**Abstract**

Autonomous vehicles navigate through their environments at three steps. These are perception, planning and control steps. In this study control step is focused. To control an autonomous vehicle there are several methods like geometric controllers such as Stanley Controller [5] or Pure Pursuit Controller [4]. Also some classical methods like Proportional Integral Derivative (PID) controllers are still used in the literature. Nowadays optimal control is being preferred in order to predict vehicle maneuvers in a receding horizon. Model Predictive Control (MPC) is used mainly for this reason. As the controller name suggest MPC needs plant’s dynamic model which might consist of vehicle’s equation of motions, transfer functions etc. In this research in a high fidelity simulation software Carsim; neural network (NN) model is proposed for estimation of vehicle dynamics during its lateral motion. Neural networks technique is used to predict vehicle’s angular & lateral velocities. The results show that NN model is more accurate when compared to simple physics based models.

**Keywords:** autonomous vehicle, lateral dynamics, neural network, nonparametric modelling

1. **Introduction**

The modelling of vehicle dynamics has been studied for a long time in order to design a high performance controller for driver assistance systems and autonomous navigation.

Vehicle models have been constructed analytically in the form of dynamical equations to predict vehicle motions in three degree of freedoms. These are longitudinal, lateral and normal directions. [1][2]. However, these equations based models are simple and efficient they lack accuracy in real time or high fidelity simulation models.

To create a high performance controller, it’s inevitable to model vehicle accurately. Most of the literature on vehicle models depend on models as known as single track or bicycle model. These models are parameter dependent; however, vehicle dynamics are highly nonlinear. In order to model vehicle dynamics nonlinearly one of the nonparametric modelling methods, which is neural networks based modelling, is studied.

Buraya literatür taraması gelecek

Chapter 2 describes the single track vehicle dynamics model for comparison with neural network based model. In Chapter 3, recurrent neural networks are described and modelling methodology is expressed used in this paper. Chapter 4 compares the mathematical vehicle model and neural network based model outputs.

1. **Mathematical Vehicle Model**

This chapter introduces the simplified mathematical models of the vehicle dynamics. Since the vehicle has three degree of freedoms there needs to be three different equations. In this chapter single track model is used. Variations of this model can be referred as car-like robot model, bicycle model. [3]

**Kinematic Model of Vehicle Motion**

This sections introduces a kinematic model of vehicle motion. In kinematics there is no force interaction in the model. This modelling technique is suitable under low speed maneuvers such as parking. This method is useful for path tracking problems of car-like mobile robots. In this method, vehicle motion is assumed planar and coordinate frame convention can be seen in Figure 1.

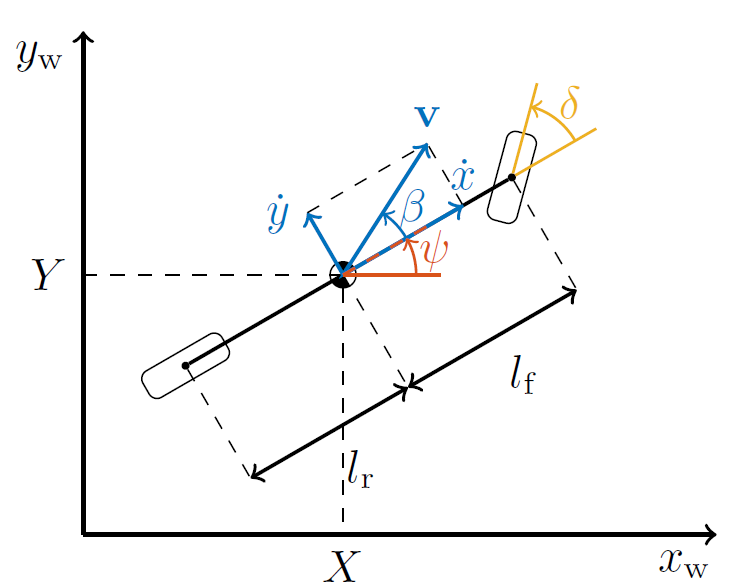


Figure 1 Kinematic Bicycle Model

The kinematic equations of motion the single track model in Figure 1 are:

Equation numaraları gelecek.

Where;

vehicle coordinates in global frame and can be found by integrating .

is the heading direction in global frame.

is the magnitude of the velocity vector, is the vehicle acceleration in vehicle frame.

is the slip angle of the vehicle.

denotes steering angle of the front wheel.

* 1. **Dynamic Model of Vehicle Motion**

This section introduces the mathematical equations of the dynamic vehicle motion. In dynamical motion equations force interactions occur on the vehicle. As mentioned in the previous section vehicle motion is assumed as planar, by doing so roll dynamics and lateral load transfer are neglected. Coordinate frame convention and forces acting on vehicle and tire can be seen in Figure 2.

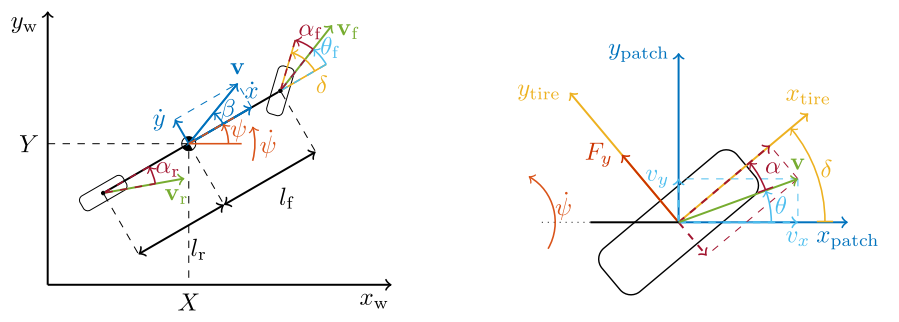


Figure 2 Coordinate frame convention used in the dynamic mode

Dinamik model denklemleri gelecek

1. **Neural Network Model**
2. **Results and Discussion**

[1] J. Filip, “Trajectory Tracking for Autonomous Vehicles,” 2018.

Jan filiip,

Modeling of vehicle dynamics from real vehicle measurements

Using a neural network with two-stage hybrid learning

For accurate long-term prediction

Comparison of architectures for

Neural-based modeling of lateral

Vehicle dynamics 1

Autonomous racing using

Model predictive control

Modeling lateral motion of a vehicle using neural networks

Technique dr. Abdullah dhayea assi

Neural network vehicle models for high-performance

Automated driving nathan a. Spielberg\*, matthew brown, nitin r. Kapania, john c. Kegelman, j. Christian gerdes

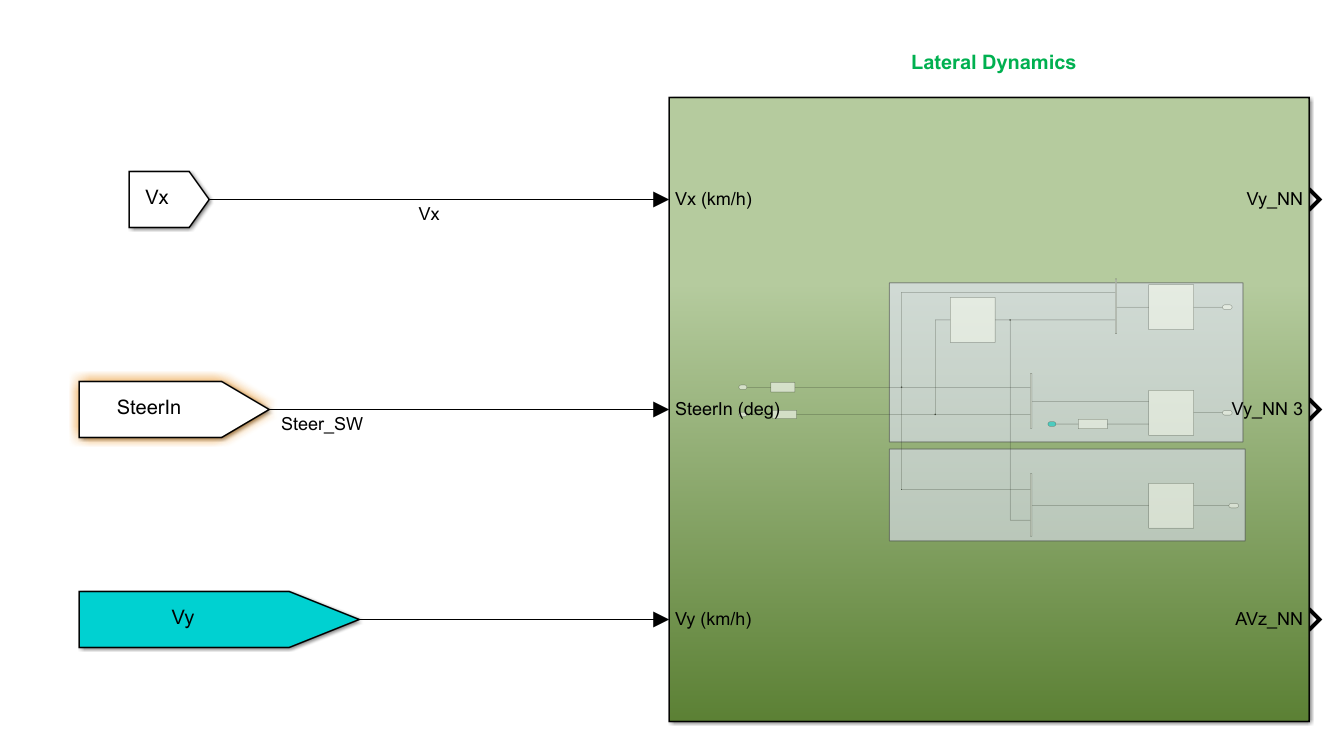
**Bibliography**

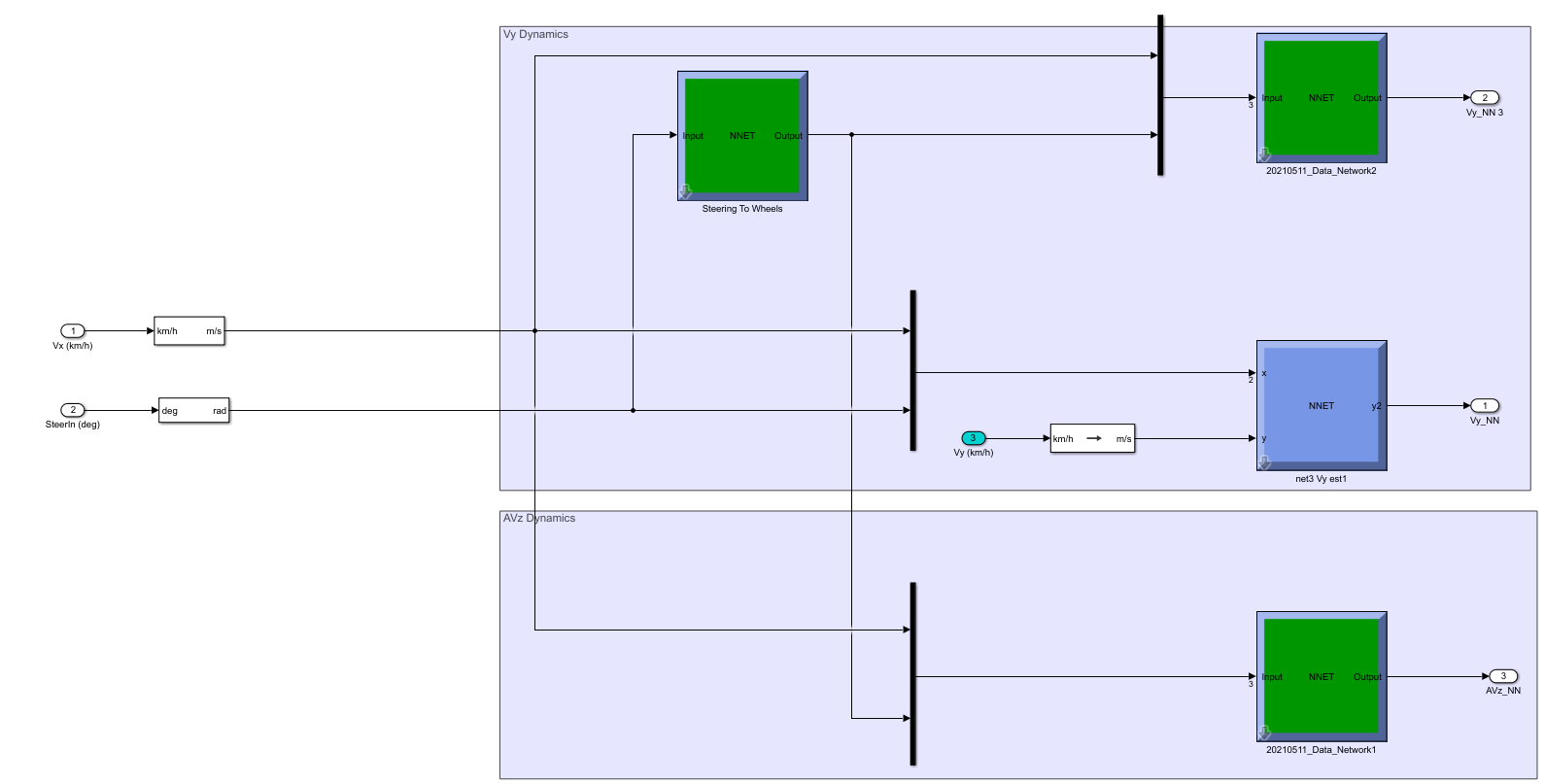
**IEEE STANDARDINA ÇEVRİLECEK**

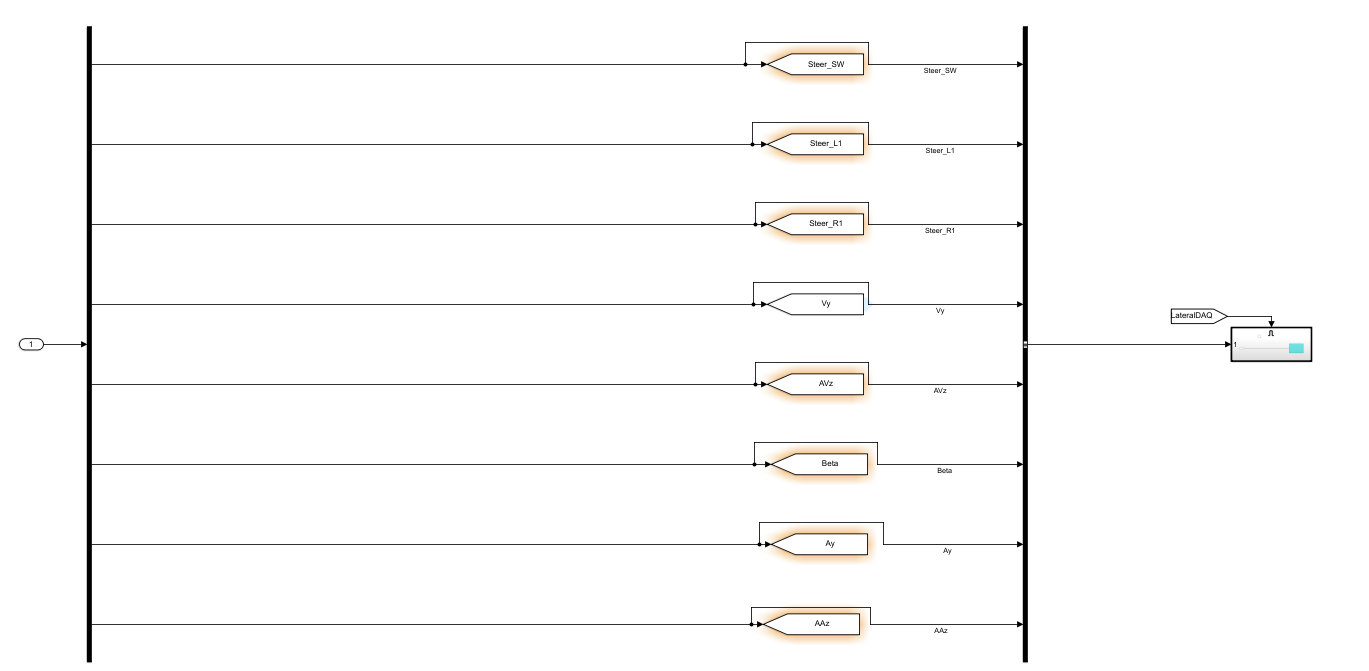
1. Rajamani, R. (2011). *Vehicle dynamics and control*. Springer Science & Business Media.
2. ANTHONY B. WILL & STANISLAW H. Z˚AK (1997) Modelling and Control of an Automated Vehicle, Vehicle System Dynamics, 27:3, 131-155, DOI: 10.1080/00423119708969326
3. Yong, S. Z., Yershov, D., Frazzoli, E., Paden, B., & Cáp, M. (2016). A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles, 8858(c), 1–23. <https://doi.org/10.1109/TIV.2016.2578706>
4. Andersen, H., Chong, Z. J., Eng, Y. H., Pendleton, S., & Ang, M. H. (2016). Geometric path tracking algorithm for autonomous driving in pedestrian environment. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, *2016*-*Septe*, 1669–1674. https://doi.org/10.1109/AIM.2016.7577010
5. Hoffmann, G. M., Tomlin, C. J., Montemerlo, M., & Thrun, S. (2007). Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. *Proceedings of the American Control Conference*, 2296–2301. https://doi.org/10.1109/ACC.2007.4282788

**Others:**

Lateral Dynamics Data:

****

****

****