GAME THEORETICAL APPROACH TO VEHICLE DYNAMIC CONTROL

Dissertation

Presented in Partial Fulfillment of the Requirements for the Degree

Doctor of Philosophy

in Graduate School of The Memorial University of Newfoundland

By

Payam Pooyafar, B.E., M.S.

Graduate Program in Mechanical Engineering

Department of Engineering and Applied Science

Memorial University of Newfoundland

St. Johns, NL

2022

Dissertation Committee:

Dr. Bruce Minaker

Dr. Geoff Rideout (Research Supervisor)

Dr. George Mann

ABSTRACT

There are multiple objectives that a controller designer pursues in vehicle dynamics, such as safety, tracking, comfort, and efficiency. These objectives are present in different dimensions of longitudinal, lateral, vertical, yaw, roll, and pitch for each vehicle and can be controlled or manipulated by all types of actuators. These actuators can affect the steering, braking, torques at each tire, suspension system, etc. The majority of studies in vehicle dynamics control address either one control objective or control action, and even the ones that address an integrated control structure (including more control objectives and actuators) neglect the counter effects of the controllers/actuators on each other' objectives. So there exists the question of what can be the most efficient strategy in an integrated structure that provides the optimal solution for any set of objectives and control actions by taking into account the effects of each actuator in a way that no actuator can achieve a better optimal result by deviating from that strategy. Game theory is a field that addresses these types of questions, and its application in vehicle dynamics is called the *theory of differential games*.

This dissertation explores differential game theory in two-player and three-player optimal games in vehicle dynamics. It presents a solution for an integrated optimal problem, where two/three actuators (players) are trying to achieve their own sets of optimal goals by taking into account other actuators' (players) actions. The control problems are categorized into two classes of Linear Quadratic Regulators (LQR) and Control Coupled Output Regulation (CCOR) due to coupling between the output objectives and control actions. The solution for the single-player game exists in the literature for both LQR and CCOR for infinite horizon linear continuous systems. This thesis extends this solution to the two-player and three-player games

for continuous systems. After presenting the solution, in theory, the control feedback gains are calculated for the following case studies using linear control models:

- The two-player game between active steering and corrective yaw moment
- The two-player game between active steering and corrective roll moment
- Three-player game between active steering, corrective yaw moment, and corrective roll moment
- The two-player player game between active suspension and corrective roll moment

The designed feedback gains are used in a closed-loop feedback system on a nonlinear vehicle model with both linear tire and nonlinear tire models. The simulation results for the two/three-player game theory approach are compared with the one player and the two/three payer decentralized approach (normal optimal approach without considering the counter effects of control actions on each others' cost function). It is shown that the game theory approach provides better performance in terms of control action cost and objective cost in some scenarios.

Keywords: linear quadratic regulator, control coupled output regulation, differential game theory, vehicle dynamics, nonlinear tire model

ACKNOWLEDGMENTS

My dissertation is dedicated to my family, who has always been supportive in my journey and my study overseas. I would like to express my sincere thanks to my advisor Dr. Geoff Rideout for his guidance and support throughout my graduate study. I would like to thank Dr. Bruce Minaker and Dr. George Mann for serving on my dissertation committee and their comments. I would also like to acknowledge the financial support and research expertise provided by Dr. Geoff Rideout, the Memorial University Graduate Studies, and the MUN Internationalization office during my graduate study. I want to thank my dear family member and internal organ specialist Dr. Behzad Mohammadkhah for all of his help and health services to control my stomach condition through these years. Last but not least, I am grateful to all of my friends at Memorial University and in the city of St. John's, NL. This dissertation would not have been completed without encouragement and assistance from these friends.

VITA

2009……………………………………B. Engineering Mechanical Engineering,

Sharif University of Technology, Tehran, Iran

2012…………………………………………………M.S. Mechanical Engineering,

Sharif University of Technology, Tehran, Iran

2014 to 2021……………………………..……………...Graduate Research Assistant,

Memorial University of Newfoundland

FIELDS OF STUDY

Major Field: Mechanical Engineering

Minor Field: Control & Vehicle Dynamics

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a_y lateral acceleration, m/s^2

b_{sl} left suspension damping, N.s/m

b_{sr} right suspension damping, N.s/m

b_{tl} right tire damping, N.s/m

b_{tl} left tire damping, N.s/m

C_f front cornering stiffness, N/rad

C_r rear cornering stiffness, N/rad

F_{al} left active suspension force, N

F_{ar} right active suspension force, N

F_{nl} left tire normal force, N

F_{nr} right tire normal force, N

F_{yf} front lateral tire force, N

F_{yr} rear lateral tire force, N

g acceleration due gravity, m/s^2

h_r height of roll axis over the ground, m

h_s nominal height of sprung mass CG over roll axis, m

I_x sprung mass roll moment of inertia, kg.m^2

I_z yaw moment of inertia, kg.m^2

K_r steady-state yaw rate gain, 1/s

k_{sl} left suspension stiffness, N/m

k_{sr} right suspension stiffness, N/m

k_{tl} left tire stiffness, N/m

k_{tr} right tire stiffness, N/m

l_f distance from the front axle to CG, m

l_r distance from the rear axle to CG, m

M total mass, kg

M_s sprung mass, kg

m_{ul} left unsprung mass, kg

m_{ur} right unsprung mass, kg

M_{yc} corrective yaw moment control input, N.m

$M_{\phi c}$ corrective roll moment control input, N.m

r vehicle yaw rate, rad/s

$\dot{r}$ vehicle yaw acceleration, rad/s^2

t track width, m

V_x longitudinal velocity, m/s

v_y lateral velocity, m/s

X_{des} vehicle desired x position, m

X_{G} vehicle global x position, m

Y_{des} vehicle desired y position, m

Y_G vehicle global y position, m

z_{rl} left wheel road input position (disturbance), m

z_{rr} right wheel road input position (disturbance), m

z_s sprung mass vertical position, m

$\dot{z}_s$ sprung mass vertical velocity, m/s

$\ddot{z}_s$ sprung mass vertical acceleration, m/s^2

z_{ul} left unsprung mass vertical position, m

$\dot{z}_{ul}$ left unsprung mass vertical velocity, m/s

$\dot{z}_{rl}$ left wheel road input velocity (disturbance), m/s

$\dot{z}_{rr}$ right wheel road input velocity (disturbance), m/s

z_{ur} right unsprung mass vertical position, m

$\dot{z}_{ur}$ right unsprung mass vertical velocity, m/s

α_f front slip angle, rad

α_r rear slip angle, rad

μ road adhesion coefficient

ψ vehicle yaw angle, rad

ψ_{des} vehicle desired yaw angle, rad

ϕ vehicle roll angle, rad

$\dot{\phi}$ vehicle roll rate, rad/s

$\ddot{\phi}$ vehicle roll acceleration, rad/s^2

δ_c (δ_{AS}) active steering control input, rad

δ_H human driver front steering input (disturbance), rad

1 CHAPTER 1: INTRODUCTION

1.1 Motivation

Safety, comfort, and efficiency have always been the most significant concerns in vehicle dynamics, especially in more sensitive applications like military and automated highways. The army deploys trucks in closely-spaced convoys where the terrain can be highly variable, and the drivers are often inexperienced. Traffic accidents are a significant cause of death for soldiers during peacetime[1]. Automated highways in which groups of vehicles travel very closely have been envisioned to improve traffic flow and safety[2]. Considering the points mentioned above, enhancing vehicles' performance towards maintaining safety and maneuverability has been the target of much research. There have been many different approaches taken to solve this problem. The previous studies focused on using a single control input like suspension forces, steering angle, braking force, and torques generated at the wheels.

Using a single input variable to control the vehicles' dynamic states limits the number of responses that can be controlled and makes it hard to provide high overall dynamic performance for the vehicle. For instance, using active suspension forces as the control input can provide ride comfort and vertical dynamic stability. However, it usually can't sufficiently improve the yaw dynamics of a vehicle to fulfill the objective of tracking the desired path. Some research studies integrated dynamic control that takes different controllers (multiple control inputs) into consideration and simultaneously achieves various objectives. For example: using active suspension forces and braking forces as two different groups of control inputs to maintain vertical vehicle stability (for ride comfort, road holding, anti-roll, and anti-pitch) and track the desired yaw rate[8]. Most of the literature related to integrated control has taken a decentralized control approach to solve the problem, which means the control problem with multiple control inputs is broken into two small problems, each with single control input, and each problem is solved in an independent way to satisfy the related control objective, and the interaction between the control variables is neglected. There is a gap to find a solution approach to the integrated control problem so that all control inputs improve the performance of the objectives considering each other's actions. As can be found in the literature review in Chapter 2, a new class of problems named a differential game [3] has been introduced in the recent research where the control problem is defined as a differential game in which the control inputs are players that are trying to optimize a control cost function under some constraints, and the solution is provided by optimal control and game theory. Chapter 2 addresses an integrated control problem solved with the game theory for two specific control inputs. In this dissertation, differential game theory is utilized as a strategy that provides an optimal solution in problems with two or more two actuators trying to satisfy a combination of vehicle chassis control goals. For instance, in one scenario, active suspension and active anti-roll bars are working cooperatively to achieve the combinations of improving ride quality (minimizing vertical acceleration) and preventing rollover (or minimizing roll angle and roll rate). After presenting the mathematical solution for the Game Theory Optimal problem, the strategy is tested on some scenarios to explore the benefits and strengths of the multiple-player game approach compared to a single-player one.

1.2 Background information on vehicle active safety systems

Vehicle safety systems cover a vast and diverse area of research that started a long time ago and has come a long way from simple suspension or steering control to complicated integrated and robust controls. The three vertical, longitudinal, and lateral subcategories are the most relevant to this thesis and will now be briefly reviewed to motivate the research problem.

Vertical dynamics control targets increasing ride comfort and road holding, reducing fatigue and energy, and improving vehicle ability to execute evasive maneuvers by avoiding situations like spinning, drifting out, and rolling over. These targets are achievable by utilizing active or semi-active suspension systems. In semi-active systems, the control input is the viscous damping coefficient of the shock absorber, but in the active suspension system, a separate actuator is used in parallel with suspension elements to exert independent control force to reduce the vibration of the car body. This method is more effective than a semi-active system but is more expensive and consumes more energy[4]. Moreover, several research papers show that taking advantage of the preview information can cause noticeable improvements to stability and robustness. The type of preview information helpful in vertical dynamic control is the road profile (roughness), and there are many ways to generate it. It can be measured directly with a roller wheel in front of vehicles or some look ahead sensors, or it can be indirectly calculated from the overall response of preceding vehicle(s) as preview information for the follower vehicle(s)[5].

The engineer is mainly concerned with longitudinal accelerations or maintaining the vehicle speed at the desired value in longitudinal dynamics. Sometimes, it pursues more ambitious targets like collision avoidance and holding the space between convoys of vehicles. The most well-known systems that are now commercially used in many cars for longitudinal control purposes are adaptive cruise control (ACC) and automated highway systems (AHS)[4].

Yaw stability control is an essential part of lateral vehicle dynamics control that improves handling and stability. The control objectives are tracking the desired yaw rate[1] or sideslip[2], and sometimes both of them together. There are two common approaches in active chassis control[5]. The first one is the direct yaw moment control (DYC) method, in which the corrective yaw moment is the control input generated with either active braking or active torque distribution (active differential) [6]. In active braking, the controller distributes braking forces transversely to generate the proper corrective yaw moment, while in the active torque distribution method, the desired yaw moment is achieved by an active differential device that distributes left-right driving torque to the wheels. Although DYC can enhance stability in critical driving scenarios, it may be less effective for emergency braking on split-mu road surfaces and adds burden to drivers in high-speed cornering by decreasing the yaw rate. Hence active steering control is proposed. A corrective steering angle is added to the driver steering input to maintain the desired yaw rate and sideslip, and there are three ways to implement this method. The steering aid can be applied to the front axle (active front steering AFS) to improve maneuverability at low speed or the rear axle (active rear steering ARS) to improve handling at high speed. AFS and ARS (four-wheel active steering 4WAS) can also be combined to control lateral and yaw motion simultaneously using two independent control inputs[6].

There are more active safety methods used in the vehicle dynamics systems, such as lane departure warning system (LDWS), lane-keeping system (LKS), vision enhancement systems (like night vision), and driver condition monitoring. These are outside the scope of this research, and we will be only focused on vehicle stability

[1] In order to achieve the goal of tracking the desired yaw rate of the vehicle is estimated from the steering angle or the desired path. Please refer to Rajamani's Chapter 8 for the formulation [4]

[2] The angle between the vehicle's longitudinal axis and velocity vector

control[4].

Most of the controllers are focused on solving a single problem in one of the three previously-mentioned areas. Even a combination of controllers acts in a decentralized pattern in which each controller is making decisions independently without considering the other ones' actions. Due to the coupled dynamics, presence of load transfer, and nonlinear tire behaviors, one can easily conclude that one actuator's control actions can affect the states of the system in all vertical, longitudinal, and lateral dimensions. The control actions of one actuator can be either constructive or destructive to the goal of other actuators.

The optimal solution must consider all the control actions and control targets together. There is a need to develop an integrated control strategy in all three aspects of longitudinal, lateral, and vertical that allows the vehicle to optimally achieve the desired stability goal using the minimum control efforts for all the actuators. The differential game theory is introduced in the next section as a class of optimal control that deals with these problems.

1.3 Introduction to differential game theory and its application in vehicle dynamics

Most of the optimal control problems in vehicle dynamics (as reviewed in Chapter 2) formulate the cost function with a single objective function or a single control action (decision maker). However, there are some problems in which multiple actuators (players) are trying to achieve objectives in their cost functions. These problems are subject to a set of differential equations and are categorized as an extension of optimal control theory called the *theory of differential games*.

Using game theory in interactions between human drivers and vehicle stability control systems has recently grasped researchers' attention. This approach has many features like cooperative and non-cooperative optimization behavior, the option of having multiple players, enduring consequences of decisions, and robustness to changes in the environment. Most of the works done in this area are two-player differential games (the differential equation is describing states evolution over time) in which the driver is one of the players, and the other player is either AFS [20,21] or DYC [22-24]. These players' interactions are modeled in different paradigms in game theory (like decentralized[3], non-cooperative Nash[4], non-cooperative Stackelberg[5], cooperative Pareto[6] [25]). The game can also be varied based on the mode of play, equilibrium type, and information pattern. The play mode illustrates each player's behavior towards its own and other players' interests in a game. Players can pursue their interests and play a non-cooperative game or enter a binding agreement of interests and play a cooperative game. The equilibrium type is more about each player's adopted strategies in a strategy profile that builds the equilibrium. For example, each player can constitute its strategy by taking others' strategies into account, and they all act simultaneously, or it can be in a leader-follower pattern. The information pattern concerns each player's open-loop or closed-loop knowledge from the states of a game. The game theory approach can be beneficial in different scenarios. For instance, in a convoy of vehicles traveling close to each other, a driver might not see a bump or obstacle. Still, the active front steering (AFS) might be aware of the upcoming barrier using preview information from front vehicles and help the car avoid the obstacles or vise versa. Using game theory in

[3] In the decentralized paradigm, each players cost function is defined independent of other players' goal or action
[4] In Nash equilibrium, each player has its own strategy by taking other's strategies into account and all players act simultaneously
[5] In Stackelberg equilibrium, one player serves as a leader and the others are followers
[6] In Pareto equilibrium, the goal of each player is identical. This equilibrium is for cooperative games to reach global optimality

developing control strategies seems to be an intelligent approach in critical situations where different interests are involved, and players must make decisions to achieve global or separate goals.

There is space for new research to investigate the potential benefits of mixing different stability control methods like direct yaw control, active steering, active suspension, active anti-roll bar, etc. The two-player differential game can pursue different control objectives like collision or obstacle avoidance and provides more stability to the vehicle. There are many different unknown scenarios in which the game theory approach can be helpful and more research should be done to discover the strength of this approach. The game can also extend to an N-player differential game by considering human driver, active steering, differential braking, and active suspension systems as players to reach a global vehicle dynamic control.

1.4 Problem statement

Consider the linear dynamic system below:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad x(t_0)\text{: is given} \tag{1-1}$$

where A is an n by n state matrix; B is an n by m matrix, C is a $k \times n$ output matrix; D is a $k \times m$ matrix; x is a $n \times 1$ state vector; u is an $m \times 1$ control input vector; y is a $k \times 1$ output vector.

According to the optimal control approach, the controlled and control variables are introduced into a quadratic cost function with respective weighting matrices Q_s and R_s.

$$J = \int_{t_0}^{\infty} [x^T(\tau) Q_s x(\tau) + \bar{u}^T(\tau) R_s \bar{u}(\tau)] d\tau = J_x + J_u \tag{1-2}$$

The whole cost function J is a combination of the state regulation cost J_x and the control effort cost J_u. Both Q_s and R_s are constructed through weighting the matrices in the output equation:

$Q_s = C^T \bar{Q} C$ symmetric positive semi-definite matrix (1-3)

$R_s = D^T \bar{Q} D$ symmetric positive definite matrix

This representation is the standard *LQR* for state regulation problems. A more generalized form is a control coupled regulation problem that has the cross-coupling term between the state and control variables as shown below:

$$J = \int_{t_0}^{\infty} [x^T(\tau) Q x(\tau) + 2x^T(\tau) N u(\tau) + u^T(\tau) R u(\tau)] d\tau = J_x + J_c + J_u \text{ [27]} \tag{1-4}$$

$$Q = C^T \bar{Q} C,$$
$$N = C^T \bar{Q} D,$$
$$R = D^T \bar{Q} D + \rho \bar{R}$$

In a single-player or decentralized paradigm, the control input comes from an actuator or a player that wants to achieve a vehicle safety or stability goal while minimizing the actuator control effort. This goal is modeled in a quadratic form in an optimal cost function, a combination of weighted goal and weighted control effort regulators. The optimal solution is the control action (or control actions) that minimizes the designed optimal cost function. Here is the list of players (actuators) and vehicle stability goals to pick from:

Players:

- Active-steering: assisted active steering angle that is added to the human driver steering angle
- Active-suspension: applies active or semi-active suspension forces
- Corrective roll moment: active anti-roll bar generates the corrective anti-roll moment around the roll center
- Corrective yaw moment: This moment is either created by applying differential brake forces or active torque distribution to tires

Goals:

- Roll stability: regulates roll angle and roll rate to provide better roll stability
- Anti rollover: regulates and maintains the roll index between (-1,1) to prevent rollover. If the roll index stays in this range, no tire will leave the ground.
- Ride quality: regulates the vertical acceleration to provide a better ride experience for the passengers
- Rattle space: regulates the suspension deflections
- Road holding: regulates the tire deflections
- Yaw stability/tracking: regulates yaw rate error, yaw error, and lateral error

Some of these goals can either be achieved directly from optimal control or indirectly observed as a by-product of other goals. In this dissertation, the objective of maintaining the roll index between (-1,1) is observed indirectly by regulating roll angle and roll rate. The rattle space and road holding are also not implemented directly in the cost function definition, but they are monitored in simulations.

The single-player game consists of only one actuator. A two-player game cost function includes the control efforts of two actuators trying to achieve a goal together, and so on for an N-player game. The paradigm is called decentralized if each player's cost function is defined independently of the other player's control action. The Nash paradigm cost function includes control actions of multiple players. The solution to this problem is the set of control actions (strategy) in which no player can improve the optimal cost unilaterally by deviating from that strategy. Here is the representation of the $N \geq 2$ player Nash problem [3]:

Define the cost function $J_i(t, x, u_1, \ldots, u_N)$ as:

$$J_i = \int_0^T \mathrm{L}(\tau, x(t), u_1(t), \ldots, u_N(t))\, \mathrm{d}t, \mathrm{i} = 1, \ldots, \mathrm{N} \tag{1-5}$$

where x(t) is the solution to a dynamic system below

$$\dot{x} = f(\mathrm{t}, x(\mathrm{t}), u_1(\mathrm{t}), \ldots, u_N(\mathrm{t})), \mathrm{x}(0) = \mathrm{x}_0 \tag{1-6}$$

The Nash solution is a set of control actions $(u_i^*, \ldots, u_N^*)$ if for all $u_i^* \in U_i, i \in N$ the following inequality is satisfied for *i=1,2,…, N*

$$J_i^* \triangleq J_i(x(t), u_1^*, u_2^*, \ldots u_N^*) \leq J_i(x(t), u_1^*, u_2^*, \ldots, u_{i-1}^*, u_i, u_{i+1}^*, \ldots, u_N^*) \tag{1-7}$$

In this dissertation, the solution for a decentralized paradigm is compared to the solution for two-player and three-player Nash. The two-player combinations are as follows:

- Active steering + Corrective yaw moment
- Active steering + Corrective roll moment
- Active suspension + Corrective roll moment

and the three-player combination is: Active steering + Corrective yaw moment + Corrective roll moment

1.5 Objectives

This thesis will develop a solution to two-player and three-player differential games and explore the game theory approach's benefits in some vehicle dynamics optimal control problems. Here are the specific objectives explored in this research:

a) Developing linear control models and nonlinear vehicle models with linear and nonlinear tire models
b) The solution to the two-player infinite horizon linear quadratic regulator (LQR) control problem for continuous systems (this is a contribution based on the solution for a discrete system that already exists in the literature[22])
c) Solution for the two-player control coupled output regulator (CCOR) (this contribution is achieved by combining the cost functions)
d) Investigating the game theory approach for improvement of vehicle dynamic systems control in a two-player differential game between each of the two chosen players from the following list: active front steering (AFS), active suspension, corrective yaw moment, and corrective roll moment affected as active anti-roll bar (AARB)
e) Extension of the solution and scenarios to the three-player differential game between active front steering (AFS), corrective yaw moment, and corrective roll moment (the solution for the three-player game is also a contribution of this thesis)

1.6 Scope of the research

This project's overall purpose is to use differential game theory to solve optimal control problems in vehicle dynamics with more than one controller or player. This research applies the game theory approach to enhance vehicle stability and safety in lateral and vertical dynamics using a combination of actuators, including active steering, corrective yaw moment, active suspension, and corrective roll moment as an active anti-roll bar. A nonlinear plant model is designed to investigate the proposed controller on a model that accurately captures nonlinear vehicle dynamics. More specifically, the thesis will address the following research questions:

- What is the advantage of using the game theory paradigms over the decentralized set of optimal controllers?
- How to extend and solve the problem from a two-player game to a three-player game?
- What are some scenarios and objectives in lateral and vertical vehicle dynamics for which using game theory can be beneficial?

Outcomes and contributions:

- Formulation and solution of the two-player linear-quadratic problem with infinite horizon for continuous systems
- Formulation and solution of the two-player output regulation problem by combining the cost functions of players into a new general cost function
- A method for application of three-player game theory to integrated vehicle chassis control, with solutions to game-theoretic control problems involving the following players: active steering, active suspension, roll control, yaw control
- An algorithm to solve the coupled Riccati equations for a continuous three-player LQR Nash equilibrium game
- A numerical simulation study demonstrating the capability of *N-player* optimal game theory to reduce unwanted chassis responses while reducing control effort, compared to traditional decentralized control design methods

1.7 Thesis Structure

Chapter 2 represents the literature review in modeling, stability control, and optimal game theory. Once the problem is motivated, the subsequent research proceeds in three parts:

The first part is devoted to modeling the control design and plant models. Control models are low degrees of freedom (*DOF*) linear models (such as *2DOF* bicycle models for lateral dynamics and *4DOF* roll models for vertical dynamics) capable of capturing the car's dominant dynamic behaviors. These models are explained mathematically and coded in *MATLAB* software to design and test the controllers. The plant model should be as close as it can be to a real vehicle. As the goal is to explore the global vehicle stability, a car model must simulate the vehicle response for states' feedback, testing the controllers, and validation. A *6DOF* nonlinear car model is chosen as the simplest model that meets the requirements. This model combines the *2DOF* bicycle model and *4DOF* roll plane model, including lateral motion, yaw dynamics, roll dynamics, and vertical oscillation of four wheels. This model is equipped with a nonlinear tire model interface that generates tire forces. Lateral tire forces are calculated from slip angles and the tire's normal forces. In Chapter 3, the yaw plane and roll plane and combined yaw/roll plane models are introduced as the control models, and the roll index is also defined as a parameter to measure anti-roll performance. Then the combination of these two models is presented, and in Chapter 7, the nonlinear tire model is added to the plant model and tested in simulations.

The second part includes the introduction and the mathematical representation of the game theory problem for a two-player game. It provides a solution for the linear quadratic regulator (*LQR*) and control coupled output regulation problem (*CCOR*). Existing literature is used to explain why the *CCOR* method is preferred for the controller design in Chapter 4. Finally, the two-player solution in the *LQR* and *CCOR* solution for a three-player game is presented in Chapter 5.

The last major part is the simulation results for control models Chapter 6 and the plant model with nonlinear tire model Chapter 7. In Chapter 6, the two-player game is explored in a differential game between active-steering versus corrective yaw moment, active-steering versus corrective roll moment, corrective roll moment versus active suspension. A three-player game is also presented, including active steering, corrective yaw moment, and corrective roll moment. The optimal control gains are calculated for decentralized and Nash paradigms for each scenario. Then, the closed-loop simulation results are shown using the *6DOF* plant model with a linear tire model. It is shown that the Nash solution is capable of achieving higher objective performance with lower control efforts by splitting the burden between players. This benefit can help situations where actuators get saturated and are incapable of producing the required control efforts. In Chapter 7, the linear tire model with saturation and nonlinear Pacejka tire model is introduced and compared. The *6DOF* plant model from Chapter 6 is equipped with the nonlinear tire model, and the simulation results are shown for two case studies (the two-player game between active steering and corrective yaw moment and the three-player game between active steering, corrective yaw moment, and corrective roll moment).

Chapter 8 presents a summary of the work, highlights the contributions of this dissertation, and suggests future research directions.

2 CHAPTER 2: LITERATURE REVIEW

2.1 Modeling literature review

In order to examine, analyze, and design a controller, vehicle dynamics models are essential. The models are either mathematical models or built-in models from commercial car simulation software like CarSim [7]. The mathematical ones are obtained from Newton's law and can be either linear or nonlinear. The vehicle's equations of motion are written according to the desired number of degrees of freedom. Consequently, the model with more degrees of freedom is potentially closer to actual car dynamic behavior. According to desired control objectives, many different models have been used in the literature. For the simple task of controlling a suspension in one wheel, a *2DOF* quarter car model can be used, including unsprung mass, sprung mass, and a suspension system between them and the road. The quarter car model can only be used in vertical dynamic analysis. If one is interested in analyzing the pitch movements, two quarter car models can be joined together to make a *4DOF* half-car model. Roll dynamics can also be investigated using a *4DOF* half-car model. The simplest model to investigate yaw dynamics is called a bicycle model. The left and right wheels are considered one wheel, and the dynamic system states are lateral velocity or sideslip and yaw rate. A complete model that is generally used for integrated control purposes closer to the real car model is a full-car model, a combination of four quarter car models that includes all the yaw, roll pitch, and heave dynamics. More details about the mentioned models can be found in Rajamani [4]. Aripin et al. [8] generally reviewed full car models and these are summarized in the chart below.

Table 2-1: Number of DOF of the nonlinear vehicle models [8]

Number of DOF	Dynamic motions	Output variable
7 DOF	(i) Longitudinal (ii) Lateral (iii) Vertical (iv) Rotational of 4 wheels	Yaw rate & Sideslip
8 DOF	(i) Longitudinal (ii) Lateral (iii) Vertical (iv) Roll (v) Rotational of 4 wheels	Yaw rate, roll rate, and sideslip
14 DOF	(i) Longitudinal (ii) Lateral (iii) Vertical (iv) Roll (v) Pitch (vi) Bounce (vii) Rotational of 4 wheels (viii) Vertical oscillations of 4 wheels	Yaw rate, roll rate, pitch rate, and sideslip

A proper tire model is also required to obtain the tire forces, which can be either linear tire model or nonlinear. Pacejka and Dugoff are the most common nonlinear tire models that their descriptions can be found in vehicle dynamics books like [4,26].

After defining the control objectives, a simple model for control design can be selected. A more complicated model can validate and obtain real vehicle model feedback signals. Besides the mathematical model, there is also a commercial vehicle dynamics software known as *CarSim*[7] with multi-degree of freedom vehicle models that have the highest fidelity practical for simulating vehicle responses and interfacing with controllers [7].

[7] Manufacturer website: https://www.carsim.com/

2.2 Stability control literature review

There are many control strategies reported in the literature to maintain vehicle stability in each level of longitudinal, vertical, and lateral dynamics individually. Since this research focuses on lateral, vertical, and roll dynamics, this literature review does not include longitudinal dynamics control.

Hrovat [9] applied optimal control theory to generate forces in an active suspension system for the quarter, half, and full car models. Bender [10] and Hac [5] considered road preview information to generate active suspension forces, which improved car body vertical acceleration, tire, and suspension deflection performance. Marzbanrad et al. [11] studied the stochastic optimal preview control of vehicle suspension and the effect of preview time. They announced the improvement of performance compared to active or passive suspension without the preview information. The active suspension appears to be one of the common control inputs to enhance the vehicle's vertical stability. This dissertation also picks active suspension as one of the control inputs for its control objectives.

In 2014, Aripin et al. [6] did a complete review on active yaw control systems for vehicle handling and stability enhancement. All active chassis control methods, including *DYC* (Direct Yaw Control using active braking or active differential), active steering (*AFS*[8], *ARS*[9], *4WS*[10]), and integrated active steering and yaw moment control, are included in this paper review. All the control strategies are discussed and compared to each other, summarized in the chart below.

[8] Active Front Steering
[9] Active Rear Steering
[10] Four wheel Steering

Table 2-2: Yaw stability control algorithms [8]

Control algorithms	Active chassis control	Control objective	Advantages	Disadvantages
PID controller	DYC	sideslip	Anti-wind-up strategy to avoid high overshoot and large settling time	Uncertainties are not consider
LMI static state feedback	Integrated AFS-active differential	Yaw rate and sideslip	robust for uncertainties	
H_∞	Integrated chassis control, active steering	Yaw rate	Robust for uncertainties, reject disturbance	Transient response improvement is not consider
SMC	DYC, active steering	Yaw rate and sideslip	robust for uncertainties and reject disturbance	
OGCC	Integrated AFS-DYC	Yaw rate and sideslip	Robust for uncertainties	
Adaptive integrated control	Integrated AFS-DYC	Yaw rate and sideslip	Robust for uncertainties	
Mixed-sensitivity minimization control	DYC	Yaw rate	Robust for uncertainty, reject disturbance	
PI controller	4WAS	Yaw rate	Robust for uncertainties	
IMC	DYC	Yaw rate	Robust for uncertainty	Transient response improvement is not consider
QFT	AFS	Yaw rate	Robust for uncertainties, reject disturbance	
μ synthesis control	4WAS	Yaw rate and sideslip	Robust for uncertainties	
SMC-backstepping		Yaw rate and sideslip	Robust for nonlinearities	Uncertainties are not considered
SMC-FLC	Integrated steering, brake, and suspension	Yaw rate, sideslip, and roll angle	Robust for uncertainties and nonlinearities	Transient response improvement is not consider
SMC-LQR	DYC	Yaw rate and sideslip	Robust for uncertainty	

The major factor that can clarify these different strategies' advantages and disadvantages is their robustness to uncertainties. The uncertainties are variations of dynamic parameters like road surface adhesion coefficients, tire cornering stiffness, vehicle mass, vehicle speed, vehicle moment of inertia, and external disturbances like longitudinal and lateral crosswinds. This thesis chooses active front steering and corrective roll moment as the control inputs to control yaw dynamics. Besides the combined effects of these control inputs on yaw dynamics, their influence on the roll dynamics is also investigated. It is shown that active steering can either improve the tracking at the cost of decreasing roll performance or can improve roll stability by counter-steering at the cost of losing the tracking.

When a vehicle encounters an evasive maneuver like high-speed cornering or cornering on icy roads, the load transfer between the tires or low friction coefficient makes it difficult for a vehicle to track the desired path or remain stable. Different things can happen in critical cornering maneuvers, like the saturation of lateral force on tires. The vehicle loses its steerability and drifts off, spins out, or skids. In some situations, the increased roll index (mostly in vehicles with a high center of gravity) makes the vehicle rollover. Most vehicle dynamic system stability controllers are focused on yaw stability and try to use the *DYC* method to compensate for the yaw moment required to stay on the desired path. No matter how good the tires are, there is always a limit on the amount of lateral force generated. This limit depends on many factors, including friction coefficient and changes according to normal forces on tires. Load transfer during cornering will change the normal forces at each tire and makes it hard to generate the maximum required lateral force to stay on the track. Normal forces on tires can be affected by active or semi-active suspension systems. The mentioned limitations and complexity of vehicle dynamics make the controllers continuously seek new methods to approach vehicle stability and maneuverability problems. Recent research has shown that applying integrated yaw moment control and suspension control can help vehicle yaw stability and steerability and provide adequate roll stability. In this research, the game-theoretical approach is explored as one of the new methods to address the mentioned problems. It was shown that this method could split the control burden between different control inputs called players and enhance vehicle maneuverability and stability despite the presence of saturation in tire forces or some control inputs. Tim Gordon et al. [8] conducted a review on the methodologies and architectures of different control methods in integrated control for road vehicles with an emphasis on the flow of the control information, but there was no mention of the game theory approach in the listed architectures.

In 2005, Shim and Margolis[12] used a trial and error method to indirectly control the normal force at tires with suspension actuators to reduce the yaw rate error. When the vehicle makes a left-hand turn and yaw rate error is positive (understeer car), the force actuators in the right front and left rear wheels are activated to reduce normal tire forces to add an oversteer effect. Similarly, for right turns, left front and right rear actuators are activated. As mentioned in the article, the proposed PD controller at each wheel improved lateral vehicle motion, but no optimization was done. Due to the coupling of control inputs, a *MIMO* (multi-input multi-output) controller is required to achieve more effective results. The game-theoretical approach in this thesis allows us to investigate the effects of different control inputs on control objectives using the optimal control in a more sophisticated way which is superior to the trial and error approach.

Chou and D'Andrea developed a global chassis control using the collaboration of differential braking and active suspensions[13]. Desired yaw rate and longitudinal acceleration are followed by using longitudinal slip ratios of the wheels as intermediate control inputs controlled by braking torques. A constrained optimization algorithm is used in vertical dynamics to regulate roll rate, pitch rate, and vertical velocity considering vertical forces applied to the wheels as an intermediate control variable. It was also shown that the controllability problem could encounter a singularity if the yaw equation is considered explicitly in vertical dynamics. This approach was a good step towards solving the global control problem with different control inputs, but due to the control structure, there was no solution suggested that guarantees the best solution due to the combined effects of the control inputs. In this dissertation, the solution to the differential game theory problem presents the optimal response for the chosen combination set of different control inputs. The effects of each control input can also be tuned by changing the corresponding weights.

Combined control effects of brake and an active suspension with active stabilizers have been investigated in [14]. It was shown to be beneficial in tracking the desired yaw rate during critical steering, reducing the roll angle, and influencing ride comfort and road-holding capabilities. The combined effect of the brake (ESC) and the active suspension are investigated to control yaw rate, side slip angle, and roll rate. The desired yaw rate is calculated based on vehicle speed and steering angle, and the sliding mode controller tracks the calculated yaw rate by applying differential brakes on tires. To maintain ride comfort in cornering, a cost function that includes heave, pitch, and roll accelerations, four suspension deflections, four integrals of suspension deflections, and four tire deflections is defined. The optimal control method is used to derive control gains, and it is shown that keeping the chassis flat will reduce the load transfer and enhance road holding, and the flat vehicle has better yaw tracking performance than a tilted vehicle. The controllers presented in [14] were working in a decentralized manner together, and keeping the chassis flat can be one of the optimal answers in the whole domain of answers. The counter effects of the control inputs on each other's objectives must be considered in the optimal function to find the optimal solution that costs less control effort. This dissertation investigates the effects of more control inputs, including corrective roll moments, active steering, active suspension, and corrective roll moment in a game theory configuration in which all control inputs' actions are considered within each player's objective function.

Chu et al. [15] tried a hierarchical control approach using smooth sliding control as an upper-level controller to derive the desired yaw rates and roll rates that prevent a vehicle from skid and rollover. The lower controller applies brakes to each wheel to track the desired yaw rate and calculates the *MR* (magnetorheological) damper's current at each corner to prevent rollover. It was shown that differential braking partially improves roll stability, and semi-active suspension directly and efficiently maintains stability by producing anti-roll moments. Once again, the counter effects of each control input on each other are not investigated directly, so there will be no guarantee to achieve the optimal solution.

March et al. [16] applied fuzzy control to develop an active front steering control and normal force control (steering and suspension controllers) and proved that the performance is improved with an integrated approach. Li et al. [17] showed that integrative *DYC* and *4WS* controllers could significantly improve the stability performance compared to individual *DYC* or *4WS* controllers. The active-roll control is also added to reduce the body roll angle, and it is shown that it can indirectly affect vehicle handling and yaw control in a beneficial way by further tuning the controller weights in the cost function. This approach emphasizes the benefits of using integrated approach but can't guarantee the global optimal control when the effects of control inputs on each other are considered.

Different approaches and controllers are used in the literature as mentioned above to investigate the combined effects of direct yaw control and suspension control, and they all seem to improve general vehicle stability in critical maneuvers. Yet, some more works are required to reach a higher performance. According to both vehicle and controller dynamics, it takes some transient time for a vehicle to execute and maintain the desired stability in unexpected maneuvers since the vehicle has no information on what to expect in the future. Hac introduced the idea of using preview information of road elevation in active suspension control [5]. The continuous-time optimal control method was used to investigate the effects of preview information about road roughness on objectives like road holding, ride comfort, and suspension working space, and improvements shown in road-holding performance. Even driver models use preview information of the road to generate lateral preview errors or lateral error areas to calculate the required steering input to stay on the path [18]. To prevent rollover, Yim designed a linear optimal preview controller using the steering input as preview information. Control actuators are differential brakes and active suspensions [19]. Compared to non-preview controllers, higher active suspension forces are generated in preview controllers that significantly reduce the roll angle and lateral acceleration at the cost of increased yaw rate error, resulting in larger brake inputs. This paper shows how vehicle stability performance can benefit from preview information to avoid rollover. On the other hand, it introduces a trade-off between preview control and a yaw rate control that rises by making the controlled vehicle go through the understeer phase. Preview control is not considered in this dissertation, but it was assumed that the human driver already has the knowledge of the road ahead, and according to that, the open-loop human steering is generated. The game theory method presented in this thesis is capable of considering the preview information to improve tracking, but in this dissertation, this is not investigated and is suggested for future works as a horizon to pursue.

The mentioned trade-off in Yim's work raises the question of how to define a criterion that leads to higher performance of an integrated controller of differential braking and active suspension, and the next question is how to solve it. The optimal controller is the most common controller that is applied in the literature to provide ride comfort, better road holding, and regulating heave, roll, and pitch motions. All objectives are weighted in a cost function leading to desired performance criteria, and the cost function is minimized under the physical constraints of the dynamic system. Optimal control is also used in some literature to derive the braking pressures to control the vehicle's yaw dynamics. In most papers on integrated control of vehicles, the two control problems that deliver the active suspension inputs and braking pressures are solved in a decentralized way. A specific control approach is required that deals with the interactions among multiple input parameters. According to coupled lateral and vertical dynamics, one optimal controller can't simply optimize its cost function independent of the other optimal controller input.

The literature review concluded that the integrated control has the potential for higher performance and better stability. Still, most of the research on integrated control approaches the problem as two or three subproblems that are solved individually and then merged into a global chassis control that achieves the desired objectives. However, the controllers' interactive effect is being left out in most cost functions and dynamic modeling, and there is a need for research to address this issue. Differential game theory is the class of optimization problems that address the interactive effects of the control inputs, and the literature review of this method in vehicle dynamics is presented in the next section.

2.3 Game theory optimal in vehicle dynamics

As mentioned in the previous section, game theory optimal deals with a class of optimal control problems where each player's cost function considers the control action of other players. The application of this method in vehicle dynamics is relatively new, so there is a limited amount of works found in the literature, which are as follows.

Na and Cole [20, 21,30] proposed a new game-theoretical approach to model the human driver's steering interaction with active steering. Controllers are defined as players in a differential game that try to minimize their desired cost functions in path following scenarios. They investigated the players' interaction in different modes (cooperative or non-cooperative) and different paradigms (Nash, Stackelberg, and Pareto). The comparison was made with the decentralized paradigm (in which the driver and collision avoidance controller disregard each other). Noncooperative Nash and Stackelberg paradigms can predict drivers' behavior while collision avoidance controllers actively compensate for drivers' steering action. The Pareto paradigm models the interaction between the driver and the collision avoidance in a cooperative way. Model Predictive Control and Linear Quadradic dynamic optimization are used as two mathematical approaches to study and solve each paradigm. It was shown that different steering behaviors could be achieved according to changes in path-error cost function weights in each paradigm. As one of the first research efforts in the application of game theory in vehicle dynamics, this research only addressed the steering problem but showed a significant potential for future studies.

Tamaddoni et al. [22-24] also used the Nash game theory approach in a two-player cooperative difference game in which the players are the driver and direct yaw control (*DYC*). It was shown that the optimal preview control could significantly enhance vehicle stability by reducing lateral velocity, yaw rate, and roll angle. Tamaddoni et al. proposed an algorithm to solve the two-player Nash game for a discreet system. In this dissertation, the solution for two-player and three-player Nash games for continuous systems is developed using a similar approach from Tamaddoni's work, and its application is also explored in scenarios besides the game between steering and *DYC*.

Huang et al. [27] presented an extension of the *LQR* method called *CCOR* (Control Coupled Output Regulation). The Rollover Index was introduced to merit roll stability performance in heavy articulated vehicles with multiple axles. Active anti-roll bars for the different axles (as players) minimize a performance index with multiple rollover indices. At last, a design algorithm is presented to compare various control configurations to select the final design. The method presented in Huang's work is for single-player games, and there is a need to explore this method in a multi-player game. In this thesis, the application of control coupled output regulation is explored in regulating sprung mass vertical acceleration to improve the ride quality, and the solution for two-player and three-player games is also introduced for this class of problems.

Similar to the works of Na and Tamaddoni, a general review of the whole literature in the application of game theory in vehicle dynamics shows that the major focus is on lateral and longitudinal dynamics. Game-theory paradigms and the solving methods can be two factors to distinguish between the research. For instance, Li et al. [31] used noncooperative Nash paradigm and distributed model predictive control method (DMPC)[11], but Yang et al. [32] used the cooperative Stackelberg paradigm and model predictive control (MPC)[12] to solve the game between human driver and active steering and showed that more maneuver force is saved relative to the Nash paradigm. The game theory approach is also used popularly in modeling human driver behavior in [33,34,35,36] using both Nash and Stackelberg paradigms in lane change and U-turn maneuver. The interactions between human drivers with autonomous vehicles and automated vehicles with themselves (*CAVs*)[13] in highway maneuver scenarios like Lane change and cruise control were also studied in [37,38,39]. The objective in highway systems is mostly about safety and collision avoidance. These interactions were further studied in developing decision-making processes in agent-based or un-signalized intersections between different drivers [40,41,42,43,44]. Both cooperative (Stackelberg and Pareto) and noncooperative (Nash) paradigms were studied in the intersection traffic control. The studies related to highway and intersection traffic control are mostly brought here to show the scope of the application of game theory in the vehicle industry and are less related to this dissertation which is focused only on the players/actuators within the vehicle dynamic models.

Four-wheel independent actuated electric vehicles are another benchmark for

[11] Distributed model predictive control refers to a class of predictive control architectures in which a number of local controllers manipulate a subset of inputs to control a subset of outputs (states) composing the overall system
[12] MPC is an optimization method that uses a model of the plant to make predictions about its future outputs
[13] Connected Automated Vehicles

exploring the game theory approach. Zhang et al. [45] solved a cooperative Pareto game using the *DMPC* method to achieve stability in the presence of various actuator failure scenarios. The non-cooperative Nash paradigm was used by An, Q. et al. [46] to achieve trajectory control and yaw stability. The Nash problem was solved using the dynamic programming method, and the simulation results from Carsim were also validated experimentally.

A limited amount of works were done on the application of game theory outside the scope of longitudinal and lateral dynamics. Han et al. [47] work conducted a multi-objective optimization in vertical dynamics. Ride quality, suspension deflection, and tire relative dynamic load were chosen as objectives, and the design variables were suspension stiffness and the driver's seat stiffness and damping. The problem was solved in a cooperative scheme for passive suspension. Dextreit and Kolmanovsky [48] used the Stackelberg paradigm in a game between driver and power train to penalize fuel consumption, battery state ad provide good drivability. There is a shortage of studies in the application of game theory in vertical dynamics that need to be addressed.

The latest paper review of 100 articles conducted by Ramos et al. [49] showed the increasing use of autonomous systems (*AS*) to improve costs and safety in the years 2015 to 2021. It was concluded that the anticipation of both autonomous and non-autonomous systems' possible decisions during interactions is crucial to identify and analyze the hazards and risks, and the application of the game theory for analysis under risk perspective can be considered in an early stage.

The game-theoretical approach seems promising in integrated vehicle dynamics control; however, its application in vehicle stability has not been thoroughly investigated in the literature. Most of the focus of the literature were on the lateral and longitudinal dynamics, and there is a gap of studies in multi-objective controllers involving roll and vertical dynamics. This dissertation explores game theory solutions for *LQR* and *CCOR* methods for the Nash paradigm for both two-player and three-player games. Active steering, corrective yaw moment, active suspensions, and corrective roll moment are control actions of so-called players that participate in the two-player and three-player scenarios. The objectives selected for the players are to track the desired yaw rate, ride quality, roll stability, and rollover prevention.

3 CHAPTER 3: VEHICLE MODEL

3.1 Introduction

Generally, two types of models are required for any control problem. The first type is the control model used in the controller design to calculate controller output. The second one is the simulation model on which the controller performance is being investigated. The simulation model in vehicle dynamics can be a real car or a complete model close to a real vehicle (modeled using commercial software such as CARSIM). It can also be a mathematical model that is less complicated than high *DOF* models but simultaneously complex enough to show required behaviors close to a real car. On the other hand, the control model is chosen to be relatively simpler than the simulation model. This thesis uses simple linear models for the controllers' development, and a nonlinear plant model is developed from combining the control models for simulation.

The first step to investigate the problem using the game theory approach is defining the players. In this thesis, all of the control paradigms are based on the interactions between three players, which are:

- Active suspension control input (left and right suspension forces)
- Anti-roll control input [Active Anti Roll Bar (*AARB*) generates corrective roll moment]
- Active front steering input
- *DYC* (direct corrective yaw moment was chosen as control input)

It is essential to understand each player and their control action before taking any step forward in modeling.

Active suspension actuators on each wheel apply active suspension control inputs. One control input is required for a quarter car model, two control inputs for a half car model, and four for a full car model.

An anti-roll control input is a corrective roll moment that can either be created by the same active suspension actuators or created by an active anti-roll bar. The corrective roll moment from the active suspension can be divided into two actuator forces considering the distance between the left and right tires, and the resultant forces can be applied to each tire individually.

Active steering is a control input added to human driver steering input and can adjust the steering angle for different objectives like tracking, obstacle avoidance, or even counter-steering to avoid rollover. This input can be added to front wheels, rear wheels, or all four wheels. In this thesis, only active front steering is studied for the sake of simplicity and because this is the most popular configuration in actual passenger cars. Corrective yaw moment is also chosen as an input to control yaw dynamics and improve tracking.

3.2 Model development

In this section, two different models are developed. The first model describes the vehicle's vertical and roll motion dynamic behavior, and the second model illustrates the vehicle's lateral dynamics or yaw movement.

As a start point, a *4DOF* half car roll model is presented considering only one player (1^{st} player: active suspension, control action: active suspension forces). After developing the mathematical model shown schematically in figure 3-1, the model is improved by adding the second player control action (2^{nd} player: anti-roll, control action: corrective roll moment).

For the lateral dynamic model, a common *2DOF* bicycle model is presented using a linear tire model to develop a simple continuous model for the states of the system. This model includes the control action of the third player. (3^{rd} player: active yaw, control action: corrective yaw moment).

The final model combines the two developed models shown in state-space representation, which describes the vehicle's dynamic behavior in vertical, lateral, and roll motions. This model includes control actions of all four players and shows the coupling between two models due to lateral acceleration.

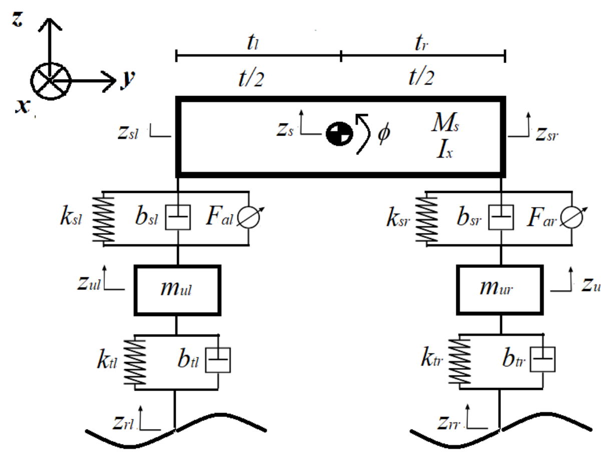

Figure 3-1: 4DOF roll model schematic

Figure 3-1 shows a simple *4DOF* half-car roll model with active suspension. It represents the motion of the axle and the vehicle viewed from behind. This model is a combination of *2DOF* quarter car models with the same sprung mass. The suspension system consists of a spring with the stiffness of (k_s), a passive damper with a damping coefficient of (b_s), and an active force actuator (F_a) placed in parallel with the damper and spring for each tire. The subscript "*r*" is being used for the right tire and "*l*" for the left tire. According to this terminology, the active suspension actuating force for a right tire is (F_{ar}), and the rest will be presented accordingly. The sprung mass (M_s) represents the half car equivalent of the total body mass of the vehicle. If the model represents the full car with two axles, then (M_s) stands for the full car sprung mass, and each suspension element will be the parallel equivalent of the two axles. Two un-sprung masses are presented as (m_{ul} and m_{ur}) to model an equivalent mass due to axle and tire for right and left tires. The vertical stiffness of each tire is represented by the spring (k_t) and damper (b_t). The variables z_s, $z_{y,}$ and z_r represent the vertical displacements from the static equilibrium of the sprung mass, unsprung mass, and the road elevation, respectively, and are subscripted by "*r*" or "*l*" according to the right or left tires.

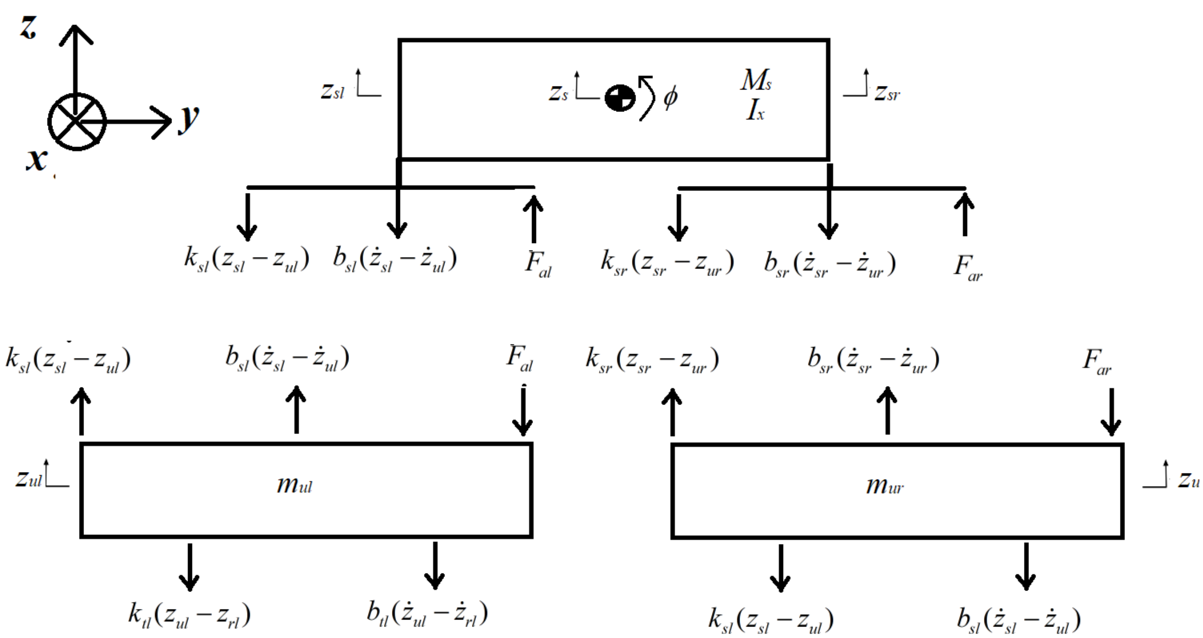

Figure 3-2: Free Body Diagram of sprung and unsprung masses

The equations of motion can be written as below based on the free body diagram in figure 3-2:

$$M_s\ddot{z}_s + k_{sl}(z_{sl} - z_{ul}) + b_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) + k_{sr}(z_{sr} - z_{ur}) + b_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) = F_{al} + F_{ar} \tag{3-1}$$

$$I_x\ddot{\phi} - k_{sl}.t_l(z_{sl} - z_{ul}) - b_{sl}.t_l(\dot{z}_{sl} - \dot{z}_{ul}) + k_{sr}.t_r(z_{sr} - z_{ur}) + b_{sr}.t_r(\dot{z}_{sr} - \dot{z}_{ur}) = -F_{al}.t_l + F_{ar}.t_r$$

$$m_{ul}\ddot{z}_{ul} + k_{tl}(z_{ul} - z_{rl}) + b_{tl}(\dot{z}_{ul} - \dot{z}_{rl}) - k_{sl}(z_{sl} - z_{ul}) - b_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) = -F_{al}$$

$$m_{ur}\ddot{z}_{ur} + k_{tr}(z_{ur} - z_{rr}) + b_{tr}(\dot{z}_{ur} - \dot{z}_{rr}) - k_{sr}(z_{sr} - z_{ur}) - b_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) = -F_{ar}$$

assuming the center of gravity in the middle of the rigid sprung mass and small values for roll angle $\phi \leq 10^\circ$ ($\sin(\phi) \simeq \phi$), equations below can be written:

$$t_l = t_r = t/2 \tag{3-2}$$

$$z_{sr} = z_s + t_r.\phi \rightarrow \dot{z}_{sr} = \dot{z}_s + t_r.\dot{\phi}$$

$$z_{sl} = z_s - t_l.\phi \rightarrow \dot{z}_{sl} = \dot{z}_s - t_l.\dot{\phi}$$

using the assumptions above and removing the four auxiliary states ($z_{sl}, \dot{z}_{sl}, z_{sr}, \dot{z}_{sr}$) equations of motion can be written as:

$$M_s\ddot{z}_s + k_{sl}(z_s - t_l.\phi - z_{ul}) + b_{sl}(\dot{z}_s - t_l\dot{\phi} - \dot{z}_{ul}) + k_{sr}(z_s + t_r\phi - z_{ur}) + b_{sr}(\dot{z}_s + t_r\dot{\phi} - \dot{z}_{ur}) = F_{al} + F_{ar} \tag{3-3}$$

$$I_x\ddot{\phi} - k_{sl}t_l(z_s - t_l\phi - z_{ul}) - b_{sl}t_l(\dot{z}_s - t_l\dot{\phi} - \dot{z}_{ul}) + k_{sr}t_r(z_s + t_r\phi - z_{ur}) + b_{sr}.t_r(\dot{z}_s + t_r\dot{\phi} - \dot{z}_{ur}) = -F_{al}t_l + F_{ar}t_r$$

$$m_{ul}\ddot{z}_{ul} + k_{tl}(z_{ul} - z_{rl}) + b_{tl}(\dot{z}_{ul} - \dot{z}_{rl}) - k_{sl}(z_s - t_l\phi - z_{ul}) - b_{sl}(\dot{z}_s - t_l\dot{\phi} - \dot{z}_{ul}) = -F_{al}$$

$$m_{ur}\ddot{z}_{ur} + k_{tr}(z_{ur} - z_{rr}) + b_{tr}(\dot{z}_{ur} - \dot{z}_{rr}) - k_{sr}(z_s + t_r\phi - z_{ur}) - b_{sr}(\dot{z}_s - t_r\dot{\phi} - \dot{z}_{ur}) = -F_{ar}$$

The corrective roll moment $M_{\phi c}$ is the control action of the 2nd player, which can be translated to the equivalent force of F_c acting as an actuator force on the right and left suspension system according to figure 3-3 and can be written as:

$$F_c = M_{\phi c}/t \tag{3-4}$$

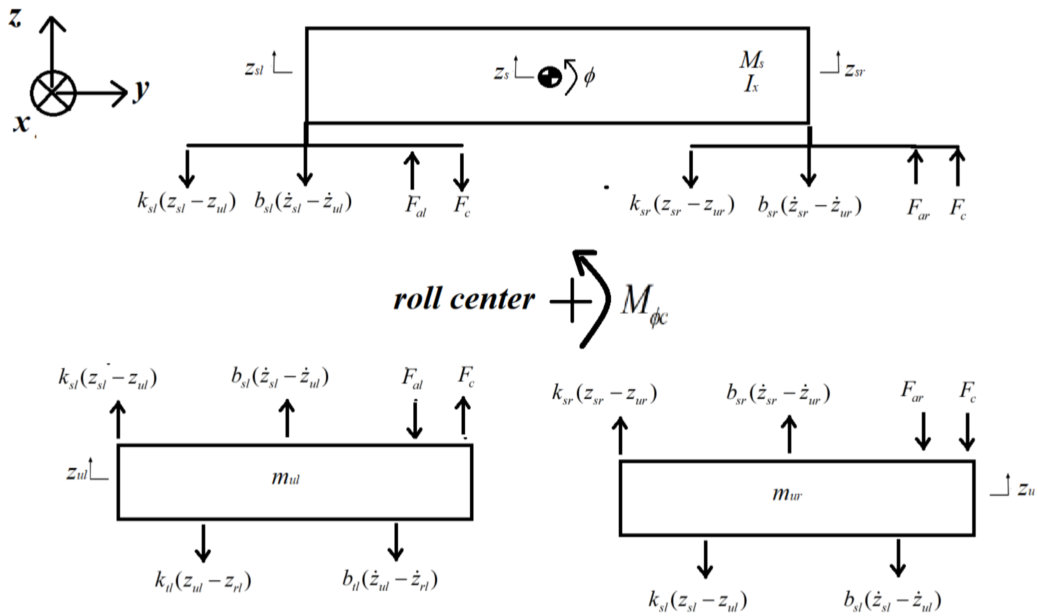

Figure 3-3 Updated free body diagram of sprung and unsprung masses

after adding the corrective roll moment, the equations are:

$$M_s\ddot{z}_s + [\sim] = F_{al} + F_{ar} - F_c + F_c = F_{al} + F_{ar} \tag{3-5}$$

$$I_x\ddot{\phi} + [\sim] = -F_{al}t_l + F_c t_l + F_{ar}t_r + F_c.t_r = -F_{al}t_l + F_{ar}t_r + M_{\phi c}$$

$$m_{ul}\ddot{z}_{ul} + [\sim] = -F_{al} + F_c = -F_{al} + M_{\phi c}/t$$

$$m_{ur}\ddot{z}_{ur} + [\sim] = -F_{ar} - F_c = -F_{ar} - M_{\phi c}/t$$

(Note that “[~]” stands for the rest of the left side of the corresponding equation. It is only used for simplicity in writing)

This means that the total active suspension forces for left and right wheels are a combination of player 1 and 2 action forces and would be calculated as follows:

$$F_{al}^t = F_{al} - F_c = F_{al} - M_{\phi c}/t \tag{3-6}$$

$$F_{ar}^t = +F_{ar} + F_c = F_{ar} + M_{\phi c}/t$$

The effects of gravity and lateral acceleration can also be added to dynamic equations. According to the figure 3-4, the distance between the center of gravity of sprung mass and roll center is labeled as h_s , and the equations of motion are updated to the following four equations:

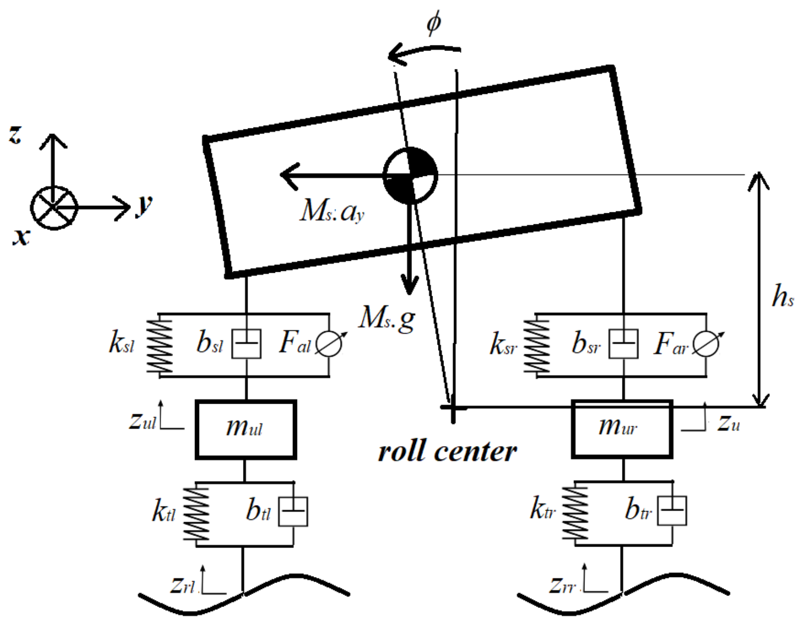

Figure 3-4: Roll plane model with roll center

$$M_s\ddot{z}_s + [\sim] = F_{al} + F_{ar} - M_s g \tag{3-7}$$

$$I_x\ddot{\phi} + [\sim] = -F_{al}t_l + F_{ar}t_r + M_{\phi c} + M_s h_s a_y + M_s g h_s \phi$$

$$m_{ul}\ddot{z}_{ul} + [\sim] = -F_{al} + M_{\phi c} / t - m_{ul} g$$

$$m_{ur}\ddot{z}_{ur} + [\sim] = -F_{ar} - M_{\phi c} / t - m_{ur} g$$

where ϕ and are roll angle and lateral acceleration of the vehicle accordingly.

considering $t_l = t_r = t/2$, all equations can be written in a standard second-order matrix form

$$\begin{bmatrix} M_s & 0 & 0 & 0 \\ 0 & I_x & 0 & 0 \\ 0 & 0 & m_{ul} & 0 \\ 0 & 0 & 0 & m_{ur} \end{bmatrix} \begin{bmatrix} \ddot{z}_s \\ \ddot{\phi} \\ \ddot{z}_{ul} \\ \ddot{z}_{ur} \end{bmatrix} + \begin{bmatrix} k_{sl}+k_{sr} & (-k_{sl}+k_{sr}).t/2 & -k_{sl} & -k_{sr} \\ (-k_{sl}+k_{sr}).t/2 & (k_{sl}+k_{sr}).t^2/4 - M_s.g.h_s & k_{sl}.t/2 & -k_{sr}.t/2 \\ -k_{sl} & k_{sl}.t/2 & k_{sl}+k_{tl} & 0 \\ -k_{sr} & -k_{sr}.t/2 & 0 & k_{sr}+k_{tr} \end{bmatrix} \begin{bmatrix} z_s \\ \phi \\ z_{ul} \\ z_{ur} \end{bmatrix}$$
$$+ \begin{bmatrix} b_{sl}+b_{sr} & (-b_{sl}+b_{sr}).t/2 & -b_{sl} & -b_{sr} \\ (-b_{sl}+b_{sr}).t/2 & (b_{sl}+b_{sr}).t^2/4 & b_{sl}.t/2 & -b_{sr}.t/2 \\ -b_{sl} & b_{sl}.t/2 & b_{sl}+b_{tl} & 0 \\ -b_{sr} & -b_{sr}.t/2 & 0 & b_{sr}+b_{tr} \end{bmatrix} \begin{bmatrix} \dot{z}_s \\ \dot{\phi} \\ \dot{z}_{ul} \\ \dot{z}_{ur} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -t/2 & t/2 \\ -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} F_{al} \\ F_{ar} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1/t \\ -1/t \end{bmatrix} M_{\phi c} +$$
$$\begin{bmatrix} 0 & 0 & 0 & -M_s \\ 0 & 0 & M_s.h_s & 0 \\ k_{tl} & 0 & 0 & -m_{ul} \\ 0 & k_{tr} & 0 & -m_{ur} \end{bmatrix} \begin{bmatrix} z_{rl} \\ z_{rr} \\ a_y \\ g \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ b_{tl} & 0 & 0 & 0 \\ 0 & b_{tr} & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{z}_{rl} \\ \dot{z}_{rr} \\ \dot{a}_y \\ \dot{g} \end{bmatrix}$$

(3-8)

The equations of motion can be summarized as follows:

$$M_{eq}\ddot{z} + K_{eq}z + C_{eq}\dot{z} = B_1u_1 + B_2u_2 + L_1w + L_2\dot{w} \tag{3-9}$$

Where

$$z = \begin{bmatrix} z_s & \phi & z_{ul} & z_{ur} \end{bmatrix}^T \tag{3-10}$$
$$w = \begin{bmatrix} z_{rl} & z_{rr} & a_y & g \end{bmatrix}^T$$

u_1 and u_2 are control action or input matrices defined as below

$$u_1 = \begin{bmatrix} F_{al} & F_{ar} \end{bmatrix}^T \tag{3-11}$$
$$u_2 = M_{\phi c}$$

and M_{eq} (inertia matrix), K_{eq} (stiffness matrix), C_{eq} (damping matrix), B_1, B_2, L_1 and L_2 are corresponding multipliers.

Damping ratios for tires can typically be ignored, which is going to cancel out the last term of the standard matrix equation.

In order to represent the equations in state-space form, the continuous states of the system are defined as:

$$x_c = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} \tag{3-11}$$

and state-space equation can be written as:

$$\dot{x}_c = A_c x_c + B_{1c} u_1 + B_{2c} u_2 + L_{1c} w \tag{3-12}$$

where

$$A_c = \begin{bmatrix} 0_{4\times4} & I_{4\times4} \\ -M_{eq}^{-1}.K_{eq} & -M_{eq}^{-1}.C_{eq} \end{bmatrix}_{8\times8}, \ B_{1c} = \begin{bmatrix} 0_{4\times2} \\ -M_{eq}^{-1}.B_1 \end{bmatrix}_{8\times2}, \ B_{2c} = \begin{bmatrix} 0_{4\times1} \\ -M_{eq}^{-1}.B_2 \end{bmatrix}_{8\times1} \tag{3-13}$$

and $L_{1c} = \begin{bmatrix} 0_{4\times4} \\ -M_{eq}^{-1}.L_1 \end{bmatrix}_{8\times4}$

So far, the lateral acceleration a_y is modeled as a disturbance. In order to consider it as one of the states or an auxiliary state, a yaw model is required. A yaw model describes the vehicle's lateral dynamic behavior and includes the control action of the third player (corrective yaw moment: M_b). The simplest yaw model is the classic bicycle model shown in figure 3-5 [4] that describes the vehicle's lateral and yaw motion.

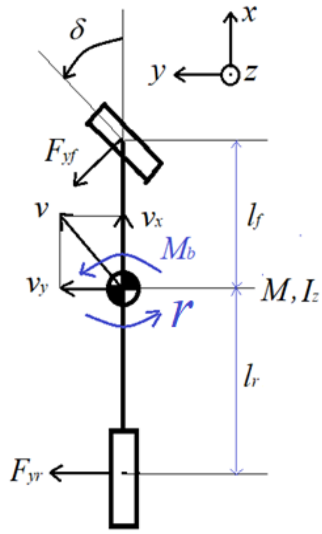

Figure 3-5: Yaw plane bicycle model [4]

The nomenclature for the figure above is as follows:

δ_f : front steering angle (in radians)

M: vehicle total mass (in kilograms)

M_s : sprung mass (in kilograms)

v_x, v_y : longitudinal/lateral velocity of vehicle (in meters per second)

α_f, α_r : tire slip angles of the front and rear wheels in radians

r : yaw rate (in radian per second)

F_{yf} , F_{yr} : front and rear lateral forces being built by the tire.

The lateral motion equation is:

$$M.a_y - M_s.h_s.\ddot{\phi} = F_{yf} + F_{yr}$$

lateral acceleration can be written as:

$$a_y = \dot{v}_y + r.v_x \tag{3-14}$$

lateral velocity can be driven as a state by combining the equations above:

$$M.\dot{v}_y = F_{yf} + F_{yr} + M_s.h_s.\ddot{\phi} - M.\mathrm{r}.v_x \tag{3-15}$$

There are different tire models available in the literature. For simplicity, this thesis's control model uses a linear tire model with no saturation limit. A more complicated tire model is presented in Chapter 7 that is used for the simulation model, which has a closer behavior to a real tire. There are different types of nonlinear tire models in which the amount of lateral force is saturated and related to the amount of normal tire force and friction coefficient (for example, Pacejka model or Dugoff model [4]).

In the linear tire model, lateral forces are proportional to their corresponding slip angles.

$$\begin{aligned} F_{yf} &= -C_f.\alpha_f = -(\frac{C_f}{v_x})v_y - (\frac{C_f.l_f}{v_x})\mathrm{r} + C_f.\delta_f \\ F_{yr} &= -C_r.\alpha_r = -(\frac{C_r}{v_x})v_y - (\frac{C_r.l_r}{v_x})\mathrm{r} \end{aligned} \tag{3-16}$$

where

l_f, l_r : Distance from CG to front/rear axle (in meters).

C_f, C_r: Cornering stiffness of front/rear tire (in Newton per radian).

the lateral motion equation can be rearranged as:

$$M.\dot{v}_y = -(\frac{C_f + C_r}{v_x}).v_y + (\frac{C_r.l_r - C_f.l_f}{v_x} - M.v_x)\mathrm{r} + M_s.h_s.\ddot{\phi} + C_f.\delta_f \quad (3\text{-}17)$$

yaw motion is described by:

$$I_x.\dot{r} + I_{zx}.\ddot{\phi} - I_{xy}.\dot{\phi}^2 = F_{yf}.l_f - F_{yr}.l_r + M_{zc} \quad (3\text{-}18)$$

where

I_x , I_z, I_{zx} and I_{xy} are roll, yaw, and combined moments of inertia ($kg.m^2$) and M_{zc} (or M_b *in figure 3-5)* is the control action that applies corrective yaw moment to the system.

In order to avoid coupling and make the state equations linear, higher derivative terms (like $M_s.h_s.\ddot{\phi}$ and $I_{zx}.\ddot{\phi}$) and higher power order of state derivatives (like $I_{xy}.\dot{\phi}^2$) can be ignored as the vehicle is symmetric.

By applying the linear tire model, the yaw equation can be written as:

$$I_z.\dot{r} = (\frac{C_r.l_r^2 - C_f.l_f^2}{v_x})v_y - \frac{C_r.l_r^2 + C_f.l_f^2}{v_x} + C_f.l_f.\delta_f + M_{zc} \quad (3\text{-}19)$$

3.2.1 Combined lateral and roll model:

In the roll model presented before, the parameter a_y was considered as a disturbance. In order to combine the roll model with the yaw model, the general equation for the roll model is required:

$$I_x.\ddot{\phi} + I_{xz}.\dot{\mathrm{r}} - I_{yz}.\mathrm{r}^2 + C_\phi.\dot{\phi} + K_\phi.\phi = -F_{al}.t_l + F_{ar}.t_r + M_{\phi c} + M_s.h_s.a_y + M_s.g.h_s.\phi \quad (3\text{-}20)$$

where C_ϕ and $K\phi$ are roll damping and stiffness coefficients that are the sum of the multipliers of $\dot{\phi}$ and ϕ in the previous roll model.

Knowing $a_y = \dot{v}_y + r.v_x$ lateral acceleration can be omitted from the equation above. The terms $I_{xz}.\dot{\mathrm{r}}$ and $I_{yz}.\mathrm{r}^2$ can also be ignored for simplicity and linearity, and the linear equation for the roll is rearranged as follows:

$$I_x.\ddot{\phi} = -C_\phi.\dot{\phi} - K_\phi.\phi - F_{al}.t_l + F_{ar}.t_r + M_{\phi c} + M_s.h_s.\dot{v}_y + M_s.h_s.v_x.\mathrm{r} + M_s.g.h_s.\phi \tag{3-21}$$

by deriving $\dot{v}_y$ from the lateral motion equation and inserting it in the roll equation, the new roll equation emerges:

$$(I_x - \frac{M_s^2.h_s^2}{M})\ddot{\phi} = -C_\phi.\dot{\phi} - (K_\phi + M_s.g.h_s)\phi - F_{al}.t_l + F_{ar}.t_r + M_{\phi c} + (\frac{M_s.h_s(C_f + C_r)}{M.v_x})v_y + (\frac{M_s.h_s(C_r.l_r - C_f.l_f)}{M.v_x})\mathrm{r} + (\frac{M_s.h_s.C_f}{M})\delta_f \tag{3-22}$$

The steering angle, which is the human driver's input, can be treated as a disturbance. The desired yaw rate γ_d can be considered as one of the states and be modeled as a first-order system based on the steering angle and time constant τ:

$$r_d = \left(\frac{K_r}{\tau s + 1}\right).\delta_f \tag{3-23}$$

where K_r is the steady-state yaw rate gain and is calculated as below according to Rajamani [4].

$$K_r = \frac{C_f.C_r(l_f + l_r).v_x}{C_f.C_r(l_f + l_r)^2 + M.v_x^2(l_r.C_r - l_f.C_f)} \tag{3-24}$$

the first-order model can be written as the following dynamic equation.

$$\dot{r}_d = -\frac{1}{\tau}.\mathrm{r}_d + \frac{K_r}{\tau}.\delta_f \tag{3-25}$$

a controller can be designed to regulate the desired yaw rate error below:

$$e_r = r - r_d \tag{3-26}$$

[note that in sections 3-6, the desired yaw rate is estimated as a linear function ($r_d = K_r.\delta_{human}$) instead of the first-order system. This estimation is used for all the simulations in this thesis. Both approaches are acceptable.]

This new driven set of equations lets us define the states, control inputs, and disturbance matrices of the system as follows:

New States: $X_c = [z_s, \phi, z_{ul}, z_{ur}, \dot{z}_s, \dot{\phi}, \dot{z}_{ul}, \dot{z}_{ur}, v_y, \mathrm{r}, \mathrm{r}_d]^T$

Control inputs: $U = [u_1; u_2; u_3]$ where u_1 , u_2 and u_3 are control actions of players 1,2 and 3 and are defined as:

$u_1 = [F_{al} \quad F_{ar}]^T$, $u_2 = M_{\phi c}$, $u_3 = M_{zc}$

Disturbance: $W = [z_{rl}, z_{rr}, \delta_f, g]^T$

The continuous state-space equation of the combined system is written as:

$$\dot{X}_c = A_c X_c + B_c U + L_c W \tag{3-27}$$

matrices in the equation above are calculated as follows:

$$A_c = \begin{bmatrix} 0_{4\times4} & I_{4\times4} & 0_{4\times3} \\ -M_{eq}^{-1}K_{eq} & -M_{eq}^{-1}C_{eq} & G_1 \\ 0_{3\times4} & 0_{3\times4} & G_2 \end{bmatrix}_{11\times11} \tag{3-28}$$

$B_c = [B_{1c}, B_{2c}, B_{3c}]_{11\times 4}$ where B_{1c}, B_{2c} and B_{3c} are corresponding multipliers for the three players' control actions and are defined as:

$$\underset{11\times 4}{B_c} U_{4\times 1} = \underset{11\times 2}{B_{1c}} \underset{2\times 1}{u_1} + \underset{11\times 1}{B_{2c}} u_2 + \underset{11\times 1}{B_{2c}} u_3 \tag{3-29}$$

$$B_{1c} = \begin{bmatrix} 0_{4\times 2} \\ M_{eq}^{-1} B_1 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{11\times 2}, B_{2c} = \begin{bmatrix} 0_{4\times} \\ M_{eq}^{-1} B_2 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{11\times 1} \text{ and}$$

$$B_{3c} = \begin{bmatrix} 0_{4\times 1} \\ 0_{4\times 1} \\ 0 \\ 1/I_z \\ 0 \end{bmatrix}_{11\times 1}$$

$$L_c = \begin{bmatrix} & & 0_{4\times 4} & \\ & & M_{eq}^{-1} L_1 & \\ & & 0_{1\times 4} & \\ 0 & 0 & \dfrac{C_f . l_f}{I_z} & 0 \\ 0 & 0 & \dfrac{K\gamma}{\tau} & 0 \end{bmatrix}_{11\times 4}$$

M_{eq}, K_{eq}, C_{eq}, G_1, G_2, B_1 , B_2 and L_1 are presented as below:

$$M_{eq} = \begin{bmatrix} M_s & 0 & 0 & 0 \\ 0 & I_x - \dfrac{M_s^2 h_s^2}{M} & 0 & 0 \\ 0 & 0 & m_{ul} & 0 \\ 0 & 0 & 0 & m_{ur} \end{bmatrix}_{4\times 4}, \tag{3-30}$$

$$K_{eq} = \begin{bmatrix} k_{sl}+k_{sr} & (-k_{sl}+k_{sr}).t/2 & -k_{sl} & -k_{sr} \\ (-k_{sl}+k_{sr}).t/2 & (k_{sl}+k_{sr}).t^2/4 - M_s.g.h_s & k_{sl}.t/2 & -k_{sr}.t/2 \\ -k_{sl} & k_{sl}.t/2 & k_{sl}+k_{tl} & 0 \\ -k_{sr} & -k_{sr}.t/2 & 0 & k_{sr}+k_{tr} \end{bmatrix}_{4\times 4}$$

,

$$C_{eq} = \begin{bmatrix} b_{sl}+b_{sr} & (-b_{sl}+b_{sr}).t/2 & -b_{sl} & -b_{sr} \\ (-b_{sl}+b_{sr}).t/2 & (b_{sl}+b_{sr}).t^2/4 & b_{sl}.t/2 & -b_{sr}.t/2 \\ -b_{sl} & b_{sl}.t/2 & b_{sl}+b_{tl} & 0 \\ -b_{sr} & -b_{sr}.t/2 & 0 & b_{sr}+b_{tr} \end{bmatrix}_{4\times 4},$$

$$L_1 = \begin{bmatrix} 0 & 0 & 0 & -M_s \\ 0 & 0 & \dfrac{M_s.h_s.C_f}{M} & 0 \\ k_{tl} & 0 & 0 & 0 \\ 0 & k_{tr} & 0 & 0 \end{bmatrix}_{4\times 4},$$

$$G_1 = \begin{bmatrix} 0 & 0 & 0 \\ \dfrac{-M_s.h_s(C_f+C_r)}{M.v_x(I_x - \dfrac{M_s^2 h_s^2}{M})} & \dfrac{M_s.h_s(C_r.l_r + C_f.l_f)}{M.v_x(I_x - \dfrac{M_s^2 h_s^2}{M})} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{4\times 3},$$

$$G_2 = \begin{bmatrix} -\dfrac{C_f+C_r}{M.v_x} & \dfrac{C_r.l_r - C_f.l_f}{M.v_x} - v_x & 0 \\ \dfrac{C_r.l_r^2 - C_f.l_f^2}{I_z.v_x} & -\dfrac{C_r.l_r^2 + C_f.l_f^2}{I_z.v_x} & 0 \\ 0 & -\dfrac{1}{\tau} & 0 \end{bmatrix}_{3\times 3}$$

$$B_1 = \begin{bmatrix} 1 & 1 \\ -t/2 & t/2 \\ -1 & 0 \\ 0 & 1 \end{bmatrix}_{4\times 2}, \quad B_2 = \begin{bmatrix} 0 \\ 1 \\ 1/t \\ -1/t \end{bmatrix}_{4\times 1}$$

3.3 Control model for the game between active steering and corrective roll moment

The equations introduced in previous sections can be used to derive the linear simplified control models for investigating the game between active steering and corrective roll moment:

Assume the state vector and control inputs described as follows:

$$x = [\phi, \dot{\phi}, v_y, r]' \tag{3-31}$$

Steering angle : $u_1 = \delta$

Corrective roll moment: $u_2 = M_{\phi c}$

The state-space representation can be written as below from the previous section equations

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (I_x + M_s h_s^2) & -M_s h_s & 0 \\ 0 & -M_s h_s & M & 0 \\ 0 & 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{v}_y \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ M_s g h_s - K & -C & 0 & M_s h_s V_x \\ 0 & 0 & -(C_f + C_r)\mu / V_x & (C_r l_r - C_f l_f)\mu / V_x - M V_x \\ 0 & 0 & (C_r l_r - C_f l_f)\mu / V_x & (C_r l_r^2 + C_f l_f^2)\mu / V_x \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v_y \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ C_f \mu \\ C_f l_f \mu \end{bmatrix} \delta + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} M_{\varphi c}$$

(3-32)

where K and C are roll stiffness and damping coefficients for passive suspensions and are calculated from the left and right wheels' suspension system using the formulas below:

$$\begin{aligned} K &= (K_{sl} + K_{sr})(t^2 / 4), \\ C &= (B_{sl} + B_{sr})(t^2 / 4) \end{aligned} \tag{3-32}$$

The state-space representation is the form below

$$E\dot{x} = Ux + V_1 u_1 + V_2 u_2 \tag{3-33}$$

which can be rearranged into standard form by defining matrices A, B_1 and B_2 as follows:

$$A = E^{-1}U, B_1 = E^{-1}V_1, B_2 = E^{-1}V_2 \tag{3-34}$$

$$\dot{x} = Ax + B_1u_1 + B_2u_2$$

This control model is used for solving the game between active steering and corrective roll moment. For tracking, yaw rate error can be chosen as the tracking goal to be minimized. In order to improve the tracking goal sometimes, the lateral error and yaw angle error can also be added to objective functions. By adding the lateral position and yaw angle to the state vector, the new matrices E U V_1 and V_2 are defined as follows.

$$x = [\mathrm{y}, v_y, \psi, \mathrm{r}, \phi, \dot{\phi}]' \tag{3-35}$$

$$E = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & M & 0 & 0 & 0 & -M_s h_s \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & I_z & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -M_s h & 0 & 0 & 0 & (I_x + M_s h_s{}^2) \end{bmatrix},$$

$$U = \begin{bmatrix} 0 & 1 & V_x & 0 & 0 & 0 \\ 0 & -(C_f + C_r)\mu / V_x & 0 & (C_r l_r - C_f l_f)\mu / V_x - MV_x & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & (C_r l_r - C_f l_f)\mu / V_x & 0 & (C_r l_r{}^2 + C_f l_f{}^2)\mu / V_x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & M_s h_s V_x & M_s g h_s - K & -C \end{bmatrix}$$

,

$$V_1 = \begin{bmatrix} 0 \\ C_f \mu \\ 0 \\ C_f l_f \mu \\ 0 \\ 0 \end{bmatrix}, V_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

3.4 Roll plane linear control model (vertical dynamics)

The roll plane control model can investigate the game between active suspension forces and the corrective roll moment. The model presented in the model development section can be simplified, considering the state vector and control inputs below:

$$X_c = [z_s, \phi, z_{ul}, z_{ur}, \dot{z}_s, \dot{\phi}, \dot{z}_{ul}, \dot{z}_{ur}]^T \tag{3-36}$$

Corrective roll moment: $u_1 = M_{\phi c}$

Active suspension forces: $u_2 = [F_{al}, F_{ar}]$

The state-space representation structure of this dynamic model is similar to the previous section

$$E\dot{x} = Ux + V_1 u_1 + V_2 u_2 \tag{3-37}$$

$$A = E^{-1}U, B_1 = E^{-1}V_1, B_2 = E^{-1}V_2$$

$$\dot{x} = Ax + B_1 u_1 + B_2 u_2$$

in which matrices E, U, V_1 and V_2 are defined as follows.

$$
E=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & M_s & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & I_x & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & m_{ul} & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{ur}
\end{bmatrix},
$$

$$
U=\begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\
-(K_{sl}+K_{sr}) & (K_{sr}-K_{sl})t/2 & K_{sl} & K_{sr} & -(\mathrm{B}_{sl}+B_{sr}) & (\mathrm{B}_{sr}-B_{sl})t/2 & B_{sl} & B_{sr}\\
(K_{sr}-K_{sl})t/2 & -(K_{sl}+K_{sr})\mathrm{t}^2/4-M_s g h_s & -K_{sl}t/2 & K_{sr}t/2 & (\mathrm{B}_{sr}-B_{sl})t/2 & -(\mathrm{B}_{sl}+B_{sr})\mathrm{t}^2/4 & -B_{sl}t/2 & B_{sr}t/2\\
K_{sl} & -K_{sl}t/2 & -(K_{sl}+K_{tl}) & 0 & B_{sl} & -B_{sl}t/2 & -(\mathrm{B}_{sl}+B_{tl}) & 0\\
K_{sr} & K_{sr}t/2 & 0 & -(K_{sr}+K_{tr}) & B_{sr} & B_{sr}t/2 & 0 & -(\mathrm{B}_{sr}+B_{tr})
\end{bmatrix}
$$

,

$$
V_1=\begin{bmatrix} 0\\ 0\\ 0\\ 0\\ 0\\ 1\\ 1/t\\ -1/t \end{bmatrix}, V_2=\begin{bmatrix} 0 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 1 & 1\\ -t/2 & t/2\\ -1 & 0\\ 0 & -1 \end{bmatrix}
$$

(3-38)

3.5 Rollover index

The Rollover Index is a variable in vehicle safety that determines if the vehicle is encountering rollover. Yedavalli [27] introduced the "unified rollover index "(RI) for each axle of the long vehicles as merit to determine if the tires are leaving the ground and the vehicle is close to experiencing rollover. In his works, RI is defined as the vertical load difference between the inside and outside tires, normalized by the total load, which is the vehicle's weight. According to Yedavalli, the RI is calculated as below:

$$RI = \frac{F_{z,o} - F_{z,i}}{F_{z,o} + F_{z,i}} = \frac{F_{z,o} - F_{z,i}}{mg} = \frac{2[m_s(\dot{v}_y + v_x r - h_s\ddot{\phi})h_r + K\phi + C\dot{\phi}]}{mgt} \tag{3-39}$$

where g is gravity, m_s (or $m2$ in figure 3-6) is sprung mass, and m is the vehicle's total mass, which is the summation of the sprung mass and unsprung masses. K and C are torsional spring stiffness and damper coefficients. In the figure below, h_r (or h_R in the figure) is the roll axis height, and h_s (or h in the figure) represents the distance from the sprung mass center of gravity to the roll axis.

As long as the value of RI stays in the range (-1,1), the vehicle does not roll over. When one of the wheels lift, the value of RI becomes 1 or -1, and that vehicle is encountering rollover.

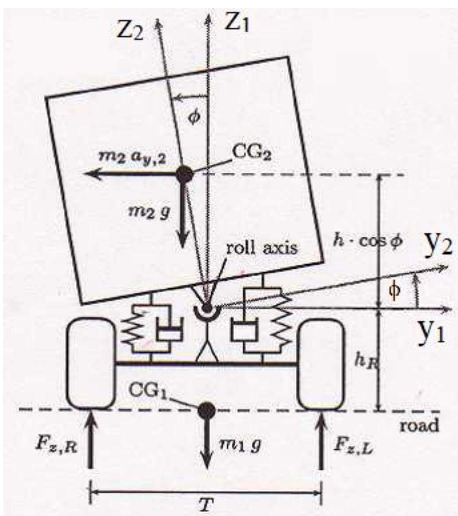

Figure 3-6: Roll plane model used in Yedavalli [27]

As RI is a linear function of the system's states (roll angle, roll rate, yaw rate, and lateral velocity), it is defined as a linear output below to be minimized in *LQR* or output regulation framework in optimal control in Yedavalli [27].

$$RI = Cx + Du \tag{3-40}$$

The control input can either be the steering input or the corrective roll moment. In this thesis, the roll index is not a part of the optimal cost function, but it is still observed and plotted as a parameter to determine the rollover performance of the controllers.

3.6 Plant model

This section demonstrates the plant model dynamic equations that are driven by combining the control models in previous sections. Feedback gains for each scenario are calculated for each optimal design, and then they can be tested on a plant model with a higher degree of freedom and complexity. This dissertation's plant model is written in an ordinary differential equations form (ODEs), based on the model development section with some modifications explained in this section. The plant system's state vector is a combination of ten main states and six sub-states used for post-processing purposes in the simulation.

Main states: $z_s, \phi, z_{ul}, z_{ur}, \dot{z}_s, \dot{\phi}, \dot{z}_{ul}, \dot{z}_{ur}, v_y, \mathrm{r}$

Sub-states: $\psi, \mathrm{X}_G, Y_G, \psi_{des}, \mathrm{X}_{des}, \mathrm{Y}_{des}$

State vector: $X = [z_s, \phi, z_{ul}, z_{ur}, \dot{z}_s, \dot{\phi}, \dot{z}_{ul}, \dot{z}_{ur}, v_y, \mathrm{r}, \psi, \mathrm{X}_G, Y_G, \psi_{des}, \mathrm{X}_{des}, \mathrm{Y}_{des}]^T$

The description of the model variables is provided in Table 3-1 and Table 3-2.

Table 3-1: States and sub-states

Main states	Description	Sub-states	Description

$x_1 = z_s$	Sprung mass vertical position	$x_{11} = \psi$	Vehicle yaw angle
$x_2 = \phi$	Vehicle roll angle	$x_{12} = X_G$	Vehicle global x position
$x_3 = z_{ul}$	Left unsprung mass vertical position	$x_{13} = Y_G$	Vehicle global y position
$x_4 = z_{ur}$	Right unsprung mass vertical position	$x_{14} = \psi_{des}$	Vehicle desired yaw angle
$x_5 = \dot{z}_s$	Sprung mass vertical velocity	$x_{15} = X_{des}$	Vehicle desired x position
$x_6 = \dot{\phi}$	Vehicle roll rate	$x_{16} = Y_{des}$	Vehicle desired y position
$x_7 = \dot{z}_{ul}$	Left unsprung mass vertical velocity		
$x_8 = \dot{z}_{ur}$	Right unsprung mass vertical velocity		
$x_9 = v_y$	Vehicle lateral velocity		
$x_{10} = r$	Vehicle yaw rate		

Table 3-2: Input variables

Input variables	Description
δ_H	Human driver front steering input (disturbance)
z_{rl}	Left wheel road input position (disturbance)
z_{rr}	Right wheel road input position (disturbance)
$\dot{z}_{rl}$	Left wheel road input velocity (disturbance)
$\dot{z}_{rr}$	Right wheel road input velocity (disturbance)
$u_1 = \delta_c$	Active steering control input
$u_2 = M_{yc}$	Corrective yaw moment control input
$u_3 = M_{\phi c}$	Corrective roll moment control input
$u_4 = [F_{al}, F_{ar}]$	Left and right active suspension forces control input vector

Here are the ordinary differential equations for the plant model main states:

$$\begin{aligned}
\dot{x}_1 &= \dot{z}_s = x_5 \\
\dot{x}_2 &= \dot{\phi} = x_6 \\
\dot{x}_3 &= \dot{z}_{ul} = x_7 \\
\dot{x}_4 &= \dot{z}_{ur} = x_8
\end{aligned} \tag{3-41}$$

$$\dot{x}_5 = \ddot{z}_s = \frac{1}{M_s}\begin{bmatrix} F_{al} + F_{ar} - M_s g + \ldots \\ -(k_{sl} + k_{sr})z_s - (\mathrm{b}_{sl} + b_{sr})\dot{z}_s + (k_{sl} - k_{sr})\frac{t}{2}\phi + (\mathrm{b}_{sl} - b_{sr})\frac{t}{2}\phi + \ldots \\ k_{sl}z_{ul} + k_{sr}z_{ur} + b_{sl}\dot{z}_{ul} + b_{sr}\dot{z}_{ur} \end{bmatrix}$$

(Note that the "…" means that the rest of the equation continues in the next line)

The derivatives of the roll rate and lateral velocity are shown as $\dot{x}_6 = \ddot{\phi}, \dot{x}_9 = \dot{v}_y$, and their dynamic behaviors are described in the two governing equations below:

$$I_x\ddot{\phi} - M_s h_s \dot{v}_y = \begin{bmatrix} k_{sl}\frac{t}{2}(z_s - \frac{t}{2}\phi - z_{ul}) + b_{sl}\frac{t}{2}(\dot{z}_s - \frac{t}{2}\dot{\phi} - \dot{z}_{ul}) + \ldots \\ \ldots - k_{sr}\frac{t}{2}(z_s + \frac{t}{2}\phi - z_{ur}) - b_{sr}\frac{t}{2}(\dot{z}_s + \frac{t}{2}\dot{\phi} - \dot{z}_{ur}) + \ldots \\ \ldots - F_{al}\frac{t}{2} + F_{ar}\frac{t}{2} + M_{\phi c} + M_s g h_s \phi + M_s h_s V_x r \end{bmatrix} \tag{3-42}$$

$$-M_s h_s \ddot{\phi} + M\dot{v}_y = \left[-\frac{(C_f + C_r)\mu}{V_x} v_y + (\frac{(C_r l_r - C_f l_f)\mu}{V_x} - MV_x)r + C_f\mu(\delta_H + \delta_c) \right]$$

The equations presented above are in the form of a system of two linear equations that can be shown in the following structure:

$$\begin{cases} a\ddot{\phi} + b\dot{v}_y = e \\ c\ddot{\phi} + d\dot{v}_y = f \end{cases} \tag{3-43}$$

This system of linear equations can be solved by Matlab using "*linsolve*" command

$$\begin{bmatrix} \dot{x}_6 \\ \dot{x}_9 \end{bmatrix} = \begin{bmatrix} \ddot{\phi} \\ \dot{v}_y \end{bmatrix} = linsolve\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}, \begin{bmatrix} e \\ f \end{bmatrix} \right) \tag{3-44}$$

Where:

$$a=(I_x+M_s h_s^2),\mathrm{b}=-M_s h_s, e=\begin{bmatrix} k_{sl}\frac{t}{2}(z_s-\frac{t}{2}\phi-z_{ul})+b_{sl}\frac{t}{2}(\dot{z}_s-\frac{t}{2}\dot{\phi}-\dot{z}_{ul})+... \\ ...-k_{sr}\frac{t}{2}(z_s+\frac{t}{2}\phi-z_{ur})-b_{sr}\frac{t}{2}(\dot{z}_s+\frac{t}{2}\dot{\phi}-\dot{z}_{ur})+... \\ ...-F_{al}\frac{t}{2}+F_{ar}\frac{t}{2}+M_{\phi c}+M_s g h_s\phi+M_s h_s V_x r \end{bmatrix} \tag{3-45}$$

$$c=-M_s h_s, d=M, f=\left[-\frac{(C_f+C_r)\mu}{V_x}v_y+(\frac{(C_r l_r-C_f l_f)\mu}{V_x}-MV_x)r+C_f\mu(\delta_H+\delta_c)\right]$$

Tire normal forces only exist when the tire is under compression. This means that if the tire deflection state in figure 3-7 $(z_{us}-z_r)$ is negative, then the normal tire force exists, and for all other values of the tire deflection state, the normal tire force is equal to zero.

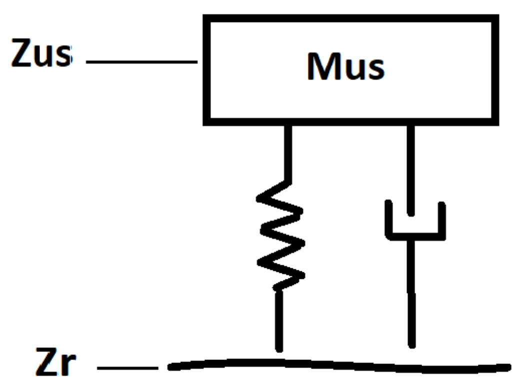

Figure 3-7: Tire deflection presentation for a quarter car model

The Matlab function "subplus" is used to apply this concept in the model, which only returns the positive part of the parameter. The figure below plots y=subplus(x).

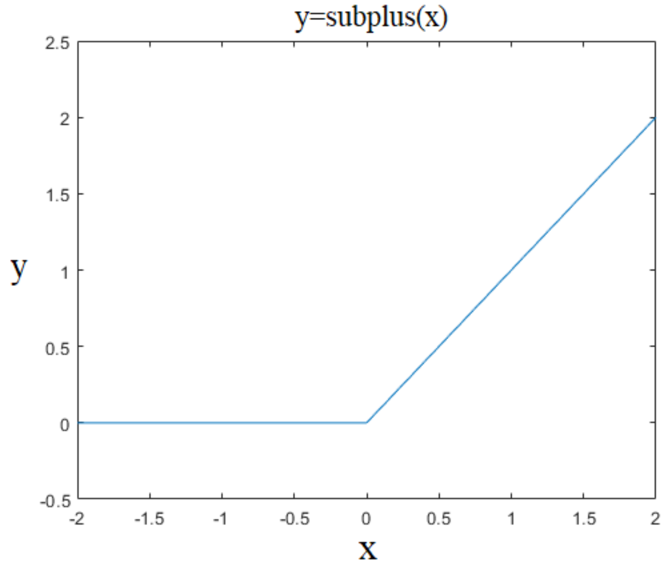

Figure 3-8: Matlab help- subplus function

using "*subplus*" function, the left and right unsprung mass accelerations ($\ddot{z}_{ul}$ and $\ddot{z}_{ur}$) are calculated as follows.

$$p = sign(subplus(z_{rl} - z_{ul})) \tag{3-46}$$

$$\dot{x}_7 = \ddot{z}_{ul} = \frac{1}{m_{ul}} \begin{bmatrix} -F_{al} + \dfrac{M_{\phi c}}{t} - m_{ul} g \\ -pk_{tl}(z_{ul} - z_{rl}) - pb_{tl}(\dot{z}_{ul} - \dot{z}_{rl}) \\ +k_{sl}(z_s + \dfrac{t}{2}\phi - z_{ul}) + b_{sl}(\dot{z}_s + \dfrac{t}{2}\dot{\phi} - \dot{z}_{ul}) \end{bmatrix},$$

$$q = sign(subplus(z_{rr} - z_{ur}))$$

$$\dot{x}_8 = \ddot{z}_{ur} = \frac{1}{m_{ur}} \begin{bmatrix} -F_{ar} + \dfrac{M_{\phi c}}{t} - m_{ur} g \\ -qk_{tr}(z_{ur} - z_{rr}) - qb_{tr}(\dot{z}_{ur} - \dot{z}_{rr}) \\ +k_{sr}(z_s + \dfrac{t}{2}\phi - z_{ur}) + b_{sr}(\dot{z}_s + \dfrac{t}{2}\dot{\phi} - \dot{z}_{ur}) \end{bmatrix}$$

and the yaw acceleration is

$$\dot{x}_{10} = \dot{r} = \frac{1}{I_z}\left[\left(\frac{(C_r l_r^{\ 2} - C_f l_f^{\ 2})}{V_x}\right)\mathrm{v}_y - \left(\frac{(C_r l_r^{\ 2} + C_f l_f^{\ 2})}{V_x}\right)\mathrm{r} + C_f l_f \mu(\delta_H + \delta_c) + \mathrm{M}_{yc} \right] \tag{3-47}$$

Ordinary differential equations for sub-states are presented as follows:

$$\dot{x}_{11} = \dot{\psi} = r = x_{10} \tag{3-48}$$

Global positions are calculated by integrating the equations below

$$\begin{aligned} \dot{x}_{12} &= \dot{X}_G = V_x \,\mathrm{Cos}(\psi) - v_y Sin(\psi) \\ \dot{x}_{13} &= \dot{Y}_G = V_x \,\mathrm{Sin}(\psi) + v_y \,\mathrm{Cos}(\psi) \end{aligned} \tag{3-49}$$

According to Rajamani (chapter 8.2.3) [4], the desired yaw rate can be obtained by multiplication of steady-state yaw rate gain and human driver steering input

$$\dot{x}_{14} = \dot{\psi}_{desired} = K_r \delta_H \tag{3-50}$$

where steady-state yaw rate gain is:

$$K_r = \frac{2C_f C_r (l_f + l_r) V_x}{2C_f C_r (l_f + l_r)^2 + M V_x^{\ 2} (\mathrm{C}_r\, l_r - C_f l_f)} \tag{3-51}$$

and the desired velocities are

$$\begin{aligned} \dot{x}_{15} &= \dot{X}_{des} = V_x \,\mathrm{Cos}(\psi_{des}) \\ \dot{x}_{16} &= \dot{Y}_{des} = V_x \,\mathrm{Sin}(\psi_{des}) \end{aligned} \tag{3-52}$$

3.7 Plant model test

In this section, a simple step steer maneuver is chosen to verify the plant model developed in section 3.6. In this maneuver, according to figure 3-9, the human driver engages a 7.5 deg (π/24 rad) steering angle at 2 s and continues this constant steering angle to make a loop. In the passive maneuver, no controller is involved, so the total steering input is the human driver steering input. Based on the table presented in the appendix, all the vehicle parameters are chosen. The vehicle's forward velocity is assumed to be constant (V_x=20 m/s), and the sampling time for the simulation is 0.01 s.

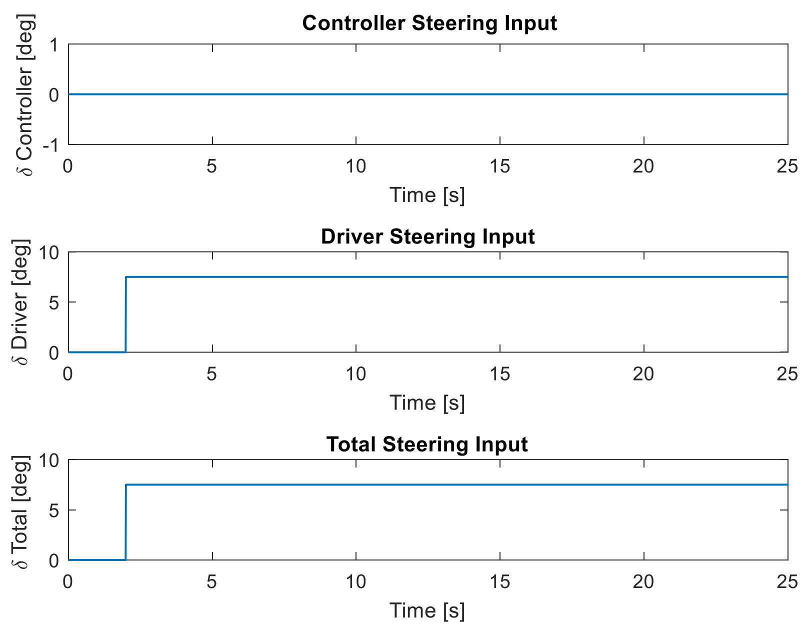

Figure 3-9: Steering angles of human driver and controller for 7.5 deg step steer maneuver

The desired trajectory and the passive maneuver for the plant model are shown in Figure 3-10.

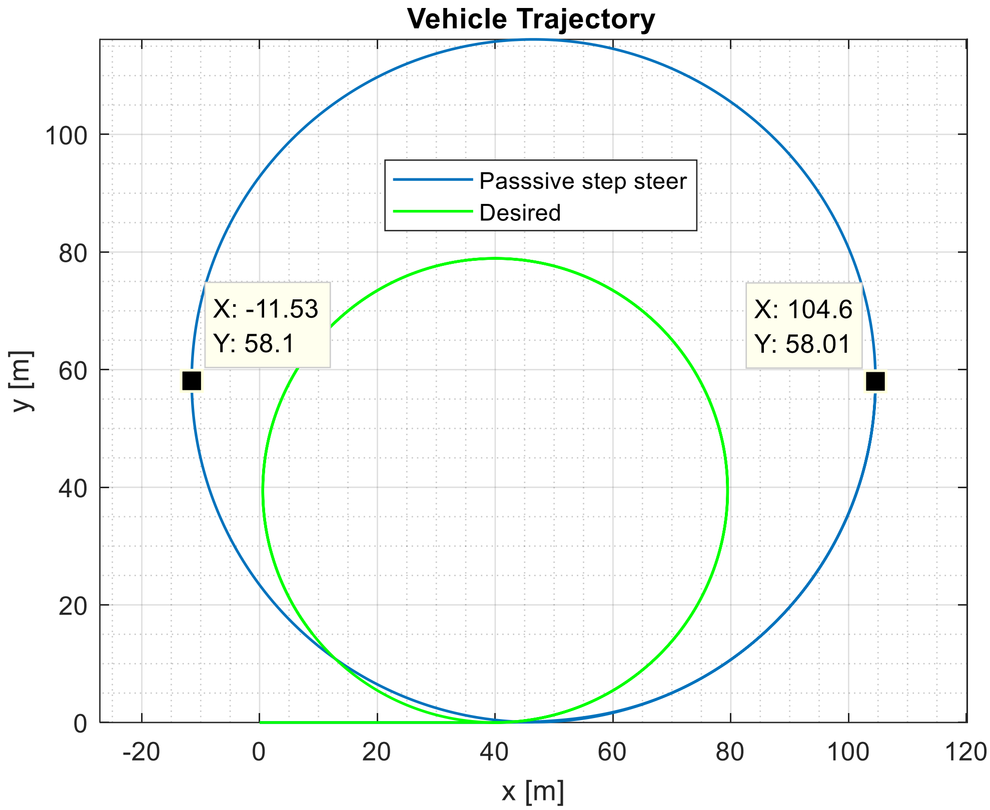

Figure 3-10: Vehicle trajectory for 7.5 deg step steer maneuver

The desired path radius of curvature (the kinematic radius) is 40 m, but the vehicle is showing understeer behavior and encountering a circle-like path with a radius of approximately 29.8 m.

$$R_1 \simeq (\frac{104.6-(-11.53)}{2}) \simeq 58\,\text{m} \tag{3-53}$$

The lateral acceleration for this maneuver is shown in Figure 3-11.

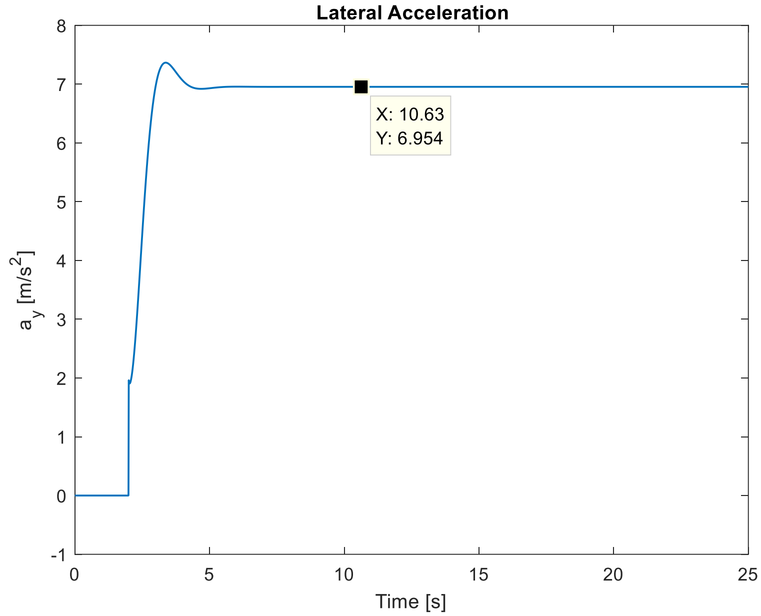

Figure 3-11: Lateral acceleration for 7.5 deg step steer maneuver

Figure 3-11 shows that the lateral acceleration settles down at the approximate amount of 6.9 m/s^2. Knowing that the longitudinal velocity for this maneuver is constant and equivalent to 20 m/s, one can calculate the radius of curvature as below:

$$R_2 = \frac{V_x^{\ 2}}{a_y} \simeq (\frac{20^2}{6.954}) \simeq 57.52\,\mathrm{m} \tag{3-54}$$

The yaw rate for the plant model settles down at the amount of 0.3477, according to Figure 3-12. The radius of curvature for this amount is as follows:

$$R_3 = \frac{V_x}{r} \simeq (\frac{20}{0.3477}) \simeq 57.52\,\mathrm{m} \tag{3-55}$$

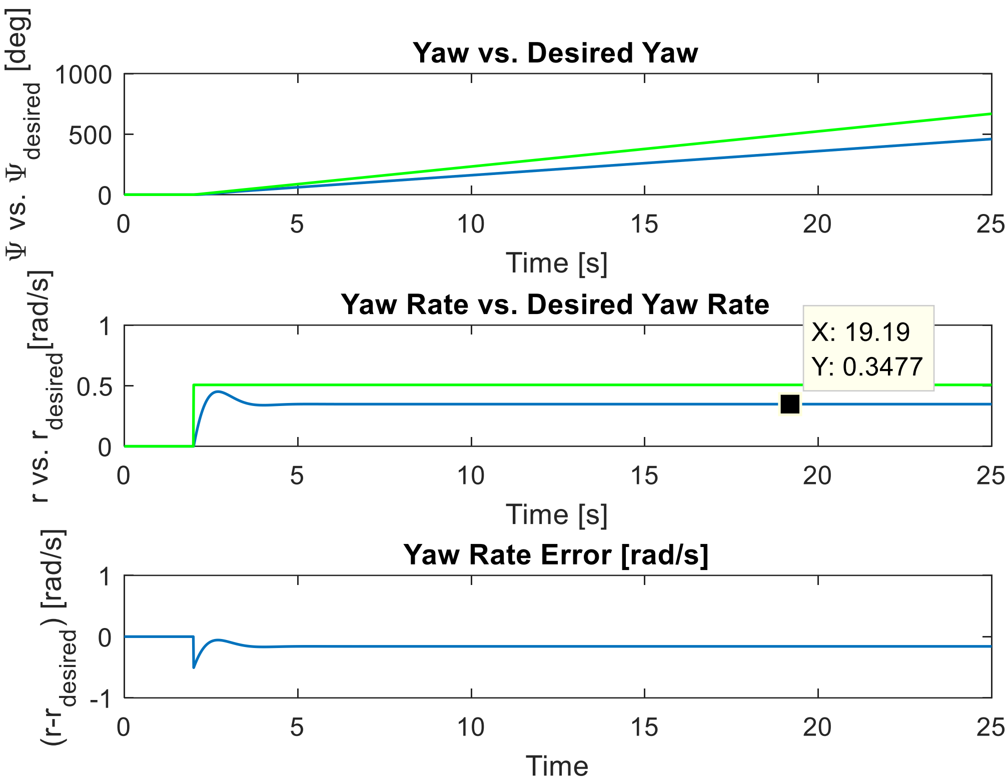

Figure 3-12: Yaw plots for 7.5 deg step steer maneuver

The values of radius of curvature recorded from lateral acceleration and yaw rate (R_2 and R_3) are identical, and they are almost close to the calculated value from the trajectory plot (R_1) with a 0.83% error, which is an acceptable error for this approximation. The following plots show the rest of the vehicle responses for this passive maneuver.

Figure 3-13 shows the roll angle, roll rate, roll index, and yaw angle of the vehicle. It is shown that the vehicle has a steady-state roll angle of less than 4 degrees. According to Figure 3-14 both front and rear tires generate the lateral force of around 5 kN.

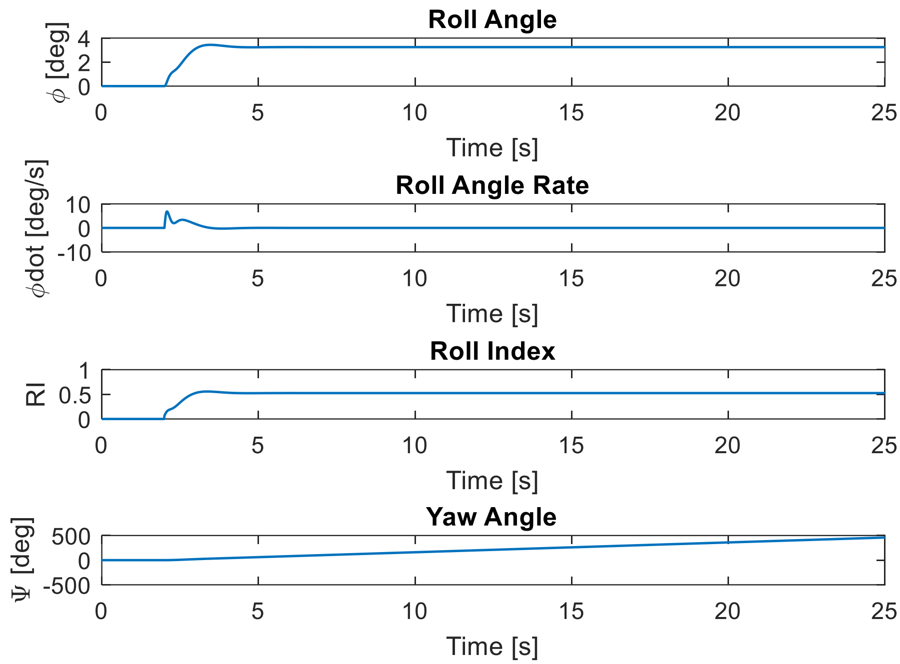

Figure 3-13: Vehicle roll angle, roll rate, roll index, and yaw angle for step steer maneuver

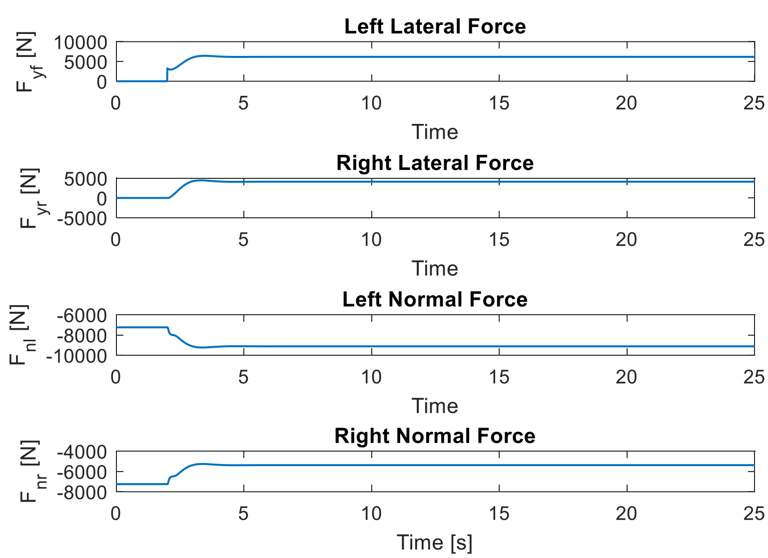

Figure 3-14: Tire lateral and normal forces for step steer maneuver

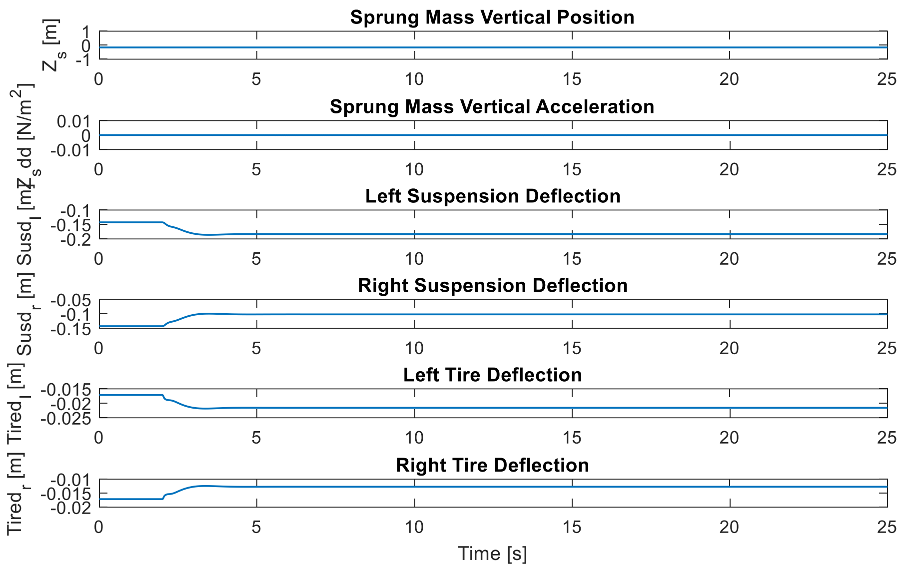

Figure 3-15: Vertical dynamic for step steer maneuver

According to Figure 3-15, both suspension and tire have steady-state deflections and zero vertical acceleration.

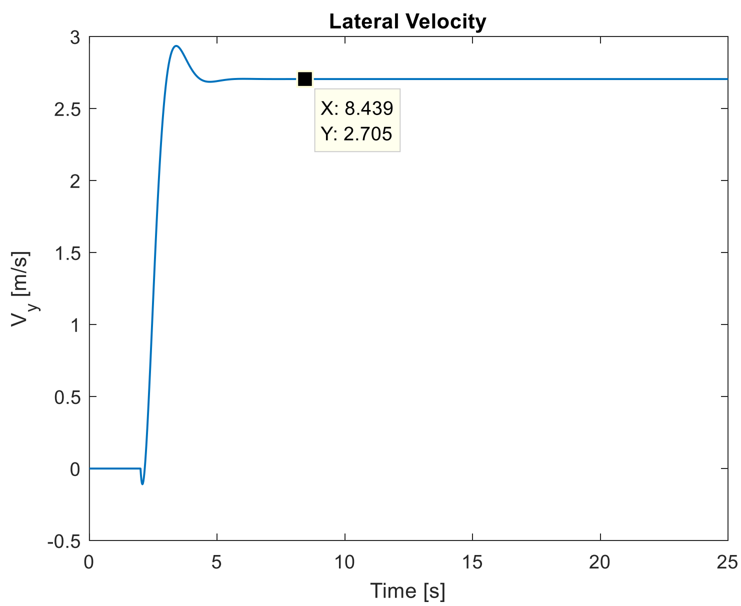

Figure 3-16: Vehicle lateral velocity for step steer maneuver

According to Figure 3-16, the vehicle's lateral velocity settles down at -2.705 m/s. The front and rear slip angle can be calculated in radians as follows:

$$\alpha_f = \delta_f - \frac{V_y + l_f \times r}{V_x} \simeq \pi / 24 - \frac{-2.705 + 1.12 \times 0.3477}{20} \simeq 0.2467 \tag{3-56}$$
$$\alpha_r = -\frac{V_y - l_r \times r}{V_x} \simeq -\frac{-2.705 - 1.68 \times 0.3477}{20} \simeq 0.1644$$

These calculated values match the amount for the plant model simulation in the figure

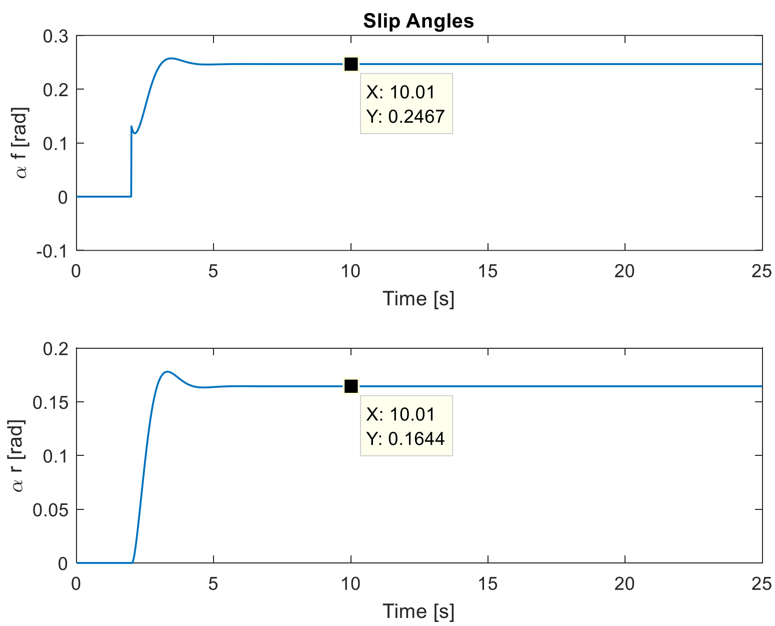

Figure 3-17: Slip angles for the step steer maneuver

3.8 Chapter summary

Two types of models (two control models and one plant model) are presented in this chapter. The free-body diagrams are drawn for the roll plane model and bicycle model. The equations of motions are written for these two models, and the combined model is presented. The driven dynamic equations are used in sections 3.3 and 3.4 to derive the linear control models. In section 3.5, the roll index parameter is introduced as a factor of merit to check if the vehicle encounters rollover. In the last section, the nonlinear plant model with the linear tire model is introduced. This plant model includes all the main states and the auxiliary states that are required for postprocessing and output plots. As the normal tire forces exist only when the tire is under compression, the "*subplus*" function is introduced to the plant model, which makes it nonlinear. Control models are used in chapter 6 to calculate the optimal gains, and then the gains are used as closed-loop feedback gains with the plant model to get the simulation results. Chapter 7 adds a nonlinear tire to the plant model, where the lateral force is a nonlinear function of normal force and slip angle. As a result, this chapter's plant model lateral forces will be replaced by the new lateral forces generated by the nonlinear tire model, and the simulations results will be shown for the new nonlinear plant model.

The next step after modeling is designing the controllers. After stating the optimal solution for the single-player game, the two-player and three-player game theory is introduced in the following chapters (chapters 4 and 5). The solutions for the Nash paradigm in the following chapters are presented for two classes of optimal problems (*LQR* and *CCOR*).

4 CHAPTER 4: TWO-PLAYER GAME THEORY OPTIMAL (GTO)

4.1 Introduction to two-player paradigms (decentralized and Nash)

In a two-player differential game, each player has its own control action to minimize the player's optimal control cost function. The optimal cost functions for players can either share some mutual objectives or be independent. This function can be designed as a quadratic combination of state or state errors, outputs, and control actions. Solving the optimal cost functions together can produce minimum control actions required to minimize the errors and cost function value and lead the system to the desired states. If each player's cost function has only that player’s control action, the cost function is decentralized. Sometimes, a cost function of a player consists of another player's control actions, called the Nash paradigm (Stackelberg paradigm also has the same configuration in which one player is the leader and the other is follower, which is not studied in this thesis). The impact of each control action or parameter in the cost function can be determined by setting different weights for each parameter. It is also possible for each player to have their own individual control goals or have a combined control goal. According to the definition of the goals, players can either play a cooperative game (which means they can both have the same goal) or a non-cooperative game (which means they have their own individual goals). Four different types of formulation for cost functions are presented in this section. The decentralized paradigm can be solved independently by the LQR method for the two cost functions. The solution approach for all the Nash formulations is the same, considering that each will provide different optimal control gains due to how their cost functions are defined.

When it comes to building the cost functions for players, four paradigm structures are defined as below according to each player's goal and action and knowledge of the other player's goal and control action.

- Structure 1 (Decentralized): Each player has its own independent goal and control action and no knowledge of other players goal and control action
- Structure 2 (Nash): each player has their own independent goal and control action, has knowledge of other player action but no knowledge of other player goals
- Structure 3 (Decentralized): players have the same mutual goal (which is a combination of their independent goals) and no knowledge of each other actions
- Structure 4 (Nash): players have the same mutual goal (which is a combination of their independent goals) and knowledge of each other's actions

In section 4.2, the quadratic mathematical representation for these structures is presented, followed by how to identify if the optimal problem is the *LQR* problem or CCOR. The solution for the single-player *LQR* problem exists in all optimal control references [28], and the solution for *CCOR* is also represented by Yedavalli [27].

Section 4.3 includes the problem statement and solution for the two-player *LQR* game for both finite and infinite horizons for continuous systems. This solution algorithm for the continuous infinite horizon is adopted from the solution for discrete systems presented by Tamaddoni [22,23], which is the contribution of this thesis. Section 4.4 introduces the *CCOR* problem for the single-player based on Yedavalli's work [27]. The solution for the two-player *CCOR* is presented in section 4.5, which is also a contribution of this thesis.

4.2 Mathematical formulation

This section presents the general mathematical representations for the various structures in quadratic form.

Structure 1 (Decentralized):

$$J_1 = \frac{1}{2}\int_0^\infty [(N_1 z(t))^T Q_1 (N_1 z(t)) + u_1(t)^T R_{11} u_1(t)]dt \tag{4-1}$$

$$J_2 = \frac{1}{2}\int_0^\infty [(N_2 z(t))^T Q_2 (N_2 z(t)) + u_2(t)^T R_{22} u_2(t)]dt$$

each players' cost function contains only its control objective and control input.

Structure 2 (Nash):

$$J_1 = \frac{1}{2}\int_0^\infty [(N_1 z(t))^T Q_1 (N_1 z(t)) + u_1(t)^T R_{11} u_1(t) + u_2(t)^T R_{12} u_2(t)]dt \tag{4-2}$$

$$J_2 = \frac{1}{2}\int_0^\infty [(N_2 z(t))^T Q_2 (N_2 z(t)) + u_1(t)^T R_{21} u_1(t) + u_2(t)^T R_{22} u_2(t)]dt$$

Each player's cost function contains its control objective and control input and other players' control input.

Structure 3 (Decentralized):

$$J_1 = \frac{1}{2}\int_0^{\infty}[(N_1 z(t))^T Q_1 (N_1 z(t)) + (N_2 z(t))^T Q_2 (N_2 z(t)) + u_1(t)^T R_{11} u_1(t)]dt \tag{4-3}$$

$$J_2 = \frac{1}{2}\int_0^{\infty}[(N_1 z(t))^T Q_1 (N_1 z(t)) + (N_2 z(t))^T Q_2 (N_2 z(t)) + u_1(t)^T R_{22} u_1(t)]dt$$

Each player's cost function contains its control objective, control input, and other player control objectives.

Structure 4 (Nash):

$$J_1 = \frac{1}{2}\int_0^{\infty}[(N_1 z(t))^T Q_1 (N_1 z(t)) + (N_2 z(t))^T Q_2 (N_2 z(t)) + u_1(t)^T R_{11} u_1(t) + u_1(t)^T R_{12} u_1(t)]dt \tag{4-4}$$

$$J_2 = \frac{1}{2}\int_0^{\infty}[(N_1 z(t))^T Q_1 (N_1 z(t)) + (N_2 z(t))^T Q_2 (N_2 z(t)) + u_1(t)^T R_{21} u_1(t) + u_1(t)^T R_{22} u_1(t)]dt$$

Each player's cost function contains both control objectives and inputs.

Variable $z(t)$ represents states or errors that are being minimized and Q_i is the error weighted matrix for the i^{th} player, which is a positive semidefinite matrix and function of time. The quadratic form $(N_i z(t))^T Q_i (N_i z(t))$ is a quadratic weighted representation of the errors or states and N_i helps each cost function to select the desired states or errors required to be minimized.

$u_i(t)$ is the i^{th} player control signal which is weighted by a parameter from a positive-definite control weighted matrix $R(t)$

$$R(t)=\begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \tag{4-5}$$

There are two challenges to finding the optimal control input. The first challenge is defining cost functions based on the states and control inputs and translating them into quadratic cost functions. The second challenge is solving the optimal control problem for an infinite horizon.

This process is elaborated in the example below. According to the definition of each cost function in each paradigm, the steps presented in the example must be followed to calculate the quadratic matrix of *Q*, *R*, and *N*.

Example: define decentralized cost functions for a two-player game described as below:

Player One is the active suspension system, which is trying to minimize vertical acceleration ($\ddot{z}_c$), suspension strokes ($(z_{sl}-z_{ul})$ and $(z_{sr}-z_{ur})$), tire deflections ($(z_{ul}-z_{rl})$ and $(z_{ur}-z_{rr})$), and control efforts of the active suspension system (Fa_l and Fa_r).

Player Two is the anti-roll system trying to minimize roll angle and its first and second derivatives (ϕ, $\dot{\phi}$ and $\ddot{\phi}$) and anti-roll actuator moment (corrective roll moment $M_{\phi c}$).

According to the modeling chapter, the state space representation of the system is:

$$\dot{x}_c = A_c x_c + B_{1c}u_1 + B_{2c}u_2 + L_{1c}w \tag{4-6}$$

where states and disturbances are defined as:

$$x_c=\begin{bmatrix} z \\ \dot{z} \end{bmatrix} \tag{4-7}$$

$$z = \begin{bmatrix} z_s & \phi & z_{ul} & z_{ur} \end{bmatrix}^T$$

$$w = \begin{bmatrix} z_{rl} & z_{rr} & a_y & g \end{bmatrix}^T$$

and control actions are

$$\begin{aligned} u_1 &= \begin{bmatrix} F_{al} & F_{ar} \end{bmatrix}^T \\ u_2 &= M_{\phi c} \end{aligned} \tag{4-8}$$

According to the problem definition, cost functions for both players are written as:

$$J_1 = \int_0^\infty [\rho_1 \ddot{z}_c{}^2 + \rho_2 (z_{sl} - z_{ul})^2 + \rho_3 (z_{sr} - z_{ur})^2 + \rho_4 (z_{ul} - z_{rl})^2 + \rho_5 (z_{ur} - z_{rr})^2 + \rho_6 Fa_l{}^2 + \rho_7 Fa_r{}^2] dt$$

$$J_2 = \int_0^\infty [\rho_8 \phi^2 + \rho_9 \dot{\phi}^2 + \rho_{10} \ddot{\phi}^2 + \rho_{11} M_{\phi c}{}^2] dt$$

(4-9)

where ρ_i are weights for each quadratic parameter and are usually normalized using Bryson's rule [29, p 400]:

$$\rho_i = \frac{1}{\eta_i^2} \tag{4-10}$$

where η_i is the maximum value of the corresponding parameter that is multiplied by the weight. Notice that Bryson's rule is a factor of merit to normalize the effect of all parameters in the cost function. The weights can still be customized according to the designer's preference to intensify a specific parameter's effect in the cost function.

For each cost function, the following steps must be followed to generate the desired linear quadratic cost function

Step 1: cost function must be formulated in a format below according to the problem definition

$$J = \int_0^\infty [\rho_1 K_1{}^2 + \rho_2 K_2{}^2 + \rho_3 K_3{}^2 + ...]dt \tag{4-11}$$

where K_i is the term that is needed to be minimized (K_i can be an error, a defined parameter based on states, or a control action) and ρ_i is the corresponding weight.

Step 2: In this step, the primary cost function in step one is translated into a standard linear quadratic form. According to the formulation of the output, there are two cases:

Case 1- If the output that is being minimized *(Y)* is only a linear function of the states, the problem will turn into a state regulation problem. The cost function can be a weighted combination of two quadratic functions of states and control actions. Assume the dynamic system below:

$$\begin{aligned} \dot{X} &= AX + BU \\ Y &= CX \end{aligned} \tag{4-12}$$

Then *J* in step one can be translated into the LQR function below:

$$J = \int_0^\infty [\mathrm{Y}^T \bar{Q} Y + U^T \bar{R} U]dt = \int_0^\infty [X^T Q X + U^T R U]dt \tag{4-13}$$

where matrix *Q* is the states weighting matrix (semi-positive definite) and *R* is a positive definite matrix that applies weighting on the inputs.

$$\begin{aligned} Q &= C^T \bar{Q} C \\ R &= \bar{R} \end{aligned} \tag{4-14}$$

This case is under a class of optimal control problems called linear quadratic regulator (*LQR*). The general solution for the one-player game is presented in optimal control reference books [28], and the solution for the two-player game is presented in section 4.3.

Case 2- Sometimes, the output is a linear function of both states and control inputs:

$$Y = CX + DU \tag{4-15}$$

In that case, the defined cost function is translated to an output regulation cost function below:

$$J = \int_0^\infty [Y^T Y]dt = \int_0^\infty [(CX + DU)^T (CX + DU)]dt \tag{4-16}$$

All the K_is in the first step must be written as a function of states or control inputs, and the whole cost function in step one will be rearranged to derive the matrices of *C* and *D*.

The cost function in the first step is rearranged again to derive the final cost function formatted as below:

$$J = \int_0^\infty [X^T QX + X^T NU + U^T N^T X + U^T RU]dt \tag{4-17}$$

Matrices *Q, N,* and *R* are linear-quadratic matrices that are calculated from *C* and *D* using the following procedure:

$$\begin{aligned}(CX + DU)^T (CX + DU) &= X^T C^T CX + X^T C^T DU + U^T D^T CX + U^T D^T DU \\ &= X^T QX + X^T NU + U^T N^T X + U^T RU\end{aligned} \tag{4-18}$$

By comparing the similar terms in the presented equation, LQ matrices are derived as follows:

$$\begin{aligned}Q &= C^T C \\ N &= C^T D\end{aligned} \tag{4-19}$$

$R = D^T D$

Case-2 is often written in a general form below where the cost function is the combination of output cost and the control cost:

$$J = \int_0^\infty [Y^T \bar{Q} Y + \rho U^T \bar{R} U] dt = \int_0^\infty [(CX + DU)^T \bar{Q} (CX + DU) + \rho U^T \bar{R} U] dt$$

$$= \int_{t_0}^\infty [(X^T C^T \bar{Q} CX + 2X^T C^T \bar{Q} DU + U^T (D^T \bar{Q} D + \rho \bar{R}) U] dt = \int_0^\infty [X^T Q X + 2X^T N U + U^T R U] dt$$

(4-20)

By comparing the terms, the matrices *Q*, *N*, and *R* are derived as follows.

$$Q = C^T \bar{Q} C, \qquad (4\text{-}21)$$
$$N = C^T \bar{Q} D,$$
$$R = D^T \bar{Q} D + \rho \bar{R}$$

The matrix *N* shows the coupling between the states and inputs. This case is under a class of optimal control problems called control coupled output regulation (CCQR). The general solution for the one-player game is presented by Yedavalli [27] and shown in section 4.4, and the solution for the two-player game is presented in section 4.5.

Here is the demonstration of applying the steps to the aforementioned example of a decentralized paradigm.

$$K_1 = \ddot{z}_c = \sum_{i=1}^{8} a_{5i} x_i + b1_{51} u1_{11} + b1_{52} u1_{21} + \sum_{i=1}^{4} L_{5i} w_i \qquad K_2 = (z_{sl} - z_{ul}) = x_1 - \frac{t}{2} x_2 - x_3$$

$$K_3 = (z_{sr} - z_{ur}) = x_1 + \frac{t}{2} x_2 - x_4 \quad K_4 = (z_{ul} - z_{rl}) = x_3 - w_1 \qquad K_5 = (z_{ur} - z_{rr}) = x_4 - w_2$$

$$K_6 = Fa_l = u_{11} \qquad K_7 = Fa_r = u_{21} \qquad K_8 = \phi = x_2 \qquad K_9 = \dot{\phi} = x_6$$

$$K_{10} = \ddot{\phi} = \sum_{i=1}^{8} a_{6i} x_i + b2_{61} u2 + \sum_{i=1}^{4} L_{6i} w_i \qquad K_{11} = M_{\phi c} = u_2$$

(4-22)

a, *b1* and *b2* are parameters of matrices A, B1, and B2 in the state space equation. The cost functions J_1 and J_2 can be written as:

$$J_1 = \int_0^\infty [(C_1 X + D_1 u_1)^T (C_1 X + D_1 u_1)] dt \tag{4-23}$$

$$J_2 = \int_0^\infty [(C_2 X + C u_2)^T (C_2 X + D_2 u_2)] dt$$

where C_1, C_2, D_1 and D_2 are calculated as below:

$$C_1 = \begin{bmatrix} \sqrt{\rho_1} a_{51} & \sqrt{\rho_1} a_{52} & \sqrt{\rho_1} a_{53} & \sqrt{\rho_1} a_{54} & \sqrt{\rho_1} a_{55} & \sqrt{\rho_1} a_{56} & \sqrt{\rho_1} a_{57} & \sqrt{\rho_1} a_{58} \\ \sqrt{\rho_2} & -\sqrt{\rho_2}\frac{t}{2} & -\sqrt{\rho_2} & 0 & 0 & 0 & 0 & 0 \\ \sqrt{\rho_3} & \sqrt{\rho_3}\frac{t}{2} & 0 & -\sqrt{\rho_3} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{\rho_4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{\rho_5} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{7\times 8}$$

$$D_1 = \begin{bmatrix} \sqrt{\rho_1} b1_{51} & \sqrt{\rho_1} b1_{52} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \sqrt{\rho_6} & 0 \\ 0 & \sqrt{\rho_7} \end{bmatrix}_{7\times 2}$$

$$C_2 = \begin{bmatrix} 0 & \sqrt{\rho_8} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{\rho_9} & 0 & 0 \\ \sqrt{\rho_{10}} a_{61} & \sqrt{\rho_{10}} a_{62} & \sqrt{\rho_{10}} a_{63} & \sqrt{\rho_{10}} a_{64} & \sqrt{\rho_{10}} a_{65} & \sqrt{\rho_{10}} a_{66} & \sqrt{\rho_{10}} a_{67} & \sqrt{\rho_{10}} a_{68} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{4\times 8}$$

$$D_2 = \begin{bmatrix} 0 \\ 0 \\ \sqrt{\rho_{10}} b2_{61} \\ \sqrt{\rho_{11}} \end{bmatrix}$$

(4-24)

Q, N, and R matrices for decentralized paradigm cost functions can be calculated from C_1, C_2, D_1 and D_2.

Player 1: $Q_1 = C_1^T C_1$ $\qquad N_1 = C_1^T D_1$ (4-25)

$R_{11} = D_1^T D_1$

Player 2: $Q_2 = C_2^T C_2$ $\qquad N_2 = C_2^T D_2$

$R_{22} = D_2^T D_2$

4.3 The general solution for two-player paradigm using the LQR approach

In the beginning, the problem is defined in a finite horizon. The finite-time linear quadratic regulator's solution is presented using the same approach from the optimal control reference book [28], and the coupled differential Riccati equations are presented as a result. In the next section, the problem is defined in the infinite horizon, and all the necessary assumptions are presented to change the Differential Riccati equations (DREs) to Algebraic Riccati equations (ARE's). A numerical algorithm is then proposed to solve the ARE's together to calculate the control signals that provide the optimal solution.

4.3.1 The continuous finite-time linear quadratic formulation for a two-player game:

Consider a linear-time dynamic system below

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u_1(t) + B_2 u_2(t) \\ x(t_0) &= x_0 \end{aligned} \tag{4-26}$$

where $u_1(t)$ and $u_2(t)$ are control inputs of players one and two accordingly, and their cost functions J_1 and J_2 are defined in the format below

$$\begin{aligned} J_1 &= x(t_f)^T F_{1f} x(t_f) + \int_{t_0}^{t_f} (x^T Q_1 x + u_1{}^T R_{11} u_1 + u_2{}^T R_{12} u_2) d\tau \\ J_2 &= x(t_f)^T F_{2f} x(t_f) + \int_{t_0}^{t_f} (x^T Q_2 x + u_1{}^T R_{21} u_1 + u_2{}^T R_{22} u_2) d\tau \end{aligned} \tag{4-27}$$

where F_{if} is the terminal cost matrix $F_i(t_f)$ and is semi-positive definite like the error weighted matrix Q_i. The control weighted matrix R_{ij} is positive definite. In some literature, there is a "1/2" multiplier for each cost function, which does not influence the result of the optimal solution.

Hamiltonians [28] for both of the cost functions can be written using the co-state vector of n^{th} order $\lambda(t)$ as below:

$$H_1(x,u_1,u_2,\lambda_1) = x^T Q_1 x + u_1^{\ T} R_{11} u_1 + u_2^{\ T} R_{12} u_2 + \lambda_1^{\ T}(Ax + B_1 u_1 + B_2 u_2) \qquad (4\text{-}28)$$
$$H_2(x,u\ \ ,u_2,\lambda_2) = x^T Q_2 x + u_1^{\ T} R_{21} u_1 + u_2^{\ T} R_{22} u_2 + \lambda_2^{\ T}(Ax + B_1 u_1 + B_2 u_2)$$

From now all the optimal parameters (including optimal state, optimal control input, and optimal co-state) are shown with the star superscript (*).

Optimal control inputs u_1^* and u_2^* can be driven from a Hamiltonian function using the control relation below:

$$\frac{\partial H_1}{\partial u_1} = 0 \rightarrow R_{11} u_1^* + B_1^T \lambda_1 = 0 \rightarrow u_1^* = -R_{11}^{\ -1} B_1^T \lambda_1 \qquad (4\text{-}29)$$
$$\frac{\partial H_2}{\partial u_2} = 0 \rightarrow R_{22} u_2^* + B_2^T \lambda_2 = 0 \rightarrow u_2^* = -R_{22}^{\ -1} B_2^T \lambda_2$$

from the feedback law, co-state functions can be written as a function of states.

$$\lambda_1 = P_1(t) x^*(t) \qquad (4\text{-}30)$$
$$\lambda_2 = P_2(t) x^*(t)$$

where $P_1(t)$ and $P_2(t)$ are yet to be determined. Comparing the boundary conditions for the terminal condition, one can write that $P(t_f) = F(t_f)$

$$\lambda_1(t_f) = F_{1f} x^*(t_f) \qquad (4\text{-}31)$$
$$\lambda_2(t_f) = F_{2f} x^*(t_f)$$

Control inputs can be rearranged using the feedback law

$$u_1^* = -R_{11}{}^{-1} B_1^T \lambda_1 = -R_{11}{}^{-1}(t) B_1^T(t) P_1(t) x^*(t) = -R_{11}{}^{-1} B_1^T P_1 x^* \qquad (4\text{-}32)$$
$$u_2^* = -R_{22}{}^{-1} B_2^T \lambda_2 = -R_{22}{}^{-1} B_2^T P_2 x^*$$

Notice that all parameters are a function of time, and for simplicity, x(t) is shown as x, and likewise for the other parameters.

Optimal state equations can be calculated from the Hamiltonian function using the relation below, which is the same for both Hamiltonians since the state equation is the same for both.

$$\dot{x}^*(t) = \frac{\partial H_i}{\partial \lambda_i} \rightarrow \dot{x}^* = Ax^* - B_1 R_{11}{}^{-1} B_1^T P_1 x^* - B_2 R_{22}{}^{-1} B_2^T P_2 x^* \qquad (4\text{-}33)$$

The optimal co-state equation can also be calculated from the Hamiltonian function:

$$\lambda_i^* = -\frac{H_i}{\partial x} - \frac{H_i}{\partial u_j} \frac{\partial u_j{}^*}{\partial x} \qquad (4\text{-}34)$$

The equation above must be calculated for both players' co-states. Calculations for Player One are shown below, and the calculation for Player Two can be written based on the similarity.

$$-\frac{H_1}{\partial x} = -Q_1 x^* - A^T \lambda_1^* \qquad (4\text{-}35)$$

$$-\frac{H_1}{\partial u_2} \frac{\partial u_2{}^*}{\partial x} = (-R_{12} u_2 - B_2^T \lambda_1) \frac{\partial u_2{}^*}{\partial x} = (-R_{12} u_2 - B_2^T \lambda_1)(R_{22}{}^{-1} B_2^T P_2) =$$
$$[-R_{12}(R_{22}{}^{-1} B_2^T P_2 x^*) - B_2^T P_1 x^*](R_{22}{}^{-1} B_2^T P_2) = -P_2 B_2 R_{22}^{-1} [R_{12} R_{22}{}^{-1} B_2^T P_2 - B_2^T P_1] x^* =$$
$$[-P_2 B_2 R_{22}^{-1} R_{12} R_{22}{}^{-1} B_2^T P_2 + P_2 B_2 R_{22}^{-1} B_2^T P_1] x^* = [-P_2 S_{12} P_2 + P_2 S_2 P_1] x^*$$

Parameters S_1, S_2, S_{12} ,and S_{21} are introduced as below to simplify the writing of the equations:

$$S_1 = B_1 R_{11}^{-1} B_1^T \qquad S_2 = B_2 R_{22}^{-1} B_2^T \tag{4-36}$$

$$S_{12} = B_2 R_{22}^{-1} R_{12} R_{22}{}^{-1} B_2^T$$

$$S_{21} = B_1 R_{11}^{-1} R_{21} R_{11}{}^{-1} B_1^T$$

from the equations above, the co-state equation for Player One can be written as:

$$\dot{\lambda}_1^* = -Q_1 x^* - A^T \lambda_1^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* = -Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* \tag{4-37}$$

A derivative of optimal co-state can also be calculated from the equation below:

$$\lambda_1(t) = P_1(t) x^*(t) \rightarrow \dot{\lambda}_1^* = \dot{P}_1 x^* + P_1 \dot{x}^* \tag{4-38}$$

Comparing the two equations for co-states:

$$-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* = \dot{P}_1 x^* + P_1 \dot{x}^* \tag{4-39}$$

$\dot{x}^*$ can be entered into the equation above from the optimal state equation

$$-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* = \dot{P}_1 x^* + P_1 (A x^* - B_1 R_{11}{}^{-1} B_1^T P_1 x^* - B_2 R_{22}{}^{-1} B_2^T P_2 x^*)$$

$$-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* = \dot{P}_1 x^* + P_1 A x^* - P_1 B_1 R_{11}{}^{-1} B_1^T P_1 x^* - P_1 B_2 R_{22}{}^{-1} B_2^T P_2 x^*$$

$$-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* = \dot{P}_1 x^* + P_1 A x^* - P_1 S_1 P_1 x^* - P_1 S_2 P_2 x^*$$

(4-40)

by rearranging the equation above and omitting the multiplier x^*, the differential Riccati equation for Player One can be derived with the terminal boundary condition below

$$-\dot{P}_1 = P_1A + A^T P_1 + Q_1 - P_1S_1P_1 - P_1S_2P_2 - P_2S_2P_1 + P_2S_{12}P_2 \tag{4-41}$$
$$P_1(t_f) = P_{1f} = F_{1f}$$

Using the same procedure for the second player, the coupled differential Riccati equation for the two-player game can be written as:

$$\dot{P}_1 + P_1A + A^T P_1 + Q_1 - P_1S_1P_1 - P_1S_2P_2 - P_2S_2P_1 + P_2S_{12}P_2 = 0 \tag{4-42}$$
$$\dot{P}_2 + P_2A + A^T P_2 + Q_2 - P_2S_1P_1 - P_2S_2P_2 - P_1S_1P_2 + P_1S_{21}P_1 = 0$$
$$P_1(t_f) = P_{1f} = F_{1f}$$
$$P_2(t_f) = P_{2f} = F_{2f}$$

By solving the equations above for P_1 and P_2, co-state values of λ_1 and λ_2 are obtained, and optimal control inputs of u_1^* and u_2^* are calculated based on the states feedback.

4.3.2 Infinite time LQR system for the two-player game:

If the terminal time in previous finite equations turns into infinity, the infinite time LQR is reviewed below

$$\dot{x}(t) = Ax(t) + B_1u_1(t) + B_2u_2(t) \tag{4-43}$$
$$x(t_0) = x_0$$

$$J_1 = \int_0^\infty (x^T Q_1 x + u_1{}^T R_{11} u_1 + u_2{}^T R_{12} u_2) d\tau$$

$$J_2 = \int_0^\infty (x^T Q_2 x + u_1{}^T R_{21} u_1 + u_2{}^T R_{22} u_2) d\tau$$

where u(t) is not constrained, Q is a symmetric semi-positive definite matrix, and R is a symmetric positive definite matrix.

The system must be completely controllable. Using results similar to the finite-horizon problem P_i is still the solution of Riccati equation with the boundary condition:

$$P(t_f) = F(t_f) = 0 \tag{4-44}$$

the optimal control inputs can be obtained as:

$$\begin{aligned} u_1^* &= -R_{11}{}^{-1} B_1^T \hat{P}_1 x^* \\ u_2^* &= -R_{22}{}^{-1} B_2^T \hat{P}_2 x^* \end{aligned} \tag{4-45}$$

where

$$\hat{P}_i(t) = \lim_{t_f \to \infty} \{P(t)\} = \overline{P}_i \tag{4-46}$$

$\overline{P}_i$ is a positive definite, symmetric, constant matrix that turns the coupled differential Riccati equations into coupled algebraic Riccati equations below:

$$\begin{aligned} \frac{d\overline{P}_1}{dt} &= 0 = \overline{P}_1 A + A^T \overline{P}_1 + Q_1 - \overline{P}_1 S_1 \overline{P}_1 - \overline{P}_1 S_2 \overline{P}_2 - \overline{P}_2 S_2 \overline{P}_1 + \overline{P}_2 S_{12} \overline{P}_2 \\ \frac{d\overline{P}_2}{dt} &= 0 = \overline{P}_2 A + A^T \overline{P}_2 + Q_2 - \overline{P}_2 S_1 \overline{P}_1 - \overline{P}_2 S_2 \overline{P}_2 - \overline{P}_1 S_1 \overline{P}_2 + \overline{P}_1 S_{21} \overline{P}_1 \end{aligned} \tag{4-47}$$

Optimal control signals can be obtained from the equations below after solving the coupled ARE's for $\overline{P}_1$ and $\overline{P}_2$

$$\begin{aligned} u_1^* &= -R_{11}{}^{-1} B_1^T \overline{P}_1 x^* \\ u_2^* &= -R_{22}{}^{-1} B_2^T \overline{P}_2 x^* \end{aligned} \tag{4-48}$$

the optimal state x^* can be calculated by solving the state equation using the initial boundary condition

$$\dot{x}^* = [A - B_1 R_{11}{}^{-1} B_1^T \bar{P}_1 - B_2 R_{22}{}^{-1} B_2^T \bar{P}_2] x^* \qquad (4\text{-}49)$$
$$x^*(t_0) = x_0$$

and optimal cost functions can be obtained using the equations below:

$$J_1 = \frac{1}{2} x^*(t)^T \bar{P}_1 x^*(t) \qquad (4\text{-}50)$$
$$J_2 = \frac{1}{2} x^*(t)^T \bar{P}_2 x^*(t)$$

4.3.3 Solving algebraic Riccati equations (ARE's)

The Matlab function (*care*) is used to solve the algebraic Riccati equations. According to the Matlab library function description, this function can solve the continuous algebraic Riccati equation below and return the unique solution for *P*, the gain matrix $G = R^{-1}(B^T PE + S^T)$, and closed-loop eigenvalues vector *L*.

$$A^T PE + E^T PA - (E^T PB + S) R^{-1} (B^T PE + S^T) + Q = 0 \qquad (4\text{-}51)$$

This equation can also be simplified by omitting R and S (R=I, S=0) and setting E to the default identity matrix value.

All the Riccati equations that can be written in the format above can be solved using the "care" function if their associated Hamiltonian eigenvalues are far from the imaginary axis. There is a finite stabilizing solution for P. This function can be used straightforwardly to solve the Riccati equation for decentralized paradigms. Still, a new solution needs to be introduced due to the existence of coupling between the Riccati equations in the Nash paradigms. The proposed solution in this dissertation is a numerical based solution and is described by the following steps:

Step 1:

Initiate P_1^C and P_2^C by solving the following decoupled ARE's:

$$\begin{aligned} A^T P_1^C + P_1^C A - P_1^C S_{11} P_1^C + Q_1 = 0 \\ A^T P_2^C + P_2^C A - P_2^C S_{22} P_2^C + Q_2 = 0 \end{aligned} \quad (4\text{-}52)$$

P_1^C and P_2^C can be calculated in Matlab using the code below

$$\begin{aligned} P_1^C = care(A, B_1, Q_1, R_{11}) \\ P_2^C = care(A, B_2, Q_2, R_{22}) \end{aligned} \quad (4\text{-}53)$$

Step 2:

Compute the strong solution P_1^{C+1} and P_2^{C+1} from the following decoupled ARE's:

$$\begin{aligned} P_1^{C+1}(A - S_{22}P_2^C) + (A - S_{22}P_2^C)^T P_1^{C+1} - P_1^{C+1} S_{11} P_1^{C+1} + Q_1 + P_2^C S_{12} P_2^C = 0 \\ P_2^{C+1}(A - S_{11}P_1^C) + (A - S_{11}P_1^C)^T P_2^{C+1} - P_2^{C+1} S_{22} P_2^{C+1} + Q_2 + P_1^C S_{21} P_1^C = 0 \end{aligned} \quad (4\text{-}54)$$

The equations below can be used to simplify the format of the Ricatti equation to the acceptable format for the "*care*" function:

$$\begin{aligned} A_1' = (A - S_{22}P_2^C), \\ Q_1' = Q_1 + P_2^C S_{12} P_2^C, \\ A_2' = (A - S_{11}P_1^C), \\ Q_2' = Q_2 + P_1^C S_{21} P_1^C \end{aligned} \quad (4\text{-}55)$$

The Riccati equations can be rearranged as:

$$\begin{aligned} P_1^{C+1} A_1' + A_1'^T P_1^{C+1} - P_1^{C+1} S_{11} P_1^{C+1} + Q_1' = 0 \\ P_2^{C+1} A_2' + A_2'^T P_2^{C+1} - P_2^{C+1} S_{22} P_2^{C+1} + Q_2' = 0 \end{aligned} \quad (4\text{-}56)$$

P_1^{C+1} and P_2^{C+1} can be calculated in Matlab using the code below:

$$\begin{aligned} P_1^{C+1} = care(A_1', B_1, Q_1', R_{11}) \\ P_2^{C+1} = care(A_2', B_2, Q_2', R_{22}) \end{aligned} \quad (4\text{-}57)$$

Step 3:

This step updates P_i^C and investigates the convergence of P_i^{C+1} and P_i^C using the Euclidean norm to stop the iteration. The algorithm for convergence is presented as a "while" loop below:

While $\left\| P_i^{C+1} - P_i^C \right\| \geq \varepsilon$ (4-58)

$P_i^C = P_i^{C+1}$

Repeat Step 2

End

Note that the method presented in this section can only be applied to the problems for which their cost functions can be written in the form below:

$$\dot{x}(t) = Ax(t) + B_1 u_1(t) + B_2 u_2(t) \qquad (4\text{-}59)$$
$$x(t_0) = x_0$$

$$J_1 = \int_0^\infty (x^T Q_1 x + u_1{}^T R_{11} u_1 + u_2{}^T R_{12} u_2) d\tau$$

$$J_2 = \int_0^\infty (x^T Q_2 x + u_1{}^T R_{21} u_1 + u_2{}^T R_{22} u_2) d\tau$$

This is Case 1 (according to section 4.3), which is the class of *LQR* problems where the outputs that are being minimized are linear functions of only the states, meaning:

$$y_1 = C_1 x \qquad (4\text{-}50)$$
$$y_2 = C_2 x$$

The states weighting matrices Q_1 and Q_2 are calculated as below based on the output matrices:

$$Q_1 = C_1^T C_1 \qquad (4\text{-}51)$$
$$Q_2 = C_2^T C_2$$

The weighting matrices R_{11}, R_{12}, R_{21} and R_{22} are positive definite and are tuned arbitrarily by the controller designer.

If the outputs are the linear combination of both states and control inputs (Case 2), the cost functions can not be written in the form presented in this section. The coupling between the states and control inputs makes a new form of a cost function that belongs to the control coupled output regulation class. This class is introduced and solved in the next section.

$$Case-2(two-player): \qquad (4\text{-}52)$$
$$y_1 = C_1 x + D_{11} u_1 + D_{12} u_2$$
$$y_2 = C_2 x + D_{21} u_1 + D_{22} u_2$$

4.4 Introduction to Control Coupled Output Regulation method (*CCOR*):

4.4.1 State derivative-induced (control coupled) output regulation problem

In this section, an extension of the popular method *LQR* is presented and studied in a general mathematical form (*CCOR*). The solution for one-player *CCOR* is presented in Yadevalli [27] and is reviewed in this section. The optimal approach to build the cost function in this dissertation is to define the desired errors or parameters as output and regulate them using the *LQR* method. Assume the state space representation of the system presented as below, where the output is the linear combination of both states and control inputs:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned} \quad x(t_0)\text{: is given} \tag{4-53}$$

x represents the states of the system, and y as an output can be written as a linear combination of all the states and control inputs. The existence of coupling between states and control input in the output formula makes the output regulation different than pure state regulation.

The cost function for the output regulation can be designed as a linear combination of quadratic output cost J_y and control cost J_u

$$J = J_y + \rho J_u \tag{4-54}$$

$$J = \int_{t_0}^{\infty} [y^T(\tau)\bar{Q}y(\tau) + \rho u^T(\tau)\bar{R}u(\tau)]d\tau$$

This function can also be written based on states and control input using the equation

$$y = Cx + Du \tag{4-55}$$

$$\begin{aligned}J &= \int_{t_0}^{\infty} [y^T(\tau)\bar{Q}y(\tau) + \rho u^T(\tau)\bar{R}u(\tau)]d\tau \\ &= \int_{t_0}^{\infty} [(Cx(\tau) + Du(\tau))^T \bar{Q}(Cx(\tau) + Du(\tau)) + \rho u^T(\tau)\bar{R}u(\tau)]d\tau \\ &= \int_{t_0}^{\infty} [(x^T(\tau)C^T\bar{Q}Cx(\tau) + 2x^T(\tau)C^T\bar{Q}Du(\tau) + u^T(\tau)(D^T\bar{Q}D + \rho\bar{R})u(\tau)]d\tau\end{aligned}$$

Note that by comparing the equations above with each other J_y and J_u can be shown by

$$J_y = \int_{t_0}^{\infty} [(x^T(\tau)C^T\bar{Q}Cx(\tau) + 2x^T(\tau)C^T\bar{Q}Du(\tau) + u^T(\tau)D^T\bar{Q}Du(\tau)]d\tau \tag{4-56}$$

$$J_u = \int_{t_0}^{\infty} [u^T(\tau)\bar{R}u(\tau)]d\tau$$

By defining new weighting matrices *Q*, *N*, and *R* as follows

$$\begin{aligned} Q &= C^T\bar{Q}C, \\ N &= C^T\bar{Q}D, \\ R &= D^T\bar{Q}D + \rho\bar{R} \end{aligned} \tag{4-57}$$

The cost function can be written in the form below, where state weighting matrix *Q* is symmetric positive semidefinite, control weighting matrix *R* is symmetric positive definite, and weight matrix *N* is the coupling weight between states and the input.

$$J = \int_{t_0}^{\infty} [x^T(\tau)Qx(\tau) + 2x^T(\tau)Nu(\tau) + u^T(\tau)Ru(\tau)]d\tau \tag{4-58}$$

The solution for the coupled output regulation problem is briefly reviewed here [18].

Define a new control input as:

$$\bar{u}(\tau) = u(\tau) + R^{-1}N^T x(\tau) \tag{4-59}$$

The state equation can be written based on $\bar{u}(\tau)$, and a new state matrix $A' = A - BR^{-1}N^T$ can be defined as a result.

(Note that for writing simplification, the (τ) is omitted from equations)

$$\begin{aligned} \dot{x} &= Ax + Bu = Ax + B(\bar{u} - R^{-1}N^T x) = (A - BR^{-1}N^T)x + B\bar{u} \\ &= A'x + B\bar{u} \end{aligned} \tag{4-60}$$

It can also be written that

$$\bar{u}^T R\bar{u} = (u + R^{-1}N^T x)^T R(u + R^{-1}N^T x) = u^T Ru + 2x^T Nu + x^T NR^{-1}N^T x \tag{4-61}$$

Using the equation above and defining new weighting matrix $Q' = Q - NR^{-1}N^T$, the cost function can be written based on Q' and A':

$$J = \int_{t_0}^{\infty} [x^T Q x + 2x^T N u + u^T R u] d\tau \tag{4-62}$$
$$J = \int_{t_0}^{\infty} [\bar{u}^T R \bar{u} - x^T N R^{-1} N^T x + x^T Q x] d\tau = \int_{t_0}^{\infty} [x^T (Q - NR^{-1}N^T) x + \bar{u}^T R \bar{u}] d\tau$$
$$J = \int_{t_0}^{\infty} [x^T Q' x + \bar{u}^T R \bar{u}] d\tau$$

The optimal control law for the new control input $\bar{u}$ is given by $\bar{u}^* = -R^{-1} B^T P x(\tau)$ where P is the solution for the Algebraic Riccati Equation (ARE).

$$PA' + A'^T P - PBR^{-1} B^T P + Q' = 0 \tag{4-63}$$

The final optimal control law for the main control input *u* is given by

$$u^* = \bar{u}^* - R^{-1} N^T x(\tau) = -R^{-1} B^T P x(\tau) - R^{-1} N^T x(\tau) \tag{4-64}$$
$$u^* = -R^{-1}(B^T P + N^T) x(\tau) = -K^* x(\tau)$$
$$K^* = R^{-1}(B^T P + N^T)$$

optimal control gain K^* can be broken into two distinct gains

$$K^* = K^*_P + K^*_N \tag{4-65}$$
$$K^*_P = R^{-1} B^T P,$$
$$K^*_N = R^{-1} N^T.$$

The first gain K^*_P is similar to uncoupled LQR gain but the second gain K^*_N is the by-product of the coupling between the states and control input that is presented by matrix *N*. Association of the gain K^*_P with Ricatti matrix P contributes to the stability of closed-loop system where K^*_N gain contributes more to keeping the output cost J_y small and in some cases equal to zero (by using output zeroing method [27]). In other words, the gain K^*_P plays the stabilization role and K^*_N

separately accomplishes minimization. Having the conceptual understating of these gains' roles can be helpful to understand the differences between the coupled and uncoupled LQR framework [27].

In the design process, it is assumed that pair *(A, B)* is controllable and pair *(A, C)* is observable. The Closed Loop system is asymptotically stable if

- The pair (*A*, *B*) is stabilizable;
- $R = R^T \succ 0$ and $Q' \geq 0$; and
- The pair (Q', A') has no unobservable mode on the imaginary axis.

Note that *N* is embedded in the coupling weighting matrix Q' and is involved in ARE and control gain K^*.

4.5 Solving two-player game using control coupled output regulation (Two-Player CCOR)

In the previous section, the problem and solution for single-player CCOR were presented. Following the same concept, the solution for two-player CCOR is presented.

Each player's cost function is defined as a combination of quadratic output costs that the player wants to regulate, plus the player's quadratic control cost and weighted quadratic control cost. The output cost function is weighted and normalized by matrix $\bar{Q}$. The control cost function is normalized by $\bar{R}$ and then multiplied by ρ. Bryson's rule is used in weighting matrices $\bar{Q}$ and $\bar{R}$ to normalize the values in the cost function. The multiplier ρ is used as a design parameter that the designer can tune to get the desired performance from the controller.

The player i^{th} cost function in an N player game is defined as below:

$$J_i = Jy_i + \sum_{j=1}^{N} \rho_{ij} Ju_j = \int_{t=t_0}^{\infty} [y_i^T Q_i y_i + \sum_{j=1}^{N} \rho_{ij} (u_j^T \bar{R}_{ij} u_j)] d\tau \tag{4-66}$$

so for a two-player game, the cost functions are as follows:

$$\begin{aligned} J_1 &= \int_{t=t_0}^{\infty} (\mathrm{y}_1^T \bar{\mathrm{Q}}_1 \mathrm{y}_1) d\tau + \rho_{11} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{11} u_1) d\tau + \rho_{12} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{12} u_2) d\tau \\ J_2 &= \int_{t=t_0}^{\infty} (\mathrm{y}_2^T \bar{\mathrm{Q}}_2 \mathrm{y}_2) d\tau + \rho_{21} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{21} u_1) d\tau + \rho_{22} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{22} u_2) d\tau \end{aligned} \tag{4-67}$$

The state-space representation of the system for a two-player game can be written as below, in which x is the matrix of the states. And y_1 and y_2 are outputs of the system for players 1 and 2 accordingly.

$$\begin{aligned} \dot{x} &= Ax + B_1 u_1 + B_2 u_2 \\ y_1 &= C_1 x + D_{11} u_1 + D_{12} u_2 \\ y_2 &= C_2 x + D_{21} u_1 + D_{22} u_2 \end{aligned} \tag{4-68}$$

In order to solve this problem, the Nash equilibrium is defined in such a way that there is no incentive for any unilateral deviation by any one of the players. This means that both players are making their decisions simultaneously where each player has an outcome that cannot unilaterally improve from a change in strategy. At Nash equilibrium, the player who chooses to change the strategy cannot improve its payoff. In order to demonstrate mathematically, a pair (u_1^*, u_2^*) corresponds to Nash equilibrium if the following relations are satisfied for each admissible strategy:

$$\begin{cases} J_1(u_1, u_2^*) \geq J_1(u_1^*, u_2^*) \\ J_2(u_1^*, u_2) \geq J_2(u_1^*, u_2^*) \end{cases} \tag{4-69}$$

In order to find the Nash solution, both cost functions must be solved simultaneously. From this part, the calculations are presented for player one, and due to similarity, the results for player two follow.

The cost function for player one can also be shown based on the states of the system

$$J_1 = \int_{t=t_0}^{\infty} \begin{array}{l} [x^T C_1{}^T \bar{Q}_1 C_1 x + 2x^T C_1{}^T \bar{Q}_1 D_{11} u_1 + u_1^T (D_{11}^T \bar{Q}_1 D_{11} + \rho_{11} \bar{R}_{11}) u_1 \\ +2x^T C_1{}^T \bar{Q}_1 D_{12} u_2 + u_2^T (D_{12}^T \bar{Q}_1 D_{12} + \rho_{11} \bar{R}_{12}) u_2 + 2 u_1^T ((D_{11}^T \bar{Q}_1 D_{12}) u_2] d\tau \end{array} \tag{4-70}$$

by defining the new variables below, the cost function can be written in a neat format.

$$\begin{aligned} & Q_1 = C_1{}^T \bar{Q}_1 C_1, \\ & N_{11} = C_1{}^T \bar{Q}_1 D_{11}, N_{12} = C_1{}^T \bar{Q}_1 D_{12} \\ & R_{11} = D_{11}^T \bar{Q}_1 D_{11} + \rho_{11} \bar{R}_{11}, R_{12} = D_{12}^T \bar{Q}_1 D_{12} + \rho_{12} \bar{R}_{12}, R_{c1} = D_{11}^T \bar{Q}_1 D_{12} \\ & J_1 = \int_{t=t_0}^{\infty} [x^T Q_1 x + 2x^T N_{11} u_1 + u_1^T R_{11} u_1 + 2x^T N_{12} u_2 + u_2^T R_{12} u_2 + 2 u_1^T R_{c1} u_2] d\tau \end{aligned} \tag{4-71}$$

Similarly:

$$J_2 = \int_{t=t_0}^{\infty} [x^T Q_2 x + 2x^T N_{21} u_1 + u_1^T R_{21} u_1 + 2x^T N_{22} u_2 + u_2^T R_{22} u_2 + 2 u_1^T R_{c2} u_2] d\tau \tag{4-72}$$

where:

$$\begin{aligned} & Q_2 = C_2{}^T \bar{Q}_2 C_2, \\ & N_{21} = C_2{}^T \bar{Q}_2 D_{21}, N_{22} = C_2{}^T \bar{Q}_2 D_{22} \\ & R_{21} = D_{21}^T \bar{Q}_2 D_{21} + \rho_{21} \bar{R}_{21}, R_{22} = D_{22}^T \bar{Q}_2 D_{22} + \rho_{22} \bar{R}_{22}, R_{c2} = D_{21}^T \bar{Q}_2 D_{22} \end{aligned} \tag{4-73}$$

Note that matrix Q is a state weighting matrix, R is a control weighting matrix, and N represents the coupling between states and control inputs.

also, consider that by defining control input matrix as $u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$, control weight matrix as $R_1 = \begin{bmatrix} R_{11} & R_{c1} \\ R_{c1}{}^T & R_{12} \end{bmatrix}$ and coupling weighting matrix as $N_1 = [N_{11}, N_{12}]$ the cost function can be shown exactly in the general form of coupled output LQR.

$$J_1 = \int_{t=t_0}^{\infty} [\mathrm{x}^T Q_1 x + 2x^T N_1 u + \mathrm{u}^T R_1 \mathrm{u}] d\tau \tag{4-74}$$

and similarly:

$$J_2 = \int_{t=t_0}^{\infty} [\mathrm{x}^T Q_2 x + 2x^T N_2 u + \mathrm{u}^T R_2 \mathrm{u}] d\tau \tag{4-75}$$

where:

$$N_2 = [N_{21}, N_{22}] \tag{4-76}$$

$$R_2 = \begin{bmatrix} R_{21} & R_{c2} \\ R_{c2}{}^T & R_{22} \end{bmatrix}$$

4.5.1 Combining the two cost functions

Consider the state space equations and optimal set of cost functions below:

$$\begin{aligned} \dot{x}_{n\times1} &= A_{n\times n} x_{n\times1} + B_{1_{n\times p}} u_{1_{p\times1}} + B_{2_{n\times q}} u_{2_{q\times1}} \\ y_{1_{m\times1}} &= C_{1_{m\times n}} x_{n\times1} + D_{11_{m\times p}} u_{1_{p\times1}} + D_{12_{m\times q}} u_{2_{q\times1}} \\ y_{2_{k\times1}} &= C_{2_{k\times n}} x_{n\times1} + D_{21_{k\times p}} u_{1_{p\times1}} + D_{22_{k\times q}} u_{2_{q\times1}} \end{aligned} \tag{4-77}$$

$$J_1 = \int_{t=t_0}^{\infty} (\mathrm{y}_1^T \bar{\mathrm{Q}}_1 \mathrm{y}_1) d\tau + \rho_{11} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{11} u_1) d\tau + \rho_{12} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{12} u_2) d\tau$$

$$J_2 = \int_{t=t_0}^{\infty} (\mathrm{y}_2^T \bar{\mathrm{Q}}_2 \mathrm{y}_2) d\tau + \rho_{21} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{21} u_1) d\tau + \rho_{22} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{22} u_2) d\tau$$

If the output equations are combined into one equation and input vectors into one input vector, the state space equations can be rewritten as:

$$\dot{x}_{n\times1} = A_{n\times n} x_{n\times1} + B_{1_{n\times p}} u_{1_{p\times1}} + B_{2_{n\times q}} u_{2_{q\times1}} = A_{n\times n} x_{n\times1} + [B_{1_{n\times p}}, B_{2_{n\times q}}] \begin{bmatrix} u_{1_{p\times1}} \\ u_{2_{q\times1}} \end{bmatrix} = A_{n\times n} x_{n\times1} + B_{n\times(p+q)} U_{(p+q)\times1}$$

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_{(m+k)\times1} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}_{(m+k)\times n} x_{n\times1} + \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = Cx + DU$$

(4-78)

The new optimal cost function can be defined as:

$$J=\int_{t=t_0}^{\infty}[\mathrm{y}^T\mathrm{Qy}+\mathrm{U}^T\bar{\mathrm{R}}\mathrm{U}]dt=\int_{t=t_0}^{\infty}[\mathrm{x}^T C^T\bar{Q}Cx+2x^T C^T\bar{Q}DU+U^T(\mathrm{D}^T\bar{\mathrm{Q}}\mathrm{D}+\rho\bar{\mathrm{R}})\mathrm{U}]dt \tag{4-79}$$

$$J=\int_{t=t_0}^{\infty}[\mathrm{x}^T\mathrm{Qx}+2x^T N\,\mathrm{U}+U^T\,\mathrm{RU}]dt$$

by comparing the general cost function above with the previous two cost functions, the matrices Q, N and R can be generated as below:

$$\bar{Q}=\begin{bmatrix}\bar{Q}_{1_{m\times m}} & 0_{m\times k}\\ 0_{k\times m} & \bar{Q}_{2_{k\times k}}\end{bmatrix}_{(m+k)\times(m+k)} \tag{4-80}$$

$$Q_{n\times n}=C^T\bar{Q}C$$

$$N_{n\times(p+q)}=C^T\bar{Q}D$$

$$\rho\bar{\mathrm{R}}=\begin{bmatrix}(\rho_{11}\bar{\mathrm{R}}_{11}+\rho_{21}\bar{\mathrm{R}}_{21})_{p\times p} & 0_{p\times q}\\ 0_{q\times p} & (\rho_{12}\bar{\mathrm{R}}_{12}+\rho_{22}\bar{\mathrm{R}}_{22})_{q\times q}\end{bmatrix}$$

$$R_{(p+q)\times(p+q)}=\mathrm{D}^T\bar{\mathrm{Q}}\mathrm{D}+\rho\bar{\mathrm{R}}$$

The solution for combined optimal cost function J can be calculated using the Matlab function "*care*". K is the optimal feedback gain for general input vector U.

```
[P,L,K] = care(A,B,Q,R,N);
```

(4-81)

4.6 Chapter summary

After a brief introduction to the two-player differential game problem, different formulation possibilities for two-player cost function structures are shown in this chapter. Then in the second section, the problem was classified into two cases of state regulation and output regulation. In the third section, the two-player game for state regulation (Case 1- LQR) is formulated for finite and infinite horizons, and the solution for coupled Riccati equations is derived. The Nash solution for the continuous two-player differential game for the infinite horizon was presented, which is in a sense similar to the approach presented by Tamaddoni [22] for discrete formulation. Both formulation and solution for the continuous problem are a contribution of this dissertation. In the fourth section, the control coupled output regulation (Case 2-CCOR) problem is introduced. The single-player solution was presented based on Yedavalli [27] using the change of variables method. In the last section, the two-player output regulation problem formulation was presented. Due to coupling between the control action parameters of the players in the cost functions, it was impossible to solve the problem by the change of variables. So a new approach was presented by combining the cost functions of players into a new general cost function. Moving forward in the next chapter, the solutions for the three-player game will be discussed based on the extension of the approaches presented in this chapter.

5 CHAPTER 5: THREE PLAYER DIFFERENTIAL GAME

5.1 Introduction to three-player paradigms

The previous chapter presented a two-player game formulation and solution for both *LQR* and *CCOR* problems. It was also shown how to formulate the quadratic cost functions for two-player decentralized and Nash paradigms. As the number of players that can partake in vehicle stability can be more than two, there is a need to expand the solutions presented in chapter 4 from a two-player game to a three-player game or higher. This allows us to explore the advantages or disadvantages of the game theory approach for integrated systems with more players to reach global stability.

This chapter presents the general formulation of the three-player game for both *LQR* and *CCOR*. In section 5.2, the player pool and the objectives are introduced. Section 5.3 and 5.4 are dedicated to defining and solving the *LQR* three-player game for the finite and infinite horizon. And finally, in section 5.5, the *CCOR* three-player game is addressed, and a solution is presented. The mathematical structure used for three players in *LQR* consists of three cost functions elaborated in section 4. This structure allows us to explore games when players have their own individual objectives. Sometimes all the players are playing a cooperative game, and they are all trying to achieve a global set of objectives together.

In general formulation, all players are playing a cooperative game together to optimize their combined goal (which is the sum of three-player individual goals in a quadratic cost function) and minimize the combined control action of all three players. Note that having a general formulation helps the controller designer investigate the effects of all parameters together in a cost function. The controller designer can play with the gains to prioritize or penalize a specific parameter, goal, error, or control action. This approach is used in section 5.5 to present a solution for *CCOR* configuration.

Similar to the previous chapter, the optimal problem can be either state regulation or output regulation. The quadratic formulation for these two different cases (*LQR* and *CCOR*) are described as below for the three-player game:

Case 1: The output that is being regulated is a linear function of only the states. Assume the state space below:

$$\begin{aligned} \dot{x} &= Ax + B_1u_1 + B_2u_2 + B_3u_3 \\ y_1 &= C_1x \\ y_2 &= C_2x \\ y_3 &= C_3x \end{aligned} \tag{5-1}$$

where y_1, y_2 and y_3 are the outputs for players 1, 2, and 3 accordingly, and the cost function for the player $i = 1, 2, 3$ is described as below:

$$\begin{aligned} J_i &= \int_0^\infty [\mathrm{y}_i^T \bar{Q}_i y_i + u_1^T \bar{R}_{i1} u_1 + u_2^T \bar{R}_{i2} u_2 + u_3^T \bar{R}_{i3} u_3] dt = \\ &= \int_0^\infty [\mathrm{x}^T \mathrm{C}_i^T \bar{Q}_i C_i \,\mathrm{x}^T + u_1^T \bar{R}_{i1} u_1 + u_2^T \bar{R}_{i2} u_2 + u_3^T \bar{R}_{i3} u_3] dt \end{aligned} \tag{5-2}$$

By comparing this cost function with the state regulation cost function below

$$J_i = \int_0^\infty [\mathrm{x}^T Q_i x + u_1^T R_{i1}\, \mathrm{u}_1 + u_2^T R_{i2}\, \mathrm{u}_2 + u_3^T R_{i3}\, \mathrm{u}_3] dt \tag{5-3}$$

the weighting matrices Q and R can be derived for each player as follows:

$$Q_i = C_i^T \bar{Q}_i C_i \tag{5-4}$$
$$R_{i1} = \bar{R}_{i1}, R_{i2} = \bar{R}_{i2}, R_{i3} = \bar{R}_{i3}$$

This problem is a standard LQR problem, and the solution for the three-player game is presented in section 5.3.

Case 2: The output is a linear function of both states and control inputs. The state space is formulated as

$$\begin{aligned} \dot{x} &= Ax + B_1u_1 + B_2u_2 + B_3u_3 \\ y_1 &= C_1x + D_{11}u_1 + D_{12}u_2 + D_{13}u_3 \\ y_2 &= C_2x + D_{21}u_1 + D_{22}u_2 + D_{23}u_3 \\ y_3 &= C_3x + D_{31}u_1 + D_{32}u_2 + D_{33}u_3 \end{aligned} \tag{5-5}$$

and the cost function for the player $i = 1, 2, 3$ is described below

$$\begin{aligned} J_i &= \int_0^\infty [\mathrm{y}_i^T \bar{Q}_i y_i + \sum_{j=1}^{3} \rho_{ij} u_j^T \bar{R}_{ij} u_j] dt = \\ &= \int_0^\infty [(C_i x + D_{i1}u_1 + D_{i2}u_2 + D_{i3}u_3)^T \bar{Q}_i (C_i x + D_{i1}u_1 + D_{i2}u_2 + D_{i3}u_3) + u_1^T \bar{R}_{i1} u_1 + u_2^T \bar{R}_{i2} u_2 + u_3^T \bar{R}_{i3} u_3] dt \end{aligned} \tag{5-6}$$

By expanding the cost function, the new cost function is:

$$J_i = \int_0^\infty \begin{bmatrix} \mathrm{x}^T \mathrm{C}_i^T \bar{Q}_i C_i \, \mathrm{x}^T + 2\,\mathrm{x}^T \mathrm{C}_i^T \bar{Q}_i D_{i1} u_1 + 2\,\mathrm{x}^T \mathrm{C}_i^T \bar{Q}_i D_{i2} u_2 + 2\,\mathrm{x}^T \mathrm{C}_i^T \bar{Q}_i D_{i3} u_3 + \ldots \\ \ldots + u_1^T (D_{i1}^T \bar{Q}_i D_{i1} + \bar{R}_{i1}) u_1 + u_2^T (D_{i2}^T \bar{Q}_i D_{i2} + \bar{R}_{i2}) u_2 + u_3^T (D_{i3}^T \bar{Q}_i D_{i3} + \bar{R}_{i3}) u_3 + \ldots \\ \ldots + 2u_1^T D_{i1}^T \bar{Q}_i D_{i2} u_2 + 2u_1^T D_{i1}^T \bar{Q}_i D_{i3} u_3 + 2u_2^T D_{i2}^T \bar{Q}_i D_{i3} u_3 \end{bmatrix} dt \tag{5-7}$$

to simplify, the new weighting matrices below are introduced:

$$Q_i = C_i^T \bar{Q}_i C_i \tag{5-8}$$
$$N_{i1} = C_i^T \bar{Q}_i D_{i1}, N_{i2} = C_i^T \bar{Q}_i D_{i2}, N_{i3} = C_i^T \bar{Q}_i D_{i3}$$
$$R_{i1} = (D_{i1}{}^T \bar{Q}_i D_{i1} + \bar{R}_{i1}), R_{i2} = (D_{i2}{}^T \bar{Q}_i D_{i2} + \bar{R}_{i2}), R_{i3} = (D_{i3}{}^T \bar{Q}_i D_{i3} + \bar{R}_{i3})$$
$$Ra_i = D_{i1}{}^T \bar{Q}_i D_{i2}, Rb_i = D_{i1}{}^T \bar{Q}_i D_{i3}, Rc_i = D_{i2}{}^T \bar{Q}_i D_{i3}$$

So the cost function can be written as:

$$J_i = \int_0^\infty \begin{bmatrix} x^T Q_i x^T + 2x^T N_{i1} u_1 + 2x^T N_{i2} u_2 + 2x^T N_{i3} u_3 + \dots \\ \dots + u_1{}^T R_{i1} u_1 + u_2{}^T R_{i2} u_2 + u_3{}^T R_{i3} u_3 + \dots \\ \dots + 2u_1{}^T Ra_i u_2 + 2u_1{}^T Rb_i u_3 + 2u_2{}^T Rc_i u_3 \end{bmatrix} dt \tag{5-9}$$

by defining control input matrix as $u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$ the state space equation will turn into the general form below

$$\dot{x} = Ax + Bu \tag{5-10}$$

where $B = [B_1, B_2, B_3]$

The general control weight matrix R_i and coupling weighting matrix N_i for the i^{th} player are defined as:

$$R_i = \begin{bmatrix} R_{i1} & Ra_i & Rb_i \\ Ra_i^T & R_{i2} & Rc_i \\ Rb_i^T & Rc_i^T & R_{i3} \end{bmatrix} \tag{5-11}$$
$$N_i = [N_{i1}, N_{i2}, N_{i3}]$$

using the general weighting matrices Q_i R_i and N_i, the cost function for the i^{th} player can be written as a general output regulation form below

$$J_i = \int_{t=t_0}^{\infty} [x^T Q_i x + 2x^T N_i u + u^T R_i u] d\tau \tag{5-12}$$

The matrix N_i shows the coupling between the states and inputs. This case is under a class of optimal control problems called control coupled output regulation (*CCOR*).

5.2 Player selection and output function

This section discusses the three players' player pool and the objectives that can be defined in cost functions. In the vehicle dynamics, there can be numerous control inputs (players) and control objectives, and exploring all of them is out of the scope of this research, so a limited player pool is chosen in this thesis as a case study to explore the benefits of game theory. This dissertation is limited to games from a player pool, including active suspension, corrective roll moment, active steering, and corrective roll moment.

The active suspension system is capable of minimizing vertical acceleration ($\ddot{z}_c$), suspension strokes ($(z_{sl} - z_{ul})$ and $(z_{sr} - z_{ur})$), tire deflections ($(z_{ul} - z_{rl})$ and $(z_{ur} - z_{rr})$) individually, or all together, and control efforts are left and right active suspension forces (Fa_l and Fa_r).

The anti-roll system as a player tries to minimize roll angle and its first derivative (ϕ and $\dot{\phi}$) by creating a corrective roll moment ($M_{\phi c}$) using an anti-roll bar actuator.

The yaw control system uses corrective yaw moment (M_{Yc}) and is mostly used for yaw stability, like minimizing yaw rate error ($e_r = r - r_{desired}$). This moment can be either be generated by differential braking torques or differential torques applied to vehicle tires using one of the direct yaw moment control approaches in vehicle dynamics. Applying differential accelerating torques and braking torques on tires can directly affect the longitudinal and lateral motion of the vehicle. The nonlinear tire model and a full car model with wheel dynamic modeling are required to investigate these effects, making the model and control problem more complicated. In this dissertation, the priority is to present a game theory approach in vehicle stability, and the corrective yaw moment is considered as a single control input for simplicity. The nonlinear tire model problem is addressed in chapter 7, and torque distribution and designing the full car model controller can be considered potential future works for this research.

Active steering control can also be treated like a player that corrects the driving steering input, and its objective can either be defined to bring yaw stability and tracking or can be used as a counter-steering to enhance vehicle roll performance.

Between these four players, any combination of three can be chosen as a case study to implement the methods shown in the following sections. The three-player game between active-steering, corrective yaw moment, and corrective roll moment is simulated in section 6.4 for the linear tire model and in section 7.5 for the nonlinear tire model.

5.3 Calculating three-player coupled DRE[14] for continuous finite horizon *LQR* system

This section presents the three-player formulation for the LQR method for continuous finite-horizon dynamic systems. Using the Hamiltonian approach and feedback laws presented in [28], the set of differential Ricatti equations (DREs) is derived. The optimal control problem that is solved in this thesis is for the infinite

[14] Differential Ricatti Equations: for finite horizon problem, the set of Riccatti equations are in the form of differential equations. This format changes to algebraic Ricatti equations (AREs) when the optimal problem is in finite horizon

horizon. To achieve the solution for the infinite horizon, the DREs calculated in this section will be changed to AREs (algebraic Ricatti equations) in the next section for the infinite horizon problem, and an algorithm is proposed to solve the AREs. The solution to AREs will lead to finding the feedback gains to construct the control inputs.

Given a linear time-varying plant

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B_1(t)u_1(t) + B_2(t)u_2(t) + B_3(t)u_3(t) \\ x(t_0) &= x_0 \end{aligned} \tag{5-13}$$

the quadratic performance index for player *i* is shown as

$$J_i = x(t_f)F_{if}x(t_f) + \int_0^{t_f} (x^T(t)Q_i(t)x(t) + u_1^T(t)R_{i1}(t)u_1(t) + u_2^T(t)R_{i2}(t)u_2(t) + u_3^T(t)R_{i3}(t)u_3(t))dt \tag{5-14}$$

where $u_i(t)$ is not constrained, t_f is specified, but $x(t_f)$ is unknown. The terminal cost function F_{if} (or $F_i(t_f)$) and weight function $Q_i(t)$ are symmetric, positive semidefinite matrices, and $R_{ij}(t)$ are symmetric, positive definite matrices.

The Hamiltonian function for the i^{th} player can be defined using a co-state vector of n^{th} order $\lambda_i(t)$

$$H_i(x, u_1, u_2, u_3, \lambda_1) = x^T Q_i x + u_1^T R_{i1} u_1 + u_2^T R_{i2} u_2 + u_3^T R_{i3} u_3 + \lambda_i^T (Ax + B_1 u_1 + B_2 u_2 + B_3 u_3) \tag{5-15}$$

Optimal control inputs for three players are given by

$$\frac{\partial H_1}{\partial u_1} = 0 \rightarrow R_{11}u_1^* + B_1^T\lambda_1 = 0 \rightarrow u_1^* = -R_{11}{}^{-1}B_1^T\lambda_1$$
$$\frac{\partial H_2}{\partial u_2} = 0 \rightarrow R_{22}u_2^* + B_2^T\lambda_2 = 0 \rightarrow u_2^* = -R_{22}{}^{-1}B_2^T\lambda_2$$
$$\frac{\partial H_3}{\partial u_3} = 0 \rightarrow R_{33}u_1^* + B_3^T\lambda_3 = 0 \rightarrow u_3^* = -R_{33}{}^{-1}B_3^T\lambda_3 \tag{5-16}$$

Co-state functions can be written as a function of states using the feedback law

$$\lambda_1 = P_1(t)x^*(t)$$
$$\lambda_2 = P_2(t)x^*(t)$$
$$\lambda_3 = P_3(t)x^*(t) \tag{5-17}$$

Where $P_1(t)$, $P_2(t)$ and $P_3(t)$ are yet to be determined. By comparing the boundary conditions for the terminal condition, it can be written that $P(t_f) = F(t_f)$

$$\lambda_1(t_f) = F_{1f}x^*(t_f)$$
$$\lambda_2(t_f) = F_{2f}x^*(t_f)$$
$$\lambda_3(t_f) = F_{3f}x^*(t_f) \tag{5-18}$$

Control inputs can be rearranged using the feedback law

$$u_1^* = -R_{11}{}^{-1}B_1^T\lambda_1 = -R_{11}{}^{-1}(t)B_1^T(t)P_1(t)x^*(t) = -R_{11}{}^{-1}B_1^TP_1x^*$$
$$u_2^* = -R_{22}{}^{-1}B_2^T\lambda_2 = -R_{22}{}^{-1}B_2^TP_2x^*$$
$$u_3^* = -R_{33}{}^{-1}B_3^T\lambda_3 = -R_{33}{}^{-1}B_3^TP_3x^* \tag{5-19}$$

Optimal state and co-state equations are given by

$$\dot{x}^*(t) = \frac{\partial H_i}{\partial \lambda_i} \rightarrow \dot{x}^* = Ax^* - B_1R_{11}{}^{-1}B_1^TP_1x^* - B_2R_{22}{}^{-1}B_2^TP_2x^* - B_3R_{33}{}^{-1}B_3^TP_3x^* \tag{5-20}$$

$$\lambda_i^* = -\frac{H_i}{\partial x} - \frac{H_i}{\partial u_j}\frac{\partial u_j{}^*}{\partial x} - \frac{H_i}{\partial u_k}\frac{\partial u_k{}^*}{\partial x}$$

Note that the (*i*) subscript in the co-state equation is for the intended co-state that is being calculated, and *j* and *k* are for other players.

$$\lambda_1^* = -\frac{H_1}{\partial x} - \frac{H_1}{\partial u_2}\frac{\partial u_2^*}{\partial x} - \frac{H_1}{\partial u_3}\frac{\partial u_3^*}{\partial x}$$
$$\lambda_2^* = -\frac{H_2}{\partial x} - \frac{H_2}{\partial u_1}\frac{\partial u_1^*}{\partial x} - \frac{H_2}{\partial u_3}\frac{\partial u_3^*}{\partial x}$$
$$\lambda_3^* = -\frac{H_3}{\partial x} - \frac{H_3}{\partial u_1}\frac{\partial u_1^*}{\partial x} - \frac{H_3}{\partial u_2}\frac{\partial u_2^*}{\partial x} \tag{5-21}$$

According to symmetry properties of equations for three players, DRE calculation for player one is presented here, and Riccati equations for other players are written based on the first player.

Consider the three terms on the left side of the co-state equation for player one

$$-\frac{H_1}{\partial x} = -Q_1 x^* - A^T \lambda_1^* = -Q_1 x^* - A^T P_1 x^* \tag{5-22}$$
$$-\frac{H_1}{\partial u_2}\frac{\partial u_2^*}{\partial x} = (-R_{12}u_2 - B_2^T\lambda_1)\frac{\partial u_2^*}{\partial x} = (-R_{12}u_2 - B_2^T\lambda_1)(R_{22}{}^{-1}B_2^T P_2) =$$
$$[-R_{12}(R_{22}{}^{-1}B_2^T P_2 x^*) - B_2^T P_1 x^*](R_{22}{}^{-1}B_2^T P_2) = -P_2 B_2 R_{22}^{-1}[R_{12}R_{22}{}^{-1}B_2^T P_2 - B_2^T P_1]x^* =$$
$$[-P_2 B_2 R_{22}^{-1} R_{12} R_{22}{}^{-1} B_2^T P_2 + P_2 B_2 R_{22}^{-1} B_2^T P_1]x^* = [-P_2 S_{12} P_2 + P_2 S_2 P_1]x^*$$
$$-\frac{H_1}{\partial u_3}\frac{\partial u_3^*}{\partial x} = (-R_{13}u_3 - B_3^T\lambda_1)\frac{\partial u_3^*}{\partial x} = (-R_{13}u_3 - B_3^T\lambda_1)(R_{33}{}^{-1}B_3^T P_3) =$$
$$[-R_{13}(R_{33}{}^{-1}B_3^T P_3 x^*) - B_3^T P_1 x^*](R_{33}{}^{-1}B_3^T P_3) = -P_3 B_3 R_{33}^{-1}[R_{13}R_{33}{}^{-1}B_3^T P_3 - B_3^T P_1]x^* =$$
$$[-P_3 B_3 R_{33}^{-1} R_{13} R_{33}{}^{-1} B_3^T P_3 + P_3 B_3 R_{33}^{-1} B_3^T P_1]x^* = [-P_2 S_{13} P_2 + P_2 S_3 P_1]x^*$$

The parameter S_{mn} is introduced below to simplify the writing of the equations:

$$S_1 = B_1 R_{11}^{-1} B_1^T \qquad S_{12} = B_2 R_{22}^{-1} R_{12} R_{22}{}^{-1} B_2^T \qquad S_{13} = B_3 R_{33}^{-1} R_{13} R_{33}{}^{-1} B_3^T$$
$$S_{21} = B_1 R_{11}^{-1} R_{21} R_{11}{}^{-1} B_1^T \qquad S_2 = B_2 R_{22}^{-1} B_2^T \qquad S_{23} = B_3 R_{33}^{-1} R_{23} R_{33}{}^{-1} B_3^T$$
$$S_{31} = B_1 R_{11}^{-1} R_{31} R_{11}{}^{-1} B_1^T \qquad S_{32} = B_2 R_{22}^{-1} R_{32} R_{22}{}^{-1} B_2^T \qquad S_3 = B_3 R_{33}^{-1} B_3^T \tag{5-23}$$

The co-state equation for Player One can be rewritten as

$$\dot{\lambda}_1^* = -Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* - P_3 S_{13} P_3 x^* + P_3 S_3 P_1 x^* \tag{5-24}$$

The derivative of optimal co-state can also be calculated from the equation below:

$$\lambda_1(t) = P_1(t) x^*(t) \rightarrow \dot{\lambda}_1^* = \dot{P}_1 x^* + P_1 \dot{x}^* \tag{5-25}$$

Comparing the two equations for co-states:

$$-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* - P_3 S_{13} P_3 x^* + P_3 S_3 P_1 x^* = \dot{P}_1 x^* + P_1 \dot{x}^* \tag{5-26}$$

$\dot{x}^*$ can be entered into the equation above from the optimal state equation

$$\begin{aligned} &-Q_1 x^* - A^T P_1 x^* - P_2 S_{12} P_2 x^* + P_2 S_2 P_1 x^* - P_3 S_{13} P_3 x^* + P_3 S_3 P_1 x^* = \\ &\dot{P}_1 x^* + P_1 (A x^* - B_1 R_{11}{}^{-1} B_1^T P_1 x^* - B_2 R_{22}{}^{-1} B_2^T P_2 x^* - B_3 R_{33}{}^{-1} B_3^T P_3 x^*) \end{aligned} \tag{5-27}$$

By rearranging the equation above and omitting the multiplier x^* the differential Riccati equation for player one can be derived with the terminal boundary condition below

$$\begin{aligned} &-\dot{P}_1 = P_1 A + A^T P_1 + Q_1 - P_1 S_1 P_1 - P_1 S_2 P_2 - P_1 S_3 P_3 - P_2 S_2 P_1 - P_3 S_3 P_1 + P_2 S_{12} P_2 + P_3 S_{13} P_3 \\ &P_1(t_f) = P_{1f} = F_{1f} \end{aligned} \tag{5-28}$$

Following the same procedures for Players Two and Three, the coupled differential Riccati equations for the three-player game can be written based on equation (5-29) for player one:

$$\dot{P}_1 + P_1 A + A^T P_1 + Q_1 - P_1 S_1 P_1 - P_1 S_2 P_2 - P_1 S_3 P_3 - P_2 S_2 P_1 - P_3 S_3 P_1 + P_2 S_{12} P_2 + P_3 S_{13} P_3 = 0 \tag{5-29}$$

$$\begin{aligned} &\dot{P}_1 + P_1 A + A^T P_1 + Q_1 - P_1 S_1 P_1 - P_1 S_2 P_2 - P_1 S_3 P_3 - P_2 S_2 P_1 - P_3 S_3 P_1 + P_2 S_{12} P_2 + P_3 S_{13} P_3 = 0 \\ &\dot{P}_2 + P_2 A + A^T P_2 + Q_2 - P_2 S_1 P_1 - P_2 S_2 P_2 - P_2 S_3 P_3 - P_1 S_1 P_2 - P_3 S_3 P_2 + P_1 S_{21} P_1 + P_3 S_{23} P_3 = 0 \\ &\dot{P}_3 + P_3 A + A^T P_3 + Q_3 - P_3 S_1 P_1 - P_3 S_2 P_2 - P_3 S_3 P_3 - P_1 S_1 P_3 - P_2 S_2 P_3 + P_1 S_{31} P_1 + P_2 S_{32} P_2 = 0 \\ &P_1(t_f) = P_{1f} = F_{1f} \\ &P_2(t_f) = P_{2f} = F_{2f} \\ &P_3(t_f) = P_{3f} = F_{3f} \end{aligned} \tag{5-30}$$

By solving the equations above for P_1, P_2 and P_3, co-state values of λ_1, λ_2 and λ_3 are obtained. Optimal control inputs of u_1^* u_2^* and u_3^* can be calculated based on state feedback. As mentioned, this thesis aims to find the solution for the infinite horizon, which means that the final horizon for the cost function will be infinite. The next section uses the equations derived in this section to address the infinite horizon problem.

5.4 Infinite time LQR for the three-player game

In order to define the problem in the infinite horizon, the final time (t_f) in the previous sections turned infinite. According to this change, the state equation remains the same, but the cost function for *i^th^* player is changed to the form below

$$J_i = \int_0^\infty (x^T Q_i x + u_1{}^T R_{i1} u_1 + u_2{}^T R_{i2} u_2 + u_3{}^T R_{i3} u_3) dt \tag{5-31}$$

The system must be completely controllable. Using results similar to the finite-horizon problem P_i is still the solution of a Riccati equation with the boundary condition:

$$P(t_f) = F(t_f) = 0 \tag{5-32}$$

The optimal control inputs can be obtained as:

$$\begin{aligned} u_1^* &= -R_{11}{}^{-1} B_1^T \hat{P}_1 x^* \\ u_2^* &= -R_{22}{}^{-1} B_2^T \hat{P}_2 x^* \\ u_2^* &= -R_{33}{}^{-1} B_3^T \hat{P}_3 x^* \end{aligned} \tag{5-33}$$

where

$$\hat{P}_i(t) = \lim_{t_f \to \infty} \{P(t)\} = \bar{P}_i \tag{5-34}$$

$\bar{P}_i$ is a positive definite, symmetric, constant matrix that turns the coupled differential Riccati equations into the coupled algebraic Riccati equations below:

$$\frac{d\bar{P}_1}{dt} = 0 = \bar{P}_1 A + A^T \bar{P}_1 + Q_1 - \bar{P}_1 S_1 \bar{P}_1 - \bar{P}_1 S_2 \bar{P}_2 - \bar{P}_1 S_3 \bar{P}_3 - \bar{P}_2 S_2 \bar{P}_1 - \bar{P}_3 S_3 \bar{P}_1 + \bar{P}_2 S_{12} \bar{P}_2 + \bar{P}_3 S_{13} \bar{P}_3$$

$$\frac{d\bar{P}_2}{dt} = 0 = \bar{P}_2 A + A^T \bar{P}_2 + Q_2 - \bar{P}_2 S_1 \bar{P}_1 - \bar{P}_2 S_2 \bar{P}_2 - \bar{P}_2 S_3 \bar{P}_3 - \bar{P}_1 S_1 \bar{P}_2 - \bar{P}_3 S_3 \bar{P}_2 + \bar{P}_1 S_{21} \bar{P}_1 + \bar{P}_3 S_{23} \bar{P}_3$$

$$\frac{d\bar{P}_3}{dt} = 0 = \bar{P}_3 A + A^T \bar{P}_3 + Q_3 - \bar{P}_3 S_1 \bar{P}_1 - \bar{P}_3 S_2 \bar{P}_2 - \bar{P}_3 S_3 \bar{P}_3 - \bar{P}_1 S_1 \bar{P}_3 - \bar{P}_2 S_2 \bar{P}_3 + \bar{P}_1 S_{31} \bar{P}_1 + \bar{P}_2 S_{32} \bar{P}_2$$

(5-35)

Optimal control signals can be obtained from the equations below after solving the coupled ARE's for $\bar{P}_1$, $\bar{P}_2$ and $\bar{P}_3$

$$u_1^* = -R_{11}{}^{-1} B_1^T \bar{P}_1 x^* \quad (5\text{-}36)$$
$$u_2^* = -R_{22}{}^{-1} B_2^T \bar{P}_2 x^*$$
$$u_3^* = -R_{33}{}^{-1} B_3^T \bar{P}_3 x^*$$

The optimal state x^* can be calculated by solving the state equation using the initial boundary condition

$$\dot{x}^* = [A - B_1 R_{11}{}^{-1} B_1^T \bar{P}_1 - B_2 R_{22}{}^{-1} B_2^T \bar{P}_2 - B_3 R_{33}{}^{-1} B_3^T \bar{P}_3] x^* \quad (5\text{-}37)$$
$$x^*(t_0) = x_0$$

and optimal cost functions can be obtained using the equations below:

$$J_1 = \frac{1}{2} x^*(t)^T \bar{P}_1 x^*(t) \quad (5\text{-}38)$$
$$J_2 = \frac{1}{2} x^*(t)^T \bar{P}_2 x^*(t)$$
$$J_3 = \frac{1}{2} x^*(t)^T \bar{P}_3 x^*(t)$$

As is clear from the formulation of control inputs, the values of u_1^* u_2^* and u_3^* are completely dependent on the solution to the set of ARE's and finding $\bar{P}_1$, $\bar{P}_2$ and

$\overline{P}_3$. In the next section, an algorithm similar to the one used for the two-player game is proposed for the three-player game to solve the ARE's.

5.4.1 Solving algebraic Riccati equations (ARE's)

The Matlab function (*care*) is used to solve the algebraic Riccati equations. According to the description of the function in the Matlab library, this function can solve the continuous algebraic Riccati equation below and return the unique solution for *P*, the gain matrix $G = R^{-1}(B^T PE + S^T)$, and closed-loop eigenvalues vector L.

$$A^T PE + E^T PA - (E^T PB + S)R^{-1}(B^T PE + S^T) + Q = 0 \tag{5-39}$$

This equation can also be simplified by omitting *R* and *S (R=I, S=0)* and setting *E* to default value *E=I*.

All the Riccati equations that can be written in the format above can be solved using the "*care*" function if their associated Hamiltonian eigenvalues are far from the imaginary axis. There is a finite stabilizing solution for *P*. This function can be used in a straightforward way to solve the Riccati equation for decentralized paradigms. Still, a new solution needs to be introduced due to the existence of coupling between the Riccati equations in the Nash paradigms. The proposed solution in this dissertation is a numerical based solution and is described by the following steps:

Step 1:

Initiate P_1^C, P_2^C and P_3^C by solving the following decoupled ARE's:

$$\begin{aligned} A^T P_1^C + P_1^C A - P_1^C S_{11} P_1^C + Q_1 = 0 \\ A^T P_2^C + P_2^C A - P_2^C S_{22} P_2^C + Q_2 = 0 \\ A^T P_3^C + P_3^C A - P_3^C S_{33} P_3^C + Q_3 = 0 \end{aligned} \tag{5-40}$$

P_1^C, P_2^C and P_3^C can be calculated in Matlab using the code below

$$
\begin{aligned}
&P_1^C = care(A, B_1, Q_1, R_{11}) \\
&P_2^C = care(A, B_2, Q_2, R_{22}) \\
&P_3^C = care(A, B_3, Q_3, R_{33})
\end{aligned} \tag{5-41}
$$

Step 2:

Compute the strong solution P_1^{C+1} P_2^{C+1} and P_3^{C+1} from the following decoupled ARE's:

$$
\begin{aligned}
&P_1^{C+1}(A - S_2P_2^C - S_3P_3^C) + (A - S_2P_2^C - S_3P_3^C)^T P_1^{C+1} - P_1^{C+1}S_1P_1^{C+1} + Q_1 + P_2^C S_{12}P_2^C + P_3^C S_{13}P_3^C = 0 \\
&P_2^{C+1}(A - S_1P_1^C - S_3P_3^C) + (A - S_1P_1^C - S_3P_3^C)^T P_2^{C+1} - P_2^{C+1}S_2P_2^{C+1} + Q_2 + P_1^C S_{21}P_1^C + P_3^C S_{23}P_3^C = 0 \\
&P_3^{C+1}(A - S_1P_1^C - S_2P_2^C) + (A - S_1P_1^C - S_2P_2^C)^T P_3^{C+1} - P_3^{C+1}S_3P_3^{C+1} + Q_3 + P_1^C S_{31}P_1^C + P_2^C S_{32}P_2^C = 0
\end{aligned} \tag{5-42}
$$

Equations below can be used to simplify the format of the Ricatti equation to the acceptable format for the "*care*" function

$$
\begin{aligned}
&A_1^{'} = (A - S_2P_2^C - S_3P_3^C), \\
&Q_1^{'} = Q_1 + P_2^C S_{12}P_2^C + P_3^C S_{13}P_3^C, \\
&A_2^{'} = (A - S_{11}P_1^C - S_3P_3^C), \\
&Q_2^{'} = Q_2 + P_1^C S_{21}P_1^C + P_3^C S_{23}P_3^C, \\
&A_3^{'} = (A - S_{11}P_1^C - S_2P_2^C), \\
&Q_3^{'} = Q_3 + P_1^C S_{31}P_1^C + P_2^C S_{32}P_2^C
\end{aligned} \tag{5-43}
$$

The Riccati equations can be rearranged as:

$$
\begin{aligned}
&P_1^{C+1}A_1^{'} + A_1^{'T}P_1^{C+1} - P_1^{C+1}S_1P_1^{C+1} + Q_1^{'} = 0 \\
&P_2^{C+1}A_2^{'} + A_2^{'T}P_2^{C+1} - P_2^{C+1}S_2P_2^{C+1} + Q_2^{'} = 0 \\
&P_3^{C+1}A_3^{'} + A_3^{'T}P_3^{C+1} - P_3^{C+1}S_3P_3^{C+1} + Q_3^{'} = 0
\end{aligned} \tag{5-44}
$$

P_1^{C+1}, P_2^{C+1} and P_3^{C+1} can be calculated in Matlab using the code below:

$$P_1^{C+1} = care(A_1^{'}, B_1, Q_1^{'}, R_{11}) \qquad (5\text{-}45)$$
$$P_2^{C+1} = care(A_2^{'}, B_2, Q_2^{'}, R_{22})$$
$$P_2^{C+1} = care(A_3^{'}, B_3, Q_3^{'}, R_{33})$$

Step 3:

This step updates P_i^C and investigates the convergence of P_i^{C+1} to P_i^C using the Euclidean norm to stop the iteration. The algorithm for convergence is presented as a "*while*" loop below:

While $\left\| P_i^{C+1} - P_i^C \right\| \geq \varepsilon$ (5-46)

$P_i^C = P_i^{C+1}$

Repeat Step 2

End

The approach presented in this chapter can also be used for N>3 player games if the algorithm converges.

5.5 Solving three-player game using control coupled output regulation

The previous section presented the problem and solution for three-player state regulation (*LQR*). The introduction section shows that multiple couplings exist between the states and inputs and between the inputs themselves in control coupled output regulation problem (*CCOR*). These couplings make it impossible to solve the three-player *CCOR* problem with the same method for *LQR* or the change of variables method.

In this section, the problem formulation for the three-player game is presented again as a review, and the solution is achieved by combining all the cost functions as a whole.

Each player's cost function is defined as a combination of quadratic output costs that the player wants to regulate plus the player's weighted quadratic control cost. The output cost function is weighted and normalized by matrix $\bar{Q}$, and the control cost function is normalized by $\bar{R}$ and multiplied by ρ. Bryson's rule is used in weighting matrices $\bar{Q}$ and $\bar{R}$ normalizes the cost function values. The multiplier ρ is used as a design parameter that the designer can tune to get the desired performance from the controller.

The player i^{th} cost function in an *N*-player game is defined as below:

$$J_i = Jy_i + \sum_{j=1}^{N} \rho_{ij} Ju_j = \int_{t=t_0}^{\infty} [y_i^T Q_i y_i + \sum_{j=1}^{N} \rho_{ij} (u_j^T \bar{R}_{ij} u_j)] d\tau \tag{5-47}$$

The state-space representation of the system for a two-player game can be written below, in which x is the states' matrix. The vectors y_1 y_2 and y_3 are outputs of the system for three players accordingly.

$$J_i = Jy_i + \sum_{j=1}^{N} \rho_{ij} Ju_j = \int_{t=t_0}^{\infty} [y_i^T Q_i y_i + \sum_{j=1}^{N} \rho_{ij} (u_j^T \bar{R}_{ij} u_j)] d\tau \tag{5-48}$$

Like the two-player game, Nash equilibrium is defined so that there is no incentive for any unilateral deviation by any one of the players. This means that all three players are making their decisions simultaneously where each player has an outcome that cannot unilaterally improve from a change in strategy. So basically, at Nash equilibrium, the player who chooses to change the strategy cannot improve its payoff. In mathematical terms, the input vector (u_1^*, u_2^*, u_3^*) corresponds to Nash equilibrium if the following relations are satisfied for each admissible strategy:

$$\begin{cases} J_1(u^*{}_1, u_2, u_3) \geq J_1(u_1^*, u_2^*, u_3^*) \\ J_2(u_1, u^*{}_2, u_3) \geq J_2(u_1^*, u_2^*, u_3^*) \\ J_3(u_1, u_2, u^*{}_3) \geq J_3(u_1^*, u_2^*, u_3^*) \end{cases} \tag{5-49}$$

In order to find the Nash solution, all three cost functions must be solved simultaneously. From this part, the calculations are presented for the ith player as follows.

In the first section, it was shown that the cost function for the i^{th} player could be written in the general form below

$$J_i = \int_{t=t_0}^{\infty} [\mathrm{x}^T Q_i x + 2x^T N_i u + \mathrm{u}^T R_i \mathrm{u}] d\tau \tag{5-50}$$

Consider the state space equations and optimal set of cost functions below:

$$\begin{aligned}
\dot{x}_{n\times1} &= A_{n\times n} x_{n\times1} + B_{1_{n\times p}} u_{1_{p\times1}} + B_{2_{n\times q}} u_{2_{q\times1}} + B_{3_{n\times r}} u_{3_{r\times1}} \\
y_{1_{m\times1}} &= C_{1_{m\times n}} x_{n\times1} + D_{11_{m\times p}} u_{1_{p\times1}} + D_{12_{m\times q}} u_{2_{q\times1}} + D_{13_{m\times r}} u_{3_{r\times1}} \\
y_{2_{k\times1}} &= C_{2_{k\times n}} x_{n\times1} + D_{21_{k\times p}} u_{1_{p\times1}} + D_{22_{k\times q}} u_{2_{q\times1}} + D_{13_{k\times r}} u_{3_{r\times1}} \\
y_{3_{l\times1}} &= C_{2_{l\times n}} x_{n\times1} + D_{31_{l\times p}} u_{1_{p\times1}} + D_{32_{l\times q}} u_{2_{q\times1}} + D_{33_{l\times r}} u_{3_{r\times1}}
\end{aligned} \tag{5-51}$$

$$\begin{aligned}
J_1 &= \int_{t=t_0}^{\infty} (\mathrm{y}_1^T \bar{\mathrm{Q}}_1 \mathrm{y}_1) d\tau + \rho_{11} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{11} u_1) d\tau + \rho_{12} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{12} u_2) d\tau + \rho_{13} \int_{t=t_0}^{\infty} (\mathrm{u}_3^T \bar{R}_{13} u_3) d\tau \\
J_2 &= \int_{t=t_0}^{\infty} (\mathrm{y}_2^T \bar{\mathrm{Q}}_2 \mathrm{y}_2) d\tau + \rho_{21} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{21} u_1) d\tau + \rho_{22} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{22} u_2) d\tau + \rho_{23} \int_{t=t_0}^{\infty} (\mathrm{u}_3^T \bar{R}_{23} u_3) d\tau \\
J_3 &= \int_{t=t_0}^{\infty} (\mathrm{y}_3^T \bar{\mathrm{Q}}_3 \mathrm{y}_3) d\tau + \rho_{31} \int_{t=t_0}^{\infty} (\mathrm{u}_1^T \bar{R}_{31} u_1) d\tau + \rho_{32} \int_{t=t_0}^{\infty} (\mathrm{u}_2^T \bar{R}_{32} u_2) d\tau + \rho_{33} \int_{t=t_0}^{\infty} (\mathrm{u}_3^T \bar{R}_{33} u_3) d\tau
\end{aligned} \tag{5-52}$$

If we combine output equations into one equation and input vectors into one input vector, we can rewrite the state space equations above as:

$$\dot{x}_{n\times1} = A_{n\times n} x_{n\times1} + B_{1_{n\times p}} u_{1_{p\times1}} + B_{2_{n\times q}} u_{2_{q\times1}} + B_{3_{n\times r}} u_{3_{r\times1}} = A_{n\times n} x_{n\times1} + [B_{1_{n\times p}}, B_{2_{n\times q}}, B_{3_{n\times r}}] \begin{bmatrix} u_{1_{p\times1}} \\ u_{2_{q\times1}} \\ u_{3_{r\times1}} \end{bmatrix} = A_{n\times n} x_{n\times1} + B_{n\times(p+q+r)} U_{(p+q+r)\times1}$$

$$y=\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_{(m+k+l)\times 1} = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix}_{(m+k+l)\times n} x_{n\times 1} + \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = Cx + DU$$

(5-53)

The new optimal cost function can be defined as:

$$J = \int_{t=t_0}^{\infty} [\mathrm{y}^T \mathrm{Qy} + \mathrm{U}^T \bar{\mathrm{R}} \mathrm{U}]dt = \int_{t=t_0}^{\infty} [\mathrm{x}^T C^T \bar{Q} C x + 2x^T C^T \bar{Q} D U + U^T (\mathrm{D}^T \bar{\mathrm{Q}} \mathrm{D} + \rho \bar{\mathrm{R}}) \mathrm{U}]dt \qquad (5\text{-}54)$$

$$J = \int_{t=t_0}^{\infty} [\mathrm{x}^T \mathrm{Qx} + 2x^T N \,\mathrm{U} + U^T \,\mathrm{RU}]dt$$

By comparing the general cost function above with the three players cost functions, the matrices *Q*, *N*, and *R* can be generated as below:

$$\bar{Q} = \begin{bmatrix} \bar{Q}_{1_{m\times m}} & 0_{m\times k} & 0_{m\times l} \\ 0_{k\times m} & \bar{Q}_{2_{k\times k}} & 0_{k\times l} \\ 0_{l\times m} & 0_{l\times k} & \bar{Q}_{3_{l\times l}} \end{bmatrix}_{(m+k+r)\times(m+k+r)}$$

$$Q_{n\times n} = C^T \bar{Q} C$$

$$N_{n\times(p+q)} = C^T \bar{Q} D$$

$$\rho \bar{\mathrm{R}} = \begin{bmatrix} (\rho_{11} \bar{\mathrm{R}}_{11} + \rho_{21} \bar{\mathrm{R}}_{21} + \rho_{31} \bar{\mathrm{R}}_{21})_{p\times p} & 0_{p\times q} & 0_{p\times r} \\ 0_{q\times p} & (\rho_{12} \bar{\mathrm{R}}_{12} + \rho_{22} \bar{\mathrm{R}}_{22} + \rho_{32} \bar{\mathrm{R}}_{32})_{q\times q} & 0_{q\times r} \\ 0_{r\times p} & 0_{r\times q} & (\rho_{13} \bar{\mathrm{R}}_{13} + \rho_{23} \bar{\mathrm{R}}_{23} + \rho_{33} \bar{\mathrm{R}}_{33})_{r\times r} \end{bmatrix}$$

$$R_{(p+q+r)\times(p+q+r)} = \mathrm{D}^T \bar{\mathrm{Q}} \mathrm{D} + \rho \bar{\mathrm{R}}$$

(5-55)

The solution for combined optimal cost function *J* can be calculated using the Matlab function "*care*". *K* is the optimal feedback gain for general input vector *U*.

```
[P,L,K]=care(A,B,Q,R,N);
```

(5-56)

5.6 Chapter summary

Like the chapter 4 pattern, the three-player game problem was introduced for the state regulation and output regulation cases, and the mathematical formulation was presented. The solution for a continuous three-player LQR game was delivered for the infinite horizon, and an algorithm was developed to solve the coupled Riccati equations. It was shown that the output regulation cost function for each player could be transformed to a standard control coupled output regulation format, and the three-player game can be solved by combining all the cost functions into one cost function. The next chapter implements the solution for the two-player and three-player game in simulations and investigates the effects of using game theory, and explores the scenarios that can be useful.

6 CHAPTER 6: SIMULATION RESULTS FOR PLANT WITH LINEAR TIRE MODEL

6.1 Introduction

In this chapter, multiple scenarios for two-player games and three-player games are presented. After stating the paradigm and defining the controller's objective in the cost function, the controller's gains are calculated accordingly using the information in previous chapters. The players are chosen from the player pool below:

- Active steering (δ_{AS})
- Corrective yaw moment (M_{yc})
- Corrective roll moment ($M_{\phi c}$)
- Active suspension forces (F_{al} and F_{ar})

Here is the list of scenarios that are explored in this chapter:

- Active steering and corrective yaw moment (two-player)
- Active steering and corrective roll moment (two-player)
- Active steering, corrective yaw moment, and corrective roll moment (three-player)
- Active suspension and the corrective roll moment

All the first three scenarios belong to the class of LQR problems (Case 1), and the last scenario belongs to the class of CCOR (Case 2).

In each scenario, at first, the control solution for the single-player game is presented, and then the solution for the multi-player game is shown in both decentralized and Nash paradigms. Control models in sections 3.3 and 3.4 are used to calculate control feedback gains, and then the feedback gains are applied on the plant model with linear tire presented in section 3.6. The simulation results for all the paradigms are plotted together with the uncontrolled passive maneuver to compare. The root mean square of the control input, the peak value of the controller input and the cost function's value were also used as factors of merit to make a comparison. For each scenario, the discussion is presented based on the simulation plots and the factors of merit.

In the Appendix, all the eigenvalues and the natural frequencies of the models are calculated, and they are all shown to be below 100 Hz. In order to cover all the frequencies, the sampling time of 0.001 sec is chosen for the simulation, which is equivalent to 1000 Hz that is at least ten times more than the highest frequency of the system.

The passive maneuver selected for the simulations is a 2.5-meter lane-change maneuver with a constant longitudinal velocity of 20 m/s. Parameters for this vehicle are presented in the appendix. The human driver applies a $\pi/24$ radian (7.5 degrees clockwise) steering angle from 1-1.5 s and reverses the steering angle ($-\pi/24$ CCW) from 1.5-2 s. The steering angle for all the other times is zero. No controller signal is added in the passive maneuver, so the total steering input signal is plotted below.

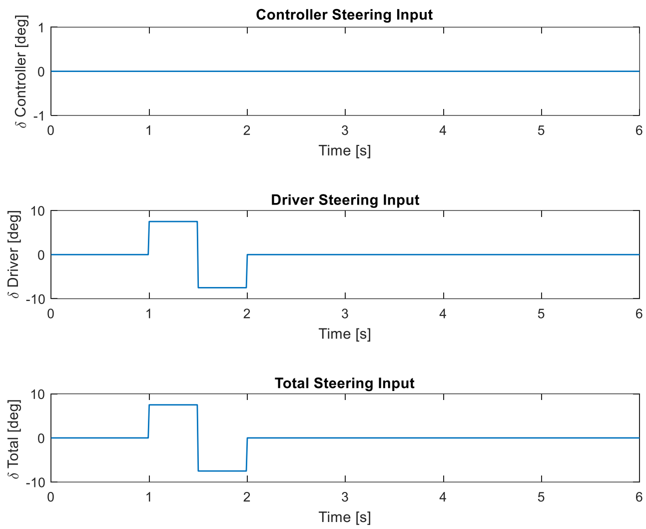

Figure 6-1: Human driver steering input for 2.5m lane-change maneuver

The "desired" trajectory is calculated based on the steady-state cornering assumption in which a no-slip kinematic relation exists between steering angle and wheel heading. The desired trajectory is shown in green in the figure below.

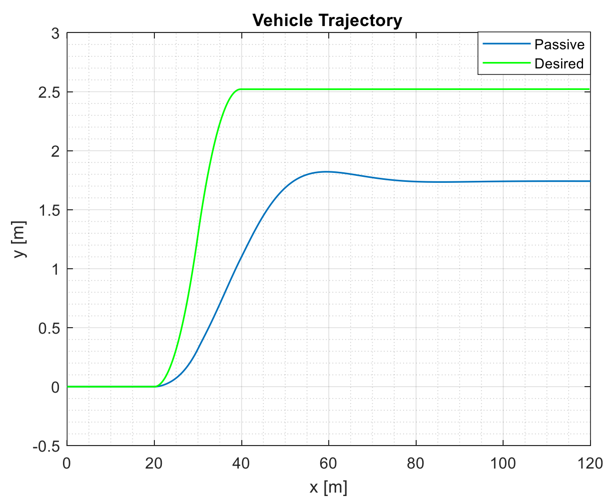

Figure 6-2: Passive maneuver and desired trajectory for 2.5m lane-change maneuver

The blue passive trajectory response indicates a side slip, and more steering is required to follow the path than what is predicted by the steering angle kinematics.

Here are the results of some other variables for the passive response simulation.

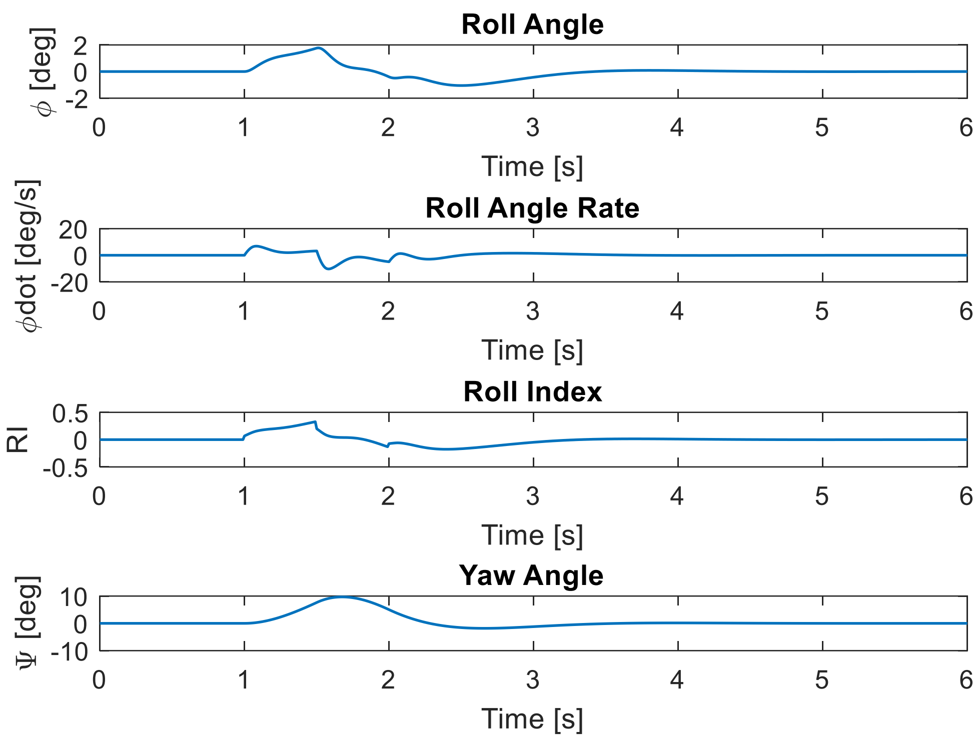

Figure 6-3: Passive lane-change maneuver results for roll angle, roll rate, roll index, and yaw angle

In this maneuver, the vehicle roll angle peaks around 2 degrees, and the roll index gets close to 0.5 at the time values of 1 and 2 seconds.

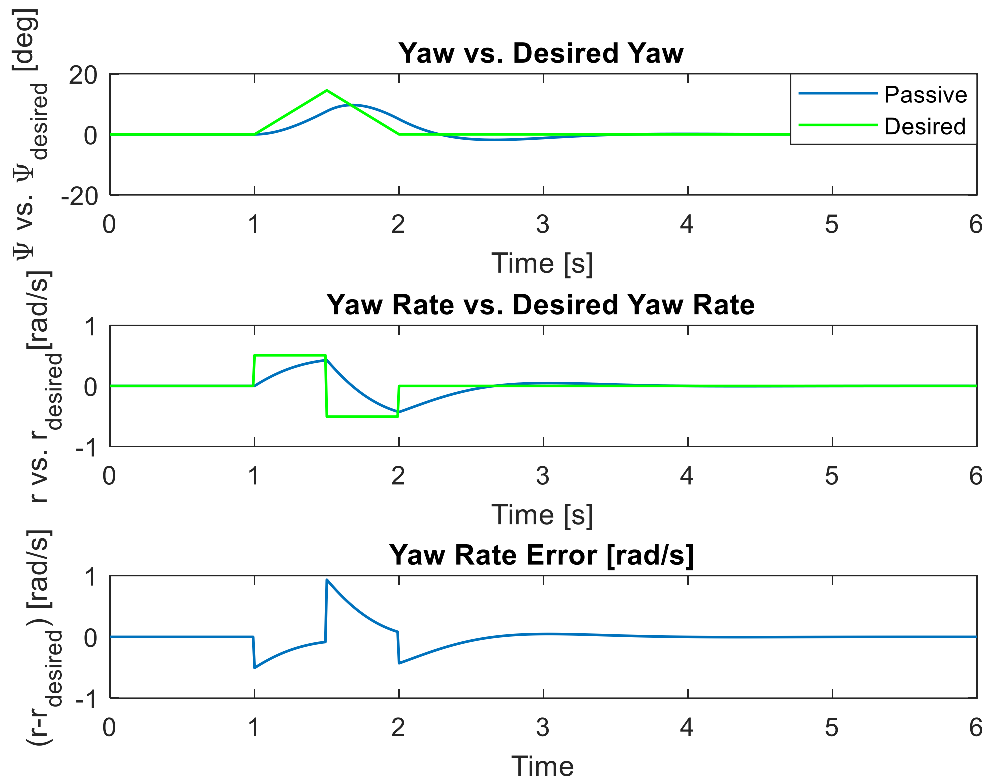

Figure 6-4: Yaw plots for passive lane-change maneuver

The nonzero yaw rate error indicates that the vehicle needs the help of the controller to regulate this error to improve tracking performance.

Figure 6-5 shows the normal and lateral forces at the left and right tires. Lateral tire forces are modeled as a linear function of slip angles using the model in section 3-6. The normal forces are the combination of the tires' linear stiffness and damping forces.

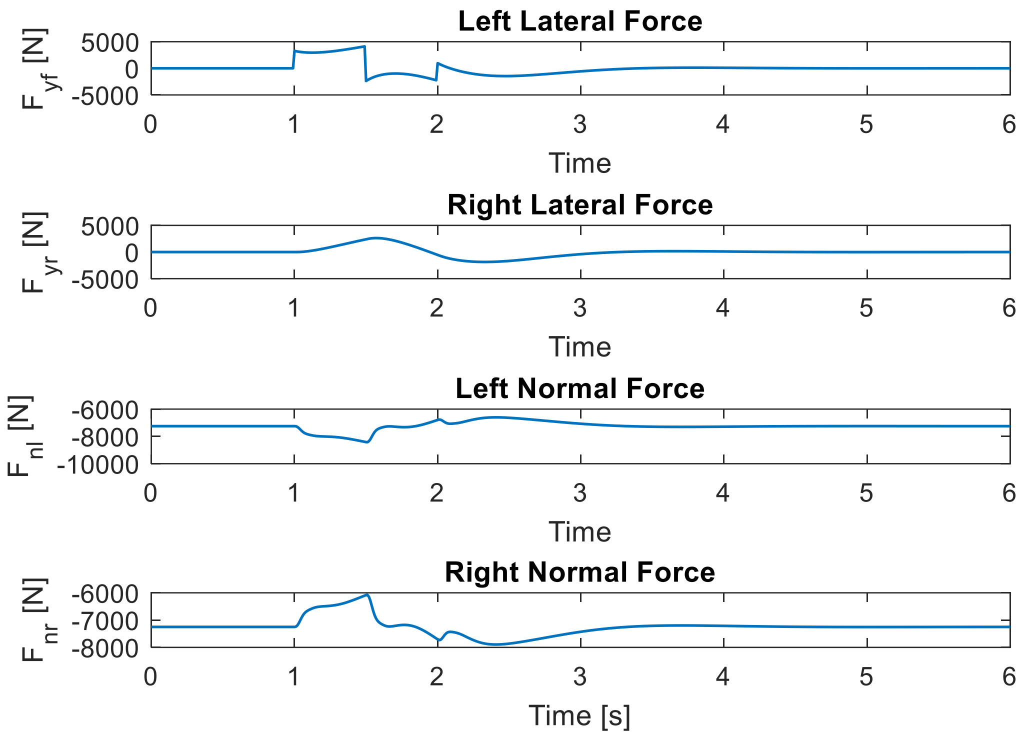

Figure 6-5: Lateral and normal tire forces for passive lane-change maneuver

Figure 6-6 shows the vertical dynamic performance of the model. Three merits of ride quality, rattle space, and road holding are monitored in this figure. The sprung mass acceleration is chosen as a merit for ride quality. The suspension deflections show the rattle space, and the road-holding performance is observed through the left and right tire deflections.

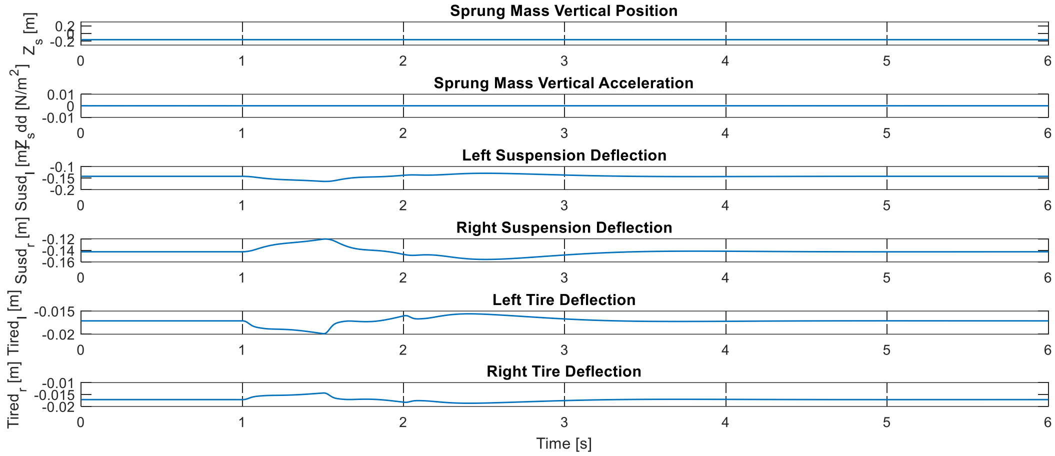

Figure 6-6: Vertical dynamics plots for passive lane-change maneuver

Figure 6-7 shows the lateral acceleration of the plant model in section 3-6 that is calculated as below:

$$a_y = \dot{v}_y + r.v_x \tag{6-1}$$

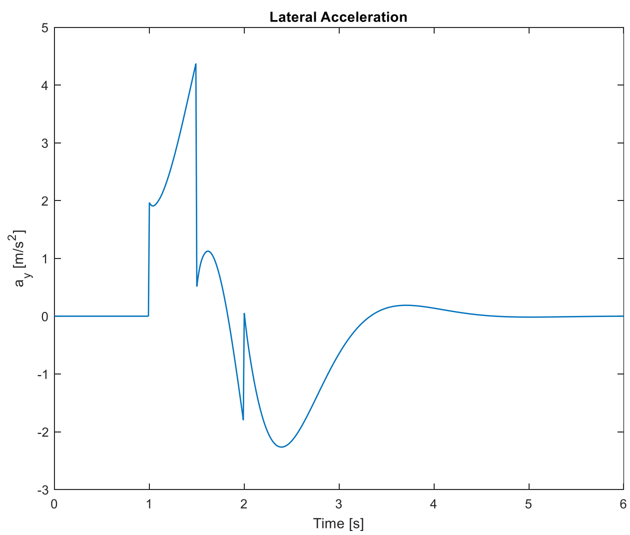

Figure 6-7: Lateral acceleration for passive lane-change maneuver

6.2 Active Steering (δ_{AS}) vs. Corrective Yaw Moment (M_{yc})

Both active steering and corrective yaw moment are commonly used for the purpose of yaw stability or tracking. Direct yaw control methods normally generate the corrective yaw moment. This moment can either be generated by differential braking or torque distribution. This dissertation models the corrective yaw moment as a direct control input and investigates the games between this control input and the active steering. The application of a direct yaw moment, while not a high-fidelity representation of how the moment would be delivered in a real vehicle, is sufficient to compare the new game theory methods' performance relative to decentralized methods. The objective for both players is defined to regulate the yaw rate error. The control model presented in chapter 3.3 is used to calculate the feedback gains for this scenario.

The passive results are shown with the color blue.

Four paradigms are simulated and compared with each other in this section.

- **One-player** δ_{AS}
- **One-player** M_{yc}
- **Two-player** $\delta_{AS} + M_{yc}$ **Decentralized**
- **Two-player** $\delta_{AS} + M_{yc}$ **Nash**

1. **One-player** δ_{AS}

In this paradigm, active steering is trying to track the desired yaw rate by regulating yaw rate error through additional steering input. The results for this simulation are plotted in red.

The output matrix C1, state weighting matrix Q1, and input weighting matrix R1 for this design are as follows:

```
C1=[0 0 0 1];                                                    (6-2)
Qb1=1;
Q1=C1'*Qb1*C1;
R1=6.25;
```

The simulation results are plotted for multiple values of R1. It appears that by reducing the R1 from 6.25 to 0.1, the tracking performance improves.

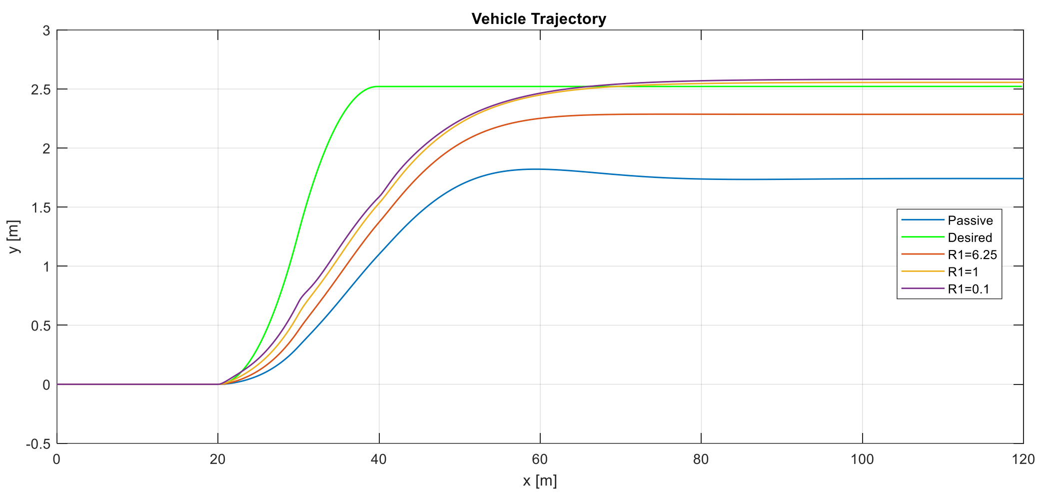

Figure 6-8: Trajectory simulation results for different values of input weight R1

But unfortunately, the peak value for active steering increases as well. As in real life, there is a limit to the amount of active steering that the controller can provide; The weight *R1* is picked to prevent controller saturation. For instance, at 1 sec, the steering input 86.47 [deg] (for *R1*=0.1) and 23.7 [deg] (for *R1*=1) are unrealistic values (we like to keep the active steering input less than 12-15 deg).

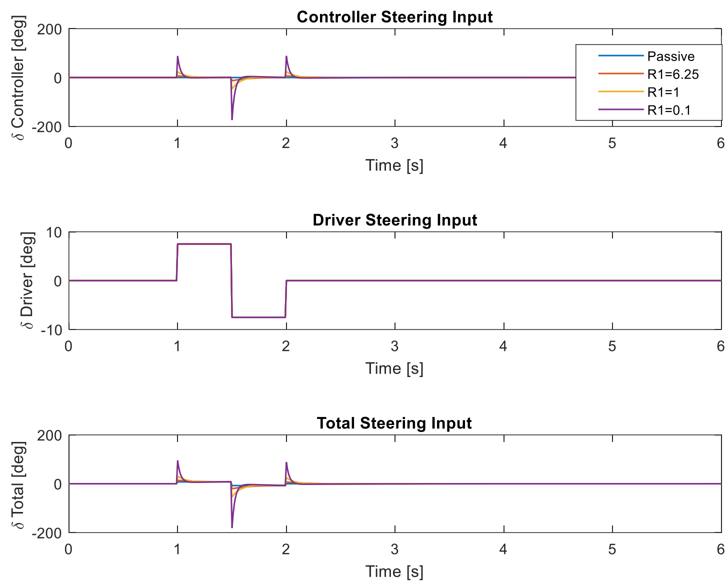

Figure 6-9: Active steering control action for different values of R1

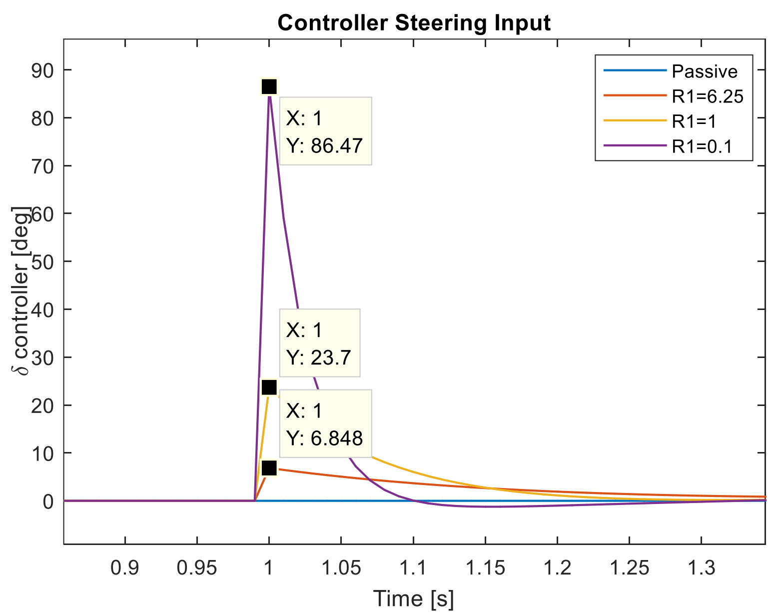

Figure 6-10: Zoomed-in version of figure 6-9 to read the controller steeing input peak values

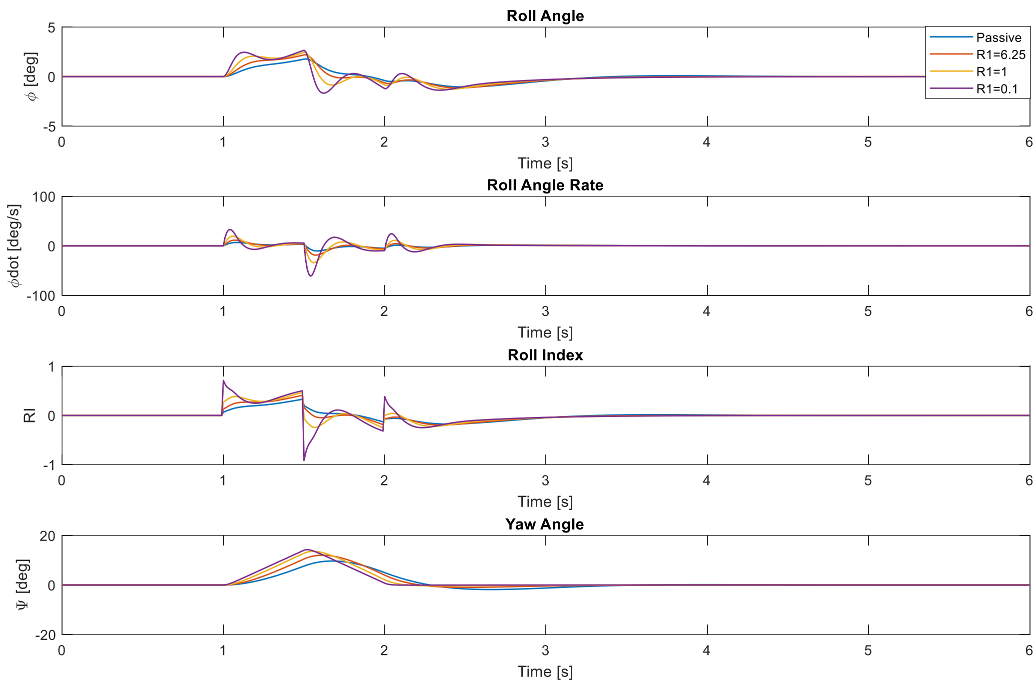

Figure 6-11: Roll index for different values of R1

The roll index plot also shows that once the value of *R1* goes below 6.25, the peak of the roll index goes over 1, which means the vehicle is encountering rollover. The value of *R1*=6.25 is chosen as the final design to avoid saturation and rollover. For this value, the tracking improves compared to passive, but still, there is a steady-state lateral error. This error can be reduced by adding the corrective yaw moment, which encourages the application of a two-player game.

By choosing the *R1* = 6.25, the optimal gain is calculated as follows.

Optimal gain: $K_\delta = [-0.0090 \quad -0.0019 \quad 0.0079 \quad 0.2358]$

RMS of steering input signal delta: $\delta_{RMS} = 0.0325\ rad$

Total cost: $J_{total} = 0.2124$

The next sections show single-player and two-player chassis control scenarios. The results will show that additional players' control actions are required for better control, thus motivating this thesis to pursue the game-theoretic approach in multi-player games for $N \geq 3$.

2. One-player M_{yc}

In this scenario, active yaw control (*DYC*) is trying to track the desired yaw rate by regulating yaw rate error. The results for this simulation are plotted in yellow. The output matrix C1, state weighting matrix Q1, and input weighting matrix R1 for this design are as follows:

```
C1=[0 0 0 1];                                    (6-3)
Qb1=1;
Q1=C1'*Qb1*C1;
R1=1e-10;
```

Optimal gain:

$K_{Myc} = 10^4 \times [-0.0487 \quad -0.0053 \quad 0.0606 \quad 9.4749]$

RMS of corrective yaw moment M_{yc}: $Myc_{RMS} = 6.2835 \times 10^3 \; N.m$

Total cost: $J_{total} = 0.2725$

3. Two-player $\delta_{AS} + M_{yc}$ Decentralized

In this scenario, active steering and yaw control (*DYC*) track the desired yaw rate by regulating yaw rate error in a decentralized combination. This means that each player is playing on their own without knowing the control action of the other player. The results for this simulation are plotted in purple

This scenario combines the previous one-player scenarios, so the same optimal weights (Q and R s) and gains (K_δ, K_{Myc}) derived from the single-player are used in this paradigm to make a meaningful comparison.

```
C1=[0 0 0 1]; %for delta                    (6-4)
regulating er
Qb1=1;
Q1=C1'*Qb1*C1;
R1=6.25;

C2=[0 0 0 1]; %for M_yc
regulating er
Qb2=eye(2);
Q2=C2'*Qb1*C2;
R2=1e-10;
```

Optimal gains:

K_δ =[-0.0090 -0.0019 0.0079 0.2358]

K_{Myc} $=10^4 \times$[-0.0487 -0.0053 0.0606 9.4749]

RMS of control inputs signals:

$\delta_{RMS} = 0.0147\ rad$

$Myc_{RMS} = 6.1986\times10^3\ N.m$

The RMS value of the active steering input is lower in the decentralized paradigm than an individual corresponding signal in the one-player game. This means that these players, in a sense, are reducing loads of work from each other. The RMS value of corrective yaw moment is slightly less for the two-player game, but the difference is not significant.

Total cost: $J_{total} = 0.2816$

4. Two-player $\delta_{AS} + M_{yc}$ Nash

In this scenario, active steering and yaw control (DYC) track the desired yaw rate by regulating yaw rate error in a Nash paradigm. This means that players play a two-player differential game that is solved using the two-player Nash solution. The results for this simulation are plotted in green. Weight matrices Q's are the same as one-player and decentralized. R11 and R22 are also chosen the same as R1 and R2 in the decentralized paradigm.

```
C1=[0 0 0 1];  %for delta regulating er                    (6-5)
C2=[0 0 0 1];  %for Myc regulating er
Qb1=eye(1);
Qb2=eye(1);
Q1=C1'*Qb1*C1;
Q2=C2'*Qb2*C2;
R11=6.25;
R12=0;
R21=0;
R22=1e-10;
```

Optimal gains:

K_{δ} =[0.0001   0.0000   0.0000   0.0225]

K_{Myc} = $10^4 \times$ [-0.0484   -0.0052   0.0600   9.4147]

RMS of control inputs signals:

$\delta_{RMS} = 0.0015 \ rad$

$Myc_{RMS} = 6.2426\times10^3 \ N.m$

Total cost: $J_{total} = 0.2723$

Table 6-1:Active steering vs. corrective yaw moment scenario RMS values and total cost

	One-player δ_{AS}	One-player M_{yc}	Two-player decentralized	Two-player Nash
δ_{RMS}	0.0325	0	0.0147	0.0015
Myc_{RMS}	0	6.2835×10^3	6.1986×10^3	6.2426×10^3
Total cost J_{total}	0.2124	0.2725	0.2816	0.2723

Both Nash and decentralized two-player games scored lower RMS than one-player scenarios for each control input. A significant 90 % reduction in the RMS value of steering input control signal in Nash can be seen by comparing the decentralized and Nash paradigms. RMS of corrective yaw moment control input slightly increased 1% in Nash.

By zooming into the active steering input plot (fig 6-15), the peak of the steering input signal in Nash is also reduced considerably, and the slight change in the peak of yaw moment control input is negligible (fig 6-6). In a scenario where the steering input is limited to realistic values, the corrective yaw moment can be used in a Nash configuration to help reduce the peak of the steering input and achieve the same tracking result. The results show that Nash is spending less energy and requires lower control inputs to achieve the same goal of tracking than the decentralized approach, which means that the Nash solution is optimal.

The Nash controller's total cost function is also lower than that of Decentralized and One-player M_{yc} controllers. The One-player δ_{AS} scored lower, but with the disadvantage of diminished tracking performance.

Figure 6-12 shows that all controllers are increasing roll angle, roll rate, and roll index, which shows that the goal of tracking is against the goal of roll regulation

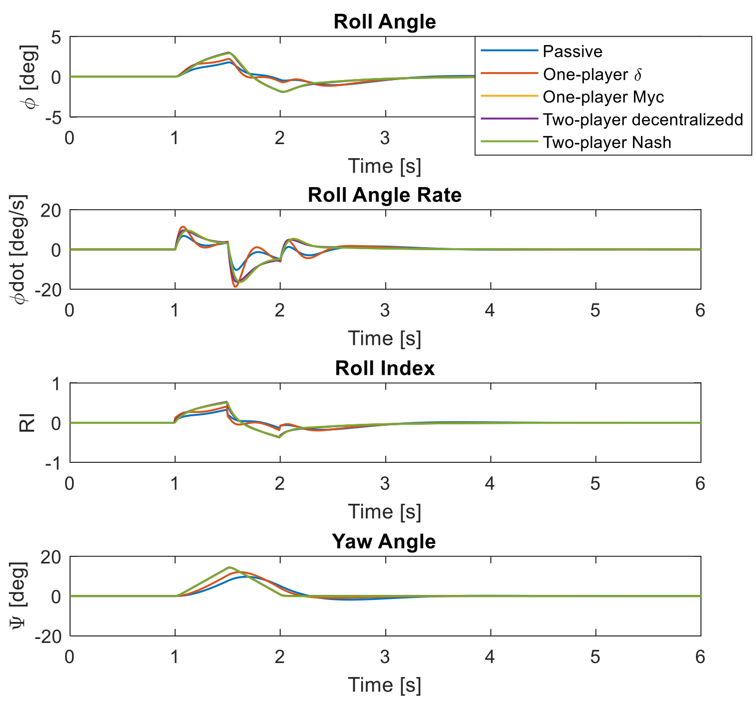

Figure 6-12: Simulation results for roll angle, roll rate, roll index, and yaw angle (active steering vs. corrective yaw moment game)

Figure 6-13 shows that active steering is not showing enough tracking improvement by itself relative to all the other controllers.

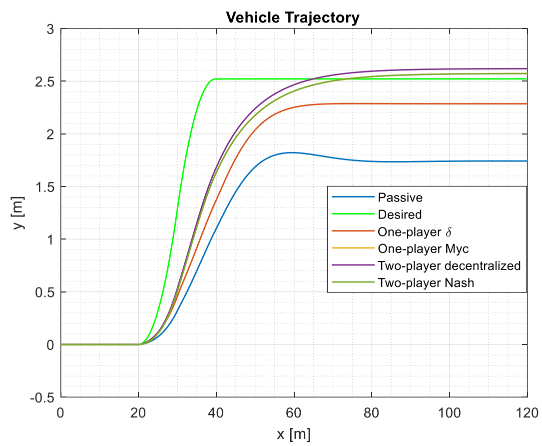

Figure 6-13: Vehicle trajectory (active steering vs. corrective yaw moment game)

The human driver steering input, active steering control input and the combination of them (total steering input) are shown in degrees in figure 6-14. The control signal amount is lower for two-players scenarios relative to one-player δ_{AS}.

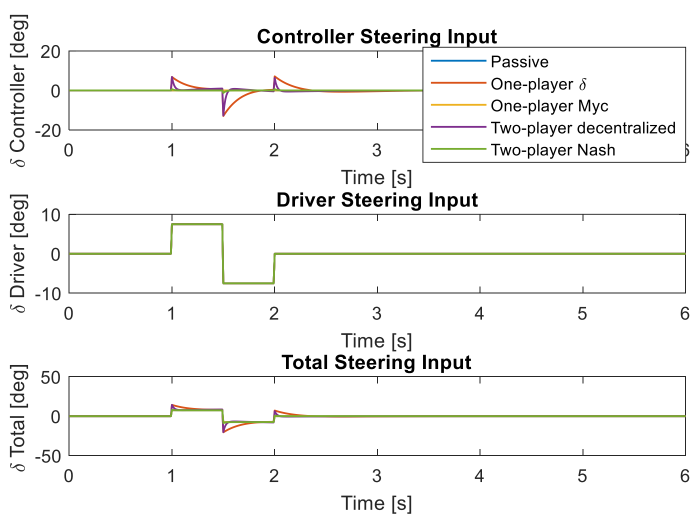

Figure 6-14: Steering inputs for human driver and controller (active steering vs. corrective yaw moment game)

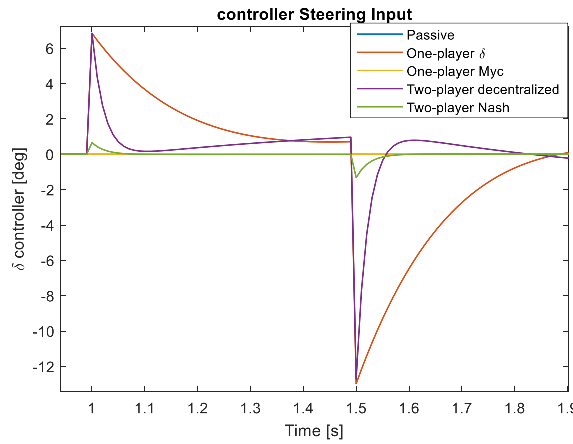

Figure 6-15: Active steering input signal zoomed-in

Nash shows almost 90% less steering input control signal (δ_{AS}) required than all other scenarios (except one-player M_{yc} where δ_{AS} is zero).

In figure 6-16, there is a nearly identical corrective yaw moment for all scenarios, with the exception of the one-player steering controller that does not apply M_{yc}.

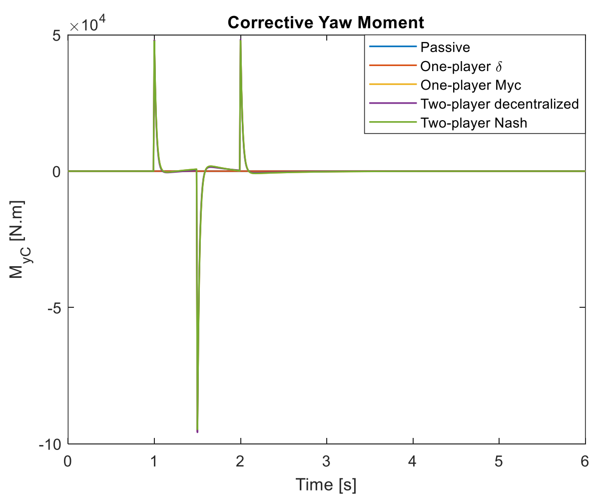

Figure 6-16: Corrective yaw moment (active steering vs. corrective yaw moment game)

Both two-player paradigms generate a higher lateral acceleration than the one-player δ_{AS} , which is due to the involvement of corrective yaw moment.

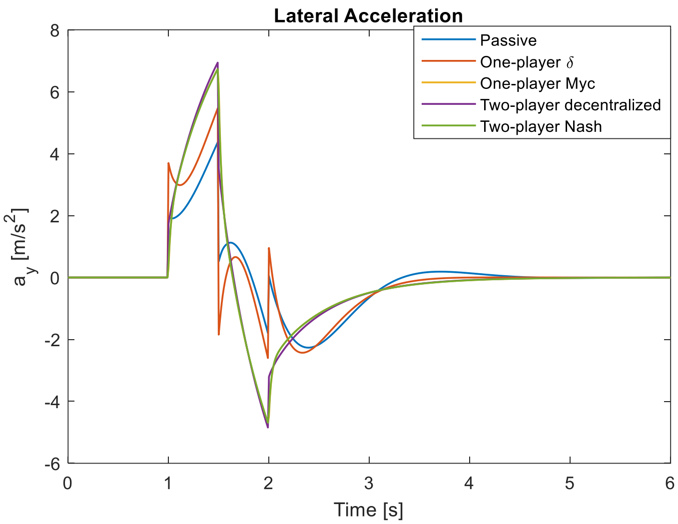

Figure 6-17: Lateral acceleration (active steering vs. corrective yaw moment game)

Conclusion:

The importance of the cooperative game between active steering and corrective yaw moment appears when practical limits must be placed on the steering input control signal. Due to this limit, high tracking performance is not achievable with only steering input. By adding the corrective yaw moment as the second player, the burden of control is divided between the players. In a two-player game, the Nash paradigm achieves the same tracking performance as decentralized using less control effort and lower cost function, which makes it the preferred paradigm for this scenario.

6.3 Active steering (δ_{AS}) vs. Corrective roll moment ($M_{\phi c}$)

In a game between these two players, the corrective roll moment's sole objective is improving roll stability by regulating roll angle and roll rate. Meanwhile, the cost function for active steering can be designed for two opposing purposes. Active steering can either minimize yaw rate error (investigated in the game between active steering and corrective roll moment for the objective of tracking) or be used as counter-steering to regulate the roll angle and roll angle rate. In counter steering, the active steering reduces the lateral acceleration to regulate the roll angle and roll rate, which means that the vehicle will lose its path tracking ability. This section investigates these two opposing objectives in two formats of non-competing and competing games between the active steering and corrective roll moment. The control model introduced in chapter 3.3 is used to calculate the feedback gains, and simulation results are plotted for the plant model presented in 3.6.

6.3.1 Non-competing:

The objective for both players is to regulate roll angle and roll rate.

- **One-player** δ_{AS}
- **One-player** $M_{\phi c}$
- **Two-player** $\delta_{AS} + M_{\phi c}$ **Decentralized**
- **Two-player** $\delta_{AS} + M_{\phi c}$ **Nash**

1. **One-player** δ_{AS}

In this paradigm, active steering is trying to regulate roll and roll rate by counter steering. Due to the loss of lateral acceleration, the vehicle will lose its tracking performance. The results for this simulation are plotted in red.

The output matrix *C1*, state weighting matrix *Q1*, and input weighting matrix *R1* for this design are as follows:

```
C1=[1 0 0 0;  %for delta regulating [phi,phid]      (6-6)
    0 1 0 0];
Qb1=eye(2);
Q1=C1'*Qb1*C1;
R1=0.001;
```

Optimal gain: K_δ =[24.0123   30.8732   -0.0931   0.0249]

RMS of steering input signal delta: δ_{RMS} = 0.0506 *rad*

Total cost: $J_{total} = 3.6008 \times 10^{-5}$

2. One-player $M_{\phi c}$

In this scenario, active roll moment control is trying to regulate roll and roll rate. The results for this simulation are plotted in yellow, and the weighting matrices are as follows.

```
C1=[1 0 0 0;  %for Mφc regulating [phi,phid]      (6-7)
     0 1 0 0];
 Qb1=eye(2);
 Q1=C1'*Qb1*C1;
```

```
R1=1e-14;
```

Optimal gain: K_{Mzc} = 10^6 ×[9.9455 9.9950 -0.0007 0.0002]

RMS of corrective yaw moment $M_{\phi c}$**:** $M\phi c_{RMS}$ = 533.7205 $N.m$

Total cost: J_{total} = 3.0447×10^{-8}

3. Two-player $\delta_{AS} + M_{\phi c}$ Decentralized

In this scenario, both active steering and active roll moment are regulating roll angle and roll rate error in a decentralized combination. This means that each player is playing on their own. The results for this simulation are plotted in purple.

This scenario is a combination of the previous two single-player scenarios δ and the roll moment, so the same optimal weights (Q and R s) and gains (K_{δ}, $K_{M\phi c}$) derived from the single-player are used in this paradigm.

(6-8)

```
C1=[1 0 0 0;  %for delta regulating [phi,phid]
    0 1 0 0];
Qb1=eye(2);
Q1=C1'*Qb1*C1;
R1=0.001;

C2=[1 0 0 0;  %for Mφc regulating [phi,phid]
    0 1 0 0];
Qb2=eye(2);
Q2=C2'*Qb1*C2;
```

```
R2=1e-14;
```

Optimal gains:

K_{δ} =[24.0123   30.8732   -0.0931   0.0249]

$K_{M\phi c}$ = 10^6 ×[9.9455   9.9950   -0.0007   0.0002]

RMS of control inputs signals:

δ_{RMS} = 0.0573 *rad*

$M\phi c_{RMS}$ = 472.8298 *N.m*

The RMS values of each control inputs are lower in the decentralized paradigm than an individual corresponding signal in the one-player game. The peak for both control inputs delta and corrective roll moment is reduced. This means that these players, in a sense, are reducing loads of work from each other.

Total cost: $J_{total} = 1.9786\times10^{-5}$

4. Two-player $\delta_{AS} + M_{\phi c}$ Nash

In this scenario, both active steering and corrective roll moment improve roll stability by regulating roll and roll rate in a Nash paradigm. This means that players play a two-player differential game that is solved using the two-player Nash solution. The results for this simulation are plotted in green. Weight matrices *Q* are the same as in the decentralized scenario. *R11* and *R22* are also chosen to be the same as *R1* and *R2* in the decentralized paradigm.

```
C1=[1 0 0 0;  %for delta regulating [phi,phid]
```
(6-9)

```
    0 1 0 0];
C2=[1 0 0 0;  %for Mφc regulating [phi,phid]
    0 1 0 0];
Qb1=eye(2);
Qb2=eye(2);
Q1=C1'*Qb1*C1;
Q2=C2'*Qb2*C2;
R11=.0001;
R12=3e-13;
R21=1e-3;
R22=1e-12;
```

Optimal gains:

$K_\delta = [34.0711 \quad 35.7702 \quad -0.0147 \quad 0.0572]$

$K_{M\phi c} = 10^6 \times [1.2223 \quad 1.2834 \quad -0.0008 \quad 0.0006]$

RMS of control inputs signals:

$\delta_{RMS} = 0.0101 \ \ rad$

$M\phi c_{RMS} = 443.6340 \ N.m$

Total cost: $J_{total} = 3.1223 \times 10^{-6}$

The table below shows the total cost and the RMS of the control input signal for the simulation time.

Table 6-2: Active steering vs. corrective roll moment (non-competing) scenario RMS values and total cost

	One-player δ_{AS} for roll	One-player $M_{\phi c}$	Two-player decentralized	Two-player Nash
δ_{RMS}	0.0506	0	0.0573	0.0101
$M\phi c_{RMS}$	0	533.7205	472.8298	443.6340
Total cost J_{total}	3.6008×10^{-5}	3.0447×10^{-8}	1.9786×10^{-5}	3.1223×10^{-6}

Both two-players paradigms (decentralized and Nash) have lower RMS for individual control signals than the corresponding active signals in single-player ones. This means that the control burden is shared between the players in a two-player paradigm.

The total cost in Nash is slightly less than the decentralized, with significantly lower RMS for steering input and somewhat higher RMS for roll moment for the current set of weights.

Simulation results for the lane-change maneuver

The decentralized combination has a better roll index performance, as shown in Figure 6-18.

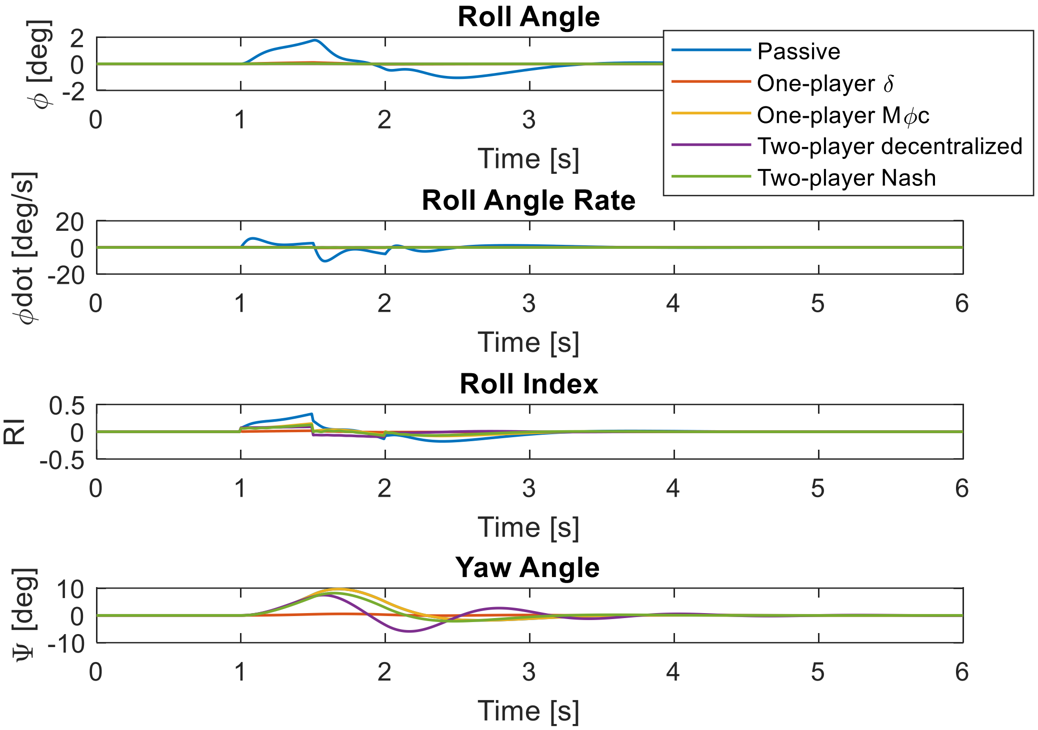

Figure 6-18: Simulation results for roll angle, roll rate, roll index, and yaw angle (active steering vs. corrective roll moment non-competing game)

In terms of roll angle and roll rate, performances are the same. The decentralized shows better performance than the rest when it comes to the roll index.

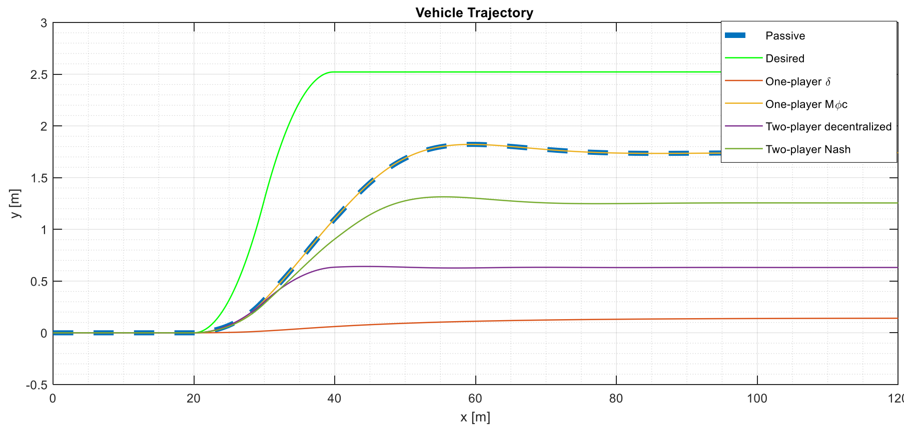

Figure 6-19: Vehicle trajectory (active steering vs. corrective roll moment non-competing game)

According to Figure 6-19, One-player δ loses tracking performance completely due to counter-steering and is incapable of regulating the yaw rate error according to Figure 6-20. The negative effect of counter-steering reduces in two-player paradigms. Nash shows relatively less loss of tracking than decentralized while maintaining the same roll performance, which is an improvement.

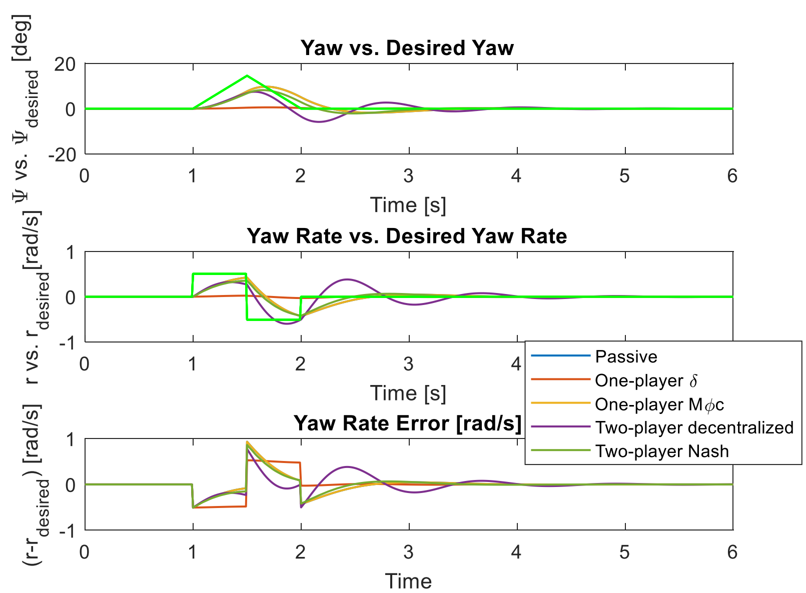

Figure 6-20: Yaw plots (active steering vs. corrective roll moment non-competing game)

Figure 6-21 shows that the peak of active steering control input reduces from one-player δ to two-player paradigms. The Nash paradigm has the lowest amplitude for active steering control input. This means that Nash requires less control effort to achieve the same objective.

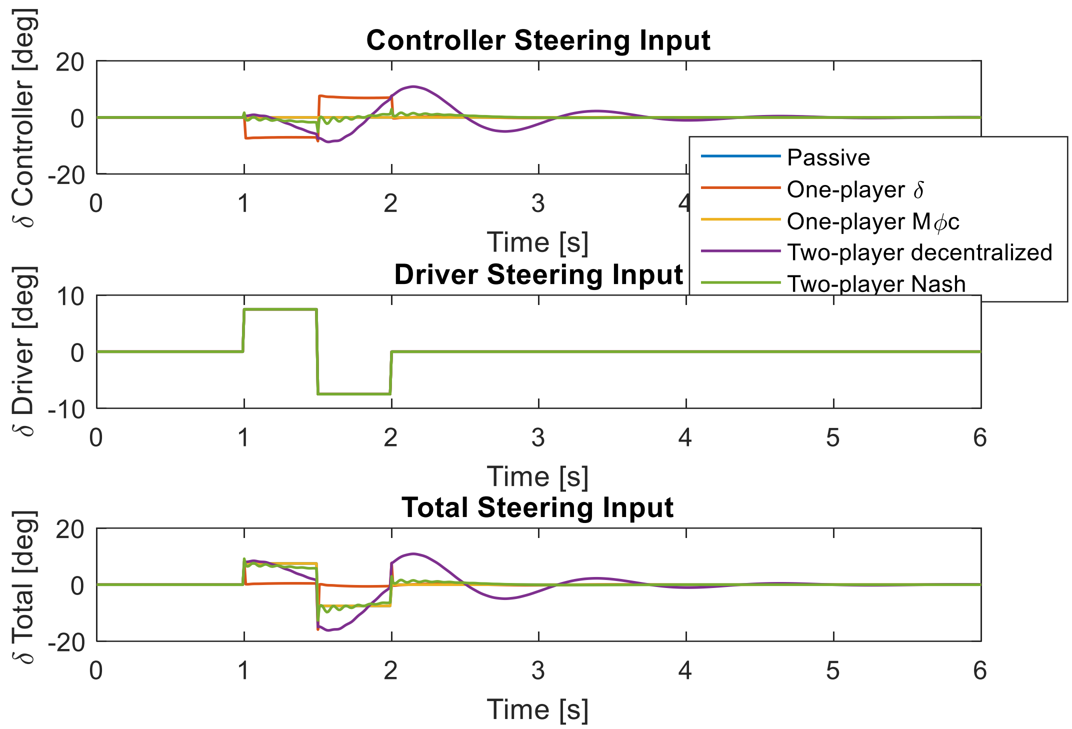

Figure 6-21: Human and active steering control inputs (active steering vs. corrective roll moment non-competing game)

As shown in Figure 6-22, the peak for corrective roll moment input is lower in both two-player paradigms than the one-player. The Nash paradigm has a smoother input signal than the rest of the paradigms.

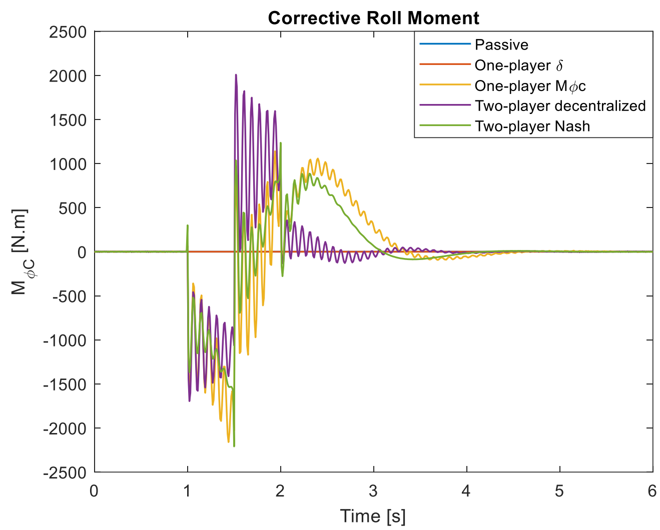

Figure 6-22: Corrective roll moment (active steering vs. corrective roll moment non-competing game)

According to Figure 6-23, the tire lateral forces for Nash are closer to the ones for one-player $M_{\phi c}$ (shown in plots as M_{zc}). The normal tire forces for Nash are also smoother relative to the other paradigms. The same smooth transition is also shown in tire deflections in Figure 6-24.

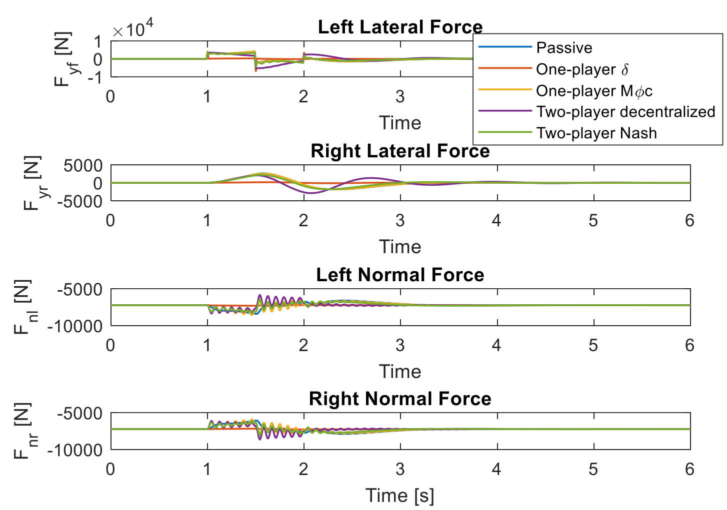

Figure 6-23: Lateral and normal tire forces (active steering vs. corrective roll moment non-competing game)

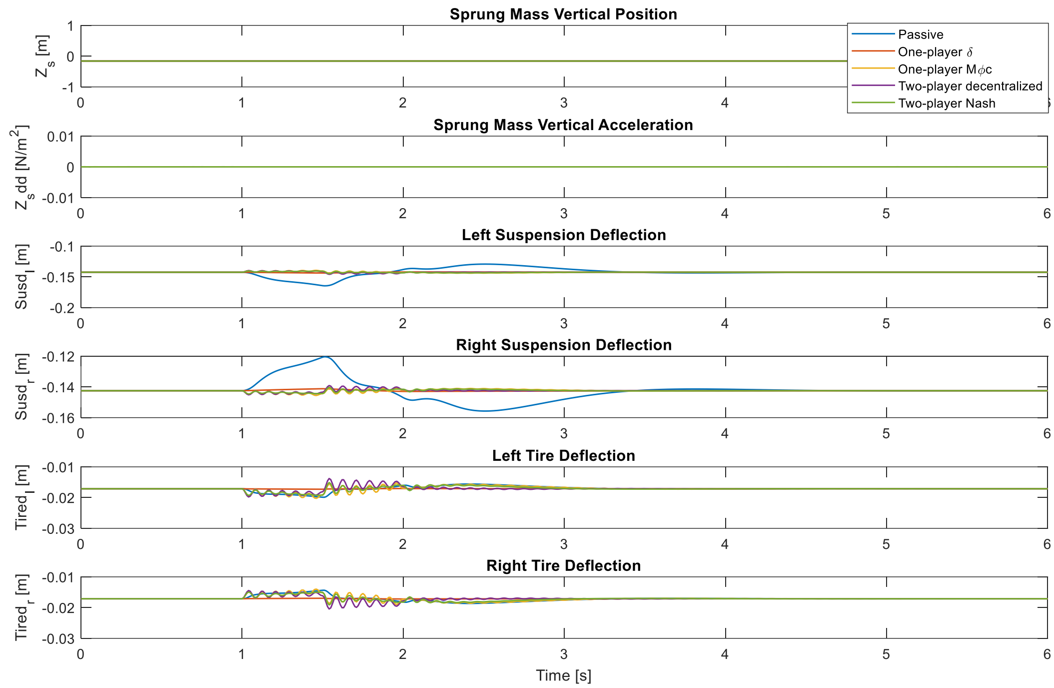

Figure 6-24: Vertical dynamic simulation results (active steering vs. corrective roll moment non-competing game)

Figure 6-25 shows that the Nash paradigm generates higher lateral acceleration than the decentralized but has a lower value than one-player $M_{\phi c}$.

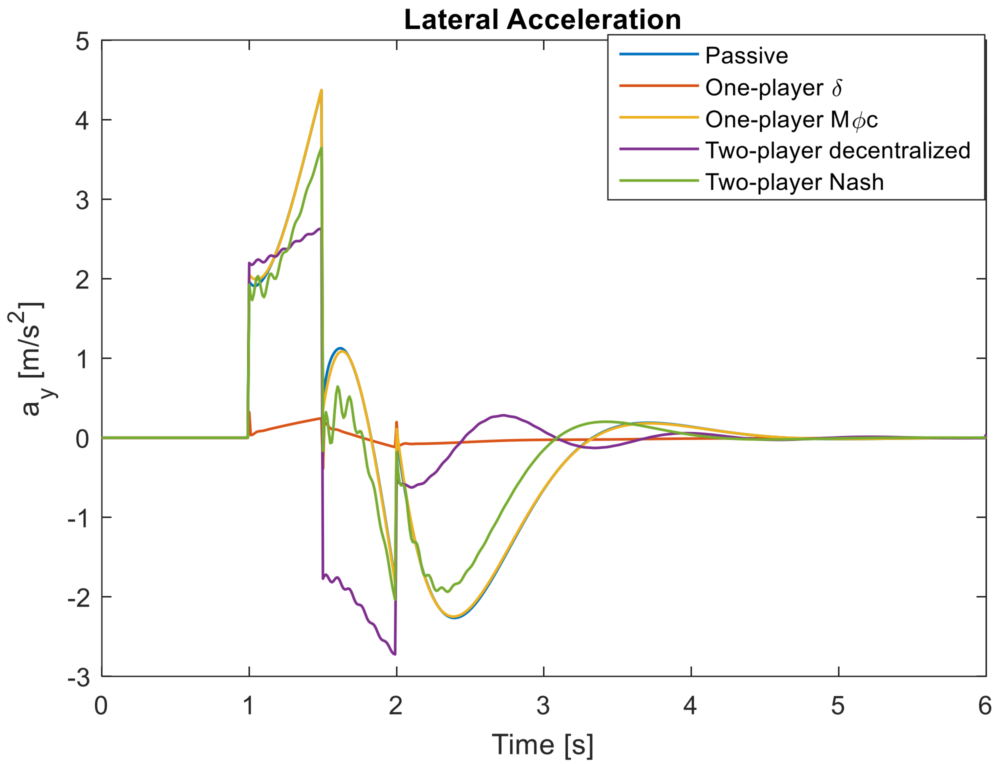

Figure 6-25: Lateral acceleration (active steering vs. corrective roll moment non-competing game)

In summary, it is shown that countersteering can help improve the roll stability, but in general, the corrective roll moment is the dominating strategy, and even in the two-player game, the Nash strategy is closer to the corrective roll moment strategy. It was also shown that the corrective roll moment control input signal, the tire deflections, and normal forces transitions are smoother in Nash relative to other paradigms.

6.3.2 Competing:

In this scenario, the objective for active steering is to track the desired path by regulating the yaw rate error, and the corrective roll moment is regulating roll angle and roll rate. In the previous section, active steering played a cooperative game with corrective roll moment and created counter-steering to improve roll stability. In this scenario, the active steering pursues its own individual goal which is regulating the yaw rate error to improve the tracking. As it is shown in Figure 6-26, this increases the roll angle, which is against the goal of the corrective roll moment. The competition between these two players is investigated in the two-player paradigms. Here is the list of paradigms simulated for this section.

- **One-player** δ_{AS}
- **One-player** $M_{\phi c}$
- **Two-player** $\delta_{AS} + M_{\phi c}$ **Decentralized**
- **Two-player** $\delta_{AS} + M_{\phi c}$ **Nash**

Passive results are shown is blue.

1. **One-player** δ_{AS}

The active steering goal is to regulate yaw rate error. The results for this simulation are plotted in red. The designed weight matrices are as follows.

```
C1=[0 0 0 1]; %for delta regulating er
Qb1=1;
Q1=C1'*Qb1*C1;
```
(6-10)

```
R1=6.25;
```

K_{δ}=[-0.0090 -0.0019 0.0079 0.2358]

δ_{RMS}=0.0325 *rad*

J_{total} =0.2124

2. One-player $M_{\phi c}$

Corrective roll moment improves roll stability by regulating roll angle and roll rate. The results for this simulation are plotted in yellow, and the weight matrices are.

```
C1=[1 0 0 0;  %for Mφc regulating [phi,phid]          (6-11)
    0 1 0 0];
Qb1=eye(2);
Q1=C1'*Qb1*C1;
R1=1e-14;
```

Here are the optimal gain, root mean square of the moment signal, and total cost:

$K_{M\phi c}$ =10^6 ×[9.9455 9.9950 -0.0007 0.0002]

$M\phi c_{RMS}$ =533.7205

J_{total} = 3.0447×10^{-8}

3. Two-player δ_{AS} + $M_{\phi c}$ Decentralized

This paradigm is the combination of previous paradigms. The results for this simulation are plotted in purple. The weighting matrices are selected the same as one-player scenarios, which create the same optimal gains.

```
C1=[0 0 0 1]; %for delta regulating e                    (6-12)
 Qb1=1;
 Q1=C1'*Qb1*C1;
 R1=6.25;
 C2=[1 0 0 0;  %for Mφc regulating [phi,phid]
     0 1 0 0];
 Qb2=eye(2);
 Q2=C2'*Qb1*C2;
 R2=1e-14;
```

K_{δ} = [-0.0090 -0.0019 0.0079 0.2358]

$K_{M\phi c}$ =10^6 ×[9.9455 9.9950 -0.0007 0.0002]

δ_{RMS} = 0.0326 *rad*

$M\phi c_{RMS}$ = 711.9162

J_{total} = 0.2119

4. Two-player $\delta_{AS} + M_{\phi c}$ Nash

The weight matrices are designed as follows. The results for this simulation are plotted in green.

```
C1=[0 0 0 1]; %for delta regulating er                (6-13)
C2=[1 0 0 0;  %for Mφc regulating [phi,phid]
     0 1 0 0];
Qb1=1;
Qb2=eye(2);
Q1=C1'*Qb1*C1;
Q2=C2'*Qb2*C2;
R11=6.25;
R12=0;
R21=.01;
R22=1e-14;
```

K_{δ} = - [0.0021 -0.0022 0.0080 0.2353]

$K_{M\phi c} = 10^6 \times$[9.9455 9.9950 -0.0007 -0.0014]

$\delta_{RMS} = 0.0325\ rad$

$M\phi c_{RMS} = 704.4610\ N.m$

$J_{total} = 0.2116$

Table 6-3: Active steering vs. corrective roll moment (competing) scenario RMS values and total cost

	One-player δ_{AS} for track	One-player $M_{\phi c}$ for roll	Two-player decentralized	Two-player Nash
δ_{RMS}	0.0325	0	0.0326	0.0325
$M\phi c_{RMS}$	0	533.7205	711.9162	704.4610
Total cost J_{total}	0.2124	3.0447×10^{-8}	0.2119	0.2116

For the selected set of weights, the results of Decentralized and Nash overlap each other on plots, and they show almost identical performances in all the areas.

The two player-game paradigms only combine the effects of both single players linearly. No noticeable improvement was observed in either the cost function or the RMS or control inputs' peaks (Figure 6-29 and Figure 6-30).

As shown in Figure 6-26, the one-player active steering increases roll angle, roll rate, and roll index. This problem is solved by adding the corrective roll moment as the second player to the game

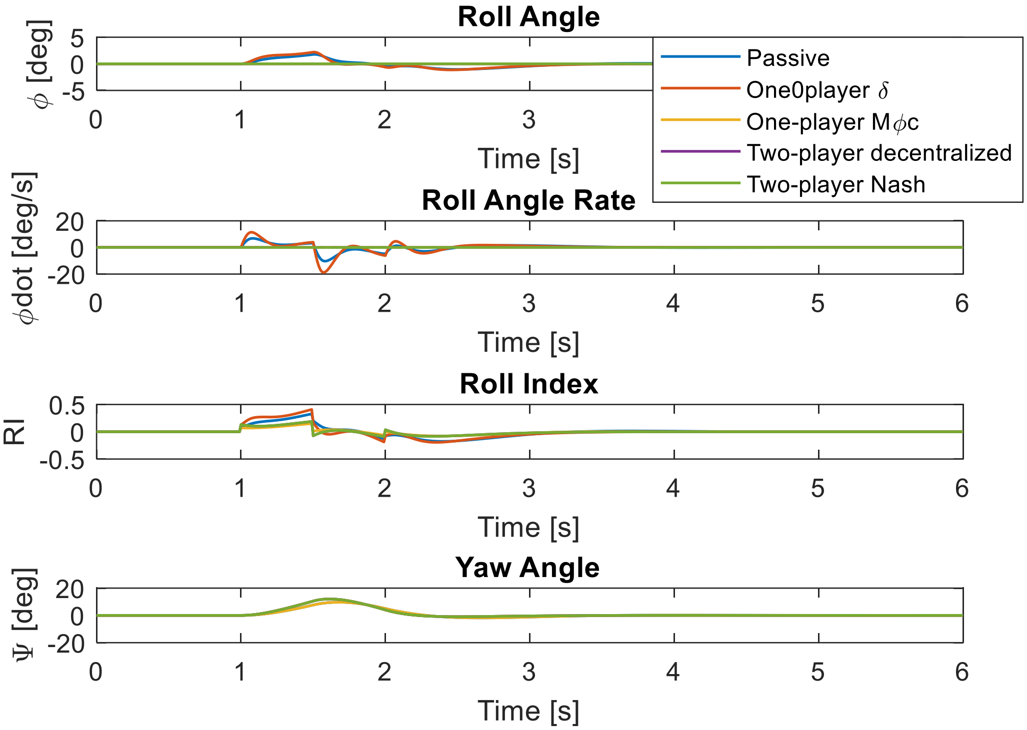

Figure 6-26: Simulation results for roll angle, roll rate, roll index, and yaw angle (active steering vs. corrective roll moment competing game)

Figure 6-27 shows that the corrective roll moment as a one-player shows lower tracking performance relative to other paradigms. Adding the active steering as the second player to the game improves minimizes the yaw rate error Figure 6-28 and improves tracking.

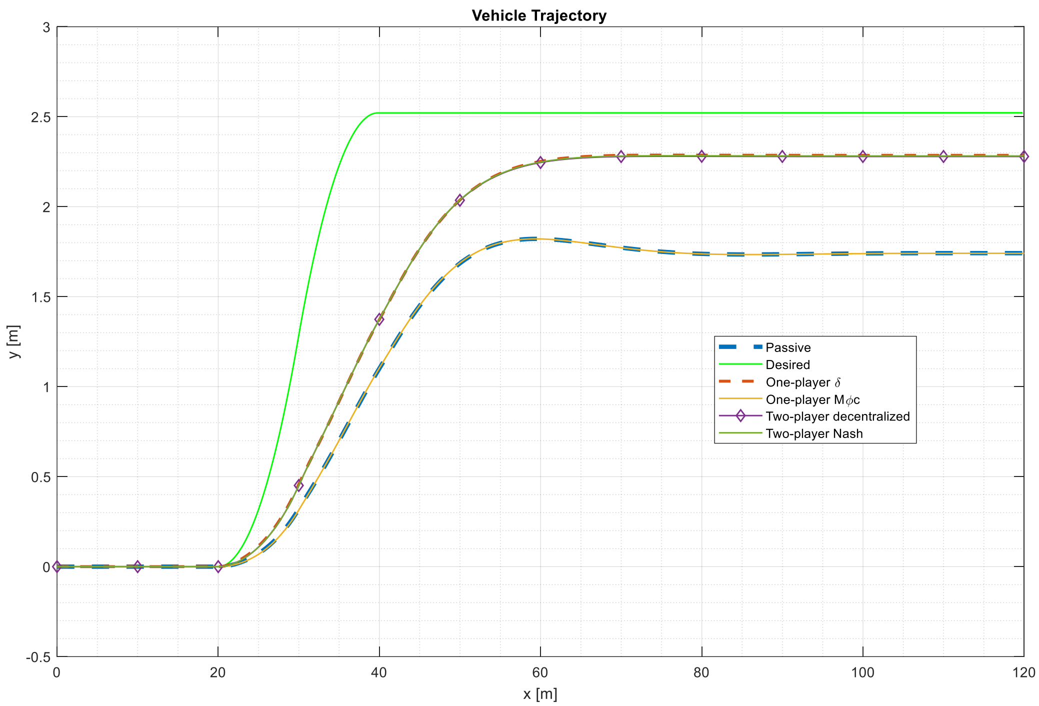

Figure 6-27: Vehicle trajectory angle (active steering vs. corrective roll moment competing game)

There is still a steady-state lateral error. One way to improve the remaining steady-state lateral error is by introducing corrective yaw moment as the third player.

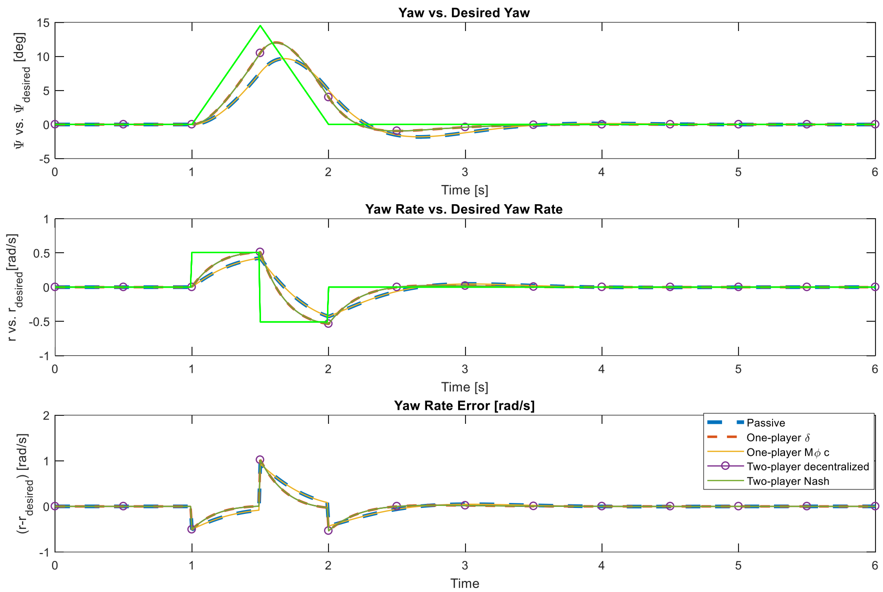

Figure 6-28: Yaw plots angle (active steering vs. corrective roll moment competing game)

Figure 6-29 and Figure 6-30 shows almost identical control inputs for all the paradigms that generate signals.

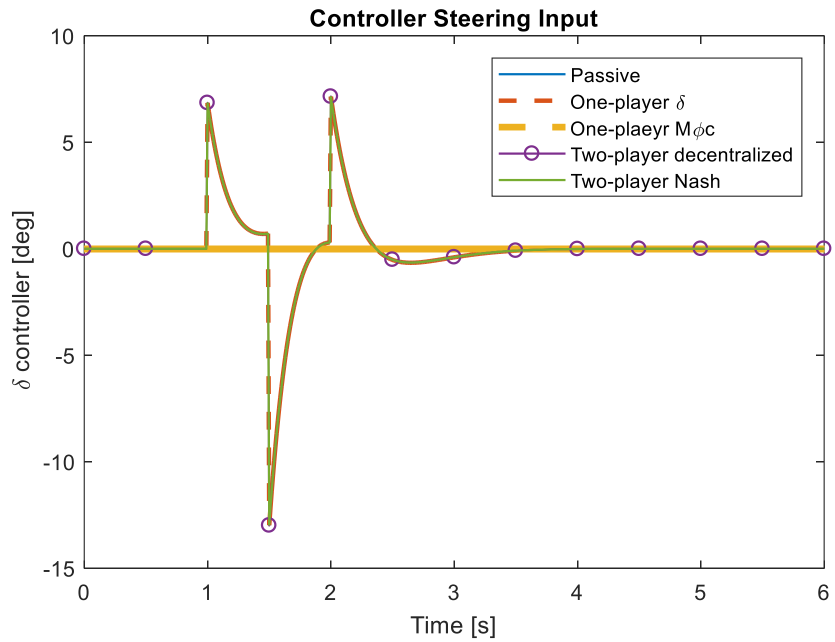

Figure 6-29: Human and active steering angles angle (active steering vs. corrective roll moment competing game)

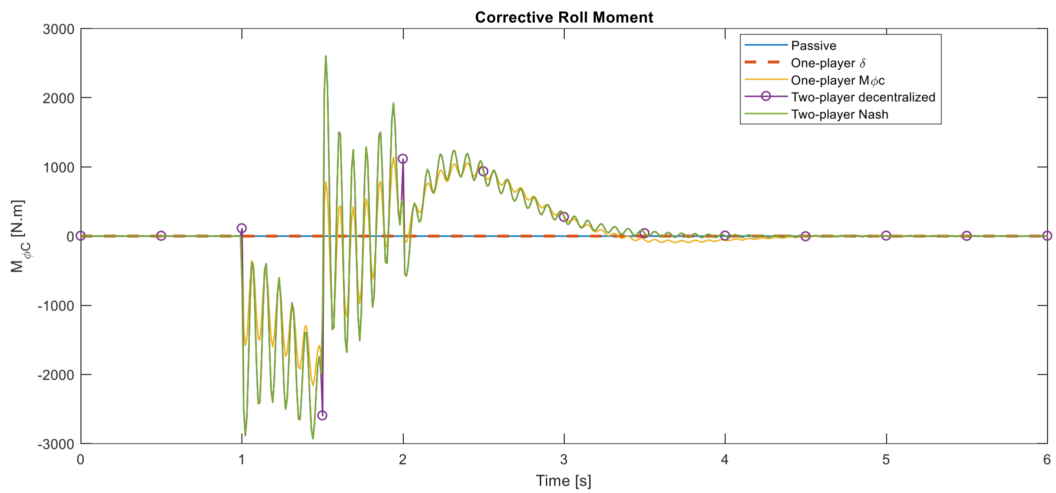

Figure 6-30: Corrective roll moment angle (active steering vs. corrective roll moment competing game)

Similar behavior is shown in all paradigms for all the normal and lateral left and tire forces, according to Figure 6-31.

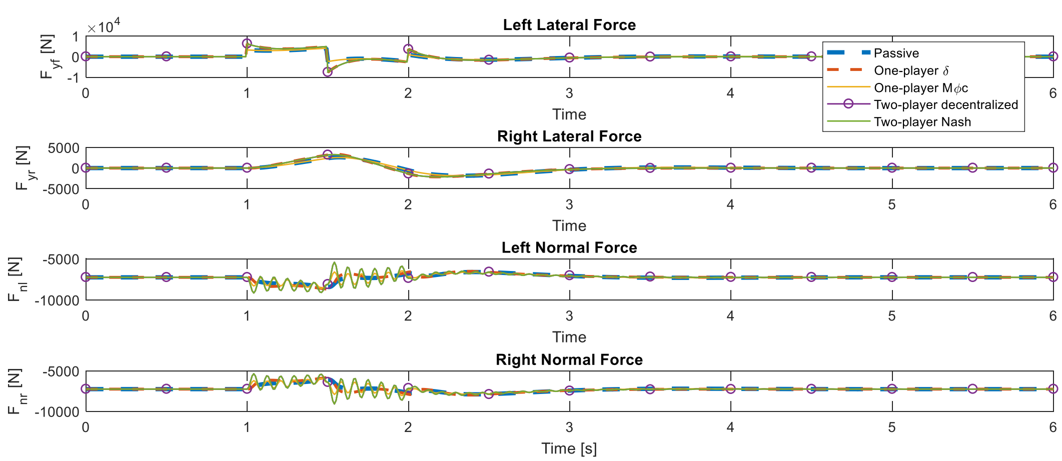

Figure 6-31: Lateral and normal tire forces angle (active steering vs. corrective roll moment competing game)

As shown in Figure 6-32, One-player δ shows higher suspension deflections, which is moderated by adding the corrective roll moment as the second player. The range for tire deflections remains the same, with more noises in one-player $M_{\phi c}$ and the two-player ones.

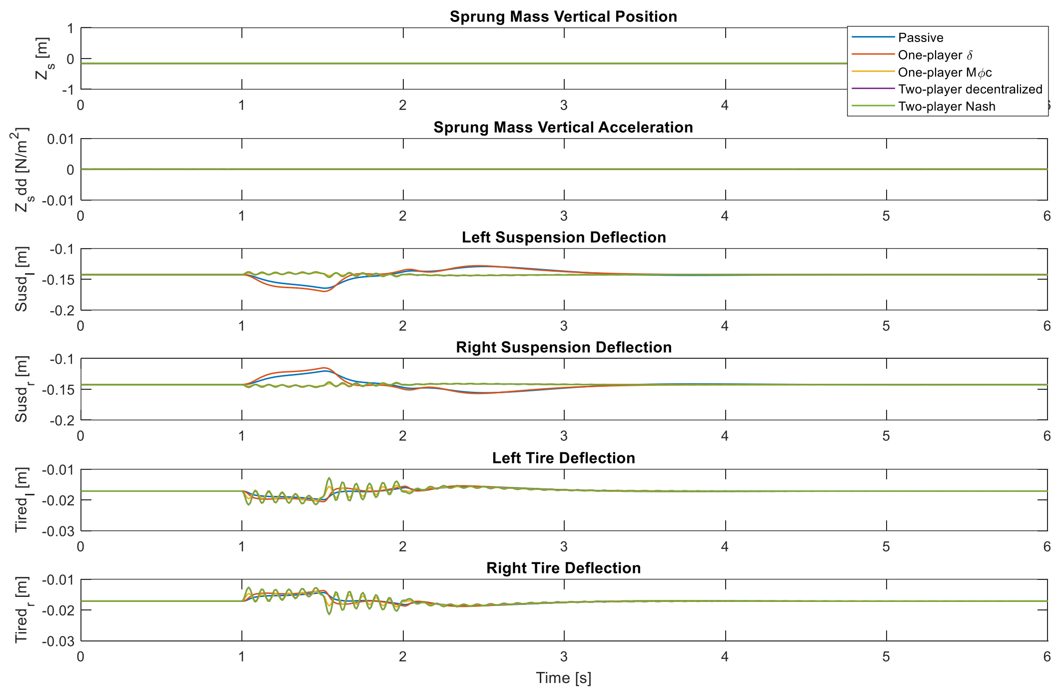

Figure 6-32: Vertical dynamic simulation results angle (active steering vs. corrective roll moment competing game)

One-player $M_{\phi c}$ and two-player paradigms generate identical but higher lateral acceleration than one-player active steering, according to Figure 6-33.

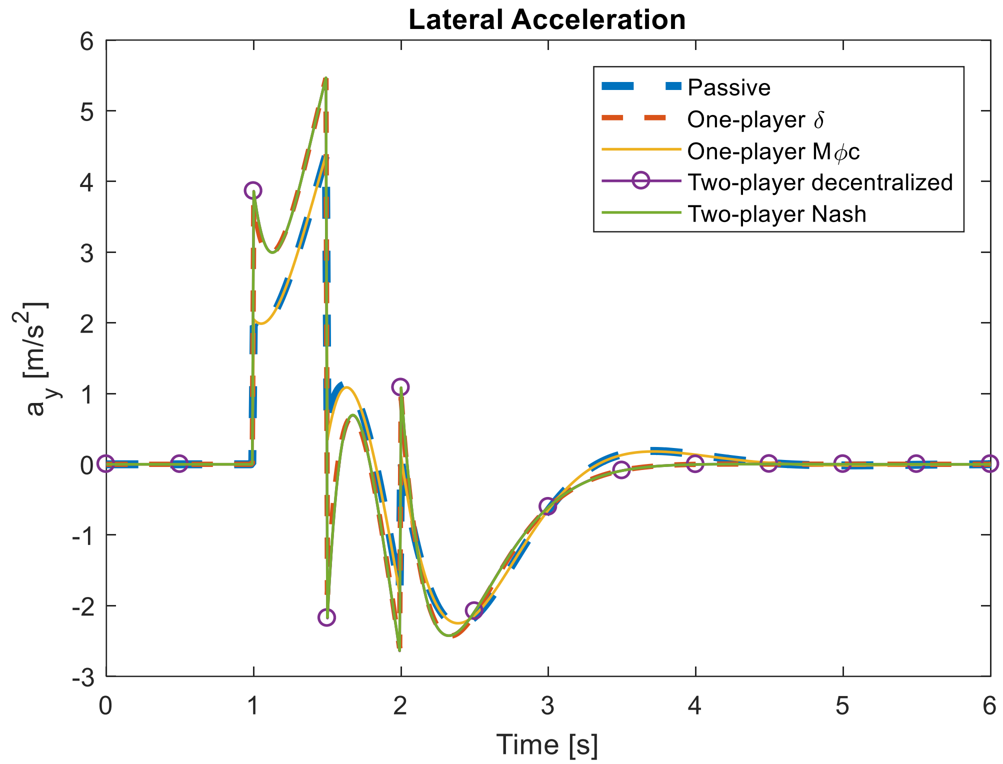

Figure 6-33: Lateral acceleration angle (active steering vs. corrective roll moment competing game)

It appears that the only benefit that is achieved here is that a two-player game has the benefit of achieving two different objectives of the two players. The state and input weights chosen for the Nash paradigm were similar to the other paradigms to compare the outputs of each paradigm. However, Nash paradigm behavior in the competing scenarios is highly dependant on how the state and input weights are chosen in the cost functions. By tuning these weights, the controller designer will be able to penalize a specific objective or control input. This decision should be made based on the priorities that can happen in dangerous maneuvers; for instance, in some maneuvers, the roll stability can be prioritized over the tracking in order to prevent the rollover and guarantee the safety of the passengers. In some other maneuvers losing tracking can cause the vehicle to hit a fatal obstacle or fall off the main road off a cliff, so the tracking must be prioritized. Exploring these scenarios, finding the right decision-making strategy, and catering the cost function weights to prioritize those decisions can be a potential target for future works.

6.4 Three-player game between δ_{AS}, M_{yc} and $M_{\phi c}$

The previous section showed that there still exists some tracking error in the two-player combination of active steering angle and corrective roll moment (Figure 6-27 and Figure 6-28). In order to take advantage of all the three control inputs (active steering, corrective yaw moment, and corrective roll moment), a three-player scenario is simulated in which both active steering and corrective yaw moment are trying to improve tracking by regulating yaw rate error. Meanwhile, the corrective roll moment improves roll performance by regulating the roll angle and roll rate.

At first, the single players' simulations are done, and then the three-player game is simulated for both decentralized and Nash paradigms. The weights and result data for the single-player paradigms are the same as in previous sections. Maintaining the same weights, the results for three-player paradigms are as follows

1) **One-player** δ_{AS}
2) **One-player** M_{yc}
3) **One-player** $M_{\phi c}$

The first three paradigms are the same as the single-player results in the previous sections. The three-player combination is investigated for both decentralized and Nash as follows:

4) Three-player decentralized $\delta_{AS} + M_{yc} + M_{\phi c}$

By combining the three single-player controllers, the matrices and weights will be assigned as follows:

```
C1=[0 0 0 1]; %for delta regulating er
Qb1=1;
Q1=C1'*Qb1*C1;
R1=6.25;
C2=[0 0 0 1]; %for Myc regulating er
Qb2=eye(2);
Q2=C2'*Qb1*C2;
R2=1e-10;
C3=[1 0 0 0;  %for Mφc regulating [phi,phid]
    0 1 0 0];
Qb3=eye(2);
Q3=C3'*Qb3*C3;
R3=1e-14;
```
(6-14)

and the optimal gains for this paradigm are

K_{δ} =[-0.0090   -0.0019   0.0079   0.2358]

K_{Myc} $=10^4 \times$[-0.0487   -0.0053   0.0606   9.4749]

$K_{M\phi c}$ $=10^6 \times$[9.9455   9.9950   -0.0007   0.0002]

The root mean square of the control input signals and total cost for the simulation are:

δ_{RMS} = 0.0147 *rad*

$Myc_{RMS} = 6.1988\times10^{3}\ N.m$

$M\phi c_{RMS} = 846.0558\ \ N.m$

$J_{total}\ =0.2814$

5) **Three-player Nash** $\delta_{AS} + M_{yc} + M_{\phi c}$

The matrices and weights for this paradigm are assigned as:

The output matrices for each player are as follows:

```
C1=[0 0 0 1]; %for delta regulating er
```
(6-15)

C2=[0 0 0 1]; %for M_{yc} regulating er

C3=[1 0 0 0; %for $M_{\phi c}$ regulating [phi,phid]

```
    0 1 0 0];
```

The state weights for each player is calculated as below:

```
Qb1=1;
Qb2=1;
Qb3=eye(2);
Q1=C1'*Qb1*C1;
Q2=C2'*Qb2*C2;
Q3=C3'*Qb3*C3;
```
(6-16)

The input weights for the active steering cost function:

```
R11=6.25;
R12=0;
R13=0;
```
(6-17)

The input weights for the corrective yaw moment cost function:

```
R21=6.25;
```
(6-18)

```
R22=1e-10;
R23=0;
```

The input weights for the corrective roll moment cost function:

```
R31=6.25;
```
(6-19)

```
R32=0;
R33=1e-11;
```

Using the weights and solving the optimization problem, the optimal gains for this paradigm are calculated as:

K_{δ} =[-0.0000 -0.0000 0.0000 0.0225]

K_{Myc} $=10^4 \times$[-0.0159 -0.0158 0.0607 9.4304]

$K_{M\phi c}$ $=10^5 \times$[2.6623 3.1118 -0.0062 0.0003]

and root mean square of the input signals and total cost for the simulation are as follows

$\delta_{RMS} = 0.0015\ rad$

$Myc_{RMS} = 6.2512\times10^3\ N.m$

$M\phi c_{RMS} = 772.5502\ N.m$

J_{total} =0.27.25

Table 6-4 shows the summary of all the paradigms for this simulation

Table 6-4: Three-player game between active steering, corrective yaw, and roll moment scenario RMS values and total cost

	One-player δ_{AS} for track	One-player M_{yc} for track	One-player $M_{\phi c}$ for roll	Three-player decentralized	Three-player Nash
δ_{RMS}	0.0325	0	0	0.0147	0.0015
Myc_{RMS}	0	6.2835×10^{3}	0	6.1988×10^{3}	6.2512×10^{3}
$M\phi c_{RMS}$	0	0	533.7205	846.0558	772.5502
Total cost J_{total}	0.2124	0.2725	3.0447×10^{-8}	0.2814	0.2725

By comparing the root mean square values for steering input, it can be concluded that the three-player Nash paradigm has the lowest control cost and the cost for the decentralized is still less than the one-player. The corrective yaw moment control cost for three-player is less than one player, but Nash scored a slightly higher value than the decentralized. Comparing the corrective roll moment, RMS shows a higher value for three players than one player and lower for Nash compared to the decentralized paradigm.

It is unfair to make a total cost comparison between the three-player and one-player paradigms, as more control objectives need to be achieved for the three-player. However, Nash shows a lower value in the three-player paradigms, which makes it a more optimal answer than the decentralized one.

The following plots show the results of the simulations for different paradigms.

Figure 6-34 shows that Nash and decentralized show almost identical roll stability and tracking performances.

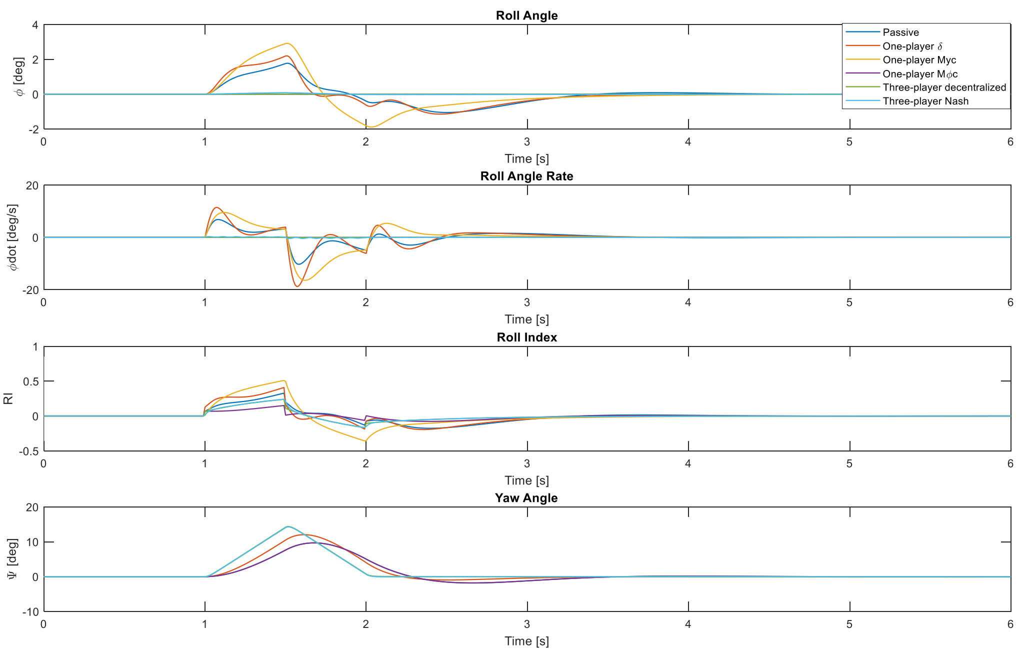

Figure 6-34: Simulation results for roll angle, roll rate, roll index, and yaw angle (three-player)

The tracking performance for the three-player paradigms is very similar to one-player corrective roll moment and better than one player active-steering. By comparing Figure 6-35 for the three-player with Figure 6-27 for the two-player game, it can be concluded that the steady-state lateral error improved significantly, which shows the necessity of adding the corrective yaw moment as the third player to improve tracking.

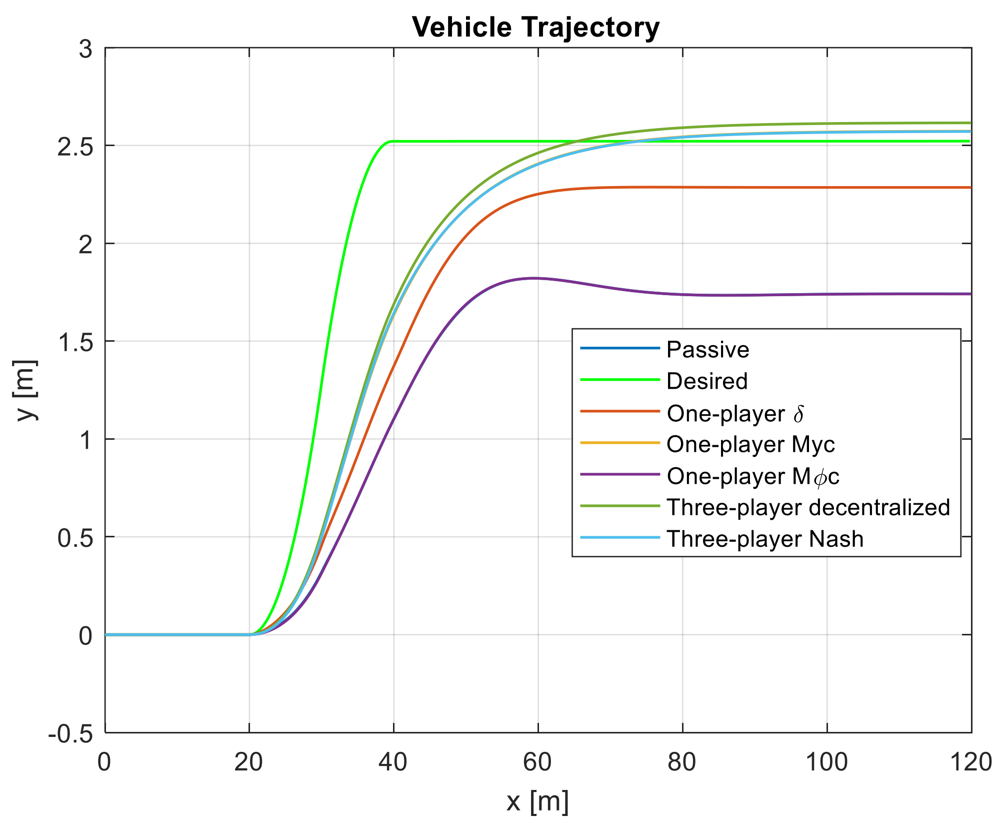

Figure 6-35: Vehicle trajectory (three-player)

Figure 6-36 shows the steering input signals for all the paradigms. One-player δ_{AS} and three-player paradigms generate higher steering input control signals. In order to see the peak of the Three-player Nash paradigm for this control input, a zoomed version of the figure is presented in Figure 6-37.

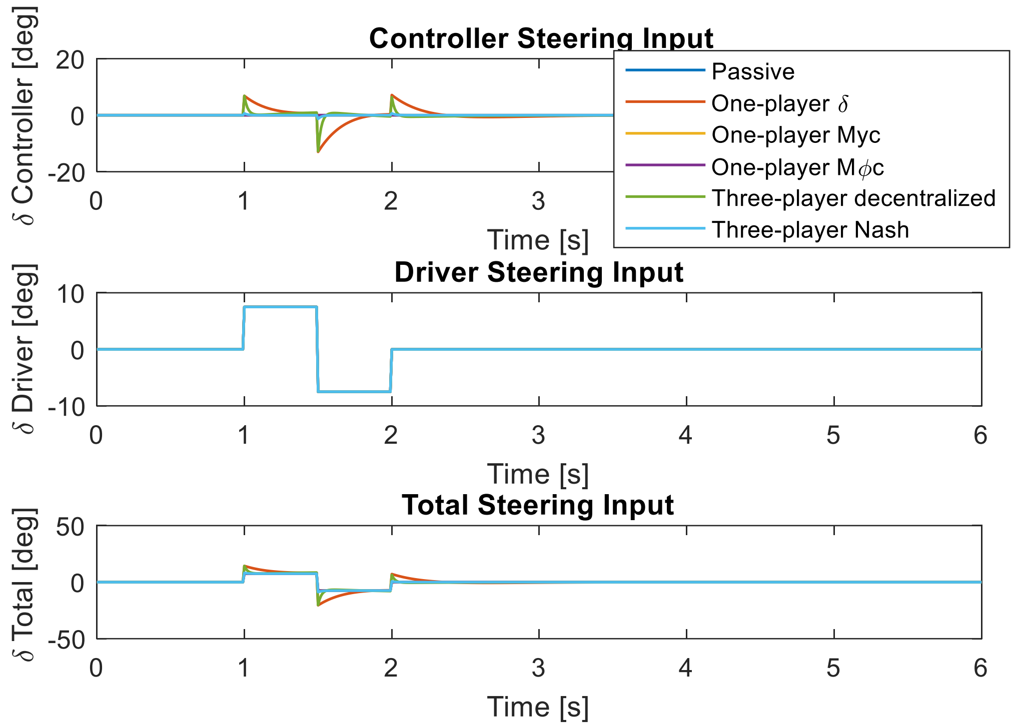

Figure 6-36: Human driver and controller steering inputs (three-player)

As it is shown in Figure 6-37, the three-player Nash peak for steering input is around 1.3 deg, significantly lower than the peak for three-player decentralized or one player δ_{AS} around 14 deg. For the same amount of tracking performance, three-player Nash requires less control effort, and its steering input is less likely to reach saturation limit if there exists a cap for this control input.

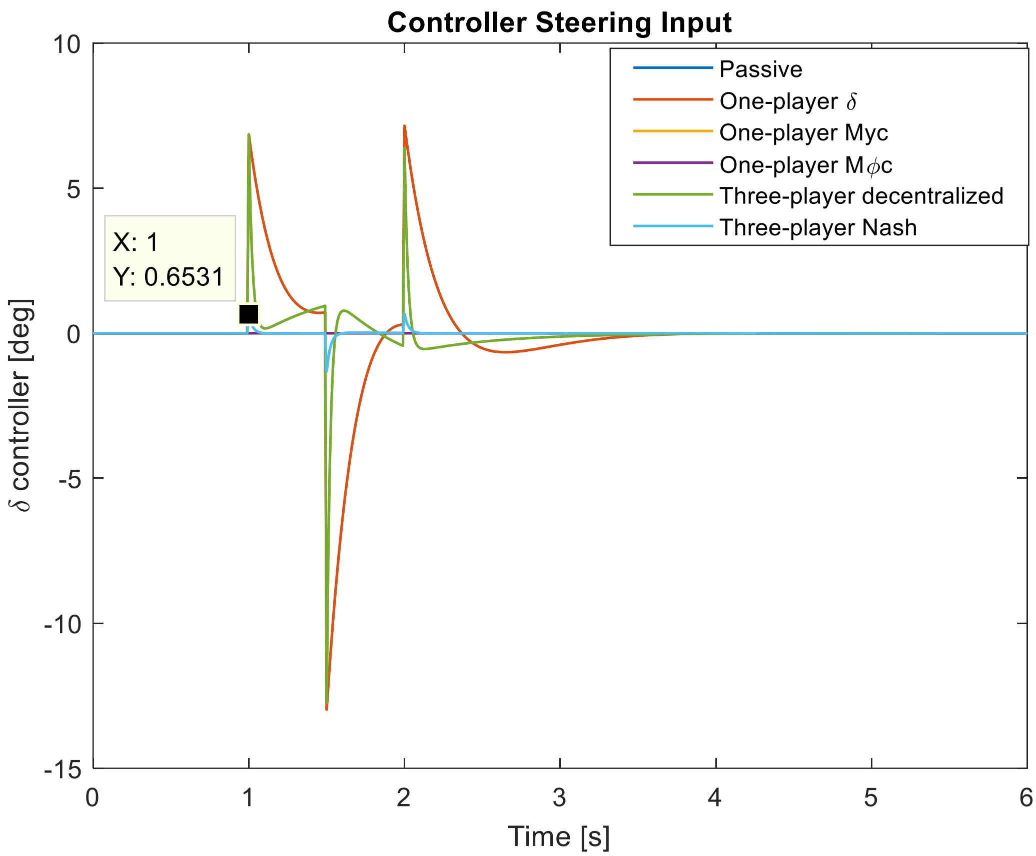

Figure 6-37: Controller steering input zoomed-in (three-player)

Figure 6-38 shows the corrective yaw moment and roll moment for all the paradigms. It is shown that the three players' paradigms have identical behavior but with higher control input amplitude relative to their corresponding single-player controller. Only one player M_{yc} and the three-player paradigms generate the corrective yaw moment. The peaks for corrective yaw moment are almost identical for all the paradigms. One-player $M_{\phi c}$ and three-player scenarios generate a corrective roll moment. The peak increases for three-player scenarios and is slightly less in the Nash paradigm in comparison to decentralized.

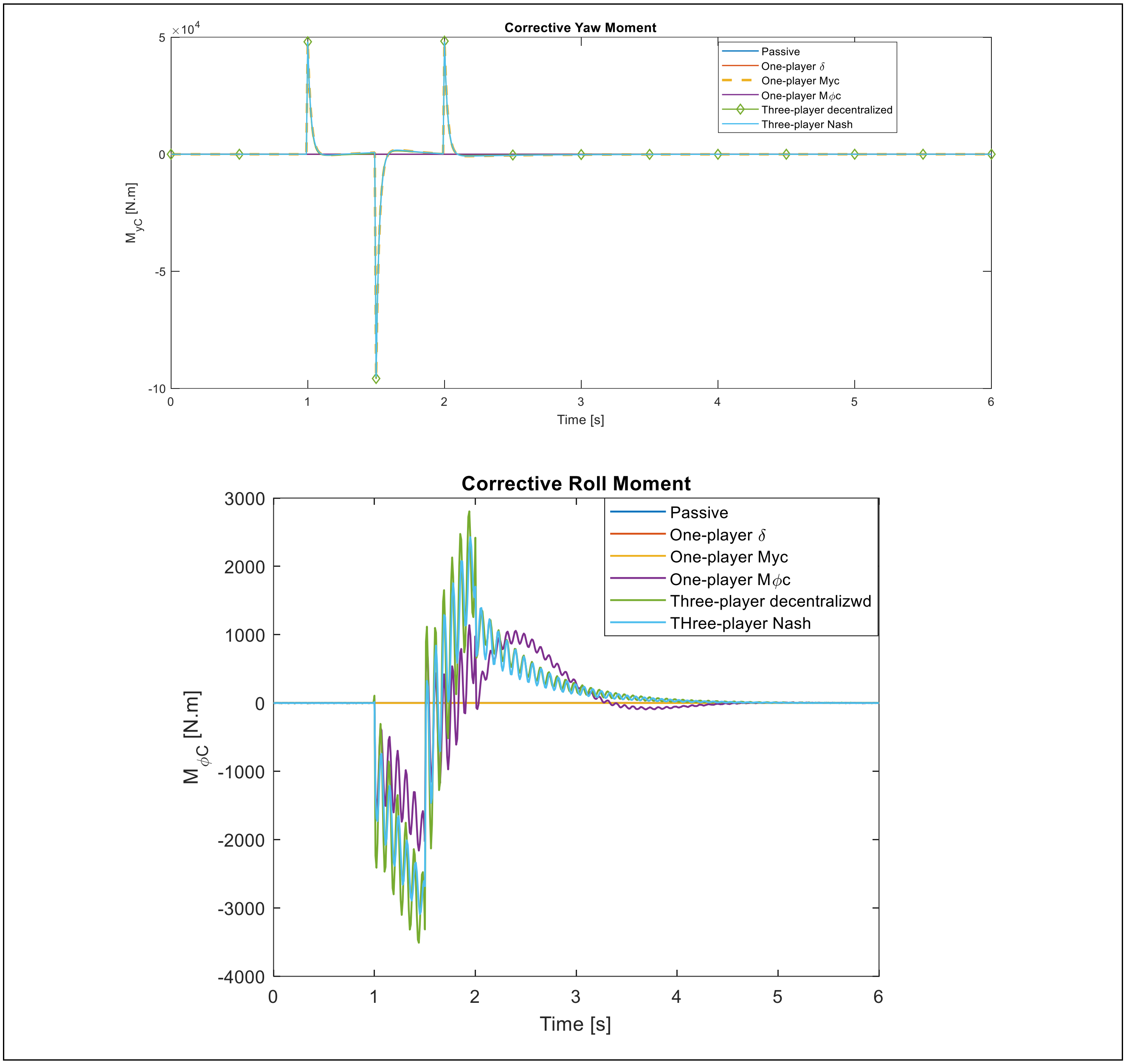

Figure 6-38: Controller inputs for corrective yaw moment and corrective roll moment (three-player)

Figure 6-39 once again confirms the almost identical tracking performances for three-player paradigms.

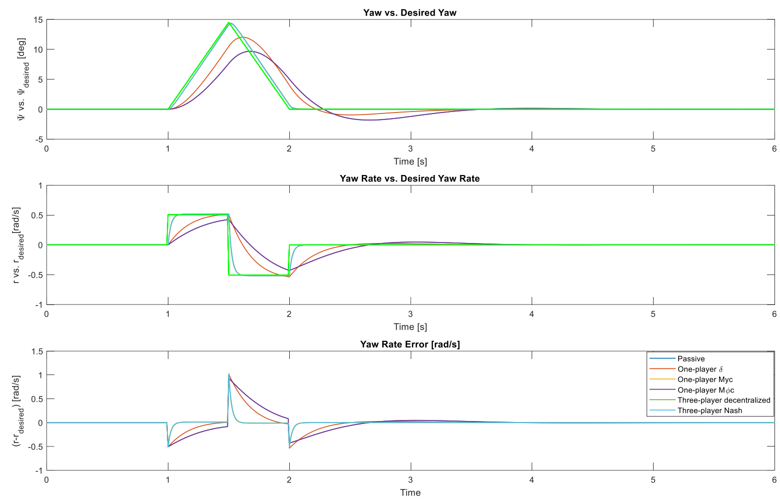

Figure 6-39: Yaw plots (three-player)

Figure 6-40 shows the normal and lateral tire forces for the left and right tires. Both the three-player paradigms generate almost identical forces as the one-player M_{yc}.

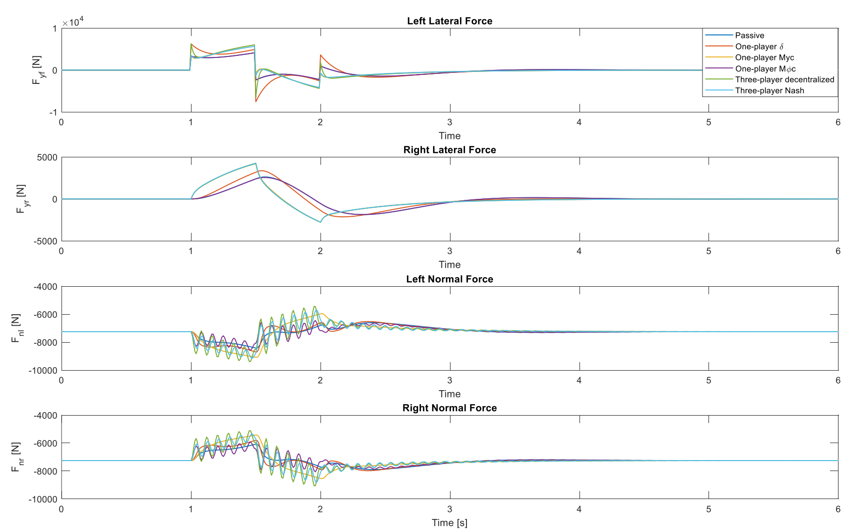

Figure 6-40: Lateral and normal tire forces (three-player)

According to Figure 6-41, It appears that both three-player paradigms show good performance in moderating the variations of suspension deflections.

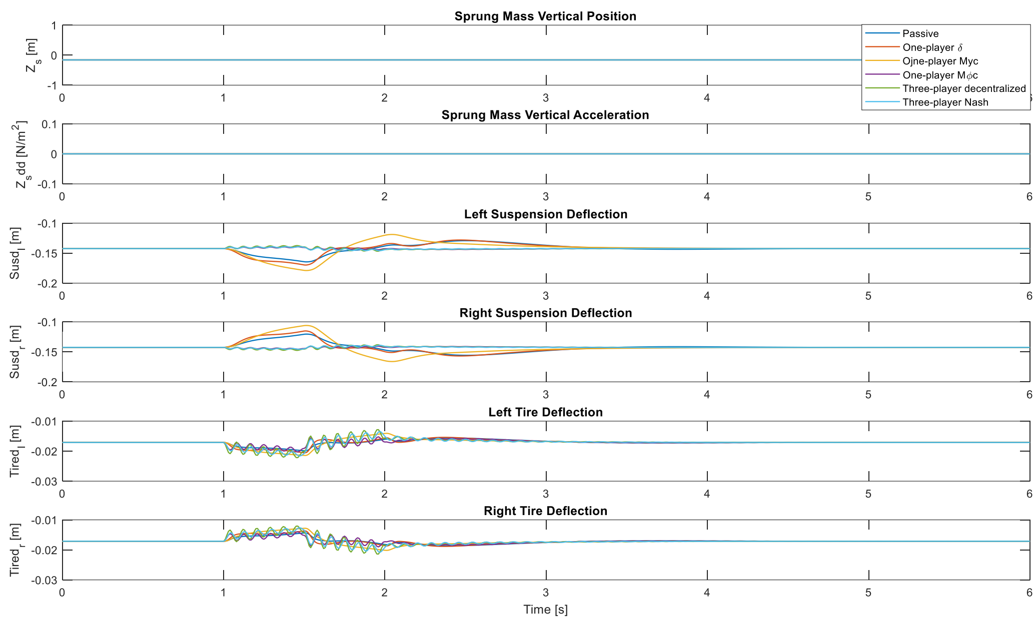

Figure 6-41: Vertical dynamic simulation results (three-player)

As it is shown in Figure 6-42, both three-player paradigms have the lateral acceleration close to the one-player M_{yc}, which is higher than all the other paradigms. This makes sense as a part of the control goal is to make the vehicle follow the path. There is a slight reduction of lateral acceleration in Nash relative to decentralized.

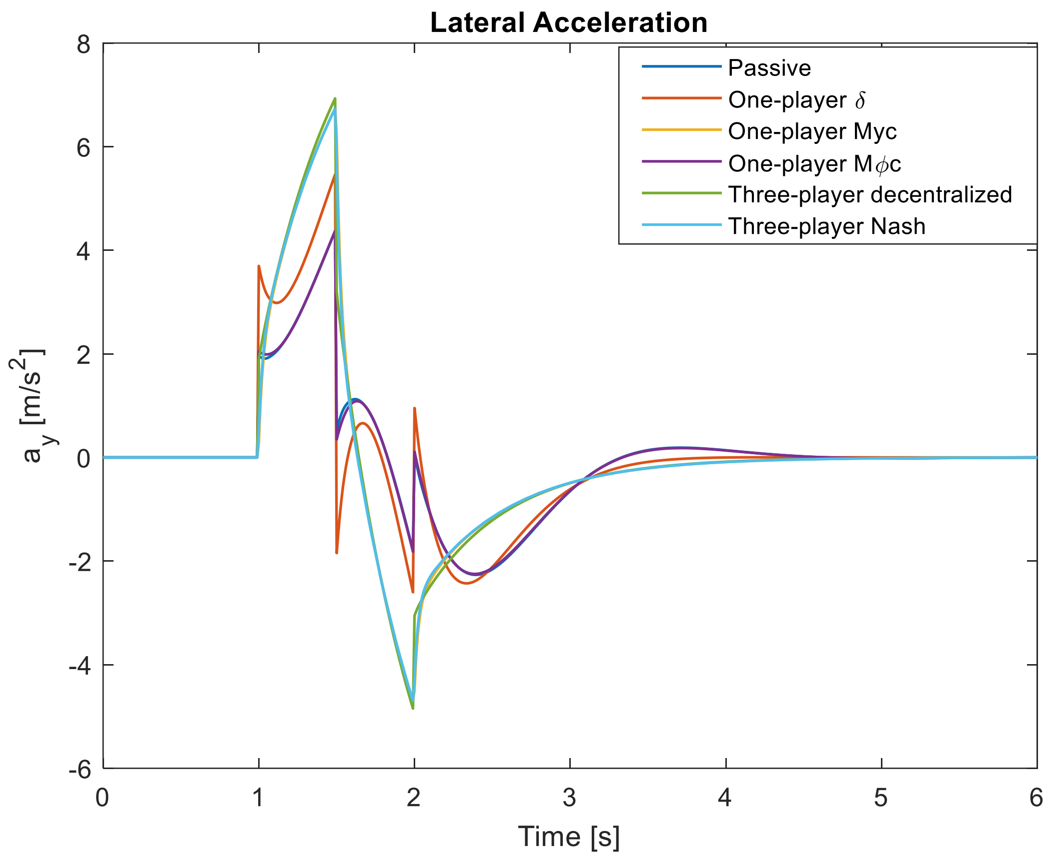

Figure 6-42: Lateral acceleration (three-player)

Adding the corrective yaw moment as a third player to the two-player game between active steering and corrective roll moment improved the tracking by reducing the steady-state tracking error. Like the two-player, the three-player game results are also dependant on the state and control input weights selected in the cost function, and different objectives can be achieved through tuning these weights. In this thesis, only one combination of three-player scenarios is explored, and there are still a wide variety of scenarios that need to be explored and can be potential for future works.

6.5 Two-Player Game between active suspension [F_{al}, F_{ar}] and Corrective roll moment $M_{\phi c}$

This section presents the game between active suspension and corrective roll moment. Active suspension forces F_{al} and F_{ar} are created by two actuators on the left and right sides of the vehicle using the roll plane model presented in chapter 3, section 4. The main objective of this player is to provide ride quality by regulating the vertical acceleration. Corrective roll moment is introduced as a second player to improve roll performance by regulating roll and roll rate. After solving the single-player, the combination of these two players is investigated in a two-player game for decentralized and Nash paradigms. Control models presented in chapter 3.4 are used to calculate the feedback gains, and the simulation results are shown for the plant model of chapter 3.6.

The same lane change maneuver as previous sections is picked for this scenario. The road roughness/profile for left and right wheels is also introduced as two disturbance inputs. This disturbance induces fluctuations in vertical acceleration, and the ride quality performance of the controller is investigated by its ability to regulate these fluctuations. Note that the frequency of this disturbance is way lower than 1000 Hz, so the sampling time of 0.001 sec is still good for the simulation

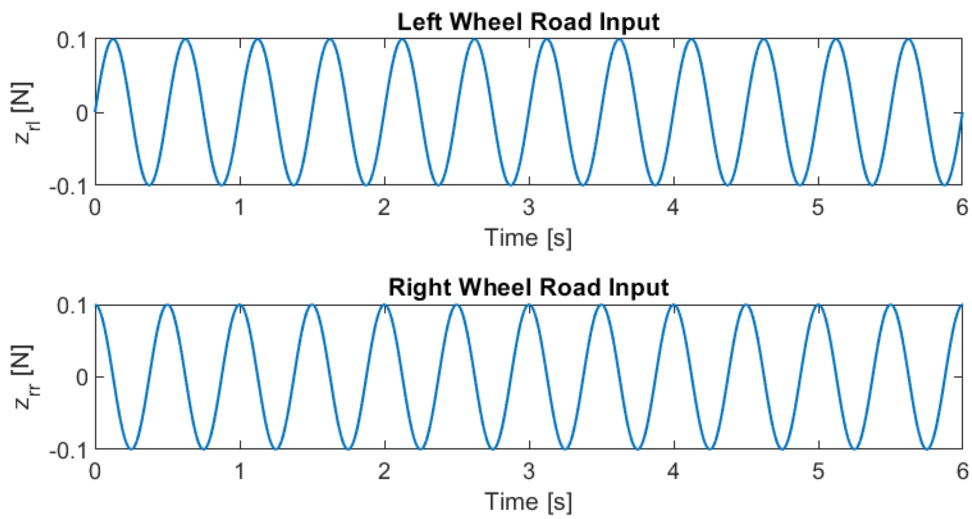

Figure 6-43: Road roughness profile for the left and right wheel in roll plane model

Note that in previous sections, the outputs were only related to the states of the system. All the paradigms in previous sections belong to the class of linear quadratic regulation (*LQR*) problems. In this section, the vertical acceleration as an output is a function of both states and control inputs. This means that the control coupled output regulation (*CCOR*) method is required to solve this class of problems.

Considering all the assumptions above, here are the four paradigms presented for the game in vertical dynamics.

- **One-player active suspension [F_{al}, F_{ar}]**
- **One-player** $M_{\phi c}$
- **Two-player [F_{al}, F_{ar}] +** $M_{\phi c}$ **Decentralized**
- **Two-player [F_{al}, F_{ar}]+** $M_{\phi c}$ **Nash**

1. One-player active suspension [F_{al}, F_{ar}]

Active suspension forces F_{al} and F_{ar} have applied to the left and right suspension systems accordingly. The objective of this controller is to improve ride quality by regulating the sprung mass vertical acceleration. The results for this simulation are plotted in red. The designed weight matrices are as follows.

```
C4=Av(5,:);                                                    (6-20)
Qb4=1e2;
D4=B4v(5,:);
Rb4=eye(2);
rob4=1e-6*eye(2);
Q1=C4'*Qb4*C4;
N1=C4'*Qb4*D4;
R1=D4'*Qb4*D4+rob4*Rb4;
```

The control gain for this paradigm is calculated as below for the selected weights:

$$K_F = 10^4 \times \begin{bmatrix} -4.1495 & 0.0000 & 2.2352 & 2.2352 & -0.1865 & -0.0000 & 0.1952 & 0.1952 \\ -4.1495 & -0.0000 & 2.2352 & 2.2352 & -0.1865 & 0.0000 & 0.1952 & 0.1952 \end{bmatrix}$$

and the RMS of the actuating forces signal and the total cost are as follows

$$F_{RMS} = 10^4 \times [5.9603 \quad 5.9603]\ N$$

$$J_{total} = 1.0164 \times 10^5$$

2. One-player $M_{\phi c}$

Like previous sections, corrective roll moment is improving roll stability by regulating roll angle and roll rate. The results for this simulation are plotted in yellow, and the weight matrices are.

```
C1=[0 1 0 0 0 0 0 0; %phi                                  (6-21)
     0 0 0 0 0 1 0 0];
%phid
Qb1=eye(2);
D1=[0;0];
Rb1=1;
rob1=1e-14;
```

Here are the optimal gain, root mean square of the moment signal, and total cost:

$$K_{M\phi c} = 10^6 \times \begin{bmatrix} 0.0000 & 9.9376 & -0.0304 & 0.0304 & 0.0000 & 9.9950 & 0.0033 & 0.0033 \end{bmatrix}$$

$$M\phi c_{RMS} = 6.4900 \times 10^3 \; N.m$$

$$J_{total} = 2.5697 \times 10^{-6}$$

3. Two-player [F_{al}, F_{ar}] + $M_{\phi c}$ Decentralized

This paradigm is the combination of previous paradigms. The results for this simulation are plotted in purple. The weighting matrices are selected the same as one-player scenarios, which create the same optimal gains.

$$K_F = 10^4 \times \begin{bmatrix} -4.1495 & 0.0000 & 2.2352 & 2.2352 & -0.1865 & -0.0000 & 0.1952 & 0.1952 \\ -4.1495 & -0.0000 & 2.2352 & 2.2352 & -0.1865 & 0.0000 & 0.1952 & 0.1952 \end{bmatrix}$$

$$K_{M\phi c} = 10^6 \times \begin{bmatrix} 0.0000 & 9.9376 & -0.0304 & 0.0304 & 0.0000 & 9.9950 & 0.0033 & 0.0033 \end{bmatrix}$$

The RMS of control inputs and the total cost is as follows

$$F_{RMS} = 10^4 \times [6.0015 \quad 6.0015] \ N$$

$$M\phi c_{RMS} = 1.1127 \times 10^4 \ N.m$$

$$J_{total} = 1.0226 \times 10^5$$

4. Two-player [F_{al}, F_{ar}] + $M_{\phi c}$ Nash

The weight matrices are designed as follows. The results for this simulation are plotted in green.

Output matrices for corrective roll moment:

```
CC1=[0 1 0 0 0 0 0 0; %phi
     0 0 0 0 0 1 0 0]; %phid
```
(6-22)

```
DD11=[0;0];
DD12=[0 0;
      0 0];
```

Input and state weights for corrective roll moment:

```
Rbb11=1;
Rbb12=eye(2);
rob11=1e-14;
rob12=1e-5;
```
(6-23)

```
Qbb1=eye(2);
```

Output matrices for active suspension:

```
CC2=Av(5,:);                                                    (6-24)
DD21=B3v(5,:);
DD22=B4v(5,:);
```

(Av is the state matrix and B3v and B4v are the input matrices for corrective roll moment and active suspension forces for control model in section 3.4)

Input and state weights for active suspension:

```
Qbb2=1e2;                                                       (6-25)
Rbb21=1;
Rbb22=eye(2);
rob21=0;
rob22=1e-6;
```

the optimal games for this Nash paradigms are

$$K_F = 10^4 \times \begin{bmatrix} -3.2150 & -0.0000 & 1.8806 & 1.8806 & -0.0416 & -0.0000 & 0.1521 & 0.1521 \\ -3.2150 & 0.0000 & 1.8806 & 1.8806 & -0.0416 & 0.0000 & 0.1521 & 0.1521 \end{bmatrix}$$

$$K_{M\phi c} = 10^6 \times \begin{bmatrix} 0.0000 & 9.9376 & -0.0304 & 0.0304 & 0.0000 & 9.9950 & 0.0033 & 0.0033 \end{bmatrix}$$

and the RMS of control inputs and the total cost as follows:

$$F_{RMS} = 10^4 \times [1.5154 \quad 1.5154] \; N$$

$$M\phi c_{RMS} = 6.6225 \times 10^3 \; N.m$$

$$J_{total} = 9.0037 \times 10^4$$

Table 6-5: Active suspension vs. corrective roll moment scenario RMS values and total cost

	One-player [F_{al}, F_{ar}]	One-player $M_{\phi c}$ for roll	Two-player decentralized	Two-player Nash
F_{RMS}	10^4×[5.9603 5.9603]	0	10^4×[6.0015 6.0015]	10^4×[1.5154 1.5154]
$M\phi c_{RMS}$	0	6.4900×10^3	1.1127×10^4	6.6225×10^3
Total cost J_{total}	1.0164×10^5	2.5697×10^{-6}	1.0226×10^5	9.0037×10^4

Both players' RMS of control inputs are higher than the corresponding RMS for single-player paradigms in the two-player decentralized. Nash two-player scored the lowest RMS for control inputs overall. It also has a lower control cost than the decentralized and single-player active suspension. Therefore the Nash paradigm is the optimal solution overall.

All three scenarios of one-player $M_{\phi c}$, two-player decentralized, and two-player Nash show identical performance when it comes to roll and roll rate regulation, and they also keep the roll index in the safe zone (Figure 6-44). The only basis for comparison will be the amount of control effort that each paradigm consumes to get the same result. From the yaw angle plot, it is clear that there is no change in lateral dynamics.

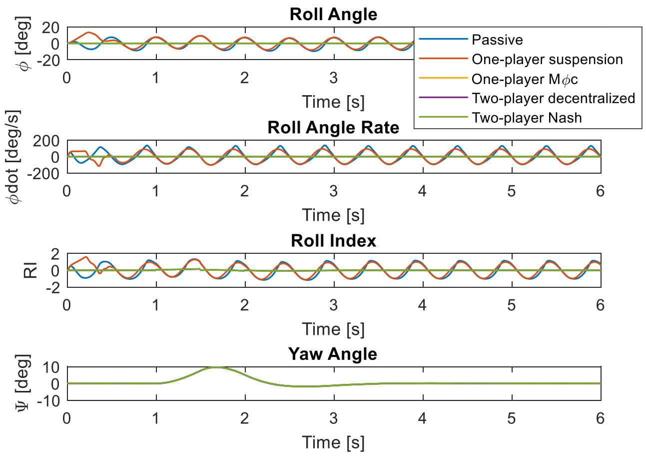

Figure 6-44: Simulation results for roll angle, roll rate, roll index, and yaw angle (active suspension vs. corrective roll moment game)

By comparing the peaks of the control input signals for corrective roll moment and active suspension forces, it is clear that the Nash paradigm has lower peaks for all the control inputs (Figure 6-45). Considering the fact that Nash scored lower RMS in all control inputs as well, one can conclude that the Nash paradigm is the optimal solution for this scenario.

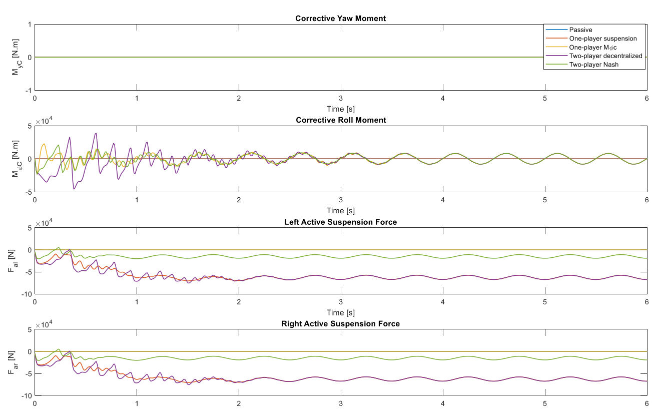

Figure 6-45: Controller inputs for corrective yaw moment, corrective roll moment, and active suspension forces (active suspension vs. corrective roll moment game)

As seen in Figure 6-46, Nash is doing a better job in preventing the normal tire forces from becoming zero, which is good for vertical stability and in some cases tracking due to maintaining the tire contact with the road.

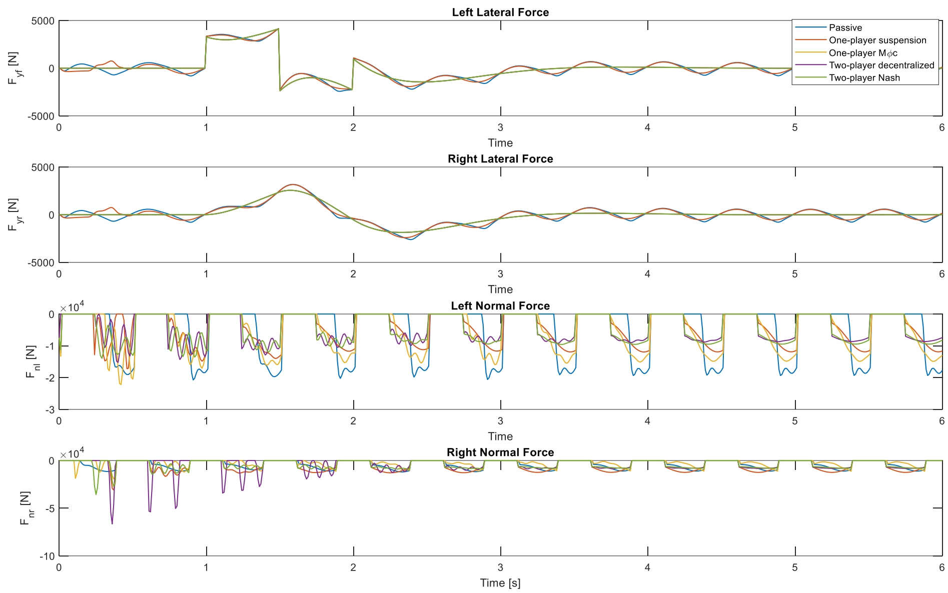

Figure 6-46: Lateral and normal tire forces (active suspension vs. corrective roll moment game)

Both two-player paradigms keep the sprung mass vertical acceleration low with slightly better performance for decentralized, which shows improvement in ride quality. Nash is doing a better job in reducing the suspension deflection and tire deflection. See Figure 6-47.

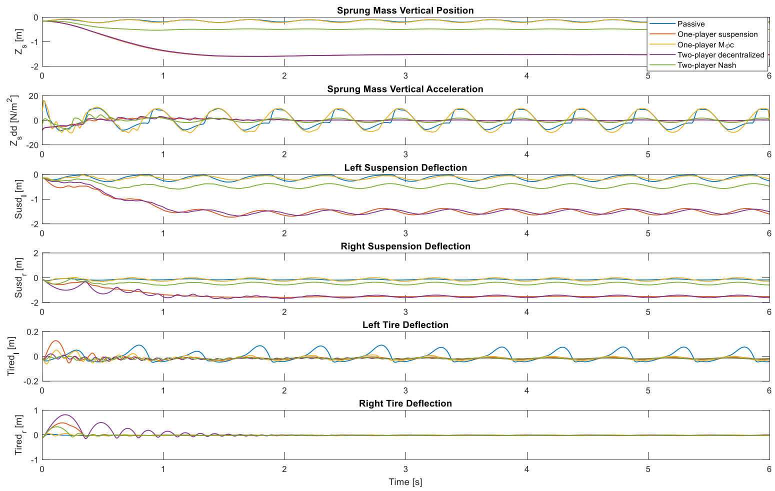

Figure 6-47: Vertical dynamic simulation results (active suspension vs. corrective roll moment game)

As it is shown in Figure 6-48, One-player $M_{\phi c}$ and the other two-player paradigms have identical lateral acceleration, which is a smoother response than passive and one-player active suspension.

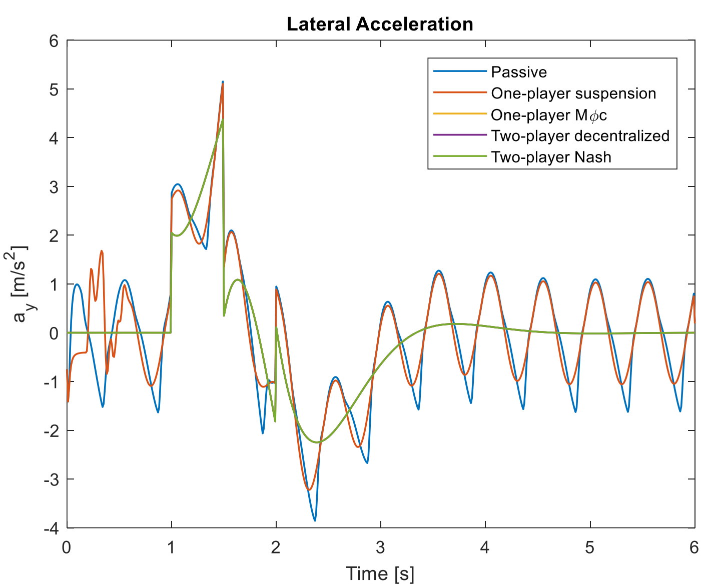

Figure 6-48: Lateral acceleration (active suspension vs. corrective roll moment game)

The game between active suspension and corrective roll moment was chosen as a case study for the CCOR problem in this section. The main objective of active suspension is to improve ride quality by regulating the sprung mass vertical acceleration, and the corrective roll moment is to pursue ride quality by regulating roll and roll rate. It was shown that the Nash paradigm shows better performance in reaching the objectives by spending less control effort, which makes it the optimal paradigm for this scenario.

6.6 Chapter summary

In this chapter, the performance of the decentralized and Nash paradigm is studied in four different game scenarios in vehicle dynamics. In each scenario, the simulation was presented for both single-player and multiple-player (two or three players). The players were chosen from the following player pool:

- Active steering (δ_{AS})
- Corrective yaw moment (M_{yc})
- Corrective roll moment ($M_{\phi c}$)
- Active suspension forces (F_{al} and F_{ar})

A simple lane-change maneuver is chosen as a benchmark for simulation and investigating the performance of different paradigms. The maneuver is induced by human driver steering input, and the objectives of tracking, roll stability, and ride quality was observed in simulations. The simulation results are shown for different paradigms in each section, and the necessary comparison is made. Here is the summary of each scenario:

In section 6.2, the game between active steering and corrective roll moment has been investigated. Both players' objective is to improve tracking by regulating the yaw rate error. At first, the scenario is solved for single-player paradigms, and then the two-player game is solved and simulated for both decentralized and Nash paradigms. Both two players' scenarios showed improvements in tracking, but Nash requires lower active steering control effort (lower δ_{RMS}) overall. The peak for the active steering input signal is considerably lower for the Nash paradigm. Suppose the required active steering input for one player game reaches its threshold (there is always a limit on the amount of steering input that the actuator can produce), then the corrective roll moment can be introduced to the game as a second player to reduce the burden of control from active steering and keep the steering input within the feasible range. The *LQR* method developed in chapter 4.3 was used for both one-player and two-player games.

In section 6.3, the game between active steering and corrective roll moment has been investigated. The corrective roll moment ($M_{\phi c}$) improves roll stability by regulating roll angle and roll rate. It was shown that active steering could either play a cooperative (non-competing) game with roll moment to improve the roll stability by counter steering. Or it can also play a competitive game by regulating yaw rate error to improve the tracking performance, which creates a higher roll angle and more control burden for the roll moment player. In the cooperative game, the roll regulation control burden is shared between the players at the cost of losing the tracking, and it is shown that the Nash paradigm has a better performance than decentralized. For the competing scenario, both decentralized and Nash show identical performances, and the only benefit that one gets is the combination of players' objectives in the two-player game rather than one-player. The *LQR* method developed in chapter 4.3 was used for both one-player and two-player games.

In section 6.4, the three-player game has been investigated between active steering, corrective yaw moment, and corrective roll moment. Both active steering and corrective roll moment control objective is yaw rate error regulation to improve tracking, and corrective roll moment is improving roll stability by regulating roll and roll rate. All the single-player paradigms were solved using the classic *LQR* method mentioned in chapter 4. The three-player Nash paradigm is solved using the *LQR* method developed in chapter 5.4. the Nash paradigm reached the control objective by using lower RMS for control inputs, resulting in lower control cost, making it the optimal paradigm.

In section 6.5, the two-player game between the active suspension forces and corrective roll moment was investigated. The objective for the corrective roll moment is chosen similar to previous sections, and the active suspension forces are improving ride quality by regulating sprung mass vertical acceleration. Road roughness is added as disturbance input to both left and right wheels. Note that the output vertical acceleration is a function of both states, and the control inputs make the paradigm a control coupled output regulation (CCOR) problem. The single-player and two-player CCOR problem was solved using the method presented in chapter 4, sections 4 and 5 accordingly. The Nash paradigm shows better performance overall by scoring lower RMS and peak values for control inputs and lower total cost. This makes the Nash paradigm an optimal paradigm overall for this scenario.

7 CHAPTER 7: SIMULATION RESULTS FOR PLANT MODEL WITH NONLINEAR TIRE MODEL

7.1 Introduction

In chapter 6, the control problem was solved for multiple scenarios based on linear control models, and the optimal control gains were calculated. Then the control gains were used in a feedback loop with the plant model introduced in chapter 3.4 to get the simulation results. This plant model was a combination of a 4 DOF roll plane model and the 2 DOF bicycle model. The only nonlinearity in that model was canceling the normal tire force when the tire was not under compression. The linear tire model was used for this model, where the lateral tire force is a linear function of slip angle multiplied by cornering stiffness. This section introduces a nonlinear tire model, and the simulation results are presented for the two scenarios below as case studies.

- The two-player game between active steering and corrective yaw moment
- The three-player game between active steering, corrective yaw moment, and corrective roll moment

Two nonlinear tire models have been generated for this report. The first model is the linear tire model with saturation, and the second one is the Magic tire formula introduced by Pajecka [26].

7.2 Linear model with saturation

In this model, the lateral force is a function of slip angle only, and it will be saturated at both ends when it reaches its thresholds. Front and rear slip angles in the bicycle model are calculated as follows

$$\alpha_f = \left(\delta_{HumanDriver} + \delta_{ActiveSeering}\right) - \frac{V_y + l_f r}{V_x} \qquad (7\text{-}1)$$
$$\alpha_r = -\frac{V_y - l_r r}{V_x}$$

and the lateral forces for front and rear (F_{yf}, F_{yr}) are calculated accordingly using the function below

$$Fy = f(\alpha) \qquad (7\text{-}2)$$
$$Fy = \begin{cases} C_\alpha \alpha & if\ |\alpha| \le \alpha_{threshhold} \\ sign(\alpha) C_\alpha \alpha_{threshhold} & else \end{cases}$$

The lateral forces are used to drive the yaw dynamic part of the nonlinear plant model

$$MV_y - M_s h_s \ddot{\phi} = -MV_x r + F_{yf} + F_{yr} \qquad (7\text{-}3)$$
$$I_z \dot{r} = l_f F_{yf} - l_r F_{yr} + M_{yc}$$

Here are the simulation results for $\alpha_{threshhold} = 0.15 rad$

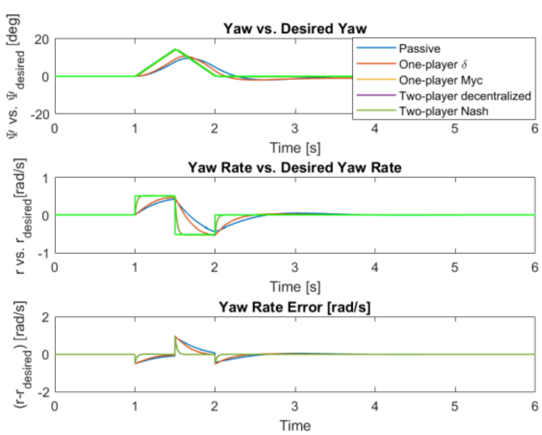

Figure 7-1: Yaw plots for tire model with saturation

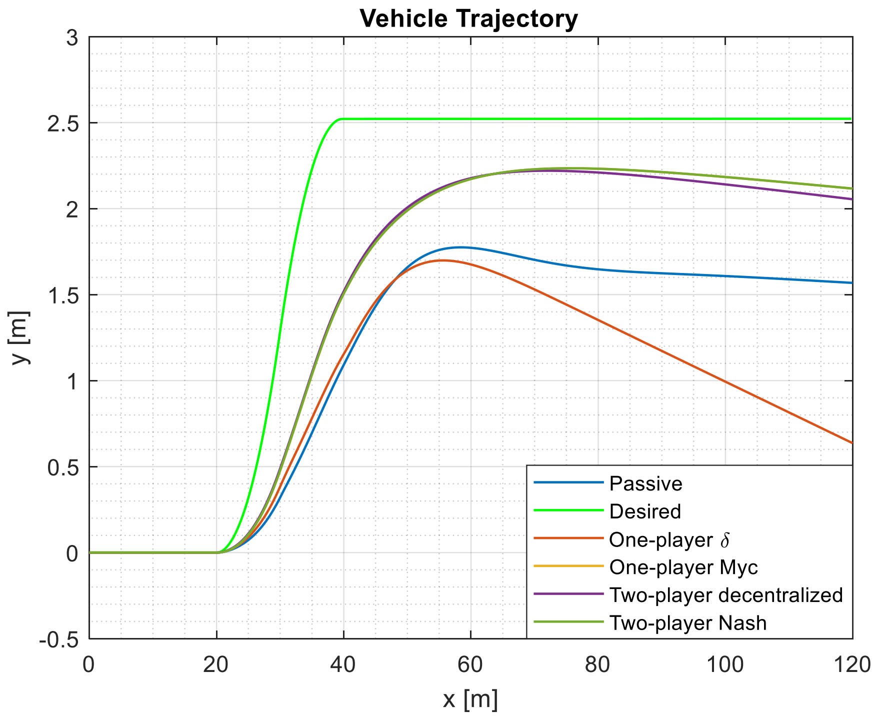

Figure 7-2: Vehicle trajectory for tire model with saturation

It appears that the saturation makes tracking hard for passive maneuver, and this problem gets intensified even in one-player δ and shows that this player fails in the objective of tracking. One-player M_{yc} and both the decentralized and Nash two-player paradigm are capable of improving the tracking.

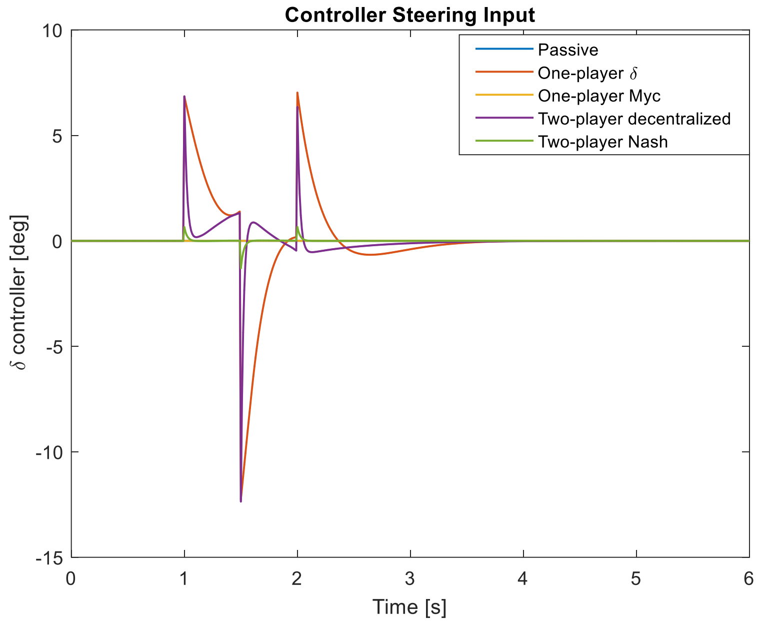

Figure 7-3: Active steering signal for tire model with saturation

The Nash paradigm has the lowest peak for steering input signal overall, which means that this paradigm is reaching the control objective by spending less control effort. (besides the one-player M_{yc} that doesn't involve in active steering)

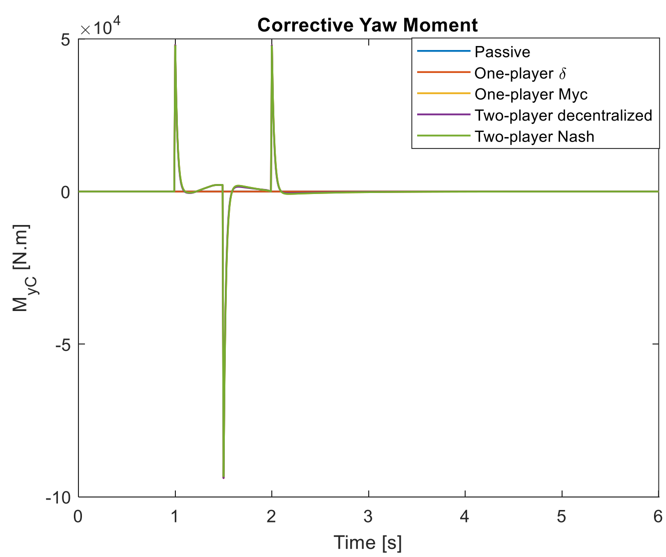

Figure 7-4: Corrective roll moment for tire model with saturation

The first peak of the corrective yaw moment is zoomed in and shown in Figure 7-5. The peak for the Nash Paradigm is slightly less than the One-player δ and two-player decentralized, so the improvement for this control input is not as significant as the one for active steering signal.

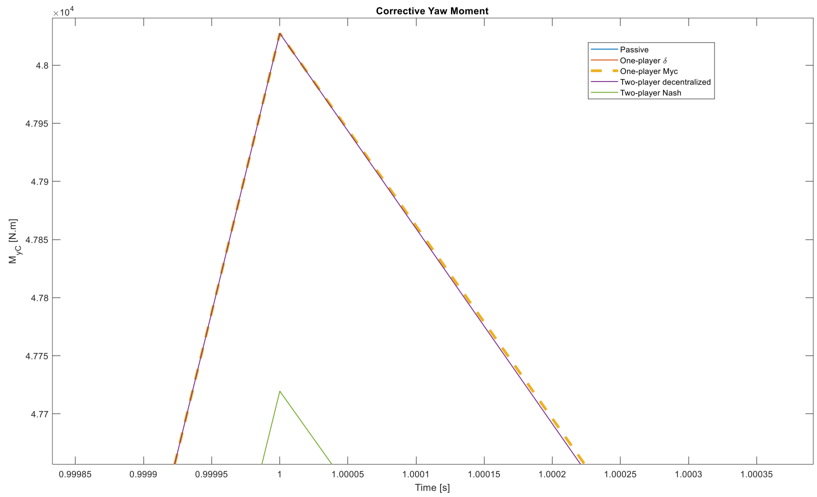

Figure 7-5: Figure 7-4 zoomed in

All the paradigms have almost identical control input performances in terms of corrective yaw moment effort, with Nash having a slightly lower peak. (One-player δ doesn't take part in this)

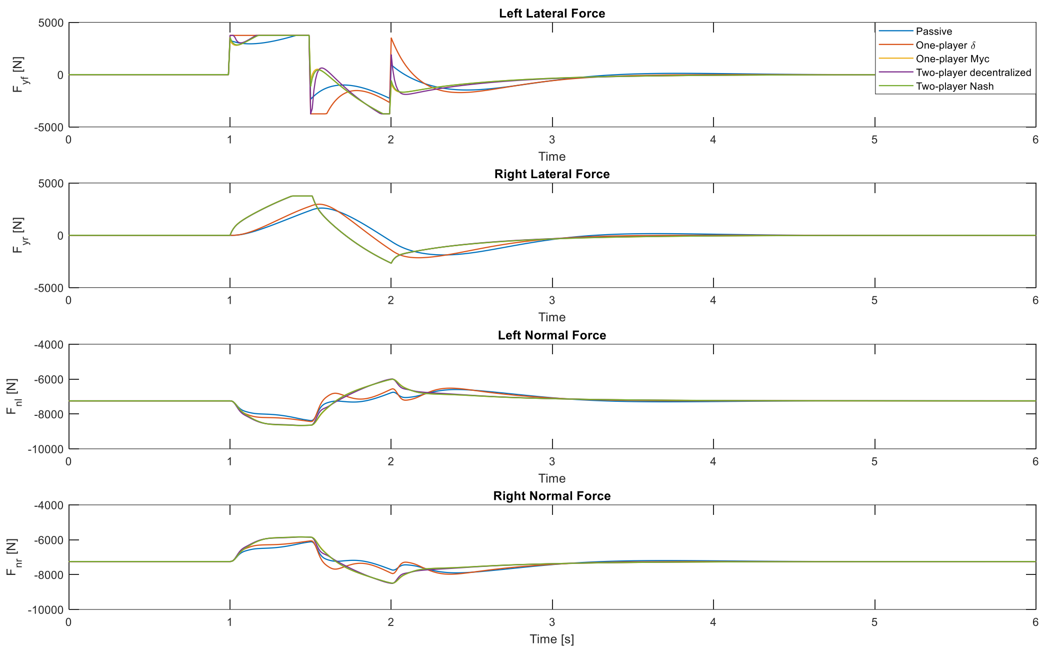

Figure 7-6: Lateral and normal tire forces for tire model with saturation

Considering that the cornering stiffness and slip threshold for both front and rear is chosen as:

$$C_{\alpha} = 25000[N / rad] \qquad (7\text{-}4)$$
$$\alpha_{threshhold} = 0.15[rad]$$

it can be calculated that the lateral forces get saturated at ± 3750 N.

The slip angles and lateral acceleration for each paradigm are as follows

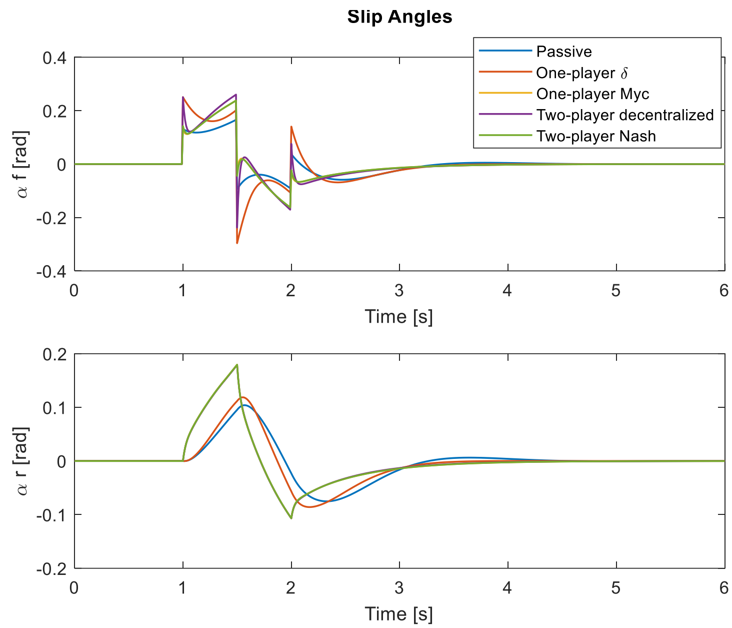

Figure 7-7 Front and rear slip angles for tire model with saturation

It appears that active steering tried to improve tracking by increasing the slip angles, but it was no help.

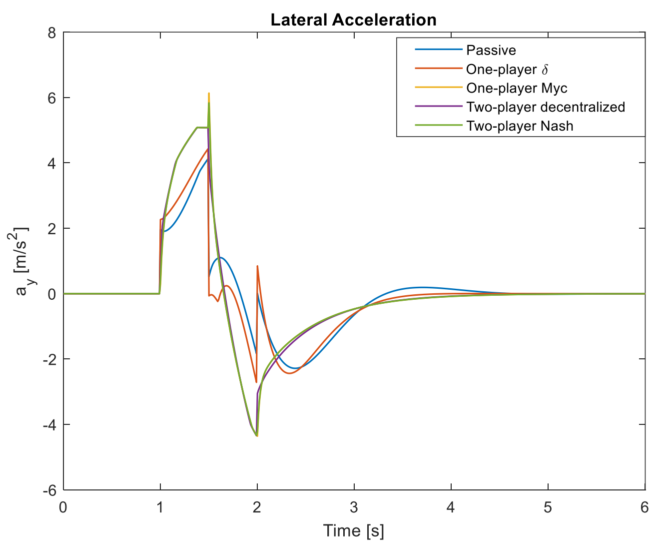

Figure 7-8: Lateral acceleration for tire model with saturation

The active steering didn't change the lateral acceleration noticeably, but all the other paradigms increased the lateral acceleration to improve tracking.

In summary, the active steering made the tracking worse in the presence of lateral force saturation in tires. There was a need for another player to help with this task. From the simulation, it can be concluded that the Nash paradigm shows better performance overall because it managed to achieve the tracking objective using the least control effort. One-player corrective yaw moment is also a good one-player candidate that only spends energy on one control objective with a close peak in the control signal to Nash and achieves the same tracking performance.

7.3 Nonlinear tire model based on Pacejka's Magic Formula

The Magic Formula tire model was developed by H. Pacejka of Delft University of Technology [26] and is commonly used in vehicle dynamics as a good choice for the nonlinear tire model. This model uses the normal tire force, slip angle, longitudinal slip, and camber angle as inputs to calculate lateral force, aligning torque, and longitudinal force generated by the tire. In the model presented in this dissertation, the velocity is constant, and the longitudinal dynamics are not studied. The camber angle is also neglected and considered as zero. A shorter version of the Pacejka model is introduced to calculate the tire lateral force based on slip angle and normal force.

$$Fy = f(\alpha, F_z) \tag{7-5}$$

This model is basically a mathematical model of an S-like shape function. The initial slope of the curve at origin for each tire is the cornering stiffness C_α described in units of force per degree and is nearly linear for small values of slip angle (less than 4 degrees).

The function is defined by stiffness factor *B*, shape factor *C*, peak value *D*, and curvature factor *E* and gets tuned for each tire by coefficients (a_i) using the data from the lateral force, slip angle, and normal force. The simple version of the model used in this dissertation is as follows:

$$\begin{aligned}
&C = 1.3 \\
&D = a_1 F_z^{\;2} + a_2 F_z \\
&BCD = a_3 \sin(a_4 \tan^{-1}(a_5 F_z)) \\
&B = \frac{BCD}{CD} \\
&E = a_6 F_z^{\;2} + a_7 F_z + a_8 \\
&\phi = (1-E)\alpha + \frac{E}{B}\tan^{-1}(B\alpha) \\
&F_y = D\sin(C\tan^{-1}(B\phi))
\end{aligned} \tag{7-6}$$

The normal force F_z is given in kN. This model is tuned to match the linear model with saturation in the previous section.

From the saturation model in the previous section, one data point was chosen for tuning the parameters in the peak value D equation. It was seen that for the normal tire force F_z =8.66 kN, the peak value for lateral force Fy is 3750 N. Parameters a_1 and a_2 are tuned accordingly.

$$D = 3750 = a_1(8.66)^2 + a_2(8.66) \tag{7-7}$$

The cornering stiffness was used as a slope at origin for the linear area to derive the tuning parameters for BCD.

$$C_\alpha = 25000 \quad [N/rad] = 436.3323 \quad [N/\deg] = BCD \tag{7-8}$$

$$BCD = 436.3323 = a_3 \sin(a_4 \tan^{-1}(a_5 \times 8.66))$$

Curvature factor E defines the shape of the plot. The threshold value for slip angle was used as the main parameter for tuning a_6, a_7, and a_7 by visual inspection of the plots.

$$\alpha_{threshhold} = 0.15 \quad [rad] = 8.6 \quad \deg \tag{7-9}$$

The tuning coefficients to match the linear tire model is chosen as the table below:

Table 7-1: Magic formula parameters

a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
-22.1	624.4114	467.2253	1.82	0.208	0	-0.2	-10

Figure 7-9 below shows the Magic Tire model for the selected set of parameters.

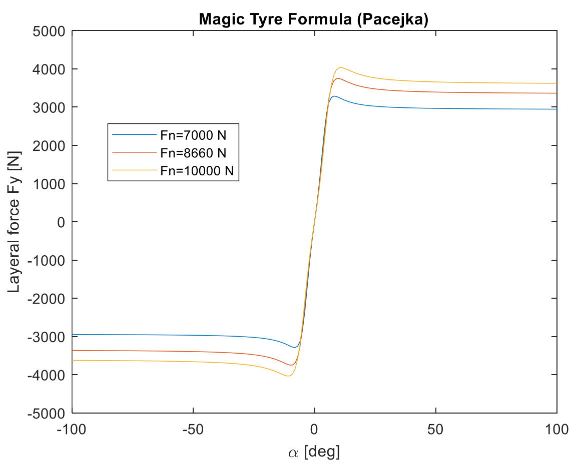

Figure 7-9 Magic Formula lateral forces versus slip angles result for three different normal forces

It appears that by increasing the normal force, the peak of the lateral force increased, but it is still bounded by slip angle.

The plant model's front and rear tire forces (F_{yf}, Fyr) are based on the bicycle model, but the normal forces are left and right normal forces (F_{zl}, F_{zr}) based on a 4DOF roll plane model. The lateral forces for front and rear are calculated as below using the Pacejka model function defined in MATLAB named (*magic*).

$$\begin{aligned} F_{zlh} &= F_{zl} / 2 \\ F_{zrh} &= F_{zr} / 2 \\ F_{yf} &= magic(\alpha_f, F_{zlh}) + magic(\alpha_f, F_{zrh}) \\ F_{yf} &= magic(\alpha_r, F_{zlh}) + magic(\alpha_r, F_{zrh}) \end{aligned} \tag{7-10}$$

Here are the simulation results for the same maneuver as the previous section.

As shown in Figure 7-10 and Figure 7-11, tracking performance behaviors are the same as the previous section for all the paradigms. A steady-state tracking error shows this tire model is incapable of producing as much lateral force as the previous model.

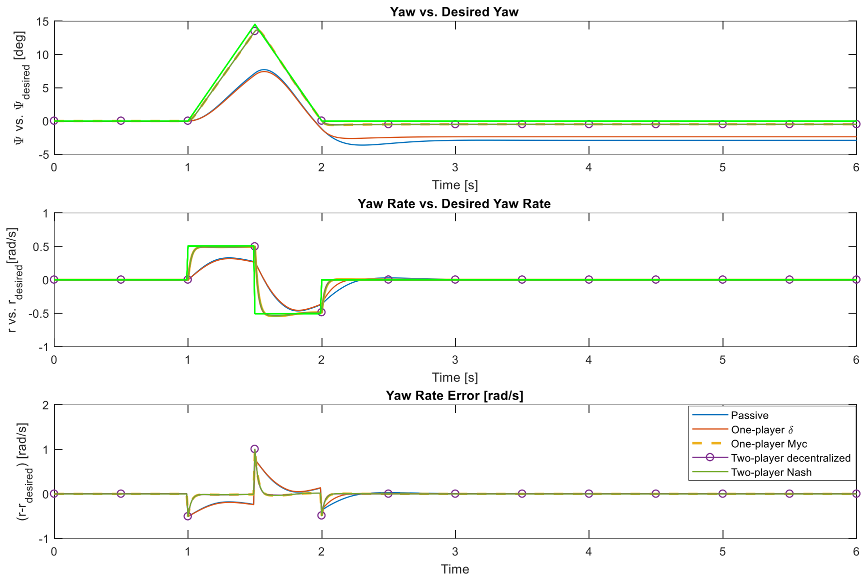

Figure 7-10: Yaw plots for Pacejka tire model (active steering vs. corrective yaw moment game)

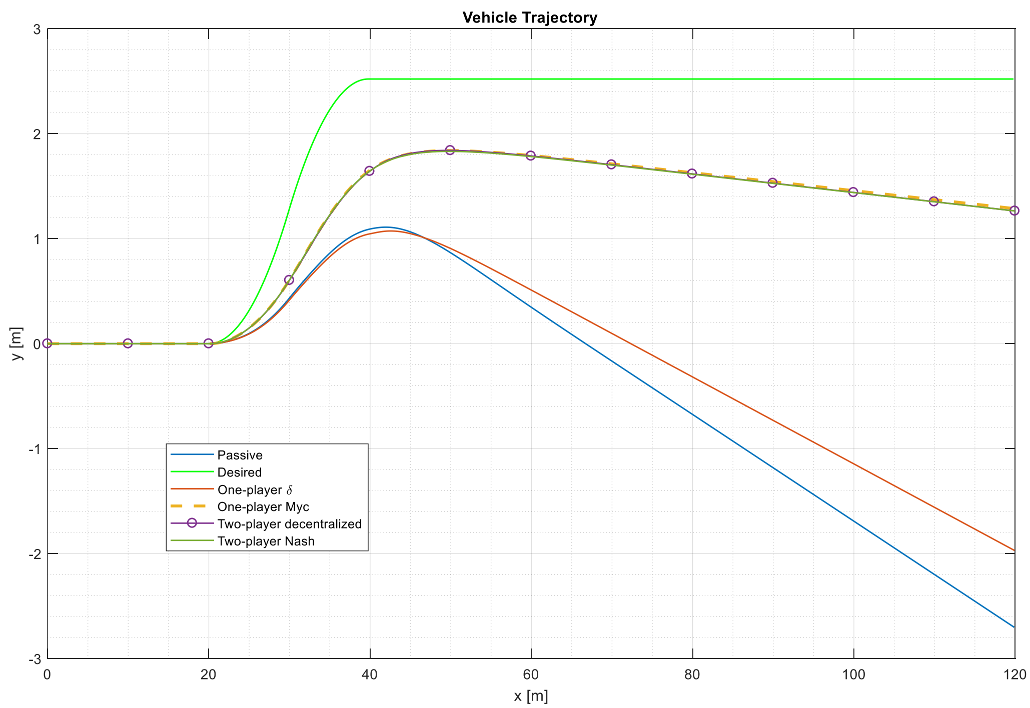

Figure 7-11: Vehicle trajectory for Pacejka tire model (active steering vs. corrective yaw moment game)

Steering angle (Figure 7-12) and corrective roll moment (Figure 7-13) control input plots show almost identical behavior for all the paradigms in the previous section. This means that Nash is scoring a lower peak for both control inputs (Figure 7-14 shows a lower corrective roll moment peak for the Nash paradigm).

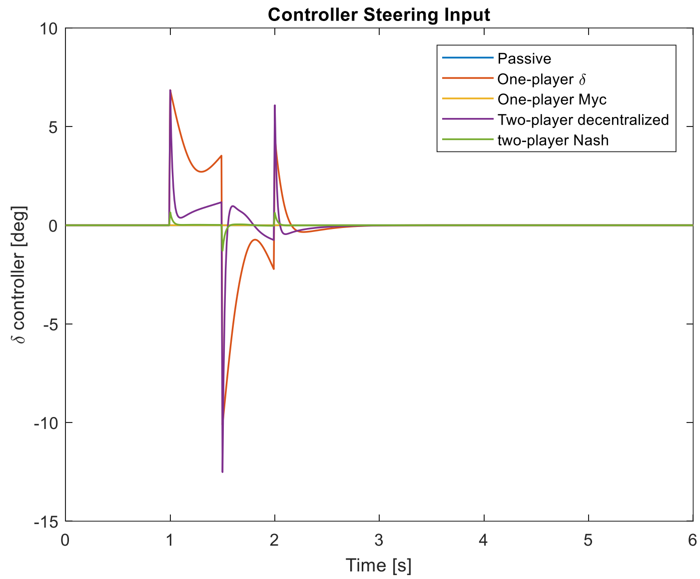

Figure 7-12: Active steering control input for Pacejka tire model (active steering vs. corrective yaw moment game)

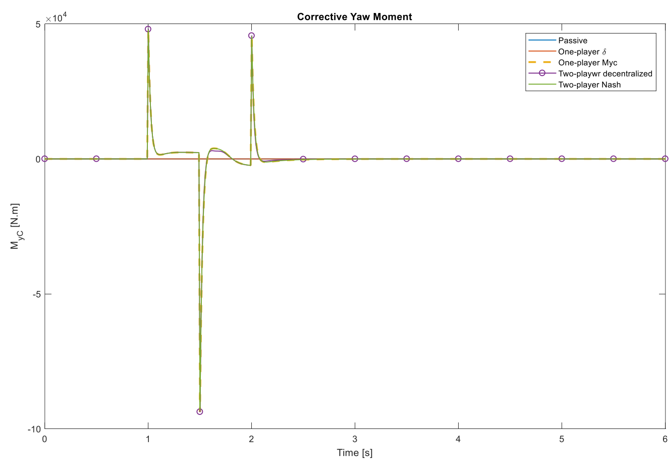

Figure 7-13: Corrective roll moment for Pacejka tire model (active steering vs. corrective yaw moment game)

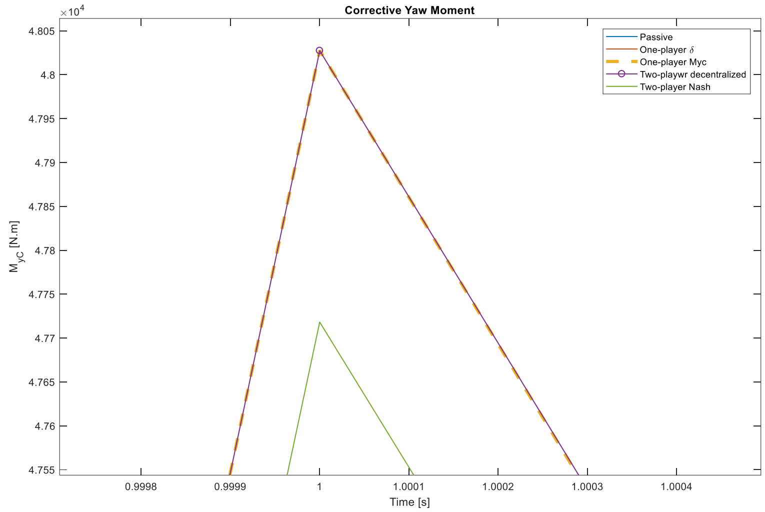

Figure 7-14: Figure 7-13 zoomed in

From Figure 7-15, It appears that the peak of lateral force for the Nash is slightly lower than decentralized for the front tire, and for the rear tire, the lateral forces are identical for the corresponding normal forces.

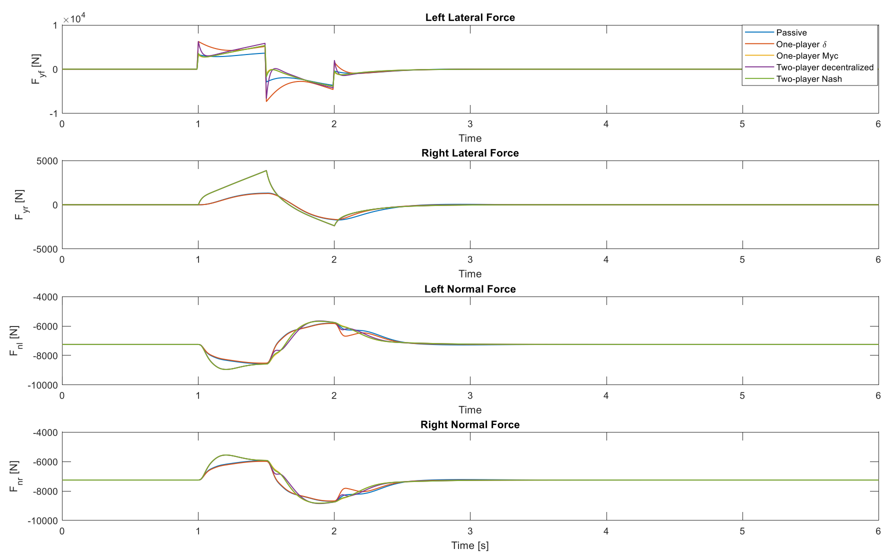

Figure 7-15: Lateral and normal tire forces for Pacejka tire model (active steering vs. corrective yaw moment game)

The slip angles and lateral acceleration are similar to the previous section (Figure 7-16 and Figure 7-17).

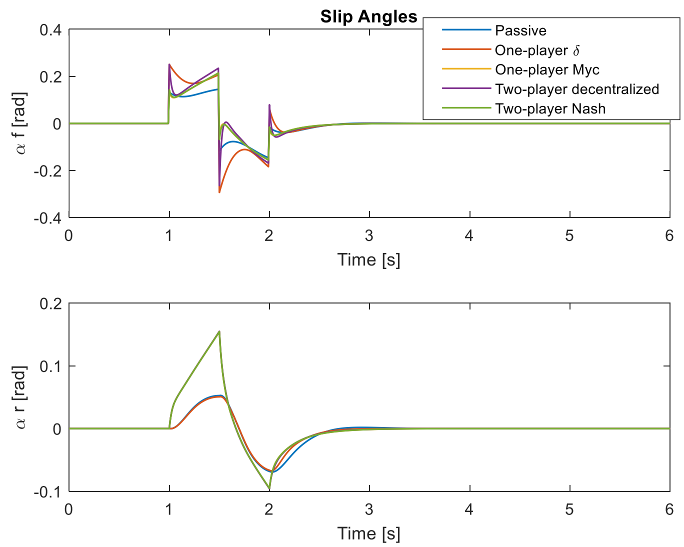

Figure 7-16: Front and rear slip angles for Pacejka tire model (active steering vs. corrective yaw moment game)

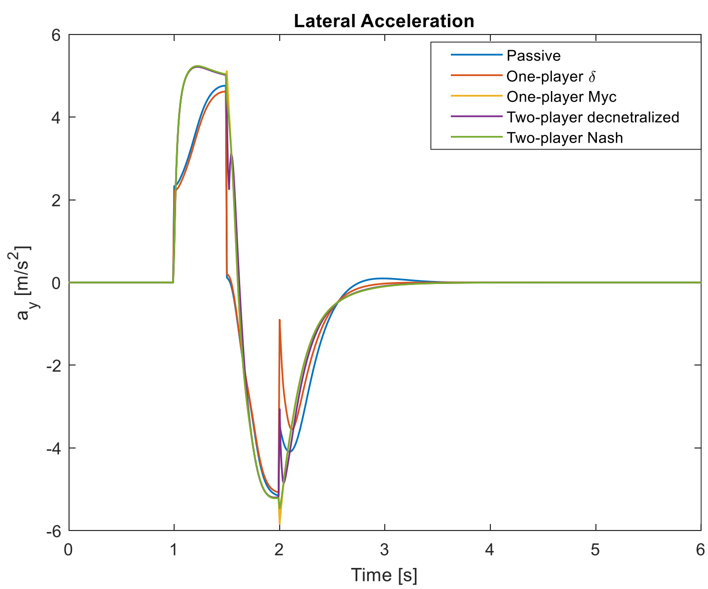

Figure 7-17: Lateral acceleration for Pacejka tire model (active steering vs. corrective yaw moment game)

7.4 Comparison of different tire models

This section compares the performance of all three tire models listed below for passive lane-change maneuver.

- Linear tire model
- Linear tire model with saturation
- Nonlinear Pacejka model

From both trajectory and yaw plots (Figure 7-18 and Figure 7-19), it is clear that tracking performance decreases for both saturation and the Pacejka model due to the saturation of lateral forces. The Pacejka model cannot track the desired yaw and yaw rate and needs a controller.

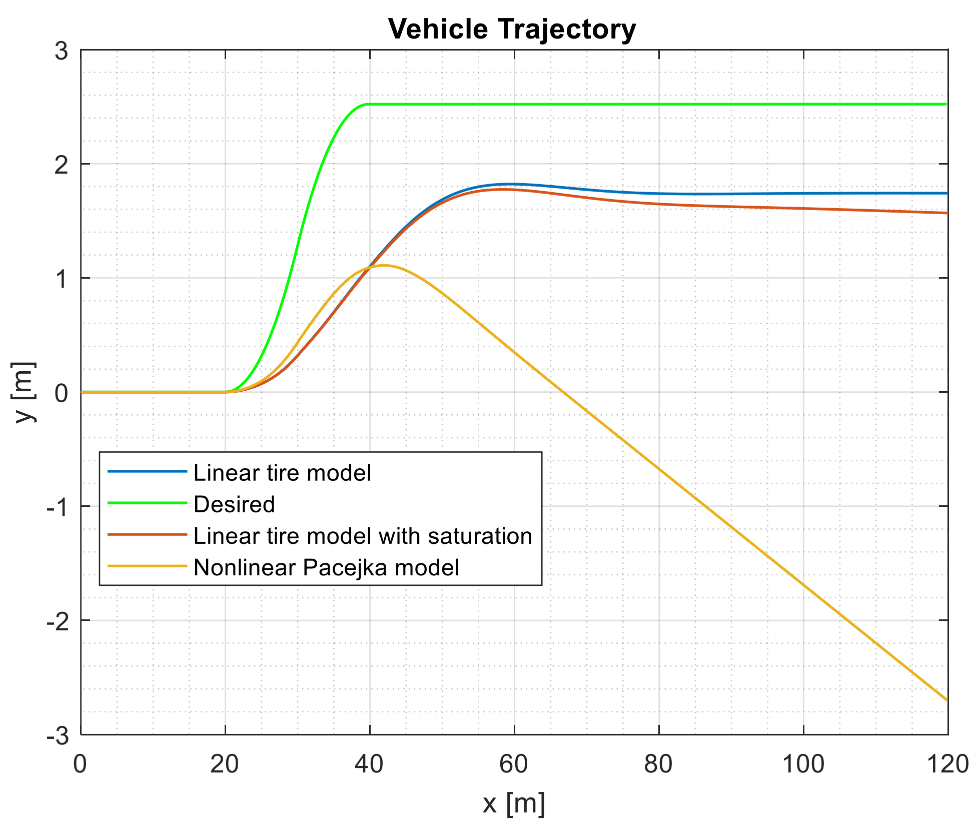

Figure 7-18: Vehicle trajectory (tire models comparison)

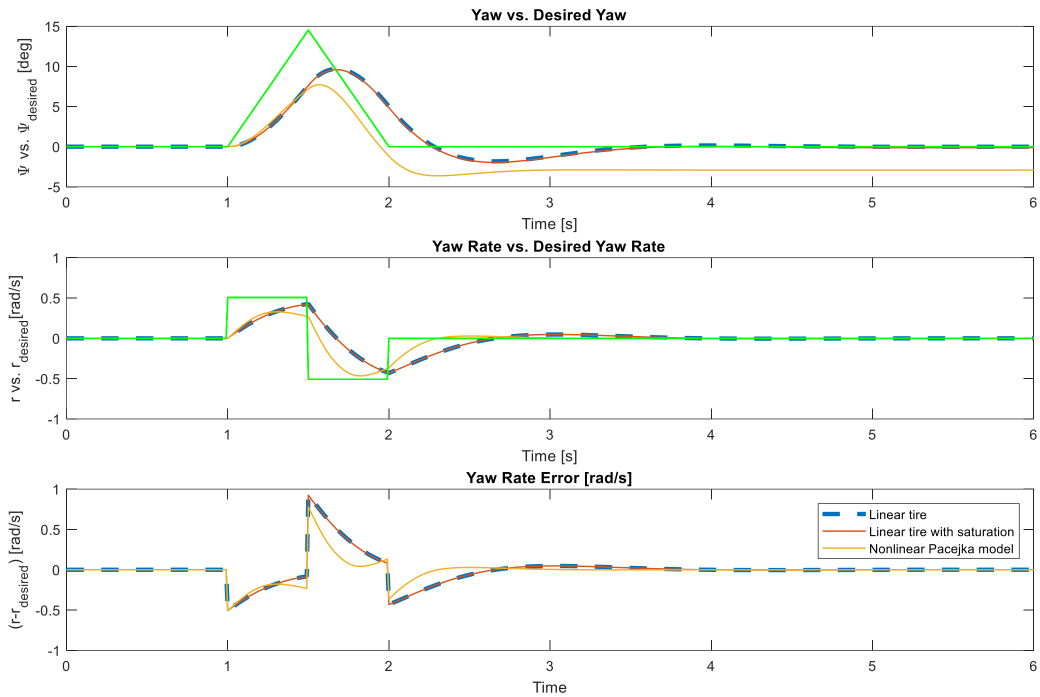

Figure 7-19:Yaw plots (tire models comparison)

Figure 7-20 shows the front and rear slip angles for the three tire models. Pacejka model is generating lower slip angles in comparison to the other models.

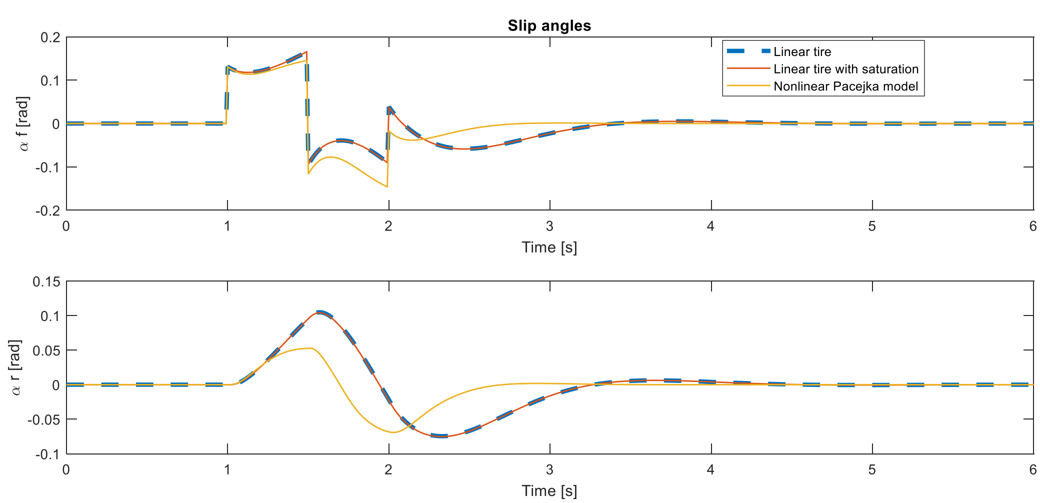

Figure 7-20: Front and rear slip angles (tire models comparison)

The lateral acceleration peak for both the saturated and Pacejka models is lower than the linear model, leading to a bigger tracking error (Figure 7-21).

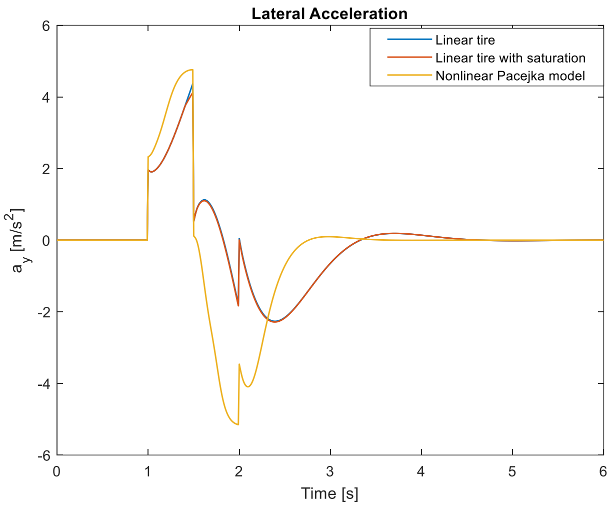

Figure 7-21: Lateral acceleration (tire models comparison)

Pacejka's model generates lower lateral forces than the others for the same range of normal forces, which can be one of the reasons that it has lower tracking performance.

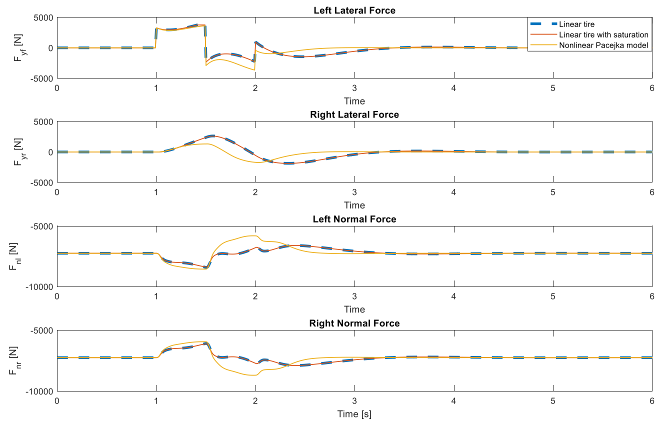

Figure 7-22: Lateral and normal tire forces (tire models comparison)

7.5 Three-player Nash $\delta_{AS} + M_{yc} + M_{\phi c}$ for plant model with nonlinear Pacejka model

Since the two-player Nash has been shown to be the optimal paradigm in a two-player-game between active steering and a corrective yaw moment, in this section, a three-player game between active steering, corrective yaw moment, and corrective roll moment has been chosen as a case study for the plant model with nonlinear Pacejka tire model. In this three-player paradigm, both active steering and yaw moment ($\delta_{AS} + M_{yc}$) improve tracking by regulating yaw rate error, whilst the corrective roll moment ($M_{\phi c}$) improves roll stability by regulating roll angle and roll rate. The gains for two-player ($\delta_{AS} + M_{yc}$) and three-player Nash ($\delta_{AS} + M_{yc} + M_{\phi c}$) are the same as what is calculated in chapter six, sections two and four.

Table 7-2 shows the control effort and total cost for these two scenarios.

Table 7-2: comparison between two-player and three-player game for Nash paradigm scenario RMS values and total cost

	Two-player ($\delta_{AS} + M_{yc}$)	Three-player Nash ($\delta_{AS} + M_{yc} + M_{\phi c}$)
δ_{RMS}	0.0015	0.0015
Myc_{RMS}	6.1073×10^3	6.1102×10^3
$M\phi c_{RMS}$	0	865.8787
Total cost J_{total}	0.2611	0.2606

The root means square of the active steering is identical with almost close RMS values for the corrective yaw moment signal as well. The total cost of the three-player is less than the two-player, making it even more inviting to utilize the three-player paradigm.

Here are the simulation results for these two paradigms alongside passive maneuver:

In trajectory and yaw rate error comparison (Figure 7-23 and Figure 7-24), both two-player and three-player paradigms show identical tracking performances. They also generate the same slip angles using the same active steering control input. From table 7-2, the RMS of the active steering input signals is identical for both two-player and three-player paradigms.

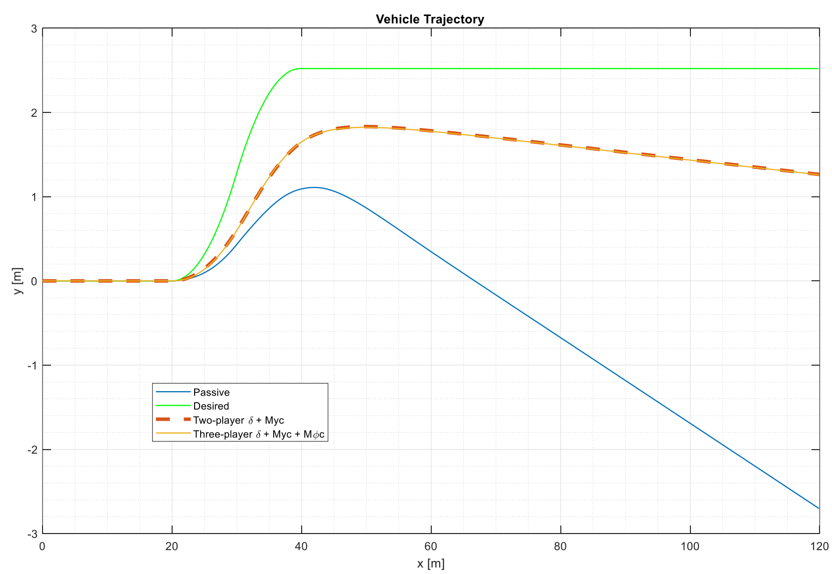

Figure 7-23: Vehicle trajectory (three-player with Pacejka tire model)

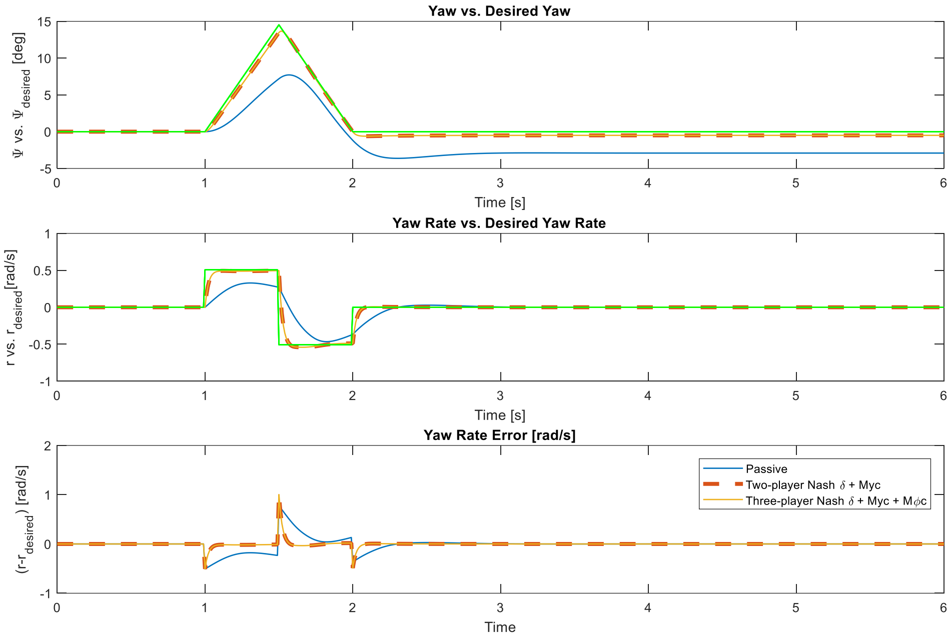

Figure 7-24: Yaw plots (three-player with Pacejka tire model)

Figure 7-25 shows the same active steering input for both two-player and three-player paradigms, which agrees with the data from Table 7-2.

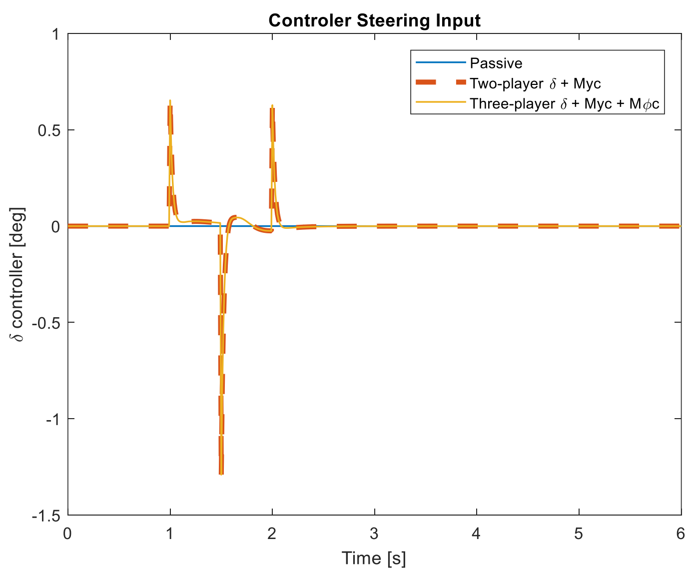

Figure 7-25: Active steering control input (three-player with Pacejka tire model)

The corrective yaw moment control signal and its root mean square are relatively identical for both two-player and three-player paradigms as well (Figure 7-26). The three-player also generates the corrective roll moment to improve roll stability. From Figure 7-27, it is clear that the roll performance decreases in the two-player paradigm relative to passive, but this problem gets fixed in the three-player paradigm by regulating roll angle and roll rate and maintaining the roll index in the safe zone (-1,1).

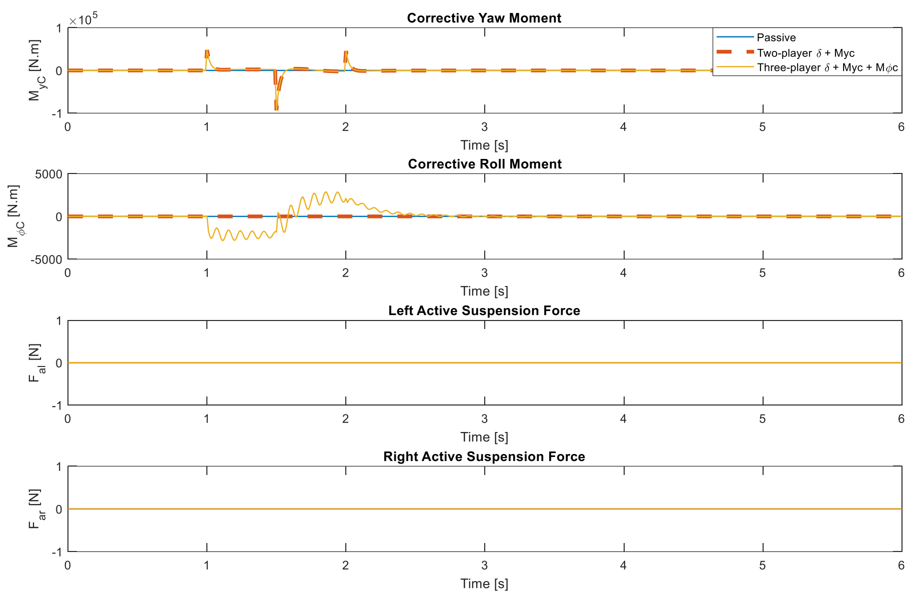

Figure 7-26: Control input signals fr corrective yaw moment, corrective roll moment, and active suspension forces (three-player with Pacejka tire model)

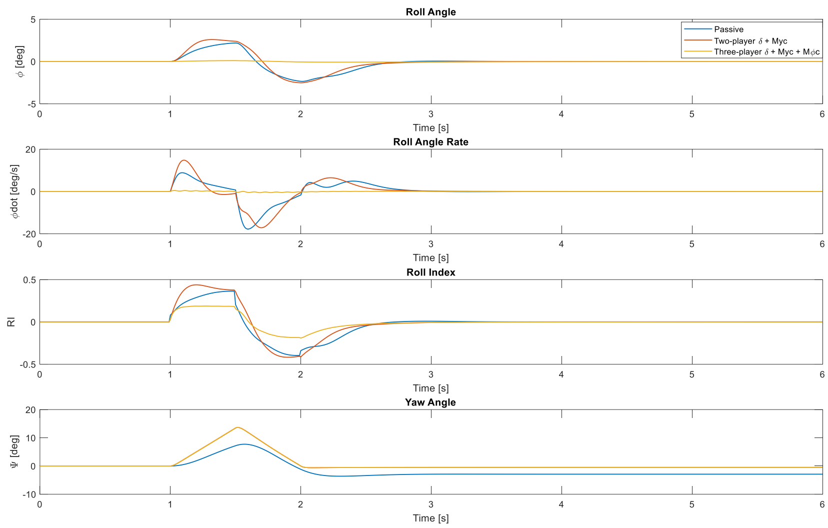

Figure 7-27: Simulation results for roll angle, roll rate, roll index, and yaw angle (three-player with Pacejka tire model)

According to Figure 7-28, both paradigms produce relatively identical lateral force and lateral acceleration even in the presence of more variation in normal force for the three-player paradigm.

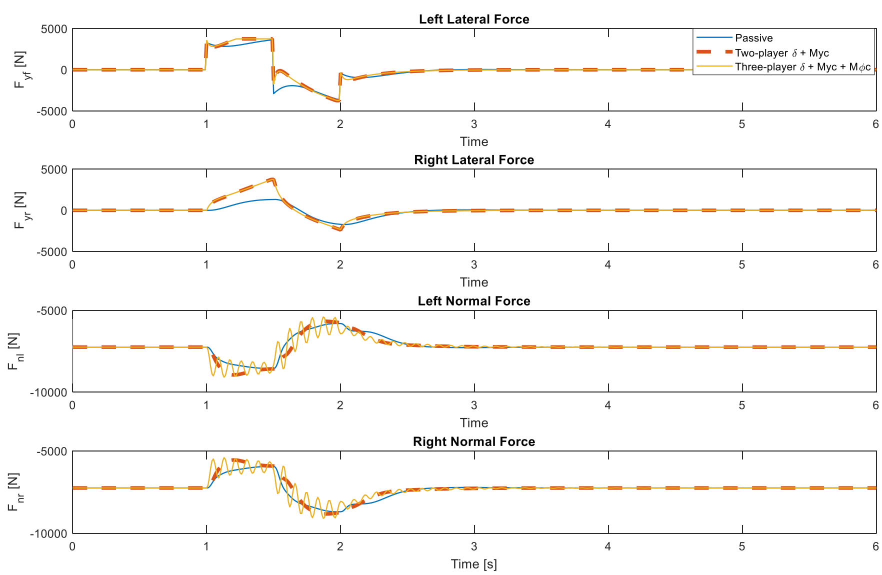

Figure 7-28: Lateral and normal tire forces (three-player with Pacejka tire model)

Identical lateral acceleration is shown for both paradigms according to Figure 7-29.

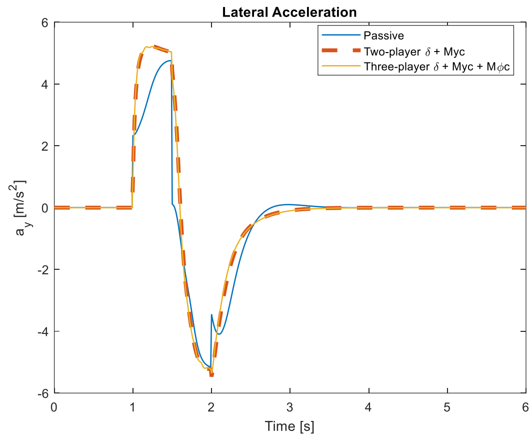

Figure 7-29: Lateral acceleration (three-player with Pacejka tire model)

As it is shown in Figure 7-30, it appears that improving roll stability in the three-player paradigm is also beneficial to reduce the variations in suspension deflection.

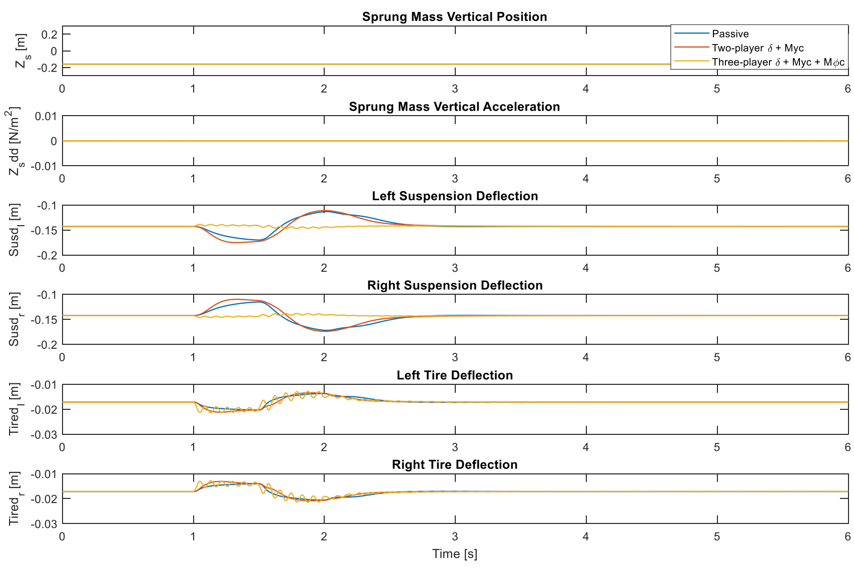

Figure 7-30: Vertical dynamic simulation results (three-player with Pacejka tire model)

7.6 Chapter Summary

In this chapter, two nonlinear tire models are used. The first model is similar to the linear tire model but with saturation. The second one is a simplified version of the famous Pacejka model, where only the lateral force is calculated based on the slip angle and normal force. The Pacejka model is tuned to have the same behavior as the linear tire model for small slip angles. This means that the slope of the lateral force versus slip angle plot is the same at the origin for all the tire models (the equivalent of cornering stiffness). A comparison is made between the tire models based on the vehicle encountering a passive lane change maneuver identical to what was presented in chapter 6. The simulation results for the two-player game between active steering and corrective yaw moment and the three-player game between active steering, corrective yaw moment, and corrective roll moment are presented using the same optimal gains from chapter 6. It was shown that in the two-player scenario, Nash still remains the optimal paradigm overall, considering the loss of traction due to saturation and nonlinearity. The comparison between the mentioned two-player and three-player scenarios shows that the three-player Nash manages to improve tracking and roll stability with a lower total cost with the benefit of reducing the suspension deflection variations.

8 CHAPTER 8: CONCLUSIONS AND FUTURE RESEARCH

8.1 Summary and Conclusions

In this dissertation, the main objective is to improve the design of controllers using differential game theory and explore the utility of game theory controllers in vehicle dynamics. First formulations were developed in which each actuator (control input/player) has its own cost function. In a decentralized paradigm, each player cost function consists of its own control objective and control action. In a game theory approach, the Nash paradigm is introduced where each player's cost function includes its control objective, its control action, and other players' control actions. In the beginning, the problem is defined mathematically in quadratic form, and the solutions are presented for the two-player and three-player game scenarios. The control problem is solved for two frameworks of *LQR* (where there is no coupling between the output objectives and control inputs) and *CCOR* (where there is a coupling between output objectives and control inputs). The solution for single-player *LQR*, single-player *CCOR*, and two-player discrete system *LQR* already exist in the literature. As its primary contribution, this thesis develops the solution for two-player and three-player games for continuous linear systems for both *LQR* and *CCOR*.

After providing the theoretical solutions, the controllers are tested in the scenarios below:

- The two-player player game between active steering and corrective yaw moment (for plant model with both linear tire model and nonlinear tire model)

- The two-player player game between active steering and corrective roll moment (for plant model with only linear tire model)
- Three player game between active steering, corrective yaw moment, and corrective roll moment (for plant model with both linear tire model and nonlinear tire model)
- The two-player player game between active suspension and corrective roll moment (for plant model with only linear tire model)

The simulation results are presented for single-player paradigms, the decentralized combination of two/three-player paradigms, and two/three-player Nash paradigms for each scenario. For most cases, it was shown that the Nash solution shows good performance in regulating the output objectives and requires less control effort with lower control signal peaks. Note that these results can also be tuned by changing the state and input weights in the cost functions, so the scenarios were simulated for a similar set of weights to make a fair comparison. It appears that the Nash paradigm manages to find the optimal amount of control inputs required to achieve the control objectives by dividing the burden of control between the control inputs. This highlights the benefit of the game-theoretical approach where the counter effects of control inputs on each other's objective are considered in the players' cost functions. As a by-product of this benefit, scoring lower peaks in control input signals can be helpful in scenarios where the decentralized approach fails due to the control input signal reaching the saturation limits. The Nash paradigm also showed better performance in simulations with the nonlinear Pacejka tire model, where the vehicle encountered a rollover in the decentralized paradigm. Here is the summary of conclusions achieved from the simulations:

- ❖ For most scenarios, the Nash paradigm performs well in regulating the output (objective) and requires <u>less control effort</u> with lower control signal peaks.

- ❖ It appears that the Nash paradigm finds the optimal amount of control inputs required by dividing the burden of control between the control inputs
- ❖ Nash paradigm's Lower peak signal can be helpful in scenarios where the decentralized paradigm fails due to the control input signal reaching the saturation limits.

8.2 Future Research Directions

The Nash solution was shown to be a promising approach among all the other paradigms for the selected scenarios in this thesis. Yet still, there are so many scenarios in vehicle dynamics where the cooperative and non-cooperative nature of integrated control for two/three-player games can be explored. This dissertation didn't explore the control objectives in longitudinal dynamics, pitch and heave motions, and control actions like differential braking and torque distribution. Even for the same scenarios that were explored here, the cost functions can still be altered by adding other control objectives like suspension deflections or tire deflection. There are so many combinations of two/three-player games that can be used as a benchmark for game theory applications.

The plant model used in this research is the combination of the bicycle model and roll plane model. This model is still far away from the complexity of a complete or more sophisticated vehicle model in terms of the states, nonlinearity, and load transfer. A full car model with higher degrees of freedom is required to investigate the performance and robustness of the controllers presented in this thesis. This step is crucial for testing the game theory approach before implementing it practically on actual vehicles.

The solution presented in this dissertation is for two/three-player continuous linear dynamic systems. This solution can be extended to a multi-player game (for more than three players) to achieve optimal global stability for integrated vehicle dynamics control. The game theory concept can also be explored for nonlinear control models by using nonlinear control methods like neural networks, genetic algorithms, or fuzzy logic.

9 REFERENCES

1. Rideout, D.G., Personal communication with D.J.N. Overholt, Editor. 2008: U.S. Army Tank-Armament and Automotive Command.
2. Peng, H., & Tomizuka, M. (1993). Preview control for vehicle lateral guidance in highway automation.
3. Johnson, M. A. (2011). *Differential game-based control methods for uncertain continuous-time nonlinear systems*. University of Florida.
4. Rajamani, R. (2011). *Vehicle dynamics and control*. Springer Science & Business Media.
5. Hać, A. (1992). Optimal linear preview control of active vehicle suspension. *Vehicle system dynamics*, *21*(1), 167-195.
6. Aripin, M. K., Md Sam, Y., Danapalasingam, K. A., Peng, K., Hamzah, N., & Ismail, M. F. (2014). A review of active yaw control system for vehicle handling and stability enhancement. *International journal of vehicular technology*, *2014*.
7. CarSim. Available at http://www.carsim.com: Mechanical Simulation Corp.
8. Gordon, T., Howell, M., & Brandao, F. (2003). Integrated control methodologies for road vehicles. *Vehicle System Dynamics*, *40*(1-3), 157-190.Hrovat, D. (1997). Survey of advanced suspension developments and related optimal control applications. *Automatica*, *33*(10), 1781-1817.
9. Bender, E. K. (1968). Optimum linear preview control with application to vehicle suspension.
10. Marzbanrad, J., Ahmadi, G., Hojjat, Y., & Zohoor, H. (2002). Optimal active control of vehicle suspension system including time delay and preview for rough roads. *Journal of Vibration and Control*, *8*(7), 967-991.

11. Shim, T., & Margolis, D. (2005). Dynamic normal force control for vehicle stability enhancement. *International journal of vehicle autonomous systems*, *3*(1), 1-14.
12. Chou, H., & D'Andréa-Novel, B. (2005). Global vehicle control using differential braking torques and active suspension forces. *Vehicle System Dynamics*, *43*(4), 261-284.
13. Tchamna, R., Youn, E., & Youn, I. (2014). Combined control effects of brake and active suspension control on the global safety of a full-car nonlinear model. *Vehicle System Dynamics*, *52*(sup1), 69-91.
14. Chu, D. F., Li, G. Y., & Hu, J. (2010, August). Hierarchical control for integrated vehicle dynamic stability. In *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)* (Vol. 3, pp. V3-590). IEEE.
15. March, C., Shim, T., & Zhang, Y. (2005, January). Integrated Control of Tire Normal Forces and Active Front Steering to Enhance Vehicle Handling. In *ASME International Mechanical Engineering Congress and Exposition* (Vol. 42150, pp. 215-224).
16. Li, D., Du, S., & Yu, F. (2008). Integrated vehicle chassis control based on direct yaw moment, active steering and active stabiliser. *Vehicle System Dynamics*, *46*(S1), 341-351.
17. Sharp, R. S., Casanova, D. A. N. I. E. L. E., & Symonds, P. (2000). A mathematical model for driver steering control, with design, tuning and performance results. *Vehicle system dynamics*, *33*(5), 289-326.
18. Yim, S. (2011). Design of a preview controller for vehicle rollover prevention. *IEEE Transactions on Vehicular Technology*, *60*(9), 4217-4226.

19. Na, X., & Cole, D. J. (2013). Linear quadratic game and non-cooperative predictive methods for potential application to modelling driver–AFS interactive steering control. *Vehicle system dynamics*, *51*(2), 165-198.
20. Na, X., & Cole, D. J. (2014). Game-theoretic modeling of the steering interaction between a human driver and a vehicle collision avoidance controller. *IEEE Transactions on Human-Machine Systems*, *45*(1), 25-38.
21. Tamaddoni, S. H., Taheri, S., & Ahmadian, M. (2009, October). Optimal VSC design based on Nash strategy for differential 2-player games. In *2009 IEEE International Conference on Systems, Man and Cybernetics* (pp. 2415-2420). IEEE.
22. Tamaddoni, S. H., Taheri, S., & Ahmadian, M. (2010, June). Cooperative DYC system design for optimal vehicle handling enhancement. In *Proceedings of the 2010 American Control Conference* (pp. 1495-1500). IEEE.
23. Tamaddoni, S. H., Taheri, S., & Ahmadian, M. (2011). Optimal preview game theory approach to vehicle stability controller design. *Vehicle system dynamics*, *49*(12), 1967-1979.
24. Na, X., & Cole, D. J. (2014). Game-theoretic modeling of the steering interaction between a human driver and a vehicle collision avoidance controller. *IEEE Transactions on Human-Machine Systems*, *45*(1), 25-38.
25. Bakker, E., Nyborg, L., & Pacejka, H. B. (1987). Tyre modelling for use in vehicle dynamics studies. *SAE Transactions*, 190-204.
26. Huang, H. H., Yedavalli, R. K., & Guenther, D. A. (2012). Active roll control for rollover prevention of heavy articulated vehicles with multiple-rollover-index minimisation. *Vehicle system dynamics*, *50*(3), 471-493.
27. Naidu, D. S. (2002). *Optimal control systems*. CRC press.

28. Franklin, G. F., Powell, J. D., & Workman, M. L. (1998). *Digital control of dynamic systems* (Vol. 3). Reading, MA: Addison-wesley.
29. Na, X., & Cole, D. J. (2019). Modelling of a human driver' s interaction with vehicle automated steering using cooperative game theory. *IEEE/CAA Journal of Automatica Sinica*, *6*(5), 1095-1107.
30. Li, M., Song, X., Cao, H., Wang, J., Huang, Y., Hu, C., & Wang, H. (2019). Shared control with a novel dynamic authority allocation strategy based on game theory and driving safety field. *Mechanical Systems and Signal Processing*, *124*, 199-216.
31. Yang, K., Zheng, R., Ji, X., Nishimura, Y., & Ando, K. (2018, June). Application of Stackelberg game theory for shared steering torque control in lane change maneuver. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 138-143). IEEE.
32. Li, N., Oyler, D., Zhang, M., Yildiz, Y., Girard, A., & Kolmanovsky, I. (2016, December). Hierarchical reasoning game theory based approach for evaluation and testing of autonomous vehicle control systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)* (pp. 727-733). IEEE.
33. Yu, H., Tseng, H. E., & Langari, R. (2018). A human-like game theory-based controller for automatic lane changing. *Transportation Research Part C: Emerging Technologies*, *88*, 140-158.
34. Zheng, Y., Ding, W., Ran, B., Qu, X., & Zhang, Y. (2020). Coordinated decisions of discretionary lane change between connected and automated vehicles on freeways: a game theory-based lane change strategy. *IET Intelligent Transport Systems*, *14*(13), 1864-1870.
35. Rahmati, Y., Hosseini, M. K., & Talebpour, A. (2021). Helping Automated Vehicles With Left-Turn Maneuvers: A Game Theory-Based Decision Framework for Conflicting Maneuvers at Intersections. *IEEE Transactions on Intelligent Transportation Systems*.
36. Li, N., Zhang, M., Yildiz, Y., Kolmanovsky, I., & Girard, A. (2019). Game theory-based traffic modeling for calibration of automated driving algorithms. In *Control Strategies for Advanced Driver Assistance Systems and Autonomous Driving Functions* (pp. 89-106). Springer, Cham.
37. Jugade, S. C., Victorino, A. C., & Cherfaoui, V. B. (2019, October). Shared Driving Control between Human and Autonomous Driving System via Conflict resolution using Non-Cooperative Game Theory. In *2019 IEEE*

Intelligent Transportation Systems Conference (ITSC) (pp. 2141-2147). IEEE.
38. Kim, C., & Langari, R. (2014). Game theory based autonomous vehicles operation. *International Journal of Vehicle Design*, *65*(4), 360-383.
39. Zohdy, I. H., & Rakha, H. (2012, September). Game theory algorithm for intersection-based cooperative adaptive cruise control (CACC) systems. In *2012 15th International IEEE Conference on Intelligent Transportation Systems* (pp. 1097-1102). IEEE.
40. Alvarez, I., & Poznyak, A. (2010, October). Game theory applied to urban traffic control problem. In *ICCAS 2010* (pp. 2164-2169). IEEE.
41. Fan, H., Jia, B., Tian, J., & Yun, L. (2014). Characteristics of traffic flow at a non-signalized intersection in the framework of game theory. *Physica A: Statistical Mechanics and its Applications*, *415*, 172-180.
42. Wu, W., Liang, Z., Luo, Q., & Ma, F. (2018). Game theory modeling for vehicle U-Turn behavior and simulation based on cellular automata. *Discrete Dynamics in Nature and Society*, *2018*.
43. Li, D., Liu, G., & Xiao, B. (2021). Human-like Driving Decision at Unsignalized Intersections Based on Game Theory. *arXiv preprint arXiv:2112.06415*.
44. Zhang, B., & Lu, S. (2020). Fault-tolerant control for four-wheel independent actuated electric vehicle using feedback linearization and cooperative game theory. *Control Engineering Practice*, *101*, 104510.
45. An, Q., Cheng, S., Li, C., Li, L., & Peng, H. (2021). Game Theory-Based Control Strategy For Trajectory Following of Four-Wheel Independently Actuated Autonomous Vehicles. *IEEE Transactions on Vehicular Technology*, *70*(3), 2196-2208.
46. Han, B. J., Gang, X. N., Wan, C. Y., Lu, W., & Zhi, S. C. (2010, July). Multi-objective optimization design of passive suspension parameters based on collusion cooperation game theory. In *2010 8th World Congress on Intelligent Control and Automation* (pp. 118-125). IEEE.
47. Dextreit, C., & Kolmanovsky, I. V. (2013). Game theory controller for hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, *22*(2), 652-663.
48. Ramos, M. A., Moura, M. C., Lins, I. D., & Ramos, F. S. (2021). The use of Game Theory for Autonomous Systems Safety: An Overview.

10 APPENDIX

10.1 Numerical Vehicle Data for all the models in this thesis

Vehicle Parameters	Numerical Vehicle Data
front cornering stiffness	C_f = 25000 [*N/rad*]
rear cornering stiffness	C_r = 25000 [*N/rad*]
height of roll axis overground	h_r = 0.3 [*m*]
nominal height of sprung mass CG over roll axis	h_s = 0.3 [*m*]
distance from front axle to CG	l_f = 1.12 [*m*]
distance from rear axle to CG	l_r = 1.68 [*m*]
sprung mass	M_s = 1330 [*kg*]
left unsprung mass	m_{ul} = 37×2 [*kg*] (front and rear axle combined)
right unsprung mass	m_{ur} = 37×2 [*kg*] (front and rear axle combined)
track width	t = 1.6 [*m*]
sprung mass roll moment of inertia	I_x = 283 [$Kg.m^2$]
yaw moment of Inertia	I_z = 2424 [$Kg.m^2$]
acceleration due gravity	g = 9.81 [m/s^2]
longitudinal velocity	V_x = 20 [*m/s*]
left suspension stiffness	K_{sl}=22891×2 [*N/m*] (front and rear axle combined)
right suspension stiffness	K_{sr}=22891×2 [*N/m*] (front and rear axle combined)
left tire stiffness	K_{tl}=211720×2 [*N/m*] (front and rear axle combined)
right tire stiffness	K_{tr}=211720×2 [*N/m*] (front and rear axle combined)
left suspension damping	B_{sl}=2081×2 [*N.s/m*] (front and rear axle combined)
right suspension damping	B_{sr}=2081×2 [*N.s/m*] (front and rear axle combined)
left tire damping	B_{tl}=100×2 [*N.s/m*] (front and rear axle combined)
right tire damping	B_{tl}=100×2 [*N.s/m*] (front and rear axle combined)
Road adhesion coefficient	$\mu = 1$

10.2 Control Model For The Game Between Active Steering And Corrective Roll Moment

Assume the state vector and control inputs described as follows:

$$x = [\phi, \dot{\phi}, v_y, r]'$$

Steering angle : $u_1 = \delta$

Corrective roll moment: $u_2 = M_{\phi c}$

The state-space representation can be written as below from the previous section equations

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (I_x + M_s h_s^2) & -M_s h_s & 0 \\ 0 & -M_s h_s & M & 0 \\ 0 & 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{v}_y \\ \dot{r} \end{bmatrix} =$$

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ M_s g h_s - K & -C & 0 & M_s h_s V_x \\ 0 & 0 & -(C_f + C_r)\mu / V_x & (C_r l_r - C_f l_f)\mu / V_x - M V_x \\ 0 & 0 & (C_r l_r - C_f l_f)\mu / V_x & (C_r l_r^2 + C_f l_f^2)\mu / V_x \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v_y \\ r \end{bmatrix}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ C_f \mu \\ C_f l_f \mu \end{bmatrix} \delta + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} M_{\varphi c}$$

where K and C are roll stiffness and damping coefficients for passive suspensions and are calculated from the left and right wheels' suspension system using the formulas below:

$$K = (K_{sl} + K_{sr})(t^2 / 4),$$
$$C = (B_{sl} + B_{sr})(t^2 / 4)$$

The state-space representation structure of this dynamic model can be written as:

$$E\dot{x} = Ux + V_1 u_1 + V_2 u_2$$

$$A = E^{-1}U, B_1 = E^{-1}V_1, B_2 = E^{-1}V_2$$

$$\dot{x} = Ax + B_1 u_1 + B_2 u_2 = Bu$$

Using the parameters from appendix 10.1, the *A*, *B1*, and *B2* matrices are as follows

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -185.3876 & -18.0597 & -2.2879 & 0.6406 \\ -50.0471 & -4.8754 & -2.3091 & -19.3534 \\ 0 & 0 & 0.2888 & -2.1023 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ 22.8790 \\ 23.0911 \\ 11.5512 \end{bmatrix}, B_2 = 10^{-3} \times \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.4125 \end{bmatrix}, B = [B_1, B_2]$$

Eigenvalues and natural frequencies of the system are calculated from the state matrix A and listed below:

Eigenvalues:

-1.8878 - 2.3821i, -1.8878 + 2.3821i, -9.3478 - 9.8560i, -9.3478 + 9.8560i

Natural frequencies:

2.3821, 9.3478 [rad/s]

The rank of matrix *A* and the controllability matrix four means that the system is fully controllable. The Matlab code below is used to confirm this result:

```
Arank=rank(A);
Co=ctrb(A,B);
Corank=rank(Co);
```

The observability can be checked through to the output matrix. The output matrix for this scenario is either yaw rate error regulation or roll angle and roll rate regulation. The output of the code below shows that both observability matrices are full rank (rank is four) which means the system is fully observable.

```
C1=[0 0 0 1]; %for delta regulating er
C2=[1 0 0 0;  %for Mzc regulating [phi,phid]
      0 1 0 0];
Ob1=obsv(A,C1);
Ob2=obsv(A,C2);
Ob1rank=rank(Ob1);
Ob2rank=rank(Ob2);
```

10.3 Roll Plane Linear Control Model (Vertical Dynamics)

The roll plane control model can investigate the game between active suspension forces and the corrective roll moment. The model presented in the model development section can be simplified, considering the state vector and control inputs below:

$$X_c = [z_s, \phi, z_{ul}, z_{ur}, \dot{z}_s, \dot{\phi}, \dot{z}_{ul}, \dot{z}_{ur}]^T$$

Corrective roll moment: $u_1 = M_{\phi c}$

Active suspension forces: $u_2 = \left[F_{al}, F_{ar}\right]$

The state-space representation structure of this dynamic model is similar to the previous section

$$E\dot{x} = Ux + V_1 u_1 + V_2 u_2$$

$$A = E^{-1}U, B_1 = E^{-1}V_1, B_2 = E^{-1}V_2$$

$$\dot{x} = Ax + B_1 u_1 + B_2 u_2$$

in which matrices E, U, V_1, and V_2 are defined as follows.

$$
E = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & M_s & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & I_x & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & m_{ul} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{ur}
\end{bmatrix},
$$

$$
U = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
-(K_{sl}+K_{sr}) & (K_{sr}-K_{sl})t/2 & K_{sl} & K_{sr} & -(\mathrm{B}_{sl}+B_{sr}) & (\mathrm{B}_{sr}-B_{sl})t/2 & B_{sl} & B_{sr} \\
(K_{sr}-K_{sl})t/2 & -(K_{sl}+K_{sr})\mathrm{t}^2/4-M_s g h_s & -K_{sl}t/2 & K_{sr}t/2 & (\mathrm{B}_{sr}-B_{sl})t/2 & -(\mathrm{B}_{sl}+B_{sr})\mathrm{t}^2/4 & -B_{sl}t/2 & B_{sr}t/2 \\
K_{sl} & -K_{sl}t/2 & -(K_{sl}+K_{tl}) & 0 & B_{sl} & -B_{sl}t/2 & -(\mathrm{B}_{sl}+B_{tl}) & 0 \\
K_{sr} & K_{sr}t/2 & 0 & -(K_{sr}+K_{tr}) & B_{sr} & B_{sr}t/2 & 0 & -(\mathrm{B}_{sr}+B_{tr})
\end{bmatrix}
$$

$$
V_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1/t \\ -1/t \end{bmatrix}, V_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \\ -t/2 & t/2 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}
$$

Using the parameters from appendix 10.1, the A, B_1, and B_2 matrices are as follows:

$$A = 10^3 \times \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -0.0688 & 0 & 0.0344 & 0.0344 & -0.0063 & 0 & 0.0031 & 0.0031 \\ 0 & -0.2209 & -0.1294 & 0.1294 & 0 & -0.0188 & -0.0118 & 0.0118 \\ 0.6187 & -0.4949 & -6.3408 & 0 & 0.0562 & -0.0450 & -0.0589 & 0 \\ 0.6187 & 0.4949 & 0 & -6.3408 & 0.0562 & 0.0450 & 0 & -0.0589 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.0035 \\ 0.0084 \\ -0.0084 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.0008 & 0.0008 \\ -0.0028 & 0.0028 \\ -0.0135 & 0 \\ 0 & -0.0135 \end{bmatrix}, \; B = [B_1, B_2]$$

Eigenvalues and natural frequencies of the system are calculated from the state matrix A and listed below:

Eigenvalues:

-29.9469 +71.7392i, -29.9469 -71.7392i, -2.6554 + 7.6247i, -2.6554 - 7.6247i,

-29.9542 +66.3722i, -29.9542 -66.3722i, -8.9311 +12.6583i, -8.9311 -12.6583i

Natural frequencies:

7.6247, 12.6583, 66.3722 , 71.7392 [rad/s]

Similar to the previous section, the rank of matrix *A* and the controllability matrix are calculated as eight, which means that the system is fully controllable.