

# Design and Optimisation of a Fuzzy Logic Motor Control System in an EV Powertrain

Finn Formica

## Abstract

Large-scale emissions from internal combustion engine vehicles (ICEVs) are a cause of climate change and may lead to long-term environmental damage if left unchecked. To mitigate this, the optimisation of battery electric vehicles (BEVs) is paramount to extend the driving range and reduce the charging time of BEVs so they can compete with ICEVs in the commercial vehicle market. The recent introduction of 800V EV powertrains into the market has helped with these issues, however, it is unclear whether traditional control schemes work on such EV architectures. This report looked to analyse Field-Oriented Control (FOC) and Fuzzy Logic Control (FLC) for their suitability within an 800V EV powertrains for torque control. The results show that both methods are viable control schemes, with FLC providing greatly reduced torque ripples, faster settling time, and reduced tuning required after initial implementation.

Supervised by Dr. J. Wang  
Department of Electrical and Electronic Engineering  
University of Bristol  
2022

## Declaration

This project report is submitted towards an application for a degree in Mechanical and Electrical Engineering at the University of Bristol. The report is based upon independent work by the candidate. All contributions from others have been acknowledged and the supervisor is identified on the front page. The views expressed within the report are those of the author and not of the University of Bristol. I hereby assert my right to be identified as the author of this report. I give permission to the University of Bristol Library to add this report to its stock and to make it available for consultation in the library, and for inter-library lending for use in another library. It may be copied in full or in part for any bona fide library or research worker on the understanding that users are made aware of their obligations under copyright legislation.

I hereby declare that the above statements are true.



Finn Formica

© Copyright, Finn Formica, 2022

Certification of ownership of the copyright in a dissertation presented as part of and in accordance with the requirements for a degree in Mechanical Engineering at the University of Bristol. This report is the property of the University of Bristol Library and may only be used with due regard to the author. Bibliographical references may be noted but no part may be copied for use or quotation in any published work without prior permission of the author. In addition, due acknowledgement for any use must be made.

# Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>  | <b>2</b>  |
| <b>2</b> | <b>Background</b>  | <b>3</b>  |
| 2.1      | DC Link . . . . .  | 3         |
| 2.2      | Inverter Topology . . . . .                                  | 4         |
| 2.3      | Inverter Modulation . . . . .                                | 5         |
| 2.4      | Control Scheme . . . . .                                     | 5         |
| 2.5      | Specification . . . . .                                      | 6         |
| <b>3</b> | <b>Current Control</b>                                       | <b>6</b>  |
| 3.1      | Clarke and Park Transformations . . . . .                    | 6         |
| 3.2      | $dq$ -Axis Decoupling . . . . .                              | 7         |
| 3.3      | PI Controller Design . . . . .                               | 8         |
| 3.4      | Simulation Results . . . . .                                 | 9         |
| <b>4</b> | <b>Torque Control</b>  | <b>11</b> |
| 4.1      | Inner-Loop Current Control . . . . .                         | 11        |
| 4.2      | Outer-Loop Torque Control . . . . .                          | 12        |
| 4.3      | Simulation Results . . . . .                                 | 13        |
| <b>5</b> | <b>Fuzzy Logic Control</b>                                   | <b>15</b> |
| 5.1      | Implementation . . . . .                                     | 15        |
| 5.2      | Simulation Results . . . . .                                 | 17        |
| <b>6</b> | <b>Conclusion and Future Work</b>                            | <b>18</b> |
|          | <b>Appendix A Control Scheme Block Diagrams</b>              | <b>21</b> |
| A.1      | Voltage-Oriented Control . . . . .                           | 21        |
| A.2      | Field-Oriented Control . . . . .                             | 21        |
|          | <b>Appendix B Simulink Model with Field-Oriented Control</b> | <b>22</b> |
|          | <b>Appendix C Fuzzy Inference System Codebase</b>            | <b>24</b> |

## 1 Introduction

Electric vehicles are growing more and more important everyday as they provide a solution to many issues faced by the transport sector. Cities and business globally rely on transport and vehicular travel to simply operate [1]. The global transport sector relies largely on internal combustion engine vehicles (ICEVs) which is problematic as ICEVs emit greenhouse gases -  $\text{CO}_2$ , nitrous oxide ( $\text{N}_2\text{O}$ ), and methane ( $\text{CH}_4$ ) - which account for a large proportion of global and on-road emissions [2]. For example, the UK transport sector was responsible for 23% of UK domestic greenhouse gas emissions in 2019, the majority (55%) of these were from ICEVs [3]. Vehicular emissions have been increasing due to growing populations, urbanisation, and socio-economic development and the resulting increase in vehicle usage [2, 4]. Through contributing to acid deposition, stratospheric ozone depletion, and climate change these emissions could have long-term effects on the environment [5]. If left uncontrolled vehicular emissions may also lead to adverse health effects like respiratory disease, heart disease, and cancer [6]. To prevent long-term damage from these emissions, the UK government have committed to ending the sale of ICEVs from 2030; and from 2035, and all new vehicles must be zero emission at the tailpipe [3]. Thus, attention has been given to EVs to aid in the reduction of emissions whilst preserving non-renewable resources [6]. These advantages have seen a rise in popularity for battery electric vehicles (BEVs) with many more commercial car companies looking to expand their range to the EV market.

One bottle neck for consumers has been the comparatively low driving range of EVs compared with ICEVs, resulting in many people shying away from commercial use from fear of running out of charge [7]. This is but one reason why the continual research and development of EVs is essential and optimising the electrification of the EV powertrain is a key aspect of research to enable further efficiency and range improvements. EVs have intensive performance requirements including high torque and power density, wide speed range, high efficiency, and high torque capability which limit which electric machines can be used [8]. DC motors were initially utilised in most EV systems as they were reliable, inexpensive, simple, and robust. However, they have low efficiency and power density which make their use in EVs undesirable with other electric machines available [9]. The development of power electronics alongside motor control schemes has resulted in permanent magnet (PM) and induction machines taking their place amongst the most used motors in EVs and hybrid electric vehicles (HEVs) [9]. Recently, there has been a rise in popularity for PM machines within powertrain systems, specifically permanent magnet synchronous machines (PMSMs) due to their high energy densities and long lifetimes [10]. However, the use of PMSMs requires complex vector and direct torque control techniques to meet the speed and torque requirements of application within EVs [9].

Another discouraging factor of EVs has been the relatively high charge time when compared with ICEVs, with some 400V BEV batteries taking up to 36 hours to completely replenish [11]. To combat this, the latest commercially available EVs - such as the EV6 Kia, Porsche Taycan, and the Audi e-tron - sport 800V DC link batteries that can massively reduce charge time down to achieving a charge of 80% in 18 minutes for the EV6 Kia [12–15]. Not only does the use of 800V batteries reduce charge time, but they also allow for a greater retention of power, as a higher voltage allows for a lower current to be used when charging and discharging, reducing overheating and contributing toward an increased driving range [16]. The standard BEV powertrain consists of a battery, power management, and the drive units (i.e., PMSMs). The power management subsystem consists of a power converter to invert the DC signals to three-phase AC signals and closed-loop control [7, 10].

Due to the relatively new appearance of 800V EV powertrains, this report was concerned with the design and optimisation of a motor control system within the power management subsystem of an 800V EV powertrain utilising PMSM drives. It was uncertain whether the control schemes utilised in industry-standard 400V EV powertrains would translate across to 800V EV powertrains, and if they did, how simple the tuning process would be when swapping the powertrain architecture. This report is presented in four steps: background research on the EV powertrain; the design of an EV powertrain Simulink model using an ideal voltage source to model the PMSM, with a focus on developing the inner-loop current controller; the replacement of the ideal voltage source with a non-ideal PMSM model, with a focus on developing the outer-loop torque controller; and the replacement of PI control with Fuzzy Logic to compare torque ripples and tuning processes.

## 2 Background

A fundamental understanding of how the power management subsystem operates was required to develop a model of the EV powertrain. As discussed in Section 1, the BEV powertrain consists of a DC link battery, power management subsystem, and PMSM drives; with the power management subsystem made up of an inverter and control scheme.

### 2.1 DC Link

The battery in a BEV was modelled as an ideal DC voltage source. For the inverter to function properly the minimum DC voltage,  $V_{DCmin}$ , needed to be higher than the peak voltage generated by the motor,  $V_{peak}$ . Since the voltage waveform generated was sinusoidal between  $\frac{V_{DC}}{2}$  and  $-\frac{V_{DC}}{2}$ :

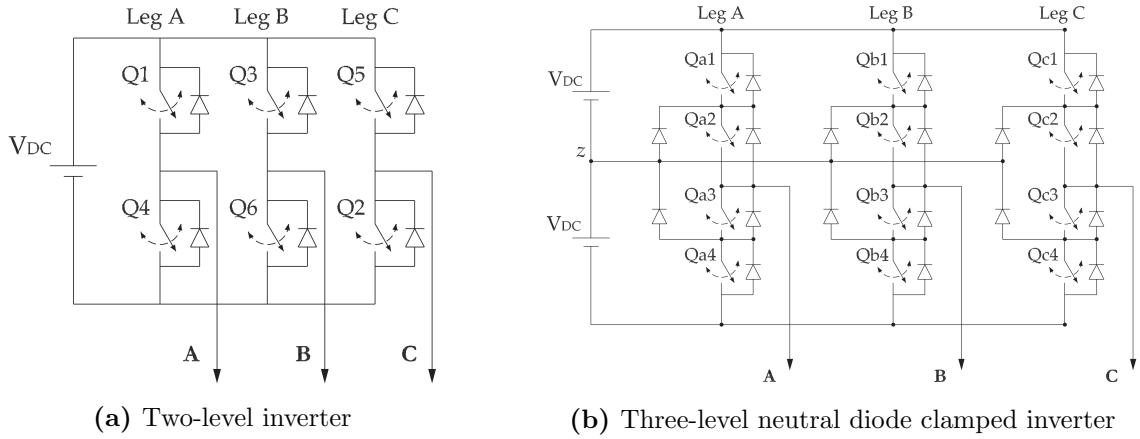
$$\frac{V_{DCmin}}{2} > V_{peak} \implies V_{DCmin} > \frac{2\sqrt{2}}{\sqrt{3}} V_{line} \quad (1)$$

where  $V_{line}$  is the RMS line to line voltage of the motor.

This inequality goes both ways, and so, for a fixed battery size, the maximum voltage rating of a motor was calculated; a  $V_{DC} = 800V$  required a motor of  $V_{line} \leq 490V$ .

## 2.2 Inverter Topology

Three-phase inverters are used in variable-frequency drive applications like in EV powertrains. The conventional three-phase inverter consists of three single-phase legs (each leg consisting of two complementary switching elements), each connected to one phase on the output side creating a line-to-ground output waveform that steps between  $\frac{+V_{DC}}{2}$  and  $\frac{-V_{DC}}{2}$  [17]. This is known as a two-level (2L) inverter as there are only two values the output phase voltage can operate at (topology shown in Fig. 1a). This inverter topology can be limited by the current and voltage ratings of the switching elements when used in applications that exceed those ratings. One way of overcoming the ratings of the hardware is to increase the number of levels on the output side of the inverter, which has led to the design of multi-level inverters [17]. The design of these inverters provides reduced switch utilisation along with a greater number of steps in the output waveform, i.e., three-level (3L) inverters step between  $\frac{+V_{DC}}{2}$ , 0, and  $\frac{-V_{DC}}{2}$  - an example is shown in Fig. 1b.



**Figure 1:** Examples of three-phase inverter topologies [17]

Alican Madan *et al.* [18] provide a comparative study focusing on the Total Harmonic Disturbance (THD) of the signals from two-level (2L) and three-level (3L) inverter topologies in an 800V EV powertrain. The results show that 3L inverters have superior performance, although, the topology has a few drawbacks relating to the floating neutral voltage and a higher number of switches which could lead to implementation difficulties and initial cost. Subhransu Satpathy *et al.* [19] conduct a comprehensive loss analysis of 2L inverters and 3L neutral point clamped (3L-NPC) inverters for 800V EV powertrains. The research analyses the conduction losses and switching losses of each topology using mathematical models. The results show that 3L-NPC inverters have high conduction losses and low switching losses whereas 2L inverters have moderate conduction losses but high switching losses. Hence, an improvement in overall efficiency is observed in the case of the 3L-NPC inverter since it shows a consistent tendency of reduced losses in the complete torque-speed operating range of the EV.

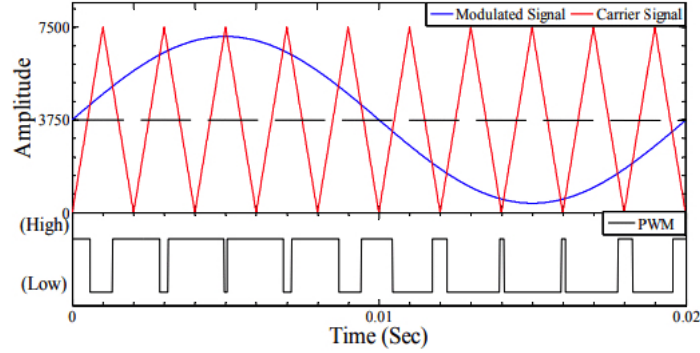
Andreas Burbet *et al.* [20] conduct a comparison of 2L and 3L-NPC inverter topologies for 800V EV powertrains. The authors model the topologies in Simulink and then validate their results using hardware. For switching frequencies above 8kHz, 3L-NPC inverters show clear superiority. 3L-NPC inverters also have lower THD, lower energy consumption, and fault tolerance (limp-home capabilities). In the event of an inverter fault, the EV should still be able to drive to the nearest garage for repairs (“limp home”). If a switch in the 2L inverter was damaged the inverter would be able to produce a single-phase at most. Single-phase excitation of drives is possible, however, in an EV application the operation of a 2L inverter in single-phase excitation mode is not a practical option. By contrast the 3L-NPC inverter would be able to function as a 2L inverter and operate the drives correctly. The authors conclude that the 3L-NPC inverter is a viable alternative to the standard 2L inverter. Furthermore, research in [21] and [22] both validate the comparisons discussed above and conclude

that 3L inverters are the most efficient topology in an 800V EV powertrains. However, the authors note that increasing the topology past three levels is not cost efficient due to the increased cost of switching elements, wiring, and gate drivers.

### 2.3 Inverter Modulation

Modulation is the method of controlling the inverter to generate the desired output waveform. By adjusting the signals sent to the inverter the on and off duration of its switches can be controlled. PMSM drives in EV applications require three-phase sinusoidal (AC) current, which is commonly accomplished through Sinusoidal Pulse Width Modulation (SPWM) or Space Vector Pulse Width Modulation (SVPWM).

SPWM utilises three reference sinusoidal signals (each out of phase by  $\frac{2\pi}{3}$ ) and compares them against a common triangular carrier signal to generate the PWM switching signals for each inverter leg. If the sinusoidal signals (modulated signal) are above the carrier signal then the output reads high, else the output reads low [23] - shown in Fig. 2. This signal is inverted for the complementary switch so that there will never be a short-circuit.



**Figure 2:** Graph showing PWM signal generation using reference sine wave [24]

Whilst SPWM is very simple to implement, it only has a peak amplitude equal to  $\frac{V_{DC}}{2}$  (or a modulation index equal to 1) [23]. However, a method called Third Harmonic Injection (THI or THISPWM) can be used to increase the modulation index by 15%.

Space Vector Pulse Width Modulation (SVPWM) utilises Space Vector Modulation (SVM) which has been developed as a vector approach to SPWM for three-phase inverters. SVM provides a higher voltage to the motor whilst lowering Total Harmonic Distortion (THD) through its unique sine wave generation. SVPWM transforms the three-phase voltages into their two-phase equivalent quantities in the stationary  $\alpha\text{-}\beta$  frame. The reference vector magnitude is found from these two-phase components and used for modulating the inverter output [25] [26].

The authors in [18] and [27] conduct analyses of SPWM and SVPWM. Research in [27] simulates a 3L inverter with linear and non-linear loads whilst [18] analyses the modulation techniques across a 2L and 3L inverter. The results across both papers show that SVPWM has lesser THD compared with its counterpart. Yan-jun Yu *et al.* [28] provide analysis of SPWM and SVPWM with the focus on losses in the inverter for PMSM drives. The analysis uses a mathematical model to determine the conduction and switching losses for the diodes and switching elements in the inverter. The results show that the switching losses of SVPWM are smaller than SPWM and the conduction losses for both techniques are comparable, and hence, SVPWM is a more efficient method of modulation.

### 2.4 Control Scheme

The closed-loop control scheme generates the signals to send to the inverter, to drive the PMSM to the desired speed or torque. Popular control schemes for PMSMs involve either vector control or direct

torque control (DTC). The most popular vector control scheme, Field-Oriented Control (FOC), consists of an outer-loop torque control which generates reference currents based on the torque demand to feed into the inner-loop current control, which controls the PMSM through its  $dq$  currents. The current control provides decoupling of the  $d$  and  $q$  axis currents to obtain a simplified expression of torque that can be controlled independently by the  $q$ -axis current. The vector control requires knowledge of the rotor position, which is typically measured through a mechanical sensor. The DTC scheme allows direct control of the flux and electromagnetic torque of the PMSM, requiring an estimation of the PMSM's flux and torque to be effective [29]. Both methods use Proportional-Integral (PI) controllers within the control scheme; however, these can be replaced with more advanced controllers researched from the literature to produce a more desirable response.

Research in [29–31] all provide a performance evaluation between FOC and DTC for PMSM drives. All the authors conclude that DTC has a faster dynamic performance, reaching steady-state quicker than FOC, but with increased torque ripples. DTC is preferable if a fast dynamic performance is priority whereas the FOC is better suited when high torque quality is demanded. The DTC scheme requires fewer motor parameters than FOC and so is less complex and has less implementation difficulties. Yifang Zhu *et al.* [32] provide a comparative performance evaluation of Model Predictive Control (MPC) and PI control for the speed control of a PMSM. MPC is a computer-aided control algorithm based on model optimisation. It boasts real-time model prediction, rolling optimisation, and feedback correction. The paper shows the effectiveness of reducing overshoots commonly seen with PI controllers and the MPC scheme boasts a fast response with small torque ripples. Guesmi Oussama *et al.* [33] provide a comparison between PI and fuzzy controllers in speed control of PMSM. Fuzzy Logic Control (FLC) is based upon logical reasoning and human readable rules to determine the behaviour of the controller. It allows the design of a controller without detailed knowledge of the dynamics of the system, merely an intuition about how it behaves. The results show FLC eliminates overshoot normally seen with PI control. It also improves the dynamic performances of the PMSM drive showing reduced torque ripples. The authors recommend FLC and vector control (FOC) to be a good combination.

## 2.5 Specification

The specifications for the EV powertrain model are from a commercial 490V PMSM (line to line voltage) and will not be disclosed publicly as requested by the company.

## 3 Current Control

For applications within an EV powertrain, high quality torque was more desirable than a fast dynamic response. This would provide a more comfortable ride to the passengers whilst also reducing the long-term effects a large torque ripple may cause on the drive units. For these reasons the vector control scheme Field-Oriented Control (FOC) was chosen over its direct torque control counterpart. The focus of this section was the development of the inner-loop current control (or current controller). To simplify the model, an ideal three-phase voltage source was utilised to replace the PMSM drive, and hence, a variation of FOC was required, Voltage-Oriented Control (VOC) [34]. VOC only utilised an inner current control loop and the block diagram of the scheme is shown in Fig. 25 in Appendix A.1.

### 3.1 Clarke and Park Transformations

It is difficult for PI controllers to control sinusoidal signals due to their oscillating nature, and so, the current controller utilised the Clarke and Park transformations to convert the sinusoidal PMSM phase currents into constant values. This involved a set of two transformations that transformed a three-phase signal into the orthogonal rotating  $dq$  reference frame. This reference frame was rotating at the speed of the machine (PMSM) and so it was much easier to control each element of the reference frame as they settled to a constant value when the system entered steady-state [35].

The Clarke and Park transformations have two forms, the Magnitude invariant, and the Power invariant. The Magnitude invariant transformation is most commonly used in motor drive systems as the

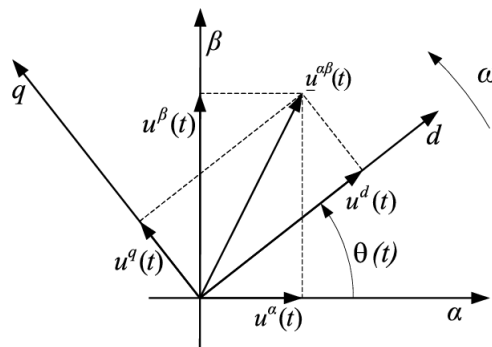
measured motor voltage is the same magnitude as the transformed signals in the  $\alpha\beta$ -frame and the  $dq$ -frame, making it very intuitive for testing. In the Power invariant transformation, the power levels in each reference frame are equal [36]. For these reasons the Magnitude invariant transformation will be used, and the matrices are shown below [34]:

$$\text{Clarke transformation: } \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \text{ and } \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

$$\text{Park transformation: } \begin{bmatrix} d \\ q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \text{ and } \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} d \\ q \end{bmatrix}$$

The Park transformation requires the angular position of the rotor of the PMSM. In a real system this is gathered through an estimation of the rotor position using a Hall-effect sensor [37]. However, since the PMSM was being modelled as an ideal voltage source, the angular position was based on the angle of the phase A voltage signal. The relationship between the  $\alpha\beta$ -frame and  $dq$ -frame can be seen in Fig. 3. From this, it was derived that:

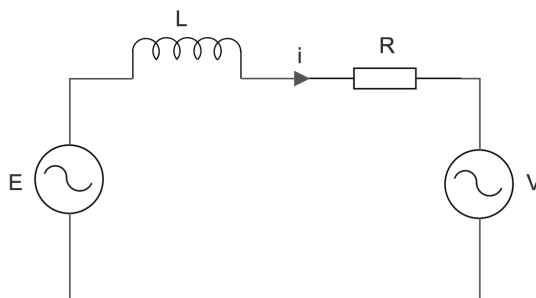
$$\theta = \text{atan}^{-1}(\frac{\beta}{\alpha}) \quad (2)$$



**Figure 3:** Relation between  $\alpha\beta$ -frame and  $dq$ -frame [38]

### 3.2 $dq$ -Axis Decoupling

A set of equations was developed to generate reference voltages for the SPWM. Since the current controller operated in the  $dq$ -frame the equations were applicable in this reference frame.



**Figure 4:** Circuit diagram of a single-phase grid connected inverter

Fig. 4 shows a single-phase circuit diagram of the model being built in Simulink.  $E$  represents the voltage of the grid whilst  $V$  represent the voltage of the inverter;  $L$  and  $R$  are the respective inductance

and resistance between the two. Analysing a single-phase system was much simpler than a three-phase system and the equations developed still applied to the three-phase system. Using KVL on the circuit, a mathematical model of the system was generated:

$$E = L \frac{di}{dt} + j\omega L + iR + V \quad (3)$$

where  $\omega$  is the fundamental frequency of the system and  $L$  and  $R$  are the inductance and resistance between the grid and inverter. Equation (3) was transformed into the  $dq$ -frame to yield:

$$E_d = L \frac{di_d}{dt} + \omega L i_q + R i_d + V_d \quad (4)$$

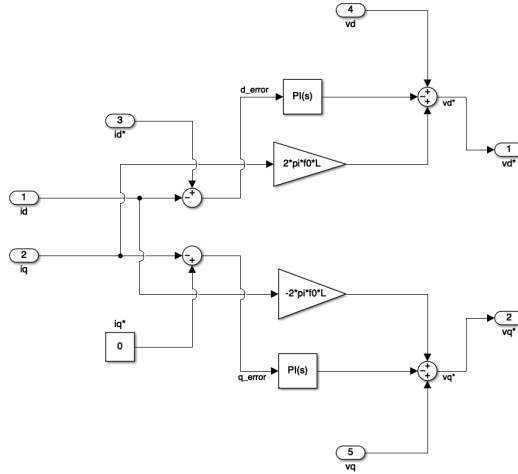
$$E_q = L \frac{di_q}{dt} - \omega L i_d + R i_q + V_q \quad (5)$$

where the subscripts  $q$  and  $d$  denote which axis in the  $dq$ -frame the variable is in. Equations (4) and (5) displayed cross-coupling between each other, due to the  $\omega L$  part. Cross-coupling makes it difficult to effectively control the signals and so it was important to decouple the equations by removing the coupling term within the current controller [39]. Using [34] as reference, (6) and (7) were implemented into Simulink to generate the reference voltages,  $V_d^*$  and  $V_q^*$ :

$$V_d^* = E_d - (k_p \epsilon_d + k_i \int \epsilon_d dt) + \omega L i_q \quad (6)$$

$$V_q^* = E_q - (k_p \epsilon_q + k_i \int \epsilon_q dt) - \omega L i_d \quad (7)$$

and this can be seen in Fig. 5. In VOC, the direct-axis represents real power, and the quadrature-axis represents reactive power. Hence, the quadrature current,  $i_q$ , was driven to zero and the direct current,  $i_d$ , held at a non-zero value.



**Figure 5:** Simulink block for reference voltage generation

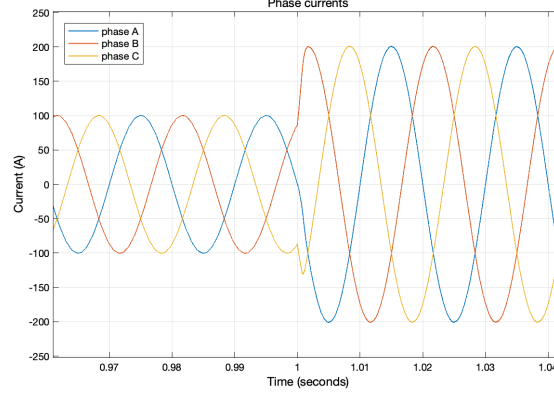
### 3.3 PI Controller Design

The PI controllers in the control loop were tuned to minimise the overshoot of the response, the settling time of the response, the steady-state error, and reduce low-frequency harmonics that could not be removed through a filter. In this control loop there were two PI controllers, one each for the errors of  $i_d$  and  $i_q$ . A common method for tuning the controllers is through trial and error, adjusting  $k_p$  and  $k_i$  (the proportional and integral gains) and assessing how they affect the response of the system. Generally, increasing  $k_p$  will help the system react to inputs quicker whilst also increasing the oscillations, whereas increasing  $k_i$  helps to reduce settling time of the system and decrease steady-state error. For both PI controllers, the gains were  $k_p = 20$  and  $k_i = 300$ .



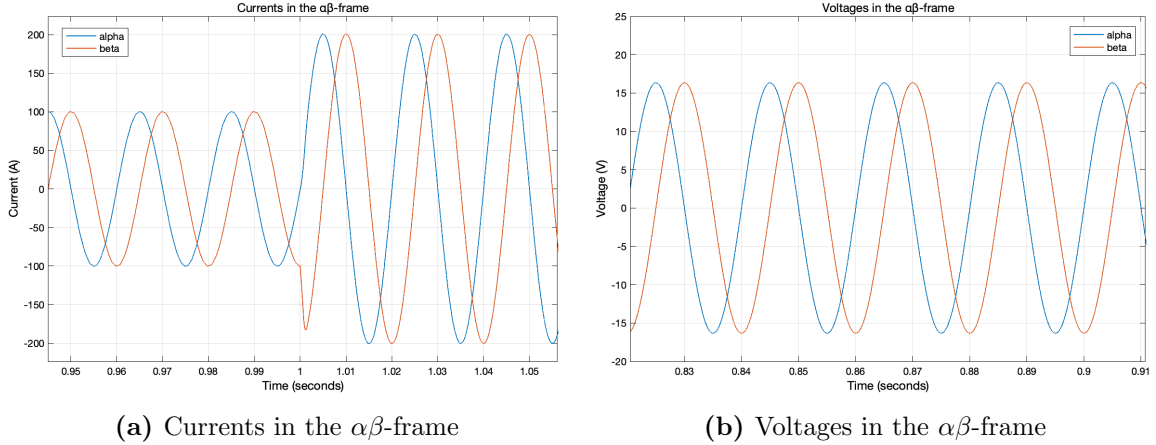
### 3.4 Simulation Results

The goal of SPWM is to produce three-phase sinusoidal current signals that feed into the load. Fig. 6 shows the phase currents of the inverter through a step input from  $i_d = 100A$  to  $i_d = 200A$ . Since the Magnitude invariant transformation was used,  $i_d$  was equal to the peak current of the three-phase signals. The modulation was working as intended in producing the correct current waveforms with little distortion.



**Figure 6:** Phase currents in the inverter

When transformed from the abc-frame to the  $\alpha\beta$ -frame two sinusoidal signals should have been produced, with the  $\alpha$  signal leading the  $\beta$  by  $90^\circ$  - as can be seen from Fig. 7 with both the voltage and current signals exhibiting this.

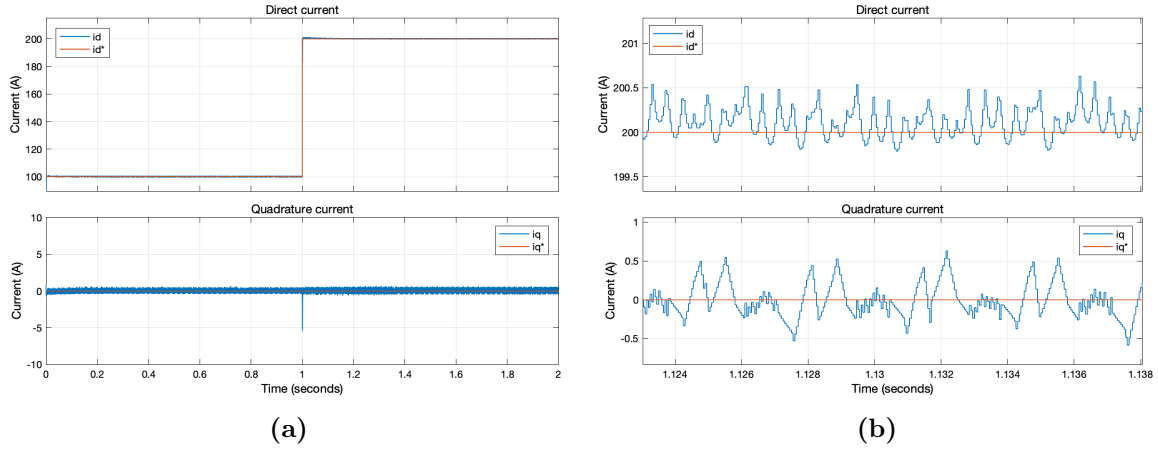


**Figure 7**

Fig. 8a shows the step response of the direct current, with minimal overshoot, a quick settling and rise time, and no overall steady-state error. Since it was infeasible to attain a perfectly tuned PI controller, this was considered moderately tuned. Although the PI controllers were tuned properly for this system, transferring the control subsystem to another EV powertrain with a slightly different DC link would require the whole tuning process to be restarted.

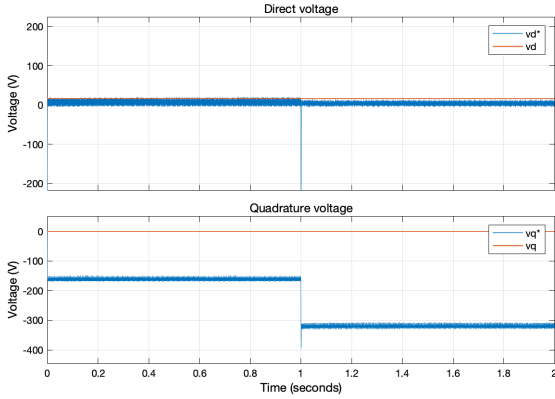
Fig. 8b shows a zoomed in figure of the current waveforms. Although in Fig. 8a it looks like the  $dq$  currents followed the references perfectly, both signals had a current ripple. This is very common in these types of applications and there are methods that can be taken to reduce it.

Fig. 9 shows the  $dq$  voltages of the grid and the reference  $dq$  voltages produced by the current controller. The grid  $dq$  voltages had no noise and were constant values because they were transformed from a purely sinusoidal signal. However, the reference  $dq$  voltages were much noisier as they were generated from several different signals. The noise from the  $dq$  currents was amplified by the PI controllers and hence produced a much noisier  $dq$  voltage signal.

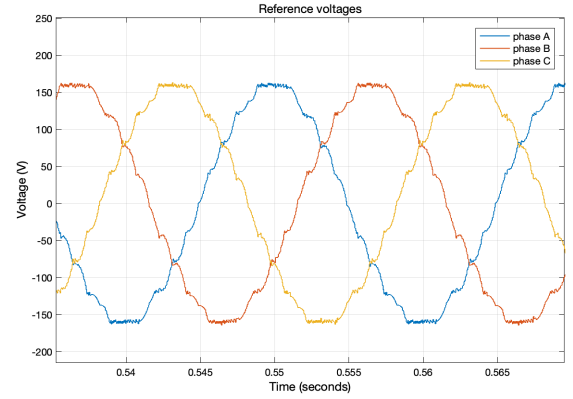


**Figure 8:** Graphs showing the direct and quadrature currents

In Fig. 9 the  $v_d$  signal was closely tracking the reference signal, however, the  $v_q$  signal was negative, not zero, and during the step response it dropped further. This was because the  $v_q^*$  signal was mainly contributed to by the decoupling term,  $-\omega L i_d$ , and caused it to be negative. The step response of  $i_d = 100A$  to  $i_d = 200A$  caused the decoupling term to become more negative, and hence, caused a drop in the reference quadrature voltage.



**Figure 9:** Graph showing the direct and quadrature voltages



**Figure 10:** Graph showing the reference voltages for SPWM

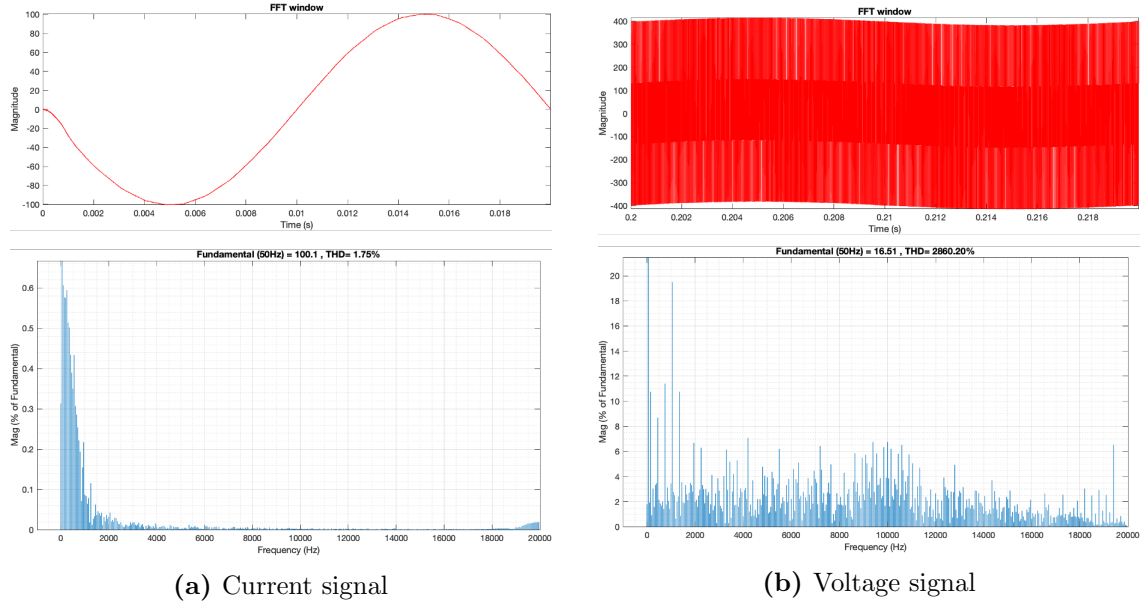
The noise from the  $dq$  voltage signals translated into a noisy sinusoidal reference signal for the SPWM block (shown in Fig. 10). Although there were lots of high frequency components in the reference signals, the general shape and phase difference of the signals were both accurate. By passing the  $v_d^*$  and  $v_q^*$  signals through a low-pass filter, the reference voltage signals were much less noisy, however, this was at the cost of a very high settling time of the system and huge oscillations within the currents which would be more than the real system could handle. Furthermore, by increasing the switching frequency of the system, the effect of the noise of the signals was reduced without sacrificing settling time of the system.

The Fast Fourier Transform (FFT) was used to analyse the periodic time-domain signals and break them down into their independent frequency components in the frequency domain. In this model, the fundamental frequency was 50Hz, so it was expected that a large proportion of the signals were made up of this frequency, however, it was also useful to see what other frequencies were leaking into the signals and causing distortion. If there were lots of high-frequency components, for example, a low-pass filter could have been introduced to help reduce this noise. The PI controllers were tuned to reduce low-order harmonics; however, this was done manually and by eye without the help of any software. FFT analysis is a tool for determining the effectiveness of the controller filtering.

Fig. 11a shows a single period of the phase A current and its corresponding FFT analysis. Most of

the signal was made up of a 50Hz signal, with small amounts of other frequencies. There was also an increase in frequencies around 20kHz which was the switching frequency of the system, showing that some high frequency components were leaking into the signal even if the effect of this was minimal. The Total Harmonic Distortion (THD) of the current signal was very small, only 1.75%, which shows that there was very little noise or disturbance in the current. This was likely due to the inductor between the grid and the inverter. It was acting as a filter for the current and helping to reduce the noise in the signal.

Fig. 11b shows a single period of the phase A voltage and its corresponding FFT analysis. The signal was made up mostly by a 50Hz signal, however, the signal was much noisier and greater amounts of other frequencies disturb the signal. The THD of the voltage signal was also much greater than the current, at 2860.20%. This was simply a by-product of PWM, and it is very common to have such high THD on the voltage waveform.



**Figure 11:** Graphs the FFT analysis of a phase voltage and current signal

## 4 Torque Control

Replacing the ideal voltage source with a PMSM model allowed a more accurate simulation of the EV powertrain and also provided data like rotor speed, rotor angle, and electromagnetic torque. Adapting VOC to FOC for the PMSM involved slight changes to the current controller and the addition of the outer-loop torque control. Fig. 26 in Appendix A.2 displays the block diagram of an FOC scheme.

The signals taken from the PMSM model needed to be processed before use within the control system. In most cases, this simply meant adding a rate transition block to remove high frequency noise and to help speed up the simulation. However, the rotor speed and rotor angle had a gain applied to them equal to the number of pole pairs in the PMSM since:

$$\omega_m = N\omega_{mech} \quad (8)$$

$$\theta_m = N\theta_{mech} \quad (9)$$

where  $\omega_{mech}$  and  $\theta_{mech}$  are the sensed rotor speed and angle,  $N$  is the number of pole pairs, and  $\omega_m$  and  $\theta_m$  are the electrical signals of the rotor speed and angle [40]. Furthermore, the modulus of the rotor angle was taken with  $2\pi$  to prevent the signal from becoming too large.

### 4.1 Inner-Loop Current Control

The inner-loop current control accomplished the same job as before, however, the decoupling method varied slightly without an ideal voltage source. Using the mathematical model of a PMSM in [40], the

following equations were derived:

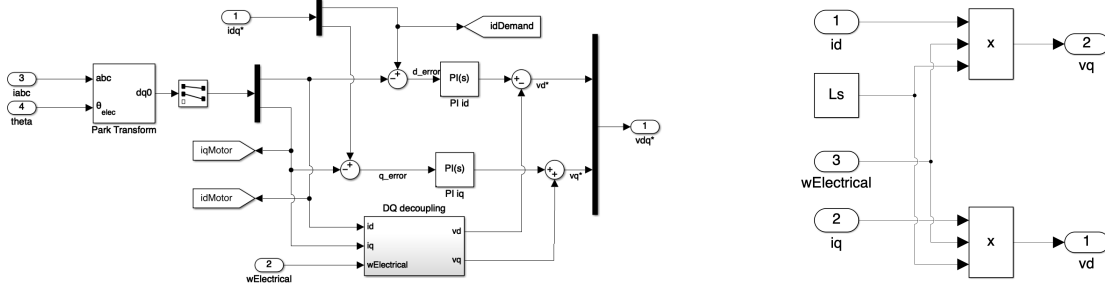
$$v_{sd} = R_s i_{sd} + \frac{d}{dt} L_s i_{sd} - \omega_m \Psi_{sq} \quad (10)$$

$$v_{sq} = R_s i_{sq} + \frac{d}{dt} L_s i_{sq} + \omega_m \Psi_{sd} \quad (11)$$

$$\Psi_{sd} = L_{sd} i_{sd} + \Psi_{PM} \quad (12)$$

$$\Psi_{sq} = L_{sq} i_{sq} \quad (13)$$

where  $v_{sd}$  and  $v_{sq}$  are the  $dq$ -axis stator voltages,  $R_s$  is the stator resistance,  $i_{sd}$  and  $i_{sq}$  are  $dq$ -axis stator currents,  $L_s$  is the stator inductance,  $\omega_m$  is the rotor speed, and  $\Psi_{PM}$  is the permanent magnet flux linkage (only present in the  $d$ -axis). For the purposes of this model the effect of  $\Psi_{PM}$  in these equations was negated and set to zero. To decouple the above equations, the coupled terms ( $\omega_m \Psi_{sd}$  and  $-\omega_m \Psi_{sq}$ ) were removed from the  $dq$  current signals. The implementation of this in Simulink is shown in Fig. 12. The gains for both controllers were  $k_p = 50$  and  $k_i = 1000$ .

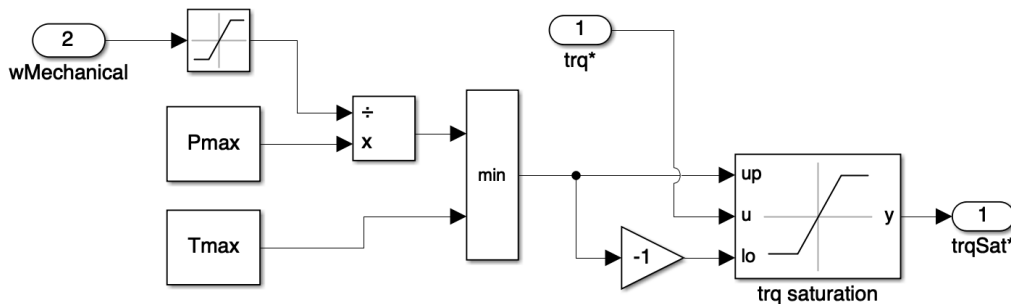


**Figure 12:** Current controller (left) and decoupling network (right) of the inner loop control

## 4.2 Outer-Loop Torque Control

The outer-loop torque control minimised the error between the desired torque,  $T_{ref}$ , and the torque of the PMSM,  $T_{em}$ . The error between the signals was fed into a PI controller to generate a reference current to feed into the current controller. The PI controller was tuned with the same method discussed in Section 3.3, and the gains for the controller were  $k_p = 50$  and  $k_i = 300$ .

For any application, if the torque demanded is too great for the PMSM to handle there may be damage caused to the machine. To prevent this from happening a torque limiter was used to consider the max power,  $P_{max}$ , and max torque,  $T_{max}$ , that the machine could handle. Since  $P = \tau_{em} \omega_m$ ,  $P_{max}$  was divided by the PMSM rotor speed to find a torque limit based on the max power. The minimum between  $P_{max}$  and  $T_{max}$  was taken and then fed into a saturation block with the demanded torque,  $trq^*$ , to produce a saturated torque,  $trqSat^*$ , that was within the limits of the PMSM. The rotor speed was also saturated to prevent  $trqSat^* = 0 \text{ Nm}$  when the machine was starting up. Fig. 13 shows how this was accomplished in Simulink.



**Figure 13:** Torque limiter Simulink block

The saturated torque was then manipulated to produce reference  $dq$  currents for the current controller. Starting with the expression for electrical power,  $p_e = \frac{3}{2}(u_d i_d + u_q i_q)$ , and using (10-13), an equation for the electromagnetic torque of the PMSM was developed [41]:

$$T_{em} = 3N(\Psi_{sd}i_{sq} - \Psi_{sq}i_{sd}) \quad (14)$$

which was rearranged with (10-13) to yield:

$$i_{sd}^* = \frac{\Psi_{sd} - \Psi_{PM}}{L_s} \quad (15)$$

$$i_{sq}^* = \frac{T_{em}}{3N\Psi_{PM}} \quad (16)$$

where  $N$  is the number of pole pairs in the PMSM [41]. Equations (15) and (16) show that to get maximum torque the quadrature current should have been maximised, and the direct current set to zero; if there is no field weakening involved [41]. This was converse to what was implemented previously with VOC. Since  $i_q \propto T_{em}$ , the output of the PI controller did not require any further manipulation and was fed straight into the inner control loop. Fig. 14 shows how the torque limiter and current reference generation was implemented within Simulink. Figure 27 in Appendix B shows the complete Simulink model with FOC implemented.

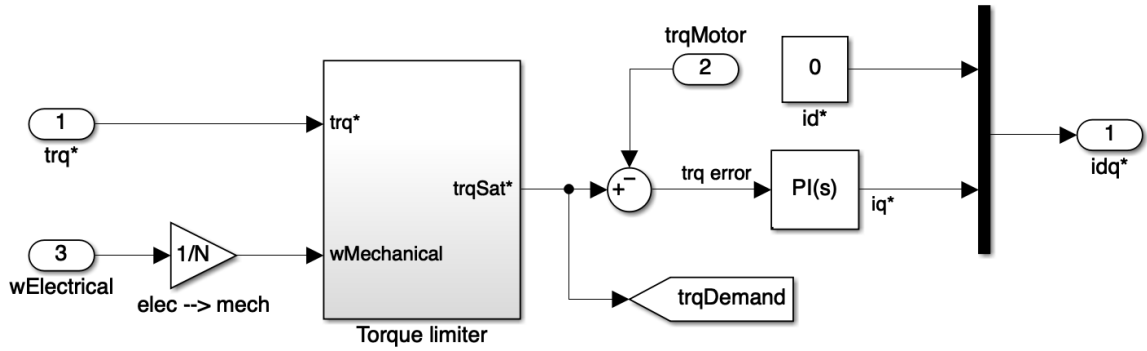


Figure 14: Simulink block for the outer loop control

### 4.3 Simulation Results

As stated previously the goal of SPWM is to produce three-phase sinusoidal current signals. Fig. 15a shows that this is working as intended in the PMSM, even if the reference voltage signals were extremely noisy (see Fig. 15b).

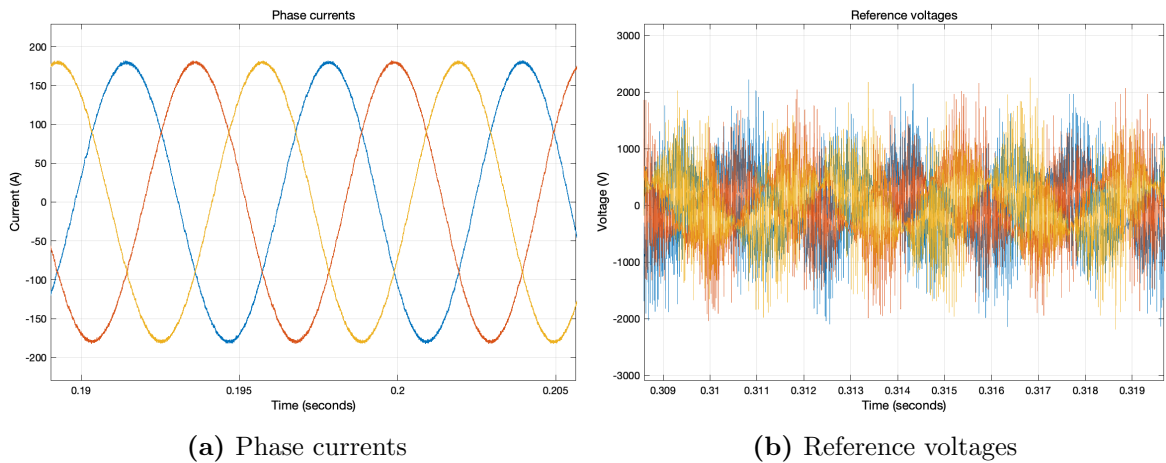
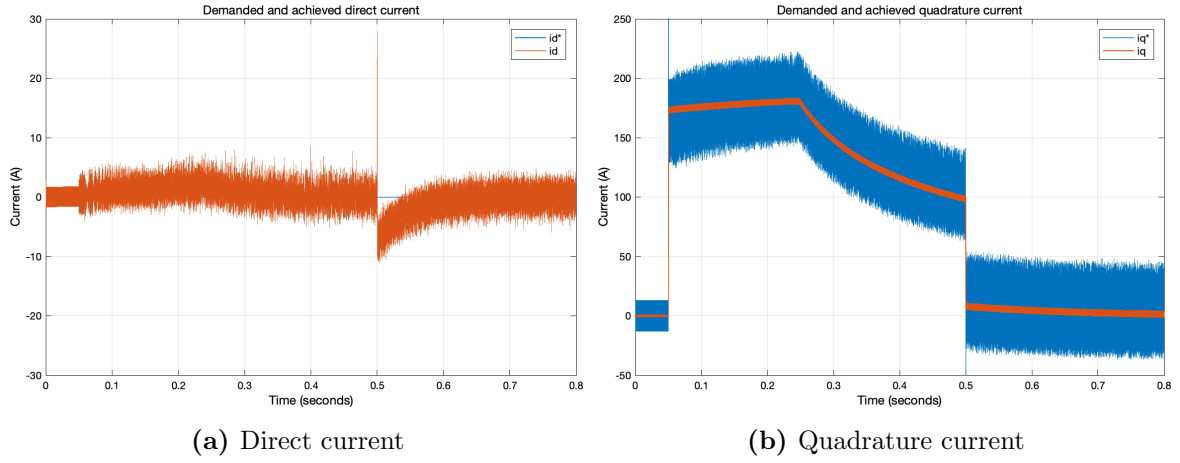


Figure 15

Fig. 16 shows the  $dq$  currents in the PMSM. These graphs show the current controller was working as intended, although,  $i_d$  was very noisy and  $i_q^*$  produced by the outer-loop control was also very noisy.

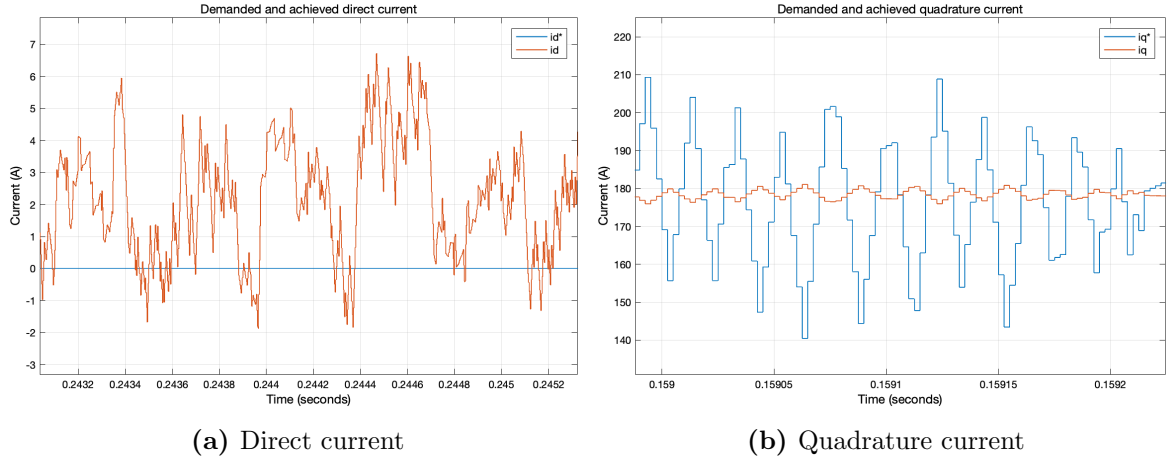
Fig. 17a shows that  $i_d$  was simply noisy, however, Fig. 17b shows that  $i_q^*$  had a current ripple that was causing a ripple in  $i_q$ . This led to a torque ripple within the PMSM which would not have been healthy



**Figure 16:** Graphs showing the demanded and achieved currents

for the machine, nor would it have been a comfortable ride for passengers in the EV if it was too large. The current ripples were calculated from Fig. 17b using (17) to give  $i_q$  a ripple of 2.8% and  $i_q^*$  a ripple of 33.3%. The ripple produced was common in PI controllers due to non-linearities introduced into the system and load torque variations [42]. To overcome this, more advanced controllers were later used to replace the outer loop PI controller which lessened the torque ripple.

$$\%ripple = \frac{i_{max} - i_{min}}{0.5(i_{max} + i_{min})} * 100 \quad (17)$$



**Figure 17:** Graphs showing the demanded and achieved current waveforms

The outer-loop torque control was designed to minimise the error between the reference torque,  $trq^*$ , and the torque produced by the motor,  $trq$ . The model was put through a step input of  $trq = 0 \text{ Nm}$  to  $trq = 50 \text{ Nm}$  and then back down to  $trq = 0 \text{ Nm}$ . In Fig. 18a the  $trq^*$  curved downward partway through the step response because the rotor speed had increased enough to trigger a decrease in the  $trqSat^*$  from the torque limiter. This prevented the PMSM from being overworked, and without it the quadrature current would have broken down and begun to oscillate around zero. As seen in Fig. 18b, the torque was producing a ripple due to the current ripple as discussed previously. Using (17), the torque ripple was calculated to 3.8%.

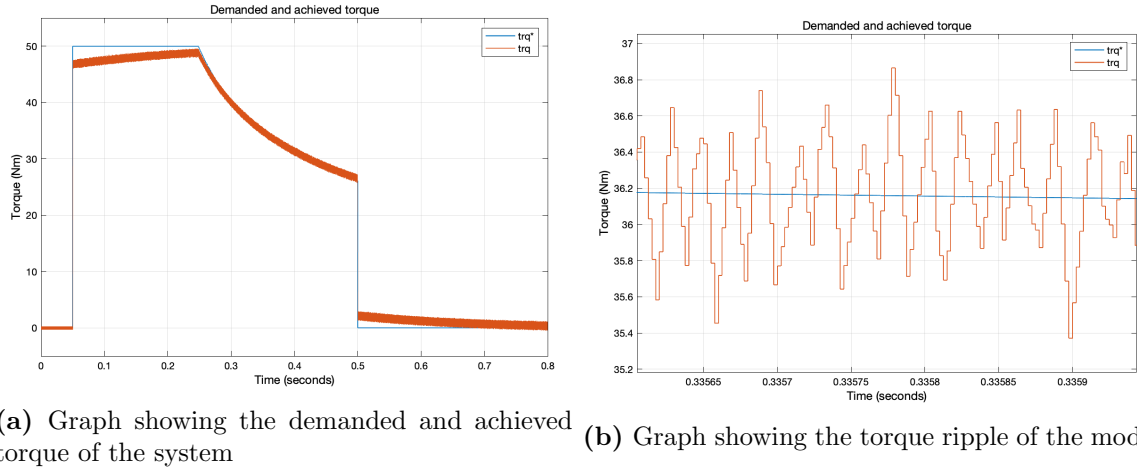


Figure 18

## 5 Fuzzy Logic Control

Fuzzy logic is a method of modelling logical reasoning where there are degrees of truth to a statement, not simply a binary answer. Fuzzy logic allows the development of a Fuzzy Inference System (FIS) which uses human interpretable rules instead of abstract mathematics like traditional PI controllers. Controllers which implement this system, Fuzzy Logic Controllers (FLCs), are a popular method for reducing torque ripples as they have superior performance to PI controllers and boast a greatly reduced overshoot. Fuzzy Logic Controllers consist of [43]:

- *Fuzzification*: the process of converting crisp input values into the fuzzy linguistic variables used in the Fuzzy Inference System.
- *Rule Base*: a collection of the human interpretable rules that govern how the controller behaves.
- *Fuzzy Inference System*: the system that uses the fuzzified inputs and maps them to fuzzy outputs using the rule base.
- *Defuzzification*: the process of converting the fuzzy linguistic variables into crisp output values.

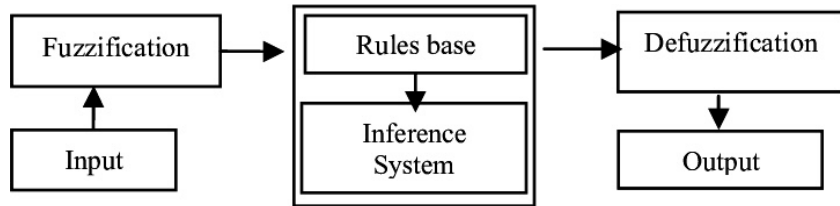


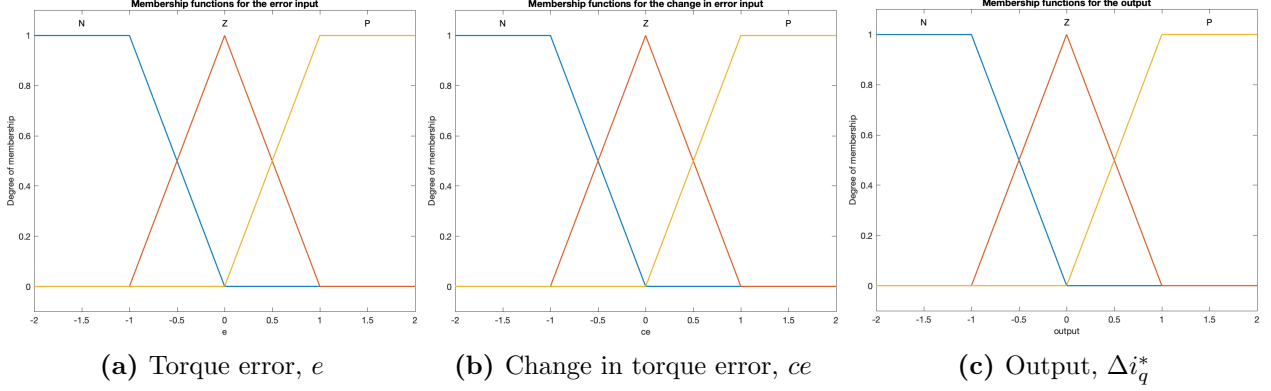
Figure 19: Block diagram of a Fuzzy Logic Controller [44]

Fig. 19 shows how these parts interact together within the FLC. Due to the FLC acting on human interpretable rules, there is no need for a model of the system being controlled. This is hugely advantageous over traditional PI control which usually requires a detailed model of the system dynamics to produce a controller with the desired design requirements. The rules instead require an intuitive understanding of how the system works without needing to know its non-linearities and intricacies. The tuning process is also greatly simplified, as the rules work on system behaviour instead of numerical values. This should allow FLCs to be transferred across similar systems, i.e., from a 400V EV powertrain to an 800V EV powertrain.

### 5.1 Implementation

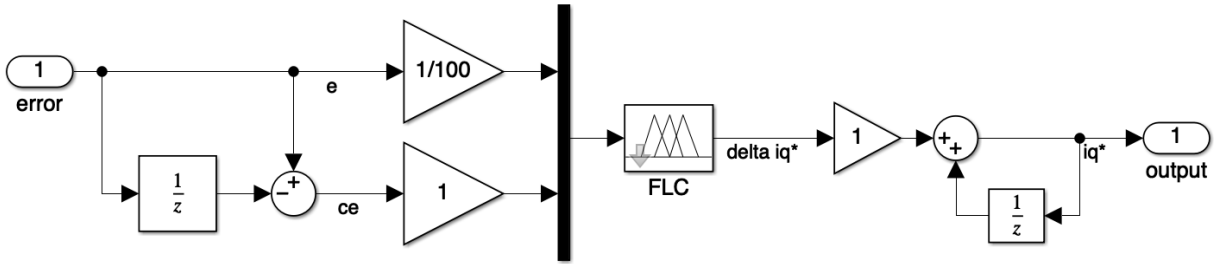
The fuzzification of crisp input variables required the use of membership functions. These functions are the different linguistic sets of fuzzy variables that the crisp value can belong to. For this model

only 3 membership functions were defined for each input and output - negative ( $N$ ), positive ( $P$ ), and zero ( $Z$ ). It is traditional to use 7 membership functions corresponding to small, medium, and large negative and positive values plus zero, however, this adds extra complexity to the FLC with negligible benefits [45]. Fig. 20 shows the membership functions defined for the inputs and output. They ranged from only  $[-2, 2]$  so that gains could be applied to either side of the FLC to tune the response in Simulink [45]. These gains allowed the response of the FLC to be customised like a PI controller without the complicated tuning process. The shapes of the membership functions (trapezoidal and triangular) were chosen for simplicity and could later be adjusted to produce a better system response.



**Figure 20:** Graphs showing the membership functions for each input and output

In this model, the FLC replaced the PI controller in the outer-loop torque control which generated the reference currents for the current controller. The FLC was fed the torque error and produced a reference  $q$ -axis current,  $i_q^*$ . For the controller to function properly, it required both the error,  $e$ , and the change in error,  $ce$ . This was accomplished by using a unit delay on the error and taking the difference with the value of the previous time-step. The output of the FLC was the action the system needed to take - the change in  $i_q^*$ . Another unit delay was used in a positive feed-forward loop to generate the  $i_q^*$  signal for the inner-loop control [45]. Fig. 21 shows how this was implemented in Simulink.



**Figure 21:** Implementation of the Fuzzy Logic Controller in Simulink

The rules of the FIS are based on simple intuition and can be developed simply by thinking through what needs to be accomplished. The rules of the FIS can be generalised by:

$$R_i : \text{If } e \text{ is } A_i \text{ and } ce \text{ is } B_i \text{ then output is } C_i \quad (18)$$

For example, if  $e$  was negative and  $ce$  was also negative, the controller needed to cause a change in current in the opposite direction to that of  $ce$  [46]. Since each input and output had three membership functions, there was a total of nine rules - one for each case of  $e$  and  $ce$ . The rules of the FIS are shown in Table 1 in a compact matrix form and Fig. 22 shows the output surface for the system that the rules and membership functions generate.

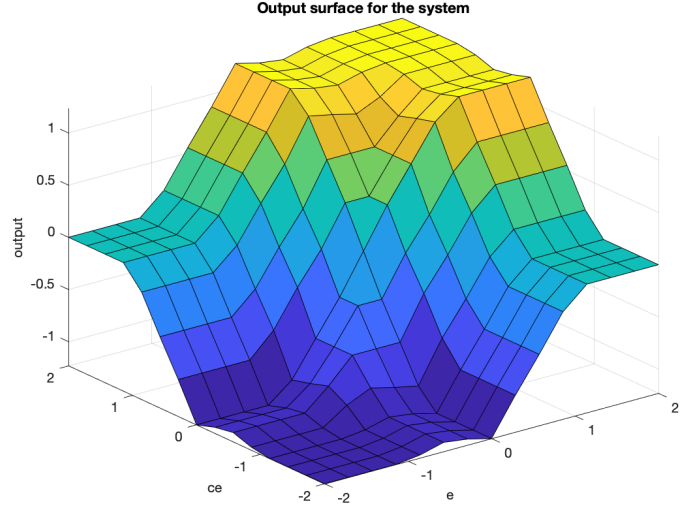
There are several different methods for the defuzzification of the output but the most popular is called the centroid method. This returns the centre of gravity for the fuzzy set along the x-axis. However,



the centroid method will never return a value at the top or bottom of the range of the inputs and output. There are other available methods of defuzzification that get around this problem, but since the inputs and output were scaled in the simulation this did not pose a problem. The full codebase for the FIS can be seen in Appendix C.

|                              |   | Torque error, $e$ |   |   |
|------------------------------|---|-------------------|---|---|
|                              |   | N                 | Z | P |
| Change in torque error, $ce$ | N | N                 | N | Z |
|                              | Z | N                 | Z | P |
|                              | P | Z                 | P | P |

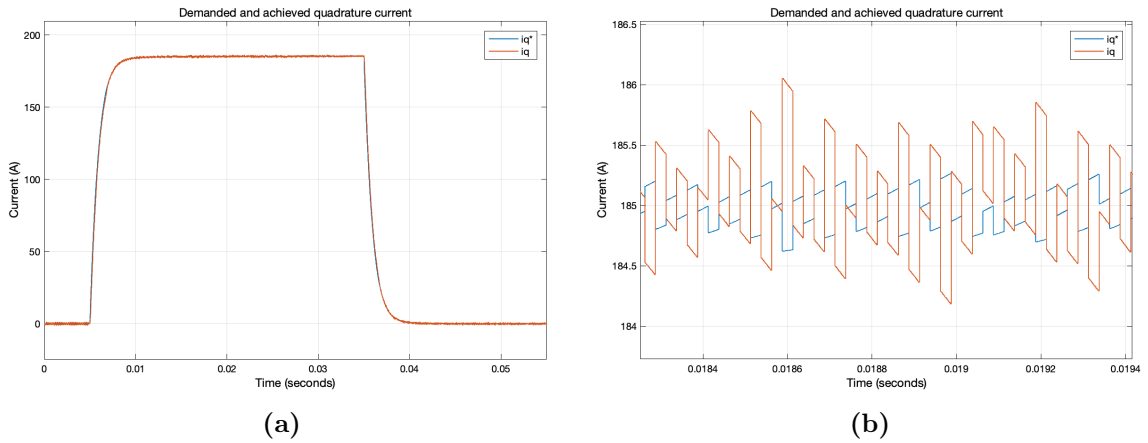
**Table 1:** Matrix of fuzzy rules



**Figure 22:** Graph showing the output surface for the Fuzzy Inference System

## 5.2 Simulation Results

FLC was implemented to reduce the ripple in  $i_q^*$  as this led to a large ripple within  $i_q$  which caused a ripple in the torque. Fig. 23a shows  $i_q^*$  and  $i_q$  with much fewer noisy signals than previously. The current ripples were calculated from Fig. 23b using (17) to give  $i_q$  a ripple of 0.86% and  $i_q^*$  a ripple of 0.32%. Both values showed a clear reduction from 2.8% and 33.3% which showed the FLC was working as intended. Furthermore, the quadrature current had a much smoother response than previously, which would be beneficial for the PMSM.

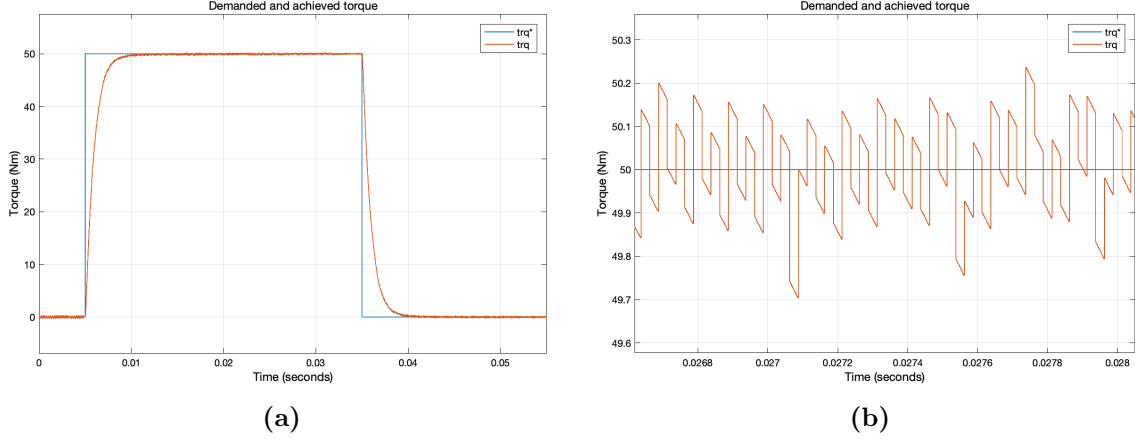


**Figure 23:** Graphs showing the demanded and achieved quadrature currents

Fig. 24a shows  $trq^*$  and  $trq$  of the PMSM. The torque from the machine had a noticeably smoother step response and a much faster settling time. From Fig. 24b the torque ripple was visibly much smaller and using (17) it was calculated to 0.90%. This was much smaller than the torque ripple of 3.8% with PI control and showed the benefits of using FLC within the model.

Although FLC had given the model a much smoother step response alongside reduced current and torque ripples, the process of implementing the controller in the system was much more complicated than PI control. However, once the FIS had been built, it required minimal effort to transplant the

control system into another EV powertrain with parameter variations. Furthermore, no detailed model of the powertrain systems was required for the development of the FIS.



**Figure 24:** Graphs showing the demanded and achieved torques

## 6 Conclusion and Future Work

This report modelled an 800V EV powertrain utilising a PMSM drive. The control scheme implemented was a vector-based approach, FOC, which initially used PI controllers and then moved to using FLC to compare the torque ripples and the tuning process. The research conducted has shown that FOC is an appropriate control scheme to implement within an 800V EV powertrain, and FLC is an effective method to reduce the torque ripples within the PMSM. FLC has also been shown to have a much simpler tuning process at the cost of a more complex implementation. However, this allowed the control system to be utilised across a range of EV powertrain architectures with similar behaviours, requiring minimal tuning to work effectively.

Although the results shown were positive, the control system was only implemented within an ideal simulation. For greater certainty, and validation of the results produced, implementation within hardware using a Digital Signal Processor (DSP) will be required. The FLC could also be tested against other advanced controllers like one using Model Predictive Control (MPC) to benchmark the results and ensure the best response possible. FLC was also only introduced into the outer-loop torque control, and the two PI controllers within the current controller could also be replaced and tested for improved performance.

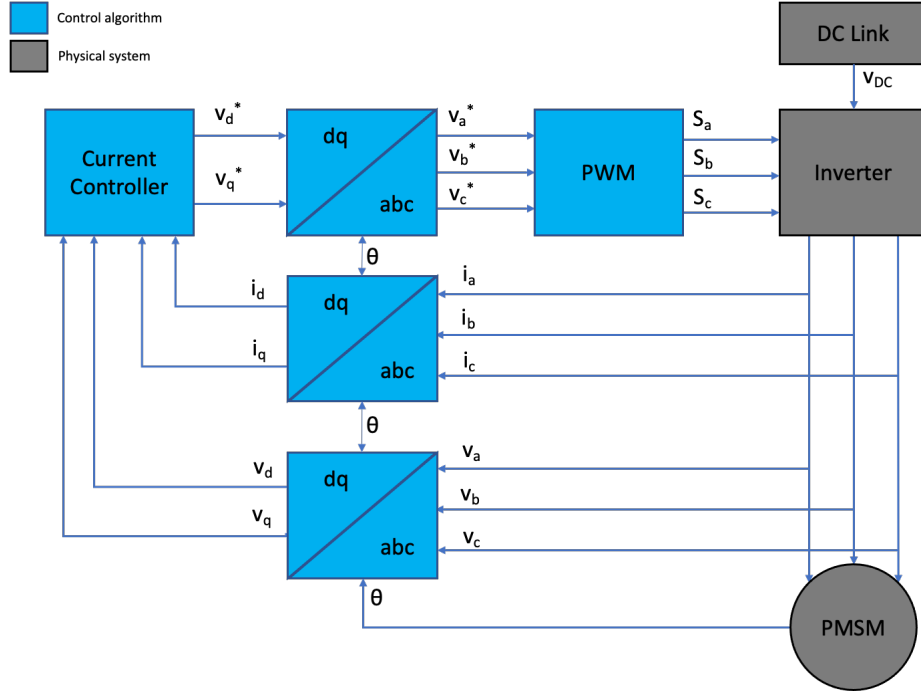
## References

- [1] Matthew Liam De Klerk and Akshay Kumar Saha. A comprehensive review of advanced traction motor control techniques suitable for electric vehicle applications. *IEEE Access*, 9:125080–125108, 2021.
- [2] Abhisek Karki, Sudip Phuyal, Daniel Tuladhar, Subarna Basnet, and Bim Prasad Shrestha. Status of pure electric vehicle power train technology and future prospects. *Applied System Innovation*, 3(3), 2020.
- [3] Department for Business; Energy; & Industrial Strategy. Net zero strategy: Build back greener. Technical report, HM Government, October 2019.
- [4] Lee Chapman. Transport and climate change: a review. *Journal of Transport Geography*, 15(5):354–367, 2007.
- [5] Institute for Advanced Sustainability Studies United Nations Environment Programme, Asian Institute of Technology. Atmospheric brown clouds: Emission inventory manual, 2013. (visited on 29/04/2022).
- [6] Bhupendra Das, Prakash V. Bhave, Siva Praveen Puppala, and Rejina M. Byanju. A global perspective of vehicular emission control policy and practices: an interface with kathmandu valley case, nepal. *Journal of Institute of Science and Technology*, 23(1):76–80, January 2019.
- [7] Ilan Aharon and Alon Kuperman. Topological overview of powertrains for battery-powered vehicles with range extenders. *IEEE Transactions on Power Electronics*, 26(3):868–876, 2011.
- [8] K. T. Chau, C. C. Chan, and Chunhua Liu. Overview of permanent-magnet brushless drives for electric and hybrid electric vehicles. *IEEE Transactions on Industrial Electronics*, 55(6):2246–2257, 2008.
- [9] Juan de Santiago, Hans Bernhoff, Boel Ekergård, Sandra Eriksson, Senad Ferhatovic, Rafael Waters, and Mats Leijon. Electrical motor drivelines in commercial all-electric vehicles: A review. *IEEE Transactions on Vehicular Technology*, 61(2):475–484, 2012.
- [10] Ahmed M. Omara and Michael A. Sleptsov. Performance assessment of battery-powered electric vehicle employing pmsm powertrain system. In *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 963–968, 2017.
- [11] Fuad Un-Noor, Sanjeevikumar Padmanaban, Lucian Mihet-Popa, Mohammad Nurunnabi Mollah, and Eklas Hossain. A comprehensive study of key electric vehicle (ev) components, technologies, challenges, impacts, and future direction of development. *Energies*, 10(8), 2017.
- [12] Audi. Audi e-tron. <https://www.audi.co.uk/uk/web/en/models/e-tron/e-tron.html>. (visited on 29/04/2022).
- [13] Porsche. Porsche taycan 4s. <https://www.porsche.com/uk/models/taycan/taycan-models/taycan-4s/>. (visited on 29/04/2022).
- [14] Kia. Kia ev6. <https://www.kia.com/uk/new-cars/ev6/>. (visited on 29/04/2022).
- [15] Dr. Klaus Specht, Dr. Martin Gall, and Georg Scheidhammer. From 400 to 800v - effects on the high-voltage on-board electrical system. Technical report, Dräxlmaier, October 2018.
- [16] E&T editorial staff. 800v systems in evs enable longer ranges, faster charging. Technical report, The Institution of Engineering and Technology, February 2021.
- [17] S.A. Saleh and M. Azizur Rahman. Introduction to power inverters. <https://catalogimages.wiley.com/images/db/pdf/9780470610480.excerpt.pdf>. (visited on 12/03/2022).
- [18] Alican Madan and Emine Bostanci. Comparison of two-level and three-level npc inverter topologies for a pmsm drive for electric vehicle applications. In *2019 International Aegean Conference on Electrical Machines and Power Electronics (ACEMP) 2019 International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, pages 147–154, 2019.
- [19] Subhansu Satpathy, Subhashish Bhattacharya, and Victor Veliadis. Comprehensive loss analysis of two-level and three-level inverter for electric vehicle using drive cycle models. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pages 2017–2024, 2020.
- [20] Andreas Bubert, Suk-Hyun Lim, and Rik W. De Doncker. Comparison of 2-level b6c and 3-level npc inverter topologies for electric vehicles. In *2017 IEEE Southern Power Electronics Conference (SPEC)*, pages 1–6, 2017.
- [21] Amirreza Poorfakhraei, Mehdi Narimani, and Ali Emadi. A review of multilevel inverter topologies in electric vehicles: Current status and future trends. *IEEE Open Journal of Power Electronics*, 2:155–170, 2021.
- [22] Wesam Taha, Babak Nahid-Mobarakeh, and Jennifer Bauman. Efficiency evaluation of 2l and 3l sic-based traction inverters for 400v and 800v electric vehicle powertrains. In *2021 IEEE Transportation Electrification Conference Expo (ITEC)*, pages 625–632, 2021.
- [23] K. Kumar, Prawin Michael, Joseph John, and S. Kumar. Simulation and comparison of spwm and svpwm control for three phase inverter. *ARPN Journal of Engineering and Applied Sciences*, 5, January 2010.
- [24] Mohamad Issa, Richard Lepage, Mazen Ghandour, H. Ibrahim, Adrian Ilinca, and Karim Ait-Yahia. Integrated a variable frequency drive for a diesel-generating set using the genset-synchro concept. *International Journal of Engineering and Technical Research*, 8:232–239, August 2019.
- [25] Simon Strobl. Space vector modulation (svm). <https://imperix.com/doc/implementation/space-vector-modulation>, August 2021. (visited on 12/03/2022).
- [26] S. Swetha. Simulation and comparison of space vector pulse width modulation for three phase voltage source inverter. *International Journal of Engineering Research and Technology*, 2, May 2013.
- [27] Akhilesh Sharma, Deepak Singh, Pukhrambam Devachandra Singh, and S. Gao. Analysis of sinusoidal pwm and space vector pwm based diode clamped multilevel inverter. In *2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pages 1–6, 2018.

- [28] Yan-jun Yu, Feng Chai, and Shu-kang Cheng. Analysis of modulation pattern and losses in inverter for pmsm drives. In *2008 IEEE Vehicle Power and Propulsion Conference*, pages 1–4, 2008.
- [29] M. Abassi, A. Khlaief, O. Saadaoui, A. Chaari, and M. Boussak. Performance analysis of foc and dtc for pmsm drives using svpwm technique. In *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 228–233, 2015.
- [30] Fatih Korkmaz, İsmail Topaloğlu, M. Faruk Çakir, and Riza Gürbüz. Comparative performance evaluation of foc and dtc controlled pmsm drives. In *4th International Conference on Power Engineering, Energy and Electrical Drives*, pages 705–708, 2013.
- [31] Rachana Garg, Priya Mahajan, Nikita Gupta, and Harsha Saroa. A comparative study between field oriented control and direct torque control of ac traction motor. In *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pages 1–6, 2014.
- [32] Yifang Zhu, Gang Xu, Jian Yin, and Yuan Liu. Speed control of permanent magnet synchronous motor drives based on model predictive control. In *2018 IEEE 3rd International Conference on Cloud Computing and Internet of Things (CCIoT)*, pages 88–93, 2018.
- [33] Guesmi Oussama and Ali Douik. Comparison between pi and fuzzy controllers in speed control of pmsm. In *2017 International Conference on Green Energy Conversion Systems (GECS)*, pages 1–7, 2017.
- [34] Sylvian Sanjuan. Voltage oriented control of three-phase boost pwm converters. Master’s thesis, Chalmers University of Technology, Göteborg, Sweden, 2010.
- [35] MATLAB. Clarke and park transforms. <https://uk.mathworks.com/solutions/power-electronics-control/clarke-and-park-transforms.html>. (visited on 12/03/2022).
- [36] Control Systems Academy. Transformations for motor drive design - clarke. <http://controlsystemsacademy.com/0031/0031.html#magnitude>. (visited on 12/03/2022).
- [37] Xinghao Zhang and Wei Zhang. An improved rotor position estimation in pmsm with low-resolution hall-effect sensors. In *2014 17th International Conference on Electrical Machines and Systems (ICEMS)*, pages 2722–2727, 2014.
- [38] Gustavo Pinares. *On the Analysis of DC Network Dynamics of VSC-Based HVDC Systems*. PhD thesis, Chalmers Energy Area of Advance, May 2014.
- [39] Jozef Sedo and Slavomir Kascak. Control of single-phase grid connected inverter system. In *2016 ELEKTRO*, pages 207–212, 2016.
- [40] Ned Mohan. *Advanced Electric Drives : Analysis, Control, and Modeling Using MATLAB / Simulink*. John Wiley & Sons, Incorporated, 2014.
- [41] Slobodan N. Vukosavic. *Electrical Machines*. Springer, 2013.
- [42] Subha Lakshmi N, Akil Kumar N, Bashkara Sethupathi K, Gowtham M, and Harish Gokul R. Fuzzy logic controller based speed ripple reduction in pmsm. In *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 202–208, 2021.
- [43] Tan Chee Siong, Baharuddin Ismail, Mohd Fayzul Mohammed, Mohd Faridun Naim Tajuddin, Siti Rafidah, Abd. Rahim, and Zainuddin Mat Isa. Study of fuzzy and pi controller for permanent-magnet brushless dc motor drive. In *2010 4th International Power Engineering and Optimization Conference (PEOCO)*, pages 517–521, 2010.
- [44] Farid Benhamida, Slimane Souag, Yacine Salhi, A. Graa, Abdelber Bendaoud, and Mohammed DJE-HAF. A fuzzy logic controller for reactive power and pq bus voltage control. In *2013 3rd International Conference on Electric Power and Energy Conversion Systems, EPECS 2013*, pages 1–6, October 2013.
- [45] Siti Noormiza Mat Isa, Zulkiflie Ibrahim, and Fazli Patkar. Comparative study of fuzzy logic speed controller in vector controlled pmsm drive: Minimum number of fuzzy rule-base. In *2009 Innovative Technologies in Intelligent Systems and Industrial Applications*, pages 112–118, 2009.
- [46] Guanrong Chen and Trung Tat Pham. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*. CRC Press LLC, 2001.

## Appendix A Control Scheme Block Diagrams

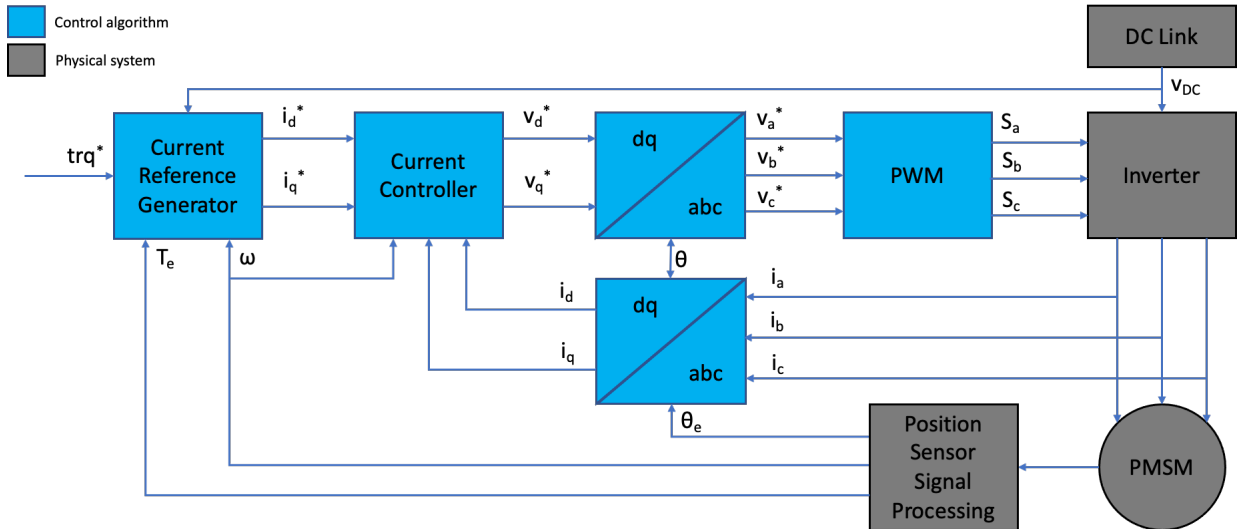
### A.1 Voltage-Oriented Control



**Figure 25:** System block diagram of Voltage-Oriented Control scheme

Fig. 25 shows the block diagram for the VOC scheme. Voltage and current signals taken from the inverter underwent transformation into the  $dq$ -frame. These signals were then fed into the current controller to produce the  $dq$  reference voltages for the PWM. These were transformed back into the  $abc$ -frame before being used to generate PWM signals for the inverter switching elements.

### A.2 Field-Oriented Control

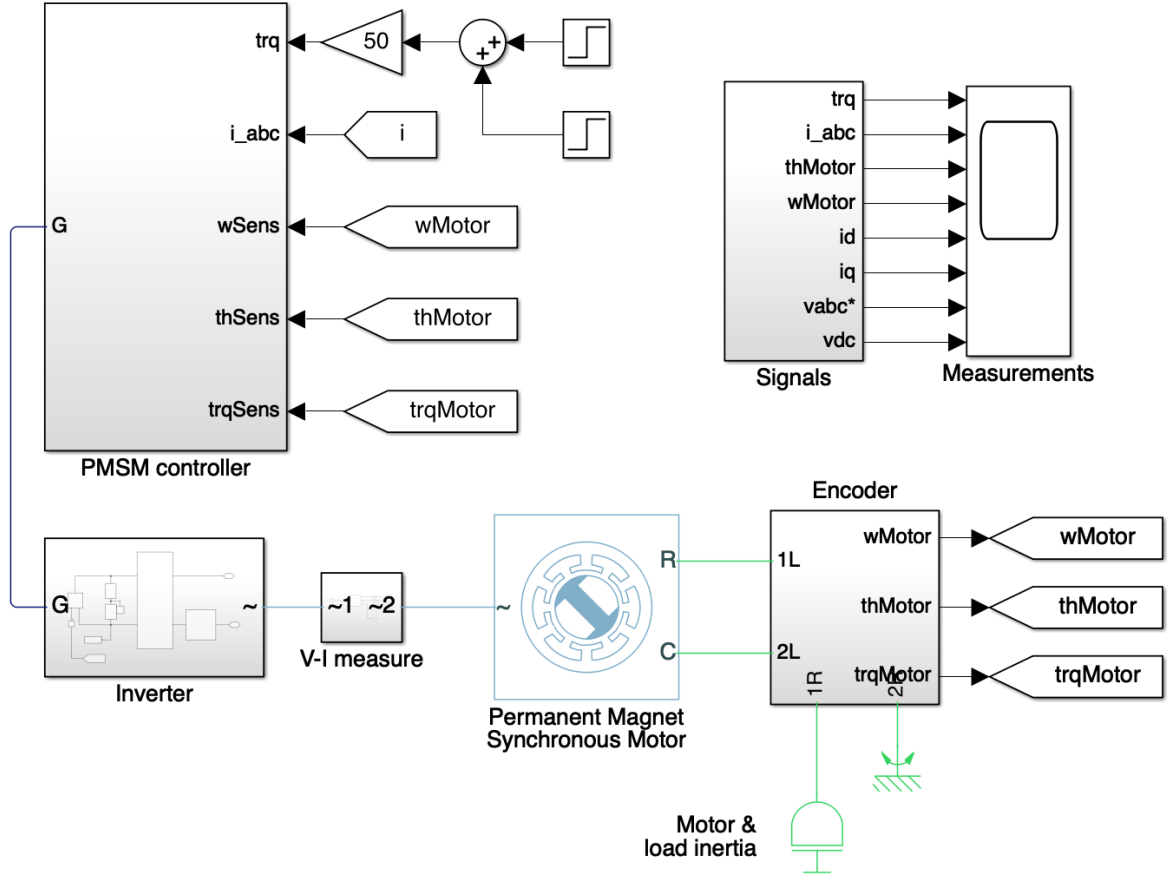


**Figure 26:** System block diagram of Field-Oriented Control scheme

Fig. 26 shows the block diagram for the FOC scheme. Current signals were taken from the inverter and transformed into the  $dq$ -frame before they were fed into the current controller. Rotor position, rotor speed, and electromagnetic torque data was taken from the PMSM model and underwent signal

processing to convert the measured mechanical signals into electrical signals. The torque demand was fed into the outer-loop control with the electromagnetic torque and rotor speed to generate the reference  $dq$  currents. The reference  $dq$  currents were then supplied to the current controller alongside the actual  $dq$  currents and the rotor speed to generate reference  $dq$  voltages for PWM. The  $dq$  voltages were transformed back into the abc-frame and were then used to generate PWM signals for the inverter switching elements.

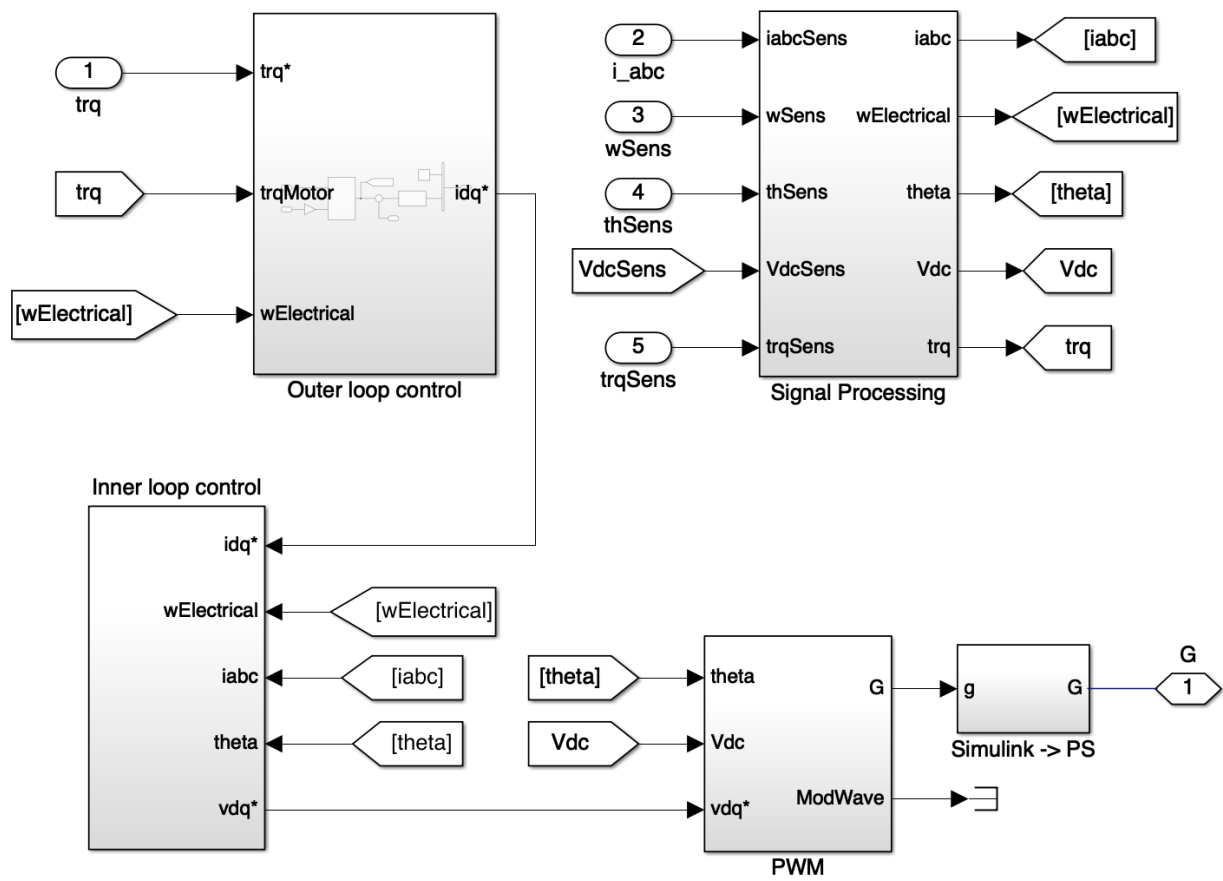
## Appendix B Simulink Model with Field-Oriented Control



**Figure 27:** Simulink model of the EV powertrain utilising FOC and a PMSM drive

Fig. 27 shows the complete Simulink model with the FOC scheme implemented. The encoder at the bottom right measured data from the PMSM, like electromagnetic torque, rotor position, and rotor speed, and passed it into the controller at the top. Within this controller were the outer and inner-loop controls that generate the desired gate signals as discussed previously. These signals were then fed into the inverter which produces the three-phase sinusoidal currents for the PMSM.

Fig. 28 shows the Field-Oriented Control scheme implemented in Simulink. The Signal Processing block took the measured mechanical signals from the PMSM and converted them to electrical signals. These were then passed through the outer-loop control which generated reference  $d$  and  $q$  currents which were fed into the inner-loop control. This then decoupled the PMSM  $dq$  currents before producing the reference  $dq$  voltages to be fed into the PWM. The reference  $dq$  voltages were then converted back to the abc-frame, and compared with the carrier signal to produce gate signals for the inverter.



**Figure 28:** Simulink implementation of Field-Oriented Control

## Appendix C Fuzzy Inference System Codebase

```
% instantiate fuzzy inference system
fis = mamfis;

% add inputs / outputs
fis = addInput(fis, [-2 2], 'Name', 'e'); % error
fis = addInput(fis, [-2 2], 'Name', 'ce'); % change in error
fis = addOutput(fis, [-2 2], 'Name', 'output'); % output

% add membership functions
fis = addMF(fis, 'e', 'trapmf', [-5 -2 -1 0], 'Name', 'N');
fis = addMF(fis, 'e', 'trimf', [-1 0 1], 'Name', 'Z');
fis = addMF(fis, 'e', 'trapmf', [0 1 2 5], 'Name', 'P');

fis = addMF(fis, 'ce', 'trapmf', [-5 -2 -1 0], 'Name', 'N');
fis = addMF(fis, 'ce', 'trimf', [-1 0 1], 'Name', 'Z');
fis = addMF(fis, 'ce', 'trapmf', [0 1 2 5], 'Name', 'P');

fis = addMF(fis, 'output', 'trapmf', [-5 -2 -1 0], 'Name', 'N');
fis = addMF(fis, 'output', 'trimf', [-1 0 1], 'Name', 'Z');
fis = addMF(fis, 'output', 'trapmf', [0 1 2 5], 'Name', 'P');

% specify rules in matrix form
rule1 = [1 1 1 1 1;
         2 1 1 1 1;
         3 1 2 1 1];

rule2 = [1 2 1 1 1;
         2 2 2 1 1;
         3 2 3 1 1];

rule3 = [1 3 2 1 1;
         2 3 3 1 1;
         3 3 3 1 1];

rules = [rule1; rule2; rule3];

% add rules
fis = addRule(fis, rules);
```