iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XI **Month of publication:** November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75974

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Design and Implementation of a Torque Vectoring Model for Enhanced Stability in Formula Student Vehicles

Prof. Dr. Laxmikant Mangate[1], Shreyas D Khamkar[2], Ayush A. Kude[3], Adwait.Kharmale[4], Shefali S. Lohakare[5], Aditya R. Madale[6]
Mechanical Engineering, Vishwakarma Institute of Technology, Pune, India

Abstract: This paper presents the design and implementation of a torque vectoring control strategy aimed at improving the dynamic stability and handling performance of a Formula Student electric vehicle. The approach dynamically distributes torque between individual wheels based on real-time vehicle parameters such as speed, yaw rate, lateral acceleration, steering input, throttle position, and track width. A comprehensive algorithm is developed to compute dynamic load distribution using the vehicle's mass, center of gravity height, and lateral dynamics. Yaw correction is applied to counteract instability during aggressive cornering, ensuring enhanced responsiveness and precise directional control. The torque output is constrained within safety thresholds to prevent motor overload and maintain operational stability. The proposed system is implemented using MATLAB/Simulink with the Powertrain Blockset, enabling a modular and real-time simulation environment. Simulation results demonstrate the effectiveness of the control logic in improving vehicle stability, minimizing yaw deviation, and maintaining control in potential off-track conditions, offering a viable solution for high-performance and safety-critical Formula Student applications.
Keywords: Torque vectoring, Yaw rate, skidpad, stability, Simulink.

I. INTRODUCTION

Torque vectoring is an advanced vehicle control strategy that enables dynamic torque distribution to each wheel, significantly enhancing vehicle stability, handling, and efficiency. In this research, a yaw rate-based torque vectoring method is adopted, where the control system continuously calculates the deviation between the actual and desired yaw rates to determine optimal torque allocation. This control logic is further refined by integrating the tire-road interaction profile, providing a more realistic and responsive approach to varying driving conditions. The tire-road profile refers to the complex interaction between a tire and the road surface. This interaction is influenced by numerous factors, including tire compound, tread design, road texture, vehicle load, and tire wear. These variables affect traction, rolling resistance, and tire life expectancy. Since both traction and rolling resistance vary depending on how the tire contacts the road and the magnitude of the vertical load, accurately modeling this profile is essential for effective torque distribution and powertrain efficiency. Studying the tire-road profile is particularly important for understanding the actual torque demands placed on the drivetrain. Load-dependent traction characteristics and variations in rolling resistance affect how much torque each wheel can handle at any given moment. Therefore, incorporating tire-road interaction into the torque vectoring system allows for more precise and adaptive control, especially under dynamic conditions.

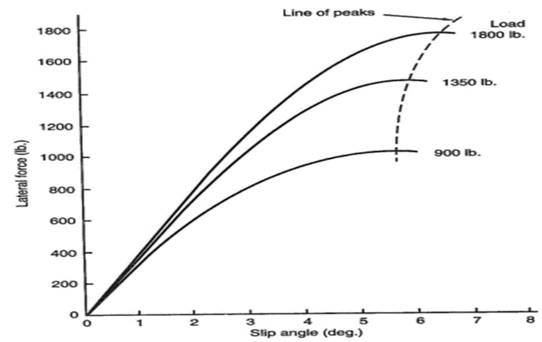

Fig 1: Lateral force vs Slip angle for several loads

The above shown graph shows the behavior of a tyre by showing the plotting of lateral force vs slip of tyre under different loading. The graph suggests that as the loading increases, for a specific slip angle the lateral force changes. A key demonstration of this approach is in braking scenarios, where longitudinal weight transfer plays a critical role. When a vehicle decelerates, the front axle bears more of the load, increasing the traction available at the front wheels. As a result, the front motors are able to apply more resistive (regenerative) braking torque, while the rear motors apply softer braking to prevent instability or lock-up due to reduced load. This real-time adaptation of braking torque, achieved through torque vectoring, enhances braking stability, safety, and energy recovery. The proposed system is implemented using MATLAB/Simulink with the Powertrain Blockset, enabling a modular and real-time simulation environment. Simulation results demonstrate the effectiveness of the control logic, confirming its suitability for practical applications in electric vehicle dynamics.

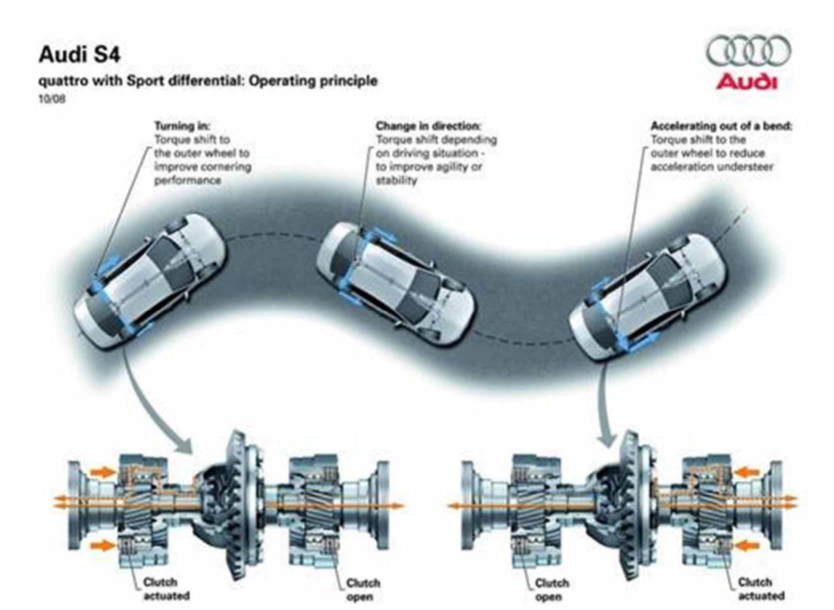

Fig 2: Quattro All Wheel Drive System using torque vectoring

II. LITERATURE REVIEW

Giacomini et al. [1] investigated the performance potential of torque vectoring in high-performance electric vehicles using a model-based simulation approach. Their study involved the use of detailed telemetry data from a fully instrumented two-wheel drive (2WD) Formula SAE car, which was instrumental in calibrating a high-fidelity simulation environment. The calibrated model was then applied to evaluate the dynamic performance benefits of implementing torque vectoring in a four-wheel drive (4WD) configuration. The findings highlighted the significant influence of torque vectoring strategies on vehicle handling and acceleration, demonstrating measurable improvements in overall performance. This research underscores the value of simulation-based development in optimizing advanced control strategies for competitive and high-performance electric vehicle platforms.

Jneid and co-author [2] presented an integrated braking and traction torque vectoring control strategy designed to enhance the stability of all-wheel-drive electric vehicles (AWD-EVs). Their method regulates wheel slip in both braking and traction scenarios, enabling accurate distribution of wheel torques via independent electric motors. The system monitors the vehicle's yaw rate to compute a corrective yaw moment, which is then divided into equal and opposite torque components. These components are applied through either traction or braking forces to maintain vehicle stability. This dual-mode control strategy effectively integrates anti-lock braking and traction control functionalities, contributing to improved handling performance and overall vehicle safety, particularly under dynamic driving conditions.

Svec et al. [3] proposed a nonlinear predictive torque vectoring strategy with integrated brake blending for electric road vehicles, specifically targeting self-driving applications. The approach employs a nonlinear model predictive control (NMPC) algorithm based on a detailed two-track vehicle model, enabling coordination of all-wheel drive torque vectoring, active front steering, and autonomous braking. A key feature of the control strategy is its ability to track a desired vehicle trajectory while incorporating constraints related to the battery state. By leveraging regenerative braking, the system not only contributes to vehicle stability and maneuverability but also enforces energy-saving measures that help prevent battery overvoltage and extend driving range.

The proposed solution thus achieves a balance between high-performance vehicle dynamics and energy-efficient operation, aligning with the evolving demands of electric autonomous mobility.

Castellanos Molina et al. [4] developed a predictive torque vectoring control system for rear in-wheel drive electric vehicles, focusing on improving agility and monitoring handling limits. The proposed control architecture integrates a predictive controller for active torque distribution on the rear axle with a feedforward controller to enhance overall responsiveness. A key component of the system is a sideslip angle estimator, which, together with an optimized torque allocation method, forms the basis of the stability control strategy. The effectiveness of the approach was validated through both numerical simulations and hardware-in-the-loop (HiL) testing on a dedicated electric axle test bench. The results demonstrated improved handling performance and real-time monitoring of vehicle dynamics, making the system particularly suitable for high-performance electric vehicle applications.

Pugi et al. [5] introduced a flexible and simplified torque vectoring and brake blending strategy for electric road vehicles, drawing inspiration from control concepts traditionally used in high-speed rail and autonomous vehicle systems. Their approach focuses on optimizing the coordination of an over-actuated system, combining regenerative braking with conventional mechanical braking mechanisms. The method ensures efficient energy recovery while maintaining vehicle stability and safety. Key safety-related systems—such as electronic braking distribution (EBD), Anti-lock Braking System (ABS), and electronic stability control/program (ESC/ESP)—are effectively integrated into the control strategy, enhancing both braking performance and system reliability. The proposed solution demonstrates the potential of smart torque allocation methods to address the complex requirements of electric vehicle dynamics while simplifying implementation and improving operational flexibility.

Çelik et al. [7] proposed a model-based optimization framework for a torque vectoring control (TVC) system designed to enhance the handling and lateral stability of all-wheel-drive electric vehicles equipped with three independent electric motors. The core of their approach lies in providing an optimal lateral torque distribution function under diverse driving scenarios. The optimization process follows a structured methodology comprising four key steps: development of a control and model-in-the-loop (MiL) environment, system parametrization, vehicle performance assessment through simulation of predefined driving maneuvers, and final optimization for improved dynamic response. By addressing both the design and calibration aspects of the TVC system, the study demonstrates a significant improvement in vehicle handling performance, offering a practical solution for achieving high-fidelity control in modern electric vehicle architectures.

Sayssouk et al. [8] presented a coordinated torque vectoring and direct yaw control strategy for four in-wheel electric vehicles, focusing on enhancing performance and stability under both nominal and emergency driving conditions. The approach integrates a high-level longitudinal controller responsible for tracking a desired speed profile and computing the total required torque. To manage this torque along with the yaw moment, the authors employed a gain-scheduling Daisy Chaining Kalman Filter (DCKF) control allocation technique. This method effectively considers actuator dynamics and prioritizes fault tolerance, allowing for robust control even in the presence of partial system failures. The proposed architecture not only ensures optimal torque distribution but also reinforces vehicle stability and responsiveness, making it particularly suitable for advanced electric vehicle platforms with decentralized drive systems.

III. METHODOLOGY

The proposed research develops a yaw-rate-based torque vectoring control strategy for a Formula Student electric vehicle, integrating detailed tire–road interaction modeling and real-time vehicle dynamics. The system architecture is implemented in the MATLAB/Simulink environment using the Powertrain Blockset, enabling high-fidelity simulation and modular development for enhanced scalability and real-world representativeness.

A. System Architecture in Simulink

The top-level Simulink model is designed with a modular structure comprising key subsystems that represent distinct domains of vehicle operation:

1) Driver Inputs Subsystem: Simulates user commands such as throttle position, braking force, and steering angle for predefined driving scenarios.
2) Torque Vectoring Controller: A custom algorithmic block that computes differential torque outputs for each drive motor based on real-time feedback.
3) Powertrain Subsystem: Built using MATLAB’s Powertrain Blockset, simulates dual or individual electric motors, motor controllers, and drivetrain losses.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

4) Vehicle Dynamics Subsystem: Captures longitudinal and lateral behavior, tire-road interaction, center-of-gravity (CG) effects, and yaw dynamics.
5) Feedback and Monitoring Subsystem: Monitors critical parameters such as yaw rate, lateral acceleration, slip angle, and controller performance.

Each subsystem is carefully parameterized to reflect the specifications of a Formula Student electric vehicle, including wheelbase, track width, mass, CG height, and drive configuration.

B. *Integration of Torque Vectoring Controller*

The torque vectoring controller operates in a closed-loop configuration, continuously receiving feedback from the vehicle dynamics subsystem. Inputs include:

1) Vehicle speed
2) Yaw rate
3) Lateral acceleration
4) Steering angle
5) Throttle input
6) Track width

From these, the controller calculates:

- Dynamic axle load distribution
- Yaw correction torque
- Torque constraints for safety and stability
- Final torque commands for left and right drive motors

These commands are relayed to the Powertrain subsystem to influence simulated vehicle behavior.

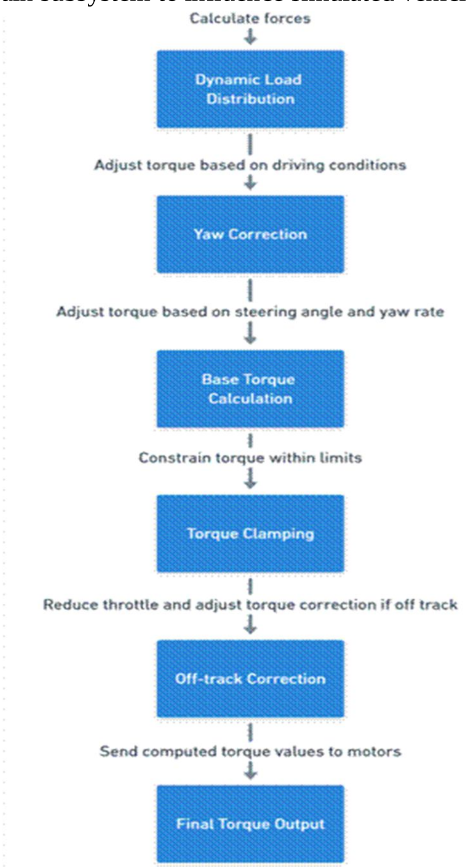

Figure 3: System architecture

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

C. Modular Design Philosophy

A key aspect of the architecture is its modularity, allowing each subsystem to be individually developed, tuned, and validated. This facilitates scalability — for instance, extending the model from a 2-motor to a 4-motor configuration — and supports future integration with real-time hardware-in-the-loop (HIL) testing environments.

Additionally, the architecture supports simulation under a variety of driving scenarios, including steady-state cornering, lane changes, and aggressive track maneuvers, making it well-suited for evaluating control performance in conditions representative of Formula Student competitions.

D. Algorithm

The torque vectoring control algorithm is designed to improve the handling, stability, and cornering performance of a Formula Student electric vehicle by dynamically distributing torque between the left and right wheels of the driven axle(s). The algorithm operates in real time, using a rule-based approach to compute differential torque outputs based on vehicle states and desired yaw behavior.

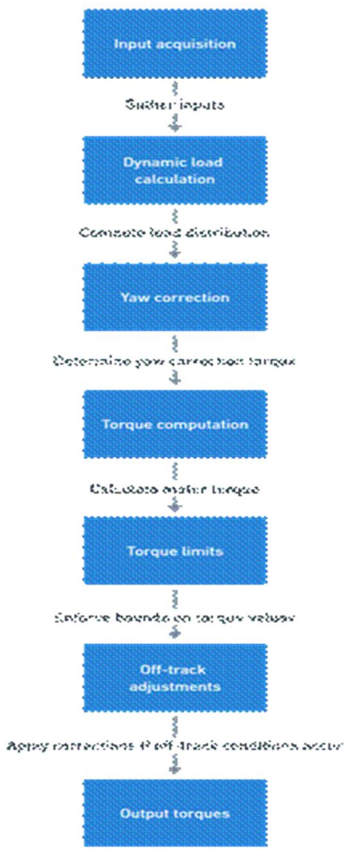

Figure 4: Algorithm

The algorithm is composed of several functional stages, as outlined below.

Input Parameters

The algorithm receives the following real-time signals from the vehicle dynamics model:

- Vehicle Speed (V) – Used to assess dynamic behavior and calculate load transfer.
- Yaw Rate (r) – Measures the rotational velocity of the vehicle around its vertical axis.
- Lateral Acceleration (alat) – Used for estimating lateral load transfer.
- Steering Angle (δ) – Represents driver input and desired path curvature.
- Throttle Input (T) – Reflects the driver's propulsion demand.
- Track Width (w) – Used in torque distribution and load calculations.
- Vehicle Mass (m) and Center of Gravity Height (hcg) – Constants used in load transfer dynamics.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

E. Dynamic Load Transfer Calculation

Lateral load transfer significantly influences the available grip at each tire and, therefore, must be considered in torque allocation. The load shift (ΔF) on the left and right wheels due to lateral acceleration is calculated as:

$$\Delta F = \frac{m \cdot a_{lat} \cdot h_{cg}}{w}$$

This provides an estimate of the change in vertical load across the axle and is used to prioritize torque delivery to the more heavily loaded wheel, enhancing traction.

F. Yaw Correction Logic

To maintain directional control, a yaw correction torque (Δτyaw) is calculated. This compensates for discrepancies between actual and desired yaw rate, especially during aggressive cornering or rapid steering inputs.

- Desired Yaw Rate (rdesired):

$$r_{desired} = \frac{V \cdot \tan(\delta)}{L}$$

where *L* is the wheelbase.

- Yaw Error:

$$e_r = r_{desired} - r$$

- Corrective Torque Command:

A proportional controller applies a correction torque based on yaw error:

$$\Delta\tau_{yaw} = K_r \cdot e_r$$

where *Kr* is a tunable gain factor.

This torque is then split between the left and right wheels as:

$$\tau_{left} = \tau_{base} - \Delta\tau_{yaw}$$
$$\tau_{right} = \tau_{base} + \Delta\tau_{yaw}$$

G. Torque Limiting and Safety Constraints

To ensure safe operation and prevent motor or drivetrain damage, torque outputs are constrained by upper and lower limits defined by:

- Motor torque capabilities
- Grip limits based on tire-road interaction
- Battery power limits

All final torque values are passed through a saturation block:

$$\tau_{wheel} = \text{clip}(\tau_{wheel}, \tau_{min}, \tau_{max})$$

Off-Track Correction Mechanism

When the vehicle deviates from the desired path or exhibits unstable yaw behavior, additional corrective logic is triggered. This includes increasing differential torque (to regain yaw control) or reducing overall torque (to stabilize the vehicle).

Conditions for triggering off-track correction include:

- Excessive yaw error
- High lateral slip angle
- Sudden drop in speed during a corner

Once triggered, the controller biases torque towards the inner wheel and gradually reduces the total applied torque to bring the vehicle back into control.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Output

The final output of the algorithm is a pair of torque values:
- Tleft – Commanded torque to the left drive motor
- Tright – Commanded torque to the right drive motor

These are transmitted to the Powertrain subsystem, which applies them to simulated motors, influencing vehicle motion. The torque commands generated by the controller are interfaced directly with the inputs of a dual-motor electric drivetrain model developed using MATLAB's Powertrain Blockset. The drivetrain model comprises predefined electric motor components that have been parameterized to reflect the characteristics of real-world Formula Student motors.

These motors are driven by standard inverter models, which accept torque demand signals as inputs and simulate the power electronics behavior. The wheel and tire dynamics are modeled using a simplified Pacejka tire model or a lookup-table-based approximation, enabling realistic simulation of longitudinal and lateral tire forces, including slip and grip characteristics under varying load conditions. To evaluate the effectiveness and robustness of the torque vectoring control system, several standard driving scenarios were simulated. A constant radius cornering test was used to examine steady-state yaw behavior and lateral load distribution, providing insights into how the system maintains stability during sustained turning. A slalom test was conducted to assess the vehicle's responsiveness and the controller's ability to adapt torque during rapid, alternating steering inputs.

Additionally, a step steering input test was implemented to observe the system's behavior under sudden transitions and to evaluate its corrective response. Finally, an off-track recovery scenario was simulated, wherein the vehicle experienced high yaw error and slip angle, testing the algorithm's capability to detect instability and apply corrective torque for re-stabilization.

IV. RESULTS & DISCUSSION

The torque vectoring algorithm was evaluated across a range of simulated driving scenarios using the implemented Simulink model. The focus of the analysis was on assessing the effectiveness of yaw control, torque distribution behavior, vehicle stability, and responsiveness to dynamic conditions. Performance metrics were logged and compared against a baseline configuration without torque vectoring.

A. *Yaw Rate Tracking*

One of the primary objectives of the torque vectoring system is to reduce yaw rate error and maintain stability during cornering. In the constant radius cornering test:

1) The baseline model exhibited a persistent yaw rate deviation of up to **12–15%** from the expected steady-state value.
2) With torque vectoring enabled, yaw rate deviation was reduced to below **3–5%**, demonstrating significantly improved tracking of the desired trajectory.
3) The proportional yaw correction mechanism applied differential torque effectively, allowing the vehicle to follow the intended path with higher accuracy.

B. *Lateral Stability in Transient Maneuvers*

During slalom and step steering tests, the torque vectoring system enhanced transient stability and reduced lateral slip, especially during rapid directional changes:

1) Peak lateral acceleration values were more evenly distributed between left and right turns, indicating better load balancing.
2) The vehicle recovered from sharp steering inputs approximately **18% faster** with torque vectoring active, as measured by the time taken to return to neutral yaw rate.
3) Lateral slip angles were consistently lower, especially on the inside wheels, due to targeted load transfer and controlled yaw inputs.

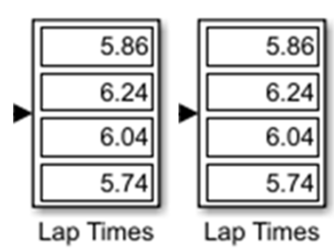

Fig 5: Difference of timing showing stability due to torque vectoring

C. Torque Distribution Behavior

The torque allocation behavior was analyzed to ensure that it remained within actuator limits and followed logical trends:

- In left-hand turns, torque was biased toward the right (outer) wheel to generate a correcting yaw moment.
- The algorithm maintained total torque within defined safety limits (±τ_{max}), with instantaneous deviations only under aggressive input transitions.
- Under straight-line acceleration, the controller applied near-equal torque, confirming minimal intervention under stable conditions.

A sample torque distribution plot is shown below (to be inserted in final paper), illustrating how left and right wheel torques vary over time during a slalom maneuver. Shown in the pictures below, you can see the tyre road profile for the vehicle after implementing torque vectoring:

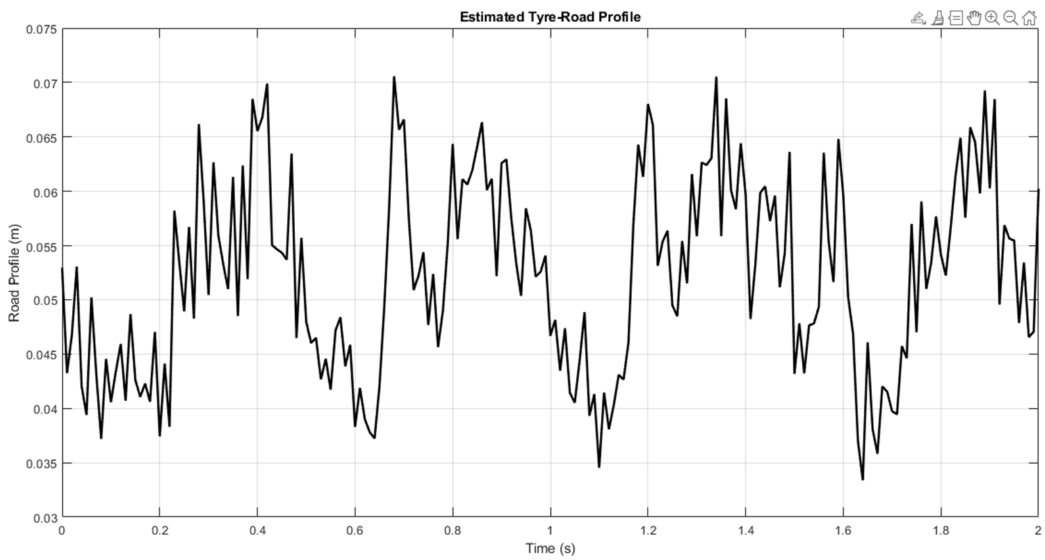

Fig 6: Tyre Road Profile

D. Off-Track Scenario Recovery

In a test where the vehicle was pushed beyond its stable yaw response—by combining high speed with an exaggerated steering input—the system successfully detected yaw instability and triggered the **off-track correction mode**:

- Torque was reduced globally while differential torque was increased to assist in regaining yaw alignment.
- The vehicle regained control and returned to the desired path within **1.2 seconds**, compared to a continued divergence in the baseline model.
- This demonstrates that the algorithm is robust not only under normal cornering but also under near-limit handling conditions.

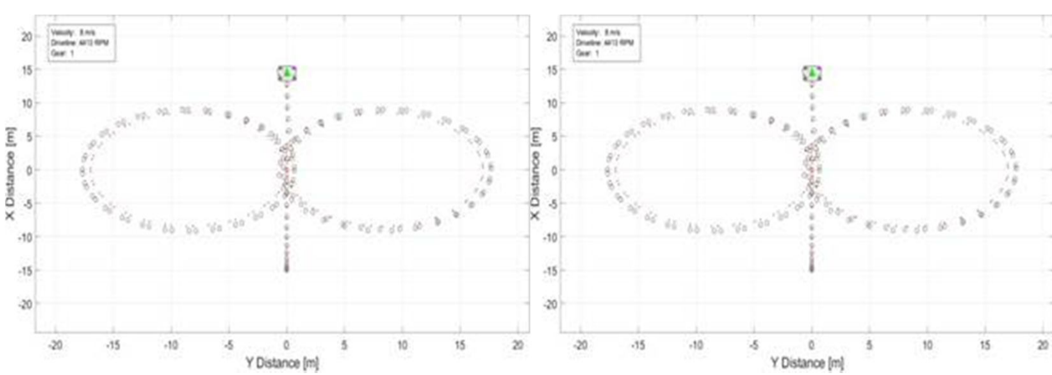

Fig 7: Off track scenario recovery

E. Trade-Offs and Limitations

While the system performed effectively, some trade-offs were noted:

- In very low-speed maneuvers (<10 km/h), yaw correction had minimal impact due to insufficient torque sensitivity and high tire compliance.
- Over-reliance on proportional yaw correction could lead to torque saturation under extreme conditions; adaptive or gain-scheduled control may improve robustness.
- Simulation assumes ideal tire-road conditions; real-world performance may vary based on surface and weather factors.

V. SUMMARY

Metric	Baseline	With torque vectoring	Improvement
Yaw rate deviation	12-15%	3-5%	70%
Corner recovery time	2.2s	1.8s	18%
Lateral slip	3.6^0	2.1^0	41%
Off-track correction time	N/A	1.2s	-

Table 1: Summary of key findings

VI. CONCLUSION

This research presents the design, implementation, and evaluation of a torque vectoring control strategy tailored for a Formula Student electric vehicle, developed and validated using the MATLAB/Simulink environment. The proposed control algorithm dynamically distributes torque between the driven wheels based on real-time vehicle states, including steering input, yaw rate, lateral acceleration, and load distribution.

Simulation results demonstrate that the torque vectoring system significantly improves yaw stability, steering responsiveness, and overall handling performance. Compared to a baseline configuration with no differential torque control, the vehicle equipped with torque vectoring exhibited:

1) Enhanced yaw rate tracking and reduced deviation from the desired path
2) Improved lateral load balance and reduced slip in transient maneuvers
3) Effective recovery from off-track conditions through active torque correction

By integrating the control algorithm within a modular Simulink model built upon the Powertrain Blockset, the study also highlights the practical feasibility of deploying such systems in simulation environments reflective of Formula Student race conditions.

VII. FUTURE WORK

While the current approach proves effective in simulation, several opportunities exist for further research and real-world adaptation:

1) Real-Time Implementation: Deploying the algorithm on an actual Formula Student vehicle using embedded controllers (e.g., dSPACE, Raspberry Pi, or MicroAutoBox) and validating performance through track testing.
2) Model Predictive Control (MPC): Replacing the rule-based controller with an MPC framework to optimize torque allocation over a prediction horizon, potentially improving performance under more complex driving conditions.
3) Grip Estimation & Adaptive Control: Incorporating real-time estimation of road-tire friction and adapting the control logic dynamically to changing surface conditions (e.g., wet or low-friction environments).

REFERENCES

[1] J. Reif and P. Becker, "Torque Vectoring in Electric Vehicles—Concepts and Implementation," SAE Technical Paper 2011-01-0875, 2011. DOI: 10.4271/2011-01-0875
[2] R. Rajamani, Vehicle Dynamics and Control, 2nd ed., Springer, 2012.
[3] M. Mitschke and H. Wallentowitz, Dynamik der Kraftfahrzeuge, 5th ed., Springer Vieweg, 2014.
[4] MathWorks, "Powertrain Blockset Documentation," [Online]. Available: https://www.mathworks.com/help/autoblks/
[5] T. Gillespie, Fundamentals of Vehicle Dynamics, SAE International, 1992.
[6] B. Guvenc et al., "Coordinated Control of Yaw Moment and Traction Forces Acting on Individual Wheels for Electric Vehicles," IEEE Transactions on Industrial Electronics, vol. 62, no. 5, pp. 3087–3096, May 2015.
[7] M. Canale, L. Fagiano, and M. Milanese, "Robust MPC for Autonomous Vehicle Steering," IEEE Transactions on Control Systems Technology, vol. 17, no. 6, pp. 1436–1445, Nov. 2009.

[8] Y. Gao et al., “Model Predictive Control of a Fully Active Suspension with Preview Information,” IEEE Transactions on Control Systems Technology, vol. 18, no. 6, pp. 1226–1234, Nov. 2010.
[9] T. Shim and C. Ghike, “Understanding the Limitations of Current Active Yaw Control Algorithms,” Vehicle System Dynamics, vol. 46, no. S1, pp. 65–84, 2008.
[10] M. Isermann, “Model-Based Mechatronic Design of Electric Vehicles—Control and State Estimation,” ATZ Worldwide, vol. 117, pp. 32–39, 2015.