

Dynamic Analysis and Control of Narrow Track Vehicles via a Multibody Modeling Approach

Quentin Docquier

Master in Mechanical Engineering

Ph.D. thesis

January 2020

Examining committee

Prof. Laurent Delannay	UCLouvain - iMMC	<i>Chairman</i>
Prof. Paul Fisette	UCLouvain - iMMC	<i>Supervisor</i>
Prof. Julien Hendrickx	UCLouvain - ICTEAM	<i>Secretary</i>
Prof. Arend Schwab	TU Delft	
Prof. Olivier Brûls	Université de Liège	

Remerciements

Cette thèse n'aurait pu être réalisée sans l'aide et le support des nombreuses personnes qui m'ont entouré durant ces quatre années. Avec ces quelques mots, je souhaite leur exprimer ma reconnaissance.

Tout d'abord, j'aimerais remercier mon promoteur, Paul Fisette de m'avoir permis de réaliser cette thèse sous sa direction, pour son aide inconditionnelle, pour les nombreuses discussions techniques et en particulier pour son soutien et sa compréhension durant les derniers mois qui ont été plus difficiles.

Je remercie également Arend Schwab qui a participé à l'encadrement de ce travail depuis les premiers mois et dont l'expertise dans le domaine de la dynamique des deux-roues a réellement été essentielle.

Merci aussi aux autres membres du jury : Julien Hendrickx, Olivier Brûls et Laurent Delannay pour leurs commentaires et remarques lors de la défense privée : ils m'ont permis d'améliorer la qualité de ce travail.

Merci à l'ensemble de mes collègues et anciens collègues MEEDiens qui ont contribué à la bonne humeur et à la joie de vivre au sein du laboratoire : je me levais tous les matins avec l'envie d'aller travailler. Un merci plus particulier à Bruno Dehez et Hervé Jeanmart d'avoir fait partie du comité d'encadrement de cette thèse ainsi qu'à Aubain Verlé pour les nombreuses discussions intéressantes sur la dynamique de ces fameux NTVs. Mais surtout, je tiens à remercier Oliver (sans i !) Lantsoght pour son soutien durant ce doctorat, que ce soit d'un point de vue purement technique (code multicorps), mais aussi moral durant la rédaction et les moments de doute.

Merci également aux personnes des services technique et administratif de l'iMMC pour leur disponibilité et leur efficacité face à mes nombreuses requêtes. Plus particulièrement à Axel Jottard, Thierry Daras ainsi qu'à Catherine Bauwens, Astrid Leduc et Nathalie Sergoigne.

Merci à Joris Gillis qui a été d'une grande aide dans la réalisation du chapitre 5. Son expertise dans le domaine du contrôle optimal a vraiment permis d'améliorer la qualité de cette thèse. Merci au Professeur Dochain pour les discussions que nous avons eues ensemble sur la commande non linéaire.

Je tiens aussi à remercier ma famille qui est un élément central de ma vie. Merci à mes parents qui m'ont toujours soutenu dans mes projets et plus particulièrement pendant mes études. Merci aussi à mon frère Augustin pour les nombreuses corrections et conseils en anglais souvent demandées en dernière minute ! Je souhaite exprimer aussi ma profonde reconnaissance à mon grand frère Nicolas qui a toujours répondu présent quand j'en avais besoin. Il a été pour moi un exemple, un modèle à suivre durant l'entièreté de mes études et de mon doctorat. Merci à lui pour sa disponibilité et ses nombreux conseils éclairés qui m'ont permis de mener ma barque durant ces 4 dernières années.

Et bien sûr, je remercie mon épouse, Barbara. Merci à toi de m'avoir soutenu au jour le jour durant ces quatre ans. Malgré les nombreux moments de doute, tu as toujours été présente et tu étais là de manière inconditionnelle. Merci pour ton courage et la foi que tu as en moi.

Contents

1	Introduction	9
1.1	Context	9
1.1.1	Urban traffic congestion, an ever-growing issue	9
1.1.2	Light vehicles, an emerging solution for urban transportation	10
1.1.3	Tilting narrow track vehicles, an engineering challenge .	12
1.2	Control of narrow track vehicles	15
1.2.1	Tilt control modes	17
1.2.2	Combination of control modes	20
1.2.3	Summary of control modes	22
1.3	Thesis objectives	24
1.4	Thesis outline	25
2	State of the art	27
2.1	Short historical overview	28
2.2	Tilting NTV modeling	29
2.2.1	Karnopp’s model	29
2.2.2	Karnopp’s model extensions	31
2.2.3	Multibody modeling	32
2.2.4	Tire/ground contact modeling	33
2.3	Tilting NTV dynamic analysis	34
2.4	Tilting control modes	37
2.4.1	DTC	37
2.4.2	STC	37
2.4.3	SDTC	38
2.4.4	LTC, FTC and torque vectoring	43

2.5	Control techniques	44
2.5.1	Linear control techniques	45
2.5.2	Nonlinear control techniques	45
2.6	Discussions and conclusions	48
3	Multibody vehicle model	51
3.1	Multibody modeling	53
3.1.1	Relative coordinates approach	53
3.1.2	Multibody system topology	54
3.1.3	Equations of motion	55
3.2	State-space representation	60
3.2.1	State-space model linearization	61
3.2.2	Non-sensitive states	62
3.3	NTV vehicle model	63
3.3.1	Reference NTV morphology	63
3.3.2	Multibody model	63
3.3.3	Suspension components and motors	66
3.3.4	Tire/ground contact model	66
3.4	Concluding remarks	69
4	Vehicle dynamic analysis	71
4.1	Steady state equilibrium	72
4.1.1	MBS equilibrium equations	72
4.1.2	Steady state map	74
4.2	Stability analysis	83
4.2.1	MBS modal analysis	83
4.2.2	Stability analysis in straight line	83
4.2.3	Stability analysis in steady state cornering	85
4.3	Transient dynamic behavior	87
4.3.1	Time simulation	87
4.3.2	Transient behavior – curve entry	87
4.4	Concluding remarks	94

5	Optimal control	97
5.1	Optimal control of multibody systems	97
5.1.1	Optimal control problem	98
5.1.2	Numerical optimal control	99
5.1.3	Formulation of the multibody dynamic equations	101
5.1.4	Implementation of OCP for multibody systems	105
5.2	NTV control through optimal control	106
5.2.1	Baseline curve entry	106
5.2.2	Constrained curve entry	110
5.2.3	Perceived acceleration minimization	112
5.3	Concluding remarks	114
6	Feedback control	117
6.1	Gain-scheduling linear quadratic control	118
6.1.1	Linear quadratic control	118
6.1.2	Gain-scheduling LQR for NTV control	121
6.1.3	LQR results	123
6.2	Feedback linearization	129
6.2.1	Feedback linearization of multibody systems	129
6.2.2	Feedback linearization for NTV control	134
6.3	Model predictive control	143
6.3.1	Linear time-varying model predictive control of MBS	144
6.3.2	Model predictive control of narrow track vehicles	146
6.4	Concluding remarks	153
6.4.1	Comparison of the three control approaches	155
7	Conclusions and perspectives	157
7.1	Content overview	157
7.2	Objectives achievement	159
7.3	Original contributions	161
7.4	Thesis perspectives	161
7.5	Broader considerations	162
7.5.1	Development of passive narrow track vehicles	163
7.5.2	Autonomous control of narrow track vehicles	164

A Literature review	167
A.1 Historical review	167
A.1.1 UC Davis (1992-1998)	167
A.1.2 TU Delft (1997-2004)	171
A.1.3 University of Minnesota (2003 - 2010)	172
A.1.4 Clever project, Bath University (2005 - 2014)	179
A.1.5 National Chiao Tung University of Taiwan (2008 - 2009)	183
A.1.6 TU Wien (2011)	184
A.1.7 Federal University of Santa Catarina (2007-2016)	185
A.1.8 Nagoya University (2012-2014)	186
A.2 Current research projects	186
A.2.1 Ecole des Mines de Nantes (2011-2019)	187
A.2.2 University of Tokyo (2015-2017)	189
A.2.3 University of Waterloo (2014 - 2018)	191
A.2.4 Pune University (2015-2017)	193
A.2.5 Resolve project (2016-2019)	194
B Multibody formalisms for solving optimal control problem	197
C Numerical optimal control	221
C.1 The different approaches for OCP solving	221
C.2 Direct methods	222
C.2.1 Numerical OCP formulation	222
C.2.2 Multiple shooting	224
C.2.3 Direct collocation	224
D Feedback linearization of multibody systems	227
D.1 Relative degree and Lie-derivatives	227
D.2 Multibody systems with relative degree 1	228
D.2.1 Nonlinear output	228
D.2.2 Linear output ($\dot{\psi}$)	228
D.3 Multibody systems with relative degree 2	229
D.3.1 Nonlinear output ($a_{per,2}$)	229
D.3.2 Linear output (ϕ, δ)	229
D.4 MIMO systems	230

Introduction

1

The goal of this introductory chapter is to help the reader to understand the scientific context and the implications of the present work. First the concept of the *tilting* narrow track vehicle is introduced as a solution to the evergrowing urban traffic congestion. The advantages of these special vehicles are presented together with previous commercial attempts and different possible morphologies. In the second part of this chapter, the control of narrow track vehicles is considered by discussing the different ways and modes to control them. Eventually, the scientific objectives of this work are presented, followed by the thesis outline.

1.1 Context

1.1.1 Urban traffic congestion, an ever-growing issue

Nowadays, traffic congestion is a growing issue all over the world. Apart from being the main reason for smog in cities, it significantly contributes to the important cars operating times around cities, which is one of the main causes of air contamination and greenhouse gas emission within urban areas. Furthermore, traffic congestion affects the general productivity of our society as the time spent in traffic jams cannot be used by the commuters to work. Until now, the solution found is to expand the existing road infrastructure in order to cope with the traffic growth and reduce as far as possible the time wasted in traffic jams. Unfortunately, urban areas are reaching or have already reached their limit in terms of road infrastructure extensions. On the other hand, conventional vehicles¹ are underutilized with an average number of 1.67 occupants in 2017 for light vehicles in U.S.A. [1], unchanged from 2009 (Fig. 1.1).

These observations have led to an increasing interest in lighter one-seater and two-seater vehicles over the past two decades. These vehicles may not replace the conventional cars that are meant to live on as the number one means of transportation for families. Nevertheless, light vehicles could be a credible alternative for daily commutes as they could represent a time and a financial saving (i.e., reduced cost of fuel, maintenance and purchase price) for the

¹Conventional vehicles stand for cars, vans, sport utilities, pickups and motorcycles.

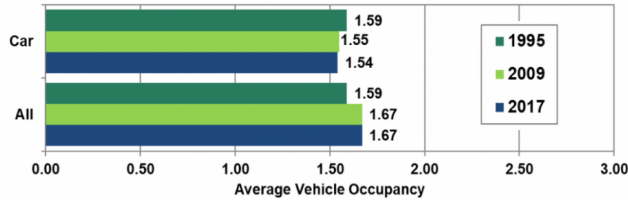


Figure 1.1: Occupancy rate for cars and conventional vehicles in U.S.A. [1].

commuter while reducing its carbon footprint. Regardless of the propulsion technology, commuting with lighter vehicles could also contribute to a reduction in greenhouse gas emission as the overall vehicle mass moved during daily travel could be drastically reduced. In addition, lighter vehicles would also monopolize fewer parking spaces within urban areas. Together with carpooling solution, it is believed that light vehicles could contribute to the solving of the urban congestion issues within and around cities.

1.1.2 Light vehicles, an emerging solution for urban transportation

When it comes to light vehicles, the range of design possibilities is huge. Generally, these vehicles have a reduced track which logically leads to a larger height-to-track ratio as height is needed to ensure a good visibility when driving among regular four-wheeled vehicles (cars, trucks, etc.).

Regarding the number of wheels, different morphologies are possible. An extensive *in-depth* classification can be found in [2].

- **4-wheeler.** These vehicles often possess the same structure as conventional cars with a steering assembly connected to the front wheels and the powertrain located on the rear axle. There have been many attempts to develop such vehicle in which the narrow track concept is generally coupled with a fully electric propulsion system (Fig. 1.2.a). However, few of them have been successfully commercialized. One can cite the Renault Twizy, whose track is still relatively large (about 1.1 meters), which has encountered a small success in Europe with young drivers as a replacement for scooters (Fig. 1.2.b).

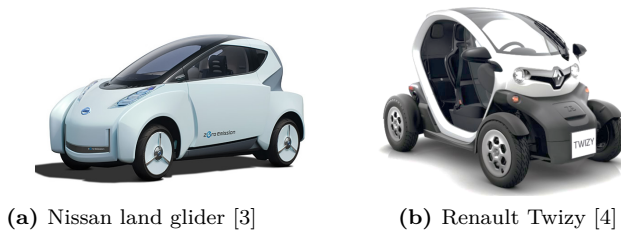


Figure 1.2: Light 4-wheeler vehicles.

- **3-wheeler.** These vehicles always have two wheels positioned side-to-side in order to ensure stability at a standstill and during cornering. Two arrangements are possible for a 3-wheeler: the *delta* with two wheels at the rear and one in the front and the *tadpole* with the opposite configuration (Fig. 1.3). Both configurations have their advantages and disadvantages [2].

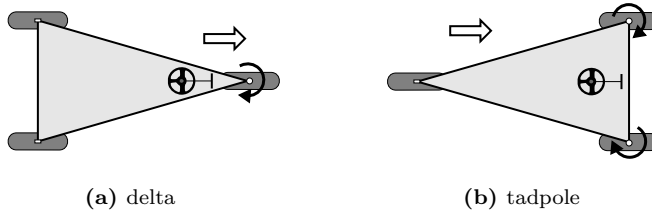


Figure 1.3: Different wheel configurations for light 3-wheeler.

Again, a plethora of prototypes have been developed but they had no significant commercial success. One can still cite the Toyota i-road and the Carver vehicle as narrow 3-wheeler, that have reached a certain level of commercialization (Fig. 1.4).



(a) Toyota i-Road [5]



(b) Carver One [6]

Figure 1.4: Light 3-wheeler prototype vehicles.

The main issue with narrow track vehicles is the risk of rollover which strongly limits their range of utilization. As a matter of fact, when reducing the vehicle track, the resultant of the centrifugal and gravitational effects while cornering might not always intersect the wheel-ground contact polygon². In this case, the vehicle might rollover around the outer wheel(s) (Fig. 1.5). The centrifugal effect³ in steady state maneuver is roughly proportional to the centripetal acceleration which depends on the squared vehicle speed V and the local curvature $1/R$ for the envisaged maneuver (i.e., R designing the curving radius).

²The polygon defined by the vertices joining the different contact points at the wheel/ground interface. For a three-wheeler, the polygon is a triangle.

³Induced by the road contact centripetal forces.

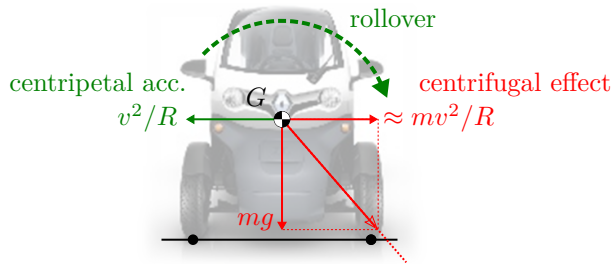


Figure 1.5: Rollover for narrow vehicles can occur during high lateral acceleration maneuver.

Even if the rollover limit is not reached, there is still a risk of failure due to the wheel load transfer. In fact, when the vehicle undergoes a high lateral acceleration, the weight is transferred from the inner wheel to the outer wheel leading to a decrease in the overall lateral forces generated at the tire/ground contact points as the tires have nonlinear properties with respect to their load. This decrease could lead the vehicle to skid and get out of control.

These two failure events can be avoided by leaning the vehicle toward the inside of the turn, in a similar way as a motorcycle (Fig. 1.6). Leaning ensures that the resultant is pointing within the wheel-ground contact polygon but also reduces the wheel load transfer as the center of mass is shifted toward the inner wheel. These observations have rapidly led to the development and study of *tilting* narrow track vehicles.

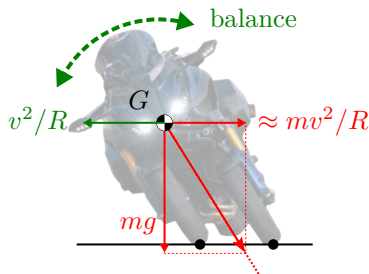


Figure 1.6: Tilting narrow track vehicles can be leaned into a curve allowing them to withstand higher lateral acceleration than non-tilting vehicle for the same track.

1.1.3 Tilting narrow track vehicles, an engineering challenge

The large height-to-track ratio, indispensable to ensure a good visibility when driving among regular four-wheeled vehicles, implies to lean these NTV toward the inside of a turn to prevent them from rollover during high-speed cornering. NTV morphologies can differ, as shown above, but generally feature more than

2 wheels (3 or 4) and a suspension assembly that allows the chassis to fully tilt. If the latter is free to rotate with respect to the suspension assembly, the NTV can be assimilated to an inverted pendulum in the transverse plane (Fig. 1.7). Similarly to single-track vehicles (STV) like the motorcycle and the bicycle, the driver can balance the vehicle by orienting the steerable wheel(s) which generates lateral forces and subsequently causes the roll motion. To enter a curve, a STV needs to be first leaned into the turn via a countersteering action by the rider. Once the roll motion is initiated, the rider steers the wheels back in the direction of the curve to produce the necessary amount of centripetal lateral forces.

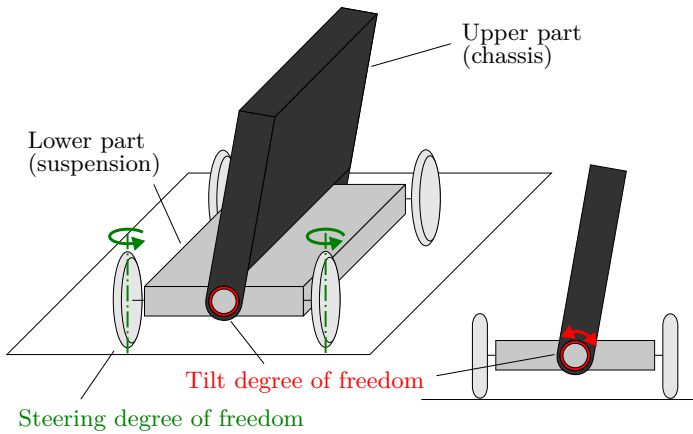


Figure 1.7: Inverted-pendulum representation of the tilt degree of freedom for a generic NTV.

The first attempt of tilting NTV dates from the 50s with the *lean machine* prototype from General Motors (Fig. 1.8). This special 3-wheeler features a non-tilting rear engine pod and a cabin that tilts. It required the driver to control the tilt with a foot pedal while simultaneously driving via the steering wheel. Although the test drivers were able to manage both tasks after a certain learning time, they came to the conclusion that it was unrealistic to propose a vehicle for the general public that required such amount of skill learning to be operated.



Figure 1.8: The lean machine developed by General Motors in 1982 [7].

Conceptually, tilting NTVs are at the limit between the world of conventional cars whose roll motion is limited by spring-type elements together with a large track, and that of the two-wheeler whose single track intrinsically implies a lean motion during cornering (Fig. 1.9). If the mobility offer has drastically expanded during the past two decades with new three-wheeled scooters and electric bicycles, it is important to state that tilting NTVs are generally aimed to be driven like conventional cars are, i.e., through a *direct* steering action on the steering wheel and without having to control the vehicle roll motion (Fig. 1.9).

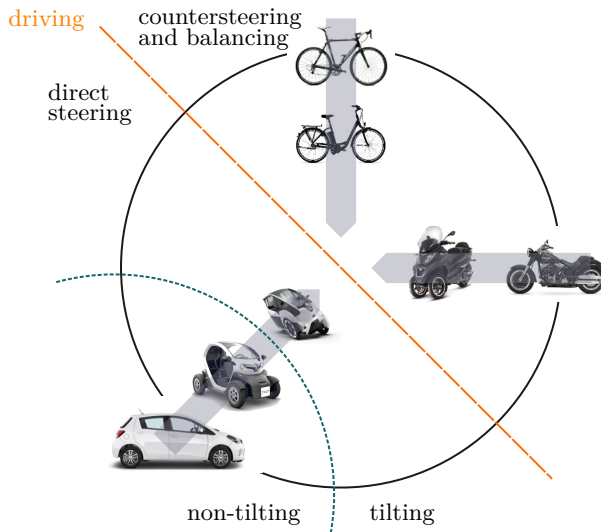


Figure 1.9: Conceptual vehicle map of the personal mobility range.

3-wheeled scooter VS. car-like NTV Tilting NTVs are obviously lighter than conventional cars but still feature a higher mass and inertia than conventional motorcycles which makes their roll motion harder to control through a human steering action. Nowadays, many scooters feature more than two wheels but are still driven as a STV because the chassis can still tilt freely. The main advantage of having more than two wheels for these scooters is the stability at a standstill which is often ensured by an electrical brake restricting the tilting motion. In addition, it may enhance safety in case of a bumpy or icy road as it offers an additional contact patch. Besides existing models such as the Piaggio MP3 or the Peugeot Metropolis, research programs still aim at the improvement of such enhanced scooters [8, 9]. Current research projects focus on the development of tilting 4-wheeler for which the gain in dynamic stability compared to a conventional STV with two wheels is not readily obvious. Unlike the case of a motorcycle and three-wheeled scooter, for which the rider can shift his weight to lean the motorcycle into the curve, the driver motion within a NTV cabin is relatively low and can thus be neglected.

NTV prototypes and commercial attempts. There have been many attempts to develop tilting NTV during the past three decades [2]. Among the different prototyped vehicles, one can cite the Toyota i-Road, the Carver and the Clever (Fig. 1.10).



Figure 1.10: Commercial attempts and prototypes of tilting NTVs (more details can be found in [2]).

Visually, these vehicles differ from one to another by their morphology. However, their main difference lies in the way in which the NTV tilt control is handled. The latter can be devolved to the driver who must manage both heading and balance at the same time or to the vehicle itself with the help of an active tilt control. In the first case, the driver has to perform a countersteering like for a conventional STV, while in the second case, NTV actively enables direct steering to the driver. The different tilt control modes are discussed and presented in the following section.

In this work, the studied narrow track vehicles will always be *fully* tilting in opposition to their non-tilting counterparts. Hence, the tilting nature will generally be omitted and the NTV abbreviation will denote a *tilting* narrow track vehicle.

1.2 Control of narrow track vehicles

As already discussed, there are two control tasks that need to be performed when driving a tilting vehicle :

- The control of the vehicle *heading* (or steering) in the horizontal plane that mainly consists in ensuring that the vehicle follows the desired trajectory (Fig. 1.11.a).
- The control of the vehicle *balancing* (or tilting) in the vehicle transverse plane for which rollover and skidding events have to be avoided (Fig. 1.11.b).

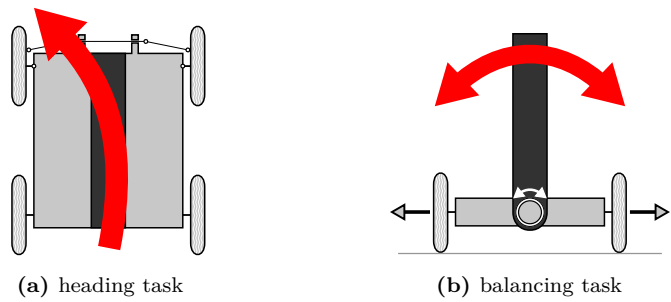


Figure 1.11: The two control tasks for the driving of a tilting vehicle.

Whereas for a conventional car, the vehicle heading is directly and simply associated to the steering wheel, it is not true for NTVs. NTV control can strongly differ depending on who or what realizes the two above-mentioned tasks, i.e., the driver or the vehicle itself. There are three types of control to handle this issue:

- *Driver control.* In this case, the driver ensures both the heading and the tilting control for the vehicle. This is what a rider already does when controlling both the heading and tilting of his motorcycle through the steering wheel and his body motion.
- *Cooperative control.* The driver and the vehicle interact and work together to achieve both control tasks. Generally, the driver performs the high level path-following tasks whereas the vehicle tilts itself through a steering and/or tilting action.
- *Autonomous control.* The vehicle controls both the heading and tilting simultaneously. The driver is completely passive and not a single driving skill is required from them.

Table 1.1 summarizes the different control *types* in terms of the two control tasks and required skills.

	Driver	Cooperative	Autonomous
			
Heading	Driver	Driver	Vehicle
Balancing	Driver	Vehicle	Vehicle
Skills required	Motorcycling	Driving	None

Table 1.1: Different types for the control of tilting NTV.

The driver control is often referred as *passive* control because the vehicle does not take any decision and behaves passively with respect to the driver input. For this type of control, he needs to perform a countersteering action to drive and balance the vehicle simultaneously while cornering. Conversely, the cooperative control is called *active* control as in that case, the vehicle might act on the tilting and/or the steering degree of freedom in order to assist the driver. In that case, the driver only needs driving skills.

Countersteering While cornering, a tilting vehicle must be leaned into the direction of the turn in order to counterbalance the rollover moment due to the centrifugal effect (Fig. 1.5). To do so, the wheels need to be first steered in the opposite direction. This opposite steering temporarily generates lateral forces directed outward which move the contact patch away from the center of mass and induce the desired fall within the intended curve (Fig. 1.12.b). This maneuver has to be immediately followed by turning the wheels in the direction of the induced fall in order to keep the vehicle in balance and produce the necessary amount of inward lateral forces to keep the vehicle in the desired turn (Fig. 1.12.c). The initial motion of the steering wheel in the direction opposite to the desired turn is referred as *countersteering*. This technique is well-known to cyclists whose vehicle inertia and mass force them to perform such maneuver, conversely to bicyclists who often rely on their upper body motion to initiate the fall.

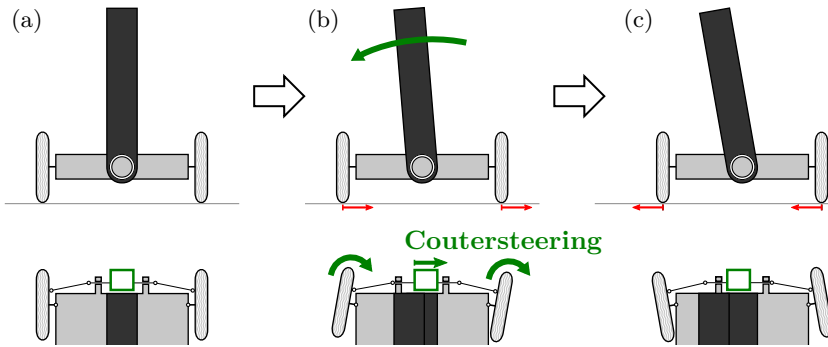


Figure 1.12: Countersteering explained based on the inverted-pendulum representation of a NTV.

1.2.1 Tilt control modes

The cooperative control can be carried through various methods called tilt control modes. The additional degree of freedom between the chassis and the suspension assembly (Fig. 1.7) is not necessarily free but can also be actuated and controlled. This leads to several possibilities for tilting control mode which are divided into two categories: the *parallel* tilt control and the *serial* tilt control.

Parallel tilt control

In this first approach, an actuator assists the driver by *actively* controlling the vehicle tilt. In that case, the vehicle and the driver work in *parallel*, the driver taking care of the vehicle steering and heading while the vehicle tilts itself compliantly (Fig. 1.13). This is often referred as direct tilt control (DTC) and has already been the subject of several studies in the literature [11, 12, 13].

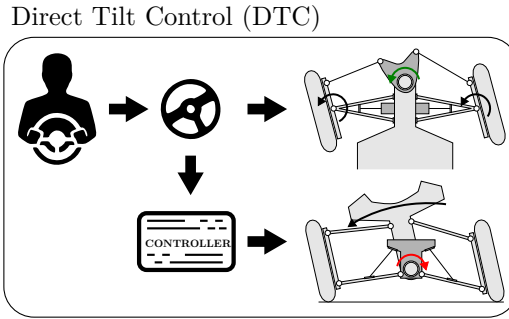


Figure 1.13: The direct tilt control (DTC) consists in a *parallel* command of both the driver and the vehicle. An active controller is necessary in this case.

DTC simplifies the vehicle control by acting as a direct response to the driver steering wheel input. As a consequence, when the NTV is steered aggressively, sudden steering inputs require large roll accelerations, and hence large torques, which may cause the inner wheel(s) of the base to leave the ground. In other words, although DTC ensures stability in steady state cornering, it may cause rollover during transient maneuvers. In addition, the large roll acceleration leads to a large perceived *transverse* acceleration at the location of the driver's head which can be uncomfortable.

Serial tilt controls

For these modes, the driver and the vehicle act somewhat in *series* in the sense that the vehicle dynamically reacts to a driver input instead of being controlled accordingly. Two actuations can be considered for the *serial* mode.

For the first mode, the vehicle is steered through the control of the steering assembly while the tilt degree of freedom is let free. In that case, the vehicle tilting is a consequence of the driver steering action and the chassis is free to roll (Fig. 1.14.a). This control mode is what any cyclist experiences where through the handlebar orientation he can balance his bicycle in a free-to-roll (FTR) manner. For the second mode, the driver induces the vehicle tilting motion mechanically (e.g., through a steering wheel input). Imposing the vehicle tilting causes the latter to dynamically steer thanks to the self-orientation capabilities

of the steerable wheel(s) whose adequate geometry⁴ makes them free to *caster* (FTC). In that case, the vehicle heading is a consequence of the driver tilting action (Fig. 1.14.b).

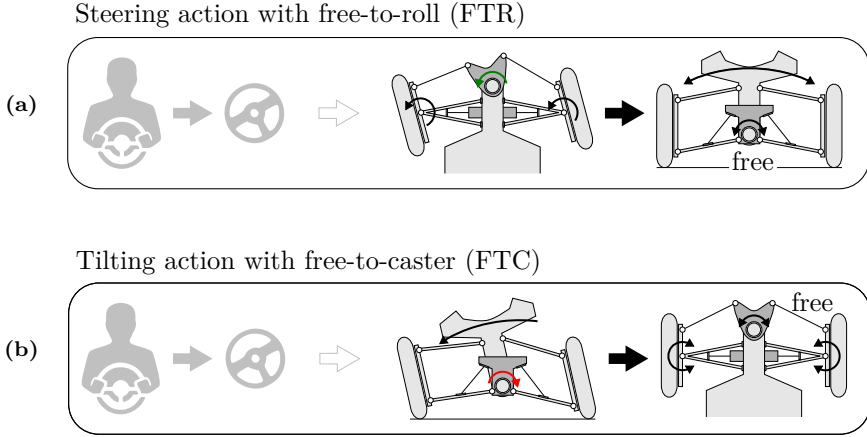


Figure 1.14: The driver and the vehicle work in series as for the free-to-roll (FTR) case corresponding to the conventional two-wheeler control. The tilt is controlled through a direct action while the wheels are free-to-caster (FTC).

The *serial* modes have the advantage of not requiring a complex *active* controller as the vehicle can react on its own to the driver steering wheel input. The driver can control both the vehicle tilting and heading through a unique input as he would do on a conventional single-track vehicle with the handlebar.

Nevertheless, *active* tilt control is still possible for the serial modes. This can be achieved by simply adding a controller between the steering wheel and the actuated degree of freedom (i.e., steering or tilting). Based on a given steering wheel input provided, the controller acts on the d.o.f. to achieve the wanted behavior. This is particularly important for the steering action that normally needs a countersteering action for balancing and driving the vehicle in the desired direction. When such active control is implemented, the FTC mode will be referred as steering tilt control (STC). In a similar way, the FTR mode can also be driven by a controller, rather than via a pure mechanical transmission, based on the driver's input. This specific control mode is called in this work LTC for leaning tilt control (Table 1.2).

⁴In order for the wheel to realign automatically, the steering axis needs to cross the ground plane in front of the tire/ground contact point. This configuration, referred as positive trail, is used for instance on shopping carts to keep them steerable and stable when pushed [14].

	passive	active
parallel	/	DTC
serial	FTR	STC
	FTC	LTC

Table 1.2: Summary of the elementary tilt control modes.

In contrast with DTC and LTC, STC does not apply a longitudinal moment about the additional tilt d.o.f. but instead uses side forces at the wheel/ground contact points. Based on the driver’s input, STC steers the base to lean the chassis over. In this way, large roll accelerations are generated by STC without producing additional wheel load transfer. In addition, the instantaneous roll axis is closer to the driver’s head which reduces the perceived acceleration and supposedly makes the ride more comfortable. The main deficiency of STC is its inherent low speed instability and the necessity to countersteer during transient phases. This necessary initial reverse steering implies a deviation from the desired trajectory, that makes the motorcycle and bicycle learning complicated.

1.2.2 Combination of control modes

Not all modes have the same behavior with respect to vehicle speed. FTR and FTC are actuation modes that are very difficult to control at low speed (e.g., it is impossible to stabilize a bicycle at a standstill by only using the steering input) while the direct tilt control (DTC) is *inadequate at higher speed due to the important load transfer*. This has led to the development of dual-mode *active* methods combining both the parallel and the serial modes. Based on the steering wheel input, the vehicle tries to control at best the vehicle heading and tilting for the overall vehicle speed range.

Switching tilt control

The first way to combine the parallel and serial control modes is simply by switching from one to another depending on the vehicle speed (Fig. 1.15). The direct tilt control (DTC) is used at low speed when the steering tilt control (STC) is known to be unstable. Above a given speed, the STC is used as it offers better performances in terms of energy consumption and stability (i.e., limited wheel load transfer). This switching approach is denoted here SDTC-S.

Switching Steering and Direct Tilt Control (SDTC-S)

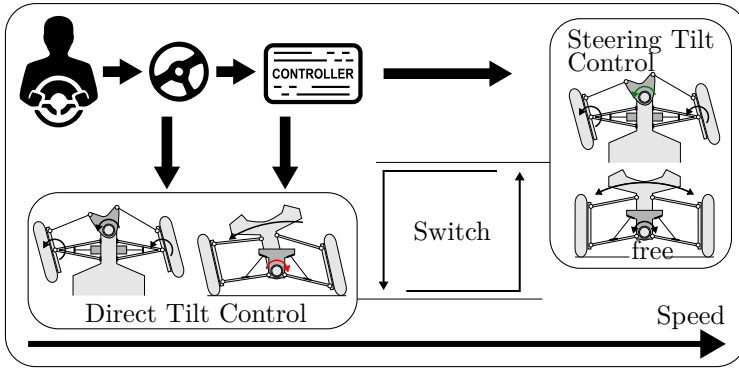


Figure 1.15: The combined SDTC-S is based on the active combination of DTC and STC at different speeds. The transition is operated at a given forward speed.

Hybrid tilt control

The second way to combine the parallel and serial control modes is the so-called hybrid tilt control. In this approach, both the steer and tilt d.o.f. can be actuated simultaneously for the whole speed range (Fig. 1.16). The tilt d.o.f. is actuated in order to follow a given lean angle (consistent with the desired vehicle steady state cornering). Generally, the steering d.o.f. is linked to the steering wheel but its angle is modulated by an additional actuator that provides countersteering when needed (i.e., for very dynamic maneuver). Such hybrid control is denoted in this work SDTC-H.

Hybrid Steering and Direct Tilt Control (SDTC-H)

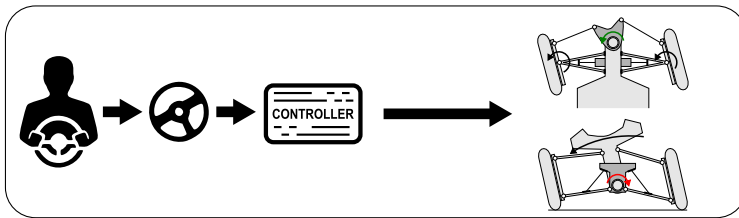


Figure 1.16: The combined SDTC-H is based on the active combination of DTC and STC for any speeds.

Serial control combined with tilt locking

As mentioned above, FTR and STC modes are known to be unstable at very low speed. In order to solve this issue, the tilt rotation between the chassis and the suspension can be locked in an upright position. In this configuration, the steering wheel is reconnected to the steering assembly: the NTV is no more tilting and is driven like a conventional car (Fig. 1.17). Let denote LTD this specific mode for locked-tilt drive.

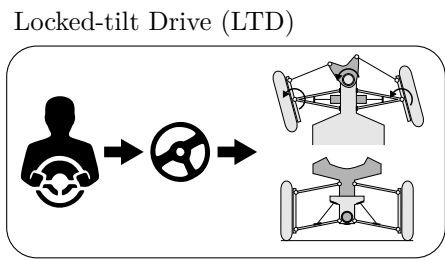


Figure 1.17: The locked-tilt mode can be used for low speed if combined with a serial control mode at high speed.

1.2.3 Summary of control modes

Depending on the actuation of the steer and tilt degree of freedom, many control modes can be envisaged. Table 1.3 summarizes the different modes and their possible combinations with their advantages and disadvantages. From this table, one can see that some modes are more relevant and more viable than others. LTD-STC seems to offer good performances as well as the combined modes SDTC-S and SDTC-H. Moreover, the LTD-FTC does not need any active control from the vehicle while enabling a direct steering for the driver. However, the main issue with this mode is the fact that from FTC to LTD, the steering wheel needs to be mechanically disconnected from the tilt d.o.f. and to be clutched to the steering d.o.f. However, it is believed that such function could still be achieved through a well-thought-out transmission allowing for the vehicle to remain completely passive. Ideally, the transition from one mode to another should always be seamless to the driver.

Mode	Acronym	Actuated d.o.f.	Type	Advantages	Disadvantages
Free-to-roll	FTR	steering	passive	no need for control	balancing skills and countersteering necessarily dangerous given the inertia/mass of NTV unstable at low speed
Free-to-caster	FTC	tilting	passive	direct steering good behavior in case of adhesion loss	important wheel load transfer (even in steady state) important actuation (not backdrivable is necessary) possible wheel lifting undriveable at low speed
Steering tilt control	STC	steering	active	direct steering no additional wheel load transfer low actuation	unstable at low speed and standstill deviation from the path (outside the curve) bad behavior in case of sudden slippery road
Leaning tilt control	LTC	tilting	active	direct steering good behavior in case of adhesion loss	same as FTC absence of steering wheel feedback
Direct tilt control	DTC	steering tilting	active	direct steering stable at a standstill/low speed mechanical link between sw and steering assembly	important wheel load transfer (even in steady state) important actuation (not backdrivable is necessary) possible wheel lifting
Locked-tilt drive	LTD	steering (locked tilt d.o.f.)	passive	stable	limited to low speed and lateral acceleration maneuver
enhanced FTC	LTD/FTC	steering tilting	passive	direct steering good behavior in case of adhesion loss stable at all speeds	important wheel load transfer (even in steady state) important actuation (no backdrivable is necessary) possible wheel lifting
enhanced STC	LTD/STC	steering	active passive	direct steering no additional wheel load transfer low actuation stable at all speeds	deviation from the path (outside the curve) bad behavior in case of sudden slippery road
/	LTD/FTC				unviable because of driving style switching (direct to counter steer in a abrupt way)
Switching tilt control	SDTC-S	steering tilting	active	direct steering stable at all speeds	2 actuators are needed not easy to implement (switch)
Hybrid tilt control	SDTC-H	steering tilting	active	direct steering good in dangerous maneuver case stable at all speeds	2 actuators are needed not easy to implement (switch)

Table 1.3: Summary of the tilt control modes and their possible combinations.

1.3 Thesis objectives

During the past three decades, the active control of narrow track vehicles has been the subject of many research projects as it will be presented in the state of the art (Chapter 2). Generally, these investigations on NTV control are carried out within the framework of an integrated project which aims at designing, manufacturing and testing a NTV prototype. In this respect, the control part of the project often focuses on the realization of a *cooperative* tilt controller and on its validation via experimental tests. In the past, different tilt control modes and control techniques have been studied but with different approaches and often with simplified modeling techniques.

The main goal of this thesis is to compare the different tilting modes in a systematic way and via an accurate modeling of the NTV tridimensional dynamics. To this extent, two main research focuses have been chosen:

- **Comparison of the 2 *passive* tilting modes (FTR and FTC).**
The two serial tilting modes are dynamically very different which should logically lead to contrasted behavior during transient curve entry and steady state cornering. The objective is to realize a proper dynamic comparison of both modes and determine the advantages and disadvantages of one mode compared to the other. This comparison is relevant as it could reveal whether a tilting action (FTC) could compete with the more traditional steering action (FTR) to control the NTV tilting motion. Contrarily to the more traditional free-to-roll (FTR), the FTC and its dynamics have never been thoroughly studied in the past. Thus, another objective is also to understand how and why the forced tilting motion in one direction causes the vehicle to automatically steer in the same direction.
- **Investigation of nonlinear active control techniques for NTVs.**
Previous works have focused on the control of NTV as it will be presented in Chapter 2. The main issue is that they are either considering simplified controllers for elaborated NTV dynamic models, or advanced nonlinear techniques with too simplified dynamic models. In this work, we aim at studying the nonlinear control of NTVs while considering their fully tridimensional dynamics. To this extent, three nonlinear techniques are considered : gain-scheduling LQR, feedback linearization and model predictive control. For each control techniques, the feasibility and performances of the four tilting modes, i.e., STC, LTC, SDTC and DTC are investigated and discussed. Tracking performance, stability and actuation costs are the main focus points envisaged for the comparison.

More specifically, we want to compare the different tilting modes via the use of advanced modeling tools in order to capture and account for the full tridimensional motion of a narrow track vehicle. The vehicle morphology and geometry used in this work correspond to a narrow three-wheeler, with a reverse-trike configuration, designed at the Université catholique de Louvain [15][2]. Although the vehicle was not built, its complete design was realized and an accurate multibody model was developed. Regarding the driver's behavior, it

was decided from the beginning to neglect it to facilitate the comparison of the modes and control methods. Implementing a basic driver model (e.g., PID controller) without knowledge of the controlled vehicle does not make sense and would only complexify the analysis. Therefore, the driver input will only be translated as a reference signal (i.e., lean angle or yaw rate) that the controller has to track.

1.4 Thesis outline

The present work has been divided into 7 chapters.

After this introduction, a thorough state of the art in the field of NTV control is presented in Chapter 2. To this extent, a condensed summary of the preview NTV control investigations is realized, structured around several entry points, such as the vehicle modeling, the NTV dynamic analysis or the envisaged control techniques. In addition, it provides the reader with some key features of NTV control that will be reused later on in this work. At the end of Chapter 2, the two focuses of this thesis are clearly identified as original contributions to the existing NTV scientific literature.

Afterwards, the reference vehicle model used in this work is presented in Chapter 3. First, the multibody (MBS) formalism is introduced together with some motivations for such a framework. The generic equations of motion for a MBS are written in a state-space form and subsequently linearized. Finally, the reference NTV morphology and its MBS model are presented into details (i.e., front assembly modeling, tire/ground contact models).

Chapter 4 focuses on the first objective of this thesis, i.e., the comparison of the passive FTR and FTC modes and their dynamics. More specifically, the cornering ability for the two envisaged modes is analyzed for a series of given steady state cornering equilibrium. The influence of the curve radius on the vehicle stability is observed and discussed for both modes. Finally, the transient behavior during a curve entry is simulated and compared for the two modes. The fact that no control is deliberately considered in this chapter restricts the maneuver to a simple step response.

In Chapter 5, optimal control is considered for the tilt control of narrow track vehicles (NTV). More specifically, the optimal framework is used to investigate the combination of the steering and tilting torques (SDTC). The chapter is divided into two main sections. In the first section, the optimal control formulation for multibody systems is presented. In the second section, the combined use of steering and tilting actuation is studied for NTV via the simulation of a curve entry at different speeds. In addition, Chapter 5 lays the basis for the model predictive control presented in Chapter 6.

Chapter 6 focuses on the feedback control of narrow track vehicles via nonlinear control techniques which is the second main objective of this thesis. First, gain-scheduling LQR is discussed, then feedback linearization is considered and finally model predictive control is investigated. For each control technique, the

theoretical bases are presented and applied to the specific case of multibody systems. Then, NTV control is investigated and results are discussed for the different control modes (i.e., STC, LTC, SDTC and DTC). At the end of the chapter, general conclusions are drawn on the methods's applicability and performances for the four envisaged control modes.

Finally, Chapter 7 presents the main conclusions of this thesis. Firstly, the content of the thesis is briefly summarized and the original contributions of this work are highlighted. Secondly, broader considerations on the future of NTVs and their control are presented together with guidelines for future works on the subject.

In this thesis, we have deliberately chosen to disseminate the theoretical bases for the dynamic analysis and the advanced control within the different chapters instead of gathering all in one theoretical chapter. To this extent, Chapter 3 accounts for the main theoretical basis while Chapters 4, 5 and 6 alternate between theoretical developments and simulation results. This was done in an attempt to offer a fluent reading of the thesis with results emerging early on in Chapter 4.

State of the art

2

This chapter aims at presenting the state of the art on narrow track vehicle control.

There have been investigations on NTV control since the early 1990s and ongoing research is still focusing on the development of viable control strategies that would enable the establishment of safe NTVs on the mobility market. For the sake of completeness, a thorough literature review has been realized gathering all the scientific research carried on the NTV control during the past three decades (Appendix A). However, for the sake of brevity, this chapter does not feature the whole literature review but rather presents a condensed summary structured around several entry points, such as the vehicle modeling or the implemented control method. For a more detailed explanation of the previous works on NTV control, the reader can always refer to the full literature review (Appendix A) or the many references cited in it. In addition, this chapter aims at providing the reader with some key features of NTV control that will be reused later on in this work. Some concepts are then more thoroughly presented than others. Finally, it is important to recall that this work focuses on the control of tilting NTVs rather than on their mechanical design and practical implementation.

First a short historical review is presented enlightening the major contributors to the existing literature. Then, the modeling approaches used by the different research teams are discussed. Afterwards, the dynamic analysis realized upstream the control design is reviewed. Then, the different control modes that have been investigated throughout the last three decades are discussed. Subsequently, the different control techniques used for NTV control are considered. Finally, concluding remarks are drawn on the overall state of the art.

In this state of the art, different quantities are defined and used by the authors. The roll and lean angles, denoted ϕ , φ or θ , are the same physical quantities and correspond to the angle that the chassis median plane makes with the perpendicular to the road. The tilt angle, often denoted θ as well, is the angle between the tilting chassis and the vehicle base. For simplified models in which the vehicle base is always aligned with the horizontal plane, the roll (i.e., lean) angle is equivalent to the tilt angle.

2.1 Short historical overview

The investigation on narrow track vehicle control was initiated by Karnopp in the early 90s at UC Davis [16, 17, 18]. Given the vehicle size and inertia, they argue that an active control of the tilt motion is necessary to offer a car-like driving experience, i.e., a direct steering through the active control presented in Section 1.2.1. Based on a simplified modeling approach, he studied the lateral and roll dynamics of NTVs and introduced the concept of DTC and STC as defined in the introduction (Section 1.2). Early on, Karnopp and his team proposed to combine both modes depending on the vehicle advance speed [11]. At the start of the 2000s, Prof. Rajamani from the University of Minnesota started to study the NTV dynamics and its control [19]. Together with his team, he improved Karnopp's model and developed new methods based on DTC, STC and their combination denoted as SDTC [20]. Rajamani and his researchers considered different definitions of the reference tilt angle together with complex nonlinear control techniques [12, 21]. In addition, they conducted experimental tests with both a reduced-size unmanned and a real size prototype [20, 22]. At the same time, the Clever project started at the University of Bath. Focusing on the design aspects, they developed a prototype featuring both STC and DTC technique [23, 24]. Throughout a numerous amount of driving tests, they studied the implementation of a SDTC system and its limitations in terms of maximum lateral acceleration and tilting torque [25, 13]. Later on, in 2011, another NTV was developed by Prof. Plöchl and its team from the Technical University of Vienna. A thorough dynamic analysis of the vehicle was performed and an electronic controller was developed enabling to stabilize the vehicle roll motion [26, 27]. From 2011 until recently, investigations have been conducted at the Ecole des Mines de Nantes under the supervision of Prof. Chevrel and Claveau. For purposes of controlling commercial NTV Smera developed by Lumineo [28], they have investigated DTC, STC and SDTC control via linear and nonlinear advanced controllers [29, 30]. More recently, a new NTV developed by Equos¹ based on the free-to-caster (FTC) principle was studied by Jeffrey Tan under the supervision of Prof. Suda from Tokyo University. The steering stability is clearly the main subject of the investigations for which sensitivity analyses on dynamic parameters (e.g., the trail) are performed [32, 33]. At the same time, control of integrated in-wheel systems is the subject of many scientific investigations conducted at the Waterloo University [34]. Their work focuses on assistive tilting throughout tilting and steering but also torque vectoring; the range of tilting is, however, limited and corresponds to roll stability improvements rather than to a full tilting control [35, 36]. Eventually, the European project Resolve concentrates on the tilting motion and control of 4-wheeler comparable to 2 or 3-wheeled scooters. Within the consortium, the University of Pisa has mainly focused on the dynamic modeling and analysis of the vehicle [9] while the University of Warwick has investigated the NTV control throughout high level nonlinear control [37, 38].

¹Equos is a research and development company specialized in automotive, collaborating with Toyota [31].

Alongside the projects cited above, other more isolated scientific studies have been carried out during the past three decades by the following researchers and universities: Prof. Pauwelussen (TU Delft), Dr. Chiou (Taiwan), Prof. Roqueiro (Brazil), Okayama-Nagoya University (Japan) or Pune University (India). More on these projects can be found in the full literature review (Appendix A). The research projects on NTV control are summarized in Figure 2.1.

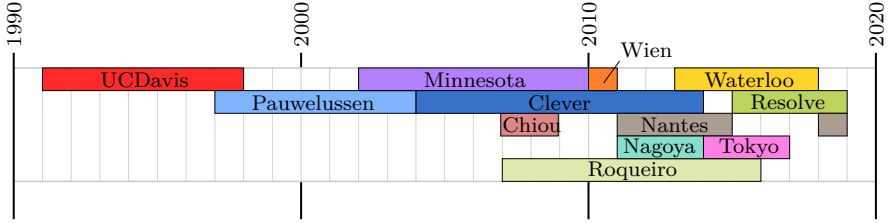


Figure 2.1: Time line of the research projects on NTV control during the past three decades (Appendix A).

From this short historical overview, it can be noticed that NTV control has often been investigated as part of projects aiming at developing a NTV. Hence, there is very little building up on previous scientific works as each project features its own specificities (i.e., vehicle morphology, control strategy, ...). Nevertheless, some tendencies can be identified. SDTC and DTC are often considered as the only viable solutions to offer a car-like driving experience for a tilting NTV. Both the modeling and control techniques have evolved since the 90s with research teams focusing either on a precise dynamic analysis via advanced modeling tools or high-level control methods based on simplified models. In the following sections, it is proposed to summarize the contributions of each project with respect to the different features of NTV control: the model, the dynamic analysis, the control mode and the control techniques.

2.2 Tilting NTV modeling

2.2.1 Karnopp's model

The first NTV model is introduced by Karnopp in the early 1990s. It consisted of a single-track model² coupled with an inverted pendulum model for the roll dynamics of the vehicle (Fig. 2.2). Assuming no tire deformation and no slip angles for both axles, Karnopp's model expresses the vehicle yaw rate³, $\dot{\psi}$ and lateral local velocity⁴, U , as a function of the longitudinal velocity, V and the

²A single-track model or bicycle model is a simplified representation of a ground vehicle with the front and rear axles replaced by two equivalent wheels [14].

³The vehicle angular velocity component along the vertical direction.

⁴The vehicle velocity component along the transverse direction of the chassis.

front axle steering angle, δ . Finally, this leads to a simple equation for the roll dynamics which can be further linearized, resulting in equation (2.2) where m and I respectively denote the mass and the yaw inertia of the vehicle.

$$(I + mh^2)\ddot{\phi} - mgh\phi = -mh(\dot{V} + \dot{\psi}U) \quad (2.1)$$

$$= -mh\left(\frac{\dot{U}}{l}b\delta + \frac{U}{l}b\dot{\delta} + \frac{U^2}{l}\delta\right) \quad (2.2)$$

The front steering angle δ is considered as an input which can either be associated to the driver or actively controlled (e.g., for STC). The second input is the tilting torque M_t directly applied on the tilting d.o.f. and that allows for direct tilt control (DTC). Generally, the NTV longitudinal dynamics is neglected and the vehicle longitudinal speed V is considered as a parameter rather than a state.

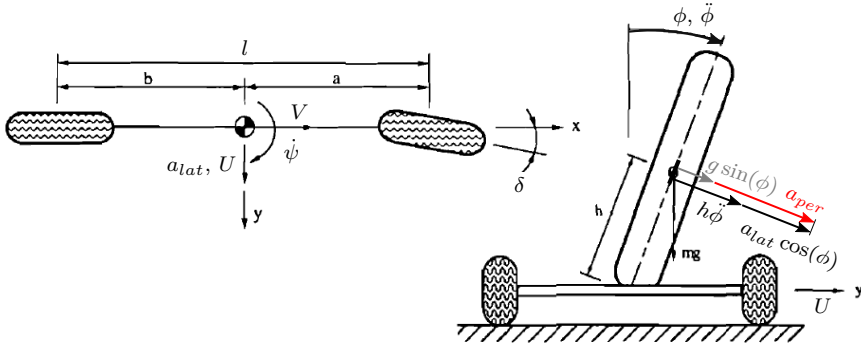


Figure 2.2: Single-track model with the inverted pendulum model (adapted from [11]).

The model allows for a simple expression of the steady state lean angle as a function of the vehicle forward speed, V and turn curvature, $1/R$. In addition, the absence of tire deformation in the model allows for further simplifications which lead to the expression of the desired tilt angle, ϕ_d as a function of the front steering angle, δ_f .

$$\phi_d = \tan^{-1}\left(\frac{a_{lat}}{g}\right) \approx \frac{a_{lat}}{g} = \frac{V^2}{R} \approx \frac{V^2 \delta_f}{l} \quad (2.3)$$

where a_{lat} is the lateral absolute acceleration.

In addition, it gives a simple definition of the acceleration perceived by the driver, a_{per} which is the component in the chassis transverse direction of the total acceleration of a specific point on the driver body. The total acceleration is defined as the sum of the lateral acceleration and the roll-induced acceleration compared to the contribution of the gravity vector and thus the perceived acceleration reads:

$$a_{per} = a_{lat} \cos(\phi) + h\ddot{\phi} - g \sin(\phi) \quad (2.4)$$

In practice, these expressions will be more or less accurate depending on the suspension geometry of the modeled NTV that influences the wheel load distribution when cornering.

Karnopp's model can easily be linearized around the upright position, $\phi = 0$ for a given advance speed, V considering the small-angle approximation. This allows for a very compact representation that can be used to design a linear controller. Beside its compactness, the fact that the sideslip dynamics is neglected limits the model applicability to low lateral accelerations and thus to small tilt angles.

2.2.2 Karnopp's model extensions

Beside its simplicity, Karnopp's model fails at representing an important part of the dynamics. In the past, single-track models have already been extensively used to study the lateral dynamics of conventional vehicles such as cars. In that case, the vehicle roll dynamics and its influence on the lateral dynamics through wheel load transfer are often accounted for via the adjustment of the cornering stiffness of the front and rear axle [14]. Conversely, Karnopp's model completely neglects the load transfer and its effect on the lateral force generation, thus omitting an important impact of the coupling between roll and lateral dynamics. Furthermore, the model does not feature the wheel axial rotation which is known to affect the roll stability through a gyroscopic moment.

Rajamani attempted to improve Karnopp's model to properly study the NTVs dynamics [19]. The longitudinal dynamics is ignored but significant improvements are made compared to the previous NTV model by taking into account the effects of slip angles, the rotating wheel gyroscopic effect and the trail dynamics. Taking into account the front and rear axle slip, the yaw rate is now a degree of freedom which depends on the vehicle yaw balance equation. Thus, the model is a single-track model composed of two bodies and comprises a total of 4 d.o.f. (Fig. 2.3).

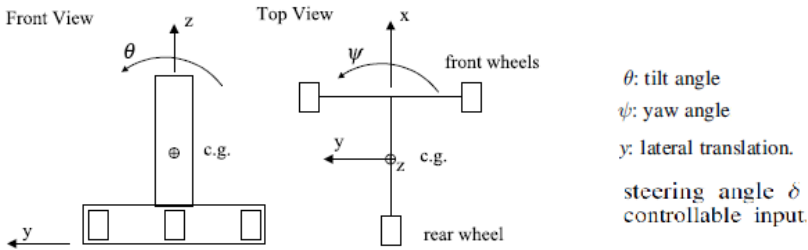


Figure 2.3: The 4-d.o.f. model developed by Rajamani [19].

In order to derive the model, two main approximations are made: (1) there is no load transfer between the left/right wheels, (2) front wheels steering does not contribute to their own lean motion. These approximations lead to the same approximated formula for the desired tilt angle than in equation (2.3). The

gyroscopic term influences the front wheel dynamics which latter contributes to the vehicle yaw and roll dynamics.

The advantage of Rajamani model is that it still features a limited number of 4 *states* to represent the NTV dynamics: the yaw rate, the lateral velocity, the roll rate and the roll angle. In fact, the steering angle is often considered as an input and the steering dynamics is consequently neglected. In that case, the model is referred as the 3-d.o.f. model.

2.2.3 Multibody modeling

If Rajamani's extension of Karnopp's model allows him to better represent and understand NTV dynamics, it is still based on some strong assumptions and does not take into account the tridimensional motion of the steering assembly and suspension between the chassis and the ground. To address these limitations, one can resort to the multibody system (MBS) approach whose use for vehicle modeling is already well-established in the literature [39]. Based on a systematic description of the articulated system to model (i.e., the NTV and its suspensions), a multibody formalism automatically generates the equations of motion, which avoids the tedious efforts and the error-prone nature of a derivation by hand. The overall dynamics of the multibody system is expressed in terms of generalized coordinates, q (relative or absolute body configuration) and their first and second derivatives (i.e., accelerations). The recent investigations on NTV dynamics have been based on those multibody approaches. As a matter of fact, research teams (e.g., TU Wien or Pisa University) have modeled the vehicle suspension and steering system in detail using dedicated MBS software (Fig. 2.4). Thanks to their increased refinement characterized by a greater number of states, these models can be used for two purposes: the validation of simplified model [25][33] or the detailed analysis of the vehicle dynamics and stability. The second reason is, of course, the most interesting and will be discussed in the next section.

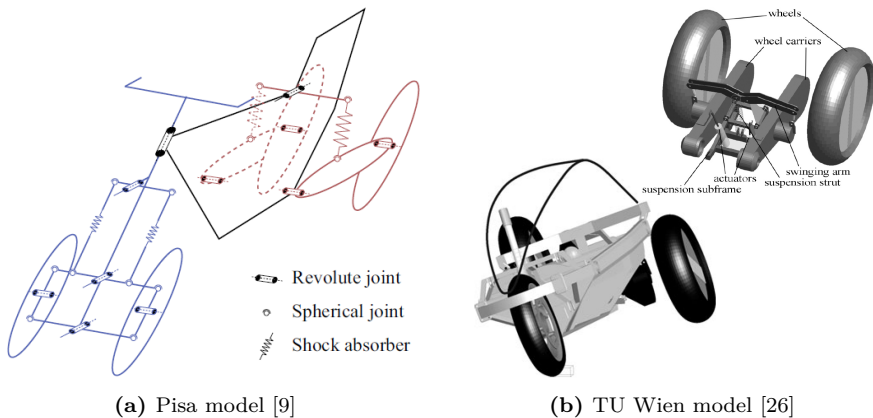


Figure 2.4: Examples of 3D MBS model of narrow track vehicles.

Furthermore, a multibody model facilitates the implementation of an advanced tire/ground model through a set of external forces and pure torques applied at a given point of the vehicles (i.e., the wheel/ground contact patch center). The MBS framework is suited to express the tire model kinematic inputs (e.g., sideslip angle, camber angle) as a function of the MBS generalized coordinates, q and to project the tire contact forces and torques on the equations of motion [15]. This feature is particularly practical when the tire slip is known to be important and when camber thrust must be taken into account. This is the case for all tilting NTVs as they all feature at least 2 wheels side by side, which unavoidably leads to sideslip angles [15].

When a MBS model is used, the considered steering input is often the torque along the steering axis, M_δ , instead of the steering angle, δ itself [26]. This approach accounts for the modeling of the steering assembly dynamics which is of the utmost importance for the vehicle self-stability assessment.

Nevertheless, accurate modeling of the NTV dynamics via a MBS approach will often lead to a bigger model whose complexity will often lead to greater computation time. In addition, a MBS modeling framework requires a minimum knowledge on MBS theory otherwise the resulting model might incorrectly represent the actual vehicle.

2.2.4 Tire/ground contact modeling

The tire model used for NTV modeling are often simple and based on front/rear axle cornering stiffness as done in a conventional single-track model. Such approach can be acceptable for a delta configuration for which there is no complex steering assembly, but when a tadpole configuration is studied, the steering and suspension geometry can lead to important sideslip angles and, in that case, constant cornering stiffness might not properly represent the tire nonlinear behavior [19]. In addition, the camber stiffness⁵ should always be taken into account for NTVs as it participates to the torque balances around the wheel steering axis in steady state cornering. Advanced tire/ground modeling has sometimes been used but only for validation purposes. In his PhD Thesis [25], Berote at the University of Bath considered the Pacejka's "Magic Formula" to model the Clever vehicle (Section A.1.4), which allows for a proper modeling of the tire nonlinearities and saturation [40]. However, the parameters used are those of a car which could be sufficient for the rear tire of the non-tilting rear pod but inaccurate for the leaned front wheel of a delta vehicle (e.g., the Clever). In order to properly model the tire/ground interaction for a tilting NTV, the magic formula parameters have to be adapted to the case of a motorcycle tire. In his work on NTV design based on dynamic modeling [15], Verlé has used and adapted the tire parameters given by Sharp in [41].

⁵The amount of lateral force produced when the tire is tilted a given angle with respect to the upright position.

2.3 Tilting NTV dynamic analysis

As the majority of projects cited above focused on the control of the NTV, a prior dynamic analysis is often performed. The quality of the latter, however, strongly depends on the model refinement.

For simple models derived based on Karnopp's and Rajamani's work, elementary characterization is carried out by performing input step on the steering angle and observing the system response [42][29][32]. In addition, sensitivity analyses are performed on parameters such as the vehicle mass or the speed [13][29]. Barker and Berote take a special interest at the pitch angle, ξ for the tilt joint axis and how this angle induces a rear steering angle, δ_r for the non-tilting pod (Fig. 2.5).

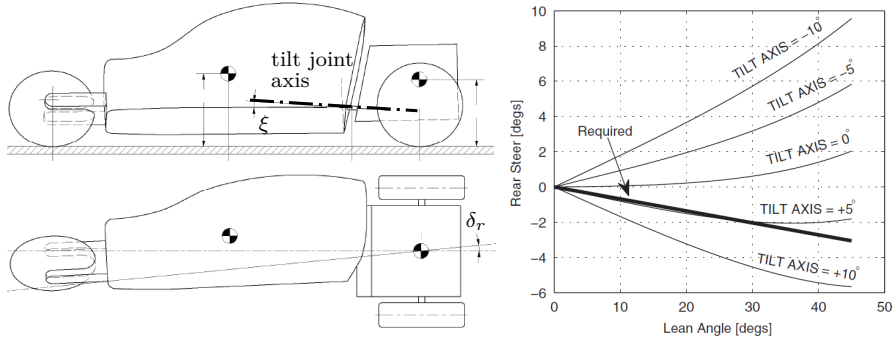


Figure 2.5: Longitudinal inclination (pitch) of the tilt axis and its influence on the induced rear steering of the non-tilting pod [24].

As the steering dynamics is often neglected with the steering angle being driven, the influence of the vehicle speed on its stability is not properly identified. The dynamic analysis is only limited to the simplified model itself and cannot be extended to the actual vehicle. Nevertheless, the simplified model still allows for the determination of the steady state steering angle, δ_{ss} that is necessary to keep the vehicle in a steady state cornering.

Based on his enhanced model, Rajamani has tried to identify the role of the gyroscopic effect on the vehicle stability [19]. Considering the trail equation (2.5) developed based on the steering axis balance (Fig. 2.6), he shows that a tilt motion to the left side will cause a torque on the steering axis that will automatically steer the vehicle in the same direction. When the wheel is steered, the lateral force generated will always create a self-aligning torque around the steering axis due to the trail (mechanical and pneumatic), which automatically realigns the wheel. For more details on trail dynamics and its influence on roll stability, one can refer to the works of Schwab and Sharp on bicycle and motorcycle stability [43][44].

$$M_{trail} = mg \frac{l_r}{l_f + l_r} \gamma \cos(\beta) \sin(\theta) - F_f \cos(\theta) (\gamma + \gamma_{pneumatic} \cos(\beta)) \quad (2.5)$$

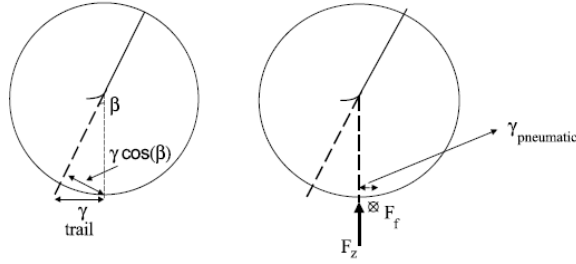


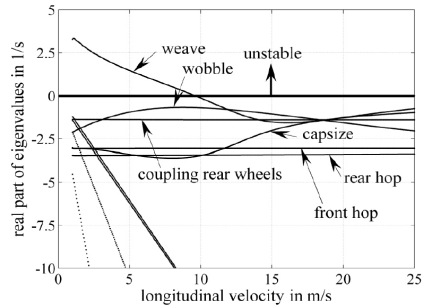
Figure 2.6: Front wheel forces and effect of the trail [19].

Beside interesting considerations on self-stability, no self-stabilizing speed is determined for the modeled NTV.

Stability analysis With more elaborated models, the dynamic analysis is more thorough and comprehensive. Thanks to the MBS model developed with Simpack [45], Plöchl and Edelmann are able to analyze the roll stability of an unmanned NTV with respect to the vehicle speed [27][26]. Based on modal analyses performed in steady state straight lines and steady state cornering, Edelmann identifies unstable motorcycle motion modes (i.e., weave, capsize and castering) and argues that a stability electronic control is necessary in order to offer a car-like driving experience (Fig. 2.7). They are able, for the developed prototype, to find the self-stable speed (i.e., about 10 m/s) above which all lateral modes are stable.



(a) prototype



(b) stability analysis

Figure 2.7: The TU Wien NTV exhibits self-stability above a speed of about 10 m/s according to the MBS model [27].

Based on a similar multibody framework, Bucchi from the University of Pisa has modeled the Resolve 4-wheeler [46] and was able to show the influence of velocity on the NTV roll stability. The vehicle behavior is closer to what is observed for a conventional motorcycle [41] with a capsize motion that is unstable over the overall speed range. In addition, Bucchi completes his dynamic

study with a sensitivity analysis on the front trail, and concludes that it has a strong influence on the weave and castering modes [9].

It is important to notice that a complex model does not always imply a complex dynamic analysis. Indeed, there are many MBS models that have been developed to validate simplified controls model without realizing any analysis on roll stability [47][25][32].

Free-to-caster dynamics When considering NTV dynamics aside from any control action, a steering input is often considered, according to the free-to-roll case presented in the introduction (Section 1.2.1). Another possible way to drive a NTV is to consider a tilt action with a free-to-caster (FTC) steering assembly (Section 1.2.1). From 2015 to 2017, the dynamics of a free-to-caster NTV has been investigated by Jeffrey Tan, under the supervision of Prof. Suda from the University of Tokyo [33][32]. Based on a simplified model, he performs tilt input and observes the steering angle dynamics and stability at a given speed of 20 km/h. The simulation results are compared to actual data collected during driving tests on the prototype vehicle (Fig. 2.8).



Figure 2.8: The prototype NTV developed by Equos working based on a FTC actuation [32].

In addition, he studies the influence of the trail on the steering stability. However, the simple modeling method he has used does not allow him to make a thorough stability analysis of the vehicle with respect to the speed. As a matter of fact, nothing is said about the prototype instability at low speed and how they handle it.

For the past two decades, Phillip James has developed several NTV prototypes based on the FTC driving principle (Fig. 2.9). Although he did not publish any scientific papers, James has addressed many interesting aspects of FTC driving and stability on his blog [48]. Actually, it is unclear how much of the Equos vehicle concept has been inspired by James's work as the latter claims that many of Equos patents have infringed his own work.

The dynamics of the FTR and FTC driving modes have been separately investigated, more or less thoroughly, depending on the research teams and the available tools. To the author knowledge, there has been no work that compares the FTR and the FTC modes in a thorough and systematic way for the same vehicle.



Figure 2.9: FTC-driven NTV prototypes developed by James [48].

2.4 Tilting control modes

The different tilting modes presented in the introduction have been more or less investigated during the past three decades. Since the beginning of the 90s, most research projects have been focusing on the development of the DTC, STC and of their combination called SDTC.

2.4.1 DTC

Generally, the DTC is firstly investigated and compared to the non-tilting case as done by Karnopp and Hibbard [17][49]. For their simplified model, they show that tilting the vehicle through DTC enables the complete cancellation of acceleration in (2.4) and a 30% reduction of the wheel load transfer compared to a non-tilting vehicle in steady state cornering. The DTC control is performed by the application of a torque, M_t in order to track the steady-state tilt angle, ϕ_d from equation (2.3).

Based on the simple PD controller in equation (2.6), Karnopp and So show that DTC action leads to a dangerous torque in the transient phase. The tilting torque in steady state should, however, be zero as the vehicle chassis is aligned with the resultant of the gravity and centrifugal effects.

$$M_t = K_p(\theta_d - \theta) - K_d\dot{\theta} \quad (2.6)$$

Based on common sense and citing previous work, many research projects realized that DTC alone is not a viable solution as it demands a large amount of torque, M_t in the transient phase. This transient torque can even lead to wheel lifting as experienced on the Clever prototype [24]. In 1997, So and Karnopp already discussed the limitation of the DTC control [11]. Nevertheless, DTC remained the subject of investigations which attempted to improve its efficiency through the use of more elaborated nonlinear control techniques [21][37].

2.4.2 STC

The steering tilt control (STC), inspired by two-wheeler riding (i.e., free-to-roll), presents itself as a viable solution to avoid the important transient tilting torque and the inherent wheel load transfer of the DTC. In 1992, Karnopp has already

shown that, similarly to a two-wheeler, a NTV can be tilted by applying a countersteering. A simple PD control scheme involving the tilt angle θ shows the necessity of countersteering before a turn and is related to the reverse action associated with non-minimum phase systems [17].

$$\delta = K_p(\theta_d - \theta) - K_d\dot{\theta} \quad (2.7)$$

This early observation has led many projects to also consider the STC mode in their investigations [29][50][26]. Generally, the steering action δ is taken into account as a modulation of the driver steering input δ_{sw} , as shown in equation (2.8). This is related to the use of Rajamani's 3-d.o.f. model by many research teams which do not take the front steering wheel dynamics into account.

$$\delta = \delta_{sw} + \delta_c \quad (2.8)$$

Among the Minnesota team led by Prof. Rajamani, Gohl has focused his work on the STC mode [50]. He proposed an improved PD controller which differs from Karnopp's work by the introduction of a steady-state feedforward term $\delta_{ss}(\theta_d)$, function of the desired lean angle θ_d . As a matter of fact, conversely to the DTC mode, STC necessitates a non-zero input value in order to keep the NTV in the turn (i.e., a steering torque, M_δ or a steering angle, δ). This additional feedforward term leads to the cancellation of the steady-state error. In cornering, the feedback part is first dominant then, once the tilt angle error starts to diminish the driver feedforward term becomes dominant and provides the steady state steering input required to achieve the turn.

$$\delta_f = K_p(\theta_d - \theta) - K_d\dot{\theta} + \delta_{ss}(\theta_d) \quad (2.9)$$

On the other hand, when the available model is further elaborated and takes the steering angle dynamics into account, the STC is performed via the application of a steering torque, M_δ rather than a steering input modulation, δ_c . In that case a steering by wire logic is used and the STC controller tracks a steady state cornering configuration which is expressed based on the steering wheel angle provided by the driver [26][51]. In a similar way to Gohl did, a feedforward term is used and computed based on the vehicle speed V and the desired lean angle θ_d which together define a given steady state cornering configuration.

STC does not limit itself to NTV but can also be applied to single-track vehicles (i.e., bicycle or motorcycle). Based on this observation, a large share of the literature on 2-wheeler control can directly be linked to the STC mode as previously defined in this work.

2.4.3 SDTC

If STC has been proposed early on for reducing the actuation in transient maneuvers, its limitations have also been addressed by Karnopp and his team. In 1997, the combination of both the DTC and STC was already proposed by So and Karnopp [11]. This has significantly influenced the subsequent investigations (from the early 2000s until now) which have been focusing on the SDTC framework for the control of NTVs.

So's work First, So and Karnopp compare both modes with PD controllers and show that, during transient maneuvers, a STC system leads to a lower perceived acceleration than that of a DTC equipped vehicle (Fig. 2.10).

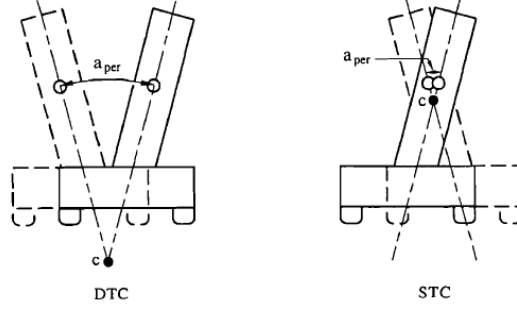


Figure 2.10: Perceived acceleration and the instantaneous roll axis c for the two envisaged control modes [11].

Thus, they propose to combine the two modes; however, they are unable to successfully combine both of them in a continuous manner due to a conflict between the two control methodologies: the countersteering operated in STC leads to an erroneous computation of the desired tilt angle for the DTC as the latter is based on the actual steering angle δ rather than the driver steering input δ_{sw} [11]. Thus, rather than blending the two modes, they consider switching from one to another at a prescribed speed of 8 m/s with a hysteresis-type strategy (Fig. 2.11) [52].

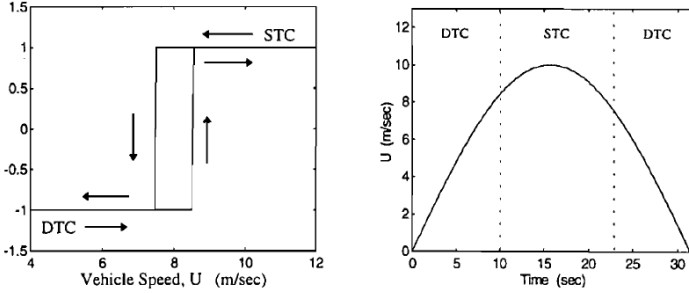


Figure 2.11: Switching SDTC approach with a hysteresis-type strategy [11].

Snell's work Based on previous work, Snell and Karnopp have proposed another control approach which consists in minimizing or eliminating the perceived acceleration, a_{per} at a selected point C on the tilting chassis median plane (Fig. 2.12), which amounts to forcing the vehicle to roll around an axis passing through the same point [53]. Considering a combined SDTC approach, Snells overcomes the two modes disagreement by computing the desired lateral acceleration a_d based on the driver's steer demand δ_{sw} , rather than on the front wheel steer angle δ .

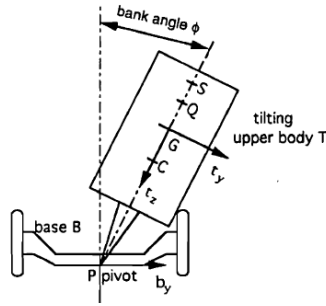


Figure 2.12: Different reference points on the tilting chassis of a NTV can be chosen to minimize the perceived acceleration [53].

Changing the C point position along the chassis vertical axis allows Snell's SDTC controller to switch from a pure DTC behavior to a STC behavior. He suggested locating the virtual roll axis where the driver senses motion (i.e., at the head level) which results in a SDTC control very close to the STC mode. However, the difference between Snell's control law and a classical STC is that the tilting torque is always considered as a controller input and can thus be used to stabilize the roll dynamics in case of external disturbances at any time.

The control approach is thus very interesting but not always well understood by the subsequent research teams. However, it is based on the perceived acceleration *cancellation* which can only be achieved in the case of Karnopp's simplified model.

Kidane's work Regarding the SDTC system, Kidane at the University of Minnesota argues that the main challenge is the transient phase for which the reference tilt angles of the two modes need to be consistent. Kidane designs his controller by considering θ_d as a function of the driver input $f(\delta_{sw})$ for both STC and DTC removing any conflicting signals between the two control modes [12]. The proposed SDTC is thus a switching strategy with exponential factors α and β that multiply the control inputs depending on the speed (Fig. 2.13).

The DTC is a simple PD controller while for the STC, he considers an integral term in order to produce the steady-state front steering angle. This integral action should ensure that, in case of vehicle parameters changes (e.g., an additional payload mass), the tilt angle error still vanishes at steady state, resulting in the right steady-state steering input without requiring knowledge of any vehicle parameters. Later on, Kidane removes the speed dependency for the SDTC exponential factors and assumes that both modes are active on the whole speed range. Simulations with a varying speed are performed and it is observed that the peak control efforts of the STC and DTC are out of phase by 180 degrees: when the DTC effort is at its highest value (lowest speed), the STC effort is at its lowest and vice versa, showing that the combined controller is capable of proportioning the STC and DTC depending on the velocity. However, the differences of performance between the SDTC and the original STC and

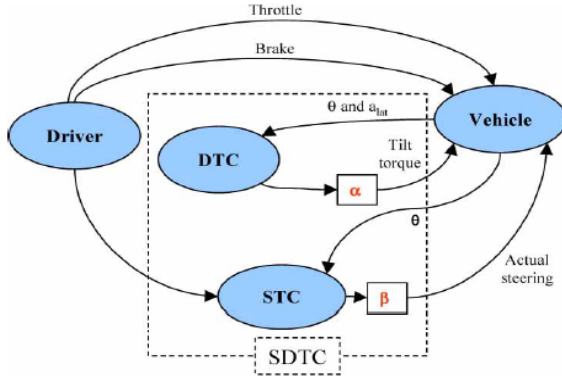


Figure 2.13: Kidane strategy for SDTC control [12].

DTC are neither observed nor discussed by Kidane.

In addition, experimental validations are considered on a man-size prototype vehicle with a tadpole design and a front double wishbone suspension (Fig. 2.14). Based on these experiments, Kidane and Rajamani concede that the omission of the advance speed, V , for the desired tilt angle calculation leads to an excessive understeer tendency. They considered a revised θ_d calculation which includes the forward speed. Although the prototype is capable of large tilting (up to 35 degrees), the experimental results are limited to 14 degrees tilt angle and acceleration to a value of around 2.5 m/s^2 . The performances of the control system in vigorous, transient maneuvers are not assessed.



Figure 2.14: Research NTV prototype developed by Kidane at the University of Minnesota [54].

Berote's and Robertsson's work Within the Clever project, NTV control through the use of a SDTC approach was also studied. The two modes are continuously combined rather than switched according to the vehicle advance speed. Berote claims that using a negative gain feedback $K_{\delta\theta}$ between the tilt error θ_e and the steer modulation δ_c reduces the amount of steering at the front wheel δ in an effort to reach the require tilt angle θ_d more rapidly, possibly leading to some countersteering (Fig. 2.15).

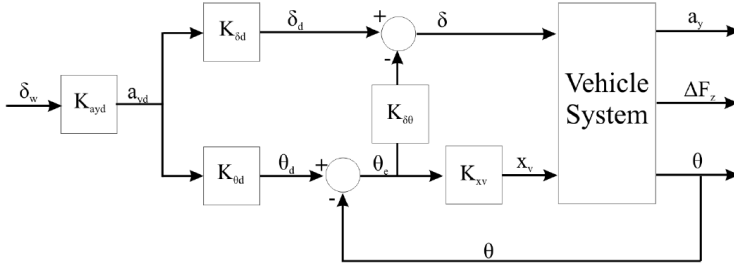


Figure 2.15: Block-diagram for SDTC as proposed by Berote in [25]. The active steering gain $K_{\delta\theta}$ links the tilt error to the steering angle modulation.

Berote notes that the optimal STC behavior varies as a function of speed with smaller steer inputs required at higher speeds. Therefore, simulations of the lateral acceleration response to a steer step input are used to derive values for a nonlinear speed-dependent active steering gain (Fig. 2.16).

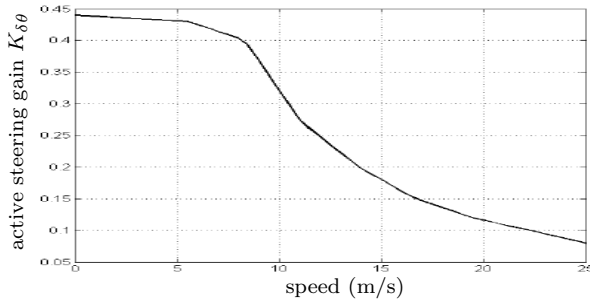


Figure 2.16: Active steering gain as function of the speed [25].

Later on, Robertsson, based on extensive track experiments, shows that drivers must adapt their steer input to SDTC system in order to maintain path following. The impact of revised steer input on roll stability is investigated through simulations [55]. The influence of the active steering gain $K_{\delta\theta}$ on the vehicle's trajectory and its roll stability is examined by Robertson who shows that higher gains improve stability but logically lower the path following ability. The subjective assessment confirmed the simulation tendencies regarding the loss of path tracking ability caused by active countersteering. With respect to drivability, drivers reported that generating a revised steer input felt natural after a short period of acclimatization [13]. However, only moderate cornering tests are conducted and a more severe maneuver with higher lateral accelerations may lead to a greater influence of the SDTC mode.

Finally, simulations are used by Robertsson to show that a feedforward element for SDTC could achieve a fast active steering response and a 30% reduction in the load transfer in transient conditions. Using feedforward of the tilt angle

error rate is found to offer better vehicle roll stability than a feedforward action of the steering angle rate [13].

Mourad’s work In 2011, NTV control was the subject of a PhD thesis conducted by Mourad under the supervision of Prof. Chevrel and Prof. Claveau from the Ecole des Mines de Nantes. In accordance with Snell’s work [53], they propose to regulate the perceived acceleration rather than tracking a tilt angle [56]. Despite a built prototype and the availability of a proper multibody model [47], Mourad developed a simplified model based on Rajamani’s work [19]. Based on that extended model (Fig. 2.17.a), Mourad defines a well-posed H_2 standard problem that minimizes the regulated signals which are composed of the output deviation (i.e., the perceived acceleration) and the 2 inputs, all multiplied by a static gain matrix, $Q_0 = \text{diag}(Q, R_1, R_2)$ (Fig. 2.17.b). Adjusting the static matrix gains allows Mourad to design either a STC, DTC or SDTC controller. The resulting controller is of low complexity but with a feedback term on the states and a feedforward term on the driver steering angle, δ_{sw} and its derivative, $\dot{\delta}_{sw}$.

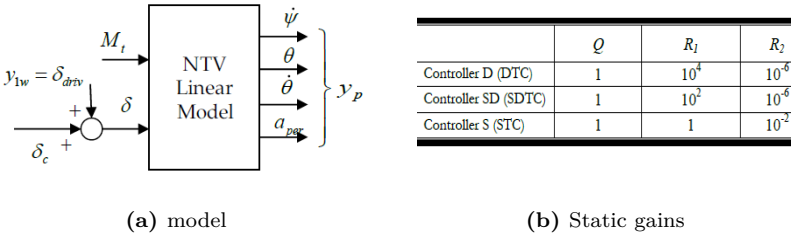


Figure 2.17: Mourad’s controller: (a) Extended linear model with perceived acceleration and steering angle modulation, (b) Matrix gains enabling the selection of the control mode [57].

In addition, a gain scheduling on the vehicle speed is considered. The control gains are computed for each speed, keeping constant parameters for the static gain matrix Q_0 . The controller robustness and performances are tested through time simulations. The controller synthesis is very interesting and allows for an efficient comparison of STC, DTC and SDTC modes. However, it is based on the perceived acceleration cancellation which is only valid in the framework of Karnopp’s simplified model.

2.4.4 LTC, FTC and torque vectoring

As it can be seen in the above section, DTC, STC and SDTC have been the main subjects of the NTV control research over the three past decades. The leaning tilt control (LTC), which consists in applying a tilting torque and letting the steering free, is also a possibility (Section 1.2.1).

In [22], Kidane also discusses the case of LTC which he calls free-to-steer DTC. Based on the prototype and the control system implemented, test maneuvers are performed with different desired tilt angles. It is observed that above a tilt angle of 10 degrees, the controller is not able to properly track the desired tilt angle and oscillations appear. Based on the simplified model available, Kidane explains the controller failure by a difference between the tilt angle of the front wheels and that of the vehicle, resulting in a higher lateral force and causing the tilt motor to saturate. Kidane, however, misses a big part of the LTC behavior, as the wheel load transfer influence on the vehicle steady state cornering is not considered in their simplified model.

From this observation, it seems that the inherent wheel load transfer caused by the tilt action during steady state cornering and its influence on the vehicle dynamics can only be studied with the help of a proper modeling of the suspension and steering assembly geometry. We think that the lack of accurate models together with the atypical free-to-caster driving principle has limited the study of the FTC motion and the LTC mode. Tan's work on FTC-driven NTVs certainly interesting from the dynamic point of view [33]. However, it is not clear how the controller works and how the driver input is interpreted to compute the lean angle.

More recently, new NTV control strategies have been developed based on torque vectoring⁶ methods to assist the rider in balancing a narrow tilting vehicle, thus reducing the countersteering requirements [38][35]. The proposed approach is designed via the combination of two torque controllers: a steer angle-based torque vectoring controller and a tilting compensator-based torque vectoring controller (Fig. 2.18). The developed controllers have the ability to reduce the countersteering requirements from the rider and improve the tilting behavior during turning an NTV [38]. However, such control approach can only be achieved if the vehicle features two side-to-side powered wheels .

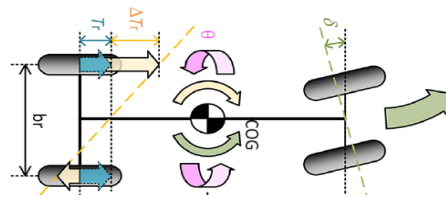


Figure 2.18: Torque vectoring is used to improve the NTV tilt stability [38].

2.5 Control techniques

Regardless of the tilting control mode considered, there exist many control techniques to synthesize a control law or design a controller. This section aims at summarizing the different methods used during the past three decades for

⁶Torque vectoring consists in varying the torques applied to each driven wheel in order to influence the vehicle lateral behavior (e.g., the Electronic Stability Program (ESP)).

NTV control. If the first NTV controllers were using simple linear techniques, there have also been attempts to develop more elaborated controllers motivated by the need to find an optimum solution to combine the steering and the tilting actions within the above-mentioned SDTC framework.

2.5.1 Linear control techniques

Historically, NTV controllers have often been designed based on simplified linear approaches. In fact, many researcher groups worked with the simplified models developed by Karnopp and Rajamani [11][19] which they often linearized around the upright position allowing them to design simple P, PD or PID controllers. This concerns most of the NTV control research including Karnopp's work [11][53], Minnesota's investigations [12] and the Clever project [25][13] but also other more isolated research projects [58][59][15].

Often, the linearization of the vehicle model allows for the formulation and solving of a linear quadratic regulator (LQR) [26][35]. LQR enables to weigh the controller errors with respect to the input cost through a quadratic function, while ensuring the stability of the linear system. Linear quadratic integral (LQI) control is also investigated and allows for cancellation of the static error while tracking the vehicle lean angle [60].

Finally, thanks to a simplified linear representation of a NTV (Fig. 2.15 and 2.17), feedforward terms on the driver's steering input δ_{sw} can be easily taken into account [25][29][26].

2.5.2 Nonlinear control techniques

Gain scheduling Given the inherent nonlinearity of NTV dynamics with respect to the advance speed, a gain-scheduling approach on the vehicle speed has often been adopted [25][29]. By realizing series of linearizations at different vehicle speeds, controller gains can be obtained and then applied according to the actual vehicle speed, during a speed-varying simulation (Fig. 2.16). More generally, this gain-scheduling can be performed on the vehicle velocity and lean angle allowing for a more precise determination of feedback gains, but also for the computation of the steady-state feedforward gains (i.e., steering input) [26].

Receding horizon control The first to consider an advanced nonlinear control method is Piyabongkarn, under the supervision of Rajamani at the University of Minnesota. In 2004, he tried to develop a systematic SDTC approach that synchronizes the tilt and cornering maneuvers in order to reduce actuator requirements and improve ride qualities [21]. He argues that the tilting and steering need to be synchronized because any lag in the tilting dynamics necessitates large peaks in the required actuator torque. He considers a receding horizon control (RHC) as defined in [61], assuming that short-term preview information about the road curvature in front of the vehicle is available (e.g., provided by a lane-detection vision system or a combination of GPS tracking and

a road map database). The RHC implies solving an optimization problem on the receding horizon for the actual desired tilt angle, while satisfying constraints on the tilting torque (Fig. 2.19). Simulation results show the torque saturation provided by the solution of the constrained optimization problem. Nevertheless, model mismatch between the linearized and the nonlinear model is so significant that the linear RHC would lead to the use of extremely large torque values, several orders of magnitude larger than with the linear model assumption. This explains why the simulations in curves are limited to large radii of 500 meters.

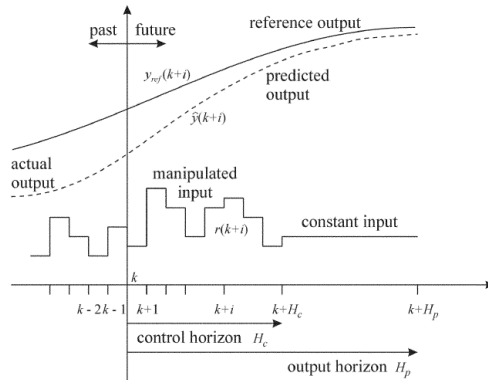


Figure 2.19: Receding horizon control strategy as envisaged by Piyabongkarn [21].

Sliding modes In 2015, nonlinear methods have also been considered by Mourad and Prof. Claveau at the Ecole des Mines de Nantes considering sliding modes [62]. Based on few assumptions on the tire forces model, the flatness⁷ of the simplified model used can be verified and a set of three flat outputs is found [30]. Then, the longitudinal-lateral dynamics of NTV is linearized thanks to a static state feedback, and a simplified PID controller is used to track the three flat outputs. The trajectories for the latter are generated based on the driver steering angle and the propulsion force. The system linearization is verified and the controller is tested with respect to the linear SDTC approach developed by Mourad. If the nonlinear approach has the advantage to treat the longitudinal dynamics and bypass the gain-scheduling design, it is important to remind that feedback linearization has a tendency to erase a part of the original dynamics making the stability analysis more complicated. Similar research has been carried out in Brazil from 2007 to 2016 by Prof. Roquero from the Federal University of Santa Catarina [63][64]. It is claimed by Roquero and his team that such nonlinear approach performs better than the previous one based on a reduced linear model. Nevertheless, it is not clear whether the model developed is accurate when it comes to vehicle geometry (suspension motion) and dynamics (wheel inertia and tire/ground contact).

⁷more details on this concept and on sliding mode control can be found in [62]

Feedback linearization Another nonlinear control method that has been investigated for NTV control is feedback linearization [62]. It is again Piyabongkarn who has been the first to investigate such nonlinear method. He, however, limits himself to direct tilt control (DTC). Nonlinear feedback linearization consists in applying a nonlinear input f_{nl} leading to an input-output linear system (the reference tilt angle θ_d being the input). Taking advantage of the model simplicity, the roll equation is readily usable to derive a nonlinear input law for the tilting torque M_t such that the error dynamics, is linear:

$$M_t = f_{nl}(\ddot{\theta}, \dot{\theta}, \theta_d, K_D, K_P) \quad (2.10)$$

$$\ddot{e}_\theta + K_D \dot{e}_\theta + K_P e_\theta = 0 \quad (2.11)$$

Where $e_\theta = \theta - \theta_d$ and K_D, K_P are two parameters that can be tuned to define the error dynamics in (2.11). The nonlinear command f_{nl} is composed of several terms whose influence on the error dynamics has been investigated by Piyabongkarn. The performances of the nonlinear control are assessed by comparison with LQR and PD controllers. The nonlinear controller keeps the tilt angle error e_θ close to zero while the amount of torque needed for the control is strongly reduced compared to the two other linear controllers. However, only low lateral acceleration simulations are performed.

In 2018, Ren investigated DTC-based control under the supervision of Prof. Greenwood at the University of Warwick. Although they are part of the resolve project which included the Pisa University team, they have based their study on a simplified model for the NTV roll and lateral dynamics [37]. A 8-shape path is used together with a simple PI virtual rider model to validate the two developed controllers (Section A.2.5). The nonlinear controller performs better than the conventional linear and gain-scheduling controllers with less roll angle tracking error under different speeds. In addition, it can help riders to improve the yaw rate tracking performance [37].

If the control methods presented above are more complex than the one previously used by Karnopp in the 1990s, the vehicle model used remains very simple. While using simplified model is common to better understand the physics behind a system, it has been shown that these models do not properly represent NTVs stability behavior with respect to the forward speed, the inverted pendulum being unstable for the whole speed range. Most of the coupling between the roll and lateral dynamics depends on the suspension motion which can be highly nonlinear when high lateral accelerations are considered. Nevertheless, the investigations presented above show that nonlinear methods could be of interest to optimize and improve the control of NTV for the STC, DTC and SDTC mode.

2.6 Discussions and conclusions

From this state of the art, one can observe that NTV control has often been investigated as part of projects aiming at developing a vehicle. Hence, there is very little building up on previous scientific works as each project features its own specificities (i.e., vehicle morphology, control strategy, ...). Nevertheless, some general tendencies on the model, the dynamic analysis and the control can be identified based on the previous sections.

On the model Generally, the analyses are based on the simplified model developed by Karnopp and Rajamani. If this model is enough to highlight the necessity of countersteering and some speed dependency in the dynamics (i.e., tilting effort), it is clearly limited when it comes to precisely understand and analyze the vehicle dynamics. The omission of the wheel load transfer, the tire nonlinearities, the gyroscopic effect and the suspensions/steering assembly geometry cannot properly account for the strong coupling occurring between the lateral and roll dynamics for tilting NTV.

Multibody modeling, on the other hand, allows for a proper awareness of the aforementioned dynamic phenomena. However, the past studies have shown that precise vehicle modeling through MBS does not automatically come with an insightful dynamic analysis of the studied NTV, as MBS models have been mainly used for validation of simplified ones.

Regarding the input modeling, the choice of a steering angle δ as a kinematic input is appealing but erases the steering assembly dynamics, which is indispensable in order to study the vehicle stability with respect to speed. To this extent, the steering torque should be alternatively envisaged as an enlightening steering input rather than the angle.

On the dynamic analysis Interesting dynamic analyses are performed by two research teams in particular: the TU Wien and the University of Pisa. Based on accurate MBS models, they are able to look at the stability of their NTV in straight line. Additionally, considerable efforts are made to identify the nature of the different modes and make the parallel with the two-wheeler literature. Moreover, sensitivity analyses are performed on key parameters such as the front trail. However, the analysis is either limited to the upright straight line stability or to low lateral acceleration. The transient effects of the decreasing curve radius, corresponding to an increasing lateral acceleration, on the vehicle stability and the steady-state configuration (i.e., roll angle, steering torque) are not always soundly analyzed. Nonetheless, there are no thorough investigations on the tridimensional nonlinearities inherent to the suspension, tire and steering system geometry that occur at high lateral acceleration.

In addition, these dynamic analyses generally focus on a steering action to keep the vehicle (FTR) in cornering but very little investigation has been done on the free-to-caster (FTC) case which is based on a tilting action. The dynamics of the FTR and FTC driving mode have been separately investigated, more

or less thoroughly but, to the author knowledge, there has been no work that compares the FTR and the FTC mode in an accurate and systematic way for the same vehicle.

On the NTV control It is clear from the above sections that SDTC is considered as the only applicable form of active tilt control. DTC alone will always come along with a higher energy consumption (large tilting torque) in transient maneuvers coupled with a greater vehicle instability (i.e., wheel load transfer). In addition, it can also cause discomfort due to the forced tilt motion which can result in an important amount of perceived acceleration for the driver. Piyabongkarn rightly argues that the tilting and steering need to be synchronized because any lag in the tilting dynamics necessitates large peaks in the required actuator torque to tilt the chassis back in the right direction. Nevertheless, there have been attempts to use more elaborated nonlinear control methods for DTC-based controllers and it has been shown that such methods have led to the reduction of the tilting torque.

The combined use of tilting and steering action on the overall speed range seems more appropriate than a switching strategy as it allows for a better handling of road irregularities and the realization of extreme maneuvers [53][12]. The larger the amount of active steering, the more the SDTC mode will improve the vehicle stability by reducing the necessary tilting torque. Nevertheless, it can also logically lower the path following ability because of the countersteering. Even if experiments showed that the drivers can adapt themselves to the SDTC, it is important that the driving experience does not overly differ from that of a conventional car [13].

The majority of the projects have based their NTV control on the tracking of the lean angle. There have been different approaches to define this reference tilt angle, θ_d [12]. While the majority have based their controller on the simplified formula from equation (2.3), others have assumed that the tilt angle should be a function of the steering wheel angle fixed by the driver. In order to obtain a zero tilting torque in steady state cornering, the reference lean angle needs to be properly computed. As many of the control laws have been developed on the simplified Rajamani model, the reference lean angle has always been defined based on the roll balance between the centrifugal effect and the gravity effect from equation (2.3). The main issue with this approach is that no vehicle obeys such simplified equation as in reality, the suspension geometry can strongly influence the roll equilibrium and the general steady-state configuration. Only an accurate modeling of the vehicle geometry and dynamics allows for a proper determination of the steady-state tilt angle. In the literature, some investigations have been based on the cancellation of the perceived acceleration [29]. Regrettably, the latter is never really zero during the cornering as the suspension geometry and tire dynamics alter the roll balance. This observation is true for a steering action (FTR) and even more critical when tilting action (FTC) is considered for steady state cornering.

Conclusion To conclude, since Karnopp and his first NTV model, the modeling of NTV dynamics has been improving with many research projects using dedicated MBS software to represent the vehicle dynamics. On the other hand, the control methods have been more and more refined in order to address the optimality problem of SDTC and to improve the DTC performance. However, the model used is often too simple when advanced nonlinear techniques are considered and consequently omits most of the coupling between the roll and lateral dynamics, which can be highly nonlinear, especially when high lateral accelerations are considered. Based on these observations, it is obvious that there is an interest in studying the control of NTVs through the use of nonlinear methods with appropriate models, typically a MBS approach, that are able to properly account for the NTV complex dynamics and non-minimum phase behavior.

In addition, if STC has been thoroughly investigated considering the free-to-roll (FTR) principle, very little has been done on the NTV control with the free-to-caster (FTC) approach. Thus, there is a clear interest in studying transient maneuvers and steady state cornering configuration with this FTC approach. This will be possible only with a precise representation of the NTV tridimensional motion and dynamics (i.e., suspension travel, tire/ground interaction) enabled in our case by the multibody approach.

This conclusion is in accordance with the two thesis objectives that were defined in the introduction (Section 1.3) and shows that there is a clear interest in a thorough study of NTVs dynamics and their nonlinear control.

Multibody vehicle model

3

The vehicle morphology and geometry used in this thesis correspond to a narrow three-wheeler, with a reverse-trike configuration, designed at the Université catholique de Louvain [15][2]. The vehicle tricycle configuration is justified by the necessity of having two wheels side-by-side to perform a tilting actuation (Fig. 1.13). The rear axle unit is similar to the one of a motorbike and is equipped with an electrical motor for propulsion. The front wheels unit is comparable to a traditional double wishbone suspension and features a special mounting that provides the additional rotation degree of freedom (d.o.f.) between the chassis and the suspension assembly as described later on (Section 3.3.1). The steering assembly is composed of two track rods connected to the front wheel carriers and put into motion by the rotation of a steering fork. In order to keep the center of mass as low as possible, the driver sits rather low in a similar manner as inside a few of NTV prototypes such as the Clever and the Carver (Fig. 3.1). More considerations on the vehicle design and its justification can be found in Verl 's PhD thesis which focuses on the mechanical and dynamic design of the reference NTV [2]. For a complete description of the vehicle dynamic parameters (topology, masses, inertias, lengths, etc.), one can also refer to Verl 's work.



Figure 3.1: The Wemoov vehicle developed by Aubain Verl  in [2].

The present work has been based on the vehicle developed by Verl  for various reasons. Firstly, this vehicle is the result of a thorough design procedure which aimed at providing the best dynamic performances given the narrowness of the vehicle. Secondly, the vehicle and its dynamics parameters (e.g., suspension geometry) have been optimized with respect to dynamic criteria such as the minimization of the tire/ground slip during cornering. Finally, an advanced model of the vehicle was already available and could be adapted for the needs of the present work (i.e., kinematic chain reduction, rigidification of the suspensions and derivation of the state-space representation).

This thesis not only focuses on the vehicle control but also on the actuation dynamics for the serial modes (Section 1.2.1). A precise modeling of the vehicle dynamics is thus indispensable. As shown in the literature review, there are several possibilities when it comes to model the dynamics of narrow track vehicles. The majority of previous works has been based on simple single-track models extended by the authors in order to take into account the cabin roll motion and sometimes the wheel load transfer and/or the gyroscopic effect. In this work, the vehicle is modeled as a multibody system (MBS), i.e., a set of *rigid* bodies connected by any combination of prismatic and/or rotational joints. Such generic approach presents many advantages for vehicle modeling:

- The MBS modeling is based on a systematic description of the studied articulated system. First, the system is properly defined (topology, masses, inertias, lengths, etc.) within the MBS framework and then the equations of motion are automatically derived and expressed in terms of a set of coordinates. The MBS approach limits the tedious efforts and the error-prone nature of a derivation by hand of the equations. Given the complex geometry of a NTV and the specific tilt actuation, precise modeling of the vehicle tridimensional dynamics is of significant importance in order to properly account for the wheel load transfer, the gyroscopic effect and the nonlinearities inherent to high lateral acceleration maneuvers, for instance.
- MBS formulations offer a high *versatility* which enables the coupling with auxiliary constitutive models (e.g., a tire/ground or a aerodynamic model). The MBS framework can provide, in an automatic and efficient way, the input kinematic quantities (e.g., velocity of a point) for the constitutive model and project the resulting dynamic output and their contributions on the overall system equations of motion.
- The multibody formulation confers the equations a high *portability* toward other scientific disciplines which makes them easy to use for specific analyses (e.g., steady equilibrium determination, modal analysis or optimal control).
- Finally, the literature on vehicle dynamics shows the extensive use of multibody equations to analyze vehicle dynamics and implement control laws [65]. This further justifies the choice of MBS modeling for this thesis.

This chapter is divided into three main parts. Firstly, the chosen MBS formalism is introduced. The relative coordinates approach and its specificities are presented together with the corresponding formulations of the equations of motion. Secondly, the state-space representation for the MBS equations is developed

and subsequently linearized, resulting in a linear state-space representation. This linearization is of particular interest for stability analysis and feedback controller design. Finally, the 11-d.o.f. vehicle model, derived on the basis of the MBS approach, is presented.

3.1 Multibody modeling

In this section, the multibody formalism chosen to model the vehicle dynamics is discussed. First, the relative coordinates approach is briefly presented and the system topological description inherent to the latter is introduced. Then, different formulations for the equations of motion of a multibody system are presented.

3.1.1 Relative coordinates approach

Different coordinates approaches exist when it comes to MBS modeling: the absolute coordinates, the nodal absolute coordinates and the relative coordinates approaches.

In the relative coordinates approach, the configuration of a body is defined relatively to another body called its parent body. The degree of freedom (d.o.f.) is introduced between the two bodies within the corresponding joint. This principle is applied from one body to another starting from the top until reaching the inertia frame attached to the fixed body (i.e., the base); The MBS is said to have a 'tree-like structure' (Fig. 3.2.a). It is advantageous as the number of coordinates q , used to describe the system motion, is reduced compared to the absolute coordinates approach and the joint constrained motion is implicitly taken into account by the relative formulation (Fig. 3.2.b).

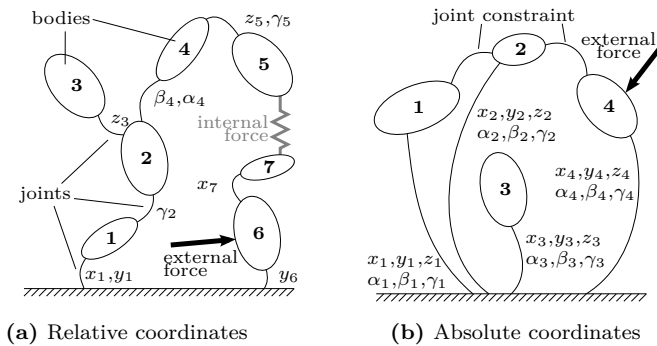


Figure 3.2: Different coordinates approaches for the multibody description (adapted from [66]).

3.1.2 Multibody system topology

In the relative coordinates approach, a multibody system is fully characterized by two types of features:

- bodies, defined by their mass, center of mass and inertia tensor,
- joints, defined by their nature and their connection on the *parent* body.

For each body, a frame is necessary in order to express dynamic quantities such as inertia tensors, center of mass locations but also to locate and characterize the joint translation or rotation. To avoid an excessive number of frames for the MBS description, the following conventions are adopted:

- One joint is associated to each body. Respecting the tree-like structure, one body i will be connected to one and only one parent body h through a joint i (Fig. 3.3). Following this rule, the number of joints n defines the size of a tree-like MBS.
- A frame, called the body-fixed frame, is attached to each body and located at its center of mass (Fig. 3.3).

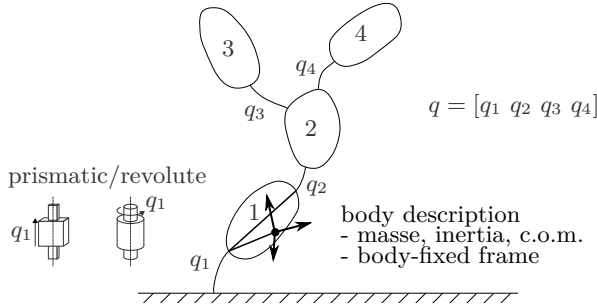


Figure 3.3: Topological description of the MBS with a relative coordinate approach.

- The axis of any (prismatic or revolute) joint must be aligned with its parent body-fixed frame (Fig. 3.4). This allows for a full kinematic description of bodies and joints. More complex joints can be modeled by combining prismatic and revolute joints (e.g., the universal joint is easily modeled via two successive revolute joints in quadrature). q^j denotes the joint relative coordinate, that is the displacement/angle within the prismatic/revolute joint j .

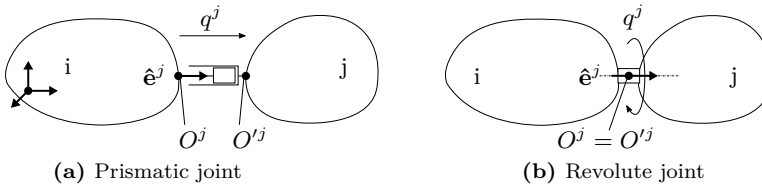


Figure 3.4: Elementary joints.

Considering only tree-like structure is quite restrictive since most of the mechanical systems (e.g. a four-bar linkage) do not present such a structure and contain loops of bodies (Fig. 3.5.a). These loops are modeled by introducing geometrical constraints between the generalized joint coordinates, q . The loop is first cut at a given joint, resulting into two independent branches of bodies (Fig. 3.5.b). Afterwards, the loop closure is restored by introducing geometrical constraints which are equivalent to the cutted joint. For instance, forcing the points of two branches to coincide would correspond to a perfect frictionless ball joint (Fig. 3.5.b). Other kinds of loop constraints can be envisaged, by cutting bodies or joints. More details on these modeling aspects can be found in [67].

Generally speaking, the n generalized coordinates associated to the tree-like structure needs to satisfy m geometrical constraints:

$$h(q) = 0 \quad [m \times 1] \quad (3.1)$$

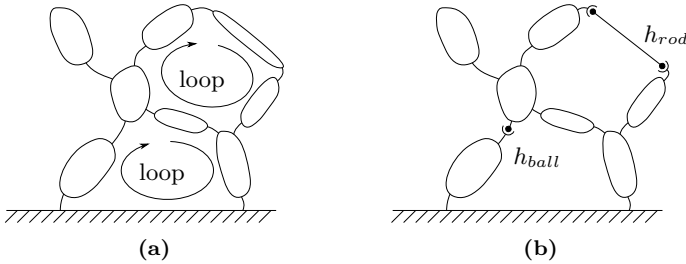


Figure 3.5: Kinematic loop modeling. (a) Mechanical systems can be composed of loops of bodies. (b) The tree-like structure is restored by cutting the system and imposing a geometrical constraint equivalent to the removed joint (adapted from [66]).

3.1.3 Equations of motion

In order to represent the dynamics of a MBS, the equations of motion can be written as a system of differential-algebraic equations (DAE) in terms of the generalized coordinates, q . First the unconstrained tree-like MBS is considered and then the constrained MBS is treated afterwards. A mathematical method for DAE index reduction, called the coordinate partitioning [68], is introduced for constrained MBS.

Unconstrained MBS equations

There are several methods to derive the equations of motion for a tree-like unconstrained system. Among them, the so-called *Newton-Euler recursive* scheme, denoted NER, is particularly suitable for relative joint coordinates [69]. This method implements two consecutive recursions on the tree-like MBS (a

forward kinematics followed by a backward dynamics). It is well-known for its easiness of computer implementation and its reduced number of operations for obtaining the equations of motion. The NER algorithm results in an expression for the vector of generalized joint forces Q as a function ϕ of the system kinematics $(q, \dot{q}, \ddot{q})$.

$$Q = \phi(\ddot{q}, \dot{q}, q) \quad [n \times 1] \quad (3.2)$$

with

- q , the generalized joint coordinates
- $\dot{q}$, the generalized joint velocities
- $\ddot{q}$, the generalized joint accelerations
- Q , the generalized joint forces (torques)
- ϕ , the function derived throughout the NER algorithm

The NER scheme has only a $\mathcal{O}(n)$ complexity thanks to its recursive formulation, n being the size of the MBS (i.e., the number of joint's d.o.f.). If this formulation is extensively used, as the so-called inverse dynamics, by roboticians to determine the actuation torques Q necessary to track a given trajectory $(q(t), \dot{q}(t), \ddot{q}(t))$ for a fully actuated robot, it can also be used to generally describe the dynamics of any MBS systems in terms of its generalized coordinates. In this case, the joint torques/forces Q are configuration-dependent (i.e., an elastic joint or one submitted to viscous friction) and are expressed, in a general sense, as a function of the generalized coordinates q and the generalized velocities $\dot{q}$.

$$\phi(\ddot{q}, \dot{q}, q) = Q(\dot{q}, q) \quad (3.3)$$

Additionally, Q can be divided into passive configuration-dependent forces (e.g., friction, spring) and active internal forces (e.g., motors).

$$Q = Q_p(\dot{q}, q) + Q_a \quad (3.4)$$

The original NER scheme leading to the formulation in equation (3.3) can be modified in order to obtain a so-called semi-explicit form of the dynamic equations. This new form still preserves the full recursiveness, even for the computation of the mass matrix [70].

$$M(q)\ddot{q} + c(q, \dot{q}) = Q(\dot{q}, q) \quad [n \times 1] \quad (3.5)$$

with

- M , the symmetric generalized mass matrix of the system,
- c , the nonlinear dynamic vector which contains the gyroscopic, centripetal and gravity terms as well as the contribution of the resultant external forces and torques.

The difference compared to equation (3.3) is that the algorithm computes M and c separately before the evaluation of the left-hand side. The computation of M in the recursive scheme requires computational efforts of $\mathcal{O}(n^2)$ complexity [70].

The system of equations (3.5) can be solved in terms of the accelerations $\ddot{q}$ via a Cholesky decomposition of the mass matrix leading to the explicit expression γ for the acceleration $\ddot{q}$ in terms of the position q and velocity $\dot{q}$. This resolution step yields an additional $\mathcal{O}(n^3)$ complexity compared to the semi-explicit form in (3.5).

$$\ddot{q} = M^{-1} [Q - c] (\dot{q}, q) \triangleq \gamma(\dot{q}, q, Q) \quad [n \times 1] \quad (3.6)$$

There exist other approaches to derive the explicit formulation from equation (3.6) via the so-called Order-N algorithms [71]. Details on these specific methods can be found in Appendix B¹. These explicit formulations are often preferred to the implicit one for time integration purposes as the explicit expression of the accelerations allows for the use of explicit integrators (e.g., the classic Runge-Kutta method, RK4).

Constrained MBS equations

Generally, the geometrical constraints $h(q)$ are nonlinear and cannot be analytically solved to express some dependent coordinates in terms of independent ones. Therefore, the Lagrange multipliers technique [67] is used and the semi-explicit form (3.5) results in the DAE system of equations (3.7-3.10) for a constrained MBS, where the first and second derivatives of the constraints must be satisfied. The generalized coordinates q are the differential variables while the Lagrange multipliers are the algebraic variables.

$$M(q)\ddot{q} + c(q, \dot{q}, g) = Q(q, \dot{q}) + J^T \lambda \quad (3.7)$$

$$h(q) = 0 \quad (3.8)$$

$$\dot{h}(q, \dot{q}) = J(q)\dot{q} = 0 \quad (3.9)$$

$$\ddot{h}(q, \dot{q}, \ddot{q}) = J(q)\ddot{q} + \dot{J}\dot{q}(q, \dot{q}) = 0 \quad (3.10)$$

with the additional terms:

- $J = \frac{\partial h}{\partial q^T}$, the constraint Jacobian matrix
- λ , the Lagrange multipliers
- h , the algebraic constraints (i.e., resulting from kinematic loops)

Coordinate partitioning

Several methods to solve and time integrate the DAE system formed by (3.7-3.10) exist. Among these, the coordinates partitioning technique [68] consists in splitting the coordinates, q into two subsets: the *independent* ones, q_u and the *dependent* ones, q_v .

$$q = \begin{pmatrix} q_u \\ q_v \end{pmatrix} \quad (3.11)$$

¹This appendix is a copy of the complete study that appeared as a book chapter in *IUTAM Symposium on Intelligent Multibody Systems – Dynamics, Control, Simulation*. published in 2019 [72].

To illustrate this concept, a simple 4-bar mechanism is modeled by defining 3 coordinates for the in-plane rotations while the fourth rotation is handled by constraining the end of the third bar on a given point of the ground in the two-dimensional plane. Subtracting the number of coordinates ($n = 3$) by the number of constraints² ($m = 2$), only 1 degree of freedom is remaining for the system and there is thus 1 independent coordinate, q_1 and two dependent coordinates, q_2 and q_3 whose values are directly related to the value of q_1 through the algebraic constraints, $h(q) = 0$.

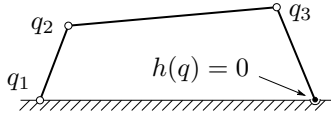


Figure 3.6: 4-bar mechanism.

$$q_u = (q_1) \quad q_v = (q_2 \ q_3)$$

$$n - m = n_{qu} = \# \text{ dll}$$

$$3 - 2 = 1$$

Via the coordinate partitioning, the initial index-3 DAE system can be reduced to an ODE (ordinary differential equations) system expressed in terms of the sole independent coordinates, q_u . This reduction is performed in two steps.

First, the differential equations can be partitioned resulting in the set of equations (3.12).

$$\begin{pmatrix} M_{uu} & M_{uv} \\ M_{vu} & M_{vv} \end{pmatrix} \begin{pmatrix} \ddot{q}_u \\ \ddot{q}_v \end{pmatrix} + \begin{pmatrix} c_u \\ c_v \end{pmatrix} = \begin{pmatrix} Q_u \\ Q_v \end{pmatrix} + \begin{pmatrix} J_u^t \\ J_v^t \end{pmatrix} \lambda \quad (3.12)$$

Using the lower part of the system (3.12), the Lagrange multipliers λ (i.e., the algebraic variables) can be eliminated from the DAE system.

Then, the linear systems (3.9) and (3.10) are partitioned accordingly and then solved in order to express the dependent velocities $\dot{q}_v$ and accelerations $\ddot{q}_v$ as functions of the independent ones.

$$\dot{q}_v = B_{vu} \dot{q}_u \quad \text{with} \quad B_{vu} \triangleq -(J_v)^{-1} J_u \quad (3.13)$$

$$\ddot{q}_v = B_{vu} \ddot{q}_u + b \quad \text{with} \quad b \triangleq -J_v^{-1} (\dot{J} \dot{q}) \quad (3.14)$$

After replacing $\dot{q}_v$ and $\ddot{q}_v$ using equation (3.13) and (3.14), the following index-1 DAE system is obtained with the dependent coordinates q_v as the new algebraic variables:

$$M_r(q_u, q_v) \ddot{q}_u + F_r(q_u, q_v, \dot{q}_u, \dot{q}_v) = 0 \quad (3.15)$$

$$h(q_u, q_v) = 0 \quad (3.16)$$

with

- M_r , the reduced mass matrix
- F_r , the reduced force vector³.

²Assumed independent.

³The expressions for M_r and F_r can be found in [73].

Finally, the dependent coordinates at position level q_v can be simply obtained by solving the remaining nonlinear constraint equations (3.16) with the help of an iterative method like the Newton-Raphson algorithm. Hence, the process leads to the following ODE system in terms of the independent positions q_u and their time derivatives $\dot{q}_u$ and $\ddot{q}_u$.

$$M_r(q_u)\ddot{q}_u + F_r(\dot{q}_u, q_u) = 0 \quad (3.17)$$

This can also be written in a so-called residual form where the term Γ_r can either be derived based on the semi-explicit formulation or, more advantageously via the implicit one derived via the original NER algorithm (3.2).

$$\Gamma_r(\ddot{q}_u, \dot{q}_u, q_u) \triangleq \phi_u - Q_u - B_{vu}^t (Q_v - \phi_v) = 0 \quad (3.18)$$

Driven variables

In various situations, some of the generalized joint coordinates q^i are no longer considered as variables because their motion is either simply *locked* (e.g., a skewed joint), or *forced* to obey a specific function of time t . In the present formalism, those variables are denoted *driven variables* and identified by the subscript c . Formally if joint i is driven, then q^i , denoted q_c^i , is such that

$$q_c^i = f(t) ; \quad \dot{q}_c^i = \dot{f}(t) ; \quad \ddot{q}_c^i = \ddot{f}(t) \quad (3.19)$$

in which the function $f(t)$ is imposed (e.g., driver steering input). Strictly speaking, equation (3.19) represents an additional constraint on the system involving the sole variable q_c^i . Its simple form enables the separate treatment of this constraint (from the set $h(q)$) in a more efficient way via a second partitioning. More details on this method can be found in [67].

Symbolic generation of equations

The main advantage of multibody modeling is the systematic generation of the equations of motion regardless of the system morphology or the application. Once the system is described as a tree-like structure (possibly with constraints), the Newton-Euler recursive scheme can be applied to compute the equations on different forms (i.e., semi-explicit or implicit). Among the possible methods to generate the multibody equations, the *symbolic* approach is a powerful tool that drastically simplifies the mathematical expressions and confer the equations a *high portability* for further treatment (e.g., time integration, equilibrium, modal analysis). For these reasons, the Robotran environment was developed at UCL-MEED and is extensively used to model multibody systems. Based on a topological description (Section 3.1.1), Robotran generates symbolic routines that express the MBS equations as functions of the coordinates and their derivatives.

For the sake of versatility, constitutive laws for internal joint forces, Q or external forces and torques being mostly problem-dependent are separately

implemented while the symbolic formulation account them automatically in the expression of the equations of motion (3.7) and (3.17). This enables the implementation of dedicated constitutive models such as tire/ground model for a vehicle, for instance. Specific symbolic computations provide kinematic input that can be used by constitutive models to express, in a given basis, forces and torques that are projected on the equations of motion.

In addition, the kinematics of any specific point of the system (e.g., the driver seat) can be symbolically generated upon request. Once the point is specified, the subroutine is generated and allows for the computation of the point kinematics in the most efficient way.

The inclusion of these specific features within the symbolic generation process allows for a precise and tailor-made modeling of the multibody system while still ensuring the exactitude of the lengthy computations (e.g., 3D projection, chain derivatives) associated with the tridimensional nature of the system motion.

3.2 State-space representation

The equations of motion derived for the NTV are used for different types of dynamic analysis, optimal control and feedback control. To this extent, the MBS equations can be reformulated as a first-order state-space model by gathering the independent generalized coordinates and velocities into a state vector, x .

$$x = \begin{pmatrix} q_u \\ \dot{q}_u \end{pmatrix} \quad (3.20)$$

The equations of motion are written as a set of nonlinear equations where u denotes the system input and y the system output.

$$\dot{x} = f(x) + g(x)u \quad (3.21)$$

$$y = h(x) \quad (3.22)$$

The function $h(x)$ expresses the system output, y as a function⁴ of the states, x . The expression f and g are derived from the semi-explicit formulation of the dynamics from the ODE system (3.17).

$$f(x) = \begin{bmatrix} \dot{q}_u \\ -M_r^{-1}F_{r,p}(\dot{q}_u, q_u) \end{bmatrix} \quad [n_x \times 1] \quad (3.23)$$

$$g(x) = \begin{bmatrix} 0 \\ M_r^{-1}N \end{bmatrix} \quad [n_x \times n_u] \quad (3.24)$$

with

- $F_{r,p}$ being the passive reduced force vector including only the passive internal forces Q_p as defined in equation (3.4)
- N being a n_{qu} by n_u logical matrix enabling the actuation of n_u of the n_{qu} d.o.f.

⁴This function, $h(x)$ must not be confused with $h(q)$, the MBS constraints equations.

Typically, the inputs are active internal forces $Q_{u,a}$ and, as the reduced force vector F_r is linear with respect to the *independent* forces Q_u , N is simply a logical matrix filled with 0 and 1.

$$Q_{u,a} = N u \Leftrightarrow u = N^t Q_{u,a} \quad (3.25)$$

The output y is generally expressed as a nonlinear function of the state $h(x)$ whose expression generally depends on the application.

3.2.1 State-space model linearization

In the present section, the process for linearizing the reduced ODE model of a MBS is described. In this thesis, linearization is a key tool for both computing the modal analysis and designing a feedback controller. The goal here is to rewrite the MBS equations as a first order state-space model (i.e., state variables, state matrix, input matrix and input vector).

First, the semi-explicit reduced form of the equations is considered:

$$M_r(q_u)\ddot{q}_u + F_r(\dot{q}_u, q_u, Q_u) = 0 \quad (3.26)$$

In which the independent joint forces Q_u have been made explicit in the F_r vector. The linearization is considered around a given configuration of the system denoted by $*$ (e.g., a steady state equilibrium):

$$(*) \triangleq (q_u^*, \dot{q}_u^*, \ddot{q}_u^*, Q_u^*) \quad (3.27)$$

In the neighborhood of this configuration, the system state is redefined considering small perturbations Δ :

$$\begin{aligned} q_u &= q_u^* + \Delta q_u \\ \dot{q}_u &= \dot{q}_u^* + \Delta \dot{q}_u \\ \ddot{q}_u &= \ddot{q}_u^* + \Delta \ddot{q}_u \\ Q_u &= Q_u^* + \Delta Q_u \end{aligned}$$

Introducing these expressions in equation (3.26) and using a first order Taylor series expansion, the equation reads:

$$\begin{aligned} M_r(q_u^* + \Delta q_u)(\ddot{q}_u^* + \Delta \ddot{q}_u) + F_r(q_u^* + \Delta q_u, \dot{q}_u^* + \Delta \dot{q}_u, Q_u^* + \Delta Q_u) &= 0 \\ \left(M_r(q_u^*) + \frac{\partial M_r}{\partial q_u} \Delta q_u \right) (\ddot{q}_u^* + \Delta \ddot{q}_u) + F_r(*) + \frac{\partial F_r}{\partial q_u} (*) \Delta q_u \\ + \frac{\partial F_r}{\partial \dot{q}_u} (*) \Delta \dot{q}_u + \frac{\partial F_r}{\partial Q_u} (*) \Delta Q_u &= 0 \end{aligned}$$

Where $(*)$ stands for a given dynamic configuration. According to the coordinate partitioning used to obtain equation (3.26), the expression of the reduced force vector leads to $\frac{\partial F_r}{\partial Q_u} = -1$. Hence by reordering the terms and noting that

equation (3.26) is verified for any configuration $(*)$, the linearized equations are written in a generic form :

$$M_r(q_u^*)\Delta\ddot{q}_u + \underbrace{\frac{\partial F_r}{\partial \dot{q}_u}(*)}_{G_r} \Delta\dot{q}_u + \underbrace{\left(\frac{\partial M_r}{\partial q_u}(*)\ddot{q}_u^* + \frac{\partial F_r}{\partial q_u}(*)\right)}_{K_r} \Delta q_u - \Delta Q_u = 0 \quad (3.28)$$

Where G_r and K_r are respectively denoting the tangent damping and the tangent stiffness matrices. These matrices are computed numerically within the Robotran environment.

To obtain a first-order linear state-space representation, the deviation vector from a given reference state, x^* is considered:

$$\Delta x = \begin{pmatrix} \Delta q_u \\ \Delta \dot{q}_u \end{pmatrix} \rightarrow \Delta \dot{x} = \begin{pmatrix} \Delta \dot{q}_u \\ \Delta \ddot{q}_u \end{pmatrix} \quad (3.29)$$

The linearized equation system (3.28) can be rewritten in terms of Δx :

$$\begin{pmatrix} I & 0 \\ 0 & M_r \end{pmatrix} \begin{pmatrix} \Delta \dot{q}_u \\ \Delta \ddot{q}_u \end{pmatrix} + \begin{pmatrix} 0 & -I \\ K_r & G_r \end{pmatrix} \begin{pmatrix} \Delta q_u \\ \Delta \dot{q}_u \end{pmatrix} = \begin{pmatrix} 0 \\ \Delta Q_u \end{pmatrix} \quad (3.30)$$

After some linear algebra manipulations, the state-space representation is obtained :

$$\underbrace{\begin{pmatrix} \Delta \dot{q}_u \\ \Delta \ddot{q}_u \end{pmatrix}}_{\Delta \dot{x}} = \underbrace{\begin{pmatrix} 0 & I \\ -M_r^{-1}K_r & -M_r^{-1}G_r \end{pmatrix}}_A \underbrace{\begin{pmatrix} \Delta q_u \\ \Delta \dot{q}_u \end{pmatrix}}_{\Delta x} + \underbrace{\begin{pmatrix} 0 \\ M_r^{-1} \end{pmatrix}}_{B_r} \Delta Q_u \quad (3.31)$$

As mentioned in the previous section, the inputs u are the active independent internal forces $Q_{u,a}$ and, as the reduced force vector F_r is linear with respect to the independent forces Q_u , the input matrix B is thus simply derived as follows:

$$\Delta Q_u = N u \rightarrow B = B_r N = \begin{pmatrix} 0 \\ M_r^{-1} N \end{pmatrix} \quad (3.32)$$

Similarly, the output function $h(x)$ is linearized and the output matrix C is obtained:

$$\Delta y = \underbrace{\frac{\partial h}{\partial x}(*)}_C \Delta x \quad (3.33)$$

3.2.2 Non-sensitive states

States whose evolution depends on but does not influence the system dynamics are called non-sensitive states. Practically it means that for such a state x_i :

$$\frac{\partial f}{\partial x_i}(*) = A_{col,i}(*) = 0 \quad (3.34)$$

If a state is non-sensitive, it can be neglected or replaced in the equilibrium analysis. If its evolution is not of interest, the associated line can simply be removed, reducing the size of the MBS model accordingly. Typical examples of such non-sensitive states for a vehicle are the wheel angular position and the vehicle location (x, y, ψ) on a perfect horizontal ground.

3.3 NTV vehicle model

3.3.1 Reference NTV morphology

The vehicle morphology and geometry used in this study correspond to a narrow three-wheeler, with a reverse-trike configuration, developed at the Université catholique de Louvain [15] (Fig. 3.1). The chosen configuration is justified by the necessity of having two wheels side-by-side to perform a tilting actuation (in green in Fig. 3.7). The rear axle unit is similar to the one of a motorbike and equipped with an electrical propulsion motor. The front wheels unit is comparable to a traditional double wishbone suspension except for the shocks top connection which is not directly attached to the chassis but to an intermediate component: an articulated T-shaped pendulum (Fig. 3.7). This mounting provides the additional rotation degree of freedom (d.o.f.) between the chassis and the suspension assembly which can either be actuated or free to rotate. The steering assembly is composed of two connecting rods connected to the front wheel carriers and put into motion by the rotation of the steering fork.

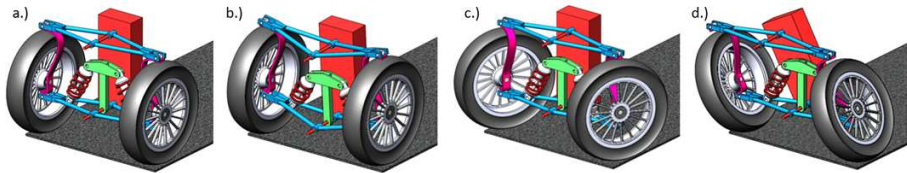


Figure 3.7: Front assembly of the NTV developed at UCLouvain [15]. In addition to the bounce (b) and steering (c) degrees of freedom characterizing a normal four-wheeled vehicle, the tilt (d) degree of freedom is specific to NTVs.

3.3.2 Multibody model

The studied NTV is modeled as a multibody system (MBS) with the help of the Robotran environment [73]. The front and rear spring/damper deflection is deliberately absent from the model to better fit with the scope of this work and limit the influence of the chosen design for the front suspension on the results. This significantly reduces the pure bounce motion of the chassis with respect to the ground (Fig. 3.7.b) and reduces the number of degrees of freedom for the vehicle. In the model, the steering wheel is omitted as the driver inputs are not

of interest in this study: the steering input is directly applied to the steering fork (Fig. 3.8). In addition, the driver motion within the vehicle can be neglected given the low position adopted for the driver seat. The model comprises a total of 19 generalized relative coordinates, among which 11 are independent and 8 are dependent. The 11 independent coordinates, corresponding to the 11 degrees of freedom of the vehicle (Fig. 3.8), are gathered in the vector q_u .

$$q_u = [q_{T1} \ q_{T2} \ q_{T3} \ q_{R3} \ q_{R1} \ q_{R2} \ q_{rr} \ q_{tilt} \ q_{lt} \ q_{rt} \ q_{steer}]^T \quad (3.35)$$

with:

- q_{T1}, x : chassis reference point position along the longitudinal inertial axis $\hat{\mathbf{I}}_1$.
- q_{T2}, y : chassis reference point position along the lateral inertial axis $\hat{\mathbf{I}}_2$.
- q_{T3}, z : chassis reference point position along the vertical inertial axis $\hat{\mathbf{I}}_3$.
- q_{R3}, ψ : chassis rotation along the vertical inertial axis $\hat{\mathbf{I}}_3$ (yaw).
- q_{R1}, ϕ : chassis rotation along the longitudinal vehicle axis $\hat{\mathbf{X}}_1$ (roll/lean).
- q_{R2}, ν : chassis rotation along the transverse vehicle axis $\hat{\mathbf{X}}_2$ (pitch).
- q_{rr} : rear wheel axial rotation.
- q_{tilt}, θ : suspension tilt rotation with respect to the chassis along its longitudinal axis.
- q_{lt} : left front wheel axial rotation.
- q_{rt} : right front wheel axial rotation .
- q_{steer}, δ_{sf} : steering fork rotation with respect to the chassis along its vertical axis.

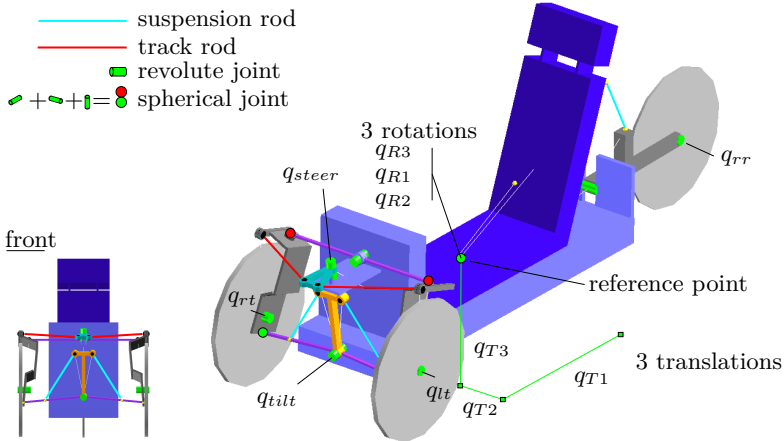
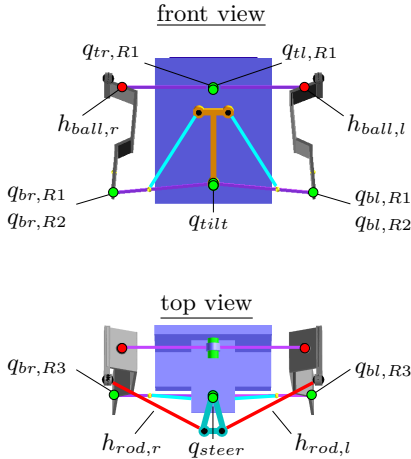


Figure 3.8: MBS model of the reference NTV: description of the 11 degrees of freedom.

A detailed characterization of the MBS model with the exact topological description and the dynamic parameters are given in [2].

Thereafter, the kinematic loops in the front assembly are presented. The double wishbone suspension and the steering assembly imply tridimensional kinematic loops which correspond to a total of 8 nonlinear algebraic constraints (3.8):

- First, the steering assembly can be studied regardless of the suspension system (Fig. 3.9 top view). The steering fork angle, q_{steer} dictates the front wheels orientation described by the two rotational joints within the suspension bottom ball joints, $q_{bl,R3}$ and $q_{br,R3}$. Mechanically, the steering fork steers the wheels through the steering rods. Mathematically, the two loops created by the steering system are handled by considering the steering rod as rigid massless body and thus replacing them by two constraint equations, $h_{rod,l}$ and $h_{rod,r}$ imposing a constant distance between the two anchor points (red lines in Fig. 3.9). Thus, q_{steer} is an independent coordinate while $q_{bl,R3}$ and $q_{br,R3}$ are dependent coordinates throughout the two rod constraints.
- The double wishbone suspension implies two loops that are handled by modeling the upper ball joints as geometrical constraints. Each ball corresponds to 3 constraints equations imposing that two different anchor points have to coincide in the tridimensional space. The lower arms are rigidly connected to the T-element and rotate together around the tilt axis. Practically, the tilt angle, q_{tilt} is the only independent coordinate as the 2 central top rotations $q_{tr,R1}$, $q_{tl,R1}$ and the 4 remaining bottom rotations, $q_{br,R1}$, $q_{br,R2}$, $q_{bl,R1}$, $q_{bl,R2}$ depend on the tilt angle throughout the 6 constraint equations associated to the ball constraints $h_{ball,l}$ and $h_{ball,r}$.



$$\begin{aligned}
 q_u &= [q_{steer} \quad q_{tilt}] \\
 q_v &= [q_{tr,R1} \quad q_{tl,R1} \\
 &\quad q_{br,R1} \quad q_{br,R2} \quad q_{br,R3} \\
 &\quad q_{bl,R1} \quad q_{bl,R2} \quad q_{bl,R3}]
 \end{aligned}$$

$$\begin{aligned}
 h &= [h_{ball} (3) \quad h_{ball} (3) \\
 &\quad h_{rod,l} (1) \quad h_{rod,r} (1)] \\
 n - m &= n_{qu} (\#dll) \\
 10 - 2(3 + 1) &= 2
 \end{aligned}$$

Figure 3.9: Front assembly.

The 8 algebraic constraints allow for the expression of the 8 dependent coordinates q_v as functions of the two independent ones q_{steer} and q_{tilt} .

3.3.3 Suspension components and motors (internal forces)

All the joints of the system are considered frictionless. Among the 11 degrees of freedom, three of them can be actuated, the steering fork rotation q_{steer} , the tilting of the T-element with respect to the chassis q_{tilt} and the rear wheel axial rotation q_{rr} (Fig. 3.8). This is modeled via input torques, respectively, Q_{steer} , Q_{tilt} and Q_{rear} which are part of the joint generalized force vector Q in equation (3.7).

The front and rear spring/damper deflection is not modeled as the components are replaced by rigid rods in the MBS model. This has been done in order to widen the scope of this work and limit the influence of the chosen design for the front suspension on the results. This modification disables the pure bounce motion of the chassis with respect to the suspension assembly.

3.3.4 Tire/ground contact model (external forces)

In this study, scooter tires are considered for NTVs as their dynamics strongly differs from conventional car tire, proper modeling is required. The contact between the tire and the ground is modeled as a set of resultant forces and torques computed according to the reduced static model of Pacejka [40]. The parameters of this model have been identified by Sharp [41] and revised by Verl  [2]. It accounts for the specific contact point location due to the tire curvature and the lowered camber thrust that characterizes scooter tires. Modeling the tire/ground contact as a set of configuration-dependent external forces and moment is necessary because of the vehicle kinematic complexity. Indeed, contact modeling via simplified *rolling without slipping* kinematic constraints would not be consistent because of the front steering assembly that inherently causes lateral slip while cornering.

The wheel/ground contact kinematic inputs and dynamic outputs are expressed in the contact frame $\{\hat{\mathbf{R}}\}$ in which $\{\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2\}$ defines the contact plane with $\hat{\mathbf{R}}_1$ tangent to the wheel circumference, $\hat{\mathbf{R}}_2$ pointing to the left and $\hat{\mathbf{R}}_3$ normal to the ground. The kinematic inputs of the tire/ground model are defined as follows (Fig. 3.10):

- α_w , the *sideslip angle* which represents the angle – measured in the contact plane $\{\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2\}$ – between $\dot{\mathbf{u}}$, the absolute velocity of the *geometrical* contact point Q , and the unit vector $\hat{\mathbf{R}}_1$ which reflects the wheel orientation (Fig. 3.10).
- κ_w , the *longitudinal slip velocity* which is the projection onto $\hat{\mathbf{R}}_1$ of the velocity of the *material* point of contact of the wheel coinciding with the geometrical contact point, Q (Fig. 3.10).
- ϕ_w , the *camber angle* which represents the angle between the wheel plane and the normal vector to the ground $\hat{\mathbf{n}}_g$ (Fig. 3.10).

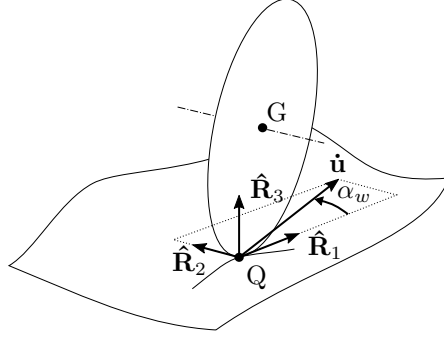


Figure 3.10: The wheel kinematic inputs are defined with respect to the contact frame $\{\hat{\mathbf{R}}\}$ (adapted from [67]).

The dynamic outputs for the tire/ground model are defined as follows (Fig. 3.11):

- F_{long} , the longitudinal force applied by the ground on the wheel in the forward direction $\hat{\mathbf{R}}_1$.
- F_{lat} , the lateral force applied by the ground on the wheel in the transverse direction $\hat{\mathbf{R}}_2$.
- M_{yaw} , the spin torque applied by the ground on the wheel around the normal vector $\hat{\mathbf{n}}_g = \hat{\mathbf{R}}_3$.

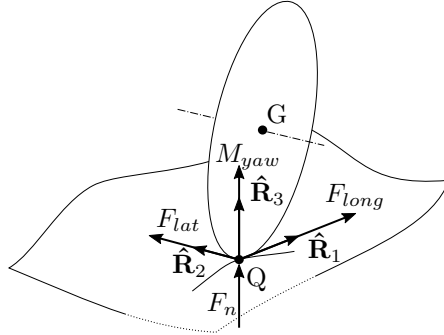


Figure 3.11: The contact forces and torques are defined in the contact frame $\{\hat{\mathbf{R}}\}$ (adapted from [67]).

The tire/ground model efforts are computed throughout a series of steps:

1. The derivation of the kinematic inputs is not straightforward and is performed based on the wheel kinematics, which is a function of the state variables that is provided symbolically by Robotran.
2. Based on the wheel kinematics, the normal force can be computed using a simplified constitutive model composed of a stiffness term based on the tire deflection e , and a damping term based on the tire deflection rate

$\dot{e}$, both computed at the contact point, Q in the normal direction to the ground $\hat{\mathbf{R}}_3$ (Fig. 3.12).

$$F_n = Ke + D\dot{e} \quad (3.36)$$

With $K = 200 \text{ kN/m}$ and $D = 10 \text{ kN/(m/s)}$ denoting the tire stiffness and the tire damping coefficient respectively.

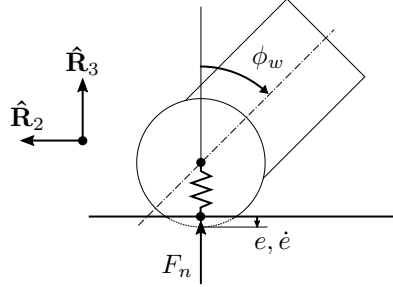


Figure 3.12: Two dimensional tire/ground contact geometry with vertical deflection e and deflection rate $\dot{e}$. Conversely to what is represented in Figures 3.8 and 3.10, the tire is not modeled as a disc and its transverse section is properly taken into account while determining the contact point.

- Once the wheel kinematics and the normal force are computed, they are fed to the contact model which outputs the longitudinal force F_{long} , the lateral force F_{lat} and the spin moment M_z (Fig. 3.13).

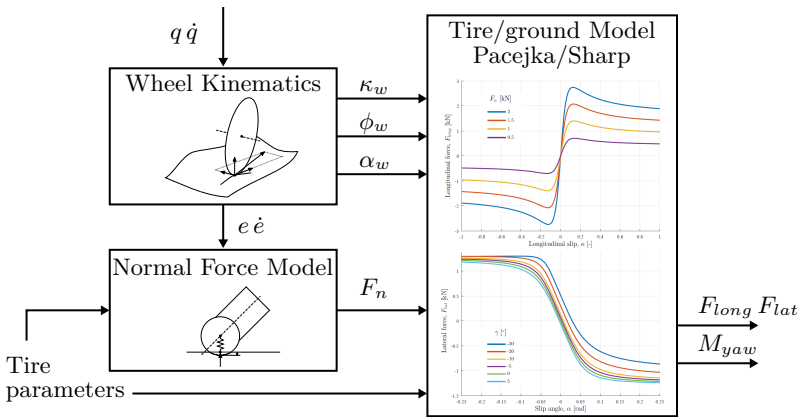


Figure 3.13: General scheme for the computation of the tire/ground contact forces within a MBS framework.

3.4 Concluding remarks

In this chapter, the model of a reference NTV obtained via the MBS formalism has been introduced. First, the MBS formalism chosen has been presented together with some motivation for such a framework. Then, the equations of motion have been written in a state-space form and subsequently linearized. This linearization will be of particular interest for stability analysis and feedback controller design. Finally, the NTV morphology and the MBS model have been presented. The model features a total of 11 degrees of freedom as the suspension spring and damper deflection is deliberately neglected. Specific multibody treatments are considered for the front suspension and the steering assembly. Finally, the model used to represent the tire/ground contact is detailed together with the required wheel kinematic inputs.

It is important to specify that the model assumes that the ground is perfectly flat and horizontal. Such assumption could easily be alleviated within the presented multibody framework [67]. Nonetheless, it was decided to consider a flat and horizontal ground in our study, as the handling of the road bank and irregularities for NTVs is a real challenge in itself [74].

Vehicle dynamic analysis

4

Many projects have focused on the development of *active* controller for NTVs. More recently, the serial actuation through the tilting action (FTC) has been studied [33]. The projects often focus on one of the serial actions or the other (i.e., FTC or FTR), however, no thorough dynamic analysis of both modes have been conducted. Hence, a proper comparison of these two modes and of their dynamics is relevant as it could reveal whether a tilting action (FTC) could compete with the more traditional steering action (FTR) to control the NTV tilting motion. This chapter aims at comparing the two aforementioned modes in terms of their influences and effects on the NTV dynamics. Firstly, straight line and steady state cornering equilibrium are considered. The method used to compute the steady state equilibrium based on the MBS equations is first explained. Then, results in the form of speed/lateral acceleration maps are obtained and used to compare both modes. Secondly, a stability analysis is carried out both in straight line and in steady state cornering. By performing a series of modal analyses around the previously obtained steady state equilibrium, stability and instability velocity range are found. In addition, the effects of the turning radius on lateral modes and their stability is analyzed for both actuation modes. Finally, transient maneuvers are studied by time-integrating the equations of motion. A curve entry is simulated and a set of dynamic quantities are analyzed in order to compare the vehicle behavior for a tilting action (FTC) or a steering action (FTR).

This chapter is divided into three main parts, each corresponding to a specific analysis. First, the vehicle stability in straight line is studied. Then, the cornering ability for the two envisaged modes is analyzed for a series of given steady state cornering equilibrium. The influence of the curve radius on the vehicle stability is observed and discussed for both modes. Finally, the transient behavior during a curve entry is simulated and compared for the two modes. The fact that no control is deliberately considered in this chapter restricts the maneuver complexity to a simple step response.

4.1 Steady state equilibrium

Generally, an equilibrium is a configuration for which forces and torques are in balance with no absolute acceleration. If a static equilibrium is considered for a MBS, the joint velocities and accelerations are zero (i.e., $\dot{q}_u = 0$ and $\ddot{q}_u = 0$). The reduced forms of the equations of motion (3.17) and (3.18) transform in a set of nonlinear algebraic equations in terms of the independent coordinates, q_u :

$$\Gamma_r(0, 0, q_u) = F_r(q_u) = 0 \quad (4.1)$$

In the case of a left/right symmetric vehicle, the static equilibrium corresponds to the standstill configuration with a given vertical position for the vehicle reference point (Fig. 3.8) and a given pitch angle for the chassis.

Equation (4.1) is solved for the independent coordinates, q_u via a Newton-Raphson algorithm with a tolerance of 1e-6 for the residue.

4.1.1 MBS equilibrium equations

Straight line equilibrium

Some of the independent coordinates, q_u , may be non-sensitive in the sense that they do not influence the equilibrium equations (i.e., the so-called residual term Γ_r in equation (4.1)). If needed, they can be omitted or replaced by other variables, according to the nature of the envisaged equilibrium (Section 3.2.2).

For a road vehicle, on a flat ground, the vehicle x and y positions, the yaw orientation, ψ and the wheels angular positions are non-sensitive variables and must be disregarded in the static equilibrium process. The *straight line* equilibrium can be obtained for a desired constant longitudinal speed $\dot{q}_{T1} = V$ (Fig. 3.8). The non-sensitive angular positions of the wheels (q_{lt} , q_{rt} and q_{rr}) are replaced by their axial rotational speeds ($\dot{q}_{lt}$, $\dot{q}_{rt}$ and $\dot{q}_{rr}$) in the set of unknown equilibrium variables, provided that a longitudinal contact force is considered in the model (Section 3.3.4). In addition, the longitudinal position q_{T1} (also non-sensitive) can be replaced by the rear wheel torque, Q_{rr} necessary to overcome the tire/ground friction at the desired constant speed V .

This ensures an equal number of 11 equilibrium equations and 11 variables:

$$q_{lt}, q_{rt}, q_{rr} \rightarrow \dot{q}_{rr}, \dot{q}_{lt}, \dot{q}_{rt}$$

$$q_{T1} \rightarrow Q_{rr}$$

$$\text{For } \dot{q}_{T1} = V,$$

$$\text{Solve } \Gamma_r(\dot{q}_{rr}, \dot{q}_{lt}, \dot{q}_{rt}, Q_{rr}, \text{remaining } q_u) = 0$$

Solving the equilibrium equations, a straight line (sl) configuration is found in terms of the generalized coordinates, velocities and joint forces/torques: $(q_u^{sl}, \dot{q}_u^{sl}, Q_u^{sl})$.

Steady state cornering

Apart from the straight line equilibrium, a vehicle can also turn at a constant speed which is often referred as steady state cornering. In that case, the vehicle is in a dynamic equilibrium, as the lateral forces cause a non-zero centripetal acceleration towards the center of the curve (Fig. 4.1). Equations (3.18) can be used to determine the steady state cornering equilibrium. To do so, a constant non-zero acceleration is set along the lateral translational d.o.f., $\ddot{q}_{T2} = a_c$. Its value has to be consistent with the desired tangent speed, V and the desired yaw rate, $\dot{\psi}$ or a desired cornering radius, R^1 .

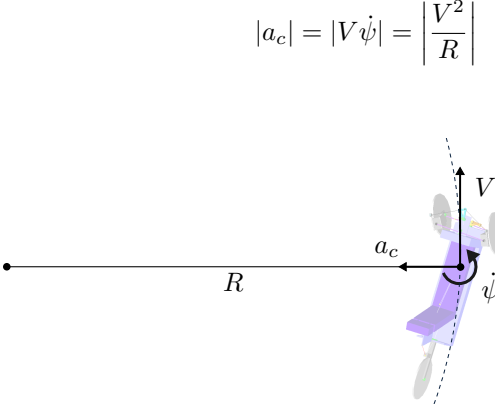


Figure 4.1: Representation of the steady state cornering equilibrium for the NTV.

To each steady state cornering (ssc) equilibrium corresponds a rear wheel torque Q_{rr} , necessary to balance the tire/ground friction and a steering torque Q_{steer} (FTR) or a tilting torque Q_{tilt} (FTC), necessary to hold the front wheels steered and to ensure the cornering at a constant radius, R . Non-sensitive variables are replaced in the equilibrium procedure as follows:

$$q_{lt}, q_{rt}, q_{rr} \rightarrow \dot{q}_{rr}, \dot{q}_{lt}, \dot{q}_{rt}$$

$$q_{T1} \rightarrow Q_{rr}$$

$$q_{T2} \rightarrow Q_{steer} \text{ or } Q_{tilt}$$

$$\text{For } \dot{q}_{T1} = V \text{ and } \dot{q}_{R3} = \dot{\psi},$$

$$\text{Solve } \Gamma_r(\dot{q}_{rr}, \dot{q}_{lt}, \dot{q}_{rt}, Q_{rr}, Q_{steer} \text{ or } Q_{tilt}, \text{remaining } q_u) = 0$$

Solving the equilibrium equations, a steady state cornering (ssc) configuration is found in terms of the generalized coordinates, velocities and joint forces/torques: $(q_u^{ssc}, \dot{q}_u^{ssc}, Q_u^{ssc})$.

¹Note that, given the model description in Chapter 3, the radius R is taken at the chassis reference point which is not the vehicle overall c.o.m.

4.1.2 Steady state map

While the straight line equilibrium corresponds to the dynamic behavior of the unmanned NTV, steady state cornering equilibrium should be more insightful for the modes comparison as it implies the use of one or the other to keep the vehicle in the curve. The stability for steady state cornering is considered by performing a series of steady state cornering equilibria (Section 4.1.1), each one being characterized by a pair of tangent velocity and curve radius (V, R). Their magnitudes are limited by the amount of lateral force that the tire/ground contact can provide to avoid non-physical equilibrium. In this regard, the centripetal acceleration at the reference point is limited to 7 m/s^2 for a speed range of 0 to 15 m/s and a decreasing radius range from 75 to 5 m.

For the sake of legibility, the results are observed in a lateral acceleration/speed map, as illustrated in Figure 4.2 for the curve radius. For this representation, the minimal and maximal radius draw two curves that delimit the area of interest, for which the steady-state equilibrium can be computed. Each cornering equilibrium is achieved through the application of a constant propulsion torque, Q_{prop} on the rear wheel to prevent the vehicle from decelerating coupled with either a steering torque Q_{steer} or a tilting torque Q_{tilt} . These two torques respectively enable the free-to-roll (FTR) mode and the free-to-caster (FTC) mode discussed in the introduction (Fig. 1.14). The differences between the two modes are studied.

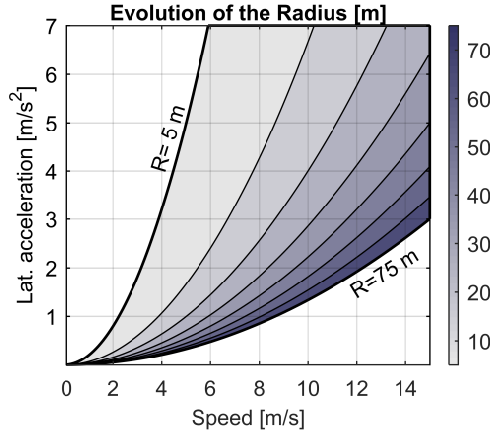


Figure 4.2: Evolution of the radius in a lateral acceleration/speed map. The lower radius ($R=5 \text{ m}$) and upper radius ($R=75 \text{ m}$) define the limits of the area of interest.

Actuation and propulsion torque

In order to quantify and compare the steering and tilting actuations in steady state conditions, the joint torque is used. As the range of motion for the two actuated joints is similar (i.e., ± 45 degrees), the torque is a good image of the energy consumption and the necessary installed power for actuation. The two

actuation modes lead to significantly different values for the actuation torques, Q . The tilting action (FTC) leads to a larger actuation torque (up to 20 times higher) than the more conventional steering action (FTR) for the whole range of steady state cornering equilibrium (Fig. 4.3). Both modes show a larger actuation when the radius decreases; however the relation is not linear. In both cases, the torque increases with higher lateral accelerations and for a given acceleration, the actuation torque decreases when the speed and radius are increased. However, the torque magnitude is clearly greater for the tilting action with value of around 250 Nm while the steering action is one order of magnitude smaller with a value of around 15 Nm for the highest lateral acceleration.

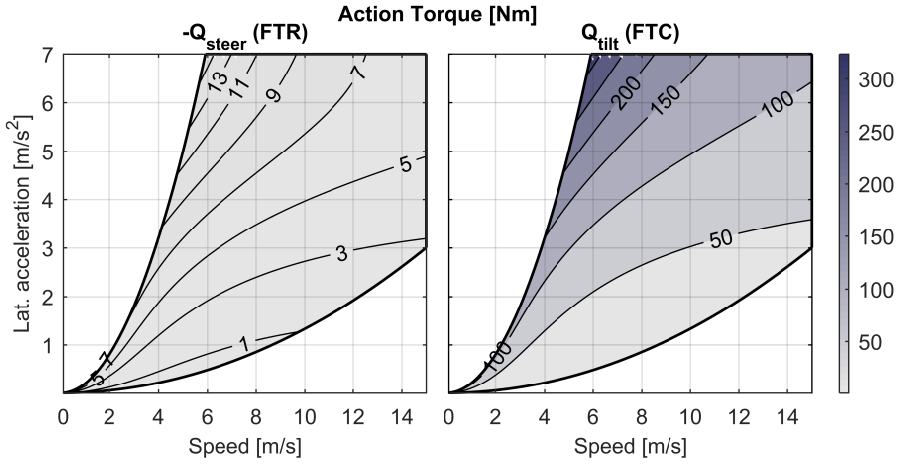


Figure 4.3: Map of the action torque for the two considered actions. The tilting action (FTC) leads to a significantly larger torque than the steering action (FTR).

The actuation torque is higher for the FTC mode because it implies a tilting torque along the vehicle longitudinal axis between the chassis and the suspension assembly. This additional torque maintains the chassis into a given lean angle and counteracts the rollover moment caused by the centrifugal effect as the vehicle is in a more upright position (Fig. 4.4 right).

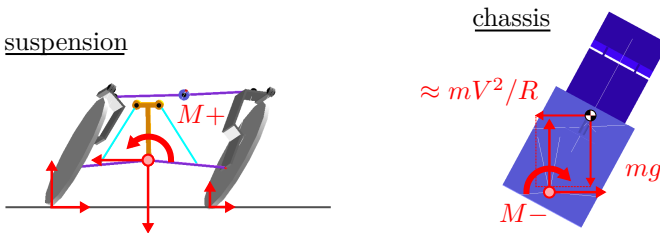


Figure 4.4: Simplified free body diagram for the chassis and the suspension assembly for the FTC mode with m the mass of the chassis.

This rollover moment is strongly related to the distance between the tilting hinge and the center of mass. A larger distance will lead to a larger tilting torque in FTC mode. For the FTR, only the steering assembly is actuated and the chassis is kept at a given lean angle which is aligned with the gravitational and centrifugal resultant. In that case, there is no rollover moment caused by the centrifugal effect and no additional tilting torque needed.

Note that the propulsion torque also differs for the two considered modes. The FTR mode requires a slightly higher rear torque than the FTC mode for the same rotational speed. This is due to a higher energy dissipation at the tire/ground interface as the vehicle is more tilted for the FTR mode. For both modes, the magnitude is more or less proportional to the lateral acceleration (Fig. 4.5).

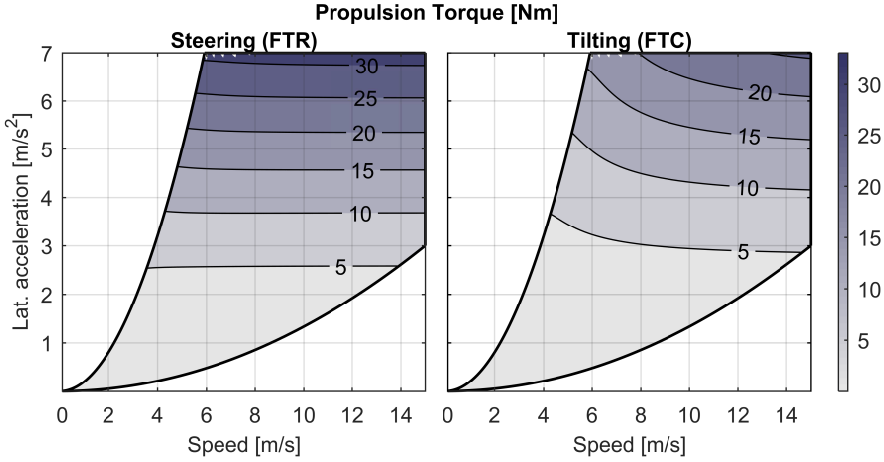


Figure 4.5: Map of the propulsion torque for the two considered actions. The FTR mode leads to a larger torque than the FTC mode.

Lean angle

The two actuation modes lead to different lean angles for the considered speeds and radius (Figs. 4.6). The steering actuation (FTR) leads to a larger lean angle than the tilting action (FTC). This difference is due to the inner joint torque, inherent to the FTC mode, applied along the tilt axis between the chassis and the suspension. This inner torque amounts to a positive action torque M_+ applied by the chassis on the suspension assembly and a negative reaction torque M_- applied by the suspension to the chassis (Fig. 4.4). If a roll balance is considered on the chassis alone, this resistive torque M_- would counteract the rollover moment and allows the vehicle to settle at a more upright position (Fig. 4.7.b) for which the centrifugal effect leads to a higher rollover than if the vehicle was aligned with the centrifugal/gravitational resultant (as it is the case for FTR). In conclusion, NTV with FTC exhibits a roll balance that is very similar to what is observed with a conventional car for which the suspension assembly provides roll stiffness through the spring and damper deflections while the chassis rolls.

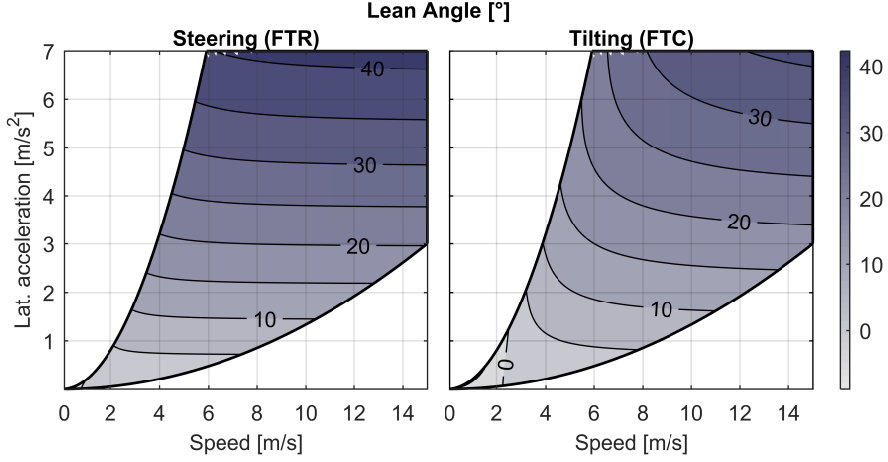


Figure 4.6: Map of the lean angle for the two considered actions. The tilting action leads to a more upright chassis position than the steering action.

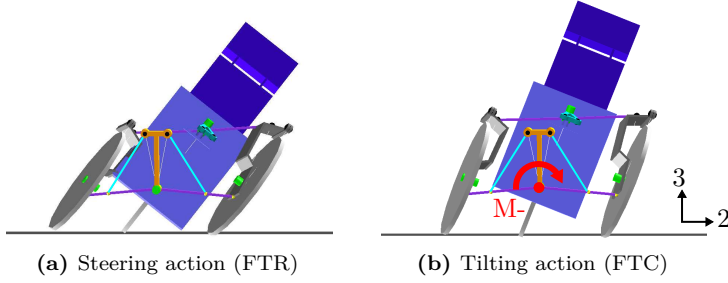


Figure 4.7: Front view of the vehicle lean angle ($V=8$ m/s and $R=10$ m).

Hence, for a given lateral acceleration, the lean angle is not the same for the two actuation modes. By comparison, for a single-track vehicle (STV), for which only the steering action is relevant, its lean angle for a given lateral acceleration reads:

$$\theta = \tan^{-1} \left(\frac{a_c}{g} \right) \quad (4.2)$$

As a matter of fact, many active tilt controllers have been developed for NTV based on this formula. However, this approach is far from ideal as the lean angle provided by the formula diverges from the steady-state angle computed for the FTR case (Fig. 4.8). The difference between the steering actions of the NTV and the STV is due to the two side-by-side wheels of the NTV which inherently cause a wheel load transfer while cornering. Note that at very low acceleration, the tilt actuation leads to an equilibrium with an opposite lean angle. This is because the tilting torque for FTC can maintain the chassis at any lean angles around the upright position.

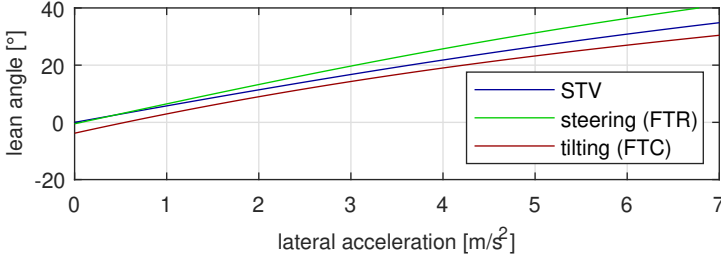


Figure 4.8: Comparison of the lean angle ($R=10$ m). The theoretical lean angle (STV) does not fit the lean angle neither for FTR nor FTC.

Tire/ground contact Forces

The two types of action lead to significant differences in the tire/ground contact forces. For the tilting action, the torque applied on the T-element leads to a greater wheel load transfer, μ , from the inner wheel to the outer wheel, this quantity being defined as the inner/outer difference in normal forces over their sum as in equation (4.3).

$$\mu = \frac{\Delta F_n}{\Sigma F_n} = \frac{F_{n,outer} - F_{n,inner}}{F_{n,outer} + F_{n,inner}} \quad (4.3)$$

For the steering action, the load transfer is smaller as no extra torque is applied to the suspension system (Fig. 4.9). Regarding the lateral forces on each wheel, differences appear again with the tilt action leading to a higher inner/outer difference with the inner wheel, generating greater lateral forces especially at high lateral acceleration (Fig. 4.10). This is due to the greater load transfer intrinsic to the FTC mode.

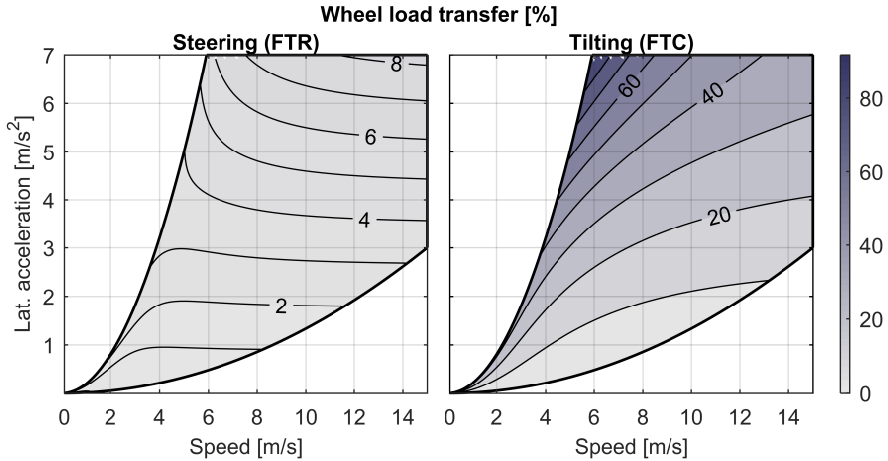


Figure 4.9: Map of the front wheels load transfer. The tilting action leads to a load transfer about 10 times larger than the steering action.

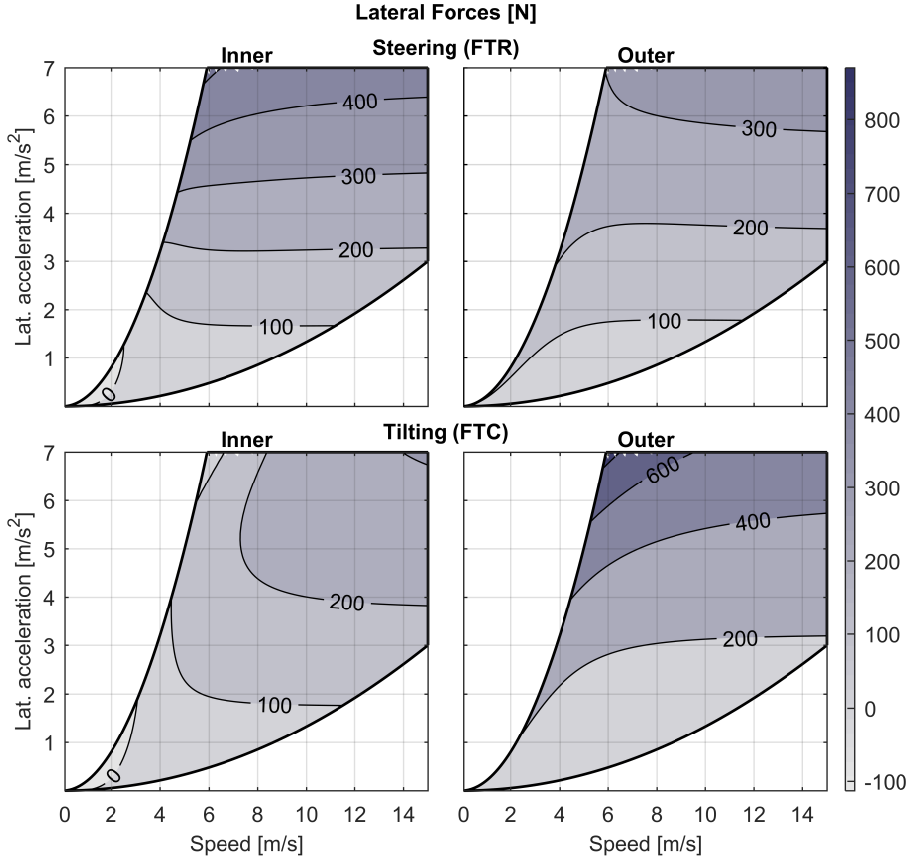


Figure 4.10: Map of the front wheels lateral forces for the two considered actions. The tilting action leads to a higher inner/outer difference in lateral forces than the steering action.

The lateral forces, F_{lat} mainly depend on the normal forces, F_n , the camber thrust, ϕ_w and the lateral slip angle, α_w , the later being responsible for tire wear. Given the chosen axis system and the left turn considered, a negative slip angle contributes to a lateral force directed toward the inside of the turn.

The slip angles profiles on the inner and outer wheel strongly differ for the two modes of actuations (Fig. 4.11). For the steering action, the outer wheel is understeered (positive slip angle) resulting in a negative tire force directed outward of the curve. A positive lateral force is still produced because of the camber thrust due to the high vehicle lean angle (Fig. 4.6). On the other hand, the tilt action leads to a negative slip angle on both wheels with higher absolute value (i.e., higher than one degree). This could be explained by the lower camber thrust for this action (due to the smaller lean angle) which necessitates a larger amount of slip angle in order to generate the necessary centripetal force to follow the curve. Additionally, it is important to mention that the

steering assembly geometry strongly affects the slip angle profiles. For the given model, the steering assembly is not particularly optimized which limits the interpretation of the slip angle profiles for high lean angle. Nevertheless, the small camber thrust arising from the more upright position should always lead to higher slip angles for the FTC case than for the FTR case. This is of great significance for the tire wear and the general vehicle stability.

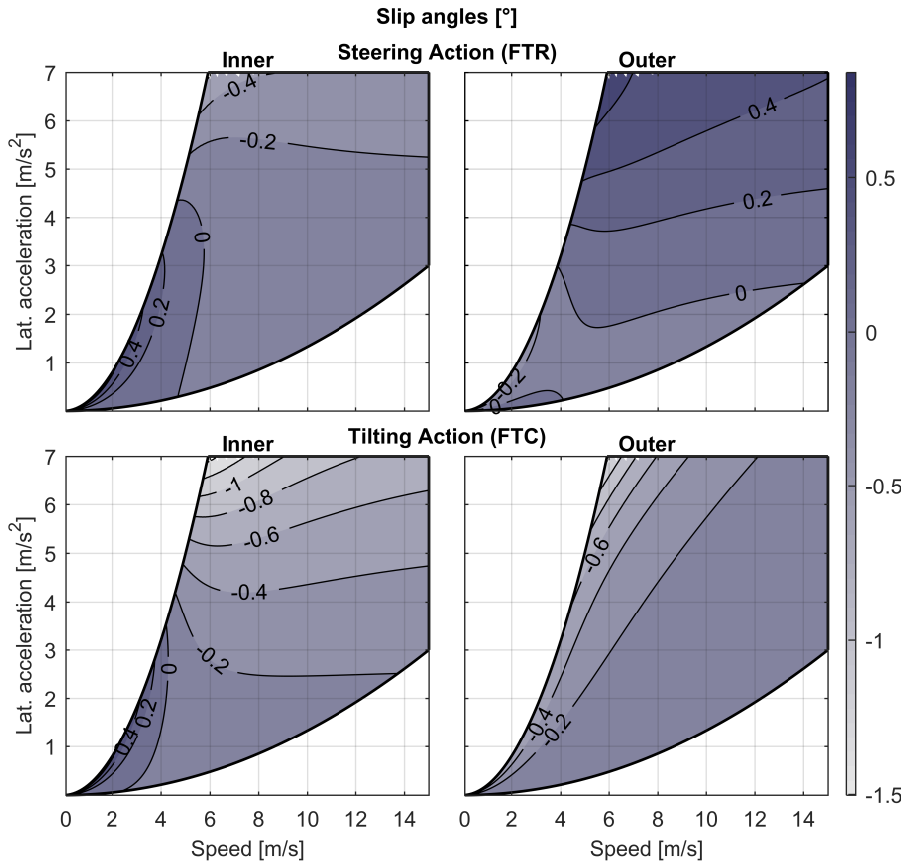


Figure 4.11: Map of the front wheels slip angles for the two considered actions. The tilting action (FTC) leads to higher slip angles in magnitude than the steering action (FTR).

Perceived acceleration

The perceived acceleration can be very insightful to compare the two tilting modes. The perceived *transverse* acceleration is defined as the component of the total *felt* acceleration of a point *P* in the chassis transverse direction (Fig. 2.2). The total felt acceleration is defined as the sum of the lateral acceleration and the roll-induced acceleration compared to the contribution of the gravity vector

(Section 2.2.1). For a simplified single-track model, the perceived acceleration at a point P , located at a distance d from the ground, is defined as follows:

$$a_{per,2} = a_{lat} \cos \theta - g \sin \theta + d\ddot{\theta} \quad (4.4)$$

This definition together with the simplified expression for the steady-state lean angle (4.2), implies that the perceived acceleration cancels out for steady state cornering.

In a generic tridimensional context, the perceived acceleration is obtained as a function of the generalized coordinates, q and their time derivatives. First, the absolute acceleration of a given point P is obtained symbolically in the inertial frame as function of the generalized coordinates q and their derivatives.

$$\mathbf{a}^P = [\hat{\mathbf{I}}]^t a_I^P(q, \dot{q}, \ddot{q}) \quad (4.5)$$

To obtain the perceived acceleration *vector*, the opposite of the gravitational vector, $\mathbf{g}$ is added to the absolute acceleration vector as the accelerometer reading of the gravity is always positive in the upward direction.

$$\mathbf{a}_{per}^P = \mathbf{a}^P - \mathbf{g} \quad (4.6)$$

with

$$\mathbf{g} = [\hat{\mathbf{I}}]^t \begin{pmatrix} 0 \\ 0 \\ -9.81 \end{pmatrix} \quad (4.7)$$

The perceived acceleration, as defined in the literature [11], is obtained by projecting the perceived acceleration vector on the axis $\hat{\mathbf{X}}_2$ attached to the vehicle chassis and always pointing toward its transverse direction (Fig. 4.12). The projection is done by using the rotation matrix between the chassis and the inertial frame, R_I^X which can also be obtained symbolically as a function of q .

$$a_{per,2}^P \triangleq (\mathbf{a}^P - \mathbf{g}) \cdot \hat{\mathbf{X}}_2 \quad (4.8)$$

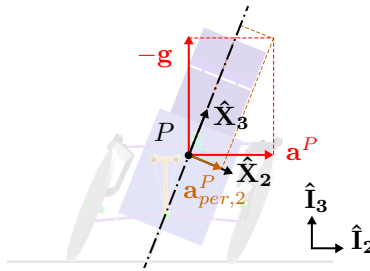


Figure 4.12: The perceived acceleration is directed in the transverse direction along the axis $\hat{\mathbf{X}}_2$.

Similarly to the previous section, steady state cornering maps for the perceived acceleration are computed for both modes (Fig. 4.13). The profile of the perceived acceleration differs strongly for the two modes. For the studied vehicle, the FTR leads to a negative acceleration corresponding to a *centripetal falling* feeling for the driver. For the FTC, positive perceived accelerations are observed corresponding to a *centrifugal uprighing* feeling for the driver. In addition, the FTR map shows a quasi-linear dependency of the perceived acceleration with respect to the lateral one. Conversely, the perceived acceleration in FTC seems to be strongly related to the turning radius with the highest values occurring at the lowest curving radius.

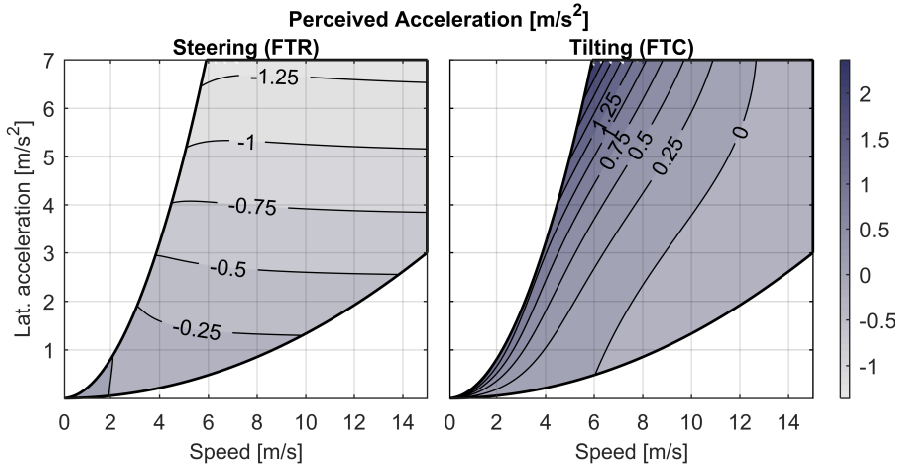


Figure 4.13: Perceived acceleration at point P (driver's head) for both action modes. The FTR leads to a negative perceived acceleration (centripetal feeling) while the FTC has a positive perceived acceleration (centrifugal feeling). The order of amplitude is quite similar.

A centrifugal effect directed outward of the curve is what a driver would normally feel on a conventional car. The driving experience in a FTC-driven NTV would be closer to what car drivers already experience in everyday life. The most important observation is that the perceived acceleration is far from zero on the overall map for both the FTR and FTC mode. This is due to the wheel load transfer inherent to the NTV front geometry which causes a different lean angle than the theoretical formula given by equation (4.2) for single-track vehicles. If the suspension geometry is based on a parallelogram mechanism (e.g., like the Piaggio scooter [75]), a lesser amount of wheel transfer should occur than for an independent wheel suspension such as a double-wishbone or a Macpherson suspension. Nevertheless, the tridimensional nature of the steady state cornering condition will always lead to some wheel load transfer in the FTR case. In fact, the geometry used in this study is quite close to the parallelogram mechanism; nonetheless the perceived accelerations are far from zero on the overall steady state cornering range. This result suggests that the cancellation of the perceived

acceleration as done in [53, 29] might not be adequate for the control of real NTV featuring complex suspensions. Whereas their simplified model predicts a zero perceived acceleration for the overall steady state cornering map, the proper modeling of the NTV dynamics shows the opposite.

4.2 Stability analysis

Once an equilibrium configuration is found, it is possible to study the linear stability of the NTV around this configuration. By performing such analysis on the overall speed and turning radius range, stability regions can be determined for the straight line case and the steady state cornering case.

4.2.1 MBS modal analysis

For single-track vehicles and tilting NTVs, the modal analysis is a useful tool as it reveals the stable or unstable behavior of the vehicle, as well as the way specific parameters can influence this behavior. Among those parameters, the traveling speed is of special interest as it prevails for the auto-stability analysis (i.e., the stability of an unmanned vehicle).

For the case of a steady state equilibrium with constant input, the eigenvalue problem corresponding to equation (4.9) can be solved via a modal analysis.

$$\dot{x} = Ax \quad (4.9)$$

The matrix A is obtained by linearizing the equations of motion around an equilibrium configuration $(q_u^{eq}, \dot{q}_u^{eq}, Q_u^{eq})$ by considering small perturbations (Section 3.2.1).

Considering the current vehicle, since its MBS model comprises 11 independent coordinates (i.e., degrees of freedom), a total of 22 eigenvalues – real or complex conjugates – are computed via the modal analysis.

4.2.2 Stability analysis in straight line

Despite its 3 wheels, the reference NTV is unstable at a standstill simply because the frictionless tilting joint allows for the chassis to fall laterally like a traditional single-track vehicle (STV) would do. Nevertheless, the NTV could exhibit the same dynamic stability that characterizes STV and thus be self-stable above a certain advance velocity. This can be studied by performing a series of modal analyses at different forward speeds, V from 0 to 15 m/s, which corresponds to an urban use of the vehicle.

Among the eigenmodes, computed based on the model (Section 3.3), only the lateral motion modes are of interest as they dictate the stability of the vehicle. The evolution of the eigenvalues real part for those modes can be observed as a function of the vehicle speed, V (Fig. 4.14).

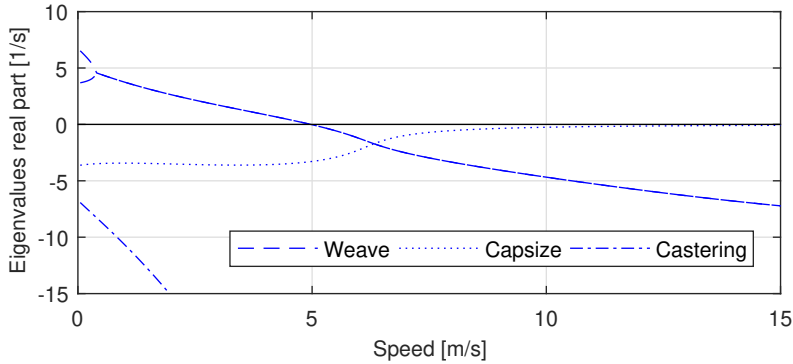


Figure 4.14: Stability analysis of the NTV in straight line. The vehicle is self-stable above a speed of 5.0 m/s.

Three main modes dictate the NTV lateral stability: the castering, the weave and the capsize. The capsize mode corresponds to a chassis motion dominated by lean: when unstable, a capsizing bicycle leans progressively into a tightening spiral with steer and lean both increasing proportionally as it falls over [43]. The weave mode is an oscillatory motion in which the vehicle steers sinuously about the headed direction with a slight phase lag relative to leaning. Finally, the castering mode is dominated by the motion of the steering assembly which tends to realign itself with the vehicle chassis like the straightening of a swivel wheel under the front of a grocery cart [43].

At nearby-zero speeds (i.e., $V < 0.4$ m/s), there are two real positive eigenvalues which correspond to an inverted-pendulum-like falling of the NTV (Fig. 4.14). When speed is increased, the two positive real eigenvalues merge to form a complex conjugate pair; this is where the oscillatory weave motion emerges. At first, the weave mode is unstable at low speed but becomes stable above a given speed of 5 m/s (Fig. 4.14). The castering mode is stable on the overall speed range while the capsize tends to an unstable behavior at higher speed. This is not an issue in practice as the monotonic lean motion associated to the mode is slow and can easily be controlled by the driver [43]. Both the castering and the capsize are non-oscillating modes as their eigenvalues are real.

The critical speed for the self-stability of the vehicle is 5.0 m/s (Fig. 4.14). It is important to note that these results concern only the studied NTV (Fig. 3.7 and 3.8) and thus cannot be directly widened to all the NTVs as the geometry and the mass distribution strongly influence the stability behavior. For example, Edelmann observed a different stability behavior with respect to speed for the NTV prototype developed at TU Wien [26].

4.2.3 Stability analysis in steady state cornering

The stability analysis around the straight line equilibrium can also be performed for steady state cornering configurations. In that case, both the steering and tilting actions are of interest. The vehicle is unmanned and does not feature any controller, a constant steering or tilting torque must be applied to keep the vehicle in a given steady state cornering configuration (Fig. 4.3). This unmanned stability analysis reflects the vehicle driveability in curves. Through a series of modal analyses, the stability is studied for the overall speed range and two different radii: 75 and 5 meters (Fig. 4.15 and 4.16). Results are compared with the straight line stability for which the vehicle was free from any actuation. For the 75-meter radius, the stability is generally similar to the straight line case; however some differences can be observed. The FTC leads to a slightly more stable weave at high speed and slightly less stable weave at low speed. On the other hand, the castering is generally less stable for FTC. Regarding the FTR, it differs from the free case at high speed as the weave mode is a bit less stable while the capsize mode is a little bit more (Fig. 4.15).

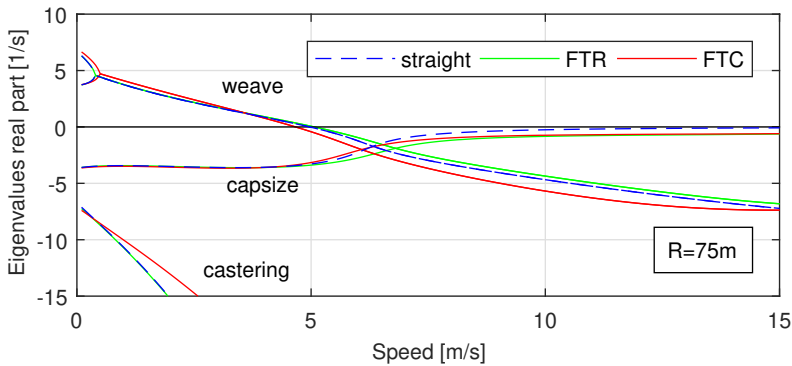


Figure 4.15: Stability analysis for a 75-meter radius. The vehicle behaves similarly to straight line.

For the 5-meter radius, the stability is greatly affected over the complete speed range (Fig. 4.16). The weave mode is unstable especially when the speed increases. Note that 5.9 m/s is the maximum speed given the 5-meter radius and the centripetal acceleration limitation. In addition, the weave mode starts to oscillate from a higher speed as the two branches merge later on to form the complex conjugate oscillating weave mode (around 1.2 m/s instead of 0.5 m/s in straight line). This shows that, at low speeds, the vehicle might be harder to control in curves than in straight line as the weave is monotonically unstable. The castering is also less stable when the curve radius is reduced and this for both the tilting and steering action. Regarding the stability behavior for the in-between radius (from 75 to 5 meters), the instability generally increases when the radius decreases. However, this tendency is far from linear as the instability increases drastically only when the radius become really small, i.e., below 10 meters (Fig. 4.17).

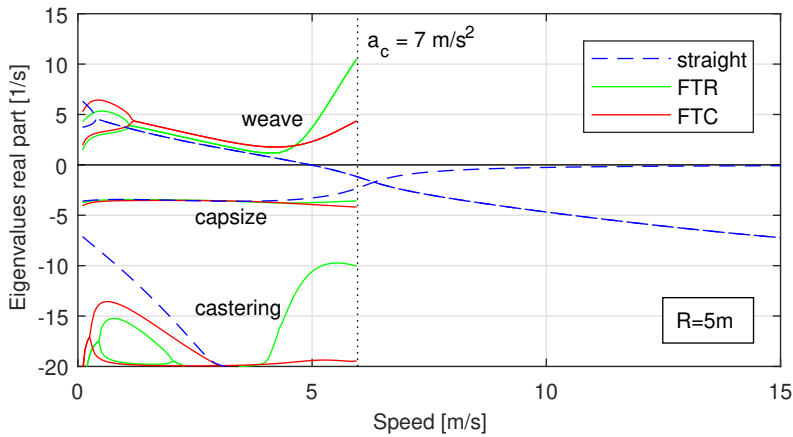


Figure 4.16: Stability analysis for a 5-meter radius. The castering and weave modes show less stable behavior especially when a steering action is used to keep the vehicle in the curve.

The capsize mode seems to be a bit more stable than for straight line motion and this for both the FTR and FTC modes (Fig. 4.16). However, the steering and tilting actions lead to different results especially for the weave and castering motions. Regarding the weave, the FTC seems to be slightly less stable than FTR (around 10 meters), however, at very low radius the FTC is the most unstable (Fig. 4.16 and 4.17). With regard to the castering, the tilting leads to a more stable motion than the steering action especially at low radius (Fig. 4.17). The general less stable behavior observed for the FTR case could be the consequence of the larger lean angle required by this mode (Fig. 4.6) compared to that of the FTC mode.

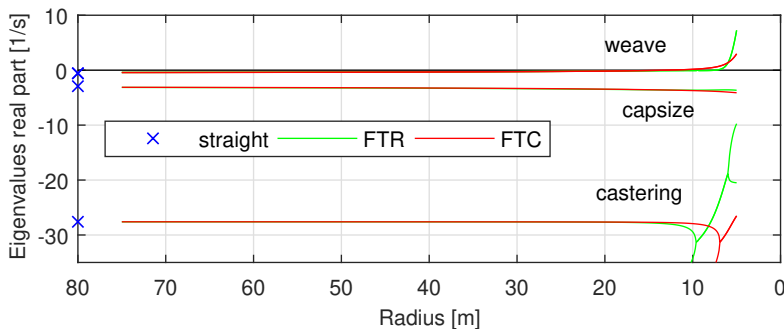


Figure 4.17: Stability analysis for a 5.5 m/s speed of and a varying radius. The steady state cornering equilibrium becomes unstable below 10 meters curving radius for both modes.

4.3 Transient dynamic behavior

4.3.1 Time simulation

The set of ODE equations (3.17) can also be used to study the nonlinear dynamic behavior of the vehicle during transient motions. To do so, the first order ODE system (3.24) corresponding to the vehicle state-space representation (Section 3.2) is time-integrated with the fourth-order Runge-Kutta method (i.e., RK4). A fixed time step of 1e-5 s is used in order to ensure the stability of the integration scheme and avoid numerical errors.

$$x = \begin{pmatrix} q_u \\ \dot{q}_u \end{pmatrix} \rightarrow \dot{x} = f(x) = \begin{pmatrix} \dot{q}_u \\ -M_r^{-1}(q_u)F_r(\dot{q}_u, q_u) \end{pmatrix} \quad (4.10)$$

4.3.2 Transient behavior – curve entry

The transient response of a vehicle to a control input is of the utmost importance when characterizing its dynamics. A different transient behavior should be observed for the NTV depending on the considered action (i.e., steering or tilting). The corner entry is studied with the help of a time simulation (Section 4.3.1): the vehicle step response results from the application of either a steering torque Q_{steer} for the FTR or a tilting action Q_{tilt} for the FTC, while a propulsion torque is also applied to keep the vehicle from decelerating. The corresponding steady state torque is applied from the first second of the simulation. The maneuver is characterized by the steady state cornering condition that the vehicle is reaching defined by a given advance speed, a curve radius and the corresponding centripetal acceleration:

- $V = 10 \text{ m/s} \quad R = 20 \text{ m} \quad a_c = 5 \text{ m/s}^2$

The curve entry is simulated for the two modes (Figs. 4.18 and 4.19). Note that the horizontal displacements are calculated at the chassis reference point (Fig. 3.8):

- Firstly, the horizontal motion (Fig. 4.18) shows that the FTC leads to a slightly faster response than the steering action. Indeed, the vehicle is steered into the curve more directly in the case of a tilting action (Fig. 4.18). This can be observed as well on the yaw rate time history (Fig. 4.19.a) as the FTR leads to a higher initial negative yaw rate than the FTC which corresponds to a higher non-minimum phase (NMP) behavior. This delay is compliant with the dynamics of a steering input torque for which, the lower part of the vehicle needs to be first moved outside of the curve to initiate the chassis roll motion before the vehicle can then be steered into the curve.
- The lean angle evolves in the same way for both actions: it increases monotonically and no overshoot is observed (Fig. 4.19.b). The steady state value is, of course, different as discussed before for the cornering equilibrium (Section 4.1.1).

- The FTC does lead to a small countersteering as it can be observed with the steering fork angle during the curve entry (Fig. 4.19.c). If countersteering is specific to the application of a steering torque, it is surprising to observe it here for the application of a tilting torque. This is probably due to the geometry of the front suspension assembly. As a matter of fact, such countersteering, in opposition to the vehicle tilting, is also observed at a standstill (i.e., static equilibrium with normal ground forces but no lateral forces).

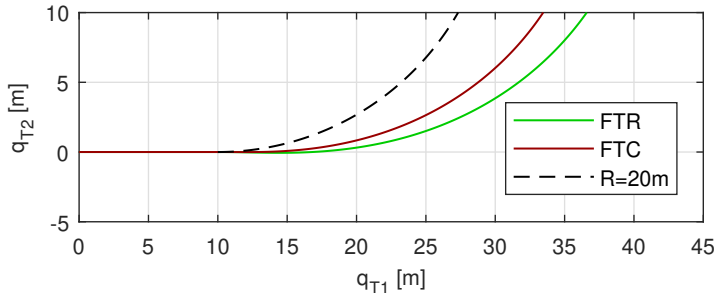


Figure 4.18: In-plane motion of the narrow track vehicle for a curve entry. The tilting torque (FTC) leads to a faster curve entry than the steering torque (FTR).

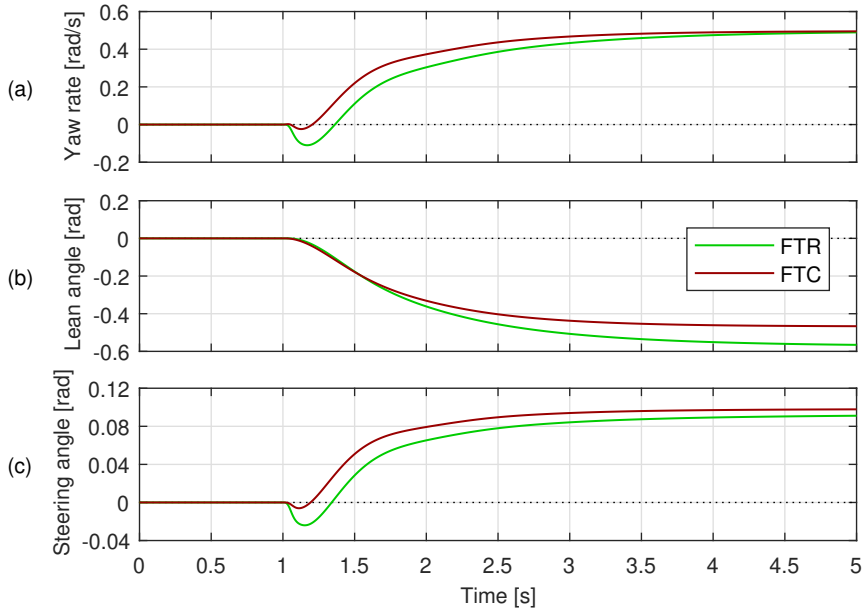


Figure 4.19: Curve entry ($V=10$ m/s, $R=20$ m). (a) The yaw rate shows the outer movement of the NTV lower part. (b) The vehicle leans monotonically. (c) Countersteering is observed for both modes.

Tire/ground contact forces

If the overall kinematic behavior of the vehicle may seem similar for both actions, with an initial countersteering in both cases, the tire/ground interaction strongly differs from one mode to another (Fig. 4.20). In accordance with the steady state results, the tilt action leads to a large wheel load transfer from the inner wheel to the outer wheel (Fig. 4.20.a). It is interesting to see that this load transfer is direct in the FTC case while some NMP behavior is observed for the FTR case. The same observation can be made for the lateral forces (Fig. 4.20.b) with again the FTR leading to initial negative lateral forces directed outward of the curve before building up positive lateral forces. This can be explained by the steering dynamics for which the application of a steering torque directed outward of the curve first forces the wheel to steer outside of the curve, which intrinsically leads to the production of outward lateral forces. For the FTC, the lateral forces are directly directed inwards with the outer wheel producing more lateral forces because of higher normal load.

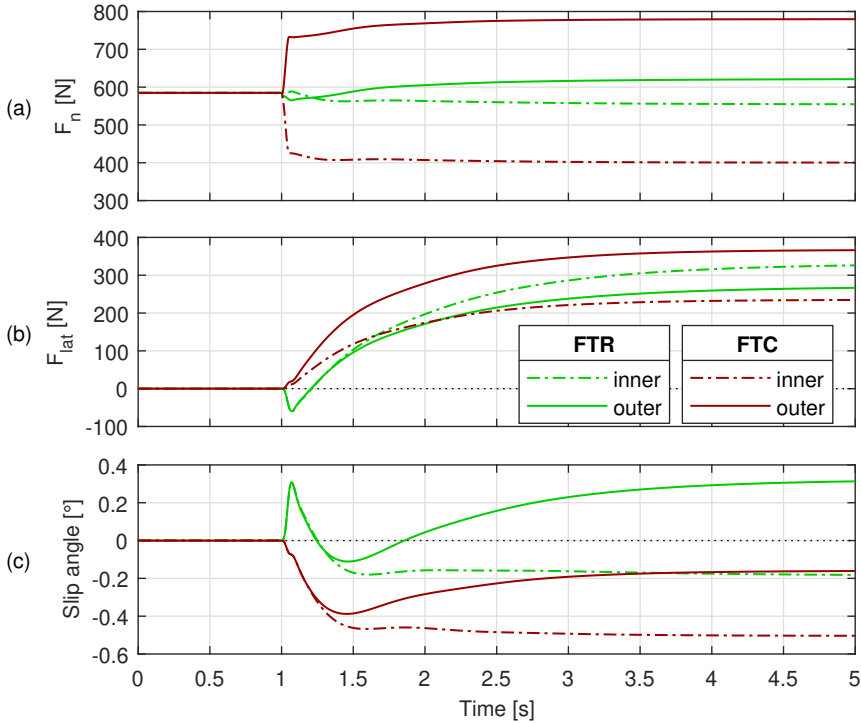


Figure 4.20: Tire ground interaction during a curve entry. (a) The wheel load transfer is higher for FTC (b) There are no lateral forces directed outward of the curve for the FTC (c) Slip angles profile are significantly different for the two modes.

Regarding the generation of slip angles (Fig. 4.20.c), the two modes lead to different behaviors. In FTR, the slip angles are first positive and contribute to

the generation of the initial outward negative lateral forces. However when the vehicle is tilted with FTR and in the steady state cornering condition, the slip angles on the inner and outer wheels are of opposite signs while both lateral forces are positive inward the curve. The difference of signs can be explained by the non-perfect steering assembly which is forced in a given position by the steering torque. The fact that a positive lateral force is still produced at the outer wheel can be explained by the camber thrust which for both wheels contributes to the generation of a lateral force directed toward the inside of the turn (Section 3.3.4). For the FTC case, the positive lateral forces are mainly generated via the negative slip angles for both wheels; as the vehicle roll angle is smaller for FTC than for FTR, the additional camber thrust is of lesser importance.

For the tilt action, the slip angles are negative for both wheels which correspond to inwards positive lateral forces. The difference between the inner and outer wheels is, however, hard to interpret as it is the result of a complex balance of forces and torques on the front suspension assembly. Generally, the order of magnitude for the slip angles is comparable for the two different actions.

Perceived acceleration

As for the steady state cornering, the perceived acceleration can be analyzed during transient maneuvers for both FTC and FTR. First the perceived acceleration vector at point P is observed (Fig. 4.21). The first component $a_{per,1}$ aligned with the chassis longitudinal axis is first positive due to the vehicle acceleration but then reaches a relatively small negative value (Fig. 4.21.a). This is explained by the vehicle body slip angle (Fig. 4.1) that makes the NTV chassis unaligned with the tangential direction and thus a part of the centripetal acceleration a_c is projected along the chassis longitudinal axis $\hat{\mathbf{X}}_1$. The transverse perceived acceleration $a_{per,2}$ first peaks at the beginning of the maneuver and then reaches the steady-state value with the FTC leading to a slightly higher peak (Fig. 4.21.b). The positive value for the measured acceleration corresponds to a centrifugal effect sensed by the driver (i.e., the driver feels pushed outward the curve).

Previous investigations on the matter argue that depending on the sensor height, d , the observed acceleration might strongly differ. Figure 4.22 shows the accelerations for three different points, P_1 corresponds to the driver's abdomen, P_2 to the driver's head and P_3 to the driver's head, assuming that he sits in a scooter-like position (i.e., with his pelvis higher than the knees). A similar tendency is observed for both FTC and FTR with the highest point leading to a higher transient peak for the perceived acceleration (Fig. 4.22.a,b). This is explained by the lateral acceleration caused by the roll acceleration, $\ddot{\phi}$ which has a greater contribution on the perceived acceleration if the point is further located from the roll axis.

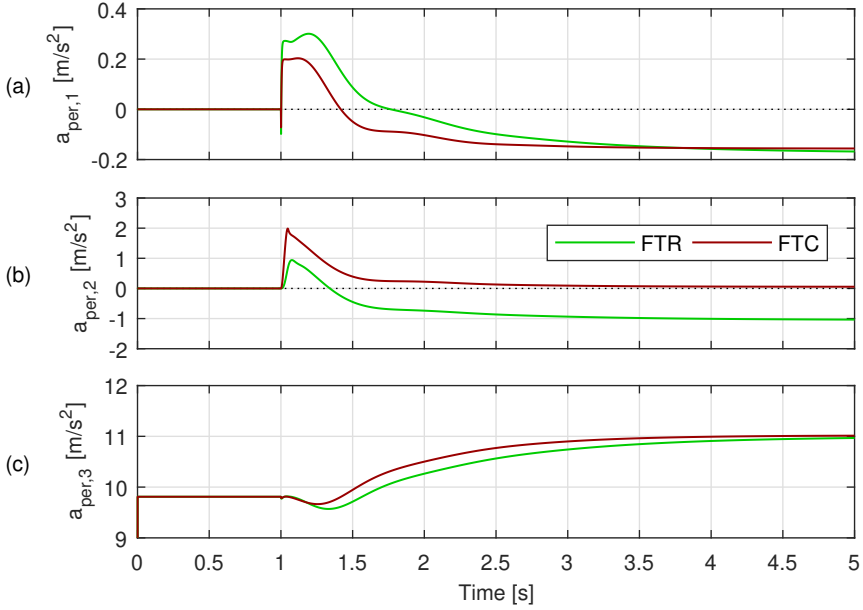


Figure 4.21: The perceived acceleration vector during a curve entry. (a) Component along the chassis longitudinal axis. (b) Component along the chassis transverse axis. (c) Component along the chassis upright axis.

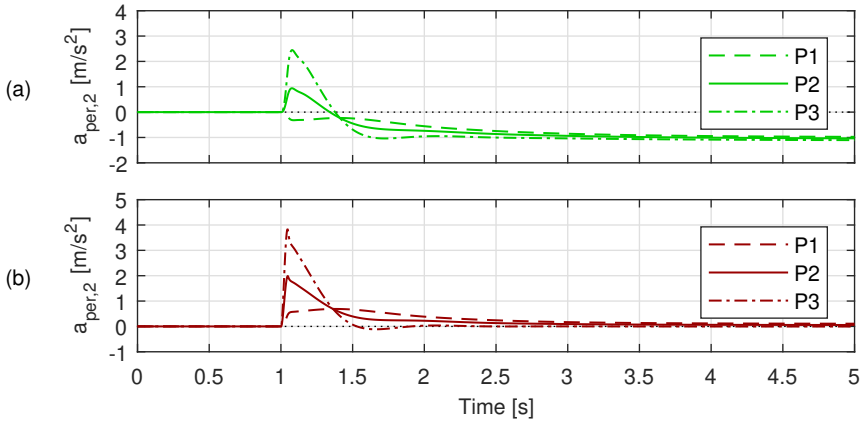


Figure 4.22: Perceived acceleration for three different points of the chassis in case of FTR (a) and FTC action (b). For both modes, the peak in perceived acceleration is larger when the P point is higher.

Dynamic and kinematically driven actuation

So far in the analysis, the two actuation modes have been considered through the application of an internal joint torque. However, for a driver, a desired yaw rate is generally associated with an angular position of the steering wheel rather than an amount of steering torque applied (although a torque feedback is necessary for a good driving experience [76]). In this respect, the application of a step in the angular position rather than a torque for the two modes could be insightful. In both cases, the associated angular degree of freedom is kinematically driven from zero to its steady state value with a smooth step introduced during a half second period (from 1 s to 1.5 s). Simulations have shown that the steering action cannot be realized through the driving of the FTR (Fig. 4.23). This is because countersteering is necessary to initiate the vehicle fall and driving the steering fork monotonically is not suited for that. On the other hand, the kinematic driving can be considered for FTC as, in that case, the vehicle roll motion is monotonic (Fig. 4.19.b and Fig. 4.23.b). This is very insightful from a mechanical design point of view as it indicates that the driver could possibly control the vehicle via a natural direct steering logic without any countersteering action needed.

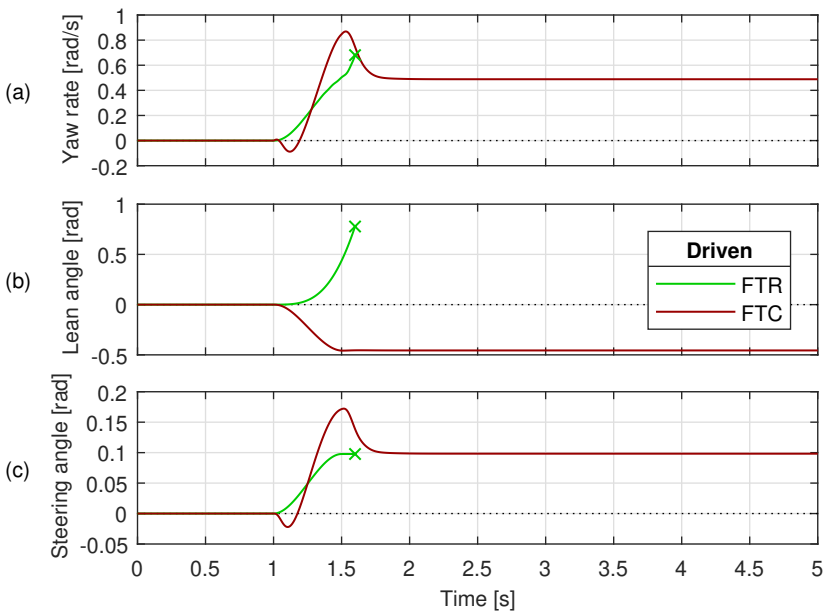


Figure 4.23: Curve entry for a kinematic input. The steering action (FTR) cannot be performed through a monotonic drive of the steering fork from its 0 initial value to the steady state value. However, position driving is compatible with the tilt action (FTC).

Nonetheless, the FTC can be compared for a torque or a kinematic input (Fig. 4.24). Both approaches are difficult to compare as the tilt motion is forced in the case of the kinematic input and free for the torque application. Nevertheless, it is interesting to notice that a stronger tilting actuation, as operated for the kinematic input (Fig. 4.24.d), leads to a larger countersteering (Fig. 4.24.c). This, however, does not necessarily correspond to a larger delay in the lateral dynamics as the steady state yaw rate is reached faster (Fig. 4.24.a).

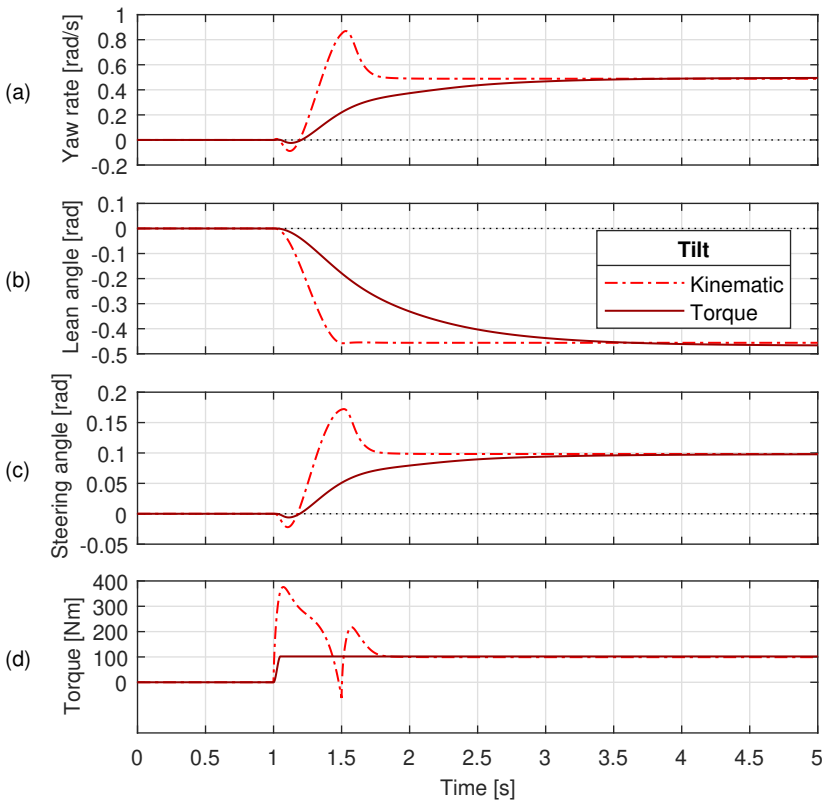


Figure 4.24: Curve entry with a torque or position input. For a fast transition, the kinematically driven tilt d.o.f. implies an important tilting torque. The faster the tilt actuation, the more important the countersteering is.

4.4 Concluding remarks

Based on the different analyses performed with the MBS model, the obtained results have allowed us to highlight the differences between the tilting actuation (FTC) and the steering actuation (FTR) modes. Although the analysis has been performed on a single NTV geometry, we do believe that the following qualitative conclusions can be extended to the dynamics of NTVs in general.

- The steady state cornering behavior of the vehicle is strongly influenced by the actuation mode. Firstly, there is a one-order-of-magnitude difference between the steering torque (FTR) and the tilting torque (FTC). The FTR is more effective as it intrinsically balances the gravity and lateral efforts rather than countering them as required by the FTC. Secondly, the additional inner tilting torque for FTC leads to a more upright position of the NTV compared with the FTR. The vehicle is not aligned anymore with the resultant of the gravitational force and the centrifugal effect. Thirdly, the FTC mode leads to a larger load transfer from the front inner wheel to the outer wheel. This results in a different distribution of the lateral forces and distinct magnitudes in tire slip angles. As the FTC leads to a more upright position, the camber thrust is lesser than for the FTR, which forces the FTC to rely more on the tire slip angles to produce the necessary lateral forces.
- The perceived acceleration for steady state cornering conditions has been studied and it was observed that its value was far from zero on the overall steady state cornering map. This is caused by the non-zero wheel load transfer due to the suspension geometry of the three-wheeler. This result is very interesting as several works in the past have based their NTV controller on the perceived acceleration cancellation [53, 29], which is only correct for simplified vehicle models. Again, FTR and FTC show divergent tendencies for the perceived acceleration with opposite signs on the overall steady state map. The latter is mostly explained by the different lean angles for the two modes.
- Regarding the stability in steady state cornering, it is shown that the two modes have little influence on the stability for a high curve radius. However, when small radii are considered, the vehicle is, as expected, less stable for both actions. The steering action leads to higher instability in weave and castering compared to the tilt action. This might be due to the higher steady state lean angle observed for the steering action. Given the unstable behavior at low speed and for a high lateral acceleration, a stability active controller is necessary to offer a car-like driving experience in which the roll stability is not devolved to the driver.
- The transient responses of the vehicle to both modes are analyzed for a given curve entry. Countersteering is observed for both actions; this is typical of the FTR case but surprising for the FTC mode. The suspension and steering geometries have arguably a great influence over this FTC countersteering. Generally, the steering action (FTR) leads to a higher delay than the tilting action (FTC) with a larger negative peak in the yaw

rate when initiating the curve. It is shown that the FTC can be performed by monotonically driving the tilt d.o.f. up to its steady state value, which is not possible for the FTR that needs countersteering. This result is very insightful on a mechanical design point of view as it indicates that the driver could possibly control the vehicle via a direct steering logic without any countersteering action needed.

- Throughout this chapter, the nonlinear nature of the NTV dynamics has been highlighted. The steady state cornering map (i.e., action torque, lean angle and wheel load transfer) have shown that the vehicle does not behave linearly with respect to the vehicle speed and lateral acceleration. In addition, the simplified formula for the steady state lean angle, used by many research projects on NTV control [11, 12], has been confronted to the MBS model prediction and it was shown that the formula was not properly representing the vehicle nonlinear dynamics when higher centripetal accelerations are considered.

Optimal control

5

Nowadays, optimal control (OC) methods are increasingly used for the control of mechanical systems. Optimal control of multibody systems (MBS) is of particular interest for many mechanical systems such as vehicles, robots or the human body as it offers the possibility to compute a control sequence on a given time interval beforehand such that a given optimality criterion is achieved for the overall motion. The OC is specifically suited for control of MBS whose dynamics can involve strong nonlinearities especially for tridimensional systems. Moreover, the computation of control inputs via optimization enables the control of underactuated systems characterized by a lower number of inputs than degrees of freedom. Finally, the optimal control underlies model predictive control (MPC) which is a powerful feedback control method that can be used for vehicle control (Section 6.3).

The different advantages mentioned above suggest that optimal control is an appropriate tool to study the tilt control of narrow track vehicles (NTV). More specifically, the optimal framework will be exploited to investigate the combined use of the steering and tilting torques (SDTC).

This chapter is divided into two main sections. In the first one, the optimal control formulation for multibody systems is presented. The numerical methods necessary to solve the optimal control problem (OCP) are introduced and then the different formulations of the MBS dynamics are thoroughly discussed with respect to computational efficiency for the solving of OCPs. In the second section, the combined use of steering and tilting actuations is investigated. Several curve entries are realized at different vehicle speeds (including the low unstable ones) and cornering radius. Constrained optimizations on a given time interval are considered in order to address the minimization of the energy consumption, of the perceived acceleration and the limitation of countersteering.

5.1 Optimal control of multibody systems

There are many ways to formulate and solve an optimal control problem (OCP). In this section, the OCP is defined as an optimization problem, based on a differential-algebraic equations (DAE) system for the MBS dynamics (Section 3.1.3). Thereafter, the discrete formulation is introduced together with the

framework used to solve it. In addition, the choice of multibody equations for the dynamic representation of the controlled system is discussed. Finally, the numerical tools developed to implement the optimal control of MBS are briefly presented.

5.1.1 Optimal control problem

In a multibody context, the optimal control problem (OCP) is formulated as an optimization problem with a DAE representation for the dynamics of the constrained MBS. The solution of the OCP is the control trajectory $u(t)$ that minimizes a given *cost function* as defined by equation (5.1) and steers the states, $x(t)$ of the multibody system and its algebraic variables $z(t)$ from a given initial state (5.2) to a given terminal state (5.4), within a given time horizon T (Fig. 5.1). The OCP thus reads:

$$\underset{x(t), u(t), z(t)}{\text{minimize}} \int_0^T l(x(t), u(t), z(t)) dt \quad (5.1)$$

subject to initial, terminal and *path* constraints on the time interval $[0, T]$.

$$x(0) - x_i = 0 \quad (5.2)$$

$$x(T) - x_d = 0 \quad (5.3)$$

$$c(x(t), u(t), z(t)) \geq 0 \quad t \in [0, T] \quad (5.4)$$

Typical examples of cost function are the system energy consumption on the time interval or the system deviation with respect to a prescribed trajectory. Regarding the path constraints, these are often used to express safety conditions on the states, $x(t)$ and $z(t)$ or actuation limitation on the inputs, $u(t)$.

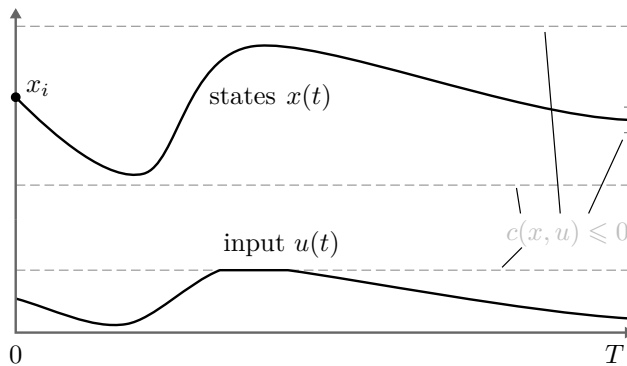


Figure 5.1: The continuous optimal control problem consists in finding the optimal state and control trajectories on the time interval $[0, T]$.

In addition, the trajectory $x(t), z(t)$ and the control inputs $u(t)$ can be subjected to *equality* constraints, g , which account for the system dynamics generally expressed as a set of DAEs. In system (5.5), f and a respectively denote the differential-algebraic equations which relate the differential states x , their first time-derivative, $\dot{x}$, the algebraic variables z and the control inputs u .

$$g(x(t), u(t), z(t)) : \begin{cases} \dot{x} = f(x, u, z) \\ a(x, z) = 0 \end{cases} \quad t \in [0, T] \quad (5.5)$$

The optimization problem defined by the set of equations (5.1-5.5) has, in general, no analytical solution. However, it can be solved numerically through its discretization on the time interval $[0, T]$. This is the purpose of numerical OCP techniques.

5.1.2 Numerical optimal control

There are several approaches to obtain the numerical solution of an OCP [77][78]. Among them, the *direct methods* consist in discretizing the time horizon $[0, T]$ into N time intervals $[t_k, t_{k+1}]$ such that $t_0 = 0$, $t_N = T$ and $k \in 0, 1, \dots, N-1$. The original time functions $x(t)$, $u(t)$ and $z(t)$, that are the unknowns of the system (5.1-5.5), are replaced by a sequence of $N+1$ discrete states x_k , z_k and N piecewise constant controls u_k which are now the decision variables of the optimization problem in its *discrete* form (Fig. 5.2).

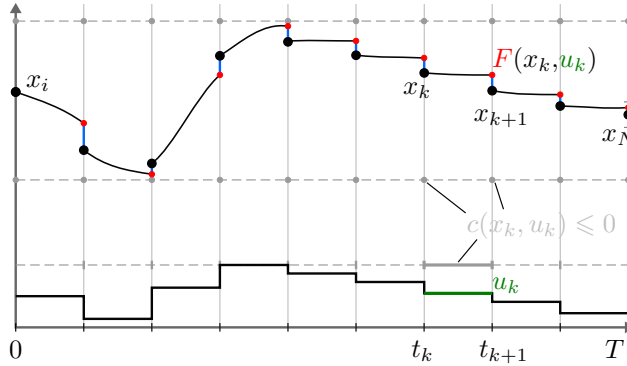


Figure 5.2: Numerical OCP with the discrete dynamic constraints (5.12) that are unsatisfied. The decision variables are the states at the discrete points and the piecewise constant control inputs.

The continuous cost function (5.1) is replaced by a discrete numerical approximation $p(w)$ which is a function of the decision variables vector, w that gathers the discrete states and piecewise constant inputs:

$$w = \{x_0, u_0, z_0, \dots, x_k, u_k, z_k, \dots, u_{N-1}, z_{N-1}, x_N, z_N\} \quad (5.6)$$

The constraints (5.2-5.4) are re-expressed in terms of the decision variables vector w . This discrete parameterization of the original continuous OCP is referred as a nonlinear program (NLP) [78] and reads:

$$\underset{w}{\text{minimize}} \quad p(w) \quad (5.7)$$

subject to constraints on the decision variables:

$$x_0 - x_i = 0 \quad (5.8)$$

$$x_N - x_d = 0 \quad (5.9)$$

$$c(w) \geq 0 \quad (5.10)$$

In the discrete formulation, the system dynamics links the constant control input, u_k and the two adjacent differential states, x_k and x_{k+1} . Starting from x_k , the continuous dynamics, f is time-integrated on the control interval $[t_k, t_{k+1}]$ for the constant control input, u_k (Fig. 5.2). To this extent, the time-integration operator, F is defined as follows:

$$F(x_k, u_k) = \int_{t_k}^{t_{k+1}} f(x, u, z) dt \quad (5.11)$$

The time integration between t_k and t_{k+1} is often performed numerically (i.e., via a Runge-Kutta method). Using the operator, F , the dynamics is taken into account at each discrete time interval via the constraint equations in (5.12).

$$x_{k+1} - F(x_k, u_k, z_k) = 0 \quad (5.12)$$

In addition, the differential and algebraic states are linked at each discrete time through the set of discretized algebraic equations associated to the algebraic part of the DAE system (5.5). At t_k , this reads:

$$a(x_k, z_k) = 0 \quad (5.13)$$

There are two well-known numerical methods to handle the constraints (5.12-5.13): the multiple shooting and the direct collocation [79]. Direct multiple shooting consists in directly time integrating the dynamics on the time interval while algebraic variables, z_k are treated internally by the integrator. For the multiple shooting, the integration and the optimization are treated separately. The choice of the integrator is left free and can be adapted to the system dynamics (e.g., an explicit Runge-Kutta algorithm if the system is purely differential). On the other hand, the direct collocation method introduces collocation points between t_k and t_{k+1} . The differential and algebraic states are then described by a set of polynomial functions that have to respect the continuous dynamics at those collocation points. The integration scheme implies the definition of extra variables (the polynomial coefficients) that are added to the existing decision variables in w . The advantages of collocation methods are that they lead to a very sparse NLP with fast local convergence. In addition,

they treat unstable system well and can easily cope with state and terminal constraints [80]. These reasons and the compared performances observed for both methods have led us to consider the collocation method in this work. More details on the two aforementioned methods and their realization can be found in Appendix C.

5.1.3 Formulation of the multibody dynamic equations

In this section, the influence of MBS modeling on the OCP solving is discussed. Two aspects of the modeling are tackled: the method used to derive the equations and the numerical handling of the algebraic equations for *constrained* MBS (Section 3.1.3).

Derivation of the MBS equations of motion

As mentioned in Chapter 3, the Newton-Euler recursive algorithm (NER) is particularly suitable for deriving the equations of motion for tree-like MBS within a relative coordinate framework. Based on this algorithm, the equations of motion for an unconstrained system can take different forms obtained via various methods which lead to different complexities as summarized in Table 5.1 [72].

	Algorithm	Dynamics	Complexity
IMPL	original NER	$\phi - Q = 0$	n
SEMI	modified NER	$M\ddot{q} + c - Q = 0$	n^2
EXPL1	modified NER + Cholesky	$\ddot{q} - \gamma = 0$	$n^2 + n^3$
EXPL2	Order-N	$\ddot{q} - \nu = 0$	n

Table 5.1: Dynamic formulations of *unconstrained* MBS where n corresponds to the MBS size (i.e., the total number of joints).

The implicit (IMPL) and the semi-explicit (SEMI) forms can both be re-expressed as a DAE system with the generalized coordinates, q and velocities, $\dot{q}$ accounting for the differential states, while the accelerations $\ddot{q}$ are considered as the algebraic states. The inputs are a subset of the independent active joint forces, $Q_{u,a}$ selected via the logical matrix, N and its transpose (Section 3.2). The implicit and semi-explicit equations are reformulated in system (5.14) and (5.15) respectively.

$$x = \begin{bmatrix} \dot{q} \\ q \end{bmatrix} \quad z = \ddot{q} \quad u = N^T Q_{u,a}$$

IMPL:

$$\begin{cases} f(x, u, z) = \begin{bmatrix} z \\ \dot{q} \end{bmatrix} \\ a(x, z) = \phi(q, \dot{q}, z) - Q(q, \dot{q}, u) = 0 \end{cases} \quad (5.14)$$

SEMI:

$$\begin{cases} f(x, u, z) = \begin{bmatrix} z \\ \dot{q} \end{bmatrix} \\ a(x, z) = M(q)z + c(q, \dot{q}) - Q(\dot{q}, q, u) = 0 \end{cases} \quad (5.15)$$

On the other hand, for the explicit formulations (EXPL1, EXPL2), the accelerations $\ddot{q}$ do not *necessarily* need to be considered as algebraic states. As a result, the dynamics can be assimilated to either a set of ODE or a set of DAE. The explicit dynamics is generally denoted, ς and represent the forward dynamics which can either be derived via the NER algorithm and solved in $\ddot{q}$ via a Cholesky decomposition of the mass matrix (EXPL1), or via the so-called order-N method (EXPL2) (Section 3.1.3). The dynamics is represented by a set of differential-algebraic equations (DAE) as follows:

$$x = \begin{bmatrix} \dot{q} \\ q \end{bmatrix} \quad z = \ddot{q} \quad u = N^T Q_{u,a}$$

$$\begin{cases} f(x, u, z) = \begin{bmatrix} z \\ \dot{q} \end{bmatrix} \\ a(x, z) = z - \varsigma(\dot{q}, q, u) = 0 \end{cases} \quad (5.16)$$

The ODE system leads to a lighter representation:

$$x = \begin{bmatrix} \dot{q} \\ q \end{bmatrix} \quad u = N^T Q_{u,a}$$

$$f(x, u) = \begin{bmatrix} \varsigma(\dot{q}, q, u) \\ \dot{q} \end{bmatrix} \quad (5.17)$$

As suspected, the different possibilities in terms of formulations (both for the MBS and the OCP) can lead to different performances for the solving of the OCP; these aspects have been investigated by realizing a case study on a 3D robot arm carrying an inverted pendulum on the end effector (Fig. 5.3.a). An OCP is formulated with the goal to reach a point in the 3D space for the end effector after a preset time of 2 seconds [72]. At the end, the system has to be at equilibrium with zero acceleration and the actuation is minimized on the overall time interval. This task is realized for a grid of target points in the 3D space and for two different definitions of the cost function resulting in two sets of tasks (Fig. 5.3.b). More details regarding this case study (i.e., results and discussions) can be found in Appendix B¹

¹This appendix is a copy of the complete study that appeared as a book chapter in *IUTAM Symposium on Intelligent Multibody Systems – Dynamics, Control, Simulation*. published in 2019 [72].

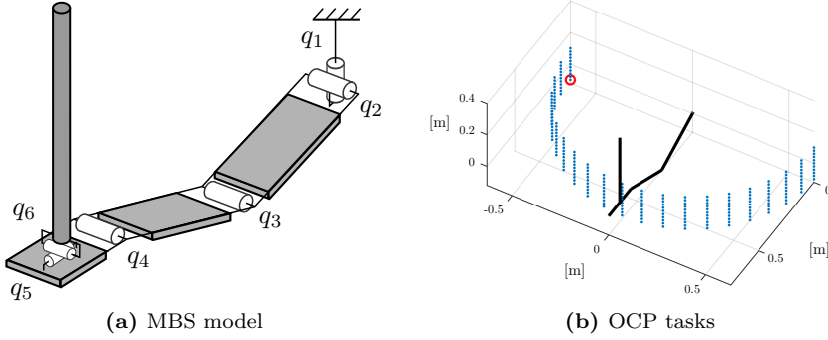


Figure 5.3: Case study for a 3D robot arm with an inverted pendulum [72].

The results of the case study showed that the implicit (IMPL) and semi-explicit (SEMI) formulations are generally performing better in terms of CPU time and number of iterations [72]. This can be explained by a more compact formulation of the dynamics which leads to a lower CPU cost per iteration (Table 5.2). Based on this observation and other considerations (Appendix B), it was concluded that the implicit formulation (IMPL) is the most appropriate to formulate and solve an optimal control problem.

	Algorithm	Dynamics	# Flops	# Eqs
IMPL	original NER	$\phi - Q = 0$	460	114
SEMI	modified NER	$M\ddot{q} + c - Q = 0$	830	264
EXPL1	modified NER + Cholesky	$\ddot{q} - \gamma = 0$	991	328
EXPL2	Order-N	$\ddot{q} - \nu = 0$	1784	395

Table 5.2: Dynamic formulations and their computational complexity for the 3D robot arm (adapted from [72]).

Numerical handling of constrained MBS

As presented in Section 3.1.3, the geometrical loops within a multibody system are modeled with the help of algebraic equations h in terms of the generalized coordinates, q . Using the Lagrange multipliers technique, the equations of motion are written as a semi-explicit DAE system:

$$M(q)\ddot{q} + c(\dot{q}, q) = Q(q, \dot{q}) + J(q)^T \lambda \quad (5.18)$$

$$h(q) = 0 \quad (5.19)$$

$$\dot{h}(q, \dot{q}) = J(q)\dot{q} = 0 \quad (5.20)$$

$$\ddot{h}(q, \dot{q}, \ddot{q}) = J(q)\ddot{q} + \dot{J}\dot{q}(q, \dot{q}) = 0 \quad (5.21)$$

It is well-known that Lagrange mechanics leads to DAE of index-3². In fact, the acceleration $\ddot{q}$ can be *explicitly* expressed as a function of the algebraic variable λ and vice versa via the equation (5.18). Time-differentiating equation (5.21) results in an expression of the constraints third derivative, d^3h/dt that explicitly involves the derivative of $\ddot{q}$ and thus the derivative of λ . By time-differentiating 3 times the algebraic constraint equations h , an explicit ODE is obtained, meaning that the DAE system (5.18-5.21) is of index 3.

Via the coordinate partitioning method [68], the original index-3 DAE can be reduced to an index-1 DAE. The coordinate partitioning consists in splitting the generalized coordinates, q into two subsets: the *independent* ones, q_u and the *dependent* ones, q_v (Section 3.1.3).

$$q = \begin{pmatrix} q_u \\ q_v \end{pmatrix} \quad (5.22)$$

This results in an index-1 DAE with the dependent coordinates, q_v as the new algebraic variables. In the following development, the implicit formulation and its reduced form are considered (Section 3.1.3).

$$\Gamma_r(\ddot{q}_u, \dot{q}_u, q_u, q_v) \triangleq \phi_u - Q_u - B_{vu}^t (Q_v - \phi_v) = 0 \quad (5.23)$$

$$h(q_u, q_v) = 0 \quad (5.24)$$

The index-1 is verified: by differentiating one time equation (5.24), an expression for the time-derivative of the algebraic variables q_v is obtained.

$$\dot{h}(q, \dot{q}) = J(q)\dot{q} = 0 \rightarrow \dot{q}_v = -(J_v)^{-1} J_u \dot{q}_u \quad (5.25)$$

If there exists a function, $\eta(q_u)$ that expresses the algebraic states, q_v for any value of the independent coordinates, q_u (i.e., a part of the differential states), then the index-1 DAE system can further be reduced to an ODE system. The equations of motion are represented by a set of ODE in terms of q_u and their time derivatives.

$$\Gamma_r(\ddot{q}_u, \dot{q}_u, q_u, \eta(q_u)) = 0 \quad (5.26)$$

The function $\eta(q_u)$ is in fact an explicit re-expression of the implicit geometric constraints, h that links the independent coordinates, q_u and the dependent ones, q_v . For some simple multibody systems, an analytical solution exists: $q_u = \eta(q_v)$. However, for tridimensional loops, the function $\eta(q_u)$ will rather be replaced by a numerical algorithm such as a Newton method to find an explicit solution of (5.24) with respect to q_v .

In sum, the dynamics of constrained MBS (e.g., with loops) and the associated algebraic equations can be written in three different ways with each time a different index. The literature on OCP solving for physical DAE systems strongly recommends the use of an index-1 formulation, if available, as they

²The differential index of a semi-explicit DAE being the number of times the algebraic part needs to be time-differentiated before an explicit ODE is obtained [78]

can be approached using standard implicit integration methods [78]. To verify this claim, another case study on the OCP solving for a simplified 3D crank mechanism was conducted [81] and the index-1 representation was retained for the NTV application. From a numerical point of view, the *implicit* dynamics with *index-1* results in the system (5.27). As the MBS dynamics is implicit, the independent accelerations, $\ddot{q}_u$ are also treated as algebraic variables, z .

$$\begin{aligned} x &= \begin{bmatrix} q_u \\ \dot{q}_u \end{bmatrix} & z &= \begin{bmatrix} q_v \\ \ddot{q}_u \end{bmatrix} \\ \begin{cases} \dot{x} &= \begin{bmatrix} \dot{q}_u \\ \ddot{q}_u \end{bmatrix} \\ 0 &= \begin{bmatrix} h(q_u, q_v) \\ \Gamma_r(\ddot{q}_u, \dot{q}_u, q_u, q_v) \end{bmatrix} \end{cases} \end{aligned} \quad (5.27)$$

5.1.4 Implementation of OCP for multibody systems

The implementation of optimal control for multibody systems realized in this present work could never have been realized from the ground up. The numerical code used in this work is based on the combined use of two software programs called Robotran [73] and CasADi [82] coupled through a dedicated OCP toolbox named Rockit [83].

Robotran - Symbolic generation of MBS equations of motion

The main advantage of multibody modeling is the systematic generation of the equations of motion. Once the system is described as a tree-like structure (possibly with constraints), the Newton-Euler recursive scheme can be applied to compute the equations on different forms (i.e., semi-explicit or implicit). In specific cases, the recursive nature of the relative coordinates can be deeply exploited by a symbolic program dedicated to multibody systems. For these reasons, the Robotran software was developed at UCL-MEED and is extensively used to model multibody systems [73]. Based on a topological description (Section 3.1.1), Robotran generates subroutines in C, Matlab or Python syntaxes that express the MBS equations as functions of the coordinates and their derivatives. These functions, computationally optimized thanks to the symbolic approach, can be used externally as a black box by another program or computer environment for any subsequent numerical analyzes. In this work, the expressions generated by Robotran are used by the CasADi software [82] to express the system dynamics for the OCP formulation. The different formulations of the dynamics summarized in Table 5.1 can all be derived with the help of Robotran.

CasADi - Symbolic algorithmic differentiation

The optimization problem (5.7-5.13) can be solved via different techniques which generally require the derivatives of the cost function and constraint equations with respect to the decision variables. To this extent, the software CasADi

[82] provides sensitivity analysis tools for ODE and DAE system that allow the automatic generation of the derivatives necessary to solve the nonlinear programming³ (NLP). To do so, CasADi uses state-of-the-art algorithmic differentiation [84]. Furthermore, it provides an efficient interface to NLP solvers such as the well-known Ipopt library which implements the Interior Point method [85].

Rockit - Rapid optimal control toolbox

The Rockit toolbox is developed by the Meco group at KU Leuven [83]. It offers a convenient framework for the formulation and solving of OCP by using CasADi's AD engine and NLP solver interface. The toolbox implements different methods to solve the OCP such as the multiple shooting method or the direct collocation used in this work. The dynamic expressions generated by Robotran are imported in Rockit. Then the OCP is easily defined in terms of a cost function and path constraints on the differential and algebraic states.

5.2 NTV control through optimal control

The OCP theory developed above for MBS systems can be applied to the control of NTVs. This section takes interest in optimal control of NTVs with combined steering and tilting actions. To do so, the MBS model of a reference NTV developed in Chapter 3 is used.

The main focus is the combined use of tilting and steering torques to drive the NTV in a given steady-state configuration. To stay in this steady-state turn, a propulsion torque, Q_{prop} needs to be also applied to the system; the input vector is thus generally composed of the propulsion torque, the tilting torque and the steering torque.

$$u = [Q_{prop} \quad Q_{tilt} \quad Q_{steer}] \quad (5.28)$$

First a baseline curve entry is realized at different speeds and radii. Then, the versatility of the optimal control framework is demonstrated by taking a closer look at the optimization of the curve entry with constraints.

5.2.1 Baseline curve entry

There are many possible ways to formulate the optimal control problem for the NTV control. In this section, the focus is on the distribution of tilting and steering actuations during cornering. The goal is to steer and tilt the NTV from an upright position to a given steady state configuration within a given time interval $[0, T]$. At the final time $t = T$, the system must be in a steady

³In mathematics, nonlinear programming (NLP) is the process of solving an optimization where some constraints or the objective function are nonlinear.

state cornering equilibrium. The objective function, $L(x, u, z)$ is designed to minimize the actuation and is formulated as the integral of the squared input vector, $u^T u$ as usually done for optimal control problems⁴. As mentioned in Chapter 4, the torque can be used to quantify and compare the steering and tilting actuation as the range of motion for the two actuated joints is similar (i.e., ± 45 degrees). Using the square torque. The steady state cornering to reach at the final time, T is defined by the speed, V and the yaw rate, $\dot{\psi}$ as done in Chapter 4 for the open-loop simulations (Section 4.3).

$$L(x, u, z) = \int_0^T l(x(t), u(t), z(t)) dt = \int_0^T u^T u dt \quad (5.29)$$

In practice, the control torques are piece-wise constant on each discrete time interval for the sake of numerical implementation. Regarding the control intervals themselves, their number is limited to $N = 20$ per second in order to keep the size of the associated NLP reasonable and solvable for a computer. If this number seems low, one as to remember that the collocation method provides additional integration points between each control interval for which an implicit integration scheme is applied (Appendix C). In addition, a step response has been simulated with the model, and by comparing with a higher number of intervals N , it was shown that 20 control intervals per second was already sufficient to properly capture the dynamics of a curve entry.

To achieve the equilibrium at the final time T , constraints are applied on the differential states, $x = [q_u \dot{q}_u]$, the inputs, u and the algebraic states, z .

- The final speed, V is imposed via an algebraic constraint between the two velocities in the horizontal plane (Fig. 3.8) :

$$\text{at } t = T : \quad \dot{q}_{T1}^2 + \dot{q}_{T2}^2 - V^2 = 0 \quad (5.30)$$

- The prescribed yaw rate, $\dot{\psi}$ at the final time is directly accounted for by constraining the joint velocity of the yaw coordinates for the chassis (Fig. 3.8):

$$\text{at } t = T : \quad \dot{q}_{R3} = \dot{\psi} \quad (5.31)$$

- The equilibrium of the whole system is achieved by canceling the joint accelerations at the final time except for the two translation accelerations in the horizontal plane which are left free to account for the centripetal acceleration at the final time.

$$\text{at } t = T : \quad \ddot{q}_u = 0 \quad (\text{except } \ddot{q}_{T1} \text{ and } \ddot{q}_{T2} \text{ which are free}) \quad (5.32)$$

- The steady state cornering at final time is a pure steering equilibrium as opposed to a pure tilting equilibrium. The tilting torque is set to zero at $t = T$.

$$\text{at } t = T : \quad Q_{\text{tilt}} = 0 \quad (5.33)$$

⁴Note in addition that the square of the torque can be related to the dissipated power via the Joule losses as the torque is proportional to the current

High lateral acceleration curve entry First a high lateral acceleration maneuver is considered, defined by a terminal speed of $V = 10$ m/s and a curve radius of $R = 20$ m, which amounts to a yaw rate of $\dot{\psi} = 0.5$ rad/s. This corresponds to the curve entry realized in the dynamic analysis with a lateral acceleration of 5 m/s^2 , which is relatively high compared to what is done in the literature. A total time of 2 seconds is considered for the maneuver which seems reasonable in view of the settling time for the step response in the dynamic analysis (i.e., approx 3 s).

The maneuver is performed for the SDTC mode and the pure STC mode (Fig. 5.4 and 5.5). The OCP results show that the SDTC mode makes very little use of the tilt action while realizing the curve entry (Fig. 5.4). This is because the optimization intends to limit the torque actuation on the overall interval and makes use of the steering action which is known to be generally smaller than the tilting action by one order of magnitude (Section 4.1 and 4.3).

The SDTC and STC lead to similar results although the end of the simulation is slightly different as SDTC can use the tilt action to stabilize the vehicle at the very end of the time interval. In both modes, the steering torque is active from the beginning of the time interval because the control is trying to initiate the NTV fall at the beginning via a steer action which is maintained and increased at the middle of the time interval (Fig. 5.4). Note that although the objective of the optimizer is to limit the actuation, the tilting torque is not used before the steering action to initiate the fall and enter the turn as it is less costly to use countersteering to initiate the fall.

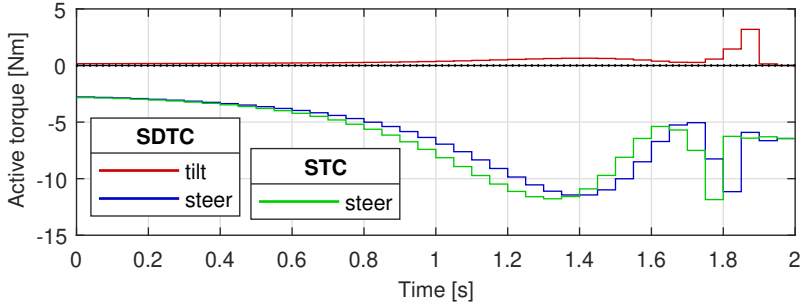


Figure 5.4: Input torques for a high lateral acceleration curve entry ($V=10$ m/s, $R=20$ m). The SDTC mainly uses the steering torque.

A similar dynamic behavior to the open-loop dynamic simulation (Fig. 4.19) can be observed with an initial countersteering reflecting a non-minimum phase (NMP) behavior for the yaw rate (Fig. 5.5.a and c). Nonetheless, the end of the maneuver differs significantly with overshoots observed for the steering angle, the yaw rate and even the roll angle (although moderate). This is explained by the important steering torque peak at $t = 1.8\text{s}$ which is necessary to reach the steady-state equilibrium in the allocated time.

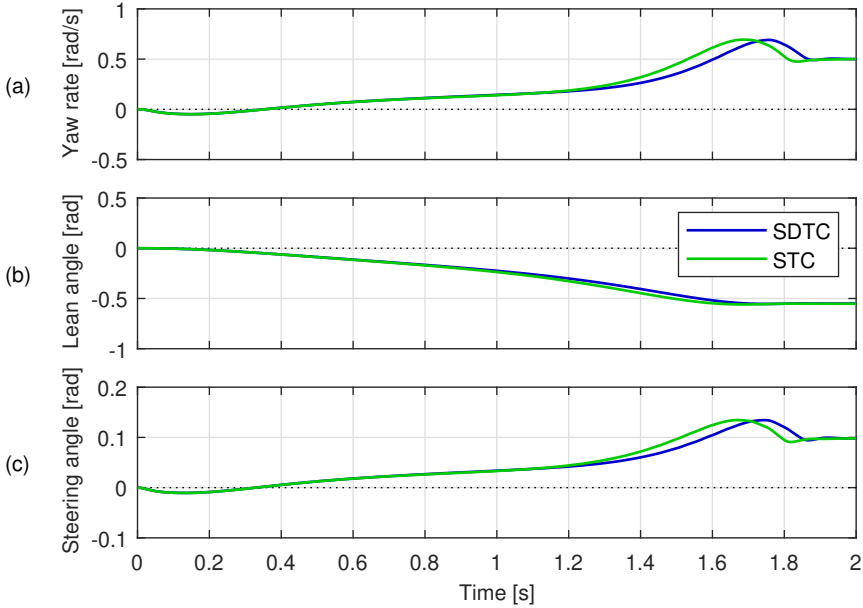


Figure 5.5: High lateral acceleration curve entry ($V=10$ m/s, $R=20$ m). A countersteering is operated from the start of the maneuver.

Low speed curve entry A second optimization is performed at low speed, i.e., 3 m/s, for which the vehicle is known to be unstable. The steady-state configuration to reach after two seconds is defined by a radius of 5 meters which corresponds to a low lateral acceleration, $a_c = 1.8m/s^2$. The first observation is that again the SDTC mainly results in a steering actuation (Fig. 5.6) as the optimization minimizes the actuation level according to the objective function in equation (5.29). Since the maneuver is less dynamic, a 2-second window is more than enough to reach the desired steady state cornering. As a consequence, the actuation during the first second is kept low by the optimization.

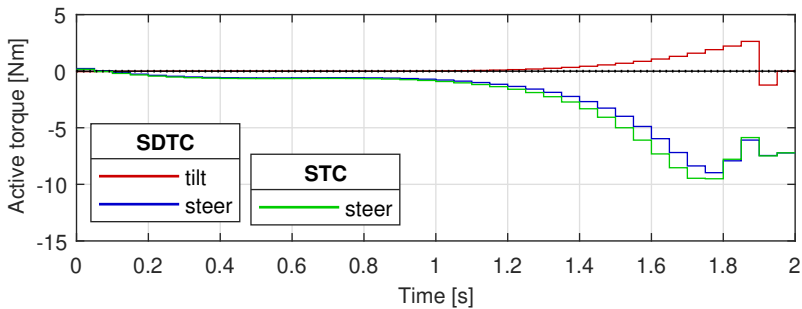


Figure 5.6: Input torques for a low speed curve entry ($V=3$ m/s, $R=5$ m). The SDTC mainly relies on the steering torque.

As the vehicle is unstable, the first second is used to initiate a slight countersteering along with a NMP behavior in both the yaw rate and the lean angle (Fig. 5.7.b and c). When the fall is initiated, then the control stabilizes the vehicle during the last half second. Generally, the actuation profiles at high and low speed have the same shape with first a constant steering torque that initiates the roll motion and then a combined tilting and steering actuations at the end that stabilize and to reach the prescribed steady-state condition. The actuation profiles for high and low speed are similar with first a constant steering torque that initiates the fall and then a combined tilting and steering actuation at the end to stabilize the NTV and reach the prescribed steady state condition.

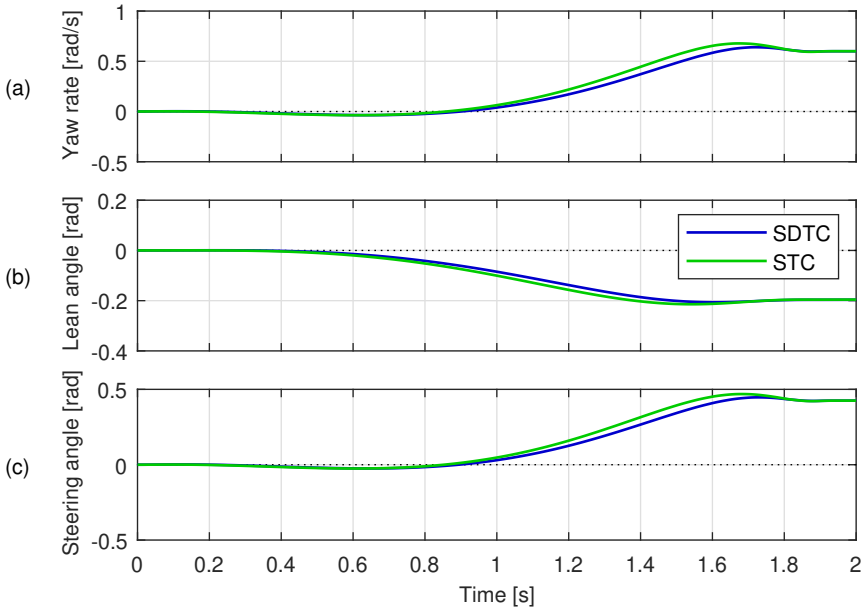


Figure 5.7: Low speed curve entry ($V=3$ m/s, $R=5$ m). The SDTC performs the low speed maneuver smoothly in a similar manner as for the high lateral acceleration maneuver.

5.2.2 Constrained curve entry

In this section, the focus is on the constrained optimal control case for NTVs. The main observation from the above section is that the optimal control for the chosen cost function is mainly using a steering torque to drive the vehicle into the desired steady state cornering. The main issue is that the countersteering can lead to path deviation and to an undesired lateral displacement toward the outside of the curve. To alleviate this issue, one can limit the countersteering by constraining some of the differential states. Typically this is achieved by

adding the following constraint (5.34) to the initial OCP defined in equations (5.29-5.33):

- The countersteering is avoided by forcing the steering rate to positive value ensuring *direct* steering on the whole time interval.

$$\forall t \quad \dot{q}_\delta \geq 0 \quad (5.34)$$

The high lateral acceleration curve entry ($V=10$ m/s, $R=20$ m) is considered and the constrained case is compared to the unconstrained one. The direct steering constraint leads to significant differences in actuation compared to the baseline case (Fig. 5.8 and 5.9). As no countersteering can be applied, a positive tilting torque is used to initiate the vehicle fall. Once this is done, a steering torque peak is applied around $t = 1$ s to generate the necessary amount of yaw acceleration, $\ddot{\psi}$ (Fig. 5.8 and 5.9.a). A roughly constant steering torque is applied until the final time and a reverse tilting torque is applied to reach the desired steady state. Let us remember that the optimization still tries to minimize the actuation torques meaning that the relatively large tilting torque values at both the beginning and the end of the interval are necessary to reach the steady state and respect the direct steering constraint. It is important to stress that the observed peak values for the tilting torque are one order of magnitude smaller than what a DTC or LTC control would exhibit to achieve such curve entry (Chapter 6).

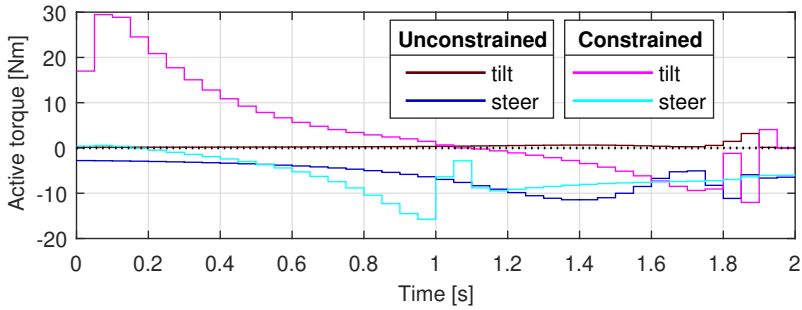


Figure 5.8: Input torques for a constrained curve entry ($V=10$ m/s, $R=20$ m). The direct steering constraint leads to a large tilting torque.

Regarding the angular quantities, one can see that the direct steering constraint leads to a three-step maneuver (Fig. 5.9). First, it is important to notice that, due to constraint (5.34), no NMP nor overshoot are observed for any of the angular quantities. During the first second, the fall is initiated by a strong tilting torque causing a large perceived acceleration; however, the vehicle is not steered into the curve (Fig. 5.9.b). At around 1 second, the yaw rate and the steering angle increase monotonically due to the steering peak torque. Then, during the last second, the vehicle slowly reaches the steady state, helped by the negative tilting torque applied at the very end.

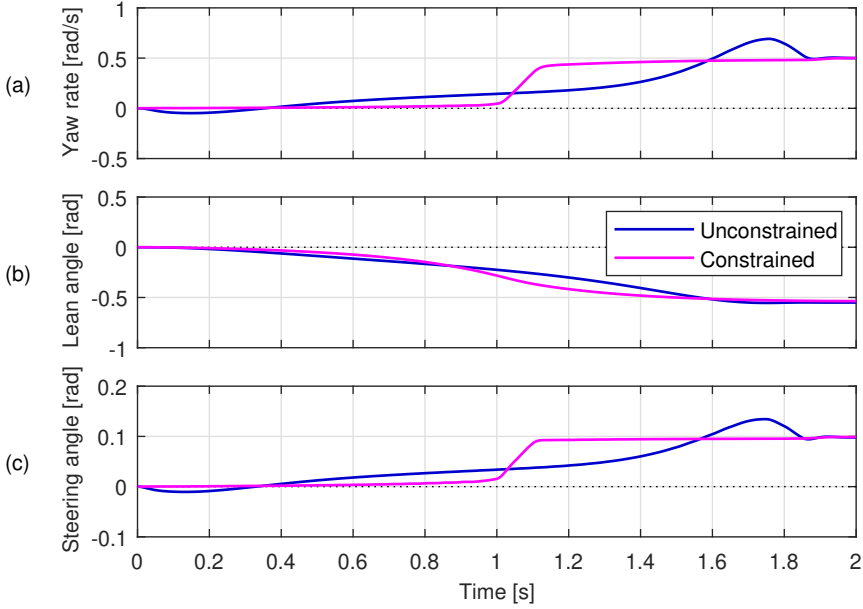


Figure 5.9: Constrained curve entry ($V=10$ m/s, $R=20$ m). The direct steering constraint leads to a three-step maneuver.

5.2.3 Perceived acceleration minimization

The optimal solution found for the unconstrained baseline maneuver leads to a relatively large yaw motion at the end of the maneuver (Fig. 5.5). This could cause a high peak in the transverse perceived acceleration at the end of the time interval which could be associated to discomfort for the driver as explained in Chapter 4 (Fig. 4.21).

Thanks to the OCP framework, it is also possible to minimize the peak in the transverse perceived acceleration at a given point P on the chassis⁵, $a_{per,2}^P$ via an additional threshold variable, α . This variable, α is added to the optimization vector, w in equation (5.6) and considered in the original cost function (5.29) via the use of weights R_1 and R_2 that balance the actuation cost minimization and the acceleration peak mitigation.

$$L(x, u, z) = R_1 \int_0^T u^T u dt + R_2 \alpha \quad (5.35)$$

where $R_1 = 10^{-3}$ and $R_2 = 1$ account for the difference of units between the two objectives. Eventually, the link between α and the peak in $a_{per,2}^P$ is done via a path constraint (5.36) stipulating that the square of the perceived acceleration should not be larger than the threshold α on the whole time interval.

$$\forall t \quad (a_{per,2}^P)^2 < \alpha \quad (5.36)$$

⁵The chosen point P is P_3 as defined in Section 4.1.2.

The OC problem is solved and the results are compared to the baseline curve entry (Fig. 5.10 and 5.11). For the minimization of α , both the tilting and steering torques are used to initiate the fall (Fig. 5.11) directly leading to a positive non-zero acceleration, which corresponds to a centrifugal effect for the driver (Fig. 5.10.a). This transverse perceived acceleration then becomes negative (i.e., centripetal feeling) as a steering torque is abruptly applied around $t=0.4$ s (Fig. 5.11). The transverse perceived acceleration stays negative until the vehicle reaches the steady state cornering condition for which the perceived acceleration is known to be negative as shown in Chapter 4 (Fig. 4.21). Generally, the peak in the perceived acceleration is drastically reduced on the whole speed range (Fig. 5.10.a). In addition, the constraint on the α -threshold leads to a smoother roll and steering motion (Fig. 5.10.b and c). Nevertheless, minimizing the α -threshold is not without cost as the overall input torques are larger than for the baseline case (Fig. 5.11).

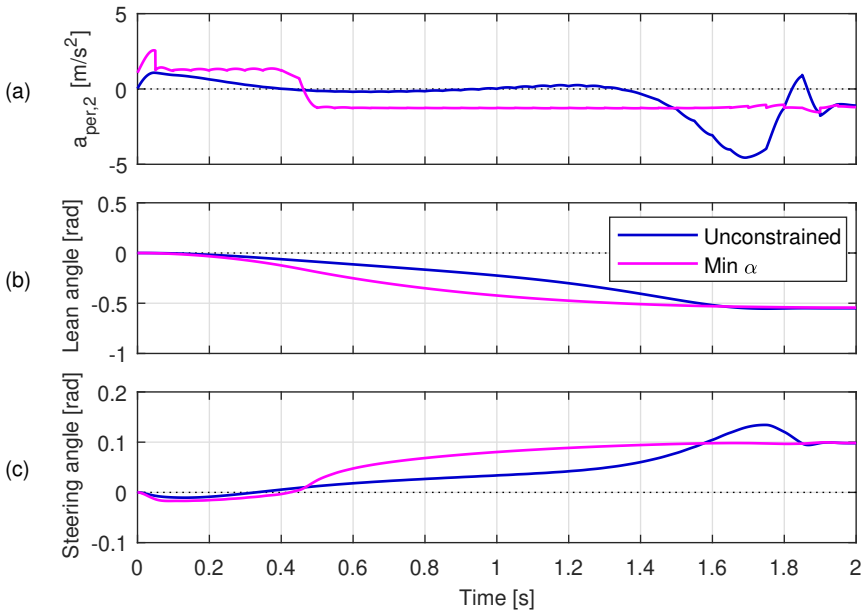


Figure 5.10: Curve entry with perceived acceleration minimization ($V=10$ m/s, $R=20$ m). The perceived acceleration is limited leading to a smoother curve entry.

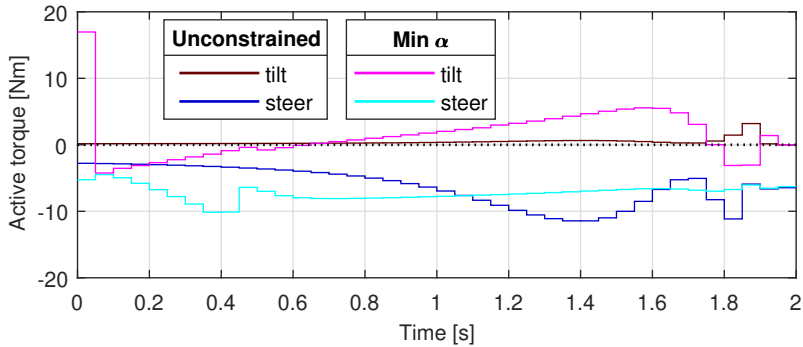


Figure 5.11: Input torques for a curve entry with perceived acceleration minimization ($V=10$ m/s, $R=20$ m). The limitation of the perceived acceleration peak value leads to larger input torques for the associated SDTC.

5.3 Concluding remarks

In this chapter, a framework for the optimal control of multibody system was developed and applied to the control of NTVs. Although the optimal control cannot be directly implemented as such on a vehicle due to its open-loop nature⁶, several conclusions can be drawn for this chapter.

- The different methods used to derive the equations of motion for a multibody system strongly influence the computation time necessary to solve a given optimal control problem. In this chapter, and more specifically in Appendix B, it is shown that an implicit formulation of the equations of motion leads to faster OCP solving, opposed to what is often observed in time numerical integration for which explicit formulations (i.e., forward dynamics) are generally preferred.
- The formulation of an optimal control problem for a multibody system is not trivial. As shown in Section 5.1.3, kinematic loops, inherent to many mechanical systems (e.g., vehicle suspension), need a specific numerical treatment which often leads to a numerical DAE system. Although there exist tools to treat this type of system, these are not trivial and require extra effort in the formulation of the OCP [81].
- Thanks to the OC approach, both self-stable speed and low unstable speed (i.e., lower than 5 m/s) have been simulated, which broadens the dynamic analysis realized in Section 4.3.
- While the SDTC mode is envisaged and the actuation torques are minimized, optimal control systematically leads to a more important usage of the steering torque than the tilting torque. The NTV fall is initiated via a countersteering action while the steady-state configuration is also

⁶The slightest perturbation or model discrepancies would influence the vehicle final configuration after the maneuver.

reached via a combined use of tilting and steering torques.

- The optimal control framework has also enabled the treatment of constrained maneuvers. Applying a direct steer constraint (i.e., positive steering rate on the overall time interval) forces the controller to solicit a tilting torque at the beginning in order to initiate the roll motion.
- The peak in perceived acceleration has also been minimized for the curve entry. Results show that it leads to a larger actuation, both in tilting and steering, and to a smoother curve entry.
- Although the optimal control has shown itself insightful for the control of NTVs, the computation effort associated to the advance nonlinear MBS model limits the use of OCP to offline investigations. In addition, the optimization is based on Newton's method which means that the optimum found might not necessarily be global.
- Moreover, the optimization process was often not converging directly because of the terminal constraints. Most of the time, it was necessary to solve several OCPs in order to reach the desired steady state, step by step: decreasing the curve radius progressively and reusing the previous step OCP solution as the initial guess.

Feedback control

6

This chapter focuses on feedback control applied to NTVs. Although optimal control is very insightful to compare the different actuation modes with respects to terminal constraints and optimal criteria, it cannot practically be used to control a vehicle in real life. This is because in reality, there is neither terminal constraint, nor trajectory on the time horizon known in advance: the only pieces of information available when considering cooperative¹ NTV control are the steering wheel orientation and the throttle position at time t and their time history. Furthermore, the NTV is generally subjected to unpredicted perturbations (e.g., side wind) and model inaccuracies (e.g., change in vehicle total mass) which would make any form of open-loop optimal control inadequate.

According to the state of the art of Chapter 2, investigations on NTV control are mainly based on the implementation of feedback controllers. From this same chapter, it appears clear that there are multiple options for controlling NTVs. The reference signal for example can either be the vehicle lean angle, the perceived acceleration or the yaw rate. Moreover, there exists a multitude of feedback-based control methods which range from linear control theory to nonlinear advanced techniques. Based on the nonlinear NTV model developed in Chapter 3, the present chapter presents three nonlinear methods for the NTV control: a gain-scheduling linear quadratic control, a feedback linearization control and a model predictive control. The applicability and performances of each method are discussed for the four tilting modes: STC, LTC, SDTC² and DTC (Section 1.2).

As the main interest is the different control modes and their comparison, several control aspects are set aside in this chapter:

- The estimation problem is not considered and thus the vehicle states, $x(t)$ are assumed to be known and available at any time t together with their time history.
- The robustness of the control methods is not investigated as no discrepancies are introduced between the *plant* NTV model and the *control* NTV model. In other words, there are no modeling errors considered. In

¹The driver and the vehicle interact and work together. Generally, the driver performs the high level path-following tasks whereas the vehicle tilts itself.

²Note that SDTC denotes here the combination of the tilting and the steering torques and not the modulation of the steering angle as done in the literature.

addition, the vehicle is assumed to be driven on a flat horizontal ground and no perturbations (e.g., side wind) to the system are considered.

- The path tracking assigned to the driver via the steering wheel control is not modeled. Instead, the steering wheel angle and its derivatives are prescribed in time and represent a desired maneuver that the driver would perform (e.g., a curve entry). The absence of driver model for the high level path-tracking limits the present analysis to simple maneuvers and does not allow for the simulation of a double-lane-change such as the Moose Test [86].

Although these simplifications seem quite restrictive, we believe that they are necessary to properly compare the envisaged control techniques and the different tilting modes. In order to include the high level path-tracking feedback loop, it would be necessary to develop an advanced NTV driver model that relies on an internal model of the active vehicle. The development of such a driver model is not an easy task and would amount to another complete thesis [87][88][89]. Therefore, we have decided to do without a driver model and to compare the tilting modes and control techniques based on their low-level performances rather than their drivability or ability to perform a given maneuver via a simple PD driver controller.

This chapter is divided into three main parts, each of them being related to a different nonlinear control techniques. First gain-scheduling LQR is discussed, then feedback linearization is considered and finally model predictive control is investigated. For each control technique, the theoretical bases are presented and applied to the specific case of multibody systems. Then, NTV control is investigated and results are obtained for the different control modes (i.e., STC, LTC, SDTC and DTC). At the end of the chapter, general conclusions are drawn on the method's applicability and performances for the four envisaged control modes.

6.1 Gain-scheduling linear quadratic control

6.1.1 Linear quadratic control

A very common method to control a system is the *design-via-linearization* approach which consists in linearizing the system around a single operating point and deriving a control law for the obtained linear system. In Chapter 3, it is explained how the nonlinear system dynamics (6.1) can be linearized by considering small perturbations Δ around a given configuration (x^*, u^*) , which results in system (6.2).

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad (6.1)$$

$$\begin{cases} \Delta \dot{x} = A^* \Delta x + B^* \Delta u \\ \Delta y = C^* \Delta x \end{cases} \quad (6.2)$$

with

$$\Delta x = x - x^* \quad \Delta u = u - u^*$$

Based on that linear system (6.2), a state feedback gain K can be computed expressing the input as a function of the system state, x around the configuration (*) which is supposed known by the controller.

$$\Delta u = K \Delta x \quad (6.3)$$

There exist many methods to derive the gain matrix K . Among the latter, the linear quadratic approach is presented first for a regulation problem then for a trajectory tracking problem.

Linear quadratic regulator Linear quadratic control enables the control of dynamic systems while minimizing a given performance index, J which depends on the state deviation Δx and the actuation Δu . It has the advantage to guarantee stability if all states are available (observable) and if the model is perfect.

$$J = \int_0^\infty [\Delta x^T(t) Q \Delta x(t) + \Delta u^T(t) R \Delta u(t)] dt \quad (6.4)$$

where Q is a symmetric gain matrix for the regulation part and R is a symmetric gain matrix for the actuation part. In this section, the infinite horizon case is considered ($T = \infty$) allowing for a static expression of the gain matrix K . The linear quadratic regulator (LQR) problem can be solved via the maximum principle and an expression for the feedback gain K is obtained by the algebraic Riccati equation [61]. The solution is a gain K_{lqr} that both stabilizes (6.2) and minimizes the performance index J in (6.4) (Fig. 6.1).

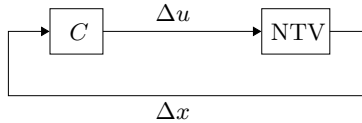


Figure 6.1: Block-diagram representation of the linear quadratic regulator.

The remaining latitudes for the controller design are the gain matrices, Q and R . The matrix Q is used to weigh the error on the different states in x . For instance, by adding weight on the velocity states, damping can be achieved for the control task. The matrix R can be adjusted in order to restrict the actuation needs associated to the control.

Linear quadratic tracking In order to design a controller to track a trajectory (x_d, u_d) , the system dynamics needs to be re-expressed in terms of the error, $e = x - x_d$ and the input $v = u - u_d$ [90]. To do so, the nonlinear dynamics (6.1) is considered.

$$\dot{e} = \dot{x} - \dot{x}_d \quad (6.5)$$

$$= f(x) + g(x)u - f(x_d) - g(x_d)u_d \quad (6.6)$$

$$= f(e + x_d) - f(x_d) + g(e + x_d)(v + u_d) - g(x_d)u_d \quad (6.7)$$

$$= F(e, v, x_d(t), u_d(t)) \quad (6.8)$$

During trajectory tracking, the error e is kept small as far as the controller is performing correctly. Based on this assumption, the error dynamics can be linearized around $e = 0$.

$$\dot{e} \approx A(t)e + B(t)v \quad (6.9)$$

Generally, $A(t)$ and $B(t)$ matrix depends on x_d and u_d only, as the trajectory is supposed to be as a sequence of equilibrium points (x_d, u_d) .

$$A(t) = \left. \frac{\partial F}{\partial e} \right|_{x_d, u_d} \quad B(t) = \left. \frac{\partial F}{\partial v} \right|_{x_d, u_d} \quad (6.10)$$

Assuming that x_d and u_d are slowly varying with respect to the performance criterion, the system dynamics can be considered as slowly varying from one linear system $(A(x_d), B(x_d))$ to another. Thus the state feedback control gain, K_{lqr} can be designed by solving the LQR problem offline for each point of the trajectory x_d and the error can be regulated using

$$v = K_{lqr}(x_d)e \quad (6.11)$$

The input is thus obtained as a combination of a feedback action based on the error and a feedforward action which is the steady-state input u_d .

$$u = K_{lqr}(x_d)(x - x_d) + u_d \quad (6.12)$$

This results in a gain-scheduling linear controller for which both the feedforward term and the gain matrix will depend on a scheduling variable (e.g., the reference signal, the vehicle speed) that parametrizes the desired trajectory (Fig. 6.2).

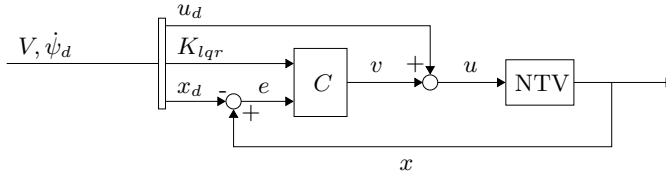


Figure 6.2: Block-diagram of the gain-scheduling LQR controller.

6.1.2 Gain-scheduling LQR for NTV control

The gain-scheduling LQ control can be applied to the control of NTVs. As shown in Chapter 4, steady state cornering equilibrium can be parametrized by the forward speed, V together with either the curving radius R , the lateral acceleration, a_c or the vehicle yaw rate, $\dot{\psi}$. In addition, Chapter 4 has shown that the vehicle stability strongly depends on the vehicle speed (Fig. 4.14). The amount of turning is also critical when the curving radius is lower than 10 meters (Fig. 4.17). The gain-scheduling control supposes that the scheduling variable (i.e., the reference signal) slowly varies in time. Given the fast lateral dynamics exhibits by NTVs (section 4.3) and the assumption of a slowly varying trajectory (x_d, u_d) , it was decided to realize the feedback gain scheduling on the speed V solely, thus $K_{lqr}(V)$. This means that the feedback gains are derived from the straight line equilibrium only. Nonetheless, the feedforward gain scheduling is realized both on the speed and the amount of turning, thus $u_d(V, R)$. For practical reasons, the yaw rate $\dot{\psi}$ is used rather than the turning radius as it allows for a better gain interpolation between the straight line and the highest curving radius computed (i.e., 75 meters). Thus the feedforward gain scheduling depends on both the speed and yaw rate, $u_d(V, \dot{\psi})$, enabling a zero static error in steady state cornering as far as the vehicle model is supposed to represent the vehicle dynamics perfectly.

The developed LQ controller considers the whole desired states, x_d rather than a desired lean angle, ϕ_d or a desired yaw rate, $\dot{\psi}_d$ for tracking. For a perfect model, this is not an issue as the lean angle and the yaw rate are linked together through the steady state cornering equilibrium (Section 4.1). In case of discrepancies between the actual vehicle and the model, an integral action could be added on either the lean angle error, e_ϕ or the yaw rate error, $e_{\dot{\psi}}$ allowing the cancellation of the static error and/or perturbation. Within the framework of this thesis, tracking the full state, x_d allows for a proper comparison of the parallel direct tilt control (DTC) with the serial modes (STC, LTC, SDTC) as done in Section 6.1.3.

Parallel control – DTC

In the DTC mode, the driver directly steers the vehicle on his own through the rigid linking (constant steering ratio) between the steering wheel and the steering fork. The actuator is forced to ensure the vehicle tilt angle which compensates for the centrifugal effect.

The steering fork rotation, δ is assumed to be kinematically controlled by the driver who performs a given maneuver, i.e., a predefined function in time, $\delta = \delta_d(t)$ where d stands for kinematically driven³. Based on this steering input, the desired yaw rate is obtained thanks to a steady-state pre-computed map (Fig. 6.3). For any configuration, each pair $(\dot{\psi}_d, V)$ allows for the determination of the overall state x_d while the gain matrix $K_{lqr}(t)$ is defined for each time, t through the forward speed $V(t)$ only.

³The trajectory of an *independent* joint in a MBS can be prescribed in time. More details on the subject are given in [67].

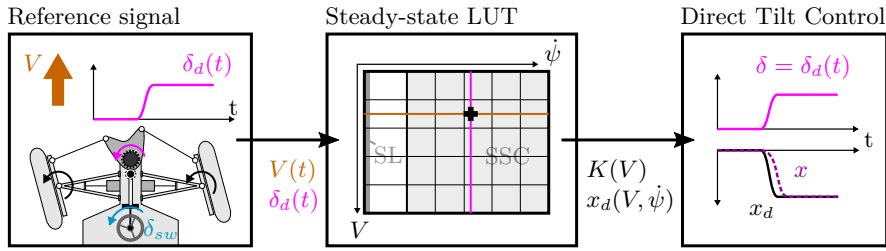


Figure 6.3: DTC principle. Based on the steering fork orientation and the forward speed, V , the desired state, x_d is computed from a lookup table (LUT) linear interpolation.

In the DTC mode, the tilt control only involves a state feedback term and no feedforward term, u_d . In a steady state cornering, there is no need to apply an additional tilting torque to ensure stability. Similarly to a traditional two-wheeler, the steady-state equilibrium in a curve only depends on the torque applied on the steering system (i.e., the steering fork or the handlebar). To implement a robust DTC controller, the cancellation of the tilting torque is a good way to deal with model inaccuracies and track the proper lean angle.

Serial control – STC/LTC/SDTC

The serial controls, namely the STC, LTC and SDTC are performed based on a desired yaw rate as the steering wheel is decoupled from the steering fork. The steering wheel orientation given by the driver δ_{sw} corresponds to a desired steering angle, δ_d that can be translated as a desired yaw rate $\dot{\psi}_d$. The latter is fed to the steady state cornering map together with the vehicle speed, V , allowing for the definition of the desired state $x_d(\dot{\psi}_d, V)$ and the corresponding feedforward gain $u_d(\dot{\psi}_d, V)$ at any time t (Fig 6.4). Again the feedback gain scheduling is applied on the speed only.

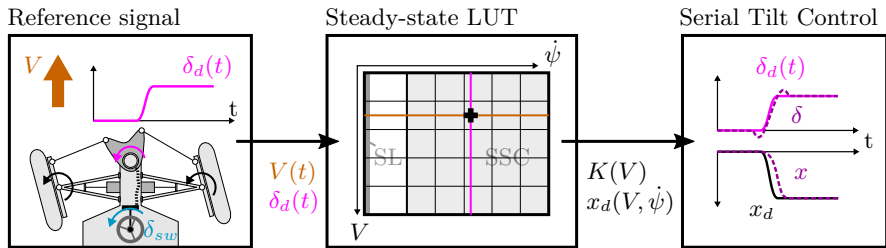


Figure 6.4: Serial tilt control principle. Based on the steering fork orientation δ_d and the forward speed, V , the desired state, x_d is computed from a lookup table (LUT) linear interpolation.

For serial modes, the control involves both a feedforward and a feedback term.

As stated above, the steady-state equilibrium in curves necessitates a torque on the steering system (STC/SDTC) or the tilting system (LTC).

6.1.3 LQR results

The last degrees of freedom left for LQR controller are the gain matrices R and Q in equation (6.4). Diagonal matrices have been considered with four nonzero entries: Q_ϕ , $Q_{\dot{\phi}}$, R_{tilt} , R_{steer} , the latest being inexistent for a DTC control. To ensure that the SDTC uses both the steering and tilting actuations, two orders of magnitude are set between R_{tilt} and R_{steer} based on the square of admissible values, i.e., 10 Nm for the steering action and 100 Nm for the tilting action (see the steady state torques in Section 4.1). Assuming an admissible error of 10^{-2} radians for the lean angle, 4 orders of magnitude have been set between the steering torque gain, R_{steer} and the tilt angle gain, Q_ϕ . Regarding the gain on the tilt angle rate, $Q_{\dot{\phi}}$, it has been set by trial and error in order to ensure a stable behavior on the whole speed range. The values are the following:

- $Q_\phi = 10^4$, $Q_{\dot{\phi}} = 10^2$, $R_{tilt} = 10^{-2}$, $R_{steer} = 1$

Closed-loop stability in straight line

Before considering any time simulation, the stability of the controller can be verified beforehand by realizing a similar stability analysis as for the free vehicle (Section 4.2). By performing a series of modal analyses at different advance speeds V from 0 to 15 m/s, the closed-loop stability can be assessed for the different feedback controllers, i.e., STC, LTC, SDTC and DTC mode.

First, the serial controllers are compared to the free NTV case with a free steering dynamics (Fig. 6.5 and 6.6). The LQR control stabilizes the vehicle as the real part of the weave mode is negative for the whole speed range (Fig. 6.5).

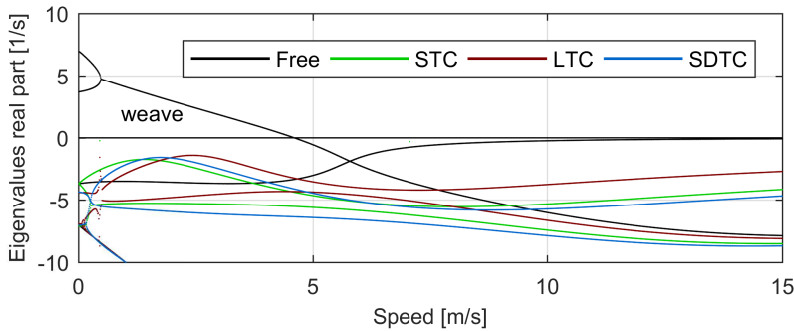


Figure 6.5: Closed-loop stability analysis of the NTV in straight line for the 3 serial modes. The 3 controllers stabilize the weave mode at low speed with negative real parts for the eigenvalues.

The closed-loop analysis predicts a similar stability behavior for all serial modes with the real part of the eigenvalues being maximum for a speed between 2

and 3 m/s. Note that tuning the LQR gains for the different speeds could lead to an improved stability at low speed. Generally, other linear methods could also be used to that extent (e.g., pole placement). On the other hand, the serial controllers all lead to a larger imaginary part when compared to the free NTV case (Fig. 6.6). The STC and SDTC inherit a larger imaginary part than the LTC mode which might cause extra oscillations for these controllers. This should be investigated via time simulation.

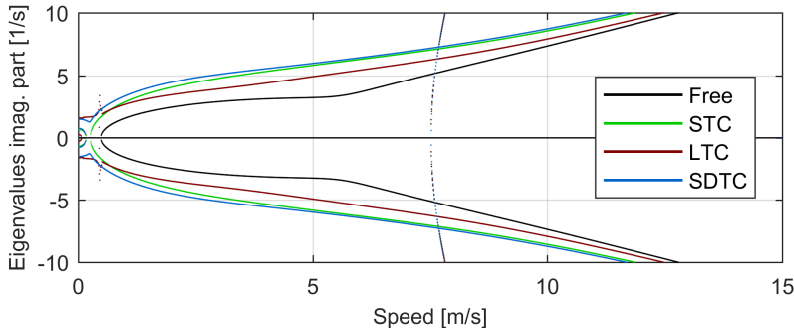


Figure 6.6: Closed-loop stability analysis of the NTV in straight line for the 3 serial modes. The 3 controllers amplify the weave mode oscillatory nature with larger imaginary part observed for all speeds.

Regarding the parallel control (i.e., the DTC), it is compared with respect to the *locked steer angle* case for which the steering angle is kept to a zero value canceling the influence of the steering dynamics (Fig. 6.7). The DTC stabilizes the locked-steer dynamics with eigenvalue real part being negative and almost constant on the whole speed range.

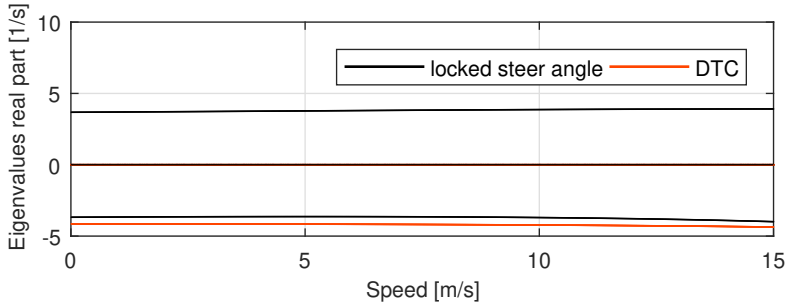


Figure 6.7: Closed-loop stability in straight line for the DTC. The control on the tilt d.o.f. stabilizes the locked steer vehicle which initially behaves like an inverted pendulum with an unstable roll mode.

The closed-loop stability analysis shows that the gain-scheduling LQR should stabilize the vehicle at any speed for all modes. However, the analysis is limited to the straight line case. The cornering stability will be investigated via two different time simulations: a curve entry and a variable speed maneuver scenario.

Curve entry

In order to assess the performances of the LQR controller, a curve entry is realized on the form of a step input for the desired steady state yaw rate, ψ_d . The step is realized as a smooth step⁴ over a time period of 1 second. After that time, the vehicle is supposed to be in a curve of 20 meters at a velocity of 10 m/s which corresponds to a yaw rate of 0.5 rad/s. The four modes manage to achieve the control task and reach the steady state after a certain time (Fig. 6.8). The DTC is performing the best with an almost perfect following of the yaw rate, which can be explained by the fact that the steering dynamics is driven (Fig. 6.8.a and c). The STC leads to a non-minimum phase (NMP) behavior (i.e., negative yaw rate) and an overshoot for the yaw rate, while the LTC only generates a small overshoot. The SDTC offers a trade-off between the STC and the LTC with a reduced NMP behavior and a reduced overshoot (Fig. 6.8.a and c). With regard to the lean angle, the STC responds the fastest while the SDTC offers again a trade-off between the STC and LTC (Fig. 6.8.b). The DTC, although it is more direct than the STC on the roll motion, does not exhibit a faster response.

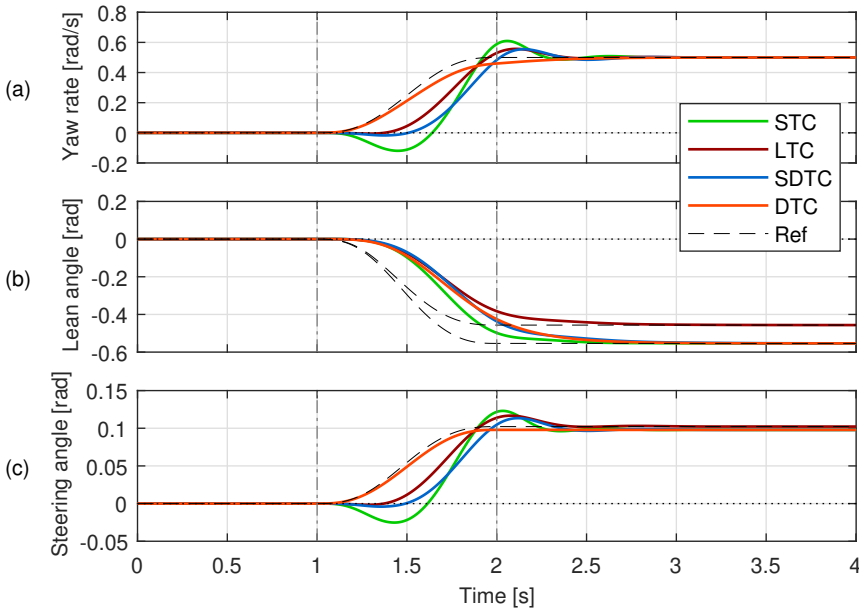


Figure 6.8: Curve entry ($V = 10$ m/s, $R = 20$ m). The DTC leads to a better tracking performance than the 3 other serial modes.

Regarding the actuation, the STC, SDTC and DTC all lead to a zero tilting torque in steady state while the LTC does not, as suspected (Fig. 6.9). The DTC

⁴The smooth step is realized with the help of a 5th-order polynomial function as suggested in [91].

leads to the highest input torque both for the controller tilting motion and the forced steering kinematic⁵. In addition, a small overshoot in the steering torque is necessary to properly drive the steering angle most likely due to the forced tilting motion (Fig. 6.9.b). Both the STC and LTC also exhibit an overshoot in their respective torque input. On the other hand, the combined SDTC is properly combining the tilting and steering action and drastically reduces the tilting torque input compared to the DTC and LTC modes (Fig. 6.9.a). Moreover, the steering input for the SDTC is very close to the steady state steering torque value (dashed lines), meaning that the steering feedforward term dominates in the transient phase for the SDTC.

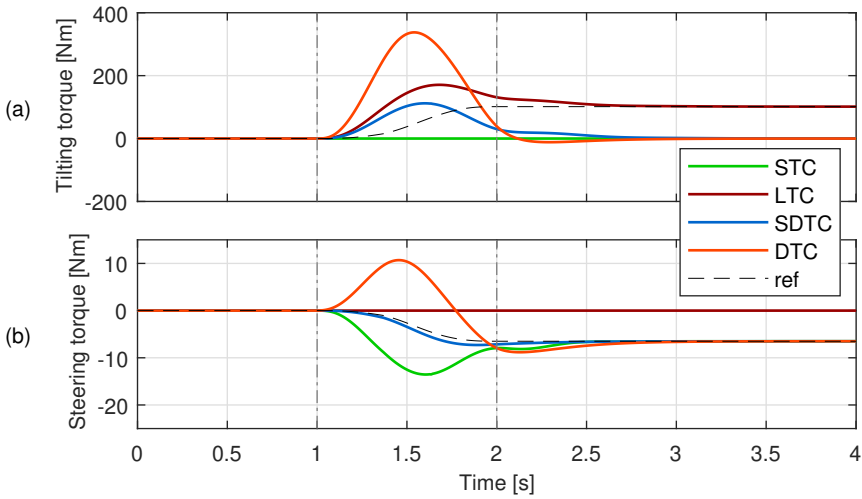


Figure 6.9: Curve entry ($V = 10$ m/s, $R = 20$ m). The DTC leads to a larger tilting and steering actuation than the other modes while the SDTC is a trade-off between STC and LTC.

If the DTC offers better yaw rate tracking, which is what counts for drivability, its nature and its working principle cause a large wheel load transfer during the transient phase (Fig. 6.10). The wheel load transfer almost reaches 100 % corresponding to the inner wheel lifting, which highlights the limitation of the DTC in transient maneuvers. A more dynamic curve entry (i.e., a step of half a second) was performed for the 4 modes and while the 3 serial modes were able to achieve the maneuver, the DTC failed and wheel lifting is observed.

In addition, the transverse perceived acceleration (Section 4.1.2), can be measured at the driver's head level during the curve entry (Fig. 6.11). The DTC leads a transverse perceived acceleration twice as large as the LTC and SDTC mode during the transient maneuver. The STC exhibits a smaller peak but a higher overshoot which might not be appreciated in terms of driving experience.

⁵When a joint of the MBS system is driven kinematically, the necessary force/torque to induce this motion can be retrieved. More details can be found in [67, 73].

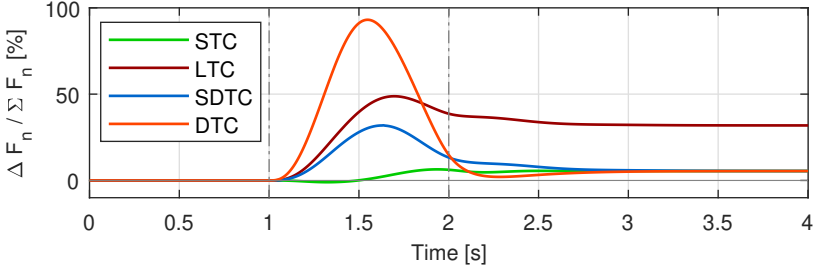


Figure 6.10: Curve entry ($V = 10$ m/s, $R = 20$ m). The load transfer is critical for the DTC mode.

Regarding the steady state cornering condition, the fact that the perceived acceleration cancels out for the LTC is not general but only specific to the given speed ($V = 10$ m/s) and radius ($R = 20$ m) of the maneuver (Section 4.1).

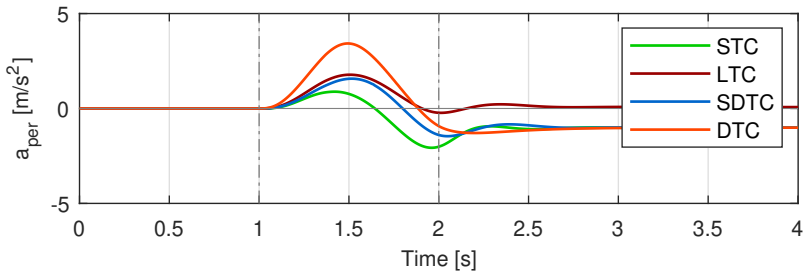


Figure 6.11: Curve entry ($V = 10$ m/s, $R = 20$ m). The DTC leads to a peak twice as large in the transverse perceived acceleration (head level) compared to the serial control modes.

Speed-varying maneuver

As explained in the previous section, the LQR approach is combined with a gain scheduling on the speed parameter in order to stabilize the vehicle on the whole speed range. In order to verify the stability observed via the closed-loop analysis in straight line, a special tracking task was imagined to test the controller at high and low speed. The tracking signals are the curve radius and the speed which, once combined, determine a desired yaw rate $\dot{\psi}_d$ and steady state configuration to follow (Fig. 6.12).

The forward speed, V is tracked via a propulsion torque which is computed as a proportional feedback controller.

All the 4 modes manage to track the yaw rate during the whole maneuver (Fig. 6.13). At high speed, the 4 modes work correctly but when the speed is reduced, the serial modes (STC, LTC, SDTC) lead to oscillations both in the tracked

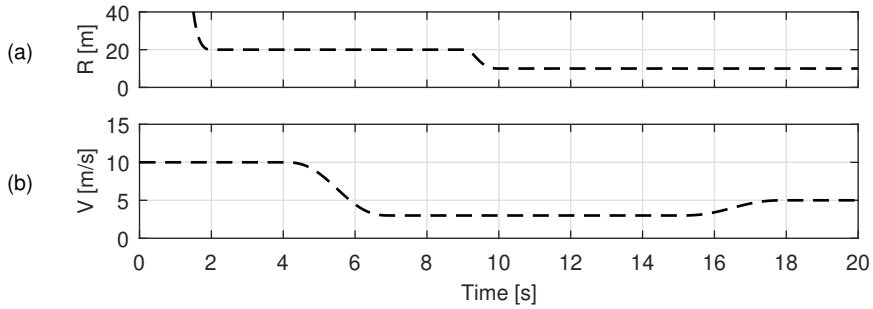


Figure 6.12: A special tracking scenario is defined by the curve radius (a) and the vehicle speed (b), which correspond to a desired yaw rate, $\dot{\psi}_d$ for the 20-second time range.

states and their inputs. The STC leads to the highest oscillations in accordance with its free-to-roll nature which is known to be less stable (Section 4.2). On the other hand, the DTC mode manages to properly track the yaw rate and does it without causing any oscillation. This is in accordance with the closed-loop stability analysis and the imaginary part of the weave mode (Fig. 6.6), which exists for the serial modes because the steering dynamics is unconstrained.

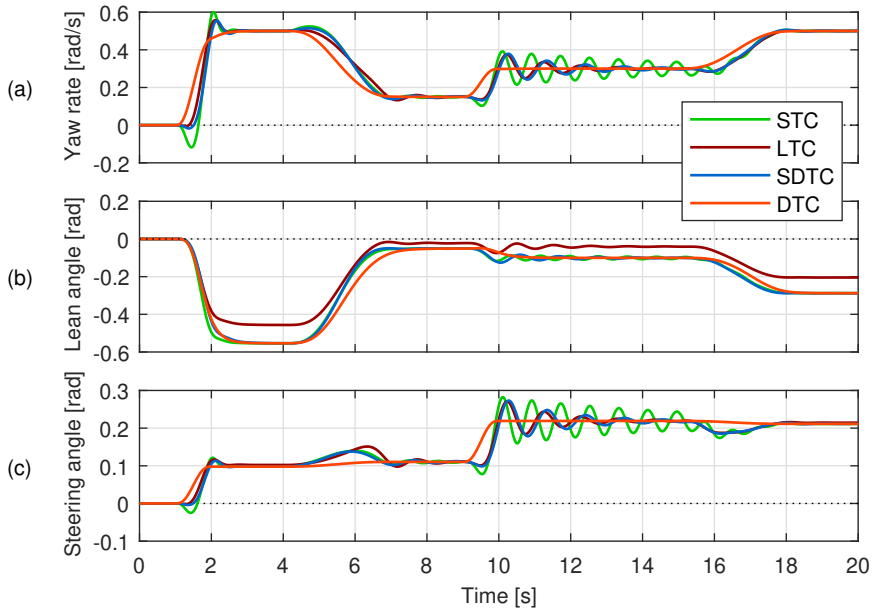


Figure 6.13: Varying speed tracking scenario. From $t = 8$ s to $t = 14$ s, at the lowest speed, the serial modes exhibit oscillations in the different tracked states that are associated to the weave mode.

6.2 Feedback linearization

In this section, another nonlinear control technique, called feedback linearization, is considered for the control of NTVs. The main underlying principle is to apply a nonlinear input function, $u(x)$ that transforms the initial nonlinear system into an equivalent linear system (Fig. 6.14). In particular, the focus is on the input-output linearization where the input-output map is linearized, while the state equations may be only partially linearized.

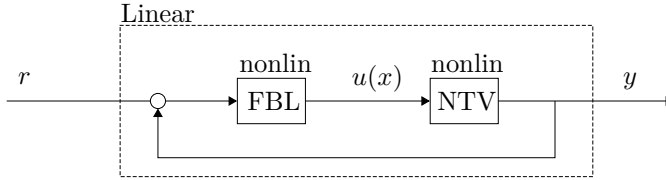


Figure 6.14: Block-diagram representation of the input-output feedback linearization.

In this section, the theoretical basis of the feedback linearization (FBL) is presented and applied for the case of multibody systems. Most of the developments and notations are based on [92]. Then, the method is considered for the four control modes already treated in the previous section. First, the feasibility of FBL is tested for each mode. Then, the performances of FBL are observed by comparing it with the LQR controller developed in the previous section.

6.2.1 Feedback linearization of multibody systems

Let us consider a generic nonlinear system with a single input, u and a single output, y :

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad (6.13)$$

where f , g and h are sufficiently smooth in a domain $D \subset R^n$. The nonlinear system has a given relative degree, ρ defined as the minimum order of the output time derivative that is affected directly by the input u . A mathematical definition can be found in Appendix (D) and in reference [92].

As shown in Chapter 3, the MBS equations of motion can be generally expressed as a set of nonlinear equations similar to the system (6.13). The input of the MBS system is generally a subset of the active joint force vector, $Q_{u,a}$. The choice of the controlled joints is defined by the selection column, N , which has

the size $[n_{qu} \times 1]$ in case of a SISO system (Section 3.2).

$$Q_{u,a} = N u = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} u \quad (6.14)$$

As shown in Chapter 3 (Section 3.2), the second-order ODE for the equations of motion can be reduced to a first order ODE system by gathering the independent position q_u and independent velocities $\dot{q}_u$ into the state vector x .

$$x = \begin{pmatrix} q_u \\ \dot{q}_u \end{pmatrix} \rightarrow \dot{x} = \begin{pmatrix} \dot{q}_u \\ \ddot{q}_u \end{pmatrix} = \begin{pmatrix} \dot{q}_u \\ -M_r^{-1} [F_{r,p}(\dot{q}_u, q_u, Q_{u,p}) - Q_{u,a}] \end{pmatrix} \quad (6.15)$$

By separating the input part from the rest of the dynamics, expressions for x , $f(x)$ and $g(x)$ can be obtained for the case of a MBS:

$$x = \begin{pmatrix} q_u \\ \dot{q}_u \end{pmatrix} \quad f(x) = \begin{pmatrix} \dot{q}_u \\ -M_r^{-1} F_{r,p} \end{pmatrix} \quad g(x) = \begin{pmatrix} 0 \\ M_r^{-1} N \end{pmatrix} \quad (6.16)$$

The output of the system is generally expressed as a nonlinear function $h(x)$ of the system state (i.e., the positions and velocities).

Input-output linearization

Considering the generic single-input-single-output (SISO) system in equation (6.13), the goal of input-output linearization is to find a nonlinear command $u(x)$ that results in an equivalent linear map between a new synthetic input, v and the output, y . In order to derive the expression of $u(x)$, the output expression, $h(x)$ is derived ρ -time with respect to time, using Lie derivatives [62].

$$y^\rho = \frac{d^\rho y}{dt^\rho} = L_f^\rho h + L_g(L_f^{\rho-1} h)u \quad (6.17)$$

where $L_f^\rho h$ is the Lie-derivative of the function h with respect to the function f as defined in Appendix D. Then, two auxiliary expression can be defined, $\alpha(x)$ and $\beta(x)$.

$$\alpha(x) = L_f^\rho h(x) \quad (6.18)$$

$$\beta(x) = L_g L_f^{\rho-1} h \quad (6.19)$$

By defining the *synthetic* input, $v = \alpha(x) + \beta(x)u$, the nonlinear system defined in (6.13) is partially reduced to the linear input-output map in equation (6.20).

$$y^{(\rho)} = \frac{d^\rho y}{dt^\rho} = v \quad (6.20)$$

The expression of the nonlinear control, $u(x)$ is simply obtained by solving for equations (6.17) and (6.20).

$$u(x) = \gamma(x)(v(x) - \alpha(x)) \quad (6.21)$$

with

$$\gamma(x) = \beta^{-1}(x) \quad (6.22)$$

Note that there is a chain of ρ integrators in equation (6.20). Thus, the linear map is not direct between the new input v and the output y and additional states need to be defined (Fig. 6.15).

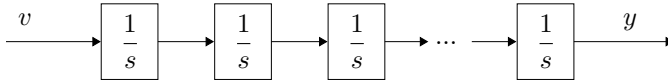


Figure 6.15: Block-diagram representation of the chain of integrators for an input-output feedback linearization.

The development of the expressions $\alpha(x)$ and $\beta(x)$ depends on the relative degree of the nonlinear system. Generally, mechanical systems are either of relative degree, $\rho = 1$ or $\rho = 2$. For the case of a linear output, $y = Cx$ (e.g., the vehicle lean angle) and considering a relative degree of 2, the expressions for $\alpha(x)$ and $\beta(x)$ are generally written in equations (6.23) and (6.24) for a MBS.

$$\alpha(x) = C \frac{\partial f}{\partial x} f(x) \quad (6.23)$$

$$\beta(x) = C \frac{\partial f}{\partial x} g(x) \quad (6.24)$$

with $f(x)$ and $g(x)$ that are known for MBS and the partial derivative of $f(x)$ expressed as follows:

$$\frac{\partial f}{\partial x} = A = \begin{bmatrix} 0 & I \\ -M_r^{-1}K_r & -M_r^{-1}G_r \end{bmatrix} \quad (6.25)$$

More details on Lie derivatives and how to obtain these expressions are given in Appendix D and [92][62].

Stability

Thanks to the linearization, the dynamics of the output can be prescribed by appropriately choosing v in equation (6.20). The input-output map can be easily stabilized for example by choosing $v = -Ky$ with $-K$ being Hurwitz. However, the main issue of input-output linearization is that only the external

part which is linearized is observable while the internal part is not. Generally, stability is studied by applying a change of variables, $z = [\xi, \eta]$ where the external input-output dynamics is represented by the states ξ which account for the chain of ρ integrators and the internal dynamics which is represented by the η variables. Thanks to the change of variables, the normal form in (6.26) can be obtained for the nonlinear system [62].

$$\begin{cases} \dot{\eta} = f_0(\eta, \xi) \\ \dot{\xi} = A_c \xi + B_c \gamma(x)[u - \alpha(x)] \\ y = C_c \xi \end{cases} \quad (6.26)$$

where $\xi \in R^\rho$, $\eta \in R^{r-\rho}$ and (A_c, B_c, C_c) is a canonical form representation of the chain of ρ integrators.

$$A_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \vdots & & 0 & 1 \\ 0 & \dots & 0 & 0 \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}, \quad C_c = [1 \quad 0 \quad \dots \quad 0] \quad (6.27)$$

The internal dynamics is described by f_0 and is very difficult to express in practice [62]. Setting $\xi = 0$ for the internal dynamics leads to the so-called zero dynamics in equation (6.28).

$$\dot{\eta} = f_0(\eta, 0) \quad (6.28)$$

The nonlinear system is said to be *minimum phase* if the zero dynamics in equation (6.28) has an asymptotically stable equilibrium point in the domain of interest. Thus, in conclusion, when feedback linearization is considered to control a MBS system, it is primordial to investigate the stability of the zero dynamics. In fact, stabilization or tracking of the output will not ensure that the internal part of the system is stable.

Tracking

The main application of feedback control is the tracking of a given reference signal, $r(t)$ for the output, y . In the following, the reference signal and its derivatives to the ρ^{th} -order are gathered in vector, ϱ and supposed available on-line at any time t .

$$\varrho = \begin{bmatrix} r \\ \vdots \\ r^{(\rho-1)} \end{bmatrix} \quad \xi = \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_\rho \end{bmatrix} \quad (6.29)$$

By defining the error vector $\varepsilon = \xi - R$ and replacing in system (6.26), a new representation can be obtained for the error dynamics:

$$\dot{\eta} = f_0(\eta, \varepsilon + \varrho) \quad (6.30)$$

$$\dot{\varepsilon} = A_c \varepsilon + B_c v \quad (6.31)$$

with the control input $u(x)$ that now needs to include an additional term $r^{(\rho)}$.

$$u(x) = \alpha(x) + \beta(x) \left(v + r^{(\rho)} \right) \quad (6.32)$$

Again, v can be chosen in order to define a given dynamics for the error vector ε , typically $v = -Ke$, such that the system is stable. FBL allows us to choose the evolution of $e^{(\rho)}$ via v but does not enable the direct definition of the error, e as there is still a chain of integrators between the system synthetic input, v and the error, e (Fig. 6.15).

Multiple input and multiple output (MIMO) system

Although the previous sections focused on a SISO system, the feedback linearization can also be performed for a MIMO system. Considering a *square* system for which $m = n_y = n_u$, the system can be written as follows:

$$\begin{cases} \dot{x} = f(x) + \sum_{i=1}^{n_u} g_i(x) u_i \\ y = [h_1(x) \dots h_{n_y}(x)]^T \end{cases} \quad (6.33)$$

Given the n_y output, there is one relative degree ρ_k for each of the k output. Based on that, a $m \times m$ matrix called the *invertibility matrix* can be written using Lie derivatives [92].

$$J(x) = \begin{bmatrix} L_{g_1}(L_f^{\rho_1-1} h_1) & \dots & L_{g_m}(L_f^{\rho_1-1} h_1) \\ \vdots & \ddots & \vdots \\ L_{g_1}(L_f^{\rho_m-1} h_m) & \dots & L_{g_m}(L_f^{\rho_m-1} h_m) \end{bmatrix} \quad (6.34)$$

This matrix reduces to $\gamma(x)$ for the SISO case and is assumed to be non-singular. In addition, one can see that each line k of this matrix will depend on the relative degree ρ_k of the corresponding output.

Let us define two new column vectors, Υ and Λ , both of size $m \times 1$.

$$\Upsilon = \begin{bmatrix} \frac{d^{\rho_1} y_1}{dt^{\rho_1}} \\ \vdots \\ \frac{d^{\rho_m} y_1}{dt^{\rho_m}} \end{bmatrix} \quad \Lambda(x) = \begin{bmatrix} L_f^{\rho_1}(h_1) \\ \vdots \\ L_f^{\rho_m}(h_m) \end{bmatrix} \quad (6.35)$$

By differentiating the expression of the output y in equation (6.33) and using the definition of Υ , Λ and J , the following equation (6.36) can be obtained.

$$\Upsilon = \Lambda(x) + J(x)u \quad (6.36)$$

In a similar manner as for the SISO case, a control input $u(x)$ can be determined such that the system (6.36) results in a set of n_y decoupled input-output maps for which v is the synthetic input, a vector of size $m \times 1$:

$$\begin{bmatrix} \frac{d^{\rho_1} y_1}{dt^{\rho_1}} \\ \vdots \\ \frac{d^{\rho_m} y_1}{dt^{\rho_m}} \end{bmatrix} = \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} \rightarrow \Upsilon = v \quad (6.37)$$

The expression of $u(x)$ is simply obtained by solving for equations (6.36) and (6.37).

$$u(x) = J^{-1}(v - \Lambda) \quad (6.38)$$

The v vector can be designed in any way to achieve the given dynamics for the output. The stability and the tracking considerations in the aforementioned sections for SISO systems can directly be applied to a MIMO system keeping in mind that each decoupled input-output map has its own chain of ρ_k integrators and its own error vector, ε_k .

6.2.2 Feedback linearization for NTV control

Based on the above theoretical development of the previous section, the feedback linearization (FBL) is applied for the control of NTV. Firstly, the FBL of the single input modes, i.e., STC, LTC and DTC, is discussed and then separately compared to the LQR implementation. Secondly, the multiple input control mode, i.e., SDTC is studied.

SISO: STC - LTC - DTC

As mentioned in the previous section, it is primordial that the internal dynamics of the system is stable while performing an input-output linearization. Verifying the internal stability of a nonlinear system is generally quite complicated [62]. Instead, it can already be insightful to study the non-minimum phase (NMP) behavior of the different modes around the upright position by considering the linear representation of the vehicle at a given speed, V .

Feedback linearization implies the choice of one output for the controller. In the past, many investigations were based on the control of the lean angle [12][13] but the vehicle yaw rate could also be an interesting output for the system as it could be more reflective of the driver's intent. For a forward speed of 10 m/s, the transfer function between the control input (i.e., the tilting or the steering torque) and the lean angle, ϕ can be computed and a pole-zero map can be obtained (Fig. 6.16). All the zeros are in the left half of the complex plane meaning that the system is minimum phase and that the lean angle could be controlled for the envisaged forward velocity.

For the serial mode (STC and LTC), a similar pole-zero map is computed for the yaw rate, $\dot{\psi}$ as the output (Fig. 6.17). For both modes, the transfer function exhibits a zero in the right half-plane meaning that the system is non-minimum phase and could not be controlled via feedback linearization [62]. Logically, prescribing a given trajectory for the yaw rate, while leaving the roll dynamics as part of the internal dynamics, will cause an unstable behavior for the whole speed range, similarly to what is observed for an inverted pendulum.

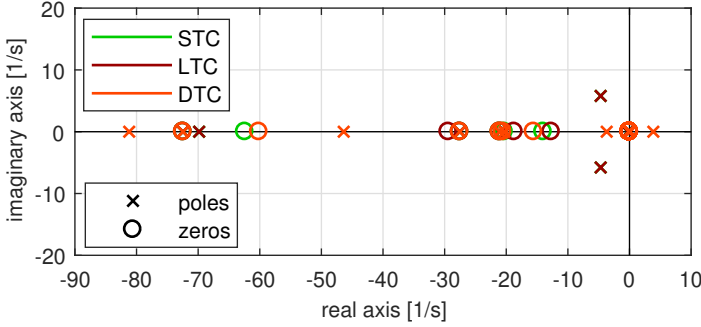


Figure 6.16: Pole-zero map for the different SISO modes considering the lean angle, ϕ as an output. The NTV dynamics is linearized around an upright position and a forward velocity, $V=10$ m/s.

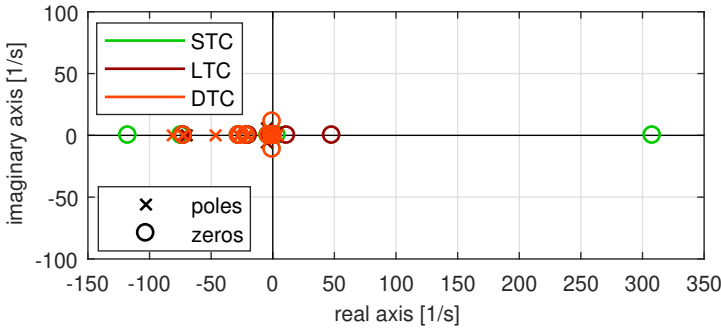


Figure 6.17: Pole-zero map for the different SISO modes considering the yaw rate, $\dot{\psi}$ as an output. The NTV dynamics is linearized around an upright position and a forward velocity, $V=10$ m/s.

Lean angle control should always stabilize the vehicle roll motion. However, this does not ensure that the vehicle is stable for the whole speed as instabilities could come from the steering assembly dynamics which is part of the NTV internal dynamics. Although it does not exactly correspond to the *zero dynamics* as defined in equation (6.28), the straight line steady state with the tilt angle locked to zero, $\theta = 0$, could account for the internal dynamics considering FBL lean control. Based on a series of modal analyses, the vehicle stability with the locked tilt angle is easily analyzed (Fig. 6.18). The vehicle with locked tilt angle is generally more stable than the free vehicle with a self-stable speed of 2.25 m/s instead of 5 m/s. Nevertheless, it is important to notice that, at low speed, the steering and lateral dynamics are unstable with a mode characterized by a lateral deviation and an increasing yaw angle. Such observation restricts the use of feedback linearization to the higher speeds for the considered NTV.

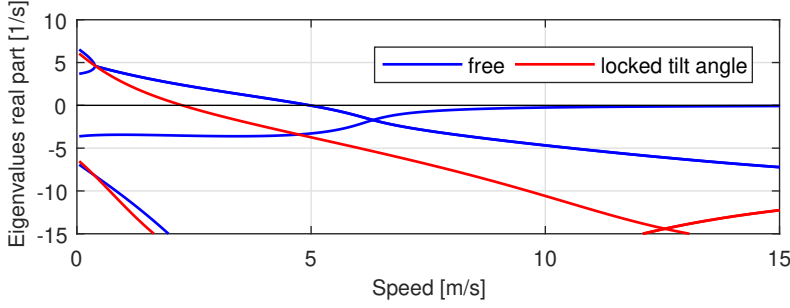


Figure 6.18: Stability analysis of the NTV in straight line with locked tilt angle, $\theta = 0$. The vehicle is self-stable above a speed of 2.25 m/s. The unstable mode is a lateral yaw motion.

On the other hand, DTC via feedback linearization (FBL) stabilizes the vehicle on the whole speed range as, in that case, the steering dynamics is not free and supposedly controlled by the driver. The roll dynamics should be easily stabilized by the tracking of the vehicle lean angle in a similar way as the LQR controller does (Fig. 6.3).

In order to verify FBL applicability for NTV control, time simulations are performed for the different control modes. Considering the lean angle, ϕ as the output leads to a relative degree equal to two, $\rho = 2$. In fact, as the output y is part of the independent coordinate q_u , it only needs to be differentiated twice to obtain the independent acceleration $\ddot{q}_u$ whose expressions directly feature the input torque, u . In that case, the input-output map involves a chain of 1 integrator and the error, ε is a two by one vector: $\varepsilon = [e_\phi \ \dot{e}_\phi]$. Considering a simple linear feedback law, $v = -K\varepsilon$ and developing equation (6.26), the following expression can be obtained for the roll angle error dynamics:

$$\ddot{e}_\phi + K_{12}\dot{e}_\phi + K_{11} = 0 \quad (6.39)$$

For the above-mentioned SISO controllers, the following values for the gain have been chosen: $K_{11} = 2 \cdot 10^2$ and $K_{12} = 1 \cdot 10^4$, resulting in two equal real negative roots⁶, $\lambda_{1,2} = -100$ for the error dynamics.

FBL is considered for each of the 3 SISO control modes and compared separately to its LQR counterpart. The maneuver is the same as in the previous section, i.e., a curve entry at a speed of 10 m/s and for a curve radius of 20 meters.

Steering tilt control Feedback linearization is successful for the NTV control of STC and performs better than its LQR counterpart (Fig. 6.19). FBL leads to a more reactive control with a sudden jump in the steering torque at both the beginning and the end of the maneuver⁷ ($t=1$ s and $t=2$ s).

⁶There is no interest in considering more negative roots as it did not influence the accuracy of the controller.

⁷Note that these glitches in the steering torque are *partly* due to the definition of the reference trajectory that inherits a continuous but non-smooth second derivative at $t = 1$ [91] which directly influences the input as shown in equation (6.32).

The steering torque for FBL exhibits a similar shape to that of the LQR one but is applied directly at $t = 1$ s (Fig. 6.19.a). With regard to the lean angle, the FBL is able to perfectly track the desired trajectory with a maximal tracking error of 10^{-3} radians⁸ (Fig. 6.19.b). In accordance with the faster steering torque, the countersteering is operated directly at time $t=1$ and is slightly larger than for the LQR case (Fig. 6.19.c).

Simulations at low speed have also been performed showing that the STC was able to stabilize the NTV but at a velocity no lower than 2.15 m/s approximately. This is in accordance with what was predicted via the straight line stability analysis with locked tilt angle (Fig. 6.18).

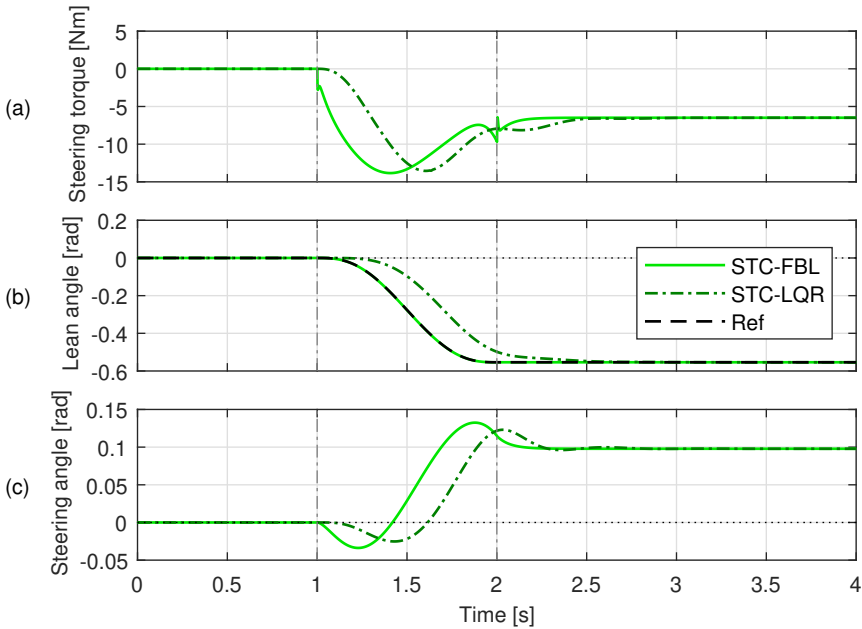


Figure 6.19: Curve entry performed with STC ($V = 10$ m/s and $R = 20$ m). FBL for STC leads to better tracking of the lean angle and a faster response than its LQR counterpart.

Leaning tilt control Feedback linearization for LTC is successful and leads again to a faster control than its LQR counterpart (Fig. 6.20). The tilting torque is applied directly when the maneuver starts and has a similar shape to that of the LQR controller (Fig. 6.20.a). However, the maximum tilting torque is higher than for the LQR case which will logically cause a higher wheel load transfer and thus will reduce the lateral stability during the transient maneuver. The lean angle is well tracked by the FBL controller with again a maximum error of about 10^{-3} radians (Fig. 6.20.b). The faster tilting actuation leads to a

⁸The tracking error is meaningful since the integration time step is fixed to $1e-5$ s in order to avoid numerical errors as stated in Section 4.3.1.

particular profile for the yaw rate with a slight NMP behavior at the beginning and a larger overshoot at the end of the maneuver (Fig. 6.20.c). This fast yaw rate raise can be translated as a more nervous response of the vehicle to the reference lean angle which, in practice, could be associated to the driver input.

Simulations at low speed were performed and resulted in an inadequate behavior of the vehicle going in the opposite direction. The lean angle was correctly tracked but, at very low speed, the dynamic effects causing the countersteering are not large enough and the steering assembly could not be dynamically steered in the right direction. The vehicle is laterally unstable at low speed ($V < 2.15$ m/s) as predicted by the stability analysis with the locked tilt angle (Fig. 6.18).

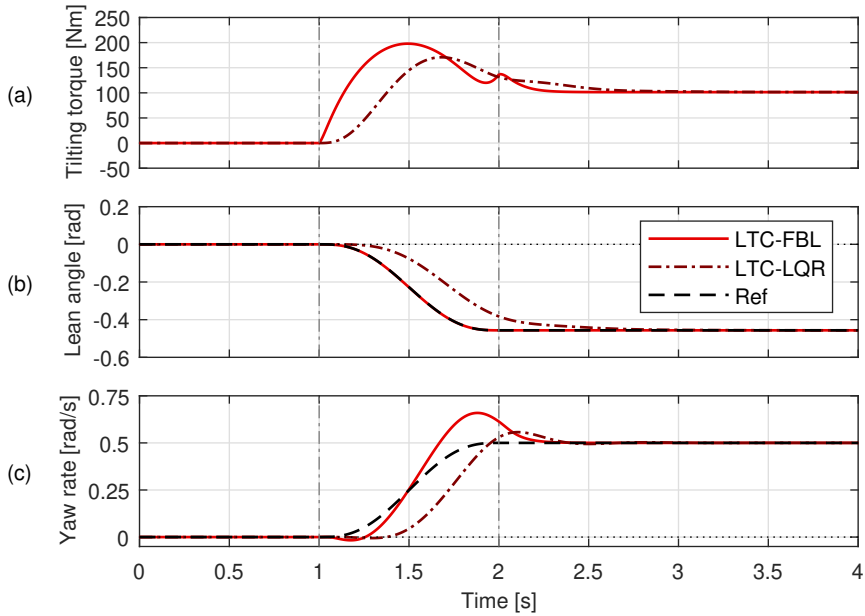


Figure 6.20: Curve entry performed with LTC ($V = 10$ m/s and $R = 20$ m). FBL for LTC leads to a faster response and a better tracking of the lean angle than its LQR counterpart with a faster rise in yaw rate (reflecting a more nervous vehicle).

Direct tilt control Feedback linearization can also be applied to the DTC and improves the NTV control compared to the LQR approach (Fig. 6.21). Regarding the tilting torque, the FBL leads to a completely different actuation than LQR during the transient phase with first a positive and then a negative torque (Fig. 6.21.a). Compared to the LQR case, it seems that the FBL thoroughly takes advantage of its nonlinear nature to anticipate the necessary torque to perfectly track the vehicle lean angle. The peak tilting torque is reduced by about 30% compared to the LQR case which corresponds to a lower energy consumption in the transient phase (Fig. 6.21.a). For the lean angle, the

error dynamics is again well tracked with an error no larger than 10^{-3} radians (Fig. 6.21.b). Eventually, the lower tilting torque also leads to a reduced wheel load transfer compared to the LQR case (Fig. 6.21.c). Generally, the faster the DTC control, the lower the tilting torque will be. This is in accordance with the DTC nature for which the vehicle is first steered by the driver and then tilted accordingly *in parallel*. It is important to notice that such a *bang-bang* tilting action cannot be achieved with a simple proportional control or by just considering larger gain in the LQR control. As a matter of fact, the FBL takes advantage of the available nonlinear model to track the lean angle as fast as possible, which consequently reduces the dynamic delay between the steering and tilting dynamics.

By nature the DTC can be performed via feedback linearization on the whole speed range. Simulations at low speed have been carried out, confirming this intuition.

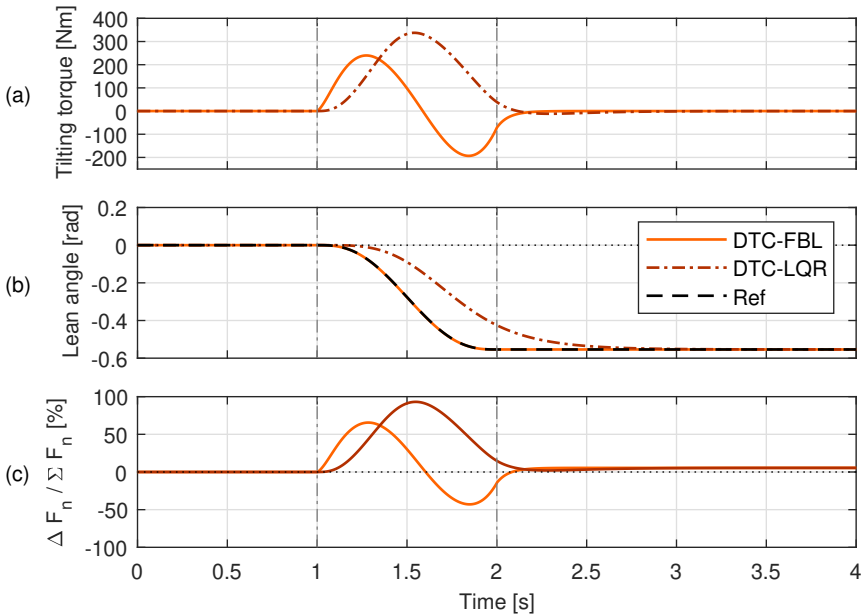


Figure 6.21: Curve entry performed with DTC ($V = 10$ m/s and $R = 20$ m). Feedback linearization for DTC leads to a reduction of the tilting torque during transient maneuver while compared to the LQR control as FBL takes full advantage of the nonlinear nature of the vehicle roll dynamics.

MIMO: combined steering and tilting controls (SDTC)

As developed in Section 6.2.1, feedback linearization can also be considered for MIMO systems and thus for combined steering and tilting control (SDTC). While the lean angle clearly needs to be controlled, the choice of the second

output is less trivial. Two quantities come to mind: the steering angle, δ and the yaw rate, $\dot{\psi}$.

It was shown above that controlling the yaw rate, $\dot{\psi}$ might not be simple as its transfer function is non-minimum phase (NMP) for both a steering and a tilting input. In fact, combining both steering and tilting to control the yaw rate is not feasible for the studied NTV as a slight NMP behavior is still observed in the yaw rate. This behavior could possibly be explained by the front suspension geometry which associates to the beginning of the tilt motion a slight yaw motion in the outward direction. Nevertheless, it is still possible to perform SDTC based on the steering angle, δ as the second output. In that case, the reference signal will be simply the smooth steer ramp which was applied for the DTC (supposedly by the driver).

Controlling the roll and steer angle, which are both part of the vector q_u , implies 2 separate chains of 1 integrator for the SDTC ($\rho = 2$ for both output). In that case, the error vectors are defined as follows:

$$\varepsilon_\phi = [e_\phi \ \dot{e}_\phi] \quad (6.40)$$

$$\varepsilon_\delta = [e_\delta \ \dot{e}_\delta] \quad (6.41)$$

The left degrees of freedom are the v_1 and v_2 that directly influence the second derivatives of the two errors:

$$\ddot{e}_\phi = v_1 \quad (6.42)$$

$$\ddot{e}_\delta = v_2 \quad (6.43)$$

Linear feedback law The first approach to design the SDTC via feedback linearization is to consider a simple linear feedback law, $v_i = -K_i \varepsilon_i$ for each of the output, which results in the following error dynamics:

$$\ddot{e}_\phi + K_{\phi,12} \dot{e}_\phi + K_{\phi,11} = 0 \quad (6.44)$$

$$\ddot{e}_\delta + K_{\delta,12} \dot{e}_\delta + K_{\delta,11} = 0 \quad (6.45)$$

The same values as for the SISO modes have been chosen for the gains: $K_{11} = 2 \cdot 10^2$ and $K_{12} = 1 \cdot 10^4$ and applied to both error dynamics. The same maneuver as before is considered (i.e., the curve entry at $V=10$ m/s) and the results of the feedback linearization (FBL) are compared to the LQR case (Fig. 6.22). By tracking both the steering and the lean angle, FBL leads to a more important tilting and steering actuation compared to the LQR case (Fig. 6.22.a and b). In reality, the SDTC as implemented via FBL corresponds more to the DTC mode for which the steering dynamics was prescribed and the tilting torque exhibits two peaks in the transient phase (Fig. 6.21).

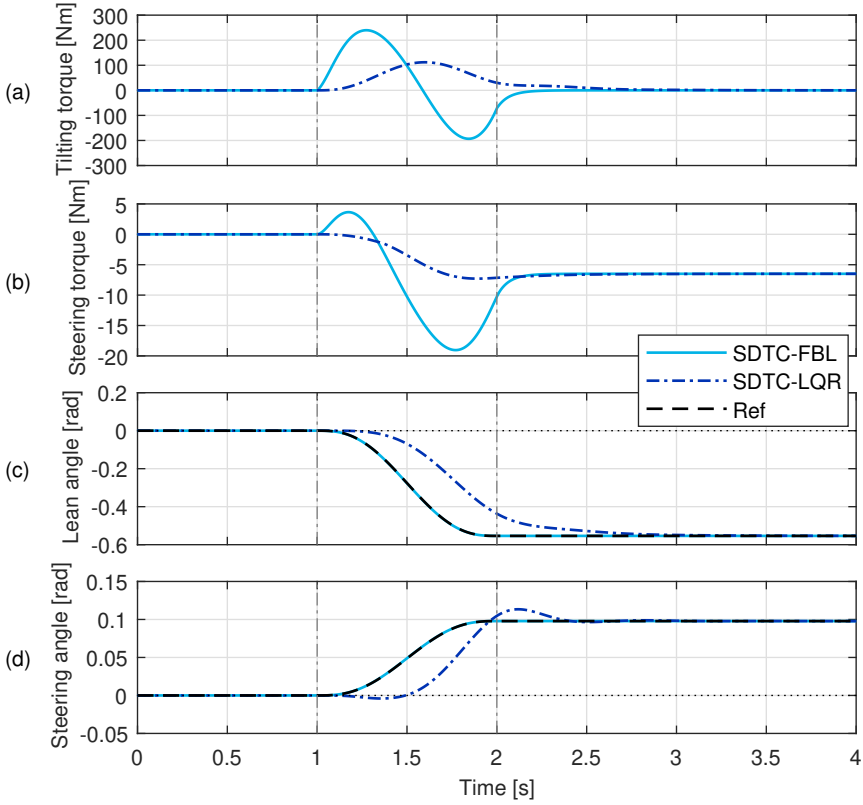


Figure 6.22: Curve entry performed with SDTC and a linear feedback law ($V = 10$ m/s and $R = 20$ m). SDTC via feedback linearization perfectly follows the lean and steer angle references but leads to an important tilting torque in a similar way as the DTC mode.

Steering delay Exact simultaneous control of the lean and steering angles results in a quite high tilting torque for the transient maneuver. In order to remedy this issue, a delay in the steering dynamics could be implemented in order to first lean the vehicle then steer it and not conversely. This can be done by considering a given dynamics between the steering angle reference $r_\delta(t)$ and the steering angle δ :

$$\frac{\delta}{r_\delta} = \frac{-as + \omega^2}{s^2 + 2\zeta\omega_0s + \omega^2} \quad (6.46)$$

The transfer function in equation (6.46) can be tuned in order to achieve a given non-minimum phase behavior and a given delay for the steer response. The following values are selected: $\omega = 20$, $\zeta = 0.5$ and $a = 50$. Delaying the steering response leads to interesting results in terms of torque actuation (Fig. 6.23). Both the tilting and the steering torque are reduced at the beginning of the maneuver (Fig. 6.23.a and b) as some forced countersteering is operated

(Fig. 6.23.d). Nonetheless, the steering angle does not catch up while the lean angle is tracked perfectly (Fig. 6.23.c), which leads to a larger peak for both the steering and tilting torque at the end of the maneuver.

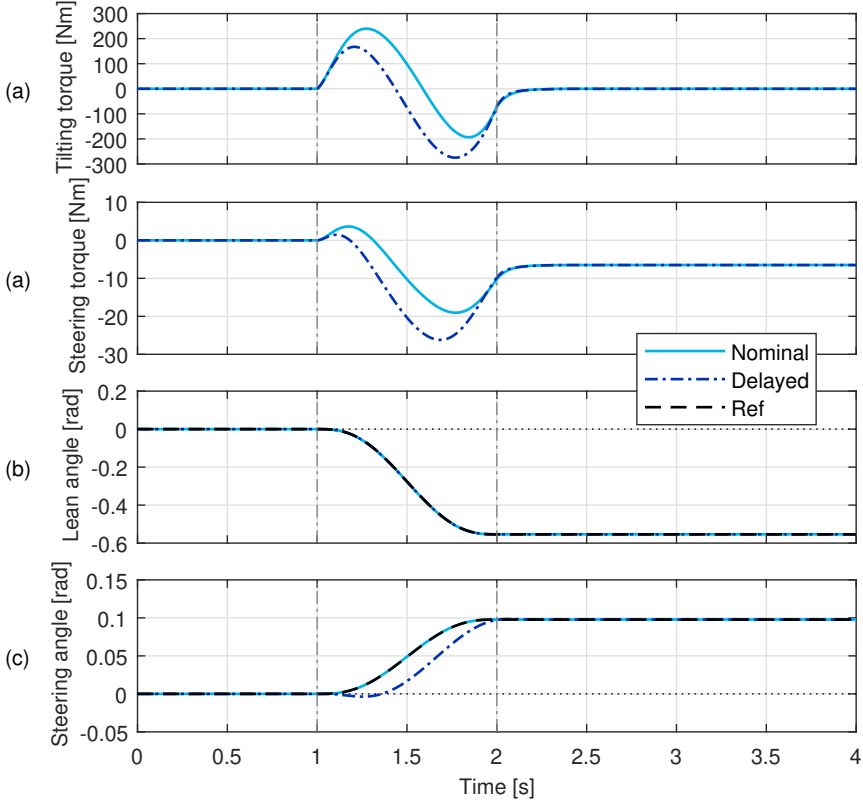


Figure 6.23: Curve entry performed with SDTC and a steering delay ($V = 10$ m/s and $R = 20$ m). Delay in the steering angle together with a non-minimum phase behavior leads to a reduction of the input torques at the beginning of the maneuver but causes a larger torque peak at the end.

Ideally, the lean angle should exhibit a sharp countersteering via the NMP behavior and then directly reach the reference steering angle during the maneuver. In addition, an oversteering at the end of the maneuver, as performed by the STC, should also lead to a lower tilting torque. To do so, one solution could be to link both the steering and rolling dynamics by having: $e_\delta = -K_{\delta\theta}e_\theta$. However, this is not trivial to achieve as only the second time derivative of the error, $\ddot{e}_\delta$ and $\ddot{e}_\theta$ can be tuned. In practice, this approach was not successful, maybe because the control of the lean angle was too nervous not allowing enough lean angle error for the countersteering to occur.

6.3 Model predictive control

This section focuses on model predictive control (MPC) and its application to NTV control.

Model predictive control (MPC) is a control technique based on the online solving of an optimization problem. At each time t , an OCP is solved for a future time horizon, $[t, t + T]$ and an optimal sequence of input $u(\tau|t)$ is found (e.g., the sequence that minimizes the error between the output and the reference on the prediction horizon). This implies the availability of a model in order to predict the system response to the input sequence. To do so, MPC relies on a dynamic model which predicts the system response to the input sequence on the prediction horizon. Once the optimization problem is solved, the first input of the sequence, $u(t|t)$ is applied at time t (Fig.6.24). This way, MPC achieves a feedback control that accounts for state deviation from a desired trajectory and that takes advantage of the available model and the reference preview⁹ in order to optimize the current input at time t .

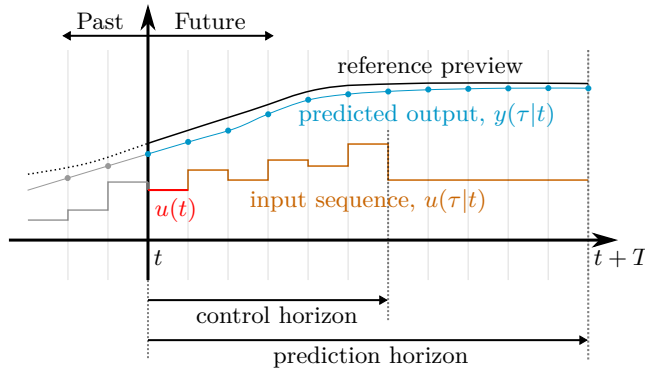


Figure 6.24: Model predictive control principle. At each time, t , the system input is determined by solving an optimal control problem for a future time horizon, $[t, t + T]$.

There exist several methods to realize model predictive control depending on the available model of the controlled system. The simplest one is to consider a linear time-invariant (LTI) model which with an appropriate cost function leads to a convex OCP that can easily be solved [77].

On the other hand, when the system dynamics, the constraints and the cost function are known to be nonlinear, the MPC might solve for the full nonlinear system resulting in a non-convex NLP problem that needs to be solved at each time step [77].

With regard to the application and the difficulties encountered with nonlinear OCP solving (Section 5.3), a trade-off was made by considering a linear time-

⁹Such preview information could be obtained via cameras or a combination of GPS tracking and road map database.

varying (LTV) model and a quadratic cost function for the OCP [93][94][95]. The latter has the advantage to account for the system nonlinearities while still offering a fast and robust solving of the OCP at each time step.

This section is organized as follows. First, the LTV-MPC framework is explained together with its practical implementation for a multibody system (MBS). Then, simulations are performed in order to compare the performances of MPC with respect to LQR and FBL for the serial control modes.

6.3.1 Linear time-varying model predictive control of multibody systems

First, let us consider the nonlinear dynamics representation of a MBS as defined in equations (6.47-6.48).

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad (6.47)$$

with

$$x = \begin{pmatrix} q_u \\ \dot{q}_u \end{pmatrix} \quad f(x) = \begin{pmatrix} \dot{q}_u \\ -M_r^{-1} F_{r,p} \end{pmatrix} \quad g(x) = \begin{pmatrix} 0 \\ M_r^{-1} N \end{pmatrix} \quad u = N^T Q_{u,a} \quad (6.48)$$

For a given time t , the prediction horizon $[t, t + T]$ can be discretized in a set of N discrete time intervals of size, T_s (Fig. 6.24). Then, a nominal trajectory on the prediction horizon can be defined by considering a vector of discrete inputs:

$$U^* = \left\{ u^*(t), u^*(t + T_s), \dots, u^*(t + (N - 1)T_s) \right\} \quad (6.49)$$

Typically the sequence U^* can be either the optimal input computed at the previous discrete control time, or a reference trajectory of the input known in advance for $[t, t + T]$. Based on this sequence of inputs, the nonlinear model in equation (6.47) can be time integrated from t to $t + T$ starting from $x^*(t)$ ¹⁰. This results in a vector of the nominal discrete states, X^* and a vector of discrete outputs, Y^* which, together with U^* , define a discrete nominal trajectory.

$$X^* = \left\{ x^*(t), x^*(t + T_s), \dots, x^*(t + (N - 1)T_s) \right\} \quad (6.50)$$

$$Y^* = \left\{ y^*(t), y^*(t + T_s), \dots, y^*(t + (N - 1)T_s) \right\} \quad (6.51)$$

For each discrete point k of the trajectory $(u_k^*(t), x_k^*(t))$, the system can be linearized. More specifically, the *discrete* tangent matrices (A_k^d, B_k^d, C_k^d) between two adjacent discrete states k and $k + 1$ are derived from the continuous tangent

¹⁰The starting state, $x^*(t)$ is generally the current state.

matrices (A_k, B_k, C_k) using the matrix exponential:

$$A_k^d(t) = e^{A_k T_s} \quad \text{with} \quad A_k(t) = \frac{\partial \dot{x}^*}{\partial x}(t + kT_s) \quad (6.52)$$

$$B_k^d(t) = (A_k^d - I) B_k A_k^{-1} \quad \text{with} \quad B_k(t) = \frac{\partial \dot{x}^*}{\partial u}(t + kT_s) \quad (6.53)$$

$$C_k^d(t) = C_k \quad \text{with} \quad C_k(t) = \frac{\partial y^*}{\partial x}(t + kT_s) \quad (6.54)$$

Eventually, the *continuous* nonlinear system in (6.47) can be approximated by a series of *discrete* linear systems along the discretized trajectory (U^*, X^*) .

$$\begin{aligned} x(t + (k+1)T_s) - x^*(t + (k+1)T_s) &= A_k^d(t)(x(t + kT_s) - x^*(t + kT_s)) \\ &\quad + B_k^d(t)(u(t + kT_s) - u^*(t + kT_s)) \end{aligned} \quad (6.55)$$

$$y(t + kT_s) - y^*(t + kT_s) = C_k^d(t)(x(t + kT_s) - x^*(t + kT_s)) \quad (6.56)$$

By defining the discrete variables x_{k+1} , x_k , u_k and y_k as the deviations from the discrete trajectory $(u_k^*(t), x_k^*(t))$, the linear time-varying (LTV) model in equation (6.61) is obtained.

$$x_{k+1} \triangleq x(t + (k+1)T_s) - x^*(t + (k+1)T_s) \quad (6.57)$$

$$x_k \triangleq x(t + kT_s) - x^*(t + kT_s) \quad (6.58)$$

$$u_k \triangleq u(t + kT_s) - u^*(t + kT_s) \quad (6.59)$$

$$y_k \triangleq y(t + kT_s) - y^*(t + kT_s) \quad (6.60)$$

$$\begin{cases} x_{k+1} &= A_k^d(t)x_k + B_k^d(t)u_k \\ y_k &= C_k^d(t)x_k \end{cases} \quad (6.61)$$

Based on system (6.61), an optimization problem can be defined for the prediction horizon $[t, t+T]$. Considering quadratic cost function and linear constraints on both the states and the inputs, the resulting OCP in equation (6.63), defined for tracking, is still a convex quadratic problem (QP) [96]. More details on the derivation of this tracking optimization problem can be found in [97][96]. Note that the difference with an LTI system is that for the LTV model, the QP matrices $H(t)$, $F(t)$, $G(t)$ and $S(t)$ depend on time and need to be updated at each time step, t .

$$\min_z \quad \frac{1}{2} z^t H(t) z + v^t F(t) z \quad (6.62)$$

$$\text{subject to} \quad G(t)z \leq W(t) + S(t)v^t \quad (6.63)$$

with z , the sequence of input discrete increments on the future time horizon and v , the state, x and reference, r at the initial time t .

$$z = \begin{pmatrix} \Delta u_0 \\ \Delta u_1 \\ \vdots \\ \Delta u_{N-1} \end{pmatrix} \quad v = \begin{pmatrix} x(t) \\ r(t) \\ u(t-1) \end{pmatrix} \quad (6.64)$$

Such framework allows for an easy definition of constraints and weights for the control inputs, their rates and the error with respect to the reference signal.

Practical implementation

Despite the many developments realized on the solving of OCP of MBS with CasADi (Section 5.1.4), the LTV-MPC for MBS is implemented with the Simulink toolbox, *Model Predictive Control Toolbox* developed by Prof. Bemporad [96]. The discrete time integration, the successive linearizations and the matrix discretization are implemented within the Robotran framework (Fig. 6.25).

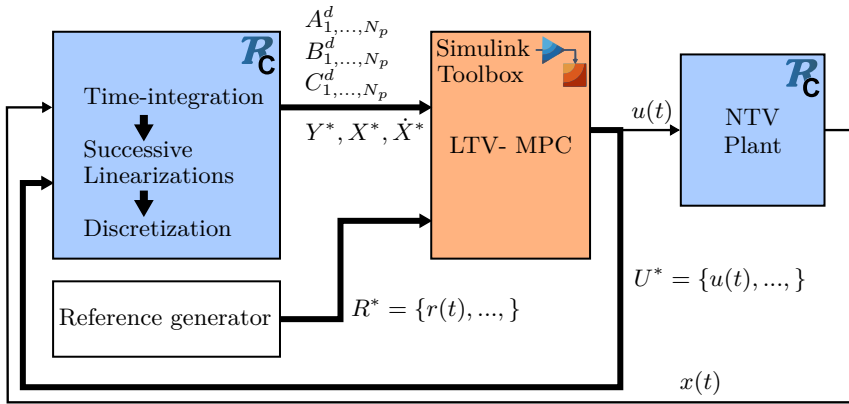


Figure 6.25: LTV-MPC implementation. An important part of the computation, i.e., the time integration, the successive linearizations and discretization, is realized within the Robotran environment. The vehicle plant is also simulated within the Robotran framework.

6.3.2 Model predictive control of narrow track vehicles

The LTV-MPC introduced in the previous section can be applied to the control of NTV. First, the MPC parametrization is presented. Then different time simulations are realized in order to assess the MPC performances and verify its applicability to NTV control.

MPC parametrization

The MPC high versatility (i.e., preview capability, multivariable, constraints handling) leads to a high number of degrees of freedom (weights, horizons, constraints), which makes MPC design particularly complex.

The first parameters to set are the preview parameters: T_s, T, N_p and N_c .

In practice, the controller sample time, T_s cannot be too large as it would lead to poor disturbance rejection, or too small which could cause excessive computation load. Similarly, the prediction horizon, T cannot be too long otherwise the system unstable mode would dominate. On the other side, if the prediction horizon is too short, constraint violations would be unforeseen.

Generally, it is recommended to have a prediction horizon, T that covers for the system open-loop response. Given the overdamped open-loop response of the studied NTV, the prediction horizon should be around 2 s (Section 4.3). However, the experience has shown that a prediction interval of $T = 0.5$ s was more suitable for the tracking of the baseline 1-second curve entry maneuver¹¹. Regarding the sample time, T_s , it was set to 0.01 s to ensure the numerical stability of the successive time integrations¹². This results in 50 prediction intervals, $N_p = 50$. Usually, only the first numbers of control inputs have a significant effect on the predicted output behavior. In order to reduce computational cost, the number of control intervals, N_c can be limited to about 10% of the total number of prediction intervals, leading to $N_c = 5$.

The output considered for MPC is the NTV lean angle, $y = \phi$ while the two inputs are the tilting torque, Q_θ and steering torque, Q_δ . The lean angle was chosen as an output in order to compare the performances to the FBL and the LQR. Nonetheless, the yaw rate can also be tracked by the MPC controller since it can also handle NMP systems. The remaining parameters are the quadratic weights on the outputs, the control inputs and the input rates.

The weights are adjusted and define the different serial controllers, e.g., an important weight on the tilting torque will result in a pure STC control (Table 6.1). For the 3 modes, weights are applied on the input rates in order to avoid high peaks in either the tilting or the steering torque. Nonetheless, these weights have not been optimized¹³.

	W_{Q_θ}	W_{Q_δ}	$W_{\Delta Q_\theta}$	$W_{\Delta Q_\delta}$	W_y
STC	10^3	0	10^{-2}	10^{-1}	10^1
LTC	0	10^3	10^{-2}	10^{-1}	10^1
SDTC	10^{-2}	10^{-2}	10^{-2}	10^{-1}	10^1

Table 6.1: MPC weights for the different serial modes as defined by us for the MPC Matlab toolbox [96]. Single input modes (STC and LTC) are obtained by choosing a large weight on the other input (e.g., STC has a large weight on the tilting torque).

The use and the availability of preview information imply a fully autonomous serial NTV control for which, based on the tracking of the path ahead (e.g.,

¹¹Larger prediction horizon led to an excessive anticipation from the MPC which results in a poor tracking of the lean angle. In addition, 0.5 s is sufficient to capture and avoid rollover events as shown in Fig. 4.23 (Section 4.3.2).

¹²A simple explicit Euler method between each discrete state is used in this work.

¹³These weights are not exactly the same as the LQR weights. The first reason is that LQR and MPC are not implemented in the same manner as for LQR it is the entire state that it is tracked while for the MPC, there is only one output (i.e., the lean angle or the yaw rate).

captured with cameras), a reference signal (lean angle or yaw rate) can be computed and tracked by the MPC. Therefore, the serial control modes (STC, LTC and SDTC) are studied but not the DTC as the preview capability of MPC does not make sense for this mode. The DTC is by nature a cooperative control mode for which the controller acts in parallel on the tilt d.o.f., while the driver steers the vehicle via the steering assembly (Section 1.2). Even if some preview on the path ahead would be available, there is, however, no way to predict the future driver steering input on the prediction horizon for the DTC.

Importance of the preview capacity

To illustrate the influence of the preview capacity on the control performances, the usual curve entry ($V=10$ m/s and $R=20$ m) is simulated for the STC and LTC. The same baseline curve entry is performed as in the previous sections. The use of the reference preview leads to the best lean angle tracking in both cases (Fig. 6.26 and 6.27).

For both modes, the MPC anticipates the curve entry about half a second before the maneuver by applying a countertorque (Fig. 6.26.a and 6.27.a). This torque causes the vehicle to slightly lean in the opposite direction (Fig. 6.26.b and 6.27.b) which enables the generation of a higher momentum toward the curve at $t = 1$ s when the maneuver starts.

For the same set of weights, the MPC with preview results in a very accurate tracking similar to what the FBL offers while MPC with no preview leads to a slower response but still slightly more reactive response than the LQR controller (Fig. 6.19 and 6.20). Note that without any preview capability, higher weight on the output could be used to compensate for the lack of preview information. With larger output weights, the tracking will be better and get closer to what FBL is offering. Except for the slight counter torque and the difference in starting time, the torque profiles are very similar with and without preview. Regarding the peak torques values, the order of magnitude observed for MPC is similar to that of the FBL control.

Combined steering and tilting torque (SDTC)

Model predictive control can be used to combine both the steering and tilting torques (SDTC). In that case, both the lean angle tracking and the yaw rate tracking are considered. In practice, the SDTC behavior depends on the weights used for the 2 inputs and their rates (Table 6.1). For SDTC, it is necessary to consider equal weights on the tilting and the steering torques in order to ensure the sole use of steering torque in steady state cornering (i.e., after $t=2$ s).

In these simulations, the preview capability of MPC is considered and the SDTC performances are compared with the STC and LTC ones.

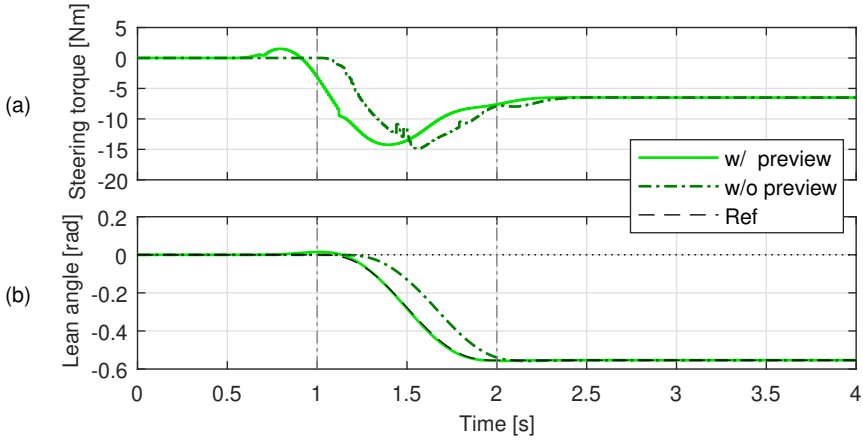


Figure 6.26: Curve entry performed with MPC ($V=10$ m/s and $R=20$ m). The reference preview for MPC-STC leads to a better tracking of the lean angle as the MPC anticipates the maneuver.

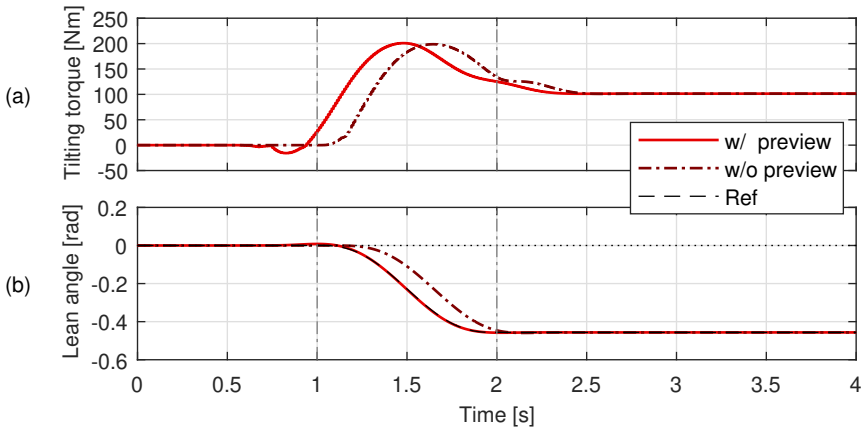


Figure 6.27: Curve entry performed with MPC ($V=10$ m/s and $R=20$ m). The reference preview for MPC-LTC leads to a better tracking of the lean angle as the MPC anticipates the maneuver.

Lean angle tracking The baseline curve entry is first simulated considering the lean angle tracking (Fig. 6.28). The first observation is that the SDTC makes very little use of the tilting torque except for a small peak around $t=1.1$ s with the same order of magnitude as the steering torque (Fig. 6.28.a). The SDTC mainly relies on steering torque to realize the maneuver at the lowest cost (Fig. 6.28.b). This is in accordance with the equal weights for the steering and tilting torques (Table 6.1). The difference between the reference lean angle signals (Fig. 6.28.c) is simply explained by the different natures of the

modes (FTR vs. FTC) as presented in Chapter 4 (Section 4.1). The SDTC weights on the inputs could be adjusted such that more tilting torque is used in the transient phase but this would not improve the tracking which is already accurate enough.

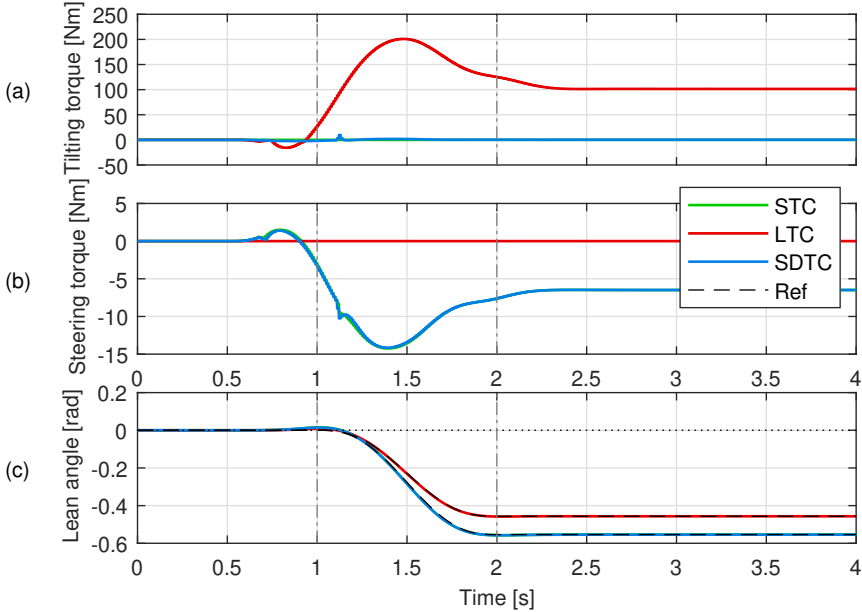


Figure 6.28: Curve entry with lean angle tracking ($V=10$ m/s and $R=20$ m). SDTC mainly uses the steering torque while LTC and STC both manage to accurately track the lean angle trajectory.

Yaw rate tracking MPC can also be implemented to track the vehicle yaw rate, $\dot{\psi}$ contrarily to the feedback linearization approach which is limited to lean angle tracking (Section 6.2). The same set of weights as for the lean angle tracking is used and the 3 serial modes are again compared for the baseline curve entry (Fig. 6.29).

The yaw rate tracking leads to a very different behavior than the lean angle tracking. First of all, it is important to notice that LTC tracks the yaw rate better than the STC (Fig. 6.29.c). This is in accordance with the unavoidable countersteering operated by the STC which implies a vehicle yaw motion in the outward direction at the beginning of the maneuver.

The SDTC offers a trade-off between the LTC and the STC by applying a nonzero tilting torque during the transient phase of the maneuver (Fig. 6.29.b). This goes with a slight diminution of the steering torque compared to the STC case (Fig. 6.29.c). The weights on the steering and tilting torque are the same, which means that the MPC optimal framework deliberately decides to use the tilting torque instead of the steering torque in order to ensure an accurate

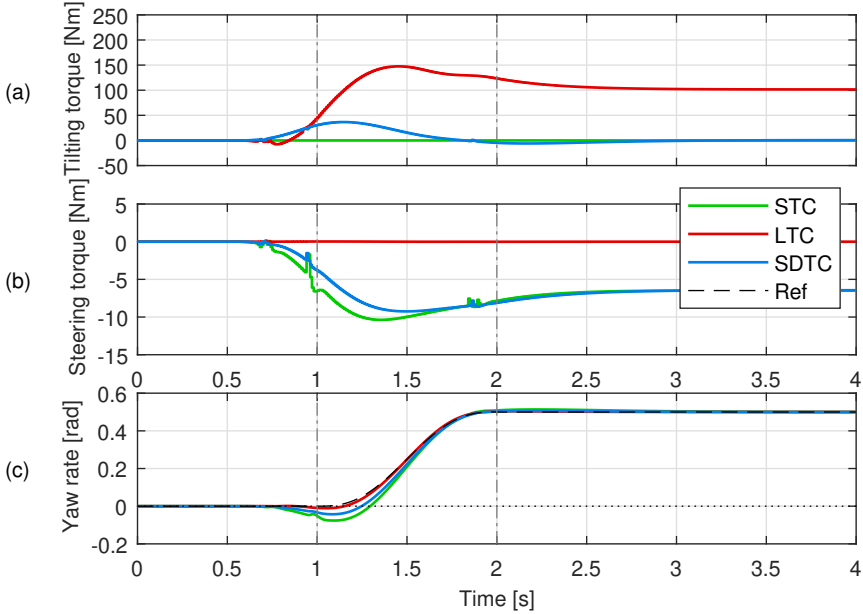


Figure 6.29: Curve entry with yaw rate tracking ($V=10$ m/s and $R=20$ m). Yaw rate tracking is performed more accurately via a tilting action than a steering one.

tracking of the vehicle yaw rate. Note that the same behavior regarding the yaw rate tracking is observed when no preview is considered.

Eventually, it is important to highlight that, for both STC and LTC, yaw rate tracking leads to a lower transient peak torque than lean angle tracking while realizing the maneuver in the same amount of time (Fig. 6.28 and Fig. 6.29).

Low speed maneuver

When serial control modes (STC, LTC and SDTC) are considered, the main limitation of the FBL and LQR techniques were their stability at very low speed. With LQR, the serial modes exhibited excessive oscillations at a speed of 3 m/s, especially the STC (Fig. 6.12). For the FBL, the lean angle control showed lateral instability for speed lower than 2.15 m/s due to the internal dynamics (Fig. 6.18).

To verify the stability with MPC, a curve entry is realized at a speed of 2 m/s and for a 5-meter radius. The 3 modes lead to an accurate tracking of the lean angle (Fig. 6.30.a). Unfortunately, the LTC results in a greater amount of yaw rate in steady state than expected $\dot{\psi} > 0.4$ rad/s. (Fig. 6.30.b).

In order to alleviate this issue, a yaw rate tracking has been tested for LTC but has led to a lean angle instability. Furthermore, a combined tracking of the yaw

rate and lean angle was envisaged for LTC but again it has failed to properly track the low speed curve entry.

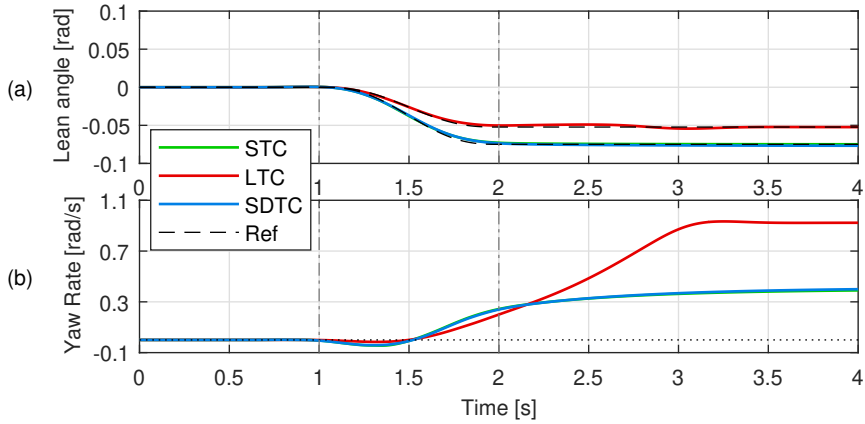


Figure 6.30: Low speed Curve entry ($V=2$ m/s and $R=5$ m). At very low speed, MPC stabilizes the STC but the LTC remains laterally unstable.

Constrained model predictive control

One of the numerous advantages of the MPC is the fact that constraints can be considered while solving the optimization problem on the future horizon. In the present framework (i.e., LTV-MPC, Fig. 6.25), one can easily constrain the inputs and or the outputs of the system. To this extent, the output vector size can always be increased if the constraints are to be applied to other states of the system (e.g., the steering angle, the lean rate). In practice, actuators are often limited in both power and maximum torque. To demonstrate the use of constraints with MPC, the LTC mode is considered with an additional constraint on the maximum tilting torque expressed in Nm.

$$-150 \leq Q_{tilt} \leq 150 \quad (6.65)$$

The input constraint is properly taken into account by the MPC controller as the tilting torque does not exceed 150 Nm during the transient behavior (Fig. 6.31.a). In addition, the constrained LTC uses the preview information to foresee the constraint limitation and increases the torque slightly before the unconstrained LTC (around $t=1.15$ s). Such constraint affects the performances of the LTC mode, especially at the end of the maneuver as the tilt constraint slows down the lean angle tracking (Fig. 6.31.b). Limiting the tilting torque also leads to the limitation of the wheel load transfer which does not exceed 50% for the constrained LTC (Fig. 6.31.c). Finally, the LTC manages to reach the steady-state condition but with a delay compared to the unconstrained LTC (Fig. 6.31.b).

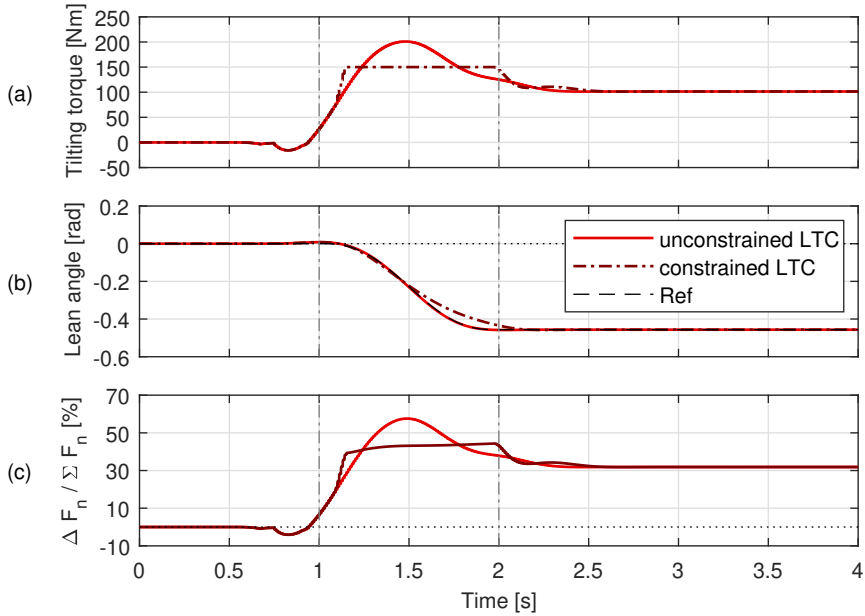


Figure 6.31: Curve entry with constrained MPC ($V=10$ m/s and $R=20$ m). The MPC strictly respects the tilting torque limitation and, moreover, is able to anticipate the limitation.

6.4 Concluding remarks

In this chapter, different nonlinear control techniques have been investigated for 4 control modes: DTC, STC, LTC and SDTC. First, a gain-scheduling LQR based on the vehicle speed was developed. Then, the applicability and performances of feedback linearization were studied for the different control modes. The NTV control via MPC was investigated for the serial modes.

Besides the strong assumptions (i.e., no driver model, no perturbations), many interesting conclusions can be drawn on the general feedback control of NTVs.

Regarding the gain-scheduling linear quadratic regulator (LQR).

- The LQR analysis highlights the responsiveness of DTC with respect to the other serial modes. However, it also shows that DTC is limited in terms of transient maneuver as it comes with an important wheel load transfer, which can cause the inner wheel to lift from the ground. In addition, the perceived acceleration at the driver's head is drastically larger for the DTC compared to the other serial modes.
- Regarding the serial modes, the combination of the steering and tilting actions does lead to better performances both in terms of tracking and energy consumption. The SDTC as implemented in this work offers a trade-off between the serial STC and LTC mode. It reduces drastically the

tilting torque input compared to the DTC and the LTC mode. Moreover, the steering input for SDTC is very close to the steady state steering torque value, meaning that the steering feedforward term dominates in the transient phase for SDTC.

- A gain-scheduling LQR allows for the NTV stabilization on the whole speed range. However, STC was observed to be less stable at low speed. Nonetheless, proper tuning of the LQR gains as function of the speed V could lead to better tracking performances at low speed.

Regarding the feedback linearization (FBL)

- FBL generally leads to a faster response than the LQR, which could be explained by two main reasons. Firstly, the LQR aims at stabilizing all the states and reach the reference, meaning that there might be conflicting signals during the transient phase, while the FBL controls one signal only and does not intent to stabilize the system's internal dynamics. Secondly, the FBL takes full advantage of the available nonlinear model which allows for a faster control of the vehicle lean angle.
- For single input serial modes (STC and LTC), the tracking of the lean angle via FBL leads to a faster tracking for the envisaged maneuver. To do so, the counter and oversteering in STC are both larger than with the LQR approach. For the LTC, the tilting torque is higher than for LQR but the lean angle is better tracked and the yaw rate response more reactive. Nonetheless, both modes are unstable below a certain speed which is lower than the self-stability speed (i.e., 2.25 m/s instead of 5 m/s).
- FBL also leads to improved tracking performances for DTC. In addition, the faster the DTC control is, the lower the tilting torque will be. This is in accordance with the DTC nature for which the vehicle is first steered by the driver and then tilted accordingly *in parallel*. It is important to notice that such a *bang-bang* tilting action as performed by FBL cannot be achieved with a simple proportional control or by just considering larger gain in the LQR control. As a matter of fact, the FBL takes advantage of the available nonlinear model to track the lean angle as fast as possible, which consequently reduces the dynamic delay between the steering and tilting dynamics.
- The combined use of the steering and tilting torques (SDTC) is also possible via FBL by considering two outputs to track: the lean angle and the steering angle. In reality, the SDTC via FBL corresponds more to the DTC mode for which the steering dynamics was prescribed and the tilting torque exhibits two peaks in the transient phase.

Regarding the model predictive control

- First, the advantage of preview capability is assessed. Both the LTC and STC lead to better tracking of the lean angle if some preview of the lean angle reference signal is available. It has also been observed that the performance of MPC without preview could be improved by considering higher weight on the output signal.

- Using its preview capability, MPC leads to a very accurate tracking of the lean angle reference signal with performances comparable to the ones of the feedback linearization (FBL). However, the anticipated input (slight countertorque) does not lead to a reduction in the input torque compared to FBL.
- For a given maneuver, MPC enables the tracking of either the lean angle or the yaw rate. Generally, LTC leads to a better tracking of the yaw rate compared to the STC and SDTC. Moreover, SDTC involves a significant use of tilting torque for yaw rate tracking but not for lean angle tracking. This is explained by the strong NMP behavior of the yaw rate which can only be counteracted using the tilting torque at the beginning of the maneuver.
- Regarding the very low speeds, the STC mode is stabilized with MPC contrarily to the LTC which properly tracks the lean angle but results in an inappropriate yaw behavior (i.e., the vehicle turning too fast in steady state cornering).

6.4.1 Comparison of the three control approaches

The gain-scheduling LQR stabilizes the NTV on the overall speed range but leads to poor tracking performances as it intends to control the whole vehicle state simultaneously. In addition, the fact that it is based on linear control methods for the upright position, its applicability is limited when higher lateral acceleration maneuvers are envisaged. On the other hand, FBL leads to very proficient tracking of the lean angle. However, its SDTC implementation is possible but limited to lean angle and steering angle tracking. In addition, FBL for single input serial modes is unstable below a speed which is lower than the self-stability one (i.e., 2.25 m/s instead of 5 m/s). Model predictive control shows very good capacity for the combination of the steering and tilting action. It does not limit the NTV control to lean angle tracking but also enables yaw-rate tracking conversely to FBL. Regarding the low-speed stability, MPC is able to stabilize the STC but not the LTC. The main issue with MPC is the applicability of its preview capability. As anticipative information on the future trajectory is necessary to generate a reference signal to track, MPC with preview would always be limited to the autonomous driving of NTVs instead of their cooperative control.

Besides its contribution to the field of NTV control, this chapter also presents a systematic way to implement 3 high-level control techniques within a relative coordinate multibody framework. It is obvious that the developed controllers could easily be used for other mechanical systems as far as those can be modeled as MBS.

Conclusions and perspectives

7

7.1 Content overview

This thesis focuses on the dynamic analysis and nonlinear control of tilting narrow track vehicles (NTVs). Tilting the vehicle while cornering is necessary and can be achieved in two ways: via the steering assembly with a vehicle free-to-roll (FTR) or via the tilting suspension with a steering assembly free-to-caster (FTC). Based on these two tilting modes, different tilting control strategies can be envisaged (e.g., STC, DTC). Generally, the NTV tridimensional dynamics is highly nonlinear which motivates the use of dedicated modeling formalism and advanced nonlinear techniques (e.g., FBL, MPC) for the study of NTV control.

Chapter 1 presents the scientific challenges associated to tilting narrow track vehicles and the transportation context of which they are part. The different tasks (i.e., heading and balancing) and types of control (i.e., driver, cooperative or autonomous) are briefly introduced. Then, the two passive tilting modes (FTR and FTC) are presented together with the different control modes derived from it.

In Chapter 2, a thorough state of the art in the field of NTV control was introduced. This chapter was presented as a condensed summary of the previous NTV control investigations structured around several entry points, such as the vehicle modeling, the NTV dynamic analysis or the envisaged control techniques. The state of the art demonstrates the interest value of investigating the NTV control using nonlinear methods and advanced multibody modeling that are able to properly account for the NTV tridimensional dynamics and non-minimum phase behavior. In addition, very little has been done on the NTV control via the free-to-caster (FTC) approach. Thus, there is also a clear interest in studying transient maneuvers and steady state cornering for this particular FTC mode and compare it to the FTR mode.

In Chapter 3, the model of a reference NTV obtained via the MBS formalism is introduced. The NTV morphology and the MBS model are presented and specific multibody treatments were considered for the front assembly (suspension and steering assembly) and for the tire/ground contact. Chapter 3 clearly indicates

that the modeling of NTVs is not straightforward and requires the use of dedicated tools to properly account for the tridimensional dynamics of NTVs.

Chapter 4 focused on the dynamic analysis of NTVs and the comparison of FTC and FTR via different analyses performed on the MBS equations of motion. Based on the reference NTV model developed in Chapter 3, the dynamic comparison shows that the steady state cornering behavior of the vehicle is strongly influenced by the actuation mode (input torques, lean angle, perceived acceleration, stability, wheel load transfer).

Moreover, the transient behavior for both modes is analyzed. Generally, the steering action (FTR) leads to a higher delay in yaw rate than the tilting action (FTC) as the observed countersteering is smaller for the latter. In addition, It is shown that the FTC can be performed by monotonically driving the tilt d.o.f. up to its steady state value, which is not possible for the FTR that needs the steering assembly to be first countersteered during a curve entry.

In Chapter 5, a framework for the optimal control of multibody system was developed and applied to the control of NTVs. While the SDTC mode is envisaged and the actuation torques are minimized, the optimal control systematically leads to the use of the steering torque rather than the tilting torque. The NTV fall is initiated via a countersteering action while the steady-state configuration is also reached via a combined use of tilting and steering torques. The optimal control framework has also enabled the study of constrained maneuvers (i.e., direct steering constraint). Although nonlinear optimal control has shown itself insightful for the control of NTVs, the computational effort associated to the advanced nonlinear program problem limits the use of OCP to offline investigations.

In Chapter 6, different nonlinear control techniques were investigated for 4 control modes: DTC, STC, LTC and SDTC. Besides the strong assumptions (i.e., no driver mode, no perturbations), many interesting conclusions can be drawn on the general feedback control of NTVs.

The gain-scheduling LQR stabilizes the NTV on the overall speed range but leads to poor tracking performances as it intends to control the whole vehicle state simultaneously. In addition, STC was observed to be less stable at low speed. Nonetheless, proper tuning of the LQR gains as function of the speed V could lead to better tracking performances at low speed.

Feedback linearization leads to a very good tracking of the lean angle as it takes full advantage of the available nonlinear model which allows for a faster control of the vehicle lean angle. However, FBL for single input serial modes is unstable at a speed which is lower than the original self-stability speed (i.e., 2.25 m/s instead of 5 m/s). FBL also leads to improved tracking performances for DTC and to a lower tilting torque in transient maneuvers than for the LQR. The FBL takes advantage of the available nonlinear model to track the lean angle as fast as possible, which consequently reduces the dynamic delay between the steering and tilting dynamics. The SDTC implementation is possible via FBL but it is limited to the exact tracking of the lean angle and steering angle.

Eventually, model predictive control enables to combine the steering and tilting actions. Using its preview capability, MPC leads to a very accurate tracking of the lean angle reference signal with performances comparable to the ones of the feedback linearization (FBL). It does not limit the NTV control to lean angle tracking but also enables yaw-rate tracking contrarily to FBL. For such tracking, SDTC involves a significant use of tilting torque conversely to lean angle tracking which mainly relies on steering torque. Regarding the low speeds, MPC is able to stabilize the STC but not the LTC. The main issue with MPC is the applicability of its preview capability.

7.2 Objectives achievement

Two objectives were defined in the introduction of this manuscript (Section 1.3). Based on the previous content overview, the achievement of the two objectives is discussed.

Comparison of the two passive tilting modes The first objective was to study the FTC mode and compare it to the more traditional FTR mode in a systematic way. In Chapter 4, we managed to analyze both modes and their difference via a thorough comparison based on a set of advanced analyses (steady state cornering equilibrium, modal analyses, open-loop response). From this comparison, it is shown that FTC is associated to a higher actuation cost than FTR in both steady-state condition and transient maneuvers. Moreover, the additional inner tilting torque for FTC leads to a higher wheel load transfer and a more upright position than the FTR. Nevertheless, FTC presents some advantages compared to the FTR. First, it results in a more reactive yaw rate response than the FTR as the countersteering is less important. More importantly, FTC can be performed by monotonically driving the tilt d.o.f. up to its steady state value, which is not possible with the FTR for which countersteering is mandatory. This result is very insightful on a mechanical design point of view as it indicates that the driver could possibly control the vehicle via a *direct steering* logic without any countersteering action needed. The mode comparison is summarized in Table 7.1.

	Free-to-roll (FTR)	Free-to-caster (FTC)
Action torque	moderate (0-15 Nm)	high (50-300 Nm)
Lean angle	~aligned with the gravity and lat. acc. resultant	more upright
Wheel load transfer	moderate (<10%)	large (up to 80%)
Cornering stability	less stable	more stable
Perceived acceleration	slight centripetal feeling	centrifugal feeling
Lateral response	slower due to significant countersteering	faster as the auto countersteering is small
Driving logic	countersteering needed	direct steering

Table 7.1: Comparison of the passive FTR and FTC and their dynamics.

Nonlinear control of NTV via nonlinear techniques The second objective of this thesis was to investigate the use of nonlinear control techniques based on advanced modeling of the NTVs dynamics. In Chapter 6, we showed that feedback linearization (FBL) and model predictive control (MPC) could be implemented, leading to an important increase in tracking performances while compared to a more conventional linear technique, i.e, linear quadratic regulator (LQR).

It was shown that FBL was well suited for single input control modes (i.e., STC, LTC and DTC) as it leads to a very accurate tracking of the lean angle and a reduction of the input torques during the transient maneuver while compared to LQR. However, instability at low speeds (below 2.25 m/s for the studied vehicle) was still observed for STC and LTC and generally FBL does not enable the control of the vehicle yaw rate as a non-minimum phase behavior was observed for such output.

Regarding the implementation of the multiple input control mode (i.e., SDTC), MPC showed a very good capacity to combine the steering and tilting actions. Using its preview capability, MPC leads to a very accurate tracking of the lean angle reference signal with performances comparable to the ones of the feedback linearization (FBL). Moreover, MPC enables an accurate yaw rate tracking which is mainly achieved through a significant use of the tilting torque in accordance with the more reactive yaw rate response observed for a tilting action (FTC mode). Regarding the low speeds, MPC is able to stabilize the STC but not the LTC. However, the main issue with MPC remains the applicability of its preview capability. As previous information on the future trajectory is necessary to generate a reference signal to track, MPC with preview would always be limited to the autonomous driving of NTV instead of their cooperative control. Nevertheless, MPC would still be the best choice for the design of a SDTC controller especially if constraints need to be taken into account, as illustrated for a bounded tilting torque. Table 7.2 summarizes the advantages and disadvantages of the three nonlinear control approaches.

	LQR	FBL	LTV-MPC
lean angle tracking, ϕ	OK	OK	OK
yaw rate tracking, $\dot{\psi}$	OK	KO ^a	OK
DTC	slow	fast	KO
STC	slow	fast	OK
LTC	slow	fast	OK
SDTC	acceptable	KO	most suitable
constraints handling	limited	KO	most suitable
autonomous driving	KO	KO	possible

Table 7.2: Comparison of the three control methods. MPC is the most appropriate method for the constrained active control of NTVs.

^aThis result might strongly depend on the NTV morphology.

7.3 Original contributions

Nonlinear control of multibody systems Besides its contribution to the field of NTV control, this thesis also presents the implementation of three high-level control techniques within a relative coordinate multibody framework. It is obvious that the controllers developed could also be used to control other mechanical systems as far as those can be modeled as multibody system (e.g., industrial robots, cars). The numerical code developed in the Robotran framework for the nonlinear control of MBS is ready-to-use and can be easily adapted to other applications.

Framework for the systematic comparison of the passive tilting modes

In this thesis, the comparison of the two passive tilting modes (FTR and FTC) has been realized in a systematic way using a set of advanced analyses (i.e., steady state cornering maps, modal analysis, curve entry). The systematic framework developed could easily be reused for another NTV vehicle with a different morphology, enabling an easy and fast assessment of the FTR and FTC dynamics.

Thorough understanding of the concept of perceived acceleration

The perceived acceleration for steady state cornering conditions was studied and it was observed that its value was far from zero on the overall steady state cornering map. This is caused by the non-zero wheel load transfer due to the suspension geometry of the tilting three-wheeler. This result is very interesting as several works in the past based their NTV controller on the perceived acceleration regulation [53, 29], which is limited to a simplified vehicle model but not suited or even incorrect for a real NTV equipped with suspensions.

Thorough understanding of the free-to-caster mode In this thesis, we analyzed in detail the free-to-caster (FTR) dynamics and brought to light many aspects of its unconventional dynamics (i.e., more upright position, large wheel load transfer, auto countersteering, direct steering capability). In this way, we believe that we fill a void in the previous NTV literature.

7.4 Thesis perspectives

Besides the thesis objectives, there are still many scientific investigations that could be carried out based on the present work. Here, we list some of the most interesting ones from our point of view.

- **Sensitivity analysis of FTR and FTC dynamics.** It could be of interest to analyze the influence of dynamic parameters such as the trail or the caster angle on the NTV dynamics for both FTR and FTC. This could give precious insights for the design of a passive FTC-based NTV.
- **Comparison of the NTV and passenger car dynamics.** Generally, NTVs aim at offering a driving experience similar to what a driver can encounter with a conventional passenger car. Thus, it would be interesting to realize a systematic comparison of a NTV (FTR and FTC) and a

passenger car and observe to what extent their dynamic behavior differs (i.e., steady-state input torque, perceived acceleration, lateral stability, wheel load transfer).

- **Validation of the dynamic analysis and controller performances.** This thesis was mainly based on a theoretical model which was supposed to perfectly represent the vehicle dynamics. In practice, validating experimentally the model and the nonlinear controllers could reinforce the observations made in this thesis.
- **Performance of the nonlinear techniques with respect to model discrepancies.** For the three nonlinear feedback controllers that were developed, the model was supposed to be perfectly known and the states available. It would be insightful to study which of the three nonlinear techniques handle the best the model discrepancies. This could be performed numerically, for instance by degrading the controller internal model.
- **Implementation of a NTV driver model.** Developing a driver model for NTVs would enable the realization of advanced maneuvers via simulations (e.g., double lane change). However, this is not an easy task as the model should ideally account for some knowledge of the vehicle via an internal model [87][88][89].

Generally, the main continuation of the present work would be to translate the theoretical observations and developments into more tangible findings: this necessitates a NTV prototype to carry out experimental tests to validate or invalidate the obtained results.

7.5 Broader considerations

Aside from the active cooperative¹ control of narrow track vehicle, this thesis has allowed us to develop a critical eye regarding the future of tilting NTVs and their control. In this section we give directions on how developments and research investigations on NTVs should be carried out in the future.

It has been now 30 years since that the investigation of NTV *active* control started with research projects mainly focusing on the cooperative control via SDTC. None of the past research projects have led to the development of a viable commercial NTV equipped with an active tilt control. We believe that actively controlled NTVs cannot take a share of the mobility market because their active control represents too much innovation for the customers at the moment. In addition, an NTV, by its nature, already represents a novelty that people still need to apprehend. In our perspective, there are two distinct ways NTVs could make their way in the transportation market.

¹The driver and the vehicle interact and work together to achieve both control the heading and balancing tasks. Generally, the driver performs the high level path-following tasks whereas the vehicle tilts itself through a steering and/or tilting action.

7.5.1 Development of passive narrow track vehicles

A first solution could be to focus on the development of passive NTVs intended for low speeds (i.e., less than 40 km/h) enabling their use on bicycle paths. For these vehicles, we believe that a direct steering logic should be envisaged capitalizing on the already-existing driving skills that people use when driving a conventional car. To achieve such direct steering without active control, a free-to-caster logic could be considered (Fig. 7.1). Nonetheless, the large tilting torque needed in both transient maneuver and steady state cornering should not be devolved to the driver. This could be achieved by implementing a power-assisted steering² and ensure that the tilting motion is not backdrivable (e.g., mechanically and/or via a set of hydraulic valves and actuators). Another challenge is the low speed steering instability which requires a locking of the tilting d.o.f. and a reconnection of the steering wheel to the steering assembly. The latter represents a huge mechanical design challenge as the transition from FTC to locked tilt should be seamless for the driver. Moreover, the handling of critical transient maneuvers (e.g., car avoidance) remains an issue that might require an automatic limitation of the tilting to avoid outer wheel lifting and vehicle rollover. Furthermore, the case of non-horizontal ground and its irregularities also represents a hurdle that will require advanced design considerations (i.e., implementation of a g-force sensor). We want to stress that the development of such passive NTVs for moderate speeds is not only a problem of bolting and assembling but also requires an advanced knowledge of the NTV dynamics in order to make the most of the FTC mode. We believe that the systematic dynamic analysis of the FTC mode realized in this thesis could be a very helpful for the development of such vehicle.

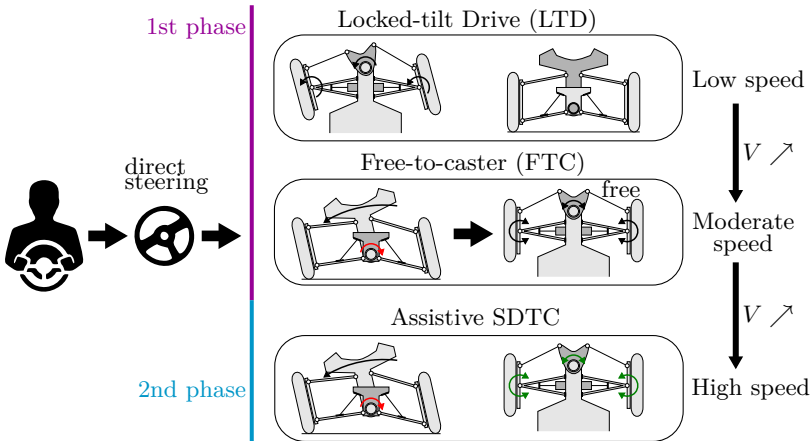


Figure 7.1: A two-step approach for the development of NTVs.

Once passive NTVs are accepted and commonly used for short drives, they could be considered for higher speeds (up to 90 km/h) and longer travels

²as installed between the steering wheel and steering assembly on a passenger car

via the implementation of an assistive control that would intervene only for critical maneuvers for which wheel lifting, assembly breaking or skidding could occur (Fig. 7.1). The assistive action could take different forms: an extra countersteering action, a torque vectoring³ or a limitation of the tilting torque. Such control could be achieved via MPC as it can treat constrained MIMO systems well.

Such a two-step approach has already been studied by Vandenbrink who is the responsible for the development of the Carver vehicle [98]. To date, we believe that the new version of the Carver, i.e., the Carver Earth, is the only NTV that could achieve a commercial success [99].

7.5.2 Autonomous control of narrow track vehicles

A second approach would be to capitalize on the future advent of autonomous vehicles. In that case, fully active autonomous driving would already be accepted and implementing it on lighter agile unipersonal vehicles would not represent a significant change for the mobility market (Fig. 7.2). For such autonomous NTV control, the questions of the active control mode and the nonlinear method would again be relevant. In that sense, we do believe that our thesis has already given a part of the answer, as claimed previously in Section 7.2.

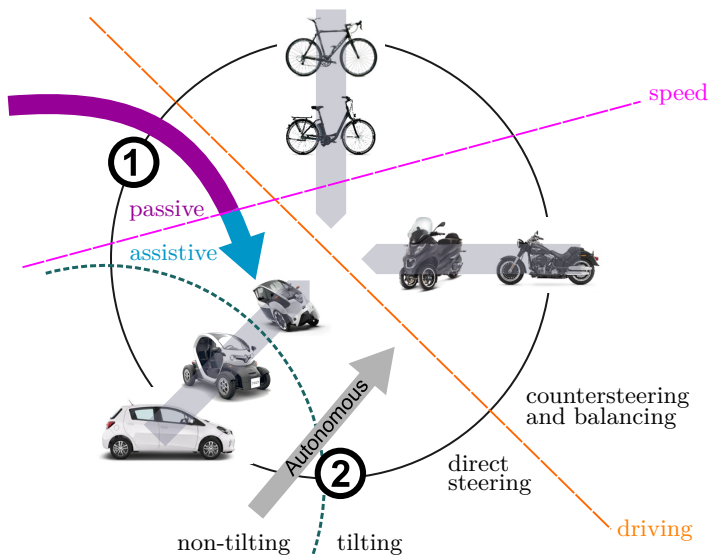
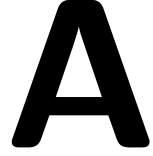


Figure 7.2: Two different approaches could be envisaged for the implementation of NTVs in the mobility market.

³Torque vectoring consists in controlling the torques applied to each driven wheel in order to influence the vehicle lateral behavior. A common example is the Electronic Stability Program (ESP) that uses braking torque on one wheel to improve the vehicle lateral stability in critical maneuvers.

Appendices

Literature review



This appendix describes the different projects that have involved works on the control of NTVs. The goal here is to give a general overview of the past projects on NTV control in the broad sense. The approach used to structure the different projects is chronological. The appendix is divided into two main sections which mainly differ by the time period they treat: the first part covers the projects that are finished while the second part focuses on ongoing projects.

In this literature review, different quantities are defined and used by the authors. The roll and lean angle, denoted ϕ , φ or θ , are the same physical quantities and correspond to the angle that the chassis median plane makes with the vertical. The tilt angle, often denoted θ as well, is the angle between the tilting chassis and the vehicle base. For simplified models in which the vehicle base is always aligned with the horizontal plane, the roll (i.e., lean) angle is the same as the tilt angle.

A.1 Historical review

A.1.1 UCDavis (1992-1998)

The research on narrow track vehicle control was initiated by Prof. Karnopp and his team from UCDavis in the early 90s. They have been the first ones to define and study both the direct tilt control (DTC) and the steering tilt control (STC) [18][17]. To do so, they use a single-track model¹ neglecting tire deformation and slip angles coupled with an inverted pendulum model for the roll dynamics of the vehicle (Fig. A.1). Karnopp's model expresses the vehicle yaw rate², $\dot{\psi}$ and lateral local velocity³, U , as a function of the longitudinal velocity V and the front axle steering angle δ . Finally, this leads to a simple equation for the roll dynamics which can be further linearized resulting in equation (A.2) where

¹A bicycle model is a simplified representation of a ground vehicle with the front and rear axles replaced by two equivalent single wheels [14].

²The vehicle angular velocity component along the vertical direction.

³The vehicle velocity component along the transverse direction of the chassis.

m and I respectively denote the mass and the yaw inertia of the vehicle.

$$(I + mh^2)\ddot{\phi} - mgh\phi = -mh(\dot{V} + \dot{\psi}U) \quad (\text{A.1})$$

$$= -mh\left(\frac{\dot{U}}{l}b\delta + \frac{U}{l}b\dot{\delta} + \frac{U^2}{l}\delta\right) \quad (\text{A.2})$$

The front steering angle δ is considered as an input which can either be associated to the driver or actively controlled (e.g., for STC). The second input is the tilting torque M_t directly applied on the tilting d.o.f. and that allows for direct tilt control (DTC). Generally, the NTV longitudinal dynamics is neglected and the vehicle longitudinal speed V is considered as a parameter rather than a state.

At an early stage, the desired tilt angle ϕ_d is expressed through the balance of centrifugal and gravitational effects at a constant speed steady state cornering. Based on a small angle approximation for the tilt angle, ϕ and the front wheel steering angle, δ , equation (A.3) is obtained for the desired tilt angle.

$$\phi_d = \tan^{-1}\left(\frac{a_{lat}}{g}\right) \approx \frac{a_{lat}}{g} = \frac{V^2}{R} \approx \frac{V^2\delta_f}{l} \quad (\text{A.3})$$

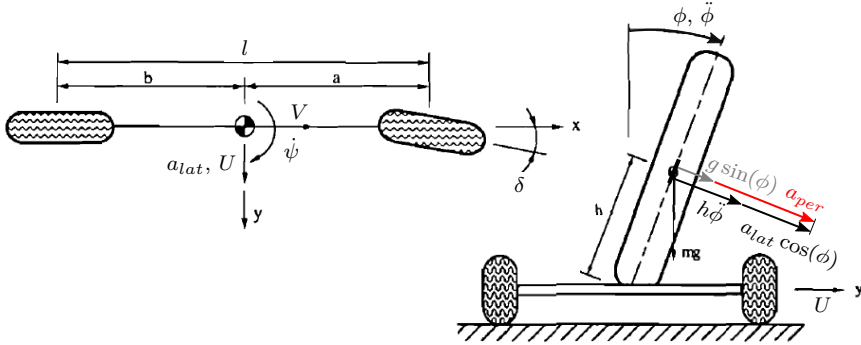


Figure A.1: Karnopp's model based on a single-track model augmented with a inverted pendulum (adapted from [11]).

In order to stabilize the vehicle roll, a simple steering control law is first proposed with the front wheel steer angle acting proportionally to the tilt angle error.

$$\delta_f = -K_d(\phi_d - \phi) \quad (\text{A.4})$$

With this simple proportional control law, Karnopp and Fang are able to demonstrate that a countersteering action is required at the beginning of a maneuver in order to stabilize the vehicle [17]. Later on, Karnopp and Hibbard have taken interest in the DTC mode for stabilizing the vehicle roll motion. By considering an input torque on the tilt d.o.f. , they implement a PD controller that tracks the aforementioned desired tilt angle. Based on this controller, they

compare the behavior of tilting and non-tilting NTVs by using the notion of perceived acceleration a_{per} defined in equation (A.5) (Fig. A.2).

$$a_{per} = a_{lat} \cos(\phi) + h\ddot{\phi} - g \sin(\phi) \quad (\text{A.5})$$

Using a more elaborated model including suspension and tire dynamics [49], they show that tilting the vehicle through DTC enables the complete cancellation of the *transverse* perceived acceleration⁴ and a 30% reduction of the wheel load transfer compared to a non-tilting vehicle in steady state cornering.

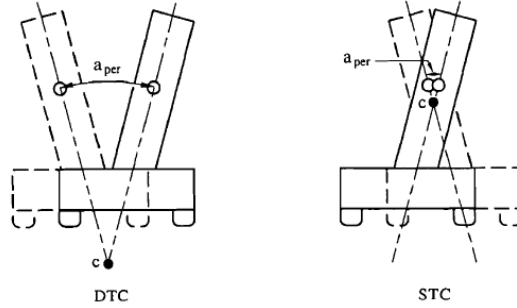


Figure A.2: Perceived acceleration and the instantaneous roll axis c for the two control modes [11].

So's work Following the previous work, So and Karnopp have started to investigate the combination of both DTC and STC on the overall speed range [11]. First, they compare both modes implemented with PD controllers, and show that, during transient maneuvers, a STC system leads to a lower perceived acceleration than that of a DTC equipped vehicle (Fig. A.2). For the two control modes to be compared, a steer-by-wire logic is considered for the STC mode: the driver, through the steering wheel angle δ_{sw} , indicates how much he wants the vehicle to be steered and the front steering angle δ_f is then controlled accordingly. In addition, they infer on the STC's instability at very low speeds based on common sense but no simulations are performed to verify their claim.

Then, they propose to combine the two modes to overcome their respective limitation. However, they were unable to successfully combine both STC and DTC in a continuous manner due to a conflict between the two control methodologies: the countersteering operated in STC leads to an erroneous computation of the desired tilt angle for the DTC [11]. Thus, rather than blending the two modes, they consider switching from one to another depending on a given advance speed of 8 m/s (Fig. A.3). Their main objective is to make the switching seamless to the driver. However, simulations show that the different natures of the controller cause a sudden jump in lateral acceleration

⁴The transverse perceived acceleration is the component of the total *felt* acceleration in the chassis transverse direction. The total felt acceleration is defined as the sum of the lateral acceleration and the roll-induced acceleration compared to the contribution of the gravity vector.

at the moment of switching from the DTC to the STC: this happens because under STC there is a derivative action in the dynamic system and the vehicle is not in a steady state configuration when the switch occurs. To solve this issue, a smoothly varying control gain for the STC system is introduced, resulting in smooth responses at the time of switching from the DTC to the STC. This switching control algorithm is extended to emergency maneuvers for which the STC is switched back to the DTC in the case of loss of traction [52].

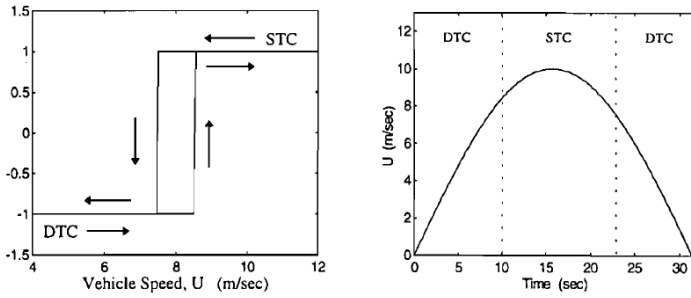


Figure A.3: Switching SDTC approach with a hysteresis-type strategy[11].

Snell's work Based on the previous works, Snell and Karnopp propose another control approach which consists in minimizing or eliminating the perceived acceleration at an arbitrarily selected point C on the tilting chassis median plane[53]. This method is equivalent to forcing the vehicle to roll about an axis passing through the same point C (Fig. A.4). First, Snell analyses the behavior of a DTC system regulating the perceived acceleration, rather than the tilt angle error with a test maneuver consisting of a 0.5 g turn. Whilst perceived acceleration is reduced, this refined DTC system is found to demand large tilt moments which could cause rollover.

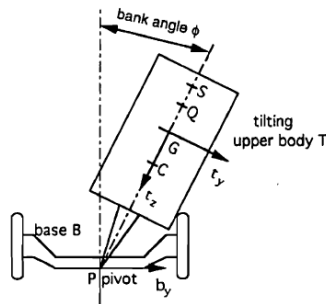


Figure A.4: Illustration of different points on the tilting chassis of a NTV [53].

Considering a combined SDTC approach, Snells overcomes the two modes disagreement by computing the desired lateral acceleration a_d based on the driver's steer demand δ_{sw} , rather than the front wheel steer angle δ_f . This allows both modes to work simultaneously in order to achieve a common goal. In this work, the yaw dynamics is omitted and the control system is designed based on the roll dynamics for which the STC input is now the lateral acceleration at the vehicle base, a_y^P , rather than the front steering angle δ_f (Fig. A.5). The proposed combined strategy use pre-filters on both the steering and tilting input which allow the tilt action to catch up with the instantaneous steering input (Fig. A.5).

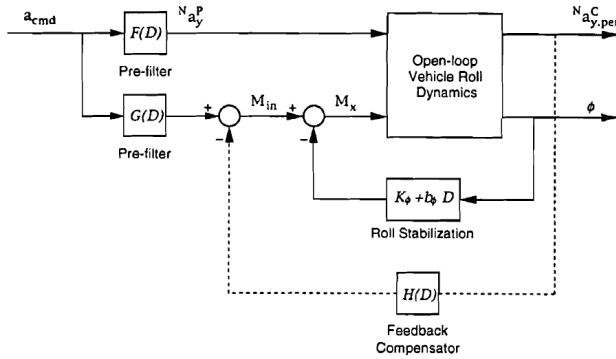


Figure A.5: Block-diagram of Snell's controller [53].

Choosing C coinciding with G , the chassis center of mass, the acceleration responses are quite similar to those of the basic STC mode; the vehicle base is moved outside of the curve through counter steering and the perceived acceleration is kept minimal. On the other hand, with C at the pivot point P , the SDTC behaves like the original DTC and the tilting torque exceeds its maximum overturning value. Snell has suggested locating the virtual axis where the driver senses motion (i.e., at the head level) which results in a SDTC control very close to the STC mode. However, the difference between Snell's control law and a classical STC is that the tilting torque is continuously active and can be used to stabilize the roll dynamics in case of extreme maneuvers at any time. To the author knowledge, no prototype has been developed during Karnopp's research.

A.1.2 TU Delft (1997-2004)

In the late 90s, the special tilting control technology DVC, that stands for dynamic vehicle control, is the subject of an academic research carried out by Prof. Pauwelussen from TU Delft. First, he has developed a linearized mathematical model (Fig. A.6) that he used to perform dynamic simulations of the Carver vehicle and assess its cornering performances [100].

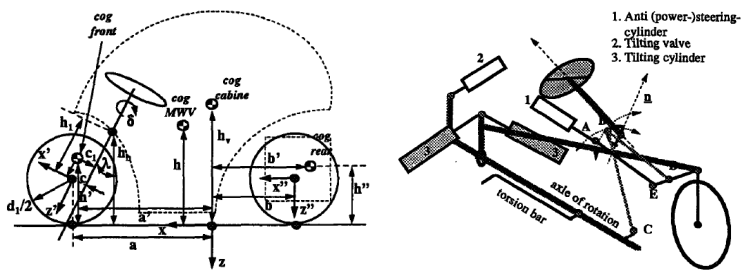


Figure A.6: The model of the Carver vehicle developed by Pauwelussen [100].

Based on that model, sensitivity analyses are performed on dynamic parameters such as tire stiffness or center of mass position to determine their influence on the tilting torque T_ϕ and the steering angle δ (Fig. A.7).

	T_ϕ	δ
cog moved forward	-	+
Slip stiffness front	0	-
Slip stiffness rear	0	+
Camber stiffness	0	-
cog moved upward	+	0
toe-in $\rightarrow$ toe-out	0	-
One extra passenger	+	+

Figure A.7: Sensitivity analysis of the Carver’s dynamic parameters. [100].

Later on, the stability of the Carver is studied with the help of modal analysis [101]. The root locus shows two stable modes for speeds between 10 m/s and 50 m/s: a yaw mode characterized by a fishtail motion and a steering mode identified by a high frequency steering oscillation. Sensitivity analyses at 120 km/h are performed on the tilt angle inclination, the center of gravity (c.o.g.) position showing a less stable behavior when the tilting axis is pitched forward and the c.o.g. moved backward. In addition, the antisteer control (automatic countersteer) is implemented to reduce the tilting torque for transient maneuvers. Finally, Pauwelussen argues that since the NTV is neither a passenger car nor a motorcycle, the results cannot be compared to existing test results and that new criteria should be established, including the vehicle feedback to the driver. As this research were based on an existing vehicle with a given morphology and its own tilting control technology, little can be retrieved and concluded for the tilting control of NTV from a general point of view.

A.1.3 University of Minnesota (2003 - 2010)

During the 2000s, investigations on narrow track vehicles are conducted at the University of Minnesota under the supervision of Prof. Rajamani. In 2003, Rajamani has started studying the NTV dynamic behavior by developing a new model for these special vehicles [19]. The model is a single-track model

composed of two bodies and comprises a total of 4 d.o.f. (Fig. A.8). This 4-d.o.f. is also referred as a 3-d.o.f. as the front steering angle is often considered as an input of the system.

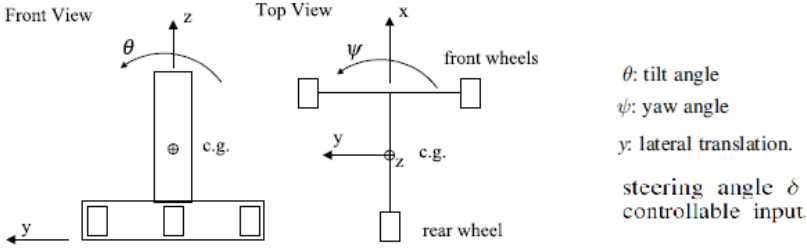


Figure A.8: The 4-d.o.f. model developed by Rajamani [19].

The longitudinal dynamics is ignored but significant improvements are made compared to the previous NTV models by taking into account the front wheel dynamics due to the trail (Fig. A.9), the effects of slip angles and the rotating wheel gyroscopic effects. In order to derive the model, two main approximations are made: (1) there is no load transfer between the left/right wheels, (2) front wheels steering do not contribute to their own lean motion. These approximations lead to the same approximated formula for the desired tilt angle as that of equation (A.3). Considering the trail equation (A.6) developed based on the steering axis balance (Fig. A.9), he shows that a tilt motion to the left side will cause a torque on the steering axis that will automatically steer the vehicle in the direction of the initial tilt motion. When the wheel is steered, the lateral force generated will always create an opposite moment around the steering axis due to the trail (mechanical and pneumatic), which will automatically realign the wheel. For more details on trail dynamics and its influence on roll stability, one can read the works of Schwab and Sharp on bicycle and motorcycle stability [43][44].

$$M_{trail} = mg \frac{l_r}{l_f + l_r} \gamma \cos(\beta) \sin(\theta) - F_f \cos(\theta) (\gamma + \gamma_{pneumatic} \cos(\beta)) \quad (A.6)$$

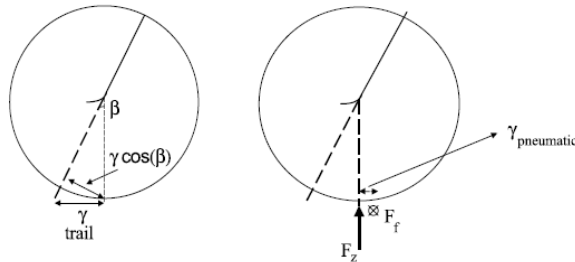


Figure A.9: Front wheel forces and effect of the trail [19].

Beside interesting considerations on self-stability, no self-stabilizing speed is determined for the modeled NTV. Based on his enhanced model, Rajamani has tried to identify the role of gyroscopic effect on the vehicle stability [19].

Gohl's work In 2004, Gohl under the supervision of Rajamani, started his work on NTV control through STC [50]. Based on Rajamani's improved 4-d.o.f. model, he considers steady-state equilibrium, takes interest in the different steady-state quantities and illustrates them with a yaw rate map function of lean angle and velocity (Fig. A.10).

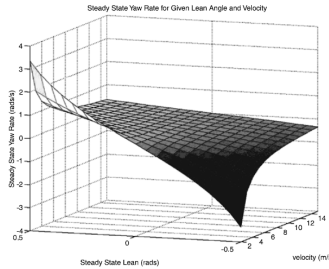


Figure A.10: Steady state cornering map [50].

Then, the nonlinear 4-d.o.f. model is linearized around the vehicle upright position. A driver model is used to produce a desired tilt angle signal based on a pre-defined path curvature and a given advance speed. Afterwards, a control law is derived with a proportional and derivative action (similar to [17]) in order to stabilize the vehicle and to track the desired tilt angle θ_d .

$$\delta_f = K_p(\theta_d - \theta) - K_d\dot{\theta} + \delta_{ss}(\theta_d) \quad (\text{A.7})$$

The proposed controller in equation (A.7) differs from Karnopp's work by the introduction of a steady-state feedforward term $\delta_{ss}(\theta_d)$ that is calculated offline as an approximation to the necessary steer angle corrfor the desired steady state cornering. This additional term leads to a reduction of the steady-state error. In cornering, the feedback part is first dominant then, once the tilt angle error starts to diminish, the driver feedforward term becomes dominant and provides the steady state steering input required to complete the turn.

The controller stability is tested with the model and the linear representation of the 4-d.o.f. model is compared to the nonlinear model. It is shown that for lean angle less than 25 degrees, the two representations lead to different steady-state configurations. Above 30 degrees, the differences lead to an instability in the controller. In addition, experiments are performed using an unmanned prototype vehicle in order to verify the controller performances observed in simulations. For moderate inputs, the vehicle is stabilized by the STC system and follows the lean angle demanded by the user.



Figure A.11: The unmanned NTV prototype from Minnesota [20].

Piyabongkarn's work In 2004, Piyabongkarn's work aimed at providing a systematic SDTC approach that synchronizes the tilt and cornering maneuvers in order to reduce actuator requirements and improve ride qualities [21]. He argues that the tilting and steering need to be synchronized because any lag in the tilting dynamics necessitates large peaks in the required actuator torque. Based on a simplified version of Rajamani's 4-d.o.f. model, two different approaches are explored by Piyabongkarn for NTV control: a receding horizon controller (RHC) [61] for desired tilt angle prediction and a nonlinear state feedback linearization [62].

For the purpose of simulating the vehicle, the driver is modeled as a state feedback controller on the lateral and yaw error states, while a DTC system is used for the control of the tilt error states with θ_d computed as in equation (A.3). A LQR framework is chosen for designing both the driver model and the tilt controller.

Assuming that short-term preview information about the road curvature ahead of the vehicle is available (e.g., provided by a lane-detection vision system or a combination of GPS tracking and a road map database), the RHC approach offers a systematic way of using this look-ahead information to improve the SDTC synchronization (Fig. A.12).

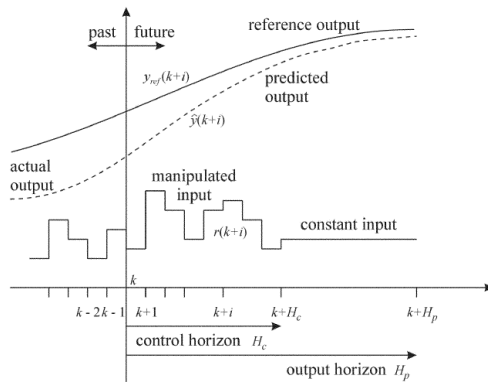


Figure A.12: Receding horizon control strategy as envisaged by Piyabongkarn [21].

The main principle of the RHC is to modify the desired tilt angle value, in order

to indirectly restrict the torque commands of the DTC controller, retain vehicle stability and yet track the *original* desired tilt angle (deduced from the actual road curvature). The RHC implies solving an optimization problem on the receding horizon for the actual desired tilt angle, while satisfying constraints on the tilting torque. Simulation results show the proper torque saturation provided by the solution of the constrained optimization problem. Nevertheless, model mismatch between the linearized and the nonlinear models is so significant that the linear RHC would lead to the use of extremely large torque values, several orders of magnitude larger than with the linear model assumption. This explains why the simulations are limited to large radii of 500 meters.

The second approach is based on nonlinear feedback linearization which consists in applying a nonlinear input f_{nl} leading to in an input-output linear system (the reference tilt angle θ_d being the input). Taking advantage of the model simplicity, the roll equation is readily usable to derive a nonlinear input law for the tilting torque M_t such that the error dynamics is linear:

$$M_t = f_{nl}(\ddot{\theta}, \dot{\theta}, \theta_d, K_D, K_P) \quad \ddot{e}_\theta + K_D \dot{e}_\theta + K_P e_\theta = 0 \quad (\text{A.8})$$

Where $e_\theta = \theta - \theta_d$ and K_D , K_P are two parameters that can be tuned to define the error dynamics in (A.8). The nonlinear command f_{nl} is composed of several terms whose influence on the error dynamics has been investigated by Piyabongkarn. The performances of the nonlinear control are assessed compared to a LQR and a PD controller. The nonlinear controller keeps the tilt angle error e_θ close to zero while the amount of torque needed for the control is strongly reduced compared to the two other linear controllers. Eventually, the RHC controller is used on top of the nonlinear DTC controller allowing for input constraints and preview utilization. Only low lateral acceleration simulations are performed [21].

Kidane's work In 2005, Kidane started to investigate and compare DTC, STC and their combined use [102]. Based on a simplified 4-d.o.f. model, he proposes an improved DTC controller with a revised formula compared to equation (A.3) for the desired tilt angle. The latter is the solution to the dynamic equation (A.9) while keeping the angular acceleration set to zero, $\ddot{\theta} = 0$ [102].

$$\theta_{d,DTC} = \frac{(\ddot{y} + \dot{\psi}V) - 2(I_{w,3} - I_{w,2})\omega\dot{\psi}/(hm)}{g} \quad (\text{A.9})$$

Equation (A.9) states that, in order for the vehicle to be stable, the latter needs to lean in such a way that the resultant of the net horizontal acceleration and the net vertical acceleration makes an angle θ_d with the vertical (Fig.A.1). On a practical point of view, this improved expression of the desired tilt angle can be realized by using a system of accelerometers to read the pure lateral acceleration of the vehicle without forgetting to filter out the angular acceleration component of the vehicle $\ddot{\theta}$. Regarding the DTC controller itself, a simple PD regulator is used. The presence of the lateral translation acceleration term, $\ddot{y}$, significantly

reduces the transient torque requirement of the tilt actuator compared to the usual steady-state formula from equation (A.3). Simulations show that the peak torque at the beginning of the transient phase is reduced and the steady-state torque is canceled during steady state cornering.

Considering STC again, Kidane proposes a steer-by-wire logic for which the driver input is appropriately scaled to determine the desired angle θ_d as shown in equation (A.10). To account for steady-state steering, the driver input is fedforward to the controller in a similar manner than in equation (A.7).

$$\theta_{d,STC} = K\delta_{sw} \quad (\text{A.10})$$

Regarding the SDTC system, the main challenge is the transient phase for which the reference tilt angles θ_d given by equations (A.9) and (A.10) differ for the two modes. It is first stated that it is not a problem as either one or the other controller should be dominant depending on the advance speed. The propose SDTC is thus a switching strategy with exponential factors that multiply the control inputs depending on the speed (Fig. A.13).

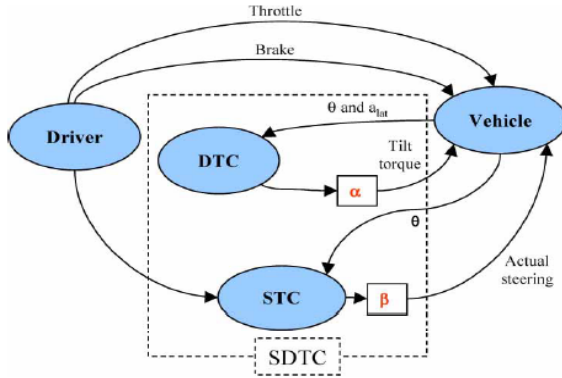


Figure A.13: Kidane strategy for SDTC control [12].

In another paper, Kidane revises his control approach by considering θ_d as a function of the driver input for both the STC and DTC mode, removing any conflicting signals between the two control modes [12]. In addition, he considers an integral term in his STC controller in order to produce the steady-state front steering angle. This integral action would ensure that in case of vehicle parameters changes (e.g., additional payload mass is added), the tilt angle error still goes to zero at steady state resulting in the right steady-state steering input without requiring the knowledge of any vehicle parameters.

Simulations with a varying speed are performed and it is observed that the peak control effort of the STC and DTC are out of phase by 180 degrees: when the DTC effort is at its highest (lowest speed), the STC effort is at its lowest and vice versa. This shows that the combined controller is capable of proportioning the STC and DTC system appropriately depending on the velocity of the vehicle.

However, the differences of performances between the SDTC and the original STC and DTC are neither observed nor discussed by Kidane.

In 2006, Kidane estimates the road bank angle⁵ based on accelerometer measurements [74] by assuming linear approximation for the vertical and lateral acceleration [74]. Afterwards, the road bank angle is easily taken into account into the expression of the desired tilt angle.

In addition, experimental validations are considered on a man-size prototype vehicle with a tadpole design and a front double wishbone suspension (Fig. A.14).



Figure A.14: Research NTV prototype developed by Kidane at the University of Minnesota [54].

For this purpose, the combined SDTC system is designed in such a way that the STC and DTC systems run in parallel for all speeds of the vehicle (the scaling factors α and β from Fig. A.13 are not used anymore). First, Kidane and Rajamani acknowledge that the omission of speed, V , from the desired tilt angle calculation leads to an excessive understeer tendency. They consider a revised θ_d calculation which includes the advance speed.

$$\theta_d = K\delta_{sw}V^2 \quad (\text{A.11})$$

Low speed experiments are performed with the two different methods for the θ_d computations defined by equations (A.10) and (A.11). Comparison of the results shows that the speed dependent formula (A.11) successfully gives neutral steering behavior during driving tests. Although the prototype is capable of large tilting, up to 35 degrees, the experimental results are limited to 14 degrees tilt angle and lateral acceleration of around 2.5 m/s^2 . The performances of the control system in vigorous, transient maneuvers are not assessed.

Last but not least, Kidane also discusses the case of LTC which he calls free-to-steer DTC [22]. Based on the prototype and the control system implemented, test maneuvers are performed with different desired tilt angles. It is observed that above a tilt angle of 10 degrees, the controller is not able to properly track the desired tilt angle and oscillations appear. Based on a simplified model, Kidane explains the controller failure by a difference between the tilt angle of the front wheels and the tilt angle of the vehicle chassis resulting in a higher

⁵The road bank angle is the angle between the road surface and the horizontal plane.

lateral force and causing the tilt motor to saturate. The wheel load transfer influence on the vehicle steady state cornering is not considered in the discussion as it is omitted in their simplified model.

A.1.4 Clever project, Bath University (2005 - 2014)

NTVs have also been the subject of research in Europe during the late 2000s and early 2010s. Developed in the frame of an EU-funded project, The Clever vehicle (Compact Low Emission VEHICLE for uRban transport) was a tilting three-wheeled NTV developed at the University of Bath under the supervision of Prof. Edge and Dr. Darling. It has been built by Drew and Barker during their respective PhD theses [103][23]. The prototype is composed of a non-tilting rear pod and a tilting cabin with a front steering wheel. A hydraulic system allows for direct tilt control while the front steering wheel is either passive or active thanks to an actuator that modulates the steering angle in real time.

Based on this prototype, the University of Bath has thoroughly investigated both the NTV dynamics and its control with a total of 4 PhD thesis on the subject from 2004 to 2014.

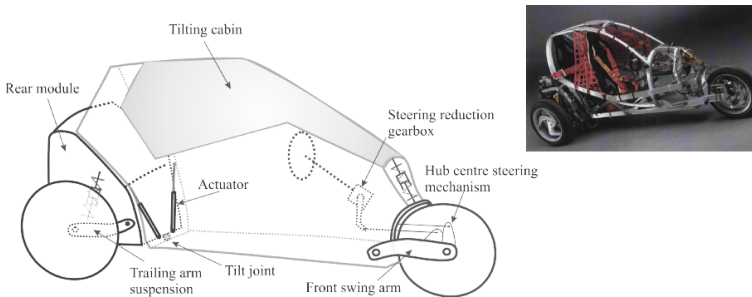


Figure A.15: The Clever prototype vehicle developed at the University of Bath [23].

Drew's and Barker's work Based on a simplified inverted pendulum model for the roll motion, Drew has studied the DTC system and its response to a step steer input of 5 degrees [23]. With a simple PD controller, simulations are performed for which large perceived accelerations are observed. Based on these observations, Drew suggests that the DTC system could lead to a discomfort but also diminish the vehicle roll stability as the tilting torque causes an important wheel load transfer. Later on, Barker has introduced a more complete simulation model of the CLEVER vehicle which featured yaw, roll, and tire dynamics together with a hydraulic model for the DTC actuation [24]. Based on that, Barker investigated the mechanical design of the vehicle chassis and parameters such as anti-roll bar stiffness, suspension damping coefficient and their influence on the vehicle dynamics and stability. In addition, he studies the tilt joint axis

inclination, ξ and the inherent rear wheel steering that together influence the vehicle under/over steering characteristics (Fig. A.16). Inclining the tilt joint axis by 5 degrees changes the vehicle behavior from over to neutral steering. The tilt axis inclination leads to a difference between the vehicle roll angle with respect to the ground and the cabin tilt angle with respect to the rear pod around the inclined axis. Besides, Barker introduces the notion of over-lean function in the desired tilt angle formula (A.12), so that the vehicle is balanced with respect to the effective roll angle.

$$\theta_d = K_{ol} \frac{\delta R V^2}{g L} \quad (\text{A.12})$$

In steady state testing, an overlean gain K_{ol} of 1.2 is used to provide the driver with a natural and balanced feel during cornering; accelerometer measurements have shown that a driver experienced a small perceived acceleration directed toward the inside of the curve [24].

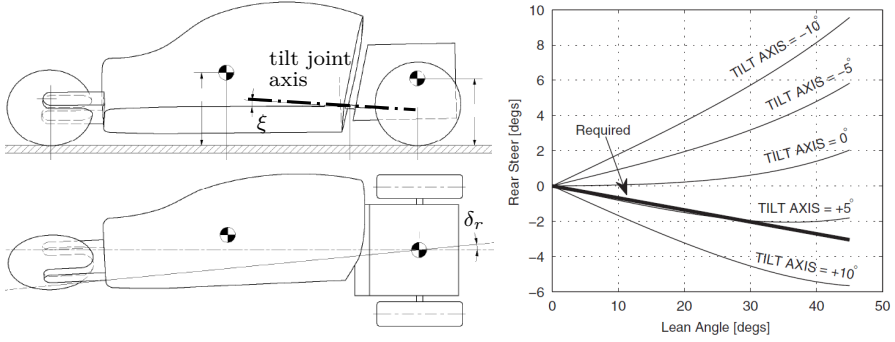


Figure A.16: Longitudinal inclination (pitch) of the tilt joint axis [24].

Experimental tests have confirmed that the DTC system could generate large moments in response to rapid steer inputs, and that vehicle rollover could occur. Barker states that using a DTC system and limiting the rate at which the driver could apply steering (through fitment of a steering damper) may provide a means of ensuring vehicle stability at the expense of some agility). Drew and Barker suggest that a STC system need to be implemented in order to ensure the vehicle stability in transient maneuvers.

Berote's work Berote continues the work of his colleagues by first developing a new dynamic model of the Clever vehicle based on a multibody formalism (MBS). The MBS model, built with the SimMechanics software package [104], involves 7 articulated rigid bodies whose relative motions properly account for the suspension dynamics (Fig. A.17). In addition, the tire nonlinear dynamics is modeled using Pacejka's 'Magic Formula' for conventional car tire [40].

The model has been compared with test rig experiments conducted on the prototype Clever vehicle and a good correlation has been obtained in roll and bounce motion. As a second step, the lateral dynamics model is validated

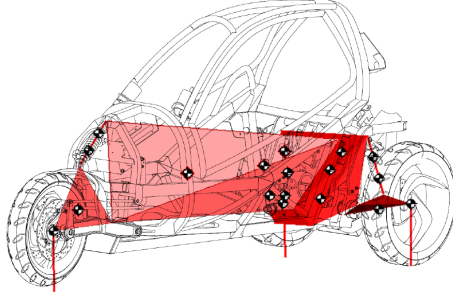


Figure A.17: Visualization of the MBS model proposed by Berote [25].

and the tire model parameters are identified based on ground test maneuvers. Simulations showed that the load transfer across the rear module becomes critical when turning from one direction to another (i.e., the transient phase in a 8-shape maneuver). This is due to the peak in the initial lateral acceleration resulting from the rear steer kinematics combined with a high load transfer associated with a large tilt angle error. Based on a simplified 2-d.o.f. model, Berote studies the limiting stability conditions [25] and introduces the notion of moment reserve, i.e., the torque that can be applied by the actuators before risking rollover, which depends on the vehicle tilt angle and lateral acceleration.

In his thesis dissertation, Berote investigates a combined SDTC system for the Clever vehicle with the aim to improve its transient stability [25]. To this extent, he linearizes an analytic model which includes the lateral and suspension dynamics and the hydraulic system. The steering action δ is taken into account as a modulation of the driver steering input δ_{sw} , as shown in equation (A.13), while the tilt input is the hydraulic tilting system valve opening (Fig. A.18).

$$\delta = \delta_{sw} + \delta_c \quad (\text{A.13})$$

The two modes are combined continuously rather than switched according to the vehicle advance speed. Berote claims that using a negative gain feedback $K_{\delta\theta}$ between the tilt error and the steer input reduces the amount of steering at the front wheel in order to reach the require tilt angle more rapidly, possibly leading to some countersteer.

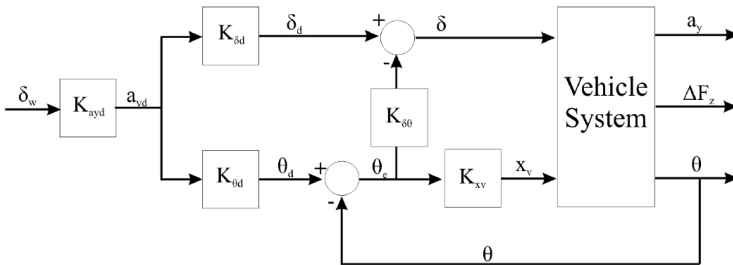


Figure A.18: Block-diagram for SDTC as proposed by Berote in [25].

Berote notes that the optimal STC behavior varies as a function of speed with smaller steer inputs required at higher speeds. Therefore, simulations of lateral acceleration response to a step steer input are used to derive values for a nonlinear speed-dependent active steering gain (Fig. A.19).

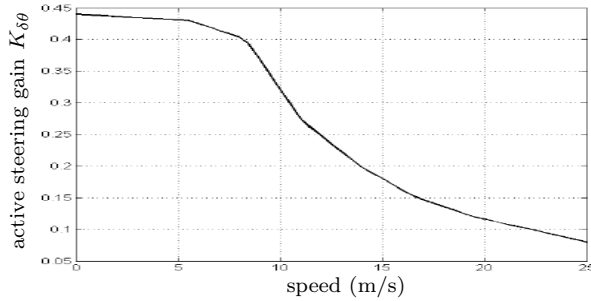


Figure A.19: Active steering gain as function of the speed [25].

Simulation results are limited to a single step steer input conducted at a forward speed of 30 km/h. No experimental verification of the combined SDTC system is carried out. An improved settling time and a lower overshoot for the SDTC are observed compared to the original DTC. The resultant load transfer for a demanding maneuver using the new control method was shown to be approximately 15% less than the original value.

Robertson's work Eventually, Robertson is the last researcher involved in the CLEVER project. He studied the SDTC improvements and its effects on the driver ability with on-track experimental tests [13].

Based on track experiments, he observes that for steady state cornering and low transience U-turn maneuvers, the roll stability is not improved by the SDTC, which is not surprising considering the low transient nature of the maneuvers. However, for highly transient ramp steer, the SDTC system appears slightly more effective for some speed range. At lower speed, the SDTC performs in a similar manner to a DTC system. However, higher vehicle's forward speed yields significant reductions in the wheel load variation while using the SDTC mode. At 10 m/s the wheel load transfer in SDTC mode is said 36% less than that observed for the original DTC mode, significantly enhancing the vehicle's stability [105]. No countersteering action is observed but the active steering system still reduces the rate at which the front wheel steer angle δ_f increases during cornering. Even without countersteering, results show that the delay in the front wheel steer angle response allows the DTC actuators to begin tilting the cabin, before large lateral accelerations are generated.

Regarding the driver ability, the track experiments shows that the driver must adapt his steer input to SDTC system in order to maintain path following. The impact of revised steer input on roll stability is investigated through simulations [55]. the influence of the active steering gain $K_{\delta\theta}$ on the vehicle's trajectory and its roll stability is examined by Robertson who shows that higher

gains improve stability but lower the path following ability. The subjective assessment confirmed the simulation tendencies with human drivers, reporting that generating the revised steer input felt natural after a short period of acclimatization. However, only moderate cornering tests are conducted and a more severe maneuver with higher lateral accelerations may lead to greater influence of the SDTC mode.

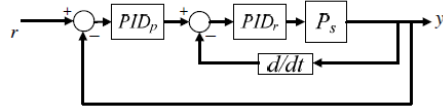
In addition to this experimental investigation, Robertson studies the controller sensitivity to dynamic parameters. The vehicle payload is shown to have little influence on the transient stability while the height of the rear module's center of gravity was found to contribute significantly to the total roll moment; lowering it brings benefits to both transient and steady state roll stability. Finally, simulations are used to show that a feedforward element for SDTC could achieve a fast active steering response and a 30% reduction in the load transfer in transient conditions [13].

A.1.5 National Chiao Tung University of Taiwan (2008 - 2009)

In 2008, a four-wheeler NTV has been developed with a diamond-shape configuration by the National Chiao Tung University [106] in Taiwan. The special vehicle layout, together with a 90-degree steering of the front and rear wheels, enables in-place yaw motion which is very practical for parking (Fig. A.20.a). The model used in their investigation is composed of 11 bodies and has been derived through a Lagrange-based approach. Regarding the NTV control, both STC and DTC have been investigated [59]. To this extent, they developed a double-loop PID with the inner loop on the tilt angle and the outer loop on the tilt rate (Fig. A.20.b). The control is performed on the vehicle tilt angle, however, there is no explanation given on how this reference tilt angle is computed. Simulation results are validated with the help of an experimental maneuver (i.e., slalom test) on the developed prototype.



(a)



$$PID(s) = \frac{K_D s^2 + K_P s + K_I}{s}$$

(b)

Figure A.20: (a) The diamond-shape prototype (b) The double PID loop controller [59].

A.1.6 TU Wien (2011)

Under the supervision of Prof. Plöchl, a NTV prototype has been developed in 2011 at the Technical University of Vienna (TU Wien). The vehicle has a delta configuration with a tilting rear suspension and a unique front steerable wheel (Fig. A.21.a). The prototype features one actuator on the steering and one on the tilting d.o.f. enabling STC and DTC respectively.

The vehicle is modeled as a multibody system with the help of the software SIMPACK [45]. In order to study the dynamics of the NTV, Edelmann and Plöchl developed a simplified model consisting in a lumped motorcycle model by merging the rear wheels into one substitutive wheel (Fig. A.21.b). The resulting 4-body system is then modeled analytically following the work of Sharp on motorcycle modeling [44][41]. In addition, the simplified model is further linearized and compared with the MBS model. Based on handling diagrams, the effects of the nonlinearities are quantitatively observed and it is concluded that the simplified model matches well the vehicle characteristics, up to a normalized lateral acceleration of 0.4 g, the maximum admissible acceleration being approximately 0.53 g for the vehicle.

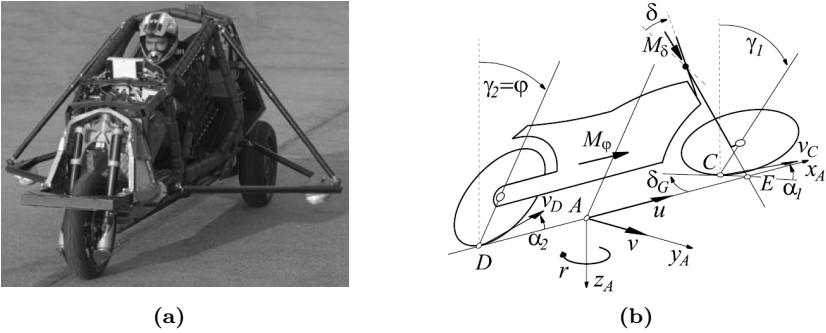


Figure A.21: (a) The delta NTV prototype (b) the simplified motorcycle model [27].

Based on modal analyses performed in steady-state straight line and steady state cornering, Edelmann identifies unstable motorcycle motion modes (i.e., weave and castering) and argues that a stability electronic control is necessary in order to offer a car-like driving experience.

Both DTC and STC are discussed but they focus on STC solely through steering by wire [26]. They use a feedforward control in order to guarantee a fast response to a change of the demanded tilt angle (Fig. A.22). For the feedback control, the error is computed with respect to the desired steady-state configuration while the feedback gains are computed thanks to a LQR approach for the different operating states. Gain scheduling is considered based on the desired lean angle φ_d and the velocity v , nevertheless the advantages of gain scheduling as a function of speed and lean angle are not demonstrated. The desired lean angle for the driver is directly proportional to the steering wheel orientation

δ_{sw} and a driver model is implemented to follow a given trajectory by adjusting the steering wheel orientation.

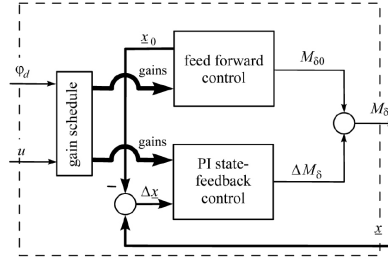


Figure A.22: Block-diagram for the steering tilt control [26].

To validate the tilt controller, a step in desired lean angle φ_d is applied. The lateral acceleration corresponding to this simulation is around 3.5 m/s^2 which is quite significative. Although the controller has been properly implemented and tested, no experimental tests have been presented by Plöchl and his team.

A.1.7 Federal University of Santa Catarina (2007-2016)

Investigations on NTV control has been carried out in Brazil from 2007 to 2016 by Prof. Roquero from the Federal University of Santa Catarina. In order to study NTV, Roquero and his team developed a 4-body model (i.e., chassis and 3 wheels) based on the Lagrange equations (Fig. A.23). The model features 9 degrees of freedom: the 6 degrees of freedom of the chassis and 3 additional degrees of freedom for the wheel vertical travel [63].

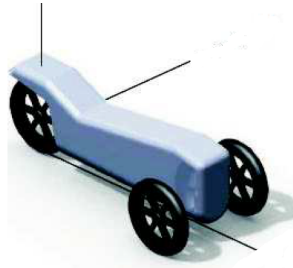


Figure A.23: NTV model with 9 degrees of freedom [63].

Regarding the control approach, steering tilt control is considered alone or coupled to a control of the vehicle advance velocity. The inputs for the system are the steering angle δ and the propulsion torque on the rear wheel while the two outputs are the tilt angle and the vehicle speed. Based on this simplified model, Roquero derives a nonlinear control law based on sliding mode control [63][62]. The reference signal for the tilt angle is computed based on the theoretical formula (A.3) as done by Karnopp and others. This controller is later on tested and validated by the application of perturbations such as side

wind and road irregularities. Roqueiro concludes that through the sliding mode approach, a simple and cheap multivariable controller can be implemented. Later on, De Faria showed that a tridimensional model of a bicycle is sufficient to design sliding mode controllers for a three-wheeled narrow vehicle [63]. In 2016, Morita under the supervision of Roqueiro investigates three methods for the multivariable control: sliding model controller of Slotine, the Unit Vector Control (UVC) method of Utkin, and a recently proposed extension of the UVC controller [64]. The goal is to track the steering angle and the velocity desired by the driver while ensuring tilt stability. It is claimed by Roqueiro and his team that such a nonlinear approach performs better than the previous ones based on reduced linear model. Nevertheless, it is not clear whether the model developed is accurate when it comes to vehicle geometry (suspension motion) and dynamics (wheel inertia and tire/ground contact).

A.1.8 Nagoya University (2012-2014)

In 2012, Furuichi and Matsuno started to investigate NTV control under the supervision of Prof. Fukuda from Nagoya University. They developed a model for a three-wheeled NTV whose wheels are not always in contact with the ground [58, 107]. The model is composed of three different running states and is derived based on the Newton-Euler equations with simplified tilt dynamics (Fig. A.24).

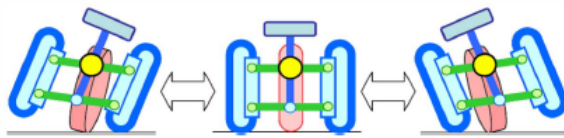


Figure A.24: Three-state dynamic model of NTV [58].

A DTC control is first considered with the desired tilt angle written as a function of the vehicle speed and the steering angle as in equation (A.3). In order to enhance the controller robustness, the perceived acceleration (as defined in Section A.1.1) is considered within a peak-to-peak gain method (LMI-optimization); the main principle is to determine limits of actuation, both in tilting and steering, based on the perceived acceleration and a switching state dynamic model. Thanks to experimental results on a prototype, the control method is validated as the control inputs remain within their pre-defined limits.

A.2 Current research projects

In this section, the focus is on ongoing research projects and studies that have ended during the course of this thesis realization.

A.2.1 Ecole des Mines de Nantes (2011-2019)

In 2011, NTV control has been the subject of a PhD thesis conducted by Mourad under the supervision of Prof. Chevrel and Claveau from the Ecole des Mines de Nantes [29]. This research was conducted in collaboration with Lumeneo, manufacturer of the Smera, an electrical NTV equipped with four wheels (Fig. A.25). The investigation concerned DTC and its improvement through a steering angle modulation (SDTC).



Figure A.25: The Smera vehicle developed by Lumeneo [28].

In accordance with Snell's work [53], they propose to regulate the perceived acceleration rather than tracking a tilt angle [56]. They claim that perceived acceleration regulation erases the steady-state error regardless of modeling approximation as in steady-state configuration, the measured perceived acceleration should always be zero. In addition, it allows for an easier treatment of the road bank (as defined in Section A.1.3).

Despite a built prototype and the availability of a proper multibody model [47], Mourad developed a simplified model based on the work of Rajamani and his team [19]. The model features 3 d.o.f.: the lateral position y , the tilt angle θ and the yaw rate $\dot{\psi}$ and is linearized around the upright position and a given advance speed. In addition, the model is reduced to 4 states $(\dot{y}, \dot{\psi}, \theta, \dot{\theta})$ by considering only the observable and controllable states of the model. Although it is based on Rajamani work, the obtained linearized model neglects the gyroscopic effect, the wheel load transfer and the tire dynamics nonlinearities. Regarding the system input, the steering control is performed by modulating the driver steering angle δ_{sw} by an input steering angle δ_c as done by Minnesota team and Clever team [12, 25, 13].

$$\delta = \delta_{sw} + \delta_c \quad (\text{A.14})$$

The tilting control is performed through the application of a tilting torque M_t . The main advantage of this approach is that no steering modulation (i.e., $\delta_c = 0$) corresponds to the DTC control for which the front steering input δ is directly computed based on the driver steering input δ_{sw} .

The linear model is extensively tested and observations are made on the influence of parameters like the speed V and the steady-state lean angle θ^* . It is observed that the speed variation has a stronger influence than the lean angle one on the transfer function modules, especially between the δ input and the $\dot{\psi}$ output.

Later on, the lateral velocity $\dot{y}$ is replaced by the perceived acceleration which is now considered as the output signal to be regulated to zero (Fig. A.26). Based on this new model, Mourad defines a well-posed H_2 problem that minimizes the regulated signals which are composed of the output deviation (i.e., the perceived acceleration) and the 2 inputs, all multiplied by a static gain matrix, Q_0 . Adjusting the static gains of the 3 regulated signals allows Mourad to design either a STC, a DTC or a SDTC controller (Fig. A.27). The resulting control architecture is of low complexity with a feedback term on the complete states and a feedforward term on the driver steering angle δ_{sw} and its derivative $\dot{\delta}_{sw}$.

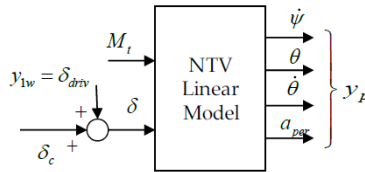


Figure A.26: The inputs and outputs for the NTV linear model [57].

In addition, a gain scheduling on the vehicle speed is considered. The control gains are computed for each speed, keeping constant parameters for the static gain matrix Q_0 . The controller robustness and performances are tested through time simulations. The scenario is defined by the application of a given driver steering angle. The latter is built up on a time interval of 7 seconds and kept constant afterwards. The steady state cornering radius is of approximately 23 meters for a speed of 8 m/s which corresponds to a lateral acceleration of approximately 2.8 m/s^2 . Both the STC and SDTC leads to countersteering. All modes ensure a perfect regulation in steady state. Although the DTC solution leads to good performances, the added use of the steering input (SDTC) decreases the perceived acceleration by 85% and the tilting torque by 60% (Fig. A.27). The stability of the speed gain-scheduling controller is also verified [29].

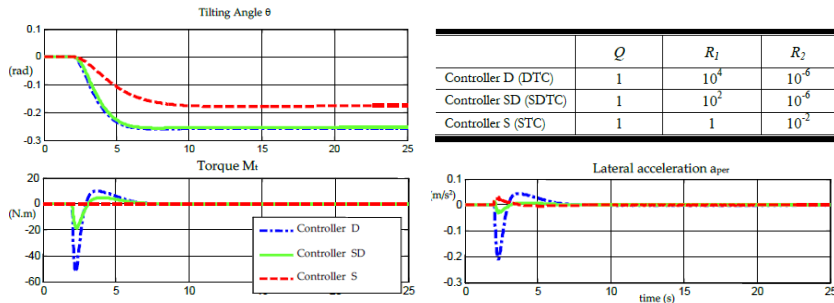


Figure A.27: Mourad's results on SDTC control [57].

In 2015, nonlinear control is envisaged by Claveau with a 5-d.o.f. bicycle model that includes the longitudinal dynamics [30]. Unlike their previous work, they consider the tracking of the tilt angle θ throughout the use of three inputs: the steering angle modulation, the tilting torque and a rear propulsion force. Based on few assumptions on the tire forces model, the flatness⁶ of the 5-d.o.f. model can be verified and a set of 3 flat outputs is found [30]. Then, the longitudinal-lateral dynamics of NTV is linearized thanks to a static state feedback and a simplified PID controller is used to track the 3 flat outputs. The trajectories for the three flat outputs are generated based on the driver steering angle and the propulsion force. The system linearization is verified and the controller is tested with respect to the linear SDTC approach developed above. Results call for some motivating perspectives but show that the nonlinear controller still needs to be optimized in order to outperform the linear controller. If the nonlinear approach has the advantage to treat the longitudinal dynamics and bypass the gain-scheduling design, it is important to remind that feedback linearization has a tendency to erase a part of the original dynamics making the stability analysis more complicated.

More recently, Nguyen has focused on DTC design for the Smera with a linear parameter variant (LPV) model [108]. He considers constraints on the state and the inputs through a LMI-based optimization in multi-objective settings [108]. He clearly demonstrates the interest in considering the DTC actuator limitations in the H2 control design. The developed controller shows a better tracking, compared to a LTI-based controller derived at a given speed of 8 m/s.

A.2.2 University of Tokyo (2015-2017)

From 2015 to 2017, the dynamics of free-to-caster NTV has been investigated by Jeffrey Tan, under the supervision of Prof. Suda from the University of Tokyo. This research has been conducted with the support of Equos⁷. In his first work, Tan focused on the use of a gyroscopic moment to counterbalance the rollover instability (i.e., inner wheel lifting) that occurs when applying a direct tilt control (Fig. A.28).

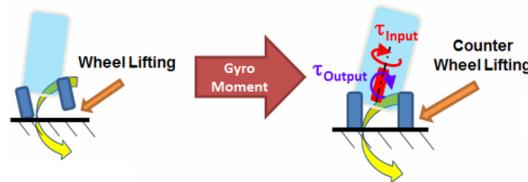


Figure A.28: Stabilization by a gyroscopic moment of a NTV roll motion [109].

⁶More details on this concept and on sliding mode control can be found in [62].

⁷Equos is a research and development company specialized in automotive collaborating with Toyota [31].

Both a simplified bicycle model and a full MBS Adams model of the studied vehicle are developed. The reference signal is the tilt angle which is computed as a function of the lateral acceleration, a_c which is itself derived as a function of the speed V and the steering angle δ , together with their respective rates $\dot{V}$ and $\dot{\delta}$ [109]. There are no active gyro control and the gimbal is simply pitched linearly as a function of the desired tilt angle, θ_d (the linear function being determined based on the maximum tilt and gimbal angle). The gimbal control is designed to be triggered with a slight delay according to the response time (0.62 s in this case) of the tilting device, in order to optimize the response timing in counterbalancing the vehicle rolling stability.

They simulate a curve entry with and without gyroscopic moment and observed improvements in terms of cornering stability (i.e., wheel lifting is avoided). For a sharp steer maneuver with a 15-degree steering angle, the vehicle behaves better with than without the gyroscopic moment; however some wheel lifting occurs. In [109], a prototype is shown but no experimental results are presented.

Later on, Tan studied the dynamic behavior of a NTV whose steerable wheels are free-to-caster [32]. For such a vehicle, the steering wheel input from the driver is transformed in a tilt angle command (angle or torque) forcing the vehicle to lean, while the steerable wheel steers themselves thanks to the caster effect (Section A.1.3). The vehicle studied has a delta configuration and is once again modeled via the MBS software Adams (Fig. A.29).

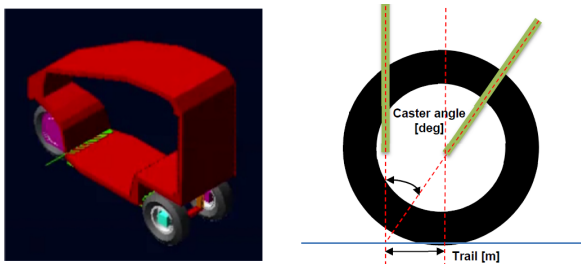


Figure A.29: Adams MBS model developed by Tan and illustration of the trail and the caster angle [32].

Tan considers three driving scenarios: a J-turn, a constant turning radius and external perturbations [33]. The first two are realized through the application of a step in the tilting angle, while the third corresponds to a reverse steering torque applied to the front wheel. For each of the scenarios, two parameters are studied: the trail and the front caster angle (Fig. A.29). The study shows that caster angle and trail have a minor impact on the steering angle response time. The turning radius increases with the caster angle but the trail still has a minor effect. Both an increase of the caster angle and the trail leads to a lower overshoot in the steering angle response which intrinsically means that the steering stability is enhanced. Finally, both parameters influence very little the steering resiliency toward steering disturbance.

Thereafter, Tan conducted three steering stability analyses: a low speed steering analysis, a slalom steering analysis and an uneven road surface steering analysis

[33]. At a speed of 20 km/h, the vehicle exhibits an unstable behavior due to a higher phase shift causing steering instability. For slalom, an important lean angle input leads to the divergence of the free steering angle. The case of an uneven road, simply modeled as a step in normal force, also leads to instability. Latter on, simulations have been validated with the help of a new NTV prototype (Fig. A.30). During steady state cornering, less than 5% error between the simulation and the actual vehicle turning radius was observed while for a single-lane change, both experiments and simulations show similar behavior. Based on these observations, Tan concludes that the vehicle model is sufficiently accurate to study the dynamics of free-to-caster NTVs.



Figure A.30: The prototype NTV developed by Equos working based on a FTC principle [32].

In addition, Tan investigates the vehicle rolling stability based on the experimental vehicles. At a given speed of 20 km/h, the vehicle is excited through the tilt input and it is shown that at the vehicle roll natural frequency (about 1 Hz), the phase shift of the steering response is about 90 degrees. Based on this observation, design adjustments are performed and it is shown that the low speed stability is improved.

A.2.3 University of Waterloo (2014 - 2018)

From 2014, a research team at the University of Waterloo has been studying active camber modules for NTVs. Their goal is to enhance the vehicle lateral and roll stability by slightly tilting the vehicle while cornering. This makes their approach different from the ones presented previously for which the vehicle is fully tilted and needs to be stabilized. Although their focus is mainly on the mechanical design of the camber module, they also consider active tilt control and the effects it can produce on the vehicle stability. This research is motivated by the development of integrated in-wheel systems which takes part of a collaboration between the University of Waterloo (Prof. Khajepour) and the electric motor supplier Nidec Corp (Fig. A.31).

In 2014, Kim developed a model of a three-wheel vehicle with an active tilting system by using the MapleSim software [110]. The tilting vehicle was compared with the non-tilting vehicle that has the same specifications, to determine the performance of the active tilting system at different longitudinal speeds and

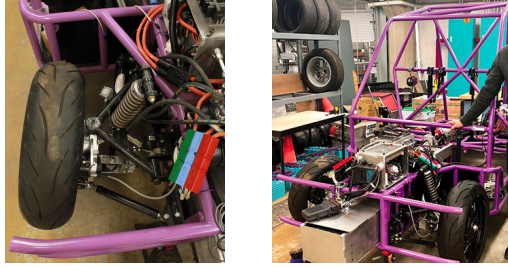


Figure A.31: Waterloo in-wheel module and testing vehicle [34].

front wheel steering angles. The active tilting system improved the vehicle's maneuverability for different sets of vehicle longitudinal speeds and steering angles, and it was observed that the effect was more significant at higher vehicle longitudinal speeds than at low ones [111].

At the same time, Goodarzi focused on the development of an integrated system that coordinates active front steering, direct yaw moment control and active tilt systems (Fig. A.32). The goal of his research is to find a compromise between improving vehicle dynamic behavior and minimizing the actuator torque requirements. Compared to previous works, he adds the *direct yaw moment control* which consists of a lateral stability enhancement through the use of braking and traction forces on the rear wheels. The three-wheeler is first modeled thanks to Lagrange's equations then linearized. By defining a desired speed, yaw rate and lean angle ($V_d, \dot{\psi}, \theta$), a handling performance index is derived as a quadratic function of the errors with respect to the desired quantities. Based on this handling performance index, a linear quadratic regulator (LQR) problem is formulated and solved, enabling the derivation of feedback gains for the three control entities. Simulation results reveal that the integrated system leads to improved results compared to each of the three control entities alone [35].

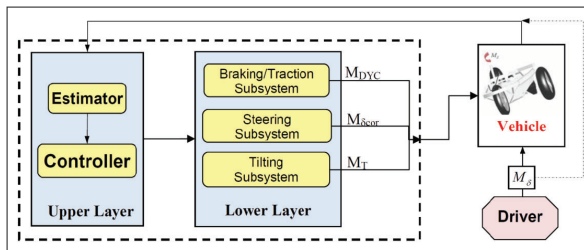


Figure A.32: Block-diagram of the integrated control system [35].

Later on, Ataei has focused his work on the reconfigurability of the integrated active system. In his thesis, the reconfigurable controller is developed using the model predictive control (MPC) approach [112]. A reconfigurable vehicle model in matrix form is employed, introducing an actuator reconfiguration matrix and

a corner reconfiguration matrix. The reconfigurability of the controller enables the use of any type of control actuation such as active steering, differential braking, torque vectoring, and active camber on all wheels to reach the stability objectives. Simulations are conducted for various vehicles with different actuators to evaluate the controller performances. Results show that the general reconfigurable controller can be effectively used for different vehicles to improve their stability and safety in different conditions.

A.2.4 Pune University (2015-2017)

Recently, NTV control has been the subject of investigation at the Savitribai Phule Pune University. In 2015, Chatterjee first started to develop a 7-d.o.f. mathematical model to represent the dynamics of a three-wheeler NTV [113]. The model is derived analytically and is based on several assumptions: the suspension, roll and pitch dynamics as well as gyroscopic effects due to the wheels rotation are not considered. With a rudimentary method, Chatterjee shows that longitudinal, lateral and vertical need to be taken into account as a whole. In 2017, Chatterjee investigates DTC and SDTC controls together with longitudinal speed tracking [60]. First a baseline DTC controller is implemented consisting of two PID controllers on the desired tilt angle θ_d and the desired speed V_x , both being derived based on driver inputs through a simplified method (Fig. A.33).

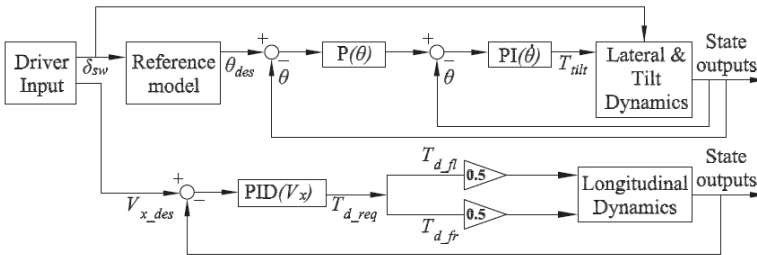


Figure A.33: Block-diagram of the baseline controller [60]. The driver inputs, θ_d and V_x are fed to the PID controllers that controls the roll and longitudinal dynamics separately.

Then a second controller, based on a LQR formulation, aims at controlling the vehicle lateral and longitudinal dynamics through a SDTC approach. The steering input is translated into a desired yaw rate ψ_d that the controller needs to track. Finally, robustness is improved by augmenting the LQR controller by integral states which results in a LQI controller (Fig. A.34).

This multivariable controller proves to be robust against speed variations and against parametric changes in terms of the mass and the inertia of the NTV. It is also observed that the input power requirement of the SDTC controller is quite lower than that of baseline DTC controller. In addition, the improved controller provides better road handling and follows the commanded trajectory

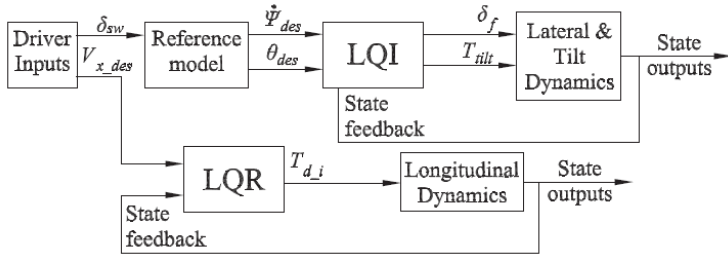


Figure A.34: Block-diagram of the SDTC with LQI controller [60]. For the lateral dynamics, the system state is augmented with the integral of the error for the two tracked signals: the desired yaw rate, $\dot{\psi}_d$ and the tilt angle, θ_d

more closely. Finally, it is less sensitive to the effects of external disturbances, such as a sinusoidal perturbation added to the longitudinal drag.

A.2.5 Resolve project (2016-2019)

The Resolve project⁸ is a European project that aims at developing new vehicle concepts (tilting and narrow track) offering attractive ELV driving experience while containing as much as possible the vehicle energy consumption (Fig. A.35)[46]. Among the European consensus, the University of Pisa has been focusing on the dynamic analysis and the design of a four-wheeler NTV while the University of Warwick has recently started to investigate NTV control through nonlinear methods.



Figure A.35: The two demonstrators of narrow tilting vehicle developed in the Resolve project [46].

⁸Standing for *Range of Electric SOLUTIONs for L-category VEHICLES*.

Dynamic analysis - University of Pisa

In 2016, Bucchi started to develop a model for a 4-wheeler tilting NTV [8]. Although no multibody modeling is mentioned, the equations of motion are derived using the Newton-Euler equations following the works of Sharp and Sponziello on 2-wheeler and 3-wheeler modeling [44][114]. The model features 23 bodies which can be grouped into three subsystems: the frame, the front tilting assembly and the rear tilting assembly (Fig. A.36.a).

Based on the model, stability analyses of the vehicle are performed showing an unstable capsaze mode on the overall speed range [9]. Then a sensitivity analysis is performed on the front trail showing significant effects on almost all the natural modes, especially the weave and second castering modes when low speeds are considered (Fig. A.36.b).

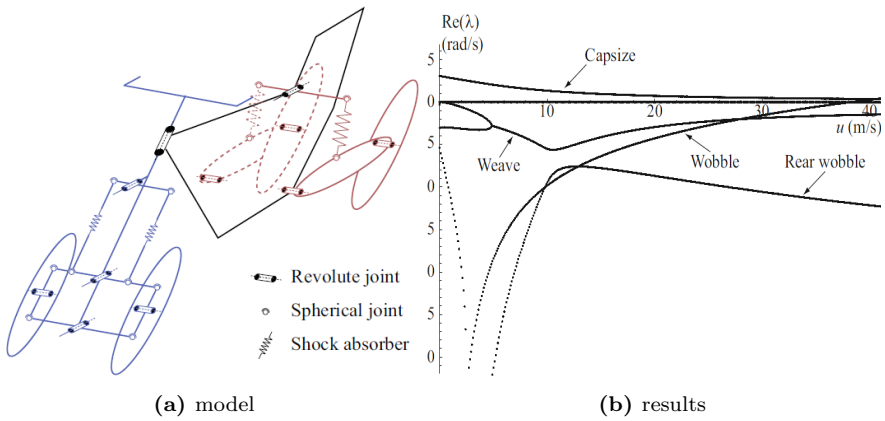


Figure A.36: The stability analysis as a function of the advance speed based on a complex MBS model for the NTV [9].

Control analysis - University of Warwick

In 2018, Ren starts investigating DTC-based control under the supervision of Prof. Greenwood at the University of Warwick. Although there are part of the Resolve project, they base their study on a simplified model for the NTV roll and lateral dynamics [37]. The model is composed of 3 states gathered in the state vector, x : the yaw angle, the lean angle and the lean rate with 2 inputs: the steering input and the tilting torque. By linearizing the model, a simple PI controller and a gain-scheduling PI controller are derived.

Based on the full nonlinear model in equation (A.15) for the roll dynamics, an input-output linearization is performed with the tilting angle θ as the output and the tilting torque M_t as an input. The desired lean angle θ^* is computed based on the steering angle and the steady state cornering assumption as in

equation (A.3). The linear output dynamics is thus rewritten in equation (A.16).

$$\ddot{\theta} = L_f(x) + B(\theta)M_t \quad (\text{A.15})$$

$$\ddot{\theta} = k_1(\theta^* - \theta) - k_2\dot{\theta} \quad (\text{A.16})$$

Later on, the $B(\theta)$ term is linearized around the nominal upright position, $\theta = 0$ and the differences between $B(\theta)$ and $B(\theta)|_0$ is seen as a disturbance which is estimated based on the previous time step (Fig. A.37).

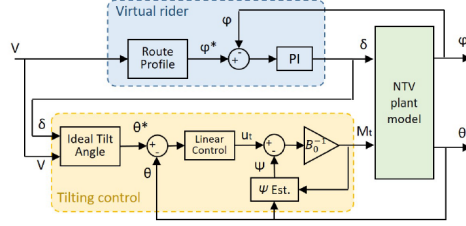


Figure A.37: Block-diagram of the tilting control of NTV [37]. The roll dynamics is linearized through feedback and the tilt error dynamics is tuned via a linear controller.

A 8-shape path is used together with a simple PI virtual rider model (Fig. A.37) to validate the two controllers developed. The nonlinear controller performs better than the conventional linear and gain-scheduling controllers with less roll angle tracking error under different speeds. In addition, it can help riders to improve the yaw rate tracking performance [37].

A year later, Ren has continued his investigations on NTV control using a torque vectoring method to assist the rider in balancing the narrow tilting vehicles, thus reducing the countersteering requirements [38]. The proposed approach is designed as the combination of two torque controllers: a steer angle-based torque vectoring controller and a tilting compensator-based torque vectoring controller (Fig. A.38). There is no torque applied on the tilting degree of freedom so the rider balances the vehicle and performs an *assisted* steering tilt control. The developed controllers have the ability to reduce the countersteering requirements from the rider and to improve the tilting behavior during turning an NTV [38].

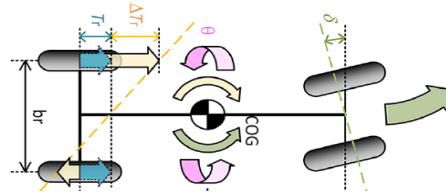


Figure A.38: Torque vectoring, consisting in the modulation of the torques on the rear driven wheels, is used to control the vehicle lateral dynamics and consequently improve the NTV tilt stability [38].

Multibody formalisms for solving optimal control problem

B

This appendix is a copy of the complete article entitled "Comparison and Analysis of Multibody Dynamics Formalisms for Solving Optimal Control Problem" that appeared as a book chapter in *IUTAM Symposium on Intelligent Multibody Systems – Dynamics, Control, Simulation*

Chapter 3

Comparison and Analysis of Multibody Dynamics Formalisms for Solving Optimal Control Problem



Quentin Docquier, Olivier Brûls, and Paul Fiset

Abstract Optimal Control methods are increasingly used for the control of multi-body systems (MBS). This work analyzes the different dynamic formulations and compare their performances in solving Optimal Control Problem. The focus is on minimal coordinates and the derivation of the dynamics via the recursive methods for tree-like MBS (i.e., the so-called Newton-Euler and Order-N recursive algorithms). The different formulations are introduced and their derivations are discussed. A benchmark case study (i.e., a 3D series manipulator balancing an inverted pendulum) is modeled and a series of manipulation tasks (movement of the end effector in the 3D space) are performed. The OCP is formulated and solved with the help of the CasADi software while the dynamic formulations are generated by the Robotran software. Results show that the implicit and semi-explicit formulations derived via the Newton-Euler recursive algorithm lead to faster computation of the OCP than the explicit formulations. This is explained by a more compact expression for the implicit dynamics. However, a lower number of high local minima is observed with the explicit formulations for the most extreme robot manipulations.

Keywords Multibody system dynamics · Direct optimal control · Newton-Euler recursive algorithm · Tree-like system · 3D serial robot · Inverted pendulum

3.1 Introduction

Nowadays, Optimal Control (OC) methods are increasingly used for the control of mechanical systems. Many industrial applications already make use of it, e.g., for the trajectory planning of industrial robots, while the scientific community continues

Q. Docquier (✉) · P. Fiset
Université Catholique de Louvain, Louvain-la-Neuve, Belgium
e-mail: quentin.docquier@uclouvain.be; paul.fisette@uclouvain.be

O. Brûls
Université de Liège, Liège, Belgium
e-mail: o.bruls@uliege.be

© Springer Nature Switzerland AG 2019
E. Zahariev, J. Cuadrado (eds.), *IUTAM Symposium on Intelligent Multibody Systems – Dynamics, Control, Simulation*, IUTAM Bookseries 33,
https://doi.org/10.1007/978-3-030-00527-6_3

55

to investigate new possibilities in the fields of active orthoses and autonomous driving. Optimal Control of multibody systems (MBS) is of particular interest for many mechanical systems such as vehicles, robots or the human body as it offers the possibility to compute beforehand a control sequence on a given time interval such that a certain optimality criterion is achieved. This makes OC methods suitable for multibody systems whose dynamics involves non-linearities and rotations in the tridimensional space. Moreover, the computation of control command inputs via optimization enables the control of underactuated MBS characterized by a lower number of inputs than degrees of freedom. However, the performances of the optimized control inputs strongly depend on the mathematical model used to represent the dynamics of the controlled system.

Optimal Control is also essential for Model Predictive Control (MPC) of non-linear systems. MPC consists in determining the current control inputs by solving an Optimal Control Problem (OPC) on a finite future time horizon. In case of mechanical systems with fast dynamics, fast updates of the control inputs are required, which means that the OCP online solving must be carried out in real-time. This leads to many investigations on the OCP solving efficiency including for multibody control purpose.

Optimal Control of multibody systems has already been approached on many different angles. There have been a multitude of works to develop general purpose algorithms for optimal trajectory planning of MBS. Among them, studies have focused on *constrained* MBS characterized by loops of bodies and algebraic equations [12]. For such MBS, different methods of index reduction for the differential algebraic equations (DAE) system have been investigated (e.g., coordinate partitioning [21] and Baumgarte stabilization [3]). Such methods are also of great importance in the case of non-minimal parameterization of the system motion. In [11] and [14], different parameterizations of the special orthogonal group $SO(3)$ have been investigated with special integrators adapted to the parametrization. These non-minimal parametrizations are of particular interest for many applications (e.g., flying vehicles such as drones) as they offer a singular-free representation for the rotations.

On the other hand, minimal coordinates have also drawn the attention of researchers as they are appropriate for the representation of many industrial applications (e.g., serial manipulators). During the two past decades, the optimal trajectory planning of industrial robots has been a very popular research subject [2, 6, 18, 19]. These robots generally possess a tree-like structure (i.e., without loop of bodies) which leads to *unconstrained* systems in the case of a joint coordinates approach.

In robotics, the trajectory planning is usually referred as the inverse dynamics problem for which the objective is to determine the joint torques for a prescribed motion of the controlled system [2]. The use of optimization methods is often necessary for the control of underactuated or overactuated systems for which there is still a latitude in the way to actuate them. Among the studied problems, the case of non-minimal phase systems represents an additional challenge as the inverse

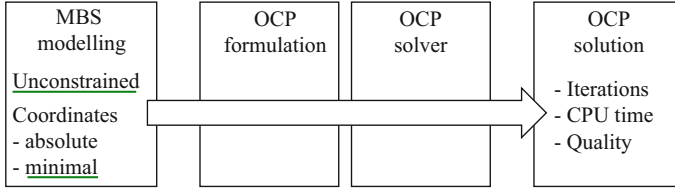


Fig. 3.1 In this work, the focus is on the influence of the MBS modeling on the solving OCP performances

dynamics is, in that case, non-causal which means that the control should start before the beginning of the trajectory [2].

Generally, the research on OC for MBS has mainly focused on the development of new methods for Numerical Optimal Control that are based on a given formulation of the multibody dynamics [2, 5, 19] and [17]. Nevertheless, in [6], Diehl gives a good overview on the different dynamic formulations and how they could influence the Optimal Control solver. He suggests that an implicit formulation of the dynamics through the Newton-Euler Recursive scheme should lead to a lower evaluation cost per iteration. However, to our knowledge, there has been no thorough investigation on the dynamic formulations and its influence on the solving of optimal control problem for *unconstrained* multibody systems. Regarding the numerical formulation of OCP, two methods exist: the multiple shooting and the direct collocation [4]. For the multiple shooting, the integration and the optimization are treated separately. The choice of the integrator is left free and can be adapted to the system dynamics. On the other hand, the direct collocation method consists in imposing the dynamics equations at intermediate collocation points. The integration and the optimization are treated at the same level. The advantages of collocation methods are that they lead to a very sparse NLP and that they show fast local convergence. In addition, they treat unstable system well and can easily cope with state and terminal constraints [6]. These reasons led us to consider the collocation method in the case study of this paper (see Sect. 3.4).

In this work, we analyze different multibody dynamic formulations and compare their performances in terms of convergence, accuracy and computation cost in the OCP framework. The aim is not to develop new numerical methods for the formulation of the OCP or to propose a new type of solver but to provide some insights on existing MBS formulations and their suitability in the formulation of Optimal Control Problems (see Fig. 3.1). Among the different MBS formalisms, we focus on minimal (i.e., relative) coordinates and the derivation of the dynamics via the recursive methods for tree-like MBS (i.e., the so-called Newton-Euler and Order-N recursive algorithms). The derivation is done symbolically through the use of the Robotran software [7]. The symbolic equation generator of Robotran has already been proved to be particularly efficient in the derivation of compact expressions for the dynamics of multibody system [15]. However, its performances for OCP solving have never been thoroughly analyzed.

In order to compare the formulations, a benchmark has been developed. It consists of a 3D serial robot arm with an inverted pendulum attached to the end effector. An important point is the underactuated nature of the system (due to the free rotation of the pendulum) which requires the use of optimization to determine the robot control inputs. In addition, the balancing of an inverted pendulum is a well-known task that has already been largely investigated. The system being tridimensional, its non-linear and complex dynamics should highlight the differences between the considered dynamic formulations.

This chapter is organized as follows: first the theory related to the formulation of the OCP and the multibody modeling are respectively presented in Sects. 3.2 and 3.3. The different formulations are introduced and their derivations are discussed. Then, Sect. 3.4 presents a benchmark case study for an *unconstrained* multibody system. First the MBS system (i.e., a 3D robot arm with an inverted pendulum) is described, then the optimal control task is defined. To this extent, the CasADi software [1] is used to formulate and solve the OCP while the dynamic formulations are provided by the symbolic MBS software Robotran [7]. Finally, results are presented for a series of tasks realized with the different representations of the dynamics. These are compared in terms of solver performances: number of iterations, CPU time to convergence and quality of the solutions. Finally, the differences observed for the different formulations are discussed in light of theoretical considerations.

3.2 Optimal Control of Multibody Systems

There are many ways to formulate and solve an Optimal Control Problem. In this Section, we introduce the main principle and define the trajectory planning problem to solve. In Sect. 3.2.1, the continuous problem is presented analytically then the discrete formulation is introduced in Sect. 3.2.2.

3.2.1 Optimal Control Formulation

In a multibody context, the optimal Control Problem (OCP) is formulated as an optimization problem whose solution provides the control trajectory $\mathbf{u}(t)$ that minimizes a given cost function (3.1) and steers the state $\mathbf{x}(t)$ of the multibody system and its algebraic variables $\mathbf{z}(t)$ from a given initial state (3.2) to a given terminal state (3.3), within a given time horizon T (see Fig. 3.2). The OCP then reads:

$$\underset{\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t)}{\text{minimize}} \int_0^T L(\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t)) dt \quad (3.1)$$

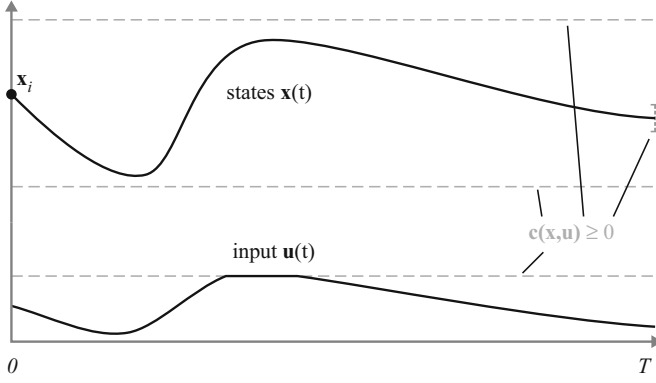


Fig. 3.2 The continuous optimal control problem consists in finding the optimal state and control trajectory on the time interval $[0, T]$

subject to possible path constraints:

$$\mathbf{x}(0) - \mathbf{x}_i = \mathbf{0} \quad (3.2)$$

$$\mathbf{x}(T) - \mathbf{x}_d = \mathbf{0} \quad (3.3)$$

$$\mathbf{c}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t)) \geq \mathbf{0} \quad t \in [0, T] \quad (3.4)$$

The trajectory $\mathbf{x}(t), \mathbf{z}(t)$ and the control inputs $\mathbf{u}(t)$ are subjected to equality constraints denoted $\mathbf{g}$, which typically represent the system dynamics expressed as a set of Differential Algebraic Equations (DAE).

$$\mathbf{g}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t)) : \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z}) \\ \mathbf{a}(\mathbf{x}, \mathbf{z}) = \mathbf{0} \end{cases} \quad t \in [0, T] \quad (3.5)$$

In this equation, $\mathbf{f}$ and $\mathbf{a}$ respectively denote the differential and the algebraic equations that described the physics of the system in terms of the states $\mathbf{x}$, the algebraic variables $\mathbf{z}$ and the control inputs $\mathbf{u}$. This general framework for the dynamics is at the core of the CasADi OC program and its permits the modeling of the dynamics of the multibody system as a system of DAE (see Sect. 3.3). The optimization problem defined by the set of Eqs. (3.1), (3.2), (3.3), (3.4) and (3.5) has in general no analytical solution. However it can be solved numerically through its discretization on the time interval $[0, T]$.

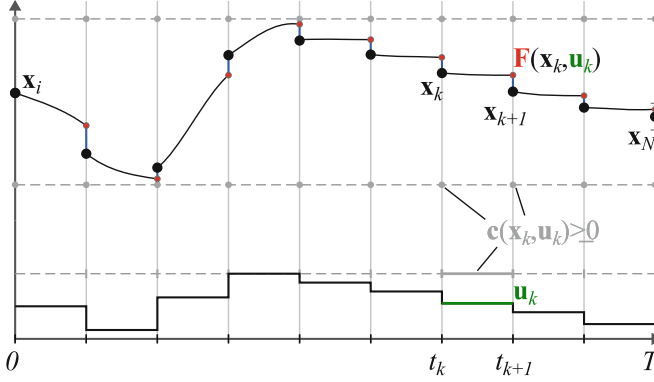


Fig. 3.3 Numerical OCP with the dynamic constraints (3.10) that are unsatisfied. The decision variables are the states at the discrete points and the piecewise constant control inputs

3.2.2 Numerical Optimal Control Methods

There are several approaches to obtain a numerical solution of an OCP. Among them, the *direct methods* consist in discretizing the time horizon $[0, T]$ into N time intervals $[t_k, t_{k+1}]$ such that $t_0 = 0$, $t_N = T$ and $k \in 0, 1, \dots, N-1$. The time functions $\mathbf{x}(t)$, $\mathbf{u}(t)$ and $\mathbf{z}(t)$ that were the unknowns of the system (3.1), (3.2), (3.3), (3.4) and (3.5) are replaced by a sequence of $N+1$ discrete states $\mathbf{x}_k$, $\mathbf{z}_k$ and N piecewise constant controls $\mathbf{u}_k$ which are now the decision variables of the optimization problem in its discrete form (see Fig. 3.3).

The cost function (3.1) is replaced by a numerical approximation p which is a function of the decision variables vector $\mathbf{w} = \{\mathbf{x}_0, \mathbf{u}_0, \dots, \mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k \dots \mathbf{u}_{N-1}, \mathbf{z}_{N-1}, \mathbf{x}_N\}$. Finally, the path constraints (3.2) and (3.3) are re-expressed in terms of the decision variables vector $\mathbf{w}$. This parameterization of the original continuous OCP is referred as a nonlinear program (NLP).

$$\underset{\mathbf{w}}{\text{minimize}} p(\mathbf{w}) \quad (3.6)$$

subject to path constraints on the decision variables:

$$\mathbf{x}_0 - \mathbf{x}_i = \mathbf{0} \quad (3.7)$$

$$\mathbf{x}_N - \mathbf{x}_d = \mathbf{0} \quad (3.8)$$

$$\mathbf{c}(\mathbf{w}) \geq \mathbf{0} \quad (3.9)$$

In order to respect the system dynamics, the constant control input $\mathbf{u}_k$ on the time interval $[t_k, t_{k+1}]$ is linked to the two adjacent differential states $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ through the explicit numerical integration $\mathbf{F}$ of the continuous dynamics (3.5) on the same discrete time interval (see Fig. 3.3).

$$\mathbf{x}_{k+1} - \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k) = 0 \quad (3.10)$$

In addition, the differential and algebraic states are linked at each discrete time through the set of algebraic equations.

$$\mathbf{a}(\mathbf{x}_k, \mathbf{z}_k) = 0 \quad (3.11)$$

As mentioned in the introduction, a collocation method is used to deal with the constraints (3.10) and (3.11). This technique consists in defining collocation points between t_k and t_{k+1} at which the states, represented as polynomial functions, have to satisfy the continuous dynamics (3.5). The integration scheme implies the definition of extra variables (i.e., the polynomial coefficients) and additional constraint equations. This leads to a very large but sparse NLP.

The optimization problem defined by Eqs. (3.6), (3.7), (3.8), (3.9), (3.10) and (3.11) can be solved with different techniques (e.g., an Interior Point method) which generally require the derivatives of the cost function and constraint equations with respect to the decision variables. To this extent, the software CasADi [1] provides sensitivity analysis tools for ODE and DAE system that allows the automatic generation of the derivatives necessary to solve the NLP. To do so, CasADi uses state-of-the-art algorithmic differentiation [10].

3.3 Multibody Systems Modeling

It is well-established that several formalisms can be used to model multibody systems (MBS). Among them, both the absolute and relative coordinates can be considered for optimal control, resulting in different structures for the NLP to solve [2]. In this study, the focus is on minimal coordinates and the different formulations of the dynamics. First the relative (i.e., minimal) coordinates approach is briefly described. Then the different forms for the MBS dynamics that can be obtained through recursive algorithms are presented. Finally, the symbolic generation of these dynamics equations is discussed.

3.3.1 Relative Coordinates Approach

In the relative coordinates approach, the parametric configuration of a body is defined relatively to another one called its parent body. The motion degrees of freedom (dof) are defined via the joints that connect the different bodies of the system. If the MBS does not contain any loops of bodies, it inherits a tree-like structure as each body can only have one parent body (see Fig. 3.4). The number of coordinates is minimal and thus reduced compared to the absolute coordinates approach, as the joint constraints are implicitly taken into account by the relative formulation, leading to a pure set of ordinary differential equations.

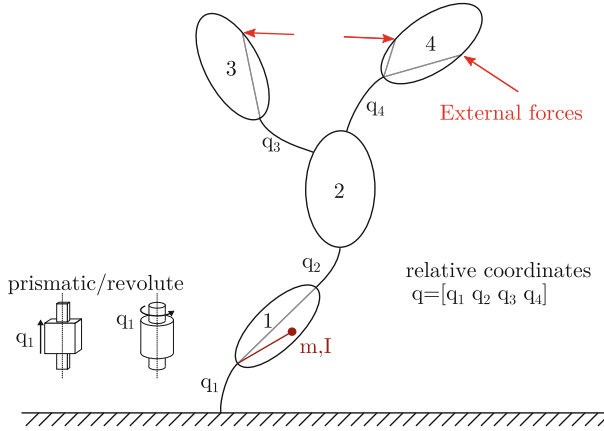


Fig. 3.4 The relative coordinates approach consists in defining the motion of each body relatively to its parent body with the help of a joint coordinate q

In the relative coordinates approach, a MBS is fully characterized by two types of features:

- bodies, defined by their mass, center of mass and inertia tensor (10 parameters in total),
- joints, defined by their location on the body and their nature (i.e., prismatic or revolute, free or constrained).

A tree-like structure might be quite restrictive since many mechanical systems contain kinematic loops (e.g., a four-bar linkage). The relative coordinates approach deals with such closed-loop systems by introducing loop constraint equations. The tree-like structure is restored in two steps. First the loop is cut resulting into two independent branches of bodies. Afterwards the loop closure is ensured by applying algebraic constraints, for instance forcing two points of the created branches to coincide or imposing a given distance between two points. Hence, the derivation of the *differential* equations is based on the tree-like structure of the MBS whether or not the system is constrained. In this work, only unconstrained tree-like MBS are considered within the frame of OCP formulation. However, as the derivation of the differential equations is the same for constrained system, the study of the different formulations of the dynamics will be insightful for both constrained and unconstrained MBS.

3.3.2 Formulations of the Dynamic Equations

In this section, different formulations of the equations of motion are presented for the case of *unconstrained* MBS. Among those formulations, the so-called *Newton-Euler recursive* scheme denoted NER is particularly suitable for minimal coordinates [13]. This method implements two consecutive recursions on the MBS

(a forward kinematics followed by a backward dynamics). It is well-known for its facility of computer implementation and its reduced number of operations necessary to obtain the equations of motion. It expresses the vector of generalized forces $\mathbf{Q}$ as a function of the system kinematics $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$.

$$\Phi(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}) = \mathbf{Q} \quad (3.12)$$

This very efficient formulation, also called inverse dynamics has only a $\mathcal{O}(N)$ complexity thanks to its recursive formulation, N being the size of the MBS (i.e., the number of joints). It is extensively used by roboticists to determine the actuation torques $\mathbf{Q}$ necessary to track a given trajectory $(\mathbf{q}(t), \dot{\mathbf{q}}(t), \ddot{\mathbf{q}}(t))$ for a fully actuated robot.

3.3.2.1 Implicit Formulation

More generally, Eq. (3.12) can be used to represent the dynamics of any MBS systems, in a residual form. In that case, the joint torques/forces $\mathbf{Q}$ can be configuration-dependent (i.e., an elastic joint or a feedback control law) and are generally expressed as a function of the generalized coordinates $\mathbf{q}$ and generalized velocities $\dot{\mathbf{q}}$. It results in an implicit formulation of the equations of motion in terms of the generalized acceleration $\ddot{\mathbf{q}}$:

$$\Phi(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q}) - \mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}) = \mathbf{0} \quad (3.13)$$

According to Sect. 3.2.1, some control inputs $\mathbf{u}(t)$ are considered for the OCP formulation. To this extent, the joint forces $\mathbf{Q}$ can be partitioned into two subsets.

- $\mathbf{Q}_a$ for the active or actuated joints (input of the system)
- $\mathbf{Q}_p$ for the passive joints that can contain dry friction, damping force, etc.

$$\mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}) = \mathbf{Q}_a + \mathbf{Q}_p(\dot{\mathbf{q}}, \mathbf{q}) \quad (3.14)$$

The system (3.13) can be re-expressed as a DAE system of the form (3.5) with the generalized coordinates and velocities representing the differential states and the accelerations considered as the algebraic states.

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} & \mathbf{z} &= \ddot{\mathbf{q}} & \mathbf{u} &= \mathbf{Q}_a \\ \left\{ \begin{array}{l} \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z}) &= \begin{bmatrix} \mathbf{z} \\ \dot{\mathbf{q}} \end{bmatrix} \\ \mathbf{a}(\mathbf{x}, \mathbf{z}) &= \Phi(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{z}) - \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}) = \mathbf{0} \end{array} \right. \end{aligned} \quad (3.15)$$

3.3.2.2 Semi-explicit Formulation

The original Newton-Euler recursive scheme leading to the formulation (3.12) can be modified in order to obtain a semi-explicit form of the dynamics equations for tree-like multibody systems. This new form still preserves the full recursivity even for the computation of the mass matrix [9].

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}) \quad (3.16)$$

$\mathbf{M}$ is the symmetric generalized mass matrix of the system and $\mathbf{c}$ is the non-linear dynamic vector which contains the gyroscopic, centripetal and gravity terms as well as the contribution of the resultant external forces. The difference compared to Eq. (3.12) is that the algorithm first computes $\mathbf{M}$ and $\mathbf{c}$ individually before the evaluation of the left-hand side.

This semi-explicit formulation is often preferred for time integration purposes. However, the factorization of the mass matrix in the recursive scheme requires additional computational efforts of $\mathcal{O}(N^2)$ complexity [9]. The semi-explicit system (3.16) can be re-expressed as a DAE system in a similar way as for the implicit formulation.

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} & \mathbf{z} &= \ddot{\mathbf{q}} & \mathbf{u} &= \mathbf{Q}_a \\ \left\{ \begin{array}{l} \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z}) &= \begin{bmatrix} \mathbf{z} \\ \dot{\mathbf{z}} \end{bmatrix} \\ \mathbf{a}(\mathbf{x}, \mathbf{z}) &= \mathbf{M}(\mathbf{q}) \mathbf{z} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{u}) = \mathbf{0} \end{array} \right. \end{aligned} \quad (3.17)$$

3.3.2.3 Explicit Formulation

The system of Eq. (3.16) can be solved in terms of the accelerations $\ddot{\mathbf{q}}$ via a Cholesky decomposition of the mass matrix leading to the explicit expression $\boldsymbol{\gamma}$ for the acceleration $\ddot{\mathbf{q}}$ in terms of the position $\mathbf{q}$ and velocity $\dot{\mathbf{q}}$. This resolution step yields an additional $\mathcal{O}(N^3)$ complexity compared to the semi-explicit form.

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} [\mathbf{Q} - \mathbf{c}] (\dot{\mathbf{q}}, \mathbf{q}) \stackrel{\Delta}{=} \boldsymbol{\gamma}(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{Q}) \quad (3.18)$$

Another way to derive the explicit form of the equations of motion in a fully recursive way can be obtained through the Order-N algorithm [16]. This method is based on three recursive steps (instead of two for the NER scheme) on the multibody system. In short, first a forward recursion covers the MBS from the inertial body to the leaves to compute the bodies *kinematics* (position, orientation, linear and angular velocities). A second backward recursion computes the bodies *dynamics* on

the basis of the classical Newton-Euler equations within which a subtle embedded block factorization of the mass matrix is performed. This process prepares the third final *kinetic* recursion that straightforwardly computes the generalized acceleration $\ddot{\mathbf{q}}$ as a function $\mathbf{v}$ of the joint positions and velocities. This three-steps algorithm has a $\mathcal{O}(N)$ complexity. Several implementations of it have been proposed in the literature, the one used in this paper is the Schwertassek-Rulka method [16].

$$\ddot{\mathbf{q}} = \mathbf{v}(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{Q}) \quad (3.19)$$

As the system is expressed in a purely explicit form, the accelerations $\ddot{\mathbf{q}}$ do not *necessarily* need to be considered as an algebraic state. As a result, the dynamics can be assimilated to either set of ODE or DAE. The explicit dynamics is generally denoted ξ and can either represent the NER explicit dynamics γ or the Order-N dynamics $\mathbf{v}$. The dynamics is represented by a set of differential algebraic equations (DAE) as

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} & \mathbf{z} &= \ddot{\mathbf{q}} & \mathbf{u} &= \mathbf{Q}_a \\ \left\{ \begin{array}{l} \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z}) &= \begin{bmatrix} \mathbf{z} \\ \dot{\mathbf{q}} \end{bmatrix} \\ \mathbf{a}(\mathbf{x}, \mathbf{z}) &= \mathbf{z} - \xi(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{u}) = \mathbf{0} \end{array} \right. \end{aligned} \quad (3.20)$$

or by a set of ordinary differential equations (ODE) as

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} & \mathbf{u} &= \mathbf{Q}_a \\ \mathbf{f}(\mathbf{x}, \mathbf{u}) &= \begin{bmatrix} \xi(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{u}) \\ \dot{\mathbf{q}} \end{bmatrix} \end{aligned} \quad (3.21)$$

3.3.3 Symbolic Generation

In practice, the MBS equations of motion can be computed either numerically or symbolically. In specific cases, the recursive nature of the relative coordinates can be deeply exploited by a symbolic program dedicated to multibody systems. This is the purpose of the Robotran software [7] that offers symbolic tools for an extensive simplification of the equations of motion. The latter are written under the form of a function (C, Matlab, Python) with adequate input/output arguments. These functions, internally optimized thanks to the symbolic approach, can be used externally as a black-box by another program or computer environment.

Table 3.1 Formulations for the dynamics equations of *unconstrained* multibody systems

	Algorithm	Section	Dynamics	Complexity
IMPL	Original NER	3.3.2.1	$\Phi - \mathbf{Q} = \mathbf{0}$	N
SEMI	Modified NER	3.3.2.2	$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{c} - \mathbf{Q} = \mathbf{0}$	N^2
EXP1	Modified NER + Cholesky	3.3.2.3	$\ddot{\mathbf{q}} - \boldsymbol{\gamma} = \mathbf{0}$	$N^2 + N^3$
EXP2	Order-N	3.3.2.3	$\ddot{\mathbf{q}} - \mathbf{v} = \mathbf{0}$	N

In this paper, the function generated by Robotran will be used by the CasADi software [1] to express the system dynamics for the OCP formulation.

The different formulations of the dynamics yield different levels of complexity as summarized in Table 3.1.

3.4 Case Study: Unconstrained MBS

Given their different complexities, the formulations of the dynamics should lead to distinct performances for the OCP solving. To this extent, the Optimal Control of a simple unconstrained multibody system has been extensively tested.

3.4.1 System Presentation

The system of interest is a 3D robot arm with an inverted pendulum attached to the end effector as represented in Fig. 3.5. The 3D arm has a morphology that resembles the human arm. It is composed of 3 bodies and 4 actuated rotational joints (with relative coordinates q_1, q_2, q_3 and q_4). The first element is connected to the base with two rotational joints enabling to point in any direction of space. Then the second and third elements are connected in series via a single rotational joint. The relative motion of the pendulum with respects to the end effector is characterized by two Cardan-type angles (q_5 and q_6) which are considered as free joints (i.e., frictionless and non-actuated). The MBS is thus composed of 4 bodies and 6 joints. The system is underactuated as only the 4 joints of the arm are actuated. The mass, inertia and length of the different bodies are given in Table 3.2.

3.4.2 OCP Formulation

There are many possible ways to formulate the Optimal Control Problem for this application. In this study, the focus is on the trajectory optimization of the 3D arm. The task is to move the system from an initial pose and position its end effector at a given location in the 3D space after an imposed time T . At this final time, the system must be at equilibrium with the pendulum in a perfect upright position.

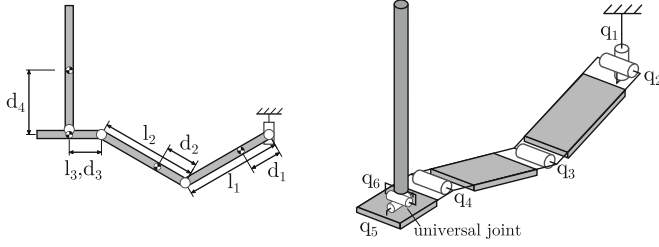


Fig. 3.5 Geometrical representation of the studied MBS system

Table 3.2 Geometric and inertia parameters for the 3D serial robot and the pendulum

	Length [m]	c.o.m. position [m]	Mass [kg]	Inertia [kg m ²]
Upperarm	$l_1 = 0.3$	$d_1 = 0.1$	2.5	[0.06, 0.1, 0.1]
Forearm	$l_2 = 0.3$	$d_2 = 0.1$	2	[0.03, 0.05, 0.05]
Hand	$l_3 = 0.1$	$d_3 = 0.1$	0.5	[0.004, 0.006, 0.006]
Pendulum		$d_4 = 0.2$	1	[0.1, 0.1, 0.05]

c.o.m. stands for the center of mass

The objective function is designed to minimize the actuation via the use of the square of the actuation torques and its time derivative:

$$L(\mathbf{x}, \mathbf{u}, \mathbf{z}) = k_1 \sum_{i=0}^4 Q_i^2 + k_2 \sum_{i=0}^4 \dot{Q}_i^2 \quad (3.22)$$

The first term aims at minimizing the torques in the four actuated joints while the second term minimizes the derivative of the controls which smooths the optimal trajectory and limits the mechanical power transmitted into the joints. In practice, the control torques are piece-wise constant on each discrete time interval, which implies the computation of the derivative via finite differences.

To achieve the equilibrium at the final time T , constraints are applied on the differential states $\mathbf{x} = [\dot{\mathbf{q}} \ \mathbf{q}]$, the inputs $\mathbf{u} = \mathbf{Q}_a$ and the algebraic states $\mathbf{z}$ when appropriate.

- The absolute position of the end effector can be expressed as a function of the angular joint positions, $\mathbf{s}_{ee}(\mathbf{q})$. Thus, the end effector can be constrained to a desired absolute Cartesian position $\mathbf{p}_d = [x_d \ y_d \ z_d]$:

$$\mathbf{p}_d - \mathbf{s}_{ee}(\mathbf{q}(T)) = \mathbf{0} \quad (3.23)$$

- The absolute velocity of the end effector being a linear function of the joint angular velocities, equating the angular velocities at the final time T to zero should then be sufficient.

$$\dot{\mathbf{q}}(T) = \mathbf{0} \quad (3.24)$$

- The equilibrium of the whole system is achieved by canceling the joint accelerations at the final time, $\ddot{\mathbf{q}}(T) = \mathbf{0}$. To this extent, the joint accelerations are either available as algebraic variables in the case of a DAE system or expressed with the function ξ in the case of a pure ODE system (see Sect. 3.3.2.3 on explicit formulations).

$$\mathbf{z}(T) = \mathbf{0} \quad \text{or} \quad \xi(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{Q}_a) = \mathbf{0} \quad (3.25)$$

At time $t = 0$, the robot is in a stable equilibrium. The initial condition of the system is given by the joint angles, $\mathbf{q}_i$.

$$\mathbf{q}(0) = \mathbf{q}_i = [0 ; \alpha ; -2\alpha ; \alpha ; 0 ; 0] \quad \dot{\mathbf{q}}(0) = \mathbf{0} \quad \ddot{\mathbf{q}}(0) = \mathbf{0} \quad (3.26)$$

In addition, path inequality constraints (3.27) are applied to the joint angles to keep a physical sense for the arm and the pendulum motions.

$$\begin{aligned} -\infty &\leq q_1 \leq \infty \\ -\pi/2 &\leq q_2 \leq \pi/2 \\ -5\pi/12 &\leq q_3 \leq 0 \\ -\pi/2 &\leq q_4 \leq \pi/2 \\ -\pi/4 &\leq q_5 \leq \pi/4 \\ -\pi/4 &\leq q_6 \leq \pi/4 \end{aligned} \quad (3.27)$$

The equilibrium condition (3.25), the terminal constraint (3.23) and the path constraints ensure all together that the pendulum is in an upright position at the final time.

3.4.3 Numerical Results

In order to assess the performances of the different formulations, we have compared the solutions obtained for the above 3D arm benchmark. For the comparison, a total of six formulations for the system dynamics have been considered. From Sect. 3.3, four different formulations for the equations of motion have been derived:

- IMPL: Implicit formulation from the Newton-Euler Recursive scheme (NER),
- SEMI: Semi-explicit formulation from the NER scheme,
- EXP1: Explicit formulation from the NER scheme,
- EXP2: Explicit formulation from the Order-N formalism.

While the first two formulations imply a DAE system for its numerical representation, the two explicit formulations can either be numerically treated as a system

of DAE or as a system of pure ODE. The difference lies in the number of decision variables and constraint equations for the numerical NLP but also in the treatment of the terminal constraints.

- EXPi-O: system of ODE with the position and velocity in the differential states,
- EXPi-A: system of DAE with the joint accelerations as algebraic variables.

The OCP described in Sect. 3.4.2 is formulated with the help of the software environment CasADi [1]. Based on the functions generated by Robotran, symbolic expressions of the dynamics are written and symbolically derived with respect to the states and input variables. Then, the numerical OCP is formulated using a direct collocation scheme [4] allowing the treatment of either a DAE or a pure ODE representation of the dynamics. The resulting NLP is then solved using the Interior Point solver IPOPT [20].

3.4.3.1 Simulation Settings

The performances of each formulation are evaluated through the trajectory optimization of the 3D arm robot. The task is described in Sect. 3.4.2. The time interval length T is 2 s and is divided into 100 time intervals for the numerical solving via a direct collocation method.

To compare the formulations, the optimization was performed for a series of end-effector terminal locations. This set of points is defined in cylindrical coordinates (φ, r, z) around a vertical axis passing through the robot arm origin. The angular position φ varies in the range $[-\pi/2, \pi/2]$ rad. On the other hand, the two other coordinates have relatively small variation ranges because the end effector has to remain in the robot working space: the radius is fixed to 0.63 m and the height varies in the range $[-0.1, 0.1]$ m (see Fig. 3.6).

Among the defined lateral surface, many points can be tested (see Fig. 3.6). The number of trajectory optimizations was chosen on the basis of a convergence analysis. The optimizations have been performed on a grid of different sizes and it has been shown that the mean number of iterations does not vary significantly between two refinements after a certain size (see Fig. 3.7). The chosen grid is composed of 21 angular orientations and 11 vertical positions for a total of 231 points in the 3D space (see Fig. 3.6).

In addition, two different sets of results are generated by tuning the k_1 and k_2 coefficients. The first cost function is defined by $(k_1 = 1e0, k_2 = 0)$ and corresponds to a more aggressive actuation while the second cost function is defined by $(k_1 = 1e0, k_2 = 1e5)$ and should result in a smoother actuation. In both cases, the initial condition (i.e., the α angle) is such that the end effector is positioned at 0.65 m from the origin with zero lateral rotation of the arm (i.e., $\varphi = 0$ rad).

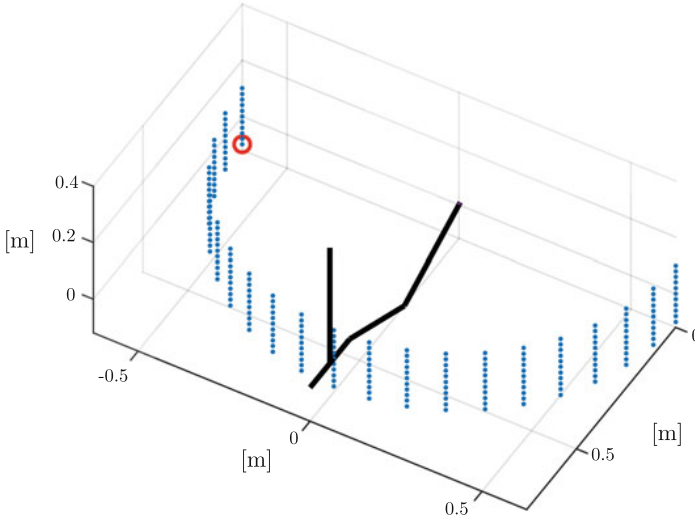


Fig. 3.6 Grid of the 3D point to reach. The multibody system is represented in its initial condition

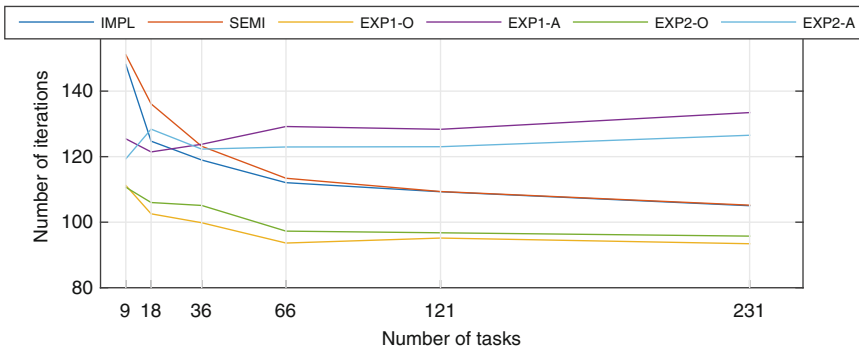


Fig. 3.7 Convergence analysis for the different formulations. After a given number of tasks, the number of iterations does not vary significantly

3.4.3.2 Simulation Results

Figure 3.8 shows the optimal trajectory obtained to reach the point $\mathbf{p}_d = \{0, 0.63, -0.1\}$ (see red point in Fig. 3.6) for the two different definitions of the cost function. The second definition leads to a smoother but more important actuation. The torques variations are reduced and there are less oscillations at the end of the trajectory. This smoother actuation is reflected in the joint velocities with lower values than for the non-smoothed actuation. Finally, for the two cost definitions, the same final angular positions and the same final actuation are obtained as the equilibrium to reach after 2 s is identical.

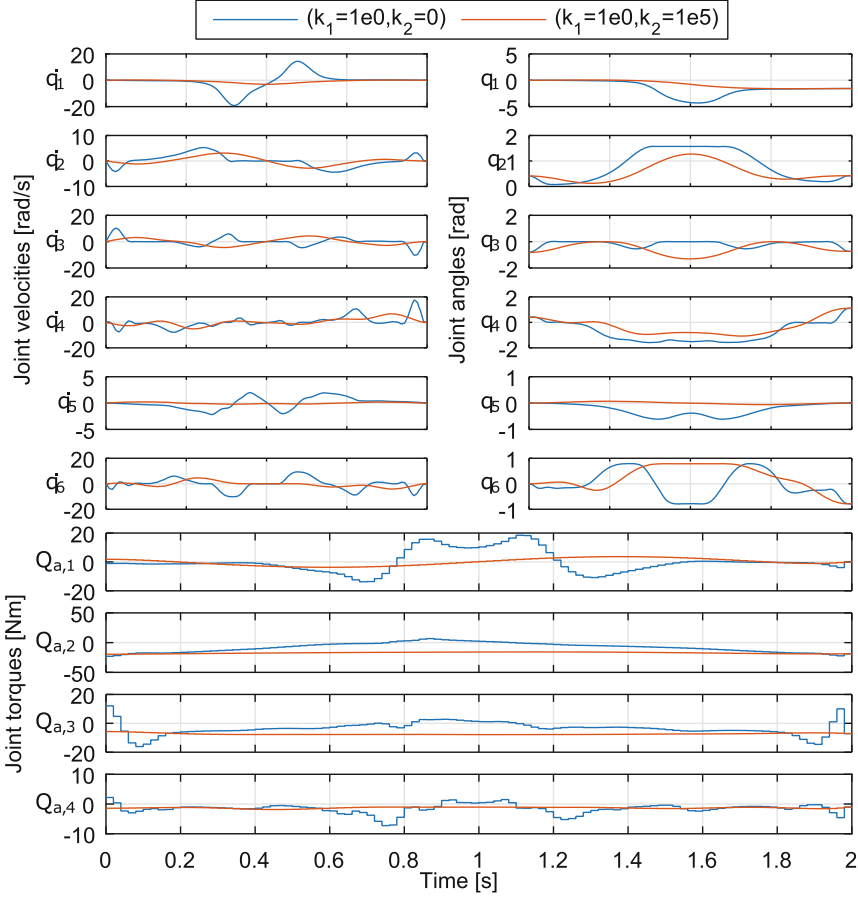


Fig. 3.8 Solution for the two different sets of cost function coefficients. A non-zero coefficient k_2 on the control derivative leads to a smoother but more important actuation

More than the solution itself, the interest of the study is to compare the different formulation performances to solve the OCP described in Sect. 3.4.2. To do so, the number of iterations before convergence, the cost function value at the optimum and the mean evaluation times of the dynamics are evaluated for the two sets of results.

First, the optimal values of the cost function are compared for the six considered representations of the dynamics. Figure 3.9 shows a box-plot¹ of the cost value for the two sets of results (i.e., different coefficients in the cost function). The OCP formulation with the smoothed actuation leads to more constant results with only

¹On the box plot, the central mark indicates the median, and the bottom and top edges of the box indicate the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted individually using the '+' symbol.

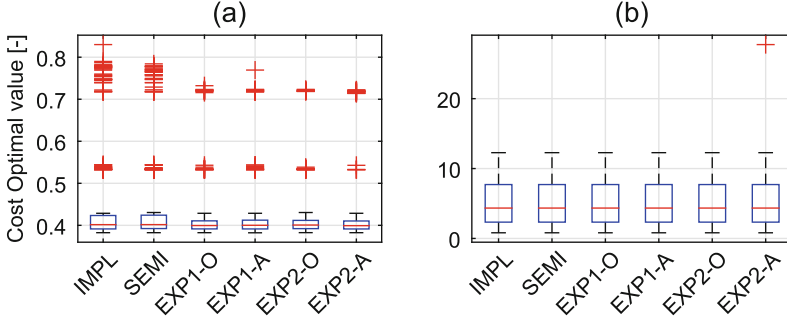


Fig. 3.9 Optimal values of the cost function represented in the form of a box plot for the two sets of solutions. **(a)** First cost function without smoothing: $k_1 = 1e0$, $k_2 = 0$. **(b)** Second cost function with smoothing: $k_1 = 1e0$, $k_2 = 1e5$

one formulation presenting an outlier which corresponds to a higher local minima (see Fig. 3.9b). On the other hand, in the case of the non-smoothed control, the solution leads to a higher number of outliers especially in the case of the implicit and the semi-explicit formulations (see Fig. 3.9a). Note that the higher group of outliers corresponds to the tasks where the point to reach is either at the far left or the far right and the lower group corresponds to the central point (i.e., no lateral movement needed for the arm). The higher outliers clearly correspond to higher local minima as lower minima are reached by the explicit formulations for the same tasks. Regarding the problem with no lateral movement, it is not clear whether it is a local minimum as all the formulations seem to struggle to solve this problem. In general, the actuation smoothing seems to reduce the problem of higher local minima.

Another important aspect of the performances is the computation efforts needed to reach the optimal solution. Figure 3.10 shows the number of iterations to convergence and the CPU time for the two sets of results. Figure 3.11 shows the CPU time per iteration. For the first set (see Fig. 3.10a), the number of iterations is characterized by a high number of outliers corresponding to the high local minima (see Fig. 3.9a). On this same set, the DAE explicit formulations result in a larger average number of iterations than the four other formulations. For the second set of tasks, the implicit and semi-explicit formulation generally perform better than the explicit ones (see Fig. 3.10b). This difference could be due to the high local minima that are encountered by the first two formulations while solving the first set of tasks. Regarding the CPU time, the implicit and the semi-explicit formulations lead to better results than the explicit ones for both sets of tasks. This tendency is also observed for the CPU time per iteration (see Fig. 3.11).

Regarding the numerical formulations, the ODE representation gives better results in terms of iterations for the first set of tasks (see Fig. 3.10a) but this difference is not so clear for the second set of tasks (see Fig. 3.10b). The two explicit

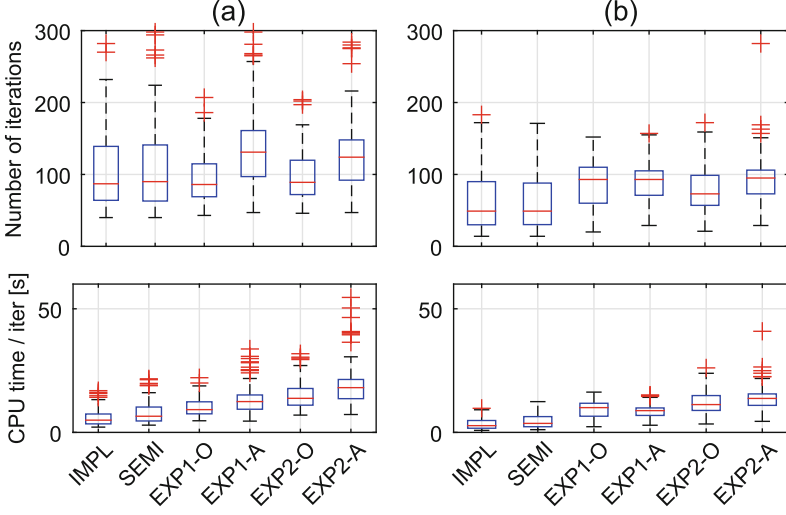


Fig. 3.10 Number of iterations and total CPU time to convergence represented in the form of a box plot for the two sets of solutions. (a) First cost function without smoothing: $k_1 = 1e0$, $k_2 = 0$. (b) Second cost function with smoothing: $k_1 = 1e0$, $k_2 = 1e5$

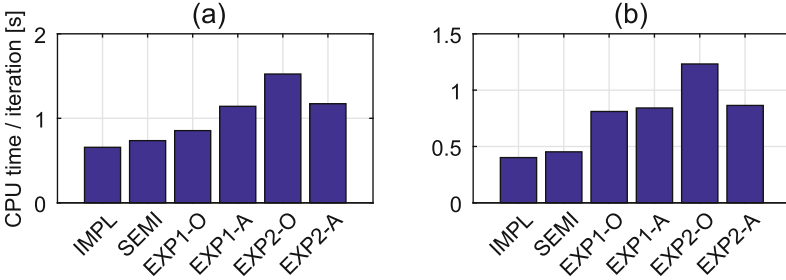


Fig. 3.11 Mean CPU time per iteration for the two sets of solutions. (a) First cost function without smoothing: $k_1 = 1e0$, $k_2 = 0$. (b) Second cost function with smoothing: $k_1 = 1e0$, $k_2 = 1e5$

forms exhibit similar performances in terms of CPU time when formulated as ODE and DAE systems (see Fig. 3.10). However, the Order-N implementation yields a lower CPU time per iteration when formulated as a DAE system (see Fig. 3.11). Its poor performances in terms of total CPU time are explained by its high number of iterations to reach convergence.

3.4.3.3 Discussion

In this section, we aim at giving some insights about the obtained results. From the simulations, it is clear that the implicit and the semi-explicit formulations perform better than the explicit ones. From a numerical point of view, the DAE form leads

Table 3.3 Size of the NLP associated to the OCP for the second cost definition (smoothed actuation)

Number of	IMP	SEMI	EXP1-O	EXP1-A	EXP2-O	EXP2-A
Variables	8812	8812	6412	8812	6412	8812
Constraint equations	8427	8427	6027	8427	6027	8427
nzt ^a in constraint Jacobian	72,435	72,435	62,519	69,635	62,519	69,635
nzt in Lagrangian Hessian	39,021	39,021	35,286	35,021	35,286	35,021

^anzt stands for non-zero terms

Table 3.4 Dynamic formulations and their generation for the studied MBS

	Algorithm	Dynamic terms	# Flops	# Eqs
IMPL	Original NER	$\Phi(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})$	460	114
SEMI	Modified NER	$\mathbf{M}(\mathbf{q})$ and $\mathbf{c}(\dot{\mathbf{q}}, \mathbf{q})$	830	264
EXP1	Modified NER + Cholesky	$\gamma(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{Q})$	991	328
EXP2	Order-N	$\nu(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{Q})$	1784	395

to a larger problem as both the number of equations and variables is higher (see Table 3.3). This difference results from the treatment of joint accelerations as algebraic variables which is mandatory for the implicit and semi-explicit dynamics. Nevertheless, compared to a pure ODE representation, the differential equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{z})$ is trivial in the DAE case. In addition, the DAE formulation allows the definition of the terminal equilibrium constraint directly in terms of the algebraic variables, which can be advantageous as the expression of the equilibrium constraints is linear in that case.

The explanation for the performance differences lies in the derivation of the dynamics itself. The different formulations are all written in the form of a function generated symbolically with Robotran. Because they inherit different complexity, the number of equations and floating point operations associated with each formulation differ as shown in Table 3.4. The $\mathcal{O}(N)$ computation of the Φ term does in fact lead to a lower number of operations, about half of the $\mathbf{M}$ and $\mathbf{c}$ terms of the semi-explicit dynamics for the system at hand. Regarding the explicit formulations, the NER modified algorithm with Cholesky decomposition leads to a number of flops comparable with the semi-explicit form. This is because the considered MBS is not too large and the Cholesky decomposition is not critical for the number of joints (i.e., 6 for the benchmark). However, the Order-N method yields a high number of operations, that is more than 4 times the number of operations for the implicit form. It may seem strange as the number of operations for the Order-N method is supposed to increase linearly with the size of the system. The problem is that this linear behavior becomes only interesting for a greater number of joints in the MBS (i.e., around 15 according to [8]).

While the explicit expressions clearly differ as they are derived differently, the implicit and the semi-explicit equations are exactly the same but expressed in different forms (see Sects. 3.3.2.1 and 3.3.2.2). As these equations are identical, their derivative should also be the same. As a matter of fact, the implicit and semi-explicit

formulations lead to the exact same number of non-zero terms in the expression of the Jacobian and the Hessian for the NLP, which is already a good indicator (see Table 3.3). The fact that the equations are identical should lead to the same iteration path while solving the numerical OCP. This is observed in the case of the smoothed actuation (see Fig. 3.10b) with an almost exact same number of iterations in the 231 tasks. For the case of non-smoothed actuation (see Fig. 3.10a), a similar but non-identical number of iterations is observed for the two formulations. These differences are attributed to the high numerical sensitivity of this Optimal Control Problem. This could cause the NLP solver to use different strategies for the two dynamic formulations. Finally, the implicit and semi-explicit formulations are more frequently trapped in higher local minima than the explicit ones. Some additional investigations would be necessary to explain this observation.

We are aware that treating the explicit dynamics with a collocation scheme may not be the best choice and that a multiple shooting could be more appropriated [6]. As a matter of fact, multiple shooting for the explicit formulations have also been investigated in this work but was abandoned as it led to lower performances in terms of robustness (i.e., lower success rate for the extreme point and slower convergence). The reason could be the high dynamics of the system and the strong terminal equilibrium constraint. In order to verify this assumption, a simpler multibody system could be studied with a smoother formulation of the OCP (i.e., with no terminal constraint).

3.5 Conclusion

This paper analyzes different formulations of the multibody dynamics and compare their performances in terms of convergence, accuracy and computation cost in the context of Optimal Control. The aim is to provide some insights on existing MBS formulations and their relevance applicability in the formulation and solution of Optimal Control Problems. Among the different modeling techniques, we focus on the derivation of the equations, using a relative coordinates approach, with the Newton-Euler Recursive scheme and the Order-N algorithm.

First, a framework for the formulation of Optimal Control Problem has been introduced. It was based on a semi-explicit DAE representation of the dynamics well suited for the treatment of multibody systems. Other formulations could have been considered (e.g., full implicit) but would have required specific integrators (i.e., DAE integrators). Regarding the discretization of the OCP, the direct collocation scheme is selected here. This decision is taken following the considerations found in [6], but also because multiple shooting methods are less robust. Then, the different formulations for the multibody dynamics are introduced and their derivations are discussed. From the Newton-Euler recursive scheme, three formulations are derived: the implicit, the semi-explicit and an explicit one. Then, a second explicit formulation is derived from the Order-N algorithm.

Finally, a case study is presented in a benchmark approach for the unconstrained MBS Optimal Control. The system consists in a 3D robot arm with an inverted pendulum attached to the end effector. An OCP is formulated with the goal to reach a point in the 3D space for the end effector after a preset time of 2 s. At the end, the system has to be at equilibrium with zero acceleration and the actuation is minimized on the overall time interval. This task is realized for a grid of target points in the 3D space and for two different definitions of the cost function resulting in two sets of tasks.

Results showed that the implicit and semi-explicit formulations are generally performing better in terms of CPU time and number of iterations. This can be explained by a more compact formulation of the dynamics which leads to a lower CPU cost per iteration. However, these two formulations appear to be more sensitive to higher local minima when the dynamics is stronger (i.e., in the case of non-smoothed actuation). This could be a drawback in applications such as offline trajectory planning where the quality of the solution is more important than its computation cost. On the other hand, the explicit formulations show larger CPU time to convergence but ensure a higher quality of the solution, as a lesser number of high local minima are observed. It seems that the Order-N explicit form works better with a DAE representation, while a pure ODE seems more appropriate for the NER explicit form. For the first set of solutions, with non-smoothed actuation, the differences in the number of iterations for the implicit and semi-explicit formulations are attributed to the strong numerical sensitivity of the problem. More investigation should be carried out on that subject. Nevertheless, this study shows the importance of the dynamic formulations on the OCP solving performances for unconstrained MBS.

There are many perspectives to extend this work. First, it would be appropriate to carry out additional simulations with different OCP formulations and different multibody systems in order to verify the observed tendencies. It has been shown that the way the dynamics is derived has a huge impact on the solver for the OCP as it strongly influences the CPU time per iteration. A recent release of Robotran offers symbolic tools for obtaining a fully analytical sensitivity of MBS equations based on the recursive differentiation of the multibody formulation. It will be interesting to see whether this could lead to better performances than when using the algorithmic differentiation provided by an optimization software like CasADi.

Acknowledgements We would like to thank Dr. Joris Gillis from KUL for his help in the formulation and solving of the Optimal Control Problems with CasADi. Quentin Docquier is a FRiA Grant Holder of the Fonds de la Recherche Scientifique – FNRS, Belgium.

References

1. Andersson, J., Gillis, J., Horn, G., Rawings, J.B., Diehl, M.: CasADi – software framework for nonlinear optimization and optimal control. In: *Mathematical Programming Computation* (2018)
2. Bastos, G., Seifried, R., Bruls, O.: Inverse dynamics of serial and parallel underactuated multibody system using a DAE optimal control approach. *Multibody Sys. Dyn.* **30**, 359–376 (2013)

3. Baumgarte, J.: Stabilization of constraints and integrals of motion. *Comput. Methods Appl. Mech. Eng.* **1**, 1–16 (1972)
4. Betts, J.T.: *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd edn. SIAM, Philadelphia (2010)
5. Björkenstam, S., Carlson, J.S., Lennartson, B.: Exploiting sparsity in the discrete mechanics and optimal control method with application to human motion planning. In: *Proceedings of 11th IEEE International Conference on Automation Science and Engineering*, Gothenburg, pp. 769–774 (2015)
6. Diehl, M., Bock, H., Diedam, H., Wieber, P.B.: Fast direct multiple shooting algorithms for optimal robot control. In: Diehl M., Mombaur, K. (eds.) *Fast Motions in Biomechanics and Robotics. Lecture Notes in Control and Information Sciences*, vol. 340. Springer, Berlin/New York (2006)
7. Docquier, N., Poncelet, A., Fisette, P.: Robotran: a powerful symbolic generator of multibody models. *Mech. Sci.* **4**, 199–219 (2013)
8. Fisette, P.: *Génération Symbolique des Equations du Mouvement de Systèmes Multicorps et Application dans le Domaine Ferroviaire*. Ph.D. thesis, Université catholique de Louvain, Louvain-la-Neuve, Belgium (1994)
9. Fisette, P., Samin, J.C.: Symbolic generation of large multibody system dynamic equations using a new semi-explicit Newton/Euler recursive scheme. *Arch. Appl. Mech.* **66**, 187–199 (1996)
10. Griewank, A., Walther, A.: *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. Other Titles in Applied Mathematics*, vol. 105. SIAM, Philadelphia (2008)
11. Gros, S., Zanon, M., Diehl, M.: Nonlinear MPC and MHE for mechanical multi-body systems with application to fast tethered airplanes. In: *IFAC Conference on Nonlinear Model Predictive Control*, Noordwijkerhout, pp. 86–93 (2012)
12. Iwamura, M., Eberhard, P., Schiehlen W., Seifried R.: A general purpose algorithm for optimal trajectory planning of closed loop multibody systems. In: Arczewski, K. (ed.) *Multibody Dynamics: Computational Methods and Applications. Computational Methods in Applied Sciences*, vol. 23. Springer, Dordrecht (2011)
13. Luh, J.-Y.-S., Walker, N.-W., Paul, R.-P.-C.: On-line computational scheme for mechanical manipulators. *J. Dyn. Syst. Meas. Control* **102**, 69–76 (1980)
14. Manara, S., Gabiccini, M., Artoni, A., Diehl, M.: On the integration of singularity-free representations of SO(3) for direct optimal control. *Nonlinear Dyn.* **90**, 1223–1241 (2017)
15. Samin, J.C., Fisette, P.: *Symbolic Modeling of Multibody Systems*. Kluwer, Dordrecht (2003)
16. Schwertassek, R., Rulka, W.: Aspects of efficient and reliable multibody systems simulation. In: Haug, E.J., Deyo, R.C. (eds.) *Real-Time Integration Methods for Mechanical System Simulation. NATO ASI Series F*, vol. 69, pp. 55–96. Springer, Berlin/Heidelberg (1989)
17. Siebert, R., Betsch, P.: A Hamiltonian conserving indirect optimal control method for multibody dynamics. In: *Proceedings in Applied Mathematics and Mechanics*, vol. 12. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim (2012)
18. Verschueren, D., Demeulenaere, B., Swevers, J., De Schutter, J., Diehl M.: Time-energy optimal path tracking for robots: a numerically efficient optimization approach. In: *10th IEEE International Workshop on Advanced Motion Control*, Trento, pp. 727–732 (2008)
19. von Stryk, O., Schlemmer, M.: Optimal control of the industrial robot manutec R3. In: Bulirsch, R. (ed.) *Computational Optimal Control. ISNM International Series of Numerical Mathematics*, vol. 115. Birkhäuser, Basel/Boston (1994)
20. Wächter, A.: *An interior point algorithm for large-scale nonlinear optimization with applications in process engineering*, Ph.D. thesis, Carnegie Mellon University (2002)
21. Wehage, R.-A., Haug, E.-J.: Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. *J. Mech. Des.* **134**, 247–255 (1982)

Numerical optimal control

C

This appendix focuses on the numerical formulation of the continuous Optimal Control Problem (OCP) introduced in Chapter 5 (Section 5.1.1). The three different approaches to solve a continuous OCP are first introduced. Then, the direct methods are presented with a special focus on the multiple shooting and the direct collocation methods for DAE system. For more details on numerical OCP, the reader can refer to *Model Predictive Control: Theory, Computation, and Design* and its Chapter 8 [77].

C.1 The different approaches for OCP solving

There are three different approaches (Fig. C.1) to solve the continuous time OCP described in Chapter 5 (Section 5.1.1).

State-space approaches use the principle of optimality stating that each subarc of an optimal trajectory must be optimal. In the continuous-time case, this leads to the so-called Hamilton-Jacobi-Bellman (HJB) equation, a partial differential equation (PDE) in the state-space. There exist methods to numerically solve this PDE but they generally suffer from Bellmans *curse of dimensionality* and are restricted to small state dimensions [77].

The second approach called *indirect* methods uses the necessary conditions of optimality of the infinite problem to derive a boundary value problem (BVP) in ordinary differential equations (ODE). Then, the resulting BVP is solved numerically by shooting techniques or by collocation [77]. This approach is often referred as *first optimize, then discretize*, as the conditions of optimality are first written in continuous time for the given problem, and then discretized in one way or another to compute a numerical solution. The two major drawbacks of the indirect methods are: (1) the underlying differential equations are strongly nonlinear and unstable which makes their solving difficult (2) changes in the control structure, i.e., the sequence of intervals where different constraints are active, are difficult to handle (they usually require a completely new problem setup).

Eventually, the third approach, called *Direct methods*, transform the original continuous-time optimal control problem into a discrete-time optimal control problem resulting in a nonlinear programming problem (NLP) which is then solved numerically. Therefore, the approach is referred as *first discretize, then optimize* [77]. The main advantage of direct methods over indirect ones is that they can easily treat all sorts of constraints (e.g., inequality path constraints) thanks to well-developed NLP methods that can efficiently deal with active set changes during the optimization procedure. Direct methods are nowadays by far the most successfully used techniques for solving constrained OCP in real-life applications; this is the reason why they will be used in this work to solve continuous OCPs.

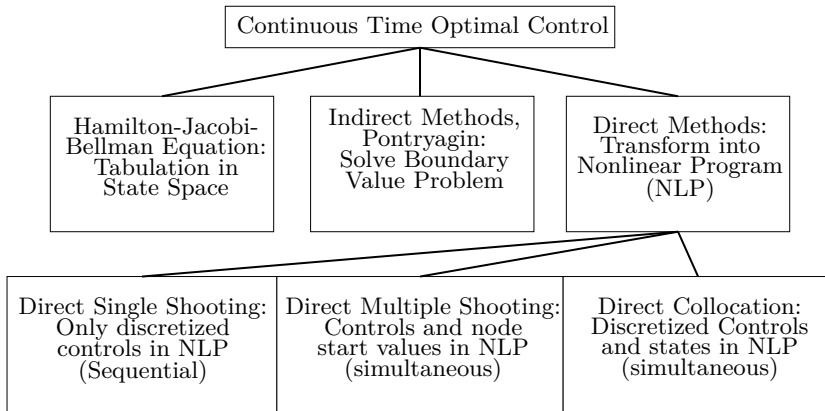


Figure C.1: There are three different approaches to solve the continuous time Optimal Control Problem (adapted from [78]).

C.2 Direct methods

C.2.1 Numerical OCP formulation

The *direct methods* consist in discretizing the time horizon $[0, T]$ into N time intervals $[t_k, t_{k+1}]$ such that $t_0 = 0$, $t_N = T$ and $k \in 0, 1, \dots, N - 1$. The time functions $x(t)$, $u(t)$ and $z(t)$ that were the unknowns of the system (5.1-5.5) are replaced by a sequence of $N + 1$ discrete states x_k , z_k and N piecewise constant controls u_k (Fig. C.2).

The discrete states and inputs form the set of decision variables, i.e., the unknowns whose values need to be determined in order to minimize the optimal criteria (i.e., the cost function). They are gathered in the vector w called the decision variable vector:

$$w = \{x_0, u_0, z_0, \dots, x_k, u_k, z_k, \dots, u_{N-1}, z_{N-1}, x_N\} \quad (\text{C.1})$$

The continuous cost function from equation (5.1) is replaced by a numerical approximation p which is a function of the decision variables vector w . The path constraints in equations (C.3-C.4) are also re-expressed in terms of w . This parameterization of the original continuous OCP is referred in the literature as a nonlinear program (NLP) [77].

$$\underset{w}{\text{minimize}} \quad p(w) \quad (\text{C.2})$$

subject to path constraints on the decision variables:

$$x_0 - x_i = 0 \quad (\text{C.3})$$

$$x_N - x_d = 0 \quad (\text{C.4})$$

$$c(w) \geq 0 \quad (\text{C.5})$$

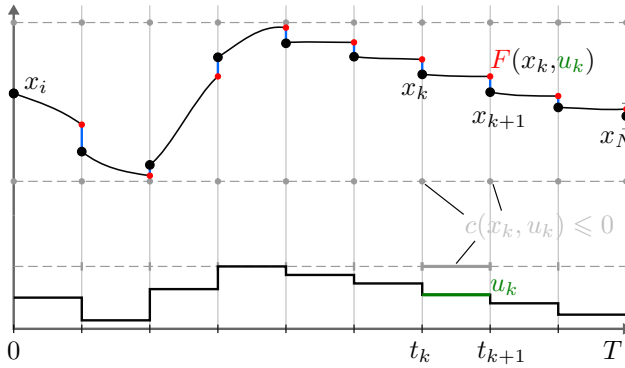


Figure C.2: Numerical OCP with the dynamic constraints (C.6) that are unsatisfied. The decision variables are the states at the discrete points and the piecewise constant control inputs.

In order to respect the system dynamics, the constant control input u_k on the time interval $[t_k, t_{k+1}]$ is linked to the two adjacent differential states x_k and x_{k+1} through the explicit numerical integration F of the continuous dynamics (C.6) on the same discrete time interval (Fig. C.2). These constraints are often referred as the *continuity* constraints.

$$x_{k+1} - F(x_k, u_k, z_k) = 0 \quad \forall k = 1, \dots, N-1 \quad (\text{C.6})$$

In addition, the differential and algebraic states are linked at each discrete time through the set of algebraic equations.

$$a(x_k, z_k) = 0 \quad \forall k = 1, \dots, N \quad (\text{C.7})$$

There are two well-known numerical methods to handle the dynamics *continuity* constraints (C.6-C.7): the multiple shooting and the direct collocation [79]. Both methods are based on a Newton scheme to solve the optimization problem.

C.2.2 Multiple shooting

The *multiple shooting* method separates the integration process from the optimization process. It consists in directly time-integrating the dynamics on the time interval $[t_k, t_{k+1}]$. In this approach, the integration method used to obtain function $F(x_k, u_k, z_k)$ is free and can be adapted to the system dynamics (e.g., a Runge-Kutta algorithm if the system is purely differential). In addition, the evolution of the algebraic variables from t_k to t_{k+1} has to be treated internally by the integrator. The formulation of the discrete OCP developed above already corresponds to a multiple shooting scheme with the continuity constraints expressed in terms of the intermediate variables x_k and z_k . Thus, the decision vector w in equation (C.1) is unchanged.

The overall NLP can be constructed as a *two-level Newton scheme* where the upper level solves the optimization problem (i.e., minimize the cost function) while the lower lever solves the equations (C.6-C.7) associated to the numerical integration.

It is important to stress that algebraic states $z(t)$ are entirely defined through the algebraic part of the DAE (i.e, $a(x, z, u) = 0$) meaning that any extra continuity condition on the algebraic states would result in a over-constrained problem. Furthermore, a discontinuous input parametrization (e.g, piecewise-constant) would logically yield discontinuous algebraic states on the discrete time grid.

C.2.3 Direct collocation

The *direct collocation* consists in imposing the dynamic equations at intermediate collocation points. The integration and the optimization are treated at the same level.

On each time interval $[t_k, t_{k+1}]$, a set of d collocation points $t_{k,i}$ is chosen, with $i = 0, \dots, d$. Each differential state on the time interval $[t_k, t_{k+1}]$ is approximated by a Lagrange polynomial, $p_k(t, \xi_k) \in R^n$ with coefficients $\xi_k \in R^{n_x(d+1)}$:

$$x(\xi_k, t) = \sum_{i=0}^d \xi_{k,i} \cdot p_{k,i}(t) \quad (\text{C.8})$$

Using Lagrange polynomial function, the value of the state x at each collocation time $t_{k,i}$ corresponds to the polynomial coefficients.

$$x(\xi_k, t_{k,i}) = \xi_{k,i} \quad (\text{C.9})$$

The collocation-based integration on a time interval $[t_k, t_{k+1}]$ starting from the initial value x_k consists in solving the following $d + 1$ collocation equations in terms of the $d + 1$ coefficients.

$$\begin{bmatrix} \xi_{k,0} - x_k \\ \sum_{i=0}^d \xi_{k,i} \dot{p}_{k,i}(t_{k,1}) - f(\xi_{k,i}, u_k) \\ \vdots \\ \sum_{i=0}^d \xi_{k,i} \dot{p}_{k,i}(t_{k,d}) - f(\xi_{k,d}, u_k) \end{bmatrix} = 0 \quad (\text{C.10})$$

When it comes to the algebraic part of the DAE, a similar approach can be used by replacing the algebraic state z using the same Lagrange polynomial functions p characterized by coefficients $\zeta_k \in R^{n_z(d)}$.

$$z(\zeta_k, t) = \sum_{i=1}^d \xi_{k,i} \cdot p_{k,i}(t) \quad (\text{C.11})$$

with

$$z(\zeta_k, t_{k,i}) = \zeta_{k,i} \quad (\text{C.12})$$

The algebraic variable integration differs by the number of coefficients for each state, which is one less than for the differential states (i.e., d instead of $d+1$). This is explained by the fact that no continuity condition needs to be enforced at the time interval boundary, meaning that there is only d constraint equations on the time interval.

$$\begin{bmatrix} a(\xi_{k,1}, \zeta_{k,1}, u_k) \\ \vdots \\ a(\xi_{k,d}, \zeta_{k,d}, u_k) \end{bmatrix} = 0 \quad (\text{C.13})$$

Eventually, these two sets of constraints defined on the time interval $[t_k, t_{k+1}]$ are added to the existing set of constraints (C.6-C.7) while the decision variable vector w has to be augmented with the intermediate coefficients ξ_k and ζ_k .

$$w = \{x_0, u_0, z_0, \xi_0, \zeta_0, \dots, x_k, u_k, z_k, \xi_k, \zeta_k, \dots\} \quad (\text{C.14})$$

$$x_{N-1}, u_{N-1}, z_{N-1}, \xi_{N-1}, \zeta_{N-1}, x_N \} \quad (\text{C.15})$$

with ξ_k and ζ_k being of size $n_x(d+1)$ and $n_z d$ respectively.

With the direct collocation method, the overall NLP can be constructed as a *one-level Newton scheme* where the integration is handled at the same level as the optimization problem. Generally, the integration scheme implies the definition of extra variables (i.e., the polynomial coefficients) and additional constraint equations leading to a very large but sparse NLP. In addition, the collocation methods show fast local convergence, handle unstable systems well and can easily cope with state and terminal constraints [80]. These reasons and the compared performances observed for both methods led us to consider the collocation method in this work.

Feedback linearization of multibody systems

D

In this appendix, the derivations of the nonlinear control input $u(x)$ in equation (D.1) are presented for the feedback linearization of multibody systems (Section 6.2.1). Different cases are considered regarding the output and its relative degree, ρ .

$$u(x) = \beta^{-1}(x)(v(x) - \alpha(x)) \quad (\text{D.1})$$

with

$$\alpha(x) = L_f^\rho h(x) \quad (\text{D.2})$$

$$\beta(x) = L_g L_f^{\rho-1} h \quad (\text{D.3})$$

D.1 Relative degree and Lie-derivatives

In order to properly define the relative degree of a nonlinear system as in equation (D.4), the notion of Lie-derivative must be used.

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases} \quad (\text{D.4})$$

The Lie-derivative of a function h with respect to a function f is defined as follows :

$$L_f h(x) = \frac{\partial h}{\partial x} f(x) \quad (\text{D.5})$$

The relative degree, ρ of a system (D.4) is the lowest integer for which, in a region $\mathbb{R}^n$:

$$L_g(L_f^{\rho-1} h) \neq 0 \quad (\text{D.6})$$

As mentioned in Chapter 6 (Section 6.2.1), the relative degree will either be 1 or 2 for multibody systems. In the following sections, the expression of $u(x)$ is developed for different outputs considering a multibody system.

D.2 Multibody systems with relative degree 1

With a relative degree of 1, the chain of one integrator from input to output is direct and results in:

$$\xi = y \quad (\text{D.7})$$

$$\alpha(x) = L_f^1 h(x) = L_f h(x) \quad (\text{D.8})$$

$$\beta(x) = L_g L_f^{1-1} h(x) = L_g h(x) \quad (\text{D.9})$$

D.2.1 Nonlinear output

In that case, the expressions of $\alpha(x)$ and $\beta(x)$ are developed as follows:

$$\alpha(x) = L_f h = \frac{\partial h}{\partial x} f(x) \quad (\text{D.10})$$

$$\beta(x) = L_g h = \frac{\partial h}{\partial x} g(x) \quad (\text{D.11})$$

the partial derivatives of $h(x)$ will depend on the expression of $h(x)$ itself while the expressions of $f(x)$ and $g(x)$ are known for MBS and developed in Chapter 6 (Section 6.2.1).

D.2.2 Linear output ($\dot{\psi}$)

In the case of linear output y , the function $h(x)$ is simplified as follows:

$$h(x) = Cx \quad (\text{D.12})$$

Generally, the case $\rho = 1$ will only be encountered if the output y is a linear expression of the independent velocities, $\dot{q}_u$ solely : $C = [0 \ C'_{\dot{q}_u}]$. This simplifies the expressions $\alpha(x)$ and $\beta(x)$.

$$\alpha(x) = \frac{\partial h}{\partial x} f(x) = C f(x) = -C_{\dot{q}_u} M_r^{-1} F_{r,p} \quad (\text{D.13})$$

$$\beta(x) = \frac{\partial h}{\partial x} g(x) = C g(x) = C_{\dot{q}_u} M_r^{-1} N \quad (\text{D.14})$$

D.3 Multibody systems with relative degree 2

With a relative degree of 2, there are 2 integrators from input to output resulting in:

$$\xi = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \end{bmatrix} \quad (\text{D.15})$$

$$\alpha(x) = L_f^2 h(x) \quad (\text{D.16})$$

$$\beta(x) = L_g L_f^{2-1} h = L_g L_f^1 h(x) \quad (\text{D.17})$$

D.3.1 Nonlinear output ($a_{per,2}$)

In that case, the expressions of $\alpha(x)$ and $\beta(x)$ are developed as follows:

$$\alpha(x) = L_f^2 h(x) = \frac{\partial \left(\frac{\partial h}{\partial x} f \right)}{\partial x} f(x) \quad (\text{D.18})$$

$$\beta(x) = L_g L_f^1 h(x) = \frac{\partial \left(\frac{\partial h}{\partial x} f \right)}{\partial x} g(x) \quad (\text{D.19})$$

the partial derivatives of $h(x)$ will depend on the expression of $h(x)$ itself while the expressions of $f(x)$ and $g(x)$ are known for MBS and developed in Chapter 6 (Section 6.2.1).

D.3.2 Linear output (ϕ, δ)

In the case of linear output y the function $h(x)$ is simplified as follows:

$$h(x) = Cx \quad (\text{D.20})$$

This strongly simplifies the expressions $\alpha(x)$ and $\beta(x)$.

$$\alpha(x) = \frac{\partial \left(\frac{\partial h}{\partial x} f \right)}{\partial x} f(x) = C \frac{\partial f}{\partial x} f(x) \quad (\text{D.21})$$

$$\beta(x) = \frac{\partial \left(\frac{\partial h}{\partial x} f \right)}{\partial x} g(x) = C \frac{\partial f}{\partial x} g(x) \quad (\text{D.22})$$

Again, expressions of $f(x)$ and $g(x)$ are known for MBS. The partial derivative of $f(x)$ can be expressed as a function of the tangent matrices K_r , G_r and the

reduced mass matrix, M_r

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial \dot{q}_u}{\partial q_u} & \frac{\partial \dot{q}_u}{\partial q_u} \\ \frac{\partial (-M_r^{-1} F_{r,p})}{\partial q_u} & \frac{\partial (-M_r^{-1} F_{r,p})}{\partial \dot{q}_u} \end{bmatrix} = A = \begin{bmatrix} 0 & I \\ -M_r^{-1} K_r & -M_r^{-1} G_r \end{bmatrix} \quad (\text{D.23})$$

The two bottom terms are computed as follows:

$$\frac{\partial (-M_r^{-1} F_{r,p})}{\partial \dot{q}_u} = -M_r \frac{\partial (F_{r,p})}{\partial \dot{q}_u} = -M_r^{-1} G_r \quad (\text{D.24})$$

$$\frac{\partial (-M_r F_{r,p})}{\partial q_u} = - \left(\frac{\partial M_r^{-1}}{\partial q_u} F_{r,p} + M_r^{-1} \frac{\partial F_{r,p}}{\partial q_u} \right) \quad (\text{D.25})$$

$$= - \left(-M_r^{-1} \frac{\partial M_r}{\partial q_u} M_r^{-1} F_{r,p} + M_r^{-1} \frac{\partial F_{r,p}}{\partial q_u} \right) \quad (\text{D.26})$$

$$= - \left(M_r^{-1} \frac{\partial M_r}{\partial q_u} \ddot{q}_u + M_r^{-1} \frac{\partial F_{r,p}}{\partial q_u} \right) \quad (\text{D.27})$$

$$= -M_r^{-1} \left(\frac{\partial M_r}{\partial q_u} \ddot{q}_u + \frac{\partial F_{r,p}}{\partial q_u} \right) = -M_r^{-1} K_r \quad (\text{D.28})$$

D.4 MIMO systems

For the MIMO system, the different cases developed above can be combined. For each output, the relative degree is first determined. Then, the corresponding k entry of the column vector, Λ and the k line of the invertibility matrix, J are computed.

$$k = 1, \dots, n_y \quad \text{compute } \Lambda_k \text{ and } J_k. \quad (\text{D.29})$$

This way the vector Λ and matrix J are built incrementally. Then, the nonlinear input, $u(x)$ can be computed as follows with the synthetic input vector, v .

$$u(x) = J^{-1}(v - \Lambda) \quad (\text{D.30})$$

Bibliography

- [1] “FOTW #1040, July 30, 2018: Average Vehicle Occupancy Remains Unchanged From 2009 to 2017 | Department of Energy.” <https://www.energy.gov/eere/vehicles/articles/fotw-1040-july-30-2018-average-vehicle-occupancy-remains-unchanged-2009-2017>.
- [2] A. Verlé, *Joint optimization of vehicle dynamics and mechanical design through a multi-body based strategy*. PhD Thesis, Université catholique de Louvain, École polytechnique de Louvain, Louvain-la-Neuve, Belgium, Sept. 2019.
- [3] “Land Glider.” <https://usa.nissannews.com/en-US/channels/us-united-states-nissan-models-concepts-land-glider#?>
- [4] “Renault Twizy.” <https://fr.renault.be/vehicules/vehicules-electriques/twizy.html>, 2018.
- [5] C. Manz, “Toyota i-Road tilting vehicle.” https://www.prestigeelectriccar.com/en/articles/735/Toyota_i-Road_tilting_vehicle.
- [6] “Carver One | conceptcarz.com.” <https://www.conceptcarz.com/vehicle/z11633/carver-one.aspx>, 2006.
- [7] P. Egan, “Lean machine: Logic and substance from the dreamer’s workshop,” *Road & Track*, vol. 34, pp. 80B–80D, 1983.
- [8] F. Bucchi, F. Frendo, D. Simic, O. Tanna, and M. Perterer, “Design of a novel tilting electric four-wheeler,” in *Proceedings of International Motorcycle Conference*, (Cologne), Oct. 2016.
- [9] F. Bucchi, F. Cerù, and F. Frendo, “Stability analysis of a novel four-wheeled motorcycle in straight running,” *Meccanica*, vol. 52, pp. 2603–2613, Sept. 2017.
- [10] “BMW ‘CLEVER’ concept.” <https://leftlanenews.com/2006/04/23/bmws-clever-concept-completed/>.

- [11] S.-G. So and D. Karnopp, "Active Dual Mode Tilt Control for Narrow Ground Vehicles," *Vehicle System Dynamics*, vol. 27, no. 1, pp. 19 – 36, 1997.
- [12] S. Kidane, A. Lee, R. Rajamani, P. Starr, and M. Donath, "A fundamental investigation of tilt control systems for narrow commuter vehicles," *Vehicle System Dynamics*, vol. 46, no. 4, pp. 295 – 322, 2008.
- [13] J. W. Robertson, *Active Control of Narrow Tilting Vehicle Dynamics*. PhD Thesis, University of Bath, 2014.
- [14] D. Karnopp, *Vehicle Stability*. CRC Press, Mar. 2004.
- [15] A. Verlé and P. Fiset, "Design of an active tilting three-wheeler: optimization of the dynamic performances via multibody simulations," in *The 24th International Symposium on Dynamics of Vehicles on Roads and Tracks*, 2015.
- [16] D. Karnopp and R. Hibbard, "Optimum roll angle behavior for tilting ground vehicles.," *ASME Dyn Syst Control Div Publ DSC*, ASME, New-York, NY (USA), vol. 44, pp. 29–37, 1992.
- [17] D. Karnopp and C. Fang, "A Simple Model of Steering-Controlled Banking Vehicles," *Transportation Systems, ASME DSC*, vol. 44, Nov. 1992.
- [18] R. Hibbard and D. Karnopp, "The Dynamics of Small, Relatively Tall and Narrow Tilting Ground Vehicles," *Advanced automotive technologies*, 1993.
- [19] R. Rajamani, J. Gohl, L. Alexander, and P. Starr, "Dynamics of narrow tilting vehicles," *Mathematical and Computer modelling of dynamical systems*, vol. 9, no. 2, pp. 209–231, 2003.
- [20] J. Gohl, R. Rajamani, P. Starr, and L. Alexander, "Development of a novel tilt-controlled narrow commuter vehicle," *Dep. of Mech. Eng. University of Minnesota, MI*, 2006.
- [21] D. Piyabongkarn, T. Keviczky, and R. Rajamani, "Active direct tilt control for stability enhancement of a narrow commuter vehicle," *International Journal of Automotive Technology*, vol. 5, pp. 77–78, Dec. 2004.
- [22] S. Kidane, R. Rajamani, L. Alexander, P. J. Starr, and M. Donath, "Development and Experimental Evaluation of a Tilt Stability Control System for Narrow Commuter Vehicles," *IEEE Transactions on Control Systems Technology*, vol. 18, pp. 1266–1279, Nov. 2010.
- [23] B. W. Drew, *Development of active tilt control for a three-wheeled vehicle*. PhD thesis, University of Bath (United Kingdom), 2006.

- [24] M. I. Barker, *Chassis design and dynamics of a tilting three-wheeled vehicle*. PhD Thesis, University of Bath, 2006.
- [25] J. J. Berote, *Dynamics and control of a tilting three wheeled vehicle*. PhD Thesis, University of Bath, 2010.
- [26] J. Edelmann and M. Plöchl, “Electronic Stability Control of a Narrow Tilting Vehicle,” *SAE International Journal of Materials and Manufacturing*, vol. 4, pp. 1006–1013, Apr. 2011.
- [27] J. Edelmann, M. Plöchl, and P. Lugner, “Modelling and analysis of the dynamics of a tilting three-wheeled vehicle,” *Multibody System Dynamics*, vol. 26, pp. 469–487, Dec. 2011.
- [28] “Smera de Lumeneo: voiture électrique 2 places.” <http://www.moteur-ecologique.com/smera-lumeneo-electrique.html>.
- [29] L. Mourad, *Active lateral acceleration control of a narrow tilting vehicle*. Theses, Ecole des Mines de Nantes, Dec. 2012.
- [30] F. Claveau, P. Chevrel, and L. Mourad, “Non-linear control of a Narrow Tilting Vehicle,” in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2488–2494, Oct. 2014.
- [31] “EQUOS RESEARCH Co.,Ltd.” <http://www.equos.co.jp/en/>.
- [32] J. T. C. Tan, H. Arakawa, and Y. Suda, “Steering Dynamics of Tilting Narrow Track Vehicle with Passive Front Wheel Design,” *Journal of Physics: Conference Series*, vol. 744, Sept. 2016.
- [33] J. T. C. Tan, H. Arakawa, Y. Suda, K. Araki, A. Mizuno, and M. Horiguchi, “Steering stability of a passive front wheel design on tilting narrow track vehicle,” in *Proceedings of the 25th International Symposium on Dynamics of Vehicles on Roads and Tracks (IAVSD 2017)*, vol. 1, (Rockhampton, Queensland, Australia), Aug. 2017.
- [34] “Researchers develop wheel units to cut vehicle costs.” <https://uwaterloo.ca/engineering/news/researchers-develop-wheel-units-cut-vehicle-costs>, Mar. 2019.
- [35] A. Goodarzi, A. Soltani, M. H. Shojaeefard, and A. Khajepour, “An integrated vehicle dynamic control strategy for three-wheeled vehicles,” *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, vol. 229, pp. 225–244, Sept. 2015.
- [36] C. Tang, M. Ataei, and A. Khajepour, “A Reconfigurable Integrated Control for Narrow Tilting Vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 234–244, Jan. 2019.
- [37] Y. Ren, T. Q. Dinh, J. Marco, D. Greenwood, and C. Hesar, “Non-linearity Compensation based Tilting Controller for Electric Narrow Tilting Vehicles,” in *2018 5th International Conference on Control*,

- Decision and Information Technologies (CoDIT)*, pp. 1085–1090, Apr. 2018.
- [38] Y. Ren, T. Q. Dinh, J. Marco, and D. Greenwood, “Torque vectoring-based drive: Assistance system for turning an electric narrow tilting vehicle,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, Jan. 2019.
 - [39] P. Lugner and J. Edelmann, “Basics of Vehicle Dynamics, Vehicle Models,” in *Vehicle Dynamics of Modern Passenger Cars* (P. Lugner, ed.), vol. 582, pp. 1–45, Cham: Springer International Publishing, 2019.
 - [40] H. Pacejka, *Tyre and Vehicle Dynamics*. Automotive engineering, Butterworth-Heinemann, 2006.
 - [41] R. Sharp, S. Evangelou, and D. Limebeer, “Advances in the Modelling of Motorcycle Dynamics,” *Multibody System Dynamics*, vol. 12, pp. 251–283, Oct. 2004.
 - [42] J. Berote, J. Darling, and A. Plummer, “Development of a tilt control method for a narrow-track three-wheeled vehicle,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 226, pp. 48–69, Jan. 2012.
 - [43] J. Meijaard, J. M. Papadopoulos, A. Ruina, and A. Schwab, “Linearized dynamics equations for the balance and steer of a bicycle: a benchmark and review,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, pp. 1955–1982, Aug. 2007.
 - [44] R. S. Sharp, “The Stability and Control of Motorcycles,” *Journal of Mechanical Engineering Science*, vol. 13, pp. 316–329, Oct. 1971.
 - [45] S. GmbH, “Multi-Body Simulation . SIMPACK MBS Software.” <http://www.simpack.com/>.
 - [46] “Resolve Project – Resolve Project.” <https://www.resolve-project.eu/>.
 - [47] S. Maakaroun, W. Khalil, M. Gautier, and P. Chevrel, “Modeling and Simulating a Narrow Tilting Car Using Robotics Formalism,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 1026–1038, June 2014.
 - [48] P. James, “Tilting Vehicle developments.” <http://tiltingvehicle.blogspot.com>.
 - [49] R. Hibbard and D. Karnopp, “Twenty First Century Transportation System Solutions - a New Type of Small, Relatively Tall and Narrow Active Tilting Commuter Vehicle,” *Vehicle System Dynamics*, 1996.
 - [50] J. Gohl, R. Rajamani, L. Alexander, and P. Starr, “Active Roll Mode Control Implementation on a Narrow Tilting Vehicle,” *Vehicle System Dynamics*, vol. 42, pp. 347–372, Dec. 2004.

- [51] Q. Docquier, A. Verlé, and P. Fisette, “Narrow Tilting Vehicle (NTV) Control through MBS Modelling: Comparison of Direct and Steering tilt Controls,” in *Bicycle and Motorcycle Dynamics 2016, Symposium on the Dynamics and Control of Single Track Vehicles*, (Milwaukee), 2016.
- [52] S.-G. So and D. Karnopp, “Switching strategies for narrow ground vehicles with dual mode automatic tilt control,” *International journal of vehicle design*, vol. 18, no. 5, pp. 518–532, 1997.
- [53] A. Snell, “An Active Roll-Moment Control Strategy for Narrow Tilting Commuter Vehicles,” *Vehicle System Dynamics*, 1998.
- [54] S. Kidane, R. Rajamani, L. Alexander, P. Starr, and M. Donath, “Experimental Investigation of a Narrow Leaning Vehicle Tilt Stability Control System,” in *2007 American Control Conference*, pp. 1612–1617, July 2007.
- [55] J. W. Robertson, J. Darling, and A. R. Plummer, “Path Following Performance of Narrow Tilting Vehicles Equipped With Active Steering,” in *Volume 1: Advanced Computational Mechanics; Advanced Simulation-Based Engineering Sciences; Virtual and Augmented Reality; Applied Solid Mechanics and Material Processing; Dynamical Systems and Control*, (Nantes, France), p. 679, ASME, July 2012.
- [56] L. Mourad, F. Claveau, and P. Chevrel, “A Lateral Control Strategy for Narrow Tilting Commuter Vehicle Based on the Perceived Lateral Acceleration,” *IFAC Proceedings Volumes*, vol. 44, pp. 6254–6259, Jan. 2011.
- [57] L. Mourad, F. Claveau, and P. Chevrel, “Direct and Steering Tilt Robust Control of Narrow Vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 1206–1215, June 2014.
- [58] H. Furuichi, J. Huang, T. Matsuno, and T. Fukuda, “Dynamic model of three wheeled Narrow Tilting Vehicle and optimal tilt controller design,” in *2012 International Symposium on Micro-NanoMechatronics and Human Science (MHS)*, pp. 331–336, Nov. 2012.
- [59] J. C. Chiou, C. Lin, C. Chen, and C. Chien, “Tilting motion control in narrow tilting vehicle using double-loop PID controller,” in *2009 7th Asian Control Conference*, pp. 913–918, Aug. 2009.
- [60] M. Chatterjee, M. Kale, and B. Chaudhari, “A dynamic stability control for electric narrow tilting three wheeled vehicle using integrated multivariable controller,” *Transportation research part D: transport and environment*, 2017.
- [61] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

- [62] H. Khalil, *Nonlinear Systems*. Pearson Education, Prentice Hall, 2002.
- [63] N. Roqueiro, M. G. De Faria, and E. F. Colet, “Sliding Mode Controller and Flatness Based Set-Point Generator for a Three Wheeled Narrow Vehicle,” *IFAC Proceedings Volumes*, vol. 44, pp. 11925–11930, Jan. 2011.
- [64] L. Y. Morita, N. Roqueiro, and H. B. Silveira, “Steering and velocity tracking of a tilting three-wheeled vehicle using sliding mode control,” in *2016 14th International Workshop on Variable Structure Systems (VSS)*, pp. 322–327, June 2016.
- [65] W. Schiehlen, ed., *Multibody Systems Handbook*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1990.
- [66] N. Docquier, *Multiphysics modelling of multibody systems—application to railway pneumatic suspensions*. PhD Thesis, Universite Catholique de Louvain, Belgium, 2010.
- [67] J.-C. Samin and P. Fiset, *Symbolic Modeling of Multibody Systems*. Solid Mechanics and Its Applications, Springer Netherlands, 2003.
- [68] R. A. Wehage and E. J. Haug, “Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic Systems,” *Journal of Mechanical Design*, vol. 104, pp. 247–255, Jan. 1982.
- [69] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, “On-Line Computational Scheme for Mechanical Manipulators,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 102, pp. 69–76, June 1980.
- [70] P. Fiset and J. C. Samin, “Symbolic generation of large multibody system dynamic equations using a new semi-explicit Newton/Euler recursive scheme,” *Archive of Applied Mechanics*, vol. 66, pp. 187–199, Jan. 1996.
- [71] R. Schwertassek and W. Rulka, “Aspects of Efficient and Reliable Multibody System Simulation,” in *Real-Time Integration Methods for Mechanical System Simulation* (E. J. Haug and R. C. Deyo, eds.), pp. 55–96, Berlin, Heidelberg: Springer Berlin Heidelberg, 1990.
- [72] Q. Docquier, O. Bröls, and P. Fiset, “Comparison and Analysis of Multibody Dynamics Formalisms for Solving Optimal Control Problem,” in *IUTAM Symposium on Intelligent Multibody Systems – Dynamics, Control, Simulation* (E. Zahariev and J. Cuadrado, eds.), vol. 33, pp. 55–77, Cham: Springer International Publishing, 2019.
- [73] N. Docquier, A. Poncelet, and P. Fiset, “ROBOTRAN: a powerful symbolic generator of multibody models,” *Mechanical Sciences*, vol. 4, pp. 199–219, May 2013.

- [74] S. Kidane, L. Alexander, R. Rajamani, P. Starr, and M. Donath, “Road bank angle considerations in modeling and tilt stability controller design for narrow commuter vehicles,” in *2006 American Control Conference*, June 2006.
- [75] “piaggio.” https://www.piaggio.com/us_EN/.
- [76] D. Toffin, G. Reymond, A. Kemeny, and J. Droulez, “Role of steering wheel feedback on driver performance: driving simulator and modeling analysis,” *Vehicle System Dynamics*, vol. 45, pp. 375–388, Apr. 2007.
- [77] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*, vol. 2. Nob Hill Publishing Madison, WI, 2017.
- [78] M. Diehl and S. Gros, *Numerical Optimal Control*. 2019.
- [79] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Society for Industrial and Applied Mathematics, second ed., Jan. 2010.
- [80] M. Diehl, H. Bock, H. Diedam, and P.-B. Wieber, “Fast Direct Multiple Shooting Algorithms for Optimal Robot Control,” in *Fast Motions in Biomechanics and Robotics* (M. Diehl and K. Mombaur, eds.), vol. 340, pp. 65–93, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.
- [81] Q. Docquier, “Analysis of MBS dynamics formulations for Solving Optimal Control problem,” Sept. 2017.
- [82] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi – A software framework for nonlinear optimization and optimal control,” *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.
- [83] “MECO - Software / rokit.” <https://gitlab.mech.kuleuven.be/meco-software/rokit>.
- [84] A. Griewank and A. Walther, *Evaluating derivatives: principles and techniques of algorithmic differentiation*, vol. 105. Siam, 2008.
- [85] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [86] “ISO 3888-2:2011, Passenger cars – Test track for a severe lane-change manoeuvre – Part 2: Obstacle avoidance,” 2011.
- [87] S. D. Keen and D. J. Cole, “Application of time-variant predictive control to modelling driver steering skill,” *Vehicle System Dynamics*, vol. 49, no. 4, pp. 527–559, 2011.

- [88] A. J. Pick and D. J. Cole, “A mathematical model of driver steering control including neuromuscular dynamics,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 130, no. 3, p. 031004, 2008.
- [89] C. C. Macadam, “Understanding and Modeling the Human Driver,” *Vehicle System Dynamics*, vol. 40, no. 1-3, pp. 101–134, 2003.
- [90] K. J. Aström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, Apr. 2010.
- [91] D. S. Ebert, F. K. Musgrave, D. Peachey, K. Perlin, and S. Worley, *Texturing & Modeling: A Procedural Approach*. Morgan Kaufmann, 2003.
- [92] J. K. Hedrick and A. Girard, “Control of nonlinear dynamic systems: Theory and applications,” *Controllability and observability of Nonlinear Systems*, p. 48, 2005.
- [93] A. Bemporad and C. Rocchi, “Decentralized linear time-varying model predictive control of a formation of unmanned aerial vehicles,” in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 7488–7493, Dec. 2011.
- [94] T. Takahama and D. Akasaka, “Model Predictive Control Approach to Design Practical Adaptive Cruise Control for Traffic Jam,” *International Journal of Automotive Engineering*, vol. 9, no. 3, pp. 99–104, 2018.
- [95] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat, “Linear time-varying model predictive control and its application to active steering systems: Stability analysis and experimental validation,” *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 18, no. 8, pp. 862–875, 2008.
- [96] A. Bemporad, N. L. Ricker, and M. Morari, “Model Predictive Control Toolbox™ User’s Guide,” Sept. 2019.
- [97] A. Bemporad, “Model Predictive Control.” http://cse.lab.imtlucca.it/~bemporad/mpc_course.html.
- [98] C. R. van den Brink and H. M. Kroonen, “DVC™undefinedspace - The banking technology driving the CARVER vehicle class,” in *Proceeding of the 7th international symposium on advanced vehicle control*, (Arnhem, Netherlands), Aug. 2004.
- [99] “Carver earth.” <https://carver.earth/en/>.
- [100] J. P. Pauwelussen, “The dynamic behaviour of man-wide vehicles with an automatic active tilting mechanism,” in *1999 EAEC Congress: Vehicle systems technology for the next century, Barcelona 30 June-2 July 1999;(30 June: Conference 2’dynamics on light vehicles’)*, 1999.

- [101] J. P. Pauwelussen, “The dynamic performance of narrow actively tilting vehicles,” *AVEC 2000, 5th Int’l Symposium on Advanced Vehicle Control*, pp. 569–576, 2000.
- [102] S. Kidane, L. Alexander, R. Rajamani, P. Starr, and M. Donath, “Control system design for full range operation of a narrow commuter vehicle,” in *ASME 2005 international mechanical engineering congress and exposition*, pp. 473–482, American Society of Mechanical Engineers, 2005.
- [103] B. Drew, K. Edge, M. Barker, J. Darling, G. Owen, and H. Johannsen, “System development for hydraulic tilt actuation of a tilting narrow vehicle,” in *9th Scandinavian International Conference on Fluid Power, SICFP*, vol. 5, pp. 1–3, 2005.
- [104] “Simscape Multibody - MATLAB & Simulink.” <https://mathworks.com/products/simmechanics.html>.
- [105] J. W. Robertson, J. Darling, and A. R. Plummer, “Combined steering–direct tilt control for the enhancement of narrow tilting vehicle stability,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, Mar. 2014.
- [106] J. Chiou and C. Chen, “Modeling and Verification of a Diamond-Shape Narrow-Tilting Vehicle,” *IEEE/ASME Transactions on Mechatronics*, vol. 13, pp. 678–691, Dec. 2008.
- [107] H. Furuichi, J. Huang, T. Fukuda, and T. Matsuno, “Switching Dynamic Modeling and Driving Stability Analysis of Three-Wheeled Narrow Tilting Vehicle,” *IEEE/ASME Transactions on Mechatronics*, vol. 19, pp. 1309–1322, Aug. 2014.
- [108] A. Nguyen, P. Chevrel, and F. Claveau, “LPV Static Output Feedback for Constrained Direct Tilt Control of Narrow Tilting Vehicles,” *IEEE Transactions on Control Systems Technology*, pp. 1–10, 2019.
- [109] J. T. C. Tan, Y. Huang, Y. Suda, A. Mizuno, and M. Horiguchi, “Cornering stability improvement by gyro moment for narrow tilting vehicle,” *Journal of Mechanical Science and Technology*, vol. 29, pp. 2705–2711, July 2015.
- [110] “MapleSim - Advanced System-Level Modeling & Simulation - Maplesoft.” <https://www.maplesoft.com/products/maplesim/>.
- [111] S. Kim, J. McPhee, and N. Lashgarian Azad, “Improving Stability of a Narrow Track Personal Vehicle using an Active Tilting System,” in *SAE 2014 World Congress & Exhibition*, pp. 2014–01–0087, Apr. 2014.
- [112] M. Ataei, *Reconfigurable Integrated Control for Urban Vehicles with Different Types of Control Actuation*. PhD Thesis, University of Waterloo, Dec. 2017.

- [113] M. Chatterjee, M. Kale, and B. N. Chaudhari, "Mathematical modelling of chassis dynamics of electric narrow tilting three wheeled vehicle," in *2015 Annual IEEE India Conference (INDICON)*, pp. 1–6, Dec. 2015.
- [114] A. Sponziello, F. Frendo, and M. Guiggiani, "Stability Analysis of a Three-Wheeled Motorcycle," *SAE International Journal of Engines*, vol. 1, pp. 1396–1401, Sept. 2008.