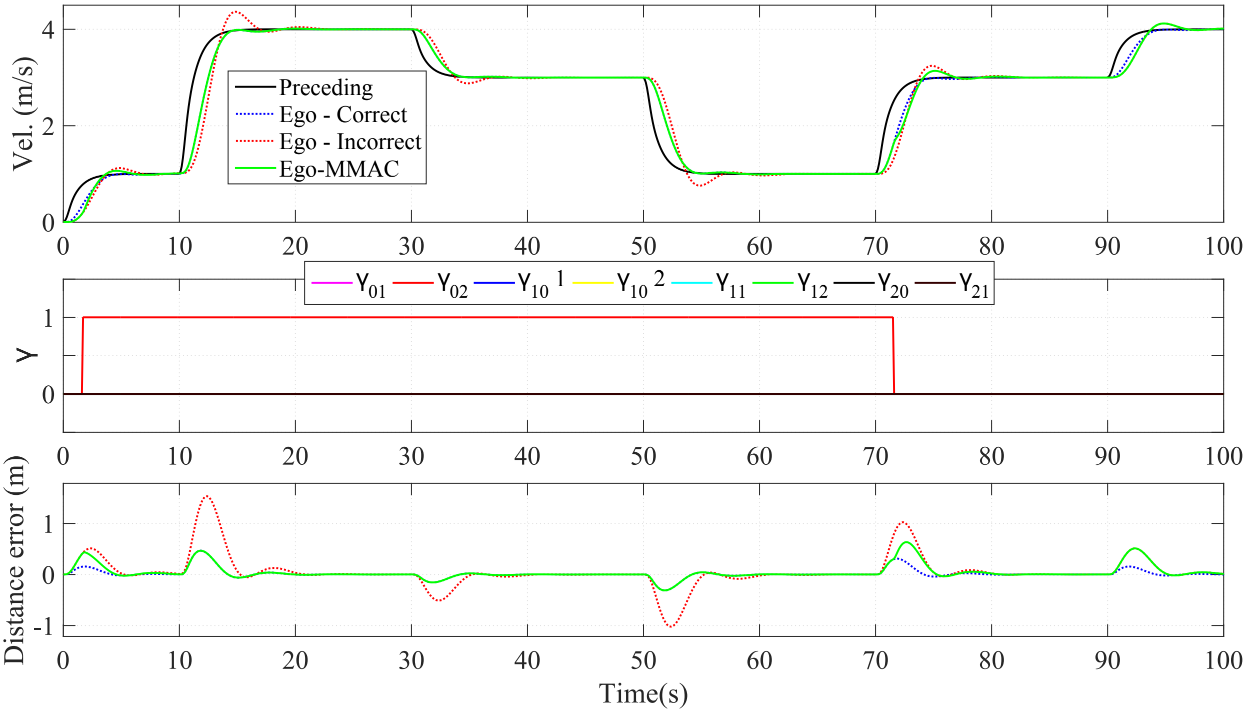


Figure 5.40: Simulation results MMAC CACC. $G_{j-1} = G_{x1}$ and $G_j = G_{x2}$.

Finally, a case with two non-matching models in preceding and ego vehicles is considered: $G_{j-1} = G_{x1}$ and ego-vehicle $G_j = G_{x2}$. This is the most critical case possible, as it considers the fastest preceding vehicle together with the slowest ego-vehicle, and different damping factors in the limits of the application range. Evolution through time of ζ_i and J_i for ego-vehicle is in Fig. 5.39. The closest plant is G_2 as ζ_2 is the lower one.

Figure 5.40 depicts the performance of MMAC CACC algorithm when preceding vehicle is $G_{j-1} = G_{x1}$ and ego-vehicle $G_j = G_{x2}$. Incorrect (K_{00}, dotted red line) and correct controllers (K_{02}, dotted blue line) are compared with MMAC solution (solid green line). Indexes are correctly identified in preceding and ego vehicles, so the corresponding γ_{02} is activated, preserving string stability. MMAC shut down is also analysed in the same figure; the system shuts off at $71.5s$ passing smoothly from a string stable behaviour to an unstable one. Notice how in the unstable case, the distance error tracking gets bigger than in previous cases. This distance error is directly associated with an amplified ego-vehicle speed.

It has been proved the correct behaviour of MMAC for CACC applications even if the real vehicle G_j is not exactly one of the candidate plants in the set $\{G\}$. Real vehicle dynamics should be within the application range. A larger set of models and controllers will be needed in practice to cover the different dynamics in a fleet of vehicles.

5.4.1.d Experimental results

A string of two cycabs is used as an experimental test platform. Experimental results serve as convergence validation of a real system. This is a non-matching case. Since both vehicles in the string have similar dynamics, the string will always be homogeneous, not being required the activation of any γ_{xr} in the switching sequence γ. The initial controller is modified to K_{01}; in that way, a modification of γ will be mandatory to obtain the best performance possible.

Figure 5.41 depicts the evolution through time of z_i and J_i during the experimental test. MMAC algorithm is activated at $0s$. The minimum value corresponds to z_0, so G_0 is the closest plant to the real dynamics of a cycab. This is again a non-matching case. It is not until $2.5s$ that the differences between J_0, J_1 and J_2 are big enough to determine the switching sequence γ.

Figure 5.42 depicts the performance of MMAC CACC algorithm when applied to a string of two cycabs. A comparison is made between the second cycab in the string with an erroneous controller and with the controller that makes the system string stable, showing the transition between both when using MMAC. The top graph plots cycabs' speeds when using K_{10} (dotted red line), K_{00} (dotted blue line) and MMAC activated at $0s$ (solid green line). First cycab's speed is also shown (black solid line). The middle graph shows the switching sequence γ determined by signals in Fig. 5.41. The bottom graph plots the corresponding distance errors when using correct, erroneous and MMAC controllers. Notice that initial distance errors are not the same for the three tests; a maximum error of $20cm$ was considered to start each of the tests.

From these results, one can see how important it is to take into account vehicle dynamics to ensure string stability. Inappropriate performance can be clearly appreciated on the distance error tracking. Even if this is less remarkable in the speed response, distance error is longer when the velocity step is big enough. The difference in the second step is less remarkable, as the speed step is smaller, but oscillations are present in the distance error. For the MMAC algorithm, it takes $2.5s$ to determine that both vehicles are closer to G_0, switching to the homogeneous controller K_{00} through γ_{00}. The transition between both controllers is smooth. The resulting transient is a combination of the initial distance error and the controller reconfiguration induced by γ_{00}. These results validate the simulation results of the previous section.

Experimental results serve as a convergence proof of a real vehicle with some of the plants in $\{G\}$. As vehicles are homogeneous, the most important task is to detect that both vehicles have similar dynamics, in order to switch to the classical homogeneous CACC controller.

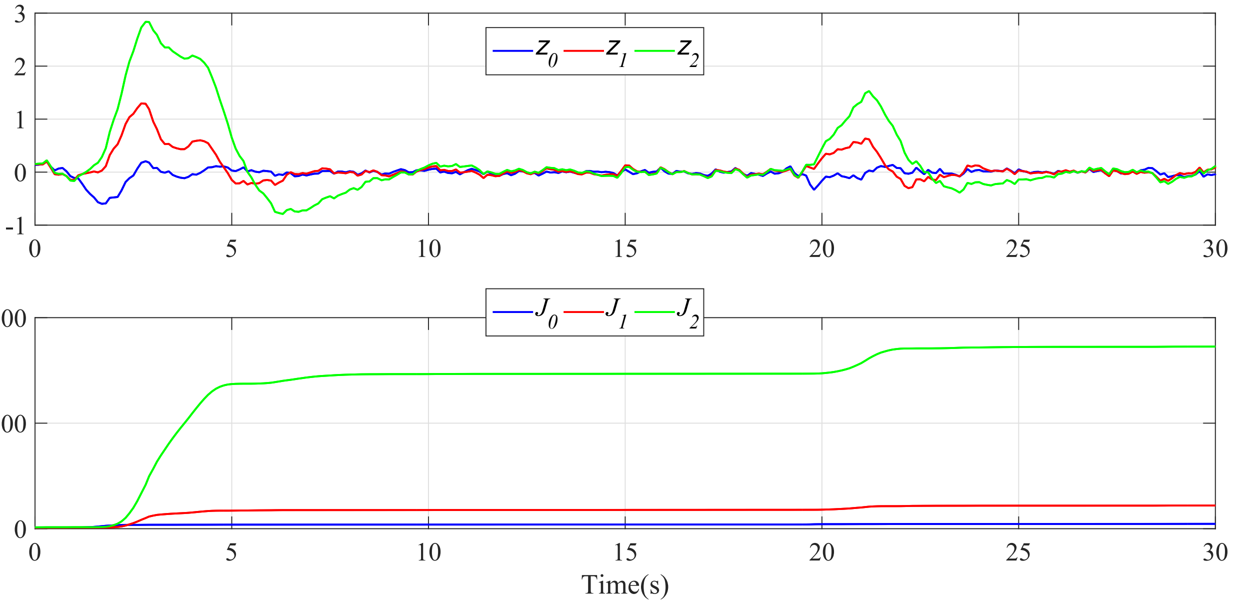


Figure 5.41: ζ_i and J_i comparison. Experimental results.

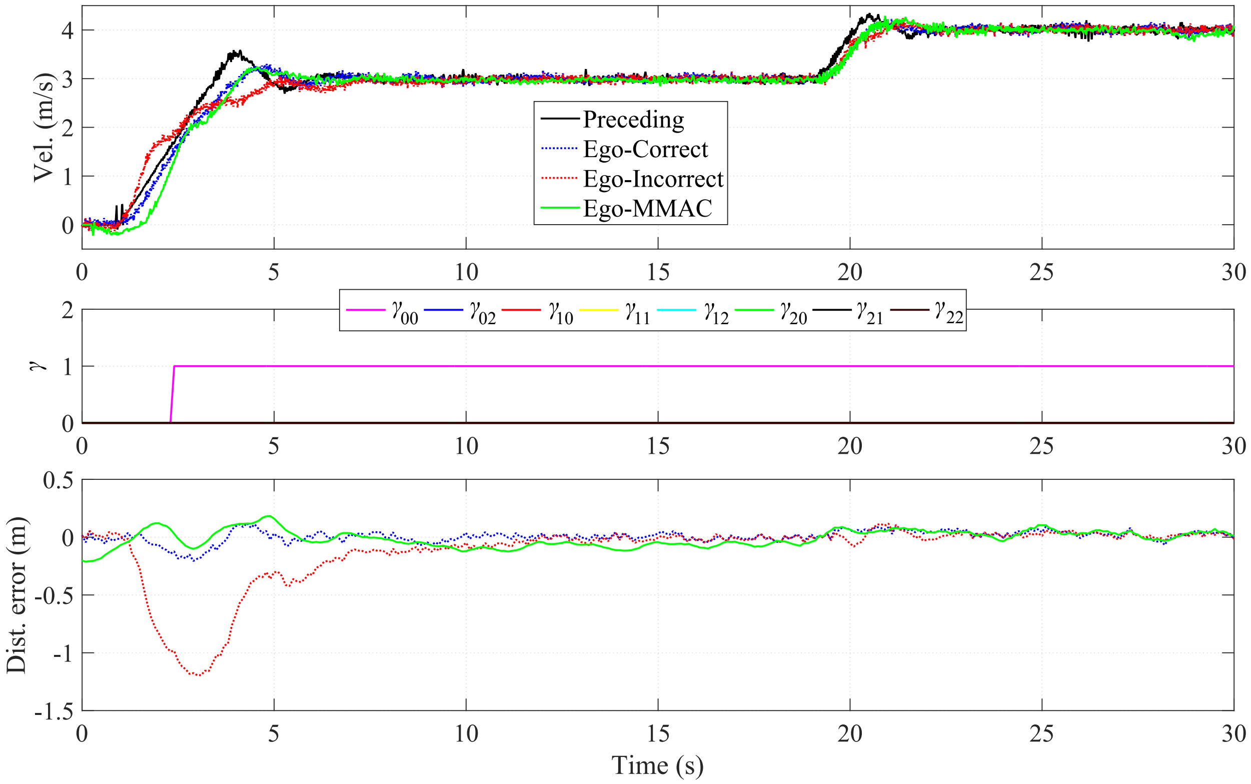


Figure 5.42: Experimental results MMAC CACC. String of two cycabs.

5.4.1.e Conclusions

This application explores the use of MMAC for CACC applications in heterogeneous/homogeneous string of vehicles. A YK-based MMAC algortihm is applied to a FOPD feedback/feedforward CACC system, so a structure able to handle different dynamics in a string without need of a identification algorithm is obtained. YK controller reconfiguration and dual YK properties are used within a MMAC approach. A MIMO FOPD is also used in order to prove YK adaptability to systems with more inputs.

Performance of MMAC CACC algorithm is analysed through simulation results. A set of three linear plants is considered for validation purposes. Dynamics matching and not maching cases with a plant in the set are considered, verifying how the supervisor is able to provide the closest plant in the set, activating the controller that ensures string stability. Dynamics estimation results much more faster than other estimation processes in the literature. A test setup of two cycabs is used, showing that MMAC CACC design is not only theoretically, but also practically feasible.

5.5 Conclusions

This chapter sumarizes different applications for both the YK and the dual YK parameterizations; different applications have been developed for CACC-equipped string of vehicles. Stable controller reconfiguration, CL identification and controller reconfiguration depending on dynamics estimation are addressed with the aim of improving the CACC state of the art. Both simulation and experimental results are provided for each of the applications.

First, developments based on YK-based stable controller reconfiguration are developed, dealing with the problem of non-available communication link with the preceding vehicle and traffic perturbations as vehicles joining/leaving the string.

CACC degrades to ACC when the communication link with the preceding vehicle is lost. This degradation occurs even if communication with another vehicle ahead exists. YK-based stable controller reconfiguration is employed to provide a hybrid behaviour between two PD-based feedback/feedforward CACC systems with different time gaps. It considers the V2V-equipped vehicle information from a vehicle ahead in the string. The correct tuning of the switching signal γ improves ACC degradation by providing tighter strings, faster responses to speed changes, and consequently traffic flow improvement. Simulation and real experiments show the proper behaviour of the designed control algorithm and encouraging results compared to existing ACC/CACC controllers. A maximum speed oscillation range of $5m/s$ is considered. In the future, it could be extended with a degradation to ACC when this value is exceeded; or with a normal CACC operation when communication with preceding vehicle is recovered. An algorithm for discriminating if the information received via V2V is coming either from vehicle preceding or another vehicle ahead should be included.

On the other hand, the ability to manage vehicles entering and exiting the string whereas keeping the stability has not been fully explored. The design of a FOPD feedback/feedforward CACC system is carried out for providing string stable responses to traffic perturbations. Then, the YK-based stable controller reconfiguration is applied to assure stable merging maneuvers as time gap variations. String stability for every value of γ is proved. High speed simulation results are provided for control validation. Finally, the whole system is implemented on INRIA low-speed real platform, validating the integration of communication and perception into the designed control algorithm. The ability of providing stable cut-in/out maneuvers is proved. In the future, this application could be integrated with the previous one in order to have a CACC system able to manage entering/exiting maneuvers of both V2V-equipped and non V2V-equipped vehicles.

Second, the dual YK parameterization is employed to improve the identification process of INRIA low-speed experimental platform. When performing and OL identification of the INRIA low-speed experimental platform, one notices a non-linear behaviour complex to represent through a single LTI transfer function. Hansen scheme is used to identify the longitudinal dynamics of the vehicle for future control performance's improvement. An OL identification of the dual YK parameter S_i allows to obtain the CL model connected to a PD-based feedback/feedforward CACC system. It is proved how the resulting model is improved in comparison with an OL identification algorithm– ARX. Notice that the same algorithm is used in both identification processes, but one identifies the plant directly and other S_i. As described in section 4.4, YK parameterization is associated with the vehicle dynamics identification. Since a particular operating point was used for modeling the experimental platform, better behaviour can be obtained if vehicle model well captures uncertainties. Model with uncertainties can be identified by employing the dual YK parameter S_i. Any change in the dual YK parameter S_i can induce a change in the YK parameter Q_i, so better performance and safe switching are ensured. However, the order of S_i is crucial. A high-order S_i could slow down both the identification process and some of the iterative solutions proposed in the literature. In a future work, it would be interesting to study order reduction techniques in relation to both YK and dual YK parameterizations.

And finally, both theorems are merged for developing a MMAC application able to deal with vehicles heterogeneity in FOPD feedback/feedforward MIMO CACC string of vehicles. The main idea of MMAC is to choose the plant in a predefined set that best approximates the system dynamics, applying the corresponding string stable predesigned controller. Matching and non-matching cases are considered in simulation, showing how the supervisor is able to choose the closest model in the set to perform with stability. Dynamics estimation through dual YK parameterization results much more faster than other methods in the literature. Controller reconfiguration is based in the YK parameterization. Simulation results are validated at INRIA low-speed experimental platform. Although the set $\{G\}$ is composed by three plants, this could be extended to a larger number of plants, so a larger dynamic range can be covered in heterogeneous CACC applications. This application could be integrated with ACACC and cut-in/out maneuvers applications in order to have a general framework for CACC applications.

Results in this chapter are useful to see the potential of this control approach. YK basis has been applied to CACC applications. Different type of controllers and structures have been considered in order to prove its adaptability: Classical SISO PD controller, extended SISO PD controller, extended SISO fractional order PD controller and extended MIMO FOPD controller.

The vast majority of applications in the literature go through the design of the YK Q that provides a performance criteria. Throughout this chapter, the idea has been rather the opposite; let's fix Q ensuring the stability of the system, and then let's see how γ can be modified in order to improve the performance of the system. In this way, in the first of the applications (ACACC) a heuristic design of γ is highlighted to improve the traffic flow. The most part of γ is not used. On the contrary, later, the whole range of γ is employed to perform stable cut-in/out maneuvers. A more clear relation exists between time gap and γ. Finally, in MMAC, several γ's are used in order to activate the controller that makes string stable a heterogeneous string of vehicles through the corresponding Q. The evolution is clear, from heuristic to several γ's design, we strongly believe that γ design plays a key role in performance improvement.

Adaptability, stability, real implementation, and extension to many other applications present in the state of the art lead us to conclude that Youla-Kucera could serve as a general control approach for secure responses in autonomous driving.

Chapter 6

Conclusions

While the long-term benefits of autonomous vehicles are genuinely exciting, the route to true autonomy in transportation will likely to be long and full of uncertainty. Nowadays solutions are mainly oriented to advanced multi-sensor fusion systems toward multi-target decision-making systems. These systems increase uncertainty and complexity when controlling an autonomous vehicle, as different control systems are activated depending on the multi-sensor decision system. Both longitudinal and lateral controls, are subdivided into simpler control systems working for a specific operation point, traffic situation and/or sensors/actuators availability. Classical, optimal, robust, adaptive or fault tolerant control solutions are found independently for specific use-cases, but a general control framework integrating all solutions and providing stability guarantees is missing. To the best of author knowledge, Youla-Kucera (YK) parameterization is the only control framework which encompasses all single control solutions with stability proof. Chapter 2 presented a broad overview of this mathematical framework, focusing in origins, robust stabilization results and YK-based applications developed by the Australian National University, Technical University of Denmark and Aalborg University. Moreover, there are still some milestones in the mathematical framework as non-linear extension of the YK parameterization; integration of intelligent control systems; transient bevavior for the different YK-based control structures for switching; analysis of the scalar factor γ, regulating the action of different controllers in order to improve system's performance; extension of fault-tolerant control solutions to a more general control structure (they are all based on observer-based feedback controllers); and experimental results under fast-dynamics cases, as it could be an automated vehicle.

The present Ph.D. thesis addressed some of the aforementioned challenges by proposing YK as methodology that could improve the security of autonomous driving systems. This control framework manages different sensor/actuator setups, dynamics and traffic situations with stability guarantees. Special emphasis was on stability when dynamics change or traffic situation demands controller reconfiguration. These changes could be a priori known, or identified in order to have an adaptive solution. Stable controller reconfiguration, adaptive control and Closed-Loop (CL) identification were the main problems addressed in chapters 3 and 4.

Chapter 3 reviewed YK parameterization basis, examining the use of doubly coprime factors to parameterize the class of all stabilizing controllers for a plant in terms of an initial stabilizing controller and a stable filter Q. The filter Q was calculated in order to perform stable controller reconfiguration between two or several controllers. Calculation of Q and coprime factors organisation vary depending on the selected YK-based control structure in the literature. Each of them can be conceived for dealing with problems such order complexity, plant disconnection or matrix inversability. Even if stability properties were preserved when using different structures, the transient behavior changed with the same step switching signal γ. From faster to slower responses, oscillations or not, the best structure for controller switching resulted the number three (section 3.4.3 for details). It presented the best transient behavior wihtout oscillations, a lower order controller complexity and no need to disconnect the initial controller. Consequently, this structure was used all along this Ph.D. thesis.

A dual version of the YK parameterization (all plants stabilized by a given controller) was presented in chapter 4. This is the basis for solving CL identification problems, by their reduction to standard Open-Loop (OL) problems. It has also been applied to an adaptive control methodology. Instead of having a stable filter Q, a filter S was introduced, representing mistmatch between model and real plant. An OL identification of this parameter permits CL identification of the real system without noise correlation problems. If the identified S is unstable, the loop would be unstable, and a filter Q needs to found to re-stabilize the loop. On the contrary, Q should be optimized for obtaining a better performance. Q and S could be used in an iterative/nested algorithms as those in the state-of-the-art of this thesis. However, with the idea of remaining close to the industrial setup, a YK-based MMAC algorithm based on a set of nominal plants and predefined controllers was proposed. MMAC is a supevisor choosing the closer plant to the real system in the set, switching to the best controller possible without identification or optimization processes that slow down the control loop.

Once both YK and dual YK basis were explained; stable controller reconfiguration, CL identification and MMAC approach were tested in Cooperative Adaptive Cruise Control (CACC) equipped string of vehicles. Simulation and experimental results were obtained in a high-speed simulation model and INRIA's low-speed experimental platform respectively. These results not only helped to obtain some of the missing experimental results in the YK state-of-the-art, but also improved the state-of-the-art of CACC systems. Hybrid behavior between CACC controllers with different time gaps was explored by means of the YK parameterization, in order to avoid Adaptive Cruise Control (ACC) degradation when the communication link with the preceding vehicle was not available; and ensure stability when other vehicles performed cut-in/out maneuvers. In both applications, Q was fixed ensuring the stability of the system, and the scalar factor γ was designed to improve system's performance. A heuristic design of γ was done in the first application for reducing intervehicle distances, and then improving traffic flow, in comparison with an ACC degradation. Communication link with a vehicle ahead (different from the preceding one) was considered. The better performance is achieved with a limited range of γ. On the contrary, in the second application, the full range of γ was employed, opening inter-vehicle gap when a vehicle cuts in, and recovering distances once the cut-in is finished, or when a vehicle goes away in a stable manner. In the other hand, dual YK properties have also been analysed; concretely, connection with a CACC system served as a demonstration of the importance of noise correlation when doing CL identification, and how OL identification of the dual YK parameter S avoids these problems. Finally, a Multi Model Adaptive Control (MMAC) application considers both YK and dual YK parameterization. A set of nominal models was used for designing CACC systems for heterogeneous string of vehicles. In order to ensure string stability of the system, both ego and preceding vehicles models need to be known. Longitudinal dynamics of two vehicles in a CACC string were estimated within a model set, so the proper CACC system was chosen guaranteeing string stability. Dynamics estimation results much more faster than other estimation processes in the literature. This last application, not only consider a unique Q, but several from an initial homogeneous-string controller to others considering possible dynamics combinations between ego and preceding vehicles dynamics. Several γ's were employed in order to activate string stable controllers. The evolution is clear, from heuristic to several γ's design, γ plays a key role in performance improvement.

Through these applications different types of controllers and structures have been used, proving the possible adaptability of the YK parameterization to any controller type/structure. Stability and real implementation is also demonstrated in fast dynamics systems as vehicles. We strongly believe that Youla-Kucera could serve as a general control approach for secure responses in autonomous driving.

6.1 Contributions to the state of the art

Main contributions of this Ph.D. thesis are listed below:

6.1.1 Youla-Kucera parameterization

- A wide review of YK-based applications is carried out.

- Transient behavior analysis for the different YK-based control structures for switching. Structure 3 is the one which transient behavior does not present oscillations with a lower order controller complexity and no need to disconnect the initial controller.

- The vast majority of YK-based applications in the state-of-the-art considers the optimal value of Q for obtaining the best performance possible. Here, given a Q between different controllers, the design of the switching signal γ is addressed for improving system's performance.

- State-of-the-art cases are mainly focused in systems with very low dynamics, except from some simulation with an aircraft model in high performance control. Here, experimental faster dynamics cases as a vehicle are considered, proving the practical application of YK parameterization in these cases.

- YK basis has been applied to different controllers types and structures, extending state-of-the-art results based on observer-based state feedback controllers.

6.1.2 CACC

- The benefit of using V2V-equipped vehicle information from a vehicle ahead in the string when the preceding vehicle is a non-equipped one is explored, providing stable responses and, more interestingly, reducing intervehicle distances in comparison with the literature CACC degradation to ACC.

- A control algorithm able to ensure stable cut-in/out maneuvers in CACC string of vehicles.

- CL identification of longitudinal dynamics of INRIA's low speed experimental platform.

- An adaptive approach without need of idenfification algortihms is proposed for heterogeneous string of vehicles. An estimation process based on the dual YK parameterization is used instead, being much faster than related works in the state-of-the-art.

6.1.3 Autonomous driving

- A new layer connecting acquisition, perception and decision blocks with the different longitudinal and lateral control applications is considered. This new layer is formed by the YK control framework, and its objective is to ensure stable responses in autonomous driving under dynamics, traffic or instrumental changes.

6.2 Future research directions

- Extension of YK parameterization to intelligent control systems as fuzzy control, neuronal networks and Model Predictive Control (MPC).

- Non-linear extension of the YK parameterization according to the control objective needs.

- Controller reduction methods for iterative (Q, S) applications.

- Analysis of effects of switching signal γ frequency in CL stability.

- Stable controller reconfiguration between controllers with different purposes needs to be studied. An example would be stable controller reconfiguration between emergency braking and cruise control.

- Integration of the different CACC solutions in the same YK framework for having a general CACC approach dealing with non-available communication with preceding vehicle, entering/exiting vehicles and different dynamics. The nominal model set in MMAC approach should be increased to obtain a general solution to string heterogeneity.

- Optimal Q design depending on the identified dual YK parameter S for obtaining the best performance possible when large systems variations are present in the vehicle.

- YK-based FTC application for dealing with different sensors/actuators availability in autonomous driving.

- Application of YK parameterization to lateral control approaches in Intelligent Transpotation Systems (ITS) domain.

Bibliography

[Alam et al., 2015] Alam, A., Mårtensson, J. and Johansson, K. H., 2015. Experimental evaluation of decentralized cooperative cruise control for heavy-duty vehicle platooning. Control Engineering Practice 38, pp. 11–25.

[Anderson et al., 2001] Anderson, B., Brinsmead, T., Liberzon, D. and Stephen Morse, A., 2001. Multiple model adaptive control with safe switching. International journal of adaptive control and signal processing 15(5), pp. 445–470.

[Anderson, 1998] Anderson, B. D., 1998. From youla–kucera to identification, adaptive and nonlinear control. Automatica 34(12), pp. 1485–1506.

[Anderson et al., 2000] Anderson, B. D., Brinsmead, T. S., De Bruyne, F., Hespanha, J., Liberzon, D. and Morse, A. S., 2000. Multiple model adaptive control. part 1: Finite controller coverings. International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal 10(11-12), pp. 909–929.

[Arem et al., 2006] Arem, B. V., Driel, C. J. V. and Visser, R., 2006. The impact of cooperative adaptive cruise control on traffic-flow characteristics. IEEE Transactions on Intelligent Transportation Systems 7(4), pp. 429–436.

[Baldi et al., 2011] Baldi, S., Battistelli, G., Mari, D., Mosca, E. and Tesi, P., 2011. Multiple-model adaptive switching control for uncertain multivariable systems. In: Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, IEEE, pp. 6153–6158.

[Barooah and Hespanha, 2005] Barooah, P. and Hespanha, J. P., 2005. Error amplification and disturbance propagation in vehicle strings with decentralized linear control. In: Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on, IEEE, pp. 4964–4969.

[Bendtsen and Trangbaek, 2012] Bendtsen, J. and Trangbaek, K., 2012. Multiple model adaptive control using dual youla-kucera factorisation. IFAC Proceedings Volumes 45(13), pp. 63–68.

[Bendtsen and Trangbaek, 2014] Bendtsen, J. and Trangbaek, K., 2014. Closed-loop identification for control of linear parameter varying systems. Asian Journal of Control 16(1), pp. 40–49.

[Bendtsen et al., 2008] Bendtsen, J., Trangbaek, K. and Stoustrup, J., 2008. Closed-loop system identification with new sensors. In: Decision and Control, 2008. CDC 2008. 47th IEEE Conference on, IEEE, pp. 2631–2636.

[Bendtsen et al., 2013] Bendtsen, J., Trangbaek, K. and Stoustrup, J., 2013. Plug-and-play control modifying control systems online. IEEE Transactions on Control Systems Technology 21(1), pp. 79–93.

[Bruzelius, 2004] Bruzelius, F., 2004. Linear Parameter Varying Systems. PhD thesis, Chalmers University of Technology.

[Bu et al., 2010] Bu, F., Tan, H.-S. and Huang, J., 2010. Design and field testing of a cooperative adaptive cruise control system. In: American Control Conference (ACC), 2010, IEEE, pp. 4616–4621.

[Chakravarty and Moore, 1986] Chakravarty, A. and Moore, J., 1986. Aircraft flutter suppression via adaptive lqg control. In: American Control Conference, 1986, IEEE, pp. 488–493.

[Chakravarty et al., 1986] Chakravarty, A., Moore, J. et al., 1986. Flutter suppression using central tendency adaptive pole assignment. In: Third Conference on Control Engineering 1986: Towards a More Competitive Industry; Preprints of Papers, The, Institution of Engineers, Australia, p. 78.

[Commision, 2012] Commision, E., 2012. Road transport. a change of gear.

[Dahleh and Pearson, 1986] Dahleh, M. A. and Pearson, J., 1986. l 1-optimal-feedback controllers for discrete-time systems. In: American Control Conference, 1986, IEEE, pp. 1964–1968.

[De Bruyne et al., 1998] De Bruyne, F., Anderson, B. and Linard, N., 1998. The hansen scheme revisited. In: Decision and Control, 1998. Proceedings of the 37th IEEE Conference on, Vol. 1, IEEE, pp. 706–711.

[Desoer et al., 1980] Desoer, C., Liu, R.-W., Murray, J. and Saeks, R., 1980. Feedback system design: The fractional representation approach to analysis and synthesis. IEEE Transactions on Automatic Control 25(3), pp. 399–412.

[Dey et al., 2016] Dey, K. C., Yan, L., Wang, X., Wang, Y., Shen, H., Chowdhury, M., Yu, L., Qiu, C. and Soundararaj, V., 2016. A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC). IEEE Trans. Intelligent Transportation Systems 17(2), pp. 491–509.

[Douma et al., 2003] Douma, S. G., Van den Hof, P. M. and Bosgra, O. H., 2003. Controller tuning freedom under plant identification uncertainty: double youla beats gap in robust stability. Automatica 39(2), pp. 325–333.

[Doyle, 1983] Doyle, J. C., 1983. Synthesis of robust controllers and filters. In: Decision and Control, 1983. The 22nd IEEE Conference on, Vol. 22, IEEE, pp. 109–114.

[Doyle et al., 1991] Doyle, J., Packard, A. and Zhou, K., 1991. Review of lfts, lmis, and mu. In: Decision and Control, 1991., Proceedings of the 30th IEEE Conference on, IEEE, pp. 1227–1232.

[Englund et al., 2016] Englund, C., Chen, L., Ploeg, J., Semsar-Kazerooni, E., Voronov, A., Bengtsson, H. H. and Didoff, J., 2016. The grand cooperative driving challenge 2016: boosting the introduction of cooperative automated vehicles. IEEE Wireless Communications 23(4), pp. 146–152.

[Fujimori, 1993] Fujimori, A., 1993. Parameterizations of stabilizing compensators by using reduced-order observers. IEEE transactions on automatic control 38(9), pp. 1435–1439.

[Gao et al., 2015] Gao, F., Dang, D. and Li, S. E., 2015. Control of a heterogeneous vehicular platoon with uniform communication delay. In: Information and Automation, 2015 IEEE International Conference on, IEEE, pp. 2419–2424.

[Gao et al., 2016] Gao, F., Li, S. E., Zheng, Y. and Kum, D., 2016. Robust control of heterogeneous vehicular platoon with uncertain dynamics and communication delay. IET Intelligent Transport Systems 10(7), pp. 503–513.

[Gevers et al., 2001] Gevers, M., Ljung, L. and Van den Hof, P., 2001. Asymptotic variance expressions for closed-loop identification. Automatica 37(5), pp. 781–786.

[Guo and Yue, 2011] Guo, G. and Yue, W., 2011. Hierarchical platoon control with heterogeneous information feedback. IET control theory & applications 5(15), pp. 1766–1781.

[Hansen et al., 1989] Hansen, F., Franklin, G. and Kosut, R., 1989. Closed-loop identification via the fractional representation: Experiment design. In: American Control Conference, pp. 1422–1427.

[Harfouch et al., 2017] Harfouch, Y. A., Yuan, S. and Baldi, S., 2017. Adaptive control of interconnected networked systems with application to heterogeneous platooning. In: Control & Automation (ICCA), 2017 13th IEEE International Conference on, IEEE, pp. 212–217.

[Hedrick and Swaroop, 1994] Hedrick, J. and Swaroop, D., 1994. Dynamic coupling in vehicles under automatic control. Vehicle System Dynamics 23(S1), pp. 209–220.

[Hespanha and Morse, 2002] Hespanha, J. P. and Morse, A. S., 2002. Switching between stabilizing controllers. Automatica 38(11), pp. 1905–1917.

[Hippe, 1989] Hippe, P., 1989. Modified doubly coprime fractional representations related to the reduced-order observer. IEEE transactions on automatic control 34(5), pp. 573–576.

[Hjalmarsson et al., 1996] Hjalmarsson, H., Gevers, M. and Bruyne, F. D., 1996. For model-based control design, closed-loop identification gives better performance. Automatica 32(12), pp. 1659–1673.

[Hsu et al., 2012] Hsu, C. W., Hsu, T. H. and Chang, K. J., 2012. Implementation of car-following system using lidar detection. In: ITS Telecommunications (ITST), 2012 12th International Conference on, IEEE, pp. 165–169.

[ICCT, 2016] ICCT, 2016. European vehicle market statistics.

[Ishihara and Sales, 1999] Ishihara, J. Y. and Sales, R. M., 1999. Doubly coprime factorizations related to any stabilizing controllers in state space. Automatica 35(9), pp. 1573–1577.

[Karaboyas and Kalouptsidis, 1991] Karaboyas, S. and Kalouptsidis, N., 1991. Efficient adaptive algorithms for arx identification. IEEE Transactions on signal processing 39(3), pp. 571–582.

[Kesting et al., 2007] Kesting, A., Treiber, M. and Helbing, D., 2007. Extending adaptive cruise control to adaptive driving strategies. Transportation Research Record: Journal of the Transportation Research Board.

[Kesting et al., 2010] Kesting, A., Treiber, M. and Helbing, D., 2010. Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 368(1928), pp. 4585–4605.

[Kesting et al., 2008] Kesting, A., Treiber, M., Schonhof, M. and Helbing, D., 2008. Adaptive cruise control design for active congestion avoidance. Transportation Research Part C: Emerging Technologies 16(6), pp. 668–683.

[Khatir and Davison, 2004] Khatir, M. E. and Davison, E. J., 2004. Decentralized control of a large platoon of vehicles using non-identical controllers. In: American Control Conference, 2004. Proceedings of the 2004, Vol. 3, IEEE, pp. 2769–2776.

[Kianfar et al., 2012] Kianfar, R., Augusto, B., Ebadighajari, A., Hakeem, U., Nilsson, J., Raza, A., Tabar, R. S., Irukulapati, N. V., Englund, C., Falcone, P., Papanastasiou, S., Svensson, L. and Wymeersch, H., 2012. Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge. IEEE Trans. Intelligent Transportation Systems 13(3), pp. 994–1007.

[Knudsen et al., 2008] Knudsen, T., Trangbaek, K. and Kallesse, C. S., 2008. Plug and play process control applied to a district heating system. IFAC Proceedings Volumes 41(2), pp. 325–330.

[Kučera, 1975] Kučera, V., 1975. Stability of discrete linear feedback systems. IFAC Proceedings Volumes 8(1), pp. 573–578.

[Lestas and Vinnicombe, 2007] Lestas, I. and Vinnicombe, G., 2007. Scalability in heterogeneous vehicle platoons. In: American Control Conference, 2007. ACC'07, IEEE, pp. 4678–4683.

[Li et al., 2010] Li, H., Luo, Y. and Chen, Y., 2010. A fractional order proportional and derivative (fopd) motion controller: Tuning rule and experiments. IEEE Transactions on Control Systems Technology 18(2), pp. 516–520.

[Lidstrom et al., 2012] Lidstrom, K., Sjoberg, K., Holmberg, U., Andersson, J., Bergh, F., Bjade, M. and Mak, S., 2012. A modular cacc system integration and design. IEEE Trans. Intelligent Transportation Systems 13(3), pp. 1050–1061.

[Lin et al., 1993] Lin, J.-L., Postlethwaite, I. and Gu, D.-W., 1993. μ-k iteration: a new algorithm for μ-synthesis. Automatica 29(1), pp. 219–224.

[Linard and Anderson, 1997] Linard, N. and Anderson, B., 1997. Identification of nonlinear plants using left coprime fractional based representations. In: Control Conference (ECC), 1997 European, IEEE, pp. 772–777.

[Linard and Anderson, 1996] Linard, N. and Anderson, B. D., 1996. Identification of nonlinear plants under linear control using youla-kucera parametrizations. In: Decision and Control, 1996., Proceedings of the 35th IEEE Conference on, Vol. 1, IEEE, pp. 1094–1098.

[Lourenco and Lemos, 2006] Lourenco, J. and Lemos, J., 2006. Learning in multiple model adaptive control switch. IEEE instrumentation & measurement Magazine 9(3), pp. 24–29.

[Luettel et al., 2012] Luettel, T., Himmelsbach, M. and Wuensche, H.-J., 2012. Autonomous ground vehicles-concepts and a path to the future. Proceedings of the IEEE 100(Centennial-Issue), pp. 1831–1839.

[Marsden et al., 2001] Marsden, G., Mcdonals, M. and Bracktone., M., 2001. Towards an understanding of adaptive cruise control. Transportation Research Part C: Emerging Technologies 9(1), pp. 33–51.

[Milanés and Shladover, 2014] Milanés, V. and Shladover, S. E., 2014. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. Transportation Research Part C: Emerging Technologies 48, pp. 285–300.

[Milanés and Shladover, 2016] Milanés, V. and Shladover, S. E., 2016. Handling cut-in vehicles in strings of cooperative adaptive cruise control vehicles. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations 20(2), pp. 178–191.

[Milanés et al., 2014] Milanés, V., Shladover, S., Spring, J., Nowakowski, C., Kawazoe, H. and Nakamura, M., 2014. Cooperative adaptive cruise control in real traffic situations. Intelligent Transportation Systems, IEEE Transactions on 15(1), pp. 296–305.

[Moore and Tay, 1989] Moore, J. and Tay, T., 1989. Loop reconvery via hinf/h2 sensitivity recovery. International Journal Control 49(4), pp. 1249–1271.

[Moore et al., 1990] Moore, J. B., Glover, K. and Telford, A., 1990. All stabilizing controllers as frequency-shaped state estimate feedback. IEEE Transactions on Automatic Control 35(2), pp. 203–208.

[Moore et al., 1989] Moore, J., Lige, X., Yang, X. et al., 1989. On active resonance and flutter suppression techniques. In: Australian Aeronautical Conference, 1989: Research and Technology, the Next Decade, Melbourne, 9-11 October 1989; Preprints of Papers, The, Institution of Engineers, Australia, p. 181.

[Morari and Zafiriou, 1989] Morari, M. and Zafiriou, E., 1989. Robust process control. Morari.

[Naus, 2010] Naus, G., 2010. Model-based control for automotive applications.

[Naus et al., 2010] Naus, G. J., Vugts, R. P., Ploeg, J., van de Molengraft, M. J. and Steinbuch, M., 2010. String-stable cacc design and experimental validation: A frequency-domain approach. IEEE Transactions on vehicular technology 59(9), pp. 4268–4279.

[Navas et al., 2016] Navas, F., Milanes, V. and Nashashibi, F., 2016. Using plug&play control for stable acc-cacc system transitions. In: In Intelligent Vehicles Symposium.

[Nett et al., 1984] Nett, C., Jacobson, C. and Balas, M., 1984. A connection between state-space and doubly coprime fractional representations. IEEE Transactions on Automatic Control 29(9), pp. 831–832.

[Newton et al., 1957] Newton, G. C., Gould, L. A. and Kaiser, J. F., 1957. Analytical design of linear feedback controls.

[Niemann, 2003] Niemann, H., 2003. Dual youla parameterisation. IEE Proceedings-Control Theory and Applications 150(5), pp. 493–497.

[Niemann, 2006a] Niemann, H., 2006a. Parameterisation of extended systems. IEE Proceedings-Control Theory and Applications 153(2), pp. 221–227.

[Niemann, 2006b] Niemann, H., 2006b. A setup for active fault diagnosis. IEEE Transactions on Automatic Control 51(9), pp. 1572–1578.

[Niemann, 2012] Niemann, H., 2012. A model-based approach to fault-tolerant control. International Journal of Applied Mathematics and Computer Science 22(1), pp. 67–86.

[Niemann and Poulsen, 2009a] Niemann, H. and Poulsen, N. K., 2009a. Controller architectures for switching. In: American Control Conference, 2009. ACC'09., IEEE, pp. 1098–1103.

[Niemann and Poulsen, 2009b] Niemann, H. and Poulsen, N. K., 2009b. Fault tolerant control-a residual based set-up. In: Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on, IEEE, pp. 8470–8475.

[Niemann and Stoustrup, 1999] Niemann, H. and Stoustrup, J., 1999. An architecture for implementation of multivariable controllers. In: American Control Conference, 1999. Proceedings of the 1999, Vol. 6, IEEE, pp. 4029–4033.

[Niemann and Stoustrup, 2000] Niemann, H. and Stoustrup, J., 2000. Tuning controllers using the dual youla parameterization. In: American Control Conference, 2000. Proceedings of the 2000, Vol. 2, IEEE, pp. 1337–1338.

[Niemann and Stoustrup, 2002] Niemann, H. and Stoustrup, J., 2002. Reliable control using the primary and dual youla parameterizations. In: Decision and Control, 2002, Proceedings of the 41st IEEE Conference on, Vol. 4, IEEE, pp. 4353–4358.

[Niemann and Stoustrup, 2004] Niemann, H. and Stoustrup, J., 2004. Fault tolerant controllers for sampled-data systems. In: American Control Conference, 2004. Proceedings of the 2004, Vol. 4, IEEE, pp. 3490–3495.

[Niemann and Stoustrup, 2005] Niemann, H. and Stoustrup, J., 2005. An architecture for fault tolerant controllers. International Journal of Control 78(14), pp. 1091–1110.

[Niemann and Stroustrup, 1999] Niemann, H. and Stroustrup, J., 1999. Gain scheduling using the youla parameterization. In: Decision and Control, 1999. Proceedings of the 38th IEEE Conference on, Vol. 3, IEEE, pp. 2306–2311.

[Niemann, 1999] Niemann, H. H., 1999. Application of the dual youla parameter. In: Control Conference (ECC), 1999 European, IEEE, pp. 2339–2344.

[Niemann et al., 1991] Niemann, H. H., Søgaard-Andersen, P. and Stoustrup, J., 1991. Loop transfer recovery for general observer architectures. International Journal of Control 53(5), pp. 1177–1203.

[Niemann et al., 2004] Niemann, H., Stoustrup, J. and Abrahamsen, R. B., 2004. Switching between multivariable controllers. Optimal control applications and methods 25(2), pp. 51–66.

[Nieuwenhuijze et al., 2012] Nieuwenhuijze, M. R., van Keulen, T., O?ncu?, S., Bonsen, B. and Nijmeijer, H., 2012. Cooperative driving with a heavy-duty truck in mixed traffic: Experimental results. IEEE Transactions on Intelligent Transportation Systems 13(3), pp. 1026–1032.

[Nowakowski et al., 2010a] Nowakowski, C., O'Connell, J., Shladover, S. and Cody, D., 2010a. Cooperative adaptive cruise control: Driver acceptance of following gap settings less than one second. In: Human Factors and Ergonomics Society Annual Meeting, Vol. 3number 24, pp. 2033–2037.

[Nowakowski et al., 2010b] Nowakowski, C., O'Connell, J., Shladover, S. E. and Cody, D., 2010b. Cooperative adaptive cruise control: Driver acceptance of following gap settings less than one second. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 54number 24, SAGE Publications Sage CA: Los Angeles, CA, pp. 2033–2037.

[Ogata, 2013] Ogata, K., 2013. Modern control engineering,(1997). ISBN: 0-13-227307-1 pp. 299–231.

[Öncü et al., 2011] Öncü, S., Van De Wouw, N. and Nijmeijer, H., 2011. Cooperative adaptive cruise control: Tradeoffs between control and network specifications. In: Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on, IEEE, pp. 2051–2056.

[Pacejka, 2005] Pacejka, H., 2005. Tire and vehicle dynamics. Elsevier.

[Ploeg et al., 2011a] Ploeg, J., Scheepers, B. T., van Nunen, E., van de Wouw, N. and Nijmeijer, H., 2011a. Design and experimental evaluation of cooperative adaptive cruise control. In: In 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 260–265.

[Ploeg et al., 2013] Ploeg, J., Semsar-Kazerooni, E., Lijster, G., van de Wouw, N. and Nijmeijer, H., 2013. Graceful degradation of cacc performance subject to unreliable wireless communication. In: In 16th International IEEE Conference on Intelligent Transportation Systems, pp. 1210–1216.

[Ploeg et al., 2011b] Ploeg, J., Serrarens, A. and Heijenk, G., 2011b. Connect & drive: design and evaluation of cooperative adaptive cruise control for congestion reduction. Journal of Modern Transportation 19(3), pp. 207–213.

[Pommaret and Quadrat, 1998] Pommaret, J. F. and Quadrat, A., 1998. Generalized bezout identity. In: Aplicable Algebra in Engineering, Communication and Computing, Vol. 9number 2, pp. 91–116.

[Rajamani, 2011] Rajamani, R., 2011. Vehicle dynamics and control.

[Redheffer, 1960] Redheffer, R., 1960. On a certain linear fractional transformation. Studies in Applied Mathematics 39(1-4), pp. 269–286.

[Salcedo and Martinez, 2008] Salcedo, J. V. and Martinez, M., 2008. Lpv identification of a turbocharged diesel engine. Applied Numerical Mathematics 58(10), pp. 1553–1571.

[Santos et al., 2007] Santos, P. L. D., Ramos, J. A. and de Carvalho, J. M., 2007. Identification of linear parameter varying systems using an iterative deterministic-stochastic subspace approach. In: In European Control Conference European, pp. 4867–4873.

[Savkin and Petersen, 1995] Savkin, A. V. and Petersen, I. R., 1995. Recursive state estimation for uncertain systems with an integral quadratic constraint. IEEE Transactions on Automatic Control 40(6), pp. 1080–1083.

[Schrama, 1991] Schrama, R., 1991. An open-loop solution to the approximate closed-loop identification problem. IFAC Proceedings Volumes 24(3), pp. 761–766.

[Seiler et al., 2004] Seiler, P., Pant, A. and Hedrick, K., 2004. Disturbance propagation in vehicle strings. IEEE Transactions on automatic control 49(10), pp. 1835–1842.

[Sekhavat and Hermosillo, 2000] Sekhavat, S. and Hermosillo, J., 2000. The cycab robot: a differentially flat system. In: Intelligent Robots and Systems, 2000.(IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on, Vol. 1, IEEE, pp. 312–317.

[Sekunda et al., 2015] Sekunda, A., Niemann, H., Poulsen, N. K. and Santos, I., 2015. Closed loop identification using a modified hansen scheme. In: Journal of Physics: Conference Series, Vol. 659number 1, IOP Publishing, p. 012009.

[Shaw and Hedrick, 2007a] Shaw, E. and Hedrick, J. K., 2007a. Controller design for string stable heterogeneous vehicle strings. In: Decision and Control, 2007 46th IEEE Conference on, IEEE, pp. 2868–2875.

[Shaw and Hedrick, 2007b] Shaw, E. and Hedrick, J. K., 2007b. String stability analysis for heterogeneous vehicle strings. In: American Control Conference, 2007. ACC'07, IEEE, pp. 3118–3125.

[Shladover et al., 2012] Shladover, S., Su, D. and Lu, X., 2012. Impacts of cooperative adaptive cruise control on freeway traffic flow. In: 91st Transportation Research Board Annual Meeting.

[Stefanovic et al., 2004] Stefanovic, M., Wang, R. and Safonov, M. G., 2004. Stability and convergence in adaptive systems. In: American Control Conference, 2004. Proceedings of the 2004, Vol. 2, IEEE, pp. 1923–1928.

[Stoustrup, 2009] Stoustrup, J., 2009. Plug & play control: Control technology towards new challenges. In: In European Control Conference, pp. 1668–1683.

[Swaroop, 1997] Swaroop, D., 1997. String stablility of interconnected systems: An application to platooning in automated highway systems. Technical report, California PATH Research, UC Berkeley.

[Swaroop and Rajagopal, 2001] Swaroop, D. and Rajagopal, K., 2001. A review of constant time headway policy for automatic vehicle following. In: Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE, IEEE, pp. 65–69.

[Sznaier and Mazzaro, 2003] Sznaier, M. and Mazzaro, M. C., 2003. An lmi approach to control-oriented identification and model (in) validation of lpv systems. IEEE Transactions on Automatic Control 48(9), pp. 1619–1624.

[Tay and Moore, 1991] Tay, T. and Moore, J. B., 1991. Enhancement of fixed controllers via adaptive-q disturbance estimate feedback. Automatica 27(1), pp. 39–53.

[Tay et al., 1989a] Tay, T., Moore, J. and Horowitz, R., 1989a. Indirect adaptive techniques for fixed controller performance enhancement. International Journal of Control 50(5), pp. 1941–1959.

[Tay et al., 1989b] Tay, T. T. et al., 1989b. Enhancing robust controllers with adaptive techniques.

[Tay et al., 1997] Tay, T.-T., Mareels, I. and Moore, J. B., 1997. High performance control. Springer Science & Business Media.

[Telford and Moore, 1989] Telford, A. J. and Moore, J. B., 1989. Doubly coprime factorizations, reduced-order observers, and dynamic state estimate feedback. International Journal of Control 50(6), pp. 2583–2597.

[Teo and Tay, 1995] Teo, Y. and Tay, T., 1995. Design of al/sup 1/-optimal regulator: the limits of performance approach. IEEE transactions on automatic control 40(12), pp. 2114–2119.

[Teo and Tay, 1996] Teo, Y. and Tay, T. T., 1996. Application of the l/sup 1/-optimal regulation strategy to a hard disk servo system. IEEE transactions on control systems technology 4(4), pp. 467–472.

[Trangbaek, 2009] Trangbaek, K., 2009. Controller reconfiguration through terminal connections based on closed-loop system identification1. IFAC Proceedings Volumes 42(6), pp. 25–30.

[Trangbaek, 2011] Trangbaek, K., 2011. Safe lpv controller switching. In: Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, IEEE, pp. 3428–3433.

[Trangbaek and Bendtsen, 2009] Trangbaek, K. and Bendtsen, J., 2009. Stable controller reconfiguration through terminal connections. a practical example. In: Control and Automation, 2009. ICCA 2009. IEEE International Conference on, IEEE, pp. 2037–2042.

[Trangbaek and Bendtsen, 2010] Trangbaek, K. and Bendtsen, J., 2010. Lpv identification of a heat distribution system. In: IEEE International Conference In Control Applications, pp. 2178–2183.

[Trangbaek et al., 2008] Trangbaek, K., Stoustrup, J. and Bendtsen, J., 2008. Stable controller reconfiguration through terminal connections. IFAC Proceedings Volumes 41(2), pp. 331–335.

[van Wingerden, 2012] van Wingerden, J. W., 2012. The asymptotic variance of the pbsidopt algorithm. IFAC Proceedings 45(16), pp. 1167–1172.

[Vidyasagar, 1985] Vidyasagar, M., 1985. Control systems synthesis: A factorization approach, volume 8 of north holland system and control series.

[Vidyasagar, 1991] Vidyasagar, M., 1991. Further results on the optimal rejection of persistent bounded disturbances. IEEE Transactions on Automatic Control 36(6), pp. 642–652.

[Vinnicombe, 2000] Vinnicombe, G., 2000. Uncertainty and Feedback: Hinf loop-shaping and the v-gap metric. World Scientific.

[Wang and Nijmeijer, 2015] Wang, C. and Nijmeijer, H., 2015. String stable heterogeneous vehicle platoon using cooperative adaptive cruise control. In: Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on, IEEE, pp. 1977–1982.

[Wang et al., 2015] Wang, D. Z., Posner, I. and Newman, P., 2015. Model-free detection and tracking of dynamic objects with 2d lidar. The International Journal of Robotics Research 34(7), pp. 1039–1063.

[Wang, 1991] Wang, Z., 1991. Performance Issues in Adaptive Control. PhD thesis, University of Newcastle.

[Wang et al., 1991] Wang, Z., Mareels, I. and Moore, J., 1991. Adaptive disturbance rejection. In: Decision and Control, 1991., Proceedings of the 30th IEEE Conference on, IEEE, pp. 2836–2841.

[Werf et al., 2002] Werf, J. V., Shladover, S., Miller, M. and Kourjanskaia., N., 2002. Effects of adaptive cruise control systems on highway traffic flow capacity. Transportation Research Record: Journal of the Transportation Research Board (1800), pp. 78–84.

[Yan and Moore, 1992] Yan, W. and Moore, J. B., 1992. A multiple controller structure and design strategy with stability analysis. Automatica 28(6), pp. 1239–1244.

[Yan and Moore, 1996] Yan, W.-Y. and Moore, J. B., 1996. Stable linear fractional transformations with applications to stabilization and multistage hˆ control design. International Journal of Robust and Nonlinear Control 6(2), pp. 101–122.

[Youla et al., 1976a] Youla, D., Bongiorno, J. d. and Jabr, H., 1976a. Modern wiener–hopf design of optimal controllers part i: The single-input-output case. IEEE Transactions on Automatic Control 21(1), pp. 3–13.

[Youla et al., 1976b] Youla, D., Jabr, H. and Bongiorno, J., 1976b. Modern wiener-hopf design of optimal controllers–part ii: The multivariable case. IEEE Transactions on Automatic Control 21(3), pp. 319–338.

[Zames, 1981] Zames, G., 1981. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. IEEE Transactions on automatic control 26(2), pp. 301–320.

[Zhou et al., 1996] Zhou, K., Doyle, J. and Glover., K., 1996. Robust and Optimal Control.

RÉSUMÉ

Les avantages des véhicules autonomes sont formidables, mais le chemin vers une vraie autonomie sera long et semé d'incertitudes. La recherche de ces dernières années s'est basée sur des systèmes multi-capteurs capables de percevoir l'environnement dans lequel le véhicule est conduit. Ces systèmes deviennent plus complexes quand on contrôle le véhicule autonome, différents systèmes de contrôle sont activés dépendant de la décision du système multi-capteurs. Chacun de ces systèmes suit des critères de performance et de stabilité lors de leur conception. Cependant, ils doivent fonctionner ensemble, garantissant une stabilité et étant capable de se charger des changements dynamiques, structuraux et environnementaux. Cette thèse explore la paramétrisation Youla-Kucera (YK) dans des systèmes dynamiques comme les voitures, en insistant sur la stabilité quand la dynamique change, ou que le trafic impose une reconfiguration du contrôleur. Concentrons-nous sur l'obtention de résultats de simulation et expérimentaux en relation avec le "Cooperative Adaptive Cruise Control" (CACC), dans le but, non pas d'utiliser, ici, pour la première fois la paramétrisation YK dans le domaine des systèmes de transport intelligents (STI), mais d'améliorer l'état de l'art en CACC aussi. Des résultats de reconfiguration stable de contrôleurs sont données quand la communication avec le véhicule précèdent n'est plus disponible, en cas de manœuvre d'entrées/sorties ou lorsqu'ils sont entourés de véhicules aux dynamiques différentes. Ceci démontrant l'adaptabilité, la stabilité et l'implémentation réelle de la paramétrisation YK comme structure générale de contrôle pour les véhicules autonomes.

MOTS CLÉS

Conduite autonome, paramétrisation Youla-Kucera, reconfiguration stable des contrôleurs, identification Boucle-Fermée et contrôle adaptatif.

ABSTRACT

Benefits of autonomous vehicles are genuinely exciting, but the route to true autonomy in transportation will likely be long and full of uncertainty. Research on the last years is on the development of multi-sensor systems able to perceive the environment in which the vehicle is driving in. These systems increase complexity when controlling an autonomous vehicle, as different control systems are activated depending on the multi-sensor decision system. Each of these systems follows performance and stability criteria for its design, but they all must work together, providing stability guarantees and being able to handle dynamics, structural and environmental changes. This thesis explores the Youla-Kucera (YK) parameterization in dynamics systems such as vehicles, with special emphasis on stability when some dynamics change or the traffic situation demands controller reconfiguration. Focus is in obtaining simulation and experimental results related to Cooperative Adaptive Cruise Control (CACC), with the aim not only of using for the very first time YK parameterization in the Intelligent Transportation Systems (ITS) domain, but improving CACC state-of-the art. Stable controller reconfiguration results are given when non-available communication link with the preceding vehicle, cut-in/out maneuvers or surrounding vehicles with different dynamics, proving adapability, stability and possible real implementation of the YK parameterization as general control framework for autonomous vehicles.

KEYWORDS

Autonomous driving, Youla-Kucera parameterization, stable controlller reconfiguration, Closed-Loop identification and adaptive Control.