Implementing Nonlinear Model Predictive Control for Enhanced Trajectory Tracking and Road Anomaly Avoidance in Autonomous Vehicles

Jibril A. Bala
Department of Electrical and Electronics Engineering
Nile University of Nigeria
Abuja, Nigeria
201233001@nileuniversity.edu.ng

Steve A. Adeshina
Department of Computer Engineering
Nile University of Nigeria
Abuja, Nigeria
steve.adeshina@nileuniversity.edu.ng

Abiodun M. Aibinu
Summit University
Offa, Nigeria
maibinu@gmail.com

***Abstract*—In the rapidly evolving realm of autonomous vehicles, trajectory tracking stands as a cornerstone for ensuring safety and precision in navigation. Traditional linear methodologies, although efficient to some extent, can be inadequate in addressing the challenges posed by complex urban environments and defective roads. Nonlinear Model Predictive Control (NMPC), with its capability to anticipate future events and adapt accordingly, presents an opportunity to redefine the trajectory tracking landscape. Therefore, this study presents the integration of NMPC for trajectory tracking in autonomous vehicles.Through a comprehensive analysis and simulations, this study showcases improved tracking accuracy, and enhanced adaptability in diverse scenarios. Our findings underscore the potential of NMPC in setting a new benchmark for trajectory tracking, ultimately pushing the boundaries of what autonomous vehicles can achieve.**

***Index Terms*—Autonomous Vehicles, Nonlinear MPC, Obstacle Avoidance, Road Anomaly,Trajectory Tracking.**

I. Introduction

Autonomous Vehicle (AV) research and development has recently achieved significant progress as a result of developments in computer and sensor technologies [1]. AVs have shown to be reliable navigation tools in places plagued by unauthorised speedbumps, potholes, and bad road conditions, which cause driving stress, misalignment, vehicle damage, and financial strain on road users [2], [3]. Early identification of these road anomalies will allow for successful navigation around them, and improvements in automobiles have led to scientists and researchers' desire and need to attain autonomy in vehicle control [4].

Accurate control in AVs has proved challenging to achieve. This is due to the significant degree of nonlinearity in vehicle dynamics, time delays, uncertainties, disturbances, and dynamic restrictions seen in AVs [1], [5]. To perform well in unstructured conditions, several control techniques, such as fuzzy control and sliding mode control, have been introduced. These solutions, on the other hand, deal with worst-case scenarios, resulting in 'overly cautious' results. Furthermore, because of their reliance on present system conditions, feedback control methods are typically incapable of predicting future events. [5]–[7]. An effective control mechanism should be capable of anticipating future events and of mitigating disturbances [1].

Model Predictive Control (MPC) has become a popular control technique for vehicle route planning, navigation, and path tracking as processing power and computer speed have improved. MPC is a closed-loop control method that can optimise and forecast [8]. MPC combines optimal and adaptive control techniques to provide a control scheme capable of optimising a system's states in order to generate an appropriate control input. MPC is akin to an adaptive control technique in that it may regulate itself in response to changing conditions. At each interval, the control technique governs its inputs and outputs by solving an optimisation problem [9]. The algorithm's capacity to handle Multiple Input and Multiple Output (MIMO) situations, its ability to enforce restrictions, and the guarantee of a viable trajectory through the technique's ability to simulate a system plant across a prediction horizon led to the selection of MPC in this work [10]. These features are unavailable in control schemes such as Internal Model Control, Fuzzy Logic Control, and Proportional-Integral-Derivative (PID) Control.

MPC is a linear control approach by definition. In autonomous cars, however, the longitudinal and lateral vehicle dynamics are nonlinearly connected. This makes it difficult for the typical MPC to create system control inputs. Furthermore, when other vehicle nonlinearities like as friction are considered, the system becomes more nonlinear, making it difficult to estimate with a linear model [11]. To overcome the aforementioned constraint, another approach is to utilise a nonlinear vehicle model, which indicates that the control scheme will be tackling a nonlinear optimisation problem [12]. This is where Nonlinear MPC (NMPC) comes in. NMPC estimates a path and follows it while minimising a cost function and imposing limitations using a nonlinear model of

979-8-3503-5883-4/23/$31.00

vehicle dynamics. Because of its capacity to provide precise trajectories and robust performance, the control scheme has been employed in unmanned vehicles and robotic mechanism control [13].

In this study an NMPC scheme for enhanced trajectory tracking and road anomaly avoidance is presented. The proposed scheme is capable of following a reference trajectory and the design incorporates road anomaly avoidance capabilities. The rest of this paper is divided into four sections. Section two presents a review of related works while the research methodology is presented in Section three. The results and discussion are presented in Section four while Section five concludes and provides future research directions.

II. Literature Review

There exists, in literature, several works in the area of MPC based navigation. The authors in [14] presented an NMPC scheme for path tracking in agricultural machine navigation. The technique utilised a 2D laser scanner, extended Kalman filter, and inertial measurement unit to effectively navigate an agricultural environment using a tractor. The results showed the system could perform as accurately as a human driver, demonstrating a maximum lateral error of 10 cm at a speed of 12 km/h. However, there were instances where the controller failed due to computational complexity, making it necessary for the backup system to take over the control.

Additionally, [15] presented a hierarchical MPC scheme for navigation of multiple robots. The technique decouples constraints in cost functions, thereby, allowing computation to be distributed to the lower control scheme of each robot and run in parallel. The simulation results demonstrated the efficiency of the system in terms of computational complexity. However, nonlinear system dynamics were not considered in this study.

In [16], an NMPC technique was developed for trajectory tracking and control for autonomous racing UAVs. The objective function was developed to model the racing environment and the method was capable of dynamically updating the position goal through internal state estimation and control input generation. The simulation results showed a reduction of approximately one second in flight time for short range manoeuvres of 2 to 4 metres. However, position errors persisted due to model errors and an inability to fully control the UAV thrust.

A motion planning technique for mobile robots was presented in [17]. The technique utilised a modified batch informed trees algorithm and MPC for collision free navigation. The results showed the effectiveness of the technique in navigating an environment with few and multiple obstacles. The scheme, however, the dynamics of the mobile robotic system was not considered in the controller design.

An NMPC technique for navigation in complex terrains by a hopping model was presented in [18]. In this study, continuous optimization approach was employed within the MPC framework to carry out trajectory planning. The results showed an increase in computation speed for the developed model. However, the method was deemed computationally intensive for real time scenarios, especially when considering longer planning horizons.

A guidance and control scheme using reinforcement learning and MPC was presented in [19] for robust embedded quadrotors. The MPC technique provided vehicle control and obstacle avoidance while the reinforcement learning algorithm guided the vehicle through complex environments. The results showed the technique exhibited robust navigation with modest computational demand.

Collision avoidance using hexacopter Unmanned Aerial Vehicles (UAVs) was presented in [13] using system identification and NMPC. In the technique implemented, system dynamics, tracking errors, obstacles, and navigation velocities were considered in the objective function. Simulations and real world experiments were conducted to validate the technique and the results showed the ability of the technique to solve the control problem and its scalability for dynamic settings.

The authors in [12] presented an aerial navigation scheme for based on embedded NMPC. The technique utilises PANOC, which is a fast numerical optimisation method to solve the MPC problem. The results demonstrate that the method is applicable to micro aerial vehicle platforms and can avoid obstacles. However, dynamic obstacle avoidance was not considered in this study.

III. Research Methodology

A. Model Overview

The NMPC scheme developed in this study consists on three major components. The first component is the system model which defines the states, inputs and outputs of the system. In this study, a kinematic vehicle model was implemented as the system model. The second component of the NMPC scheme is the objective (or cost) function. The cost function in this case is based on the motion the vehicle is required to achieve. In this application, the vehicle is desired to avoid collision with road anomalies and maintain its desired trajectory. The objective function reflects all these requirements. The third component of the NMPC scheme is the set of constraints for the model. The enforcing of these constraints help control the vehicle movement in addition to the objective function. Figure 1 shows an overview of the NMPC Architecture.

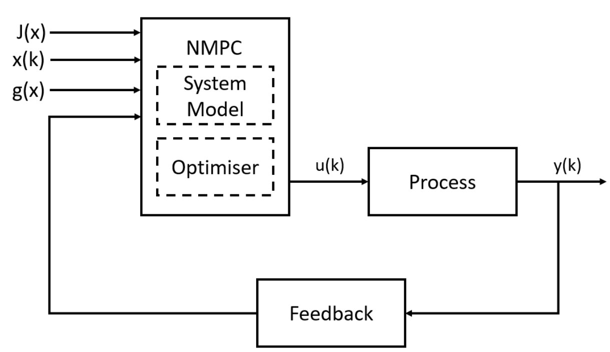

Fig. 1. NMPC Overview

B. Vehicle Model

We consider a vehicle that is modelled in 2D space with the state variables $x(t), y(t)$, and $\theta(t)$ which respectively represent the coordinates of the vehicle's centre of gravity for a fixed frame and the orientation (or heading) of the vehicle with respect to the x-axis of the global frame. These variables constitute the state vector $X(t) \in \mathbf{R}^3$. Figure 2 shows a representation of the kinematic model.

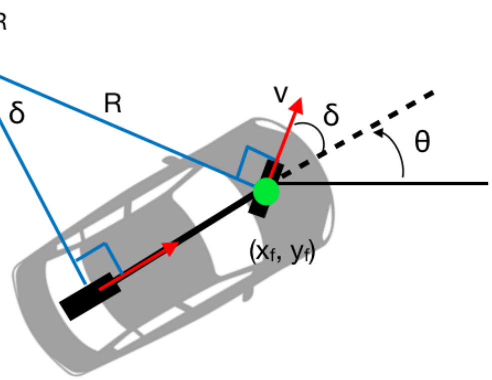

Fig. 2. Kinematic Vehicle Model [20]

$$X(t) = [x(t), y(t), \theta(t)] \tag{1}$$

The control inputs to the system at time, t are $v(t)$ and $\delta(t)$ which represent the longitudinal speed of the vehicle and the steering angle of the vehicle's front wheels respectively. These control inputs constitute the control vector $U(t) \in \mathbf{R}^2$.

$$U(t) = [v(t), \delta(t)]' \tag{2}$$

The state variables are governed by nonlinear differential equations which comprise of the vehicle model as shown in Equations 3, 4, and 5.

$$\frac{dx}{dt} = v(t)cos(\theta(t)) \tag{3}$$

$$\frac{dy}{dt} = v(t)sin(\theta(t)) \tag{4}$$

$$\frac{d\theta}{dt} = \frac{v(t)}{L}tan(\delta(t)) \tag{5}$$

In Equation 5, the variable L denotes the wheelbase of the vehicle, i.e., the distance between the vehicle's front and rear axles.

These equations are represented in the compact state-space form as $\frac{dx}{dt} = f(X(t), U(t)$ with $f : \mathbf{R}^3 \times \mathbf{R}^2 \rightarrow \mathbf{R}^3$ representing the system dynamics and defined as shown in Equation 6.

$$f(X(t), U(t) = [v(t)cos(\theta(t)), v(t)sin(\theta(t)), \frac{v(t)}{L}tan(\delta(t))]' \tag{6}$$

C. Objective Function Formulation

The model predictive control strategy solves a finite-time constrained optimal control problem with a receding horizon. Let N be the prediction horizon, path planning may be expressed as a nonlinear optimisation problem, as illustrated in Equation 7.

$$\text{Minimize } J = \lambda \sum_{k=1}^{N} \Big(W_1 \cdot d_{x_k}^2 + W_2 \cdot d_{y_k}^2 + W_3 \cdot (v_{\text{desired}} - v_k)^2 + W_4 \cdot (\theta_N - \theta_{\text{desired}})^2 \Big) \tag{7}$$

$s.t.$

$$z_{k+1} = z_k + f(z_k, u_k)T \tag{8}$$

$$z_{k+1} \in \mathcal{Z}, \quad u_k \in \mathcal{U}, \quad \lambda \in \Lambda_{M \times N} \tag{9}$$

$$g(z_{k+1}) \leq 0 \tag{10}$$

$$k = 0, ..., N-1$$

From Equation 7, λ represents a scaling factor that can be adjusted to influence the trade-off between different terms in the objective function. d_{x_k} and d_{y_k} respectively represent the longitudinal and lateral distance to the vehicle's destination. $(v_{\text{desired}} - v_k)$ represents the difference between the desired speed and the vehicle speed at time step, k. This component of the objective function ensures the vehicle maintains a specified velocity. $(\theta_N - \theta_{\text{desired}})$ represents the difference between the desired heading angle and the vehicle's heading at time step, N. This component ensures that the vehicle's orientation will align with the desired orientation.

The Vehicle Dynamics (System Model) constraint is defined in Equation 8. This determines how the state of the vehicle z_k changes over time as a result of control input u_k. The function $f(z_k, u_k)$ represents the vehicle's dynamics, and T is the time step. The State and Control Constraints are specified in Equation 9. This means that the states z_{k+1}, control inputs u_k, and design variable must all lie within feasible sets. These restrictions can reflect physical vehicle limitations, actuator limits, lane decisions, and so on. The prediction horizon for the Model Predictive Control (MPC) problem is defined by the constraint $k = 0, ..., N-1$.

The Collision Avoidance restriction is specified in Equation 10. In order to achieve the collision constraint for real time implementation, a collision penalty function is incorporated into the NMPC formulation. The aim is to add a possible collision cost to the cost function for each time step in the horizon of the prediction. When a collision is conceivable, this collision cost is set to be very high, thereby preventing the controller from selecting control inputs that may result in a collision. This can be represented mathematically as follows:

The original cost function is given by:

$$J = \sum_{k=1}^{N} C_k \tag{11}$$

and is modified to include a collision cost term:

$$J = \sum_{k=1}^{N} (C_k + P_k) \tag{12}$$

where:

- C_k is the original cost at time step k, which includes terms related to tracking error, control effort, etc.
- P_k is the collision cost at time step k. This is a function of the vehicle state X_k and a set of obstacles. If a collision is detected, P_k is a large value; otherwise, $P_k = 0$.

A collision is detected when the coordinates of the obstacle intersects with the vehicle coordinates at the current time step.

IV. Results

The primary objective of the implemented nonlinear model predictive control (NMPC) strategy was to track a predefined reference trajectory for a kinematic vehicle model. The results, as depicted subsequently, reflect the control strategy's efficacy in maintaining the vehicle's position and orientation as close as possible to the desired reference over a simulation duration. Figure 3 presents the trajectory generated by the x-state of the vehicle model.

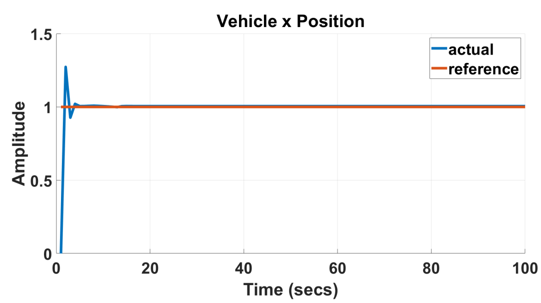

Fig. 3. Trajectory Generated by x-state of Vehicle Model

The X position, representing the vehicle's longitudinal displacement, exhibited a trajectory that closely followed the provided reference. Figure 3 portrays the NMPC-guided trajectory juxtaposed against the reference. Initial time steps witnessed minor discrepancies, indicative of the system's transient phase. However, as the simulation advanced, the vehicle's position demonstrated a convergence towards the set reference, thereby affirming the NMPC's competence in mitigating positional deviations. Additionally, Figure 4 presents the trajectory generated by the y-state of the vehicle model.

Y position, delineating the vehicle's lateral motion, was another crucial component assessed during the simulation. As illustrated in Figure 4, the Y position's trajectory closely adhered to the reference with only minor oscillations. This consistency highlights the precision and stability of the employed control strategy, even in the face of inherent model complexities. The transient deviations observed were effectively compensated for in subsequent time frames. Furthermore, Additionally, Figure 5 presents the trajectory generated by the theta-state of the vehicle model.

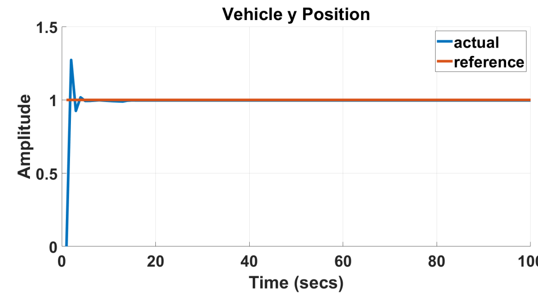

Fig. 4. Trajectory Generated by y-state of Vehicle Model

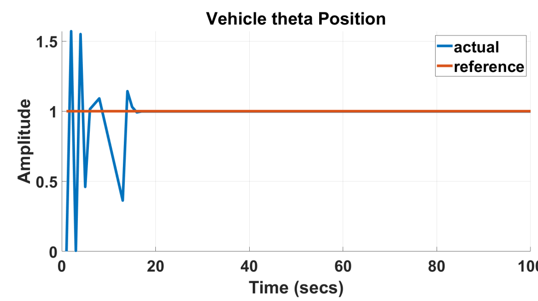

Fig. 5. Trajectory Generated by theta-state of Vehicle Model

Theta, representing the vehicle's orientation or heading, was of paramount importance, especially considering the intricate dynamics associated with vehicular movements. Figure 5 portrays the evolution of Theta over the simulation duration. The trajectory demonstrated the NMPC's adeptness in ensuring the vehicle's alignment with the desired orientation, with negligible discrepancies observed throughout.

An overarching observation from the state trajectories is the NMPC's proficiency in ensuring a coherent tracking of the reference trajectories. The minor deviations observed during initial phases can be attributed to initialization phenomena, which were efficiently rectified as the controller gained feedback from the system.

The measure of any control strategy's effectiveness lies in the tracking errors. These errors, essentially the difference between desired states and actual states, shed light on the control scheme's precision and its ability to mitigate unforeseen deviations. The errors were evaluated in terms of the RMSE and MAE values. A summary of this is presented in Figure 6.

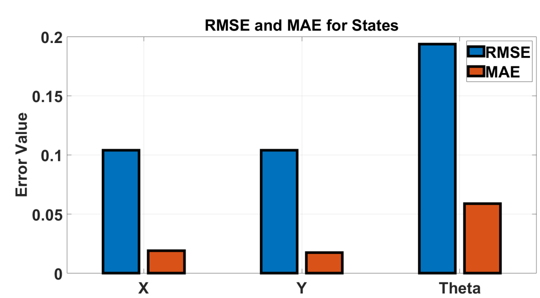

Fig. 6. RMSE and MAE of State Tracking Errors

The computed RMSE and MAE values provide an insightful assessment of the NMPC's state tracking performance. For the x state, an RMSE of 0.1041 juxtaposed against an MAE of 0.0190 suggests that while the overall deviation is moderately small, there are occasional significant deviations, given the relatively larger RMSE compared to MAE. The y state, with an RMSE of 0.1040 and an MAE of 0.0174, exhibits a similar trend as the x state. The close values between x and y states' errors imply a consistent performance of the NMPC in the lateral and longitudinal domains, suggesting that any external factors affecting the system will likely influence these states in a similar manner. However, in the case of the orientation (θ), the results are dissimilar. An RMSE of 0.1938 and an MAE of 0.0587 indicate a more pronounced disparity between the typical error (as suggested by the MAE) and the occasional larger errors (as captured by the RMSE). This might be an indication of the intricacies associated with orientation control, where sporadic large deviations might have occurred, potentially due to transient dynamics or initialization phenomena.

V. Conclusion

An NMPC scheme was developed in this study. This scheme uses the outputs of the road anomaly detection model and the RA-SLAM technique to carry out effective control action and navigation. The NMPC method was developed using a kinematic vehicle model and a custom objective function. The objective function catered for various aspects of the navigation such as collision avoidance, lane keeping, velocity control, and steering control. The technique was evaluated based on the trajectory following, control efforts, tracking errors, and optimisation cost minimisation. The technique's findings establish the usefulness of the developed method in traversing outside road sceneries riddled with anomalies. Future studies will focus on integrating this model with a road anomaly detection scheme to evaluate its performance in real world settings.

Acknowledgment

The authors wish to thank the National Information Technology Development Agency who funded this research under the 2020 NITDEF Scholarship Scheme

References

[1] L. Tang, F. Yan, B. Zou, K. Wang, and C. Lv, "An improved kinematic model predictive control for high-speed path tracking of autonomous vehicles," *IEEE Access*, vol. 8, pp. 51 400–51 413, 2020.

[2] G. Enwerem and G. Ali, "Economic effects of bad roads on vehicle maintenance in nigeria," *International Journal of Scientific and Research Publications*, vol. 6, no. 6, pp. 761–766, 2016.

[3] A. Moses and A. Fortunatus, "The use of gis in the spatial distribution of speed bumps within afikpo," *J. Geogr. Environ. Earth Sci.*, vol. 12, no. 1, pp. 1–7, 2017.

[4] A. Rasouli and J. K. Tsotsos, "Autonomous vehicles that interact with pedestrians: A survey of theory and practice," *IEEE transactions on intelligent transportation systems*, vol. 21, no. 3, pp. 900–918, 2019.

[5] J. Cao, C. Song, S. Peng, S. Song, X. Zhang, and F. Xiao, "Trajectory tracking control algorithm for autonomous vehicle considering cornering characteristics," *IEEE Access*, vol. 8, pp. 59 470–59 484, 2020.

[6] G. V. Lakhekar, L. M. Waghmare, P. G. Jadhav, and R. G. Roy, "Robust diving motion control of an autonomous underwater vehicle using adaptive neuro-fuzzy sliding mode technique," *IEEE Access*, vol. 8, pp. 109 891–109 904, 2020.

[7] S. An, K. Liu, Y. Fan, J. Guo, and Z. She, "Control design for the autonomous horizontal takeoff phase of the reusable launch vehicles," *IEEE Access*, vol. 8, pp. 109 015–109 027, 2020.

[8] Y. Mu, J. Qiao, J. Liu, D. An, and Y. Wei, "Path planning with multiple constraints and path following based on model predictive control for robotic fish," *Information Processing in Agriculture*, vol. 9, no. 1, pp. 91–99, 2022.

[9] Z. Elmi and S. Elmi, "Autonomous vehicle path planning using mpc and apf," *Motion Planning*, p. 81, 2022.

[10] D. Saccani and L. Fagiano, "Autonomous UAV Navigation in an Unknown Environment via Multi-Trajectory Model Predictive Control," *2021 European Control Conference (ECC)*, pp. 1577–1582, 2021.

[11] M. Brown and J. C. Gerdes, "Coordinating Tire Forces to Avoid Obstacles Using Nonlinear Model Predictive Control," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 1, pp. 21–31, 2020.

[12] E. Small, P. Sopasakis, E. Fresk, P. Patrinos, and G. Nikolakopoulos, "Aerial navigation in obstructed environments with embedded nonlinear model predictive control," *2019 18th European Control Conference (ECC)*, pp. 3556–3563, 2019.

[13] L. F. Recalde, B. S. Guevara, C. P. Carvajal, V. H. Andaluz, J. Varela-aldás, and D. C. Gandolfo, "System Identification and Nonlinear Model Predictive Control with Collision Avoidance Applied in Hexacopters UAVs," *Sensors*, vol. 22, no. 4712, pp. 1–29, 2022.

[14] J. Backman, T. Oksanen, and A. Visala, "Navigation system for agricultural machines : Nonlinear Model Predictive path tracking," *Computers and Electronics in Agriculture*, vol. 82, pp. 32–43, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.compag.2011.12.009

[15] C. Huang, X. Chen, Y. Zhang, S. Qin, Y. Zeng, and X. Li, "Hierarchical model predictive control for multi-robot navigation," 2016.

[16] S. T. Spronk and S. Li, "A nonlinear model predictive control approach to autonomous uav racing trajectory generation control," 2020.

[17] P. Xu, N. Wang, S.-l. Dai, and L. Zuo, "Motion Planning for Mobile Robot with Modified BIT * and MPC," *Applied Sciences*, vol. 11, no. 426, pp. 1–15, 2021.

[18] A. Zamani and P. A. Bhounsule, "Nonlinear model predictive control of hopping model using approximate step-to-step models for navigation on complex terrain," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) October 25-29, 2020, Las Vegas, NV, USA (Virtual) Nonlinear*, pp. 3627–3632, 2020.

[19] C. Greatwood and A. G. Richards, "Reinforcement learning and model predictive control for robust embedded quadrotor guidance and control," *Autonomous Robots*, vol. 43, no. 7, pp. 1681–1693, 2019. [Online]. Available: https://doi.org/10.1007/s10514-019-09829-4

[20] Y. Ding. (2021) Simple understanding of kinematic bicycle model. [Online]. Available: https://www.shuffleai.blog/blog/Simple_Understanding_of_Kinematic_Bicycle_Model.html