Real-time Ultra Short Horizon extention MPC for the control of a Formula Student Drive

Israel Sother

israelsother@usp.br

Instituto Superior Técninco, Lisboa, Portugal

Abstract—This work presents the development of a control strategy for the motors of the Formula Student Team of Instituto Superior Técnico. The control strategy was developed with the main goal of improving the dynamic response of the motor torque and the system efficiency. The motor was experimentally characterized and a simulation environment was created to test different control strategies. The explicit Model Predictive Controller (MPC) was selected as the most suitable strategy due to its fast response and low current ripple. A novel horizon extention technique was proposed to reduce model mismatch, which gave the name of the controller: Real-time Ultra Short Horizon extention MPC (RUSH MPC). The control strategy was implemented in a Field Gate Programmable Array (FPGA) and a test bench setup was developed. The experimental results showed a close match with the simulation results, validating the torque estimation, the current measurements, and the control strategy. The system efficiency was improved due to the reduction in current Total Harmonic Distortion (THD) and the torque response was improved by orders of magnitude compared to the current solution.

Index Terms—Explicit MPC, Motor Control, Electric Vehicle, Formula Student, Horizon Extension.

I. INTRODUCTION

With the increasing regulation efforts to reduce carbon footprints, an upward trend of investments in the mobility sector for the development of electric and hybrid powertrains has emerged. This field has received considerable interest for not only hardware improvements [1] but also new software alternatives with several new control strategies being proposed. From one side the advancing processing power available in microcontrollers has enabled the use of real-time predictive control strategies [2], while the use of wide bandgap semiconductors results in a substantial efficiency improvement [3].

Formula Student is an engineering competition that challenges students to design, manufacture, and test a formula-style race car inside a given set of regulations. Similarly to the industry trend, the competition has pushed teams towards electrification, encouraging students to seek solutions that are lightweight, powerful, and efficient. The Formula Student Team of Técnico Lisboa (FST Lisboa) was funded in 2001, and since then has built 12 prototypes, from the fourth model (FST04) onward they have an electric powertrain, with the last 3 having autonomous racing mode.

A major advance in the power converter field was the use of wide bandgap semiconductors, which has allowed the development of more efficient converters with a higher power density. As a relatively new technology, there aren't many off-the-shelf solutions that fulfill the specifications required for a Formula Student prototype, thus some of the top teams started to develop their own motor drive solutions. FST started working on a fully self-developed powertrain in 2017 with motor development by Sarrico [4], and inverter development by Costa [5]. This development aligns with the fundamental objectives of Formula Student, which is empowering technical and practical knowledge to better prepare students. Additionally, the development of the entire powertrain system can result in a more efficient platform, creating a system optimized for each prototype. Unfortunately, the motor prototype is not ready to be used in the car, and the inverter is missing a system able to control the currently used motors. This work intends to fill a critical gap which is the absence of adequate control for a Permanent Magnet Synchronous Motor (PMSM). By developing and implementing a control strategy that enables the use of an in-house developed inverter with the commercial motor solution, this work not only addresses the identified bottleneck in the powertrain section but also shifts the performance envelope standards, improving the overall capabilities of the next prototypes.

A. Topic Overview

Through the years, several approaches have been proposed to control synchronous machines using a 2-level three-phase inverter. Regarding the control methods, several solutions have been presented in the literature with the most common being Field Oriented Control (FOC) and Direct Torque Control (DTC), with MDC [6] being introduced more recently, seeking a compromise between the advantages of FOC and DTC. Some approaches require a modulator such as Space Vector Modulation (SVM) [7], while others can directly output the semiconductor trigger signals [8].

The use of FOC and DTC have dominated the market for many years due to their simple implementation, but each of them has downsides. FOC combined with current control PID is known for having a slow dynamic response when compared to DTC, while in steady-state behavior and disturbance rejection, FOC shows better results with smaller torque ripple and current THD [9–11]. MPC has been proposed as a combination of the two, exhibiting a good dynamic response while being able to maintain low current and torque ripple.

Technical advances in the microprocessor industry have allowed the increasing use of MPC in power converters and drives [8]. This non-linear control scheme provides easy constraint integration while optimizing the control action in real time and is adaptable to different types of electric machines being controlled. The main downside of this strategy is the increased computational cost, which is offset by the decreased cost of computational power.

MPC in power systems is usually divided into two categories: Continuous Set MPC (CSMPC) and Finite Set MPC (FSMPC) [12], depending on their output signals. The first type calculates the best possible voltage vector and then uses a modulator like SVM to compose it, while FSMPC exploits the fact that a motor drive usually has a limited number of possible voltage vectors and thus predicts the currents for each of those vectors to evaluate the best option. The second approach usually doesn't need a modulator as it only considers the finite set of all possible converter states, although some variations have tried to increase the search space by introducing synthetic vectors that are created from a combination of the native ones. This technique of subdivision and refining vectors can be used to create a quasi-continuous set MPC [13], which, when combined with reducing the

search area to the most probable sector, can improve the torque and current ripple without a prohibitive increase in computational cost.

II. TEORETICAL BACKGROUNG AND PMSM MODEL

This section briefly introduces the Formula Student competition structure and the general powertrain model used in this work.

A. Formula Student

In a formula student competition, two types of evaluation exist, the first category is comprised of static events where the design, cost, and business model of the prototype are analyzed. In the dynamic category, each prototype is evaluated through 5 different events: Skidpad, Acceleration, AutoX, Endurance, and Efficiency (evaluated in the Endurance track). The skidpad event consists of two circles of radius 9.125m, where the vehicle follows an eight pattern. The acceleration event consists of a standing start 75m straight acceleration, with the maximum battery power limited to 80kw, as it is for all Formula Student events. The AutoX is a 1km track with several corners and straights mixed. The endurance event is similar to the AutoX, with enough laps to complete 22km. Lastly, the efficiency event is evaluated in the Endurance track, where the teams are scored based on the energy consumed to complete the 22km track and the time it took to complete it.

B. Two Level Voltage Source Inverter

The usual hardware used to control synchronous machines is a 2-level Voltage Source Inverter. Such equipment is composed of six switches organized in three legs, where each pair of switches is connected to a motor terminal, as shown in Figure 1.
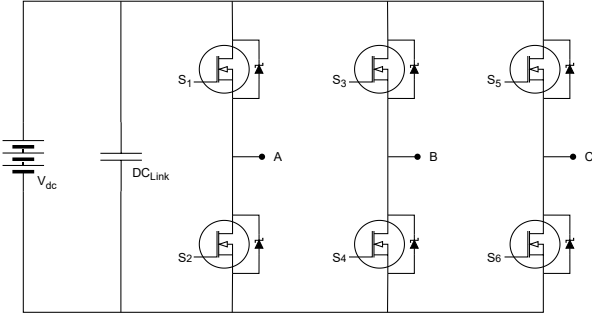

Fig. 1: 2-Level Voltage source Inverter arrangement.

Usually, each switch in an inverter leg is operated with the inverse logic of the other switch in the leg, and this arrangement allows for 8 different switching combinations where 2 of them result in null voltages. That gives 7 possible voltage vectors, as detailed in Table I and Figure 2. In Table I the Vector column denotes the top switches states (S_1, S_2, and S_3), where a 1 means the top switch is the conducting state with the bottom switch is on a cut-off state and a 0 the opposite. The α and β components which define the state space vectors are as defined by the Concordia transformation.

C. Space Vector Modulation

Several modulation techniques have been proposed in the literature like Sinusoidal Pulse Width Modulation (SPWM), Selective Harmonic Elimination (SHE) [14], SVM [7]. The most common method of modulation in digital motor control is SVM, as it is a robust, easy-to-implement technique, and allows higher voltage ratio.

Figure 2 shows a 2D representation of the space vector for a 2-level three-phase inverter. The basic voltage vectors (previously defined on

TABLE I: Space vector for a 2-level three-phase inverter

Switch State	Vector	V_α	V_β
0	000	0	0
1	001	$\sqrt{\frac{2}{3}}V_{DC}$	0
2	010	$\frac{1}{\sqrt{6}}V_{DC}$	$\frac{1}{\sqrt{2}}V_{DC}$
3	011	$-\frac{1}{\sqrt{6}}V_{DC}$	$\frac{1}{\sqrt{2}}V_{DC}$
4	100	$-\sqrt{\frac{2}{3}}V_{DC}$	0
5	101	$-\frac{1}{\sqrt{6}}V_{DC}$	$\frac{1}{\sqrt{2}}V_{DC}$
6	110	$\frac{1}{\sqrt{6}}V_{DC}$	$\frac{1}{\sqrt{2}}V_{DC}$
7	111	0	0

Table I) are shown pointing to the hexagon corners. Connecting the basic vectors produces the hexagon of possible voltage states.
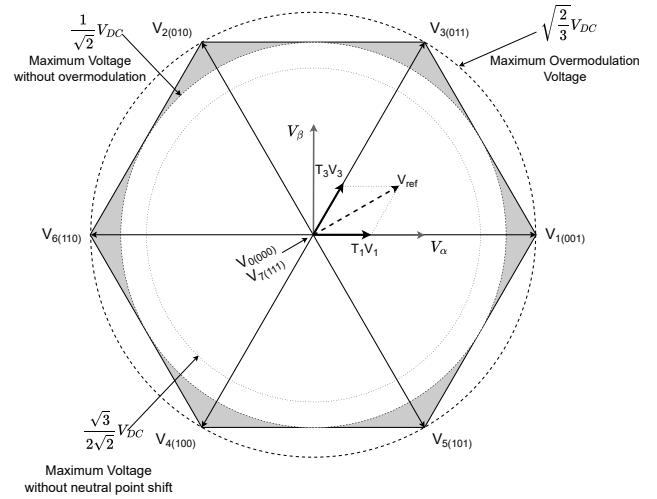

Fig. 2: Space vector for a 2-level three-phase inverter.

Using SVM it is possible to modulate any vector inside the hexagon shown in Figure 2, but if a pure sinusoidal output is desired, the vectors should be constrained to the inscribed circle, that has a radius of $\frac{1}{\sqrt{2}}V_{DC}$. The reason behind this is to keep the reference vector locus inside the hexagon, avoiding distortions in the output. Note that although it is possible to generate waveforms with higher RMS value, it is not possible to modulate a peak higher than $\frac{1}{\sqrt{2}}V_{DC}$ for every vector angle. If a higher RMS value is requested the generated voltage will be saturated on the sides of the hexagon while near the corners it will achieve the requested value, thus those waveforms become more and more distorted as the amplitude approaches $\sqrt{\frac{2}{3}}V_{DC}$. To modulate a sinusoidal output the vector should develop a circular trajectory, but in the overmodulation region the voltage constrains it to the voltage hexagon, and thus the difference between the intended and the effectively applied vector increases.

To modulate a voltage vector that is not exactly one of the basic voltage vectors a modulation technique is needed. When using the SVM method to compose a given reference vector V_{ref}, the algorithm first detects the sector on which the reference vector lays. With the sector identified a ratio between the adjacent basic vectors and a null

vector is selected so that the average vector is equal to the reference vector. This ratio is calculated as shown in Equation (1).

For example, let's consider the reference vector shown in Figure 2. According to SVM, this vector can be modulated by using V_1 for half of the active time and V_3 for the other half of the active time. The active time refers to the duration when the vector amplitude is non-zero, while the null time refers to the duration when a null vector is used to reduce the output amplitude. This modulation can be expressed by Equation (1), where $T_1 + T_3$ represents the active time and T_{null} represents the null time.

$$V_{ref} = \frac{T_{null}V_{null} + T_1V_1 + T_3V_3}{h} \tag{1}$$

D. PMSM model

A model to represent the PMSM with delta-arranged windings is needed to develop the controller. The used model is a 3-phase model in a rotor reference frame (dq0), with permanent magnets on the rotor in a spoke arrangement. The model equations are shown in Equation (2), where u_d and u_q are the d and q components of the stator voltage, r is the stator phase resistance, L_d and L_q are the d and q components of the stator inductance, i_d and i_q are the d and q components of the stator phase current, ω_e is the electrical angular velocity, ψ_{PM} is the permanent magnet flux, T_e is the electromagnetic torque, T_{load} is the load torque, T_{loss} is the total losses torque, and J is the moment of inertia of the rotor. Note that the inductances are assumed to be a function of the current in the respective DQ0 component, this allows the model to account for saturation effects. This could be used to model also the cross-magnetization effect, but limitations in the implementation of the control strategy prevented this from being used.

$$u_d = ri_d + L_d\frac{di_d}{dt} - \omega_e L_q i_q \tag{2a}$$

$$u_q = ri_q + L_q\frac{di_q}{dt} + \omega_e(L_d i_d + \psi_{PM}) \tag{2b}$$

$$T_e = p\,i_q((L_d - L_q)i_d + \psi_{PM}) \tag{2c}$$

$$\frac{d\omega_e}{dt} = \frac{T_e - T_{load} - T_{loss}}{J} \tag{2d}$$

$$\frac{d\theta_e}{dt} = \omega_e \tag{2e}$$

An approximation of the differential equations is needed to discretize the equations to be able to use the model in a discrete time control system. Two usual solutions are the Backward and the Forward Euler methods. Although simpler to compute, the Forward Euler method is prone to instabilities especially in fast systems such as power converters, in contrast with the Backward alternative that is unconditionally stable. Due to this consideration, the Euler Backward technique will be prioritized. Applying this method to Equation (2) would result in an equation that has algebraic loops so to avoid that a few approximations were made. First, the inductance change due to current variation in a timestep h is assumed to be small enough so that the inductances can be calculated using the previous time step currents ($L_{x(i_x(k+1))} \approx L_{x(i_x(k))}$). Similarly, the rotor speed is assumed to vary little between iterations, so that $\omega_{(k+1)} \approx \omega_{(k)}$.

With those considerations, and rearranging the equations, the system can be solved iteratively, as shown in Equation (3).

$$\begin{bmatrix}i_{d(k+1)} \\ i_{q(k+1)}\end{bmatrix} = \begin{bmatrix} 1 & -h\frac{\omega_{e(k)}L_{q(i_q(k))}}{hr+L_{d(i_d(k))}} \\ h\frac{\omega_{e(k)}L_{d(i_d(k))}}{hr+L_{q(i_q(k))}} & 1 \end{bmatrix}^{-1}$$
$$\left(\begin{bmatrix} \frac{L_{d(i_d(k))}}{hr+L_{d(i_d(k))}} & 0 \\ 0 & \frac{L_{q(i_q(k))}}{hr+L_{q(i_q(k))}} \end{bmatrix} \begin{bmatrix}i_{d(k)} \\ i_{q(k)}\end{bmatrix} + \right.$$
$$+ \begin{bmatrix} \frac{h}{hr+L_{d(i_d(k))}} & 0 \\ 0 & \frac{h}{hr+L_{q(i_q(k))}} \end{bmatrix} \begin{bmatrix}u_{d(k+1)} \\ u_{q(k+1)}\end{bmatrix} +$$
$$\left. + \begin{bmatrix} 0 \\ -h\frac{\omega_{e(k)}\psi_{PM}}{hr+L_{q(i_q(k))}} \end{bmatrix} \right) \tag{3a}$$

$$T_{e(k+1)} = p\,i_{q(k+1)}((L_{d(i_d(k+1))} - L_{q(i_q(k+1))})i_{d(k+1)} + \psi_{PM}) \tag{3b}$$

$$\begin{bmatrix}\omega_{e(k+1)} \\ \theta_{e(k+1)}\end{bmatrix} = \begin{bmatrix}1 & 0 \\ h & 1\end{bmatrix}\begin{bmatrix}\omega_{e(k)} \\ \theta_{e(k)}\end{bmatrix} +$$
$$+ \frac{T_{e(k+1)} - T_{load(k+1)} - T_{loss(k+1)}}{J}\begin{bmatrix}h \\ h^2\end{bmatrix} \tag{3c}$$

III. MOTOR CHARACTERIZATION

Although AMK has provided the motor datasheet with the key parameters on it, it is important to verify how well they track the real values, and how they change regarding the motor operation. The datasheet values are linear approximations of the magnetic circuit on the motor, and as such do not accurately represent the machine at high current operating points. To account for the saturation effect, a variable inductance approach will be used [15, 16], where the inductance of each axis will be a function of the current on the respective axis.

A. Phase Resistance

Assuming the machine is well balanced, the phase resistance will be equal to $\frac{3}{2}$ of the terminal resistance. The phase resistance was measured using a micro-ohmmeter (*UNI-T UT620A Micro-ohmmeter*), that has a resolution of $10\mu\Omega$ with an accuracy of $\pm(0.25\% + 25\mu\Omega)$. The measurements were taken on the terminal wires at room temperature and with the kelvin probe, resulting in $142.89 \pm 0.06m\Omega$. This value is close to the datasheet one of $135m\Omega$, and the difference is probably due to the wire terminals. The resultant phase resistance (r) is $214.335 \pm 0.091m\Omega$.

B. Flux Linkage

The flux linkage was measured by connecting the rotor to an external motor that kept it at a constant known speed without any connection between the terminals, this ensures the current is zero as in Equation (4). Then an oscilloscope (*Promax OD-571*) was connected to two terminal wires and used to measure the back EMF. The oscilloscope on the used settings has an accuracy of $\pm(3\% + 0.2508V)$. This was repeated for several speeds, resulting in Figure 3.

$$u_d = r\cancel{i_d}^{\,0} + \cancel{\frac{di_d}{dt}}^{\,0}\left(L_d + \cancel{i_d}^{\,0}\frac{dL_d}{di_d}\right) - \omega_e L_q\cancel{i_q}^{\,0} = 0 \tag{4a}$$

$$u_q = r\cancel{i_q}^{\,0} + \cancel{\frac{di_q}{dt}}^{\,0}\left(L_q + \cancel{i_q}^{\,0}\frac{dL_q}{di_q}\right) + \omega_e(L_d\cancel{i_d}^{\,0} + \psi_{PM}) = \omega_e\psi_{PM} \tag{4b}$$
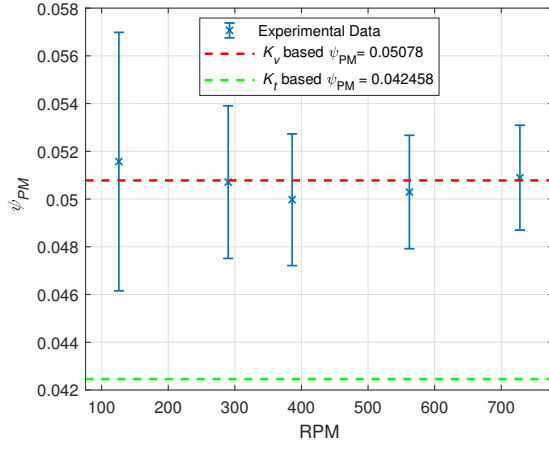
Fig. 3: Flux Linkage as a function of RPM.

(a) Flux Linkage d axis exponential fit.

(b) Flux Linkage q axis exponential fit.

Fig. 4: Flux Linkage exponential fit. The blue dots are the measured flux linkage, while the red line is the exponential fit (R-square 0.8879 for ψ_d and 0.9681 for ψ_q).

(a) Machine Inductance in d axis.

(b) Machine Inductance in q axis.

Fig. 5: Machine Inductance as the current derivative of the flux linkage.

In this graph, two other lines are shown, they represent the flux linkage estimated using the torque and speed constant provided in the datasheet. Notice that the experimental data aligns well with the value derived from the voltage constant but it is very different from the value derived from the torque constant. This is probably due to different references and transformations being used, but as there isn't much information available on the datasheet, the measured value is assumed to be the correct one.

C. Machine direct and quadrature inductances

The inductances were characterized according to Stumberger et al. [16]. This method uses a Voltage Source Inverter coupled with a control algorithm to keep the current in one of the axes constant, and then do a voltage step in the other axis. The current is then integrated to obtain the current-induced flux linkage as shown in Equation (6), and then the inductance is calculated as the derivative of the current-induced flux linkage.

The flux-induced curve is then approximated as a fit of several exponential curves, as this allows the inductance to be calculated as the analytical current derivative of this exponential fit. The current induced flux linkage and its exponential fit are shown in **??**, and the inductance presented in **??**.
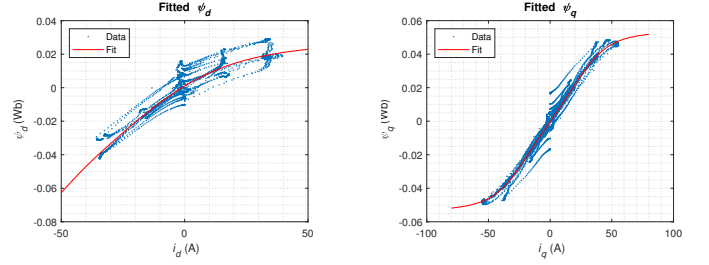
$$u_d = ri_d(t) + \frac{d\psi_d(t)}{dt} - \psi_q\omega_e^{\,\,0} \tag{5a}$$

$$u_q = ri_q(t) + \frac{d\psi_q(t)}{dt} + (\psi_d + \psi_{PM})\omega_e^{\,\,0} \tag{5b}$$

$$\frac{d\psi_d(t)}{dt} = u_d - ri_d(t) \tag{6a}$$

$$\frac{d\psi_q(t)}{dt} = u_q - ri_q(t) \tag{6b}$$

IV. CURRENT REFERENCES

To simplify the real-time computations, the current references can be calculated offline. A simple approach would be to only consider the quadrature current i_q as the main component of torque, and compute the reference using the torque constant. This approach is simple and fast, but it does not account for the motor's inductance, which can be used to increase efficiency. The maximum torque per ampere strategy actively uses the inductance differences between the direct
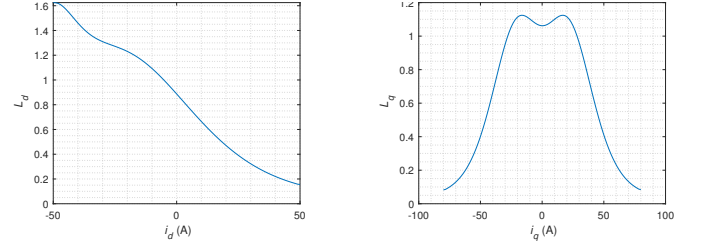
and quadrature axes to get the maximum torque for a given current, but as the inductances are variable, the optimal current reference is also variable. Although it is possible to use the MPC to optimize the current references, doing it offline not only allows for faster computational times but also results in more precise references. The downside of this approach is that it negates the possibility of acting upon online parameter estimation, however, this drawback can be mitigated by implementing regular calibration procedures.

Using Equation (2) and assuming a steady state, a constrained minimization problem can be written to minimize the current modulus with constraints to ensure the reference torque is achieved and to account for back EMF related voltage limitations. This problem is presented in Equation (7).

$$\min_{i_d, i_q} \quad \sqrt{i_q^2 + i_d^2}$$
$$\text{s.t.} \quad T_{ref} = p\, i_q((L_d - L_q)i_d + \psi_{PM}) \tag{7}$$
$$(ri_d - \omega_e\psi_q)^2 + (ri_q + \omega_e(\psi_{PM} + i_d L_d))^2 \leq V_{DC}^2$$

As the solution to this problem is not trivial, the optimization was solved using a numerical approach in *MATLAB*. The resultant current references for the nominal battery voltage are shown in Figure 6.

V. PROPOSED CONTROL STRATEGY

The proposed control strategy is a Model Predictive Control (MPC) with a horizon of 1 timestep, that uses **??** to evaluate the best possible voltage vector to apply to the motor. One of the advantages of the approximations made in the discretization process is that even though the matrices change in time, for a given moment the system is linear, and as such an inverse dynamic can be derived. So, if in Equation (3) the currents on the next step are replaced by a reference for the current vector, the necessary applied voltage can be calculated:
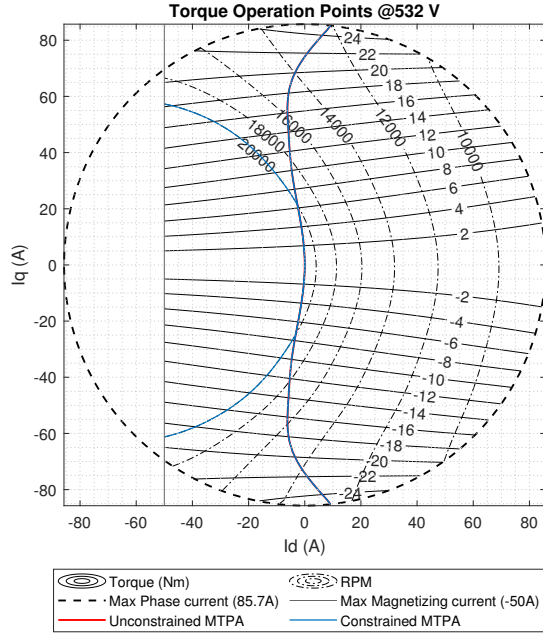
4

Fig. 6: Constrained Maximum Torque per Ampere curve at nominal battery voltage. Reference currents are shown for 0, 13000 and 2000 RPMs.

$$\begin{bmatrix} u_{d(k+1)} \\ u_{q(k+1)} \end{bmatrix} = \begin{bmatrix} \frac{h}{hr+L_{d(i_d(k))}} & 0 \\ 0 & \frac{h}{hr+L_{q(i_q(k))}} \end{bmatrix}^{-1}$$
$$\left(\begin{bmatrix} 1 & -h\frac{\omega_{e(k)}L_{q(i_q(k))}}{hr+L_{d(i_d(k))}} \\ h\frac{\omega_{e(k)}L_{d(i_d(k))}}{hr+L_{q(i_q(k))}} & 1 \end{bmatrix} \begin{bmatrix} i_{d_{ref}} \\ i_{q_{ref}} \end{bmatrix} - \right.$$
$$- \begin{bmatrix} \frac{L_{d(i_d(k))}}{hr+L_{d(i_d(k))}} & 0 \\ 0 & \frac{L_{q(i_q(k))}}{hr+L_{q(i_q(k))}} \end{bmatrix} \begin{bmatrix} i_{d(k)} \\ i_{q(k)} \end{bmatrix} -$$
$$\left. - \begin{bmatrix} 0 \\ -h\frac{\omega_{e(k)}\psi_{PM}}{hr+L_{q(i_q(k))}} \end{bmatrix} \right) \qquad (8)$$

This approach is commonly used in linear unconstrained MPCs to improve computation times, where a control law is precomputed and stored in memory. But in the presented case there is an important constraint that is not accounted for in the previous equation. When very short time steps are used, and there is a big reference change, the necessary voltage to achieve the target on just one discrete time step can be very high. This would lead to problems, like distortions due to overmodulation or not being able to reach the desired voltages. The solution to this problem is to saturate the voltage vector. Some authors have suggested saturating based on the closest possible vector [17], in this work, the saturation only limits the vector amplitude to match the DC-link voltage, maintaining the desired vector angle as shown in Equation (9).

$$\gamma = arctg\left(\frac{u_{d(k+1)}}{u_{q(k+1)}}\right) \qquad (9a)$$

$$u_{sat_{d(k+1)}} = \min\left(u_{d(k+1)} , V_{dc}\cos(\gamma)\right) \qquad (9b)$$

$$u_{sat_{q(k+1)}} = \min\left(u_{q(k+1)} , V_{dc}\sin(\gamma)\right) \qquad (9c)$$

This method greatly reduces the time taken to compute the control action, as instead of trying 7 different possible inputs (or more) it just calculates one time the necessary voltage and clamps it to the attainable values. As this control uses a continuous set, those values are then forwarded to an SVM system to modulate the Pulse Width Modulation (PWM) signals.

VI. HORIZON EXTENSION

A previously overlooked problem is the compensation of the computational time, as the acquisition and control calculations cannot be done instantly their delay needs to be compensated. The strategy adopted here is to use the motor model equations Equation (3) coupled with the previously calculated control action to predict the system state in the next time step. Usually, this is done in a fixed timestep manner, where the predictive controllers instead of picking the control action for k in the timestep k, pick the control action for $k+1$ in the timestep k, as shown in Equation (10) where x represent the currents, u is the voltage vector, A, B, C, and D are the matrices and vectors of the model in Equation (3) and are all dependent on the prediction duration $h = \frac{1}{f_{sw}}$.

$$x_{k+1} = A^{-1}\left(Bx_k + Cu_k + D\right) \qquad (10)$$

For this equation to work the system timeline needs to be as in Figure 7, starting with taking the current and voltage measurements, followed by applying the voltage vectors calculated in $k-1$, then extending the horizon by one timestep, and calculating the voltage vector for the next timestep [18].

This can be improved by only extending the horizon by the necessary time to compute the control action, this way the prediction error derived from model mismatch is reduced because the amount of time to predict is smaller. To do that the prediction duration is simply reduced to $h = t_{control}$, while the system timeline is shifted as in Figure 8. The combination of explicitly solving the MPC model equation with the hybrid horizon timestep approach, where the horizon extension has a shorter timestep than the controllable horizon is called Real-time Ultra Short Horizon extension MPC (RUSH MPC). It has the benefits of being a continuous set model predictive control, thus reducing the ripple, while being computationally efficient, as it is explicitly calculated, and reducing the prediction error due to model mismatch as it uses the reduced horizon extention technique.

The final control cycle starts with sensor acquisition, followed by horizon extension, and lastly the future control action computation. Note that this delay is not a problem with simulation in *Simulink*, as it can instantly do the calculations, but it is good practice to simulate it with the proper delays to increase the similarity between simulation and experimental results.

VII. SIMULATION

To accelerate the development time, a model was developed in *Simulink*, allowing for faster prototyping and direct comparison of the methods in a controlled environment. An adaptation of the *Simscape Specialized Power Systems* block *Permanent Magnet Synchronous Machine* was made to convert it to a delta-wound machine and to accept variable inductances as in Equation (3). This model was combined with six MOSFET blocks in three legs to simulate the VSI. To supply the MOSFETs, a model of the battery was made, where an ideal voltage source is connected through a series resistor to the DC link, where a capacitor stabilizes the voltage. In Figure 9 the resultant system is presented, where S_1, S_2, S_3, S_4, S_5, S_6, are the gate control signals that come from the control strategy.
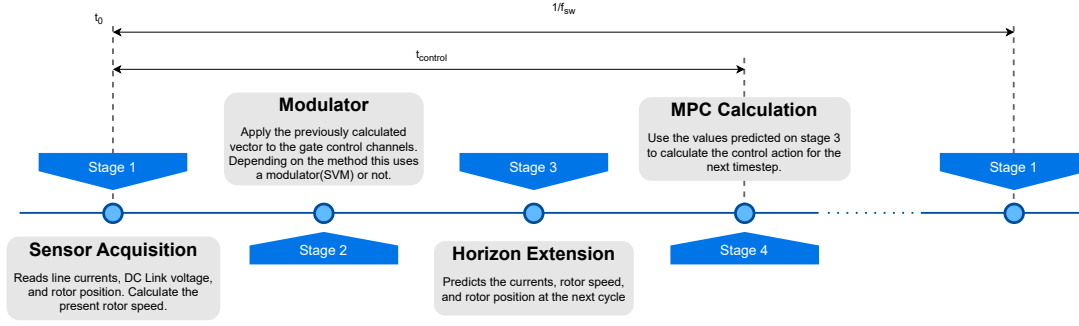
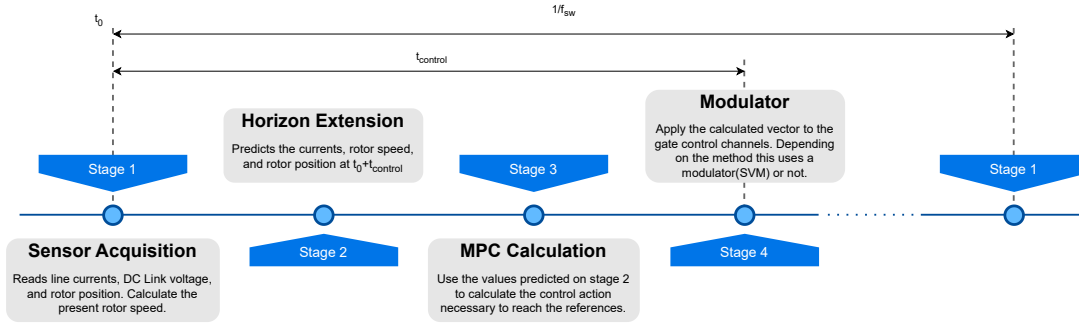Fig. 7: Horizon extention by one timestep.

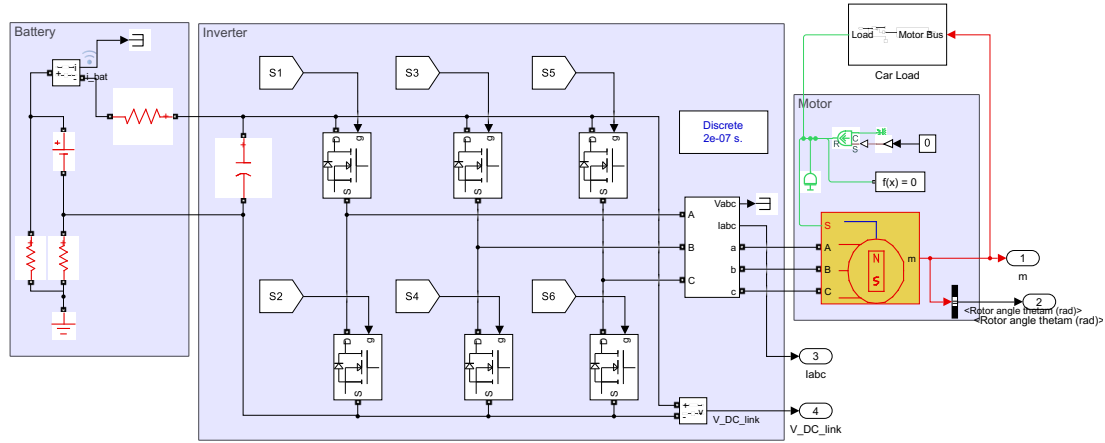Fig. 8: Horizon Extension by control time.

Fig. 9: Simulink Models, Motor and Inverter.

A. Baseline Step

With the model implemented on *Simulink*, a baseline was made using the manufacturer control scheme, FOC with the standard $8kHz$ switching frequency, and compared with the proposed methods at $50kHz$. The difference in frequency is to account for the complete system, where the use of wide bandgap semiconductors allowed for a faster switching frequency. The baseline profile is a positive torque step with the machine fixed at the nominal speed of $12kRPM$. The rising time of each method is shown in Figure 10.

Note that the predictive controller presents a rising time (0 to 100%) of $200\mu s$, while the FOC method performed much slower, at approximately $180ms$. While the rising time in the FOC can

be improved by better tunning the PID (these results are with the values recommended by the manufacturer), it would result in more pronounced overshoots and settling time, and would not reach the performance of a predictive controller, where the torque rising is only limited by the machine inductances, which shows the great dynamic performance of predictive controllers.

B. Current THD

The distortion of currents was evaluated by setting the motor at a constant speed, waiting a few periods for it to reach a steady state and then calculating the average THD of all line currents through 5 electrical periods. This was repeated in a grid pattern with some of the results shown in Table II.
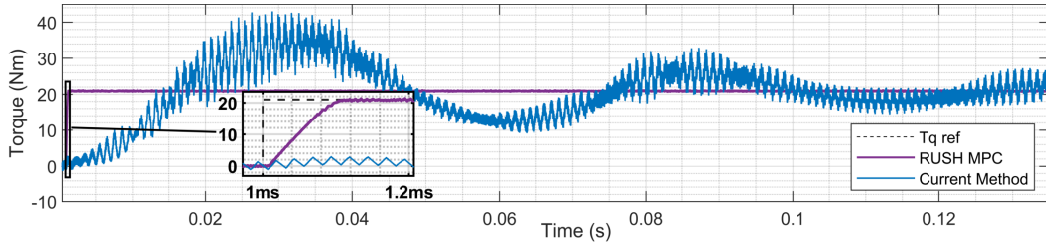
6

Fig. 10: Control methods comparison in a torque step. Proposed method (in purple) vs Currently used method (in blue).

TABLE II: Control Method Current THD comparison. THD is calculated until the Nyquist frequency, that is $250kHz$ and $40kHz$ for the controllers running at $50kHz$ and $8kHz$ respectively.

		1000 RPM	7333 RPM	13666 RPM	20000 RPM
FOC @8kHz	1Nm	57.62%	43.44%	207.54%	29.62%
	11Nm	3.22%	10.62%	12.14%	58.53%
	20Nm	5.15%	6.50%	10.18%	37.08%
FOC @50kHz	1Nm	56.95%	11.08%	14.12%	4.71%
	11Nm	2.18%	2.19%	1.93%	4.85%
	20Nm	1.41%	1.41%	2.28%	3.86%
RUSH MPC @50kHz	1Nm	7.20%	11.91%	16.82%	21.95%
	11Nm	0.76%	1.47%	1.92%	2.16%
	20Nm	0.81%	0.98%	1.19%	1.12%

Table II exposes the advantage of continuous MPC over the other alternatives, with distortions expressively smaller than the baseline FOC. Another advantage of the proposed method is that it actively uses the direct axis current to generate torque throughout the full operation map of the motor, not only in field weakening as the method currently implemented on the car. This is a result of the use of Maximum Torque per Apere references which improves efficiency, as it produces a reduction in the current vector modulus necessary to generate the same torque.

C. Robustness

To test the controller's robustness, a Monte Carlo analysis is proposed. A normal distribution is assigned to the motor parameters with a mean equal to the characterized value, and a standard deviation equal to 5% of the mean. The step of Figure 10 was simulated for 1000 samples with the modified parameters applied to the motor without updating the controller (Figure 11).

To evaluate the performance with degraded parameters three metrics were used, the steady-state mean torque, the steady-state current THD, and the torque rising time. Based on these metrics the most influential parameter is the flux linkage, with a correlation of almost 100% with the steady-state torque. This is explained by the type of motor used, where the inductances between the direct and quadrature axes are very similar, resulting in most of the torque derived from the permanent magnet's flux linkage. Note that this dependency is also present in the method currently used by the manufacturer, where the quadrature current reference is based on the machine torque constant (k_t). While relatively easy to characterize, the flux linkage is also the parameter most prone to change with the life of the motors, where the magnets can overheat or be demagnetized by a high direct axis current, thus a good approach is to develop a characterization routine that can be automatically executed whenever the user deems necessary.

The quadrature inductance has a higher correlation with the torque rise time, as expected. Since almost all of the torque is derived from

the permanent magnet's flux linkage the torque will rise as fast as the controller manages to create quadrature currents. While this correlation is not definitive proof, it reinforces the statement that the proposed controller dynamic response is limited by the machine inductances. The high flux linkage correlation with the rise time is due to the increase in steady-state torque since with higher flux the steady-state torque is higher and the rise is approximately linear, with an increase in steady-state torque the rise time also increases.

Regarding the current THD, the parameter influences are well balanced, with the direct axis inductance being a little more correlated. When combining this with the small absolute variance of the THD, it is clear that the control strategy is very robust against model mismatches, maintaining a low distortion even with the wrong parameters.

D. Acceleration Event

A load profile that simulates the car environment was used to simulate a typical acceleration event. The torque reference was set to reach $21.3kW$ of delivered power (after inverter and motor efficiency losses), this value was chosen assuming an 80% powertrain efficiency and an approximation of one-third of the car load on the rear motors. As shown in Figure 12, the RUSH MPC improved the time from $3.943s$ to $3.883s$ (*1.6%*), reaching also a higher top speed than the currently implemented control method and presenting a lower torque ripple.

VIII. EXPERIMENTAL RESULTS

To validate the simulated results, some tests were made on a test bench. The controller was implemented using Xilinx Model Composer inside Simulink, compared with the simulation implementation and then generated for the FPGA present in a Digilent Zybo Z7-20. The hardware used was comprised of the inverter developed on [5], coupled with a current measuring board, developed for this thesis to increase noise immunity. The characterized AMK motor was set in a test bench with a Sensor Technology torque transducer (RWT441-EC-PG) and another AMK motor as load.

First, a simple constant torque test was made to evaluate if the model was correctly calculating the generated torque. Figure 13 presents the torque estimated based on the motor parameters and the current measurements with the value measured with the ST transducer output. The current measurement had some outliers from Electromagnetic Interference (EMI), so the currents and torque in this section were filtered using a Median Absolute Deviation (MAD) outlier filter.

The estimated torque is constantly smaller than the measured torque, with a difference of 0.5Nm. This difference is due to the motor parameters not being perfectly matched with the real motor, and the current measurements not being perfect. The difference between the estimated and measured torque is small enough to consider the model valid for the next tests.

The next step was to verify the current waveform. Ideally, this would be done with nominal torque, but due to a limited maximum inverter
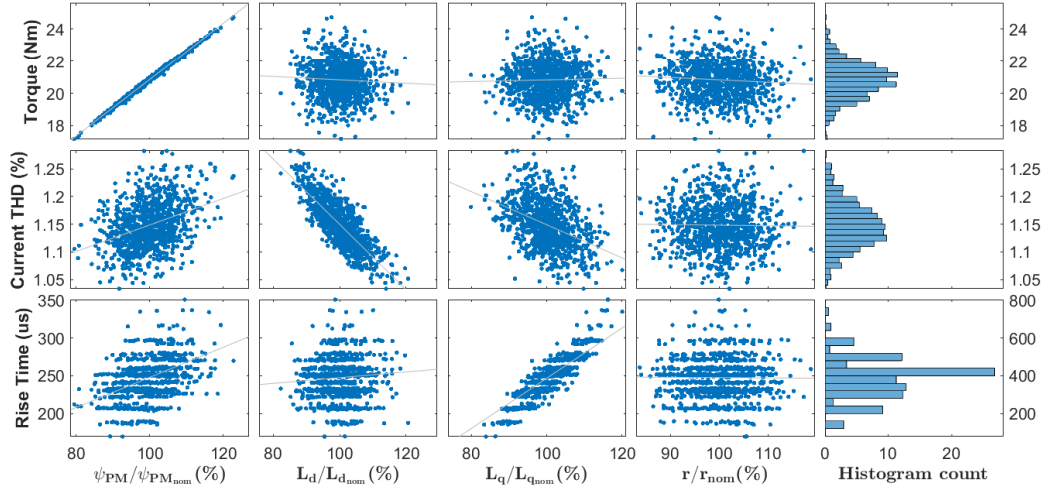
Fig. 11: Monte Carlo Robustness Analysis. The motor parameters are shown in percentual change from the controller's expected value.

(a) Torque profile in the acceleration event using the currently implemented FOC.

(b) Torque profile in the acceleration event using the proposed RUSH MPC

(c) Motor speed profile in the acceleration event. The blue line is the currently implemented FOC while the orange line is the proposed RUSH MPC.

(d) Distance traveled in the acceleration event. The blue line is the currently implemented FOC while the orange line is the proposed RUSH MPC.
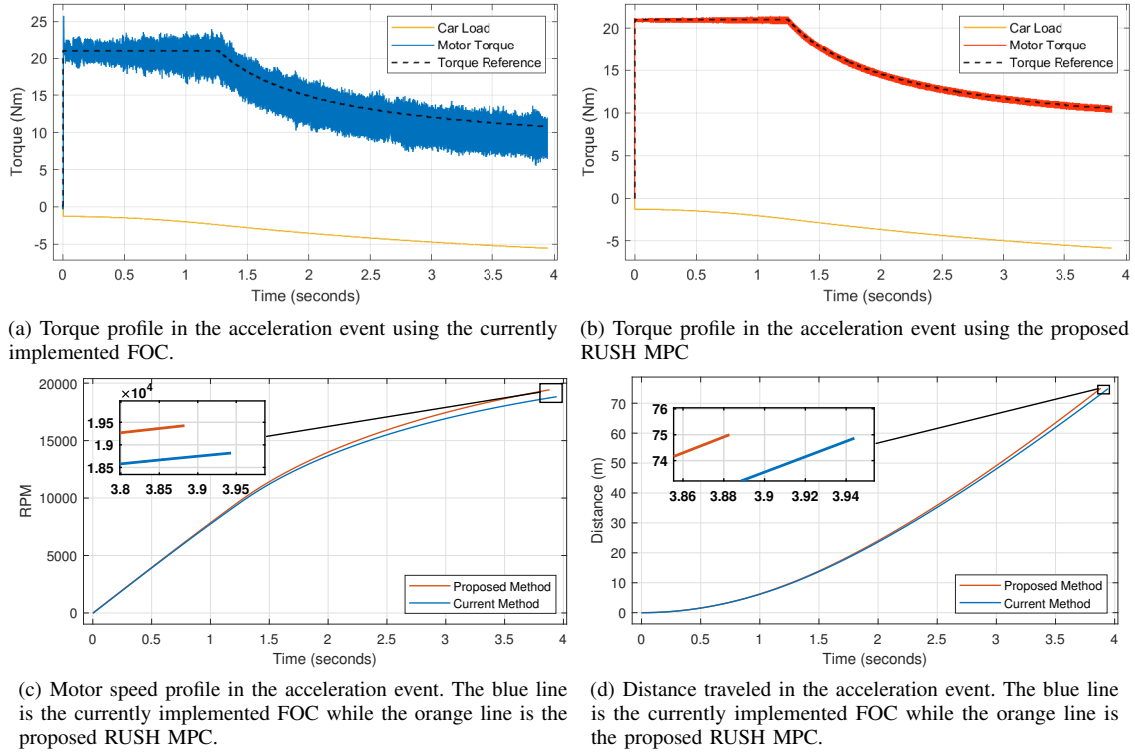
Fig. 12: Control methods comparison on an Acceleration event.

current, the motor was commanded to keep a constant torque of $7Nm$. To create a load, the phases of the other motor were short-circuited, creating a load torque that changes with the speed, this resulted in the speed stabilizing at $269RPM$. The current was measured with the developed measuring board and with an oscilloscope probe (ELDITEST CP6550), while the torque was estimated by the control algorithm and compared with the simulation results. The simulation results were obtained by running the same model used for the controller implementation, with the same parameters and inputs. The results are presented in Figure 14.

The oscilloscope measurement was made with a probe that has a sensitivity of $20mV/A$. The results show that the current measured by the developed board is identical to the one from the oscilloscope probe, proving the system's accuracy. When comparing the measured currents with the simulation results, although they have the same form and amplitude, it is clear that the simulated curve is cleaner. This is backed by the THD comparison, where the measured current has a THD of 2.44% and the simulated current has a THD of 0.47%, both of them calculated to the 50th harmonic. This difference is due to inaccuracies in the measurements and parametrization. The currents presented
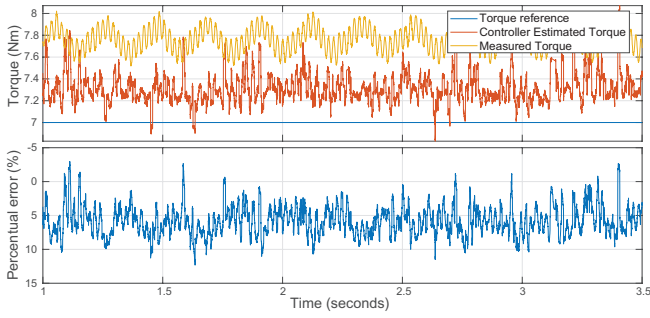
Fig. 13: Torque estimation vs Measured. At the bottom, the percentual error between the measured and estimated torques is shown.

in Figure 14 also show an almost instantaneous dynamic response, transitioning to the specified amplitude without noticeable distortions. The difference between the measured and simulated currents is small enough to consider the model valid.
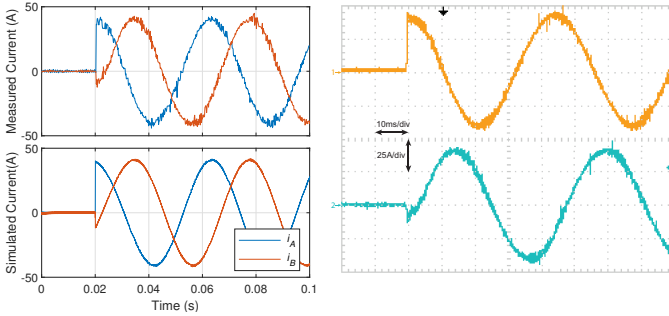
(a) Meausurement Board and Simulation measurements. The plot on top presents values from the measurement board, while the bottom plot is from the simulation.

(b) Osicloscope measurements. Channel 1 (in yellow) represents the line A current, while channel 2 (in blue) is the line B current

Fig. 14: Line Current measurements with a reference torque step from 0 to $7Nm$ at $0.02s$.

Figure 15 shows the torque dynamic response to the step. The reduced sampling time creates significant uncertainty in the measurement, but it is possible to state that the torque rise time is less than 200 microseconds, which is a great improvement when compared with the current control method. The simulation results have a greater sampling rate showing that the rise time is smaller than 100 microseconds. An improved sampling frequency would allow for a better comparison between the simulation and experimental results.

To compare the performance of the proposed method, a torque step was also applied to the current solution of the inverter and control strategy. The results presented in Figure 16 show that as expected, the current solution is much slower than the proposed method, with a rising time of $30ms$ and a delay of $10ms$.

A torque profile to simulate the driver's input was also tested and is presented in Figure 17. The controller fails to output the reference torque around $0.1s$ due to the power supply not being able to provide enough current, thus it switched from constant voltage to constant current output, reducing the available voltage to the inverter. Aside from this, the controller was able to follow the torque reference with a
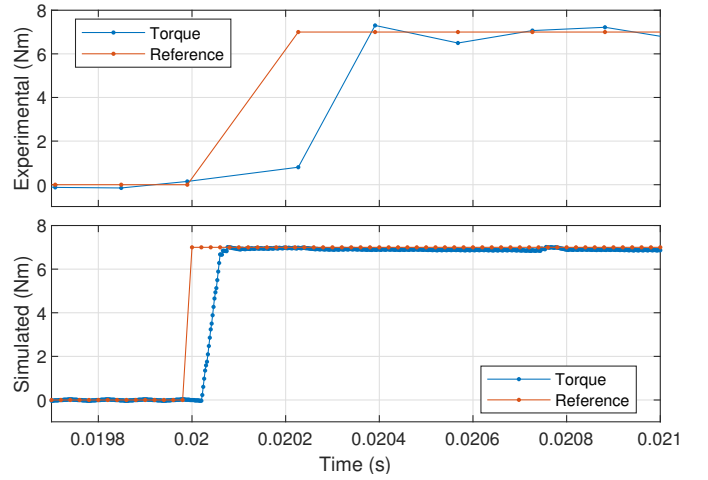

Fig. 15: Torque reference follwoing with a step from 0 to $7Nm$. The upper plot presents the experimental data, while the bottom plot is from the simulation
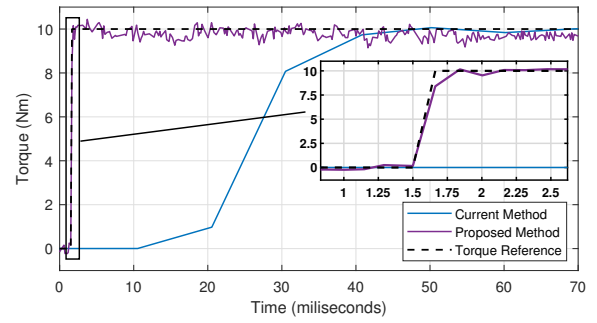

Fig. 16: Comparison of torque step response between the proposed method and the currently implemented method.

small error, averaging $0.04Nm$ and peaking at $1.6Nm$ if the period of the power supply shortage is discarded.

IX. CONCLUSIONS

This work aimed to fill the identified gap in the process of developing a fully in-house powertrain for FST Lisboa which is the control strategy for the motors. This necessity led to a study of control strategies and motor models that was performed with an emphasis on improving the dynamic response of the motor torque and the system efficiency.

To achieve those goals, the motor was characterized by performing several measurements and tests. This not only provided a plant model to simulate the control algorithms but also provided the Formula Student team of Instituto Superior Técnico with valuable information about their motor characteristics and performance.

With the motor parametrized, it was possible to set up a simulation environment and some control strategies were implemented in it. This allowed a comparison between the different strategies and the selection of the most suitable for the application. The selected strategy was the RUSH MPC due to its fast response, low current ripple, and computational efficiency.

The proposed control was then implemented in an FPGA and the hardware necessary to test the control in a test bench was developed. The motors and inverter were then assembled on a test bench to perform experimental tests. Although the testbench hardware was

(a) Line Currents.

(b) Rotor speed.

(c) Torque reference and measured values. The torque reference is derived from the speed controller but saturated to $\pm 5Nm$.

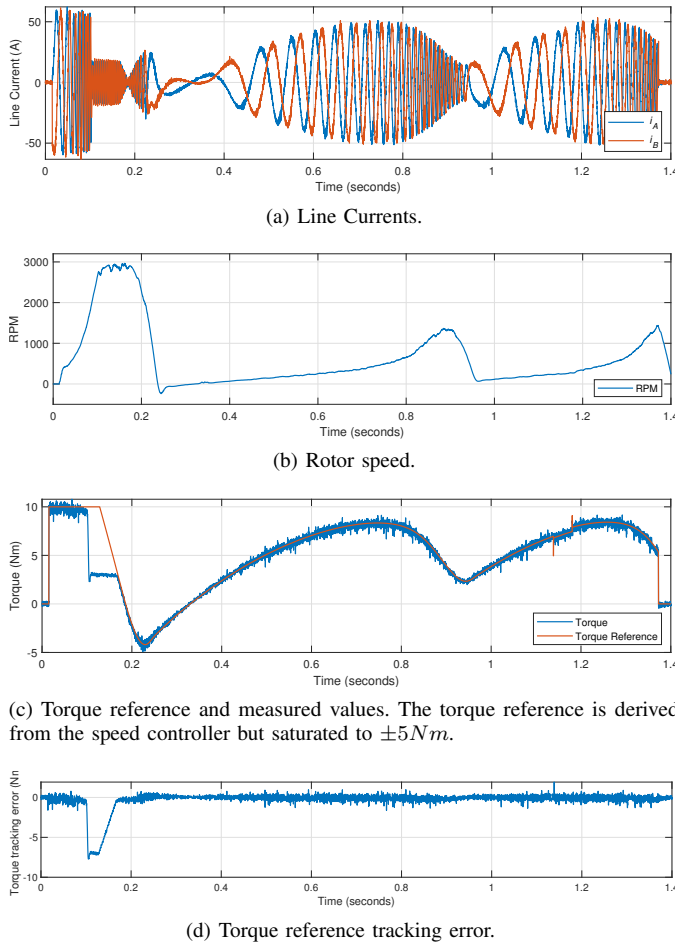

(d) Torque reference tracking error.

Fig. 17: Pedal torque profile.

later verified to be a limiting factor, the experimental results were very promising.

The simulation and experimental results were compared and analyzed showing a close match between them. The torque estimation was also validated and the control strategy was able to control the motor torque with a dynamic response orders of magnitude smaller than the current solution and with a very low torque ripple. The system efficiency was also improved due to the reduction in current THD.

While not exhaustive the developed system continued to build on the great work of the previous thesis and the team's work, creating a solid platform from which the team can continue the work and implement it on the next prototypes. The work done in this thesis was a major step towards the team's goal of developing a fully in-house powertrain.

REFERENCES

[1] Y. Wang et al. A review of high frequency power converters and related technologies. *IEEE Open Journal of the Industrial Electronics Society*, 1:247–260, 2020. ISSN 2644-1284. doi: 10.1109/OJIES.2020.3023691.

[2] P. Karamanakos et al. Model predictive control of power electronic systems: Methods, results, and challenges. *IEEE Open Journal of Industry Applications*, 1:95–114, 2020. ISSN 2644-1241. doi: 10.1109/OJIA.2020.3020184.

[3] J. W. Palmour. Energy efficient wide bandgap devices. In *2006 IEEE Compound Semiconductor Integrated Circuit Symposium*, pages 4–7, Nov 2006. doi: 10.1109/CSICS.2006.319904.

[4] J. G. V. V. Sarrico. Design and implementation of a 20kw, 12000rpm permanent magnet synchronous motor (PMSM) for the IST formula student powertrain. MSc Thesis in Electrical and Computer Engineering, Instituto Superior Técnico, Lisboa, Portugal, June 2017.

[5] P. M. B. de Sousa Correia da Costa. Compact three-phase sic inverter for the IST formula student prototype. MSc Thesis in Electrical and Computer Engineering, Instituto Superior Técnico, Lisboa, Portugal, October 2018.

[6] N. Bianchi et al. A review about flux-weakening operating limits and control techniques for synchronous motor drives. *Energies*, 15(5), 2022. ISSN 1996-1073. doi: 10.3390/en15051930. URL https://www.mdpi.com/1996-1073/15/5/1930.

[7] D. Neacsu. Space vector modulation - an introduction. *IECON*, 01 2001.

[8] S. Vazquez et al. Model predictive control for power converters and drives: Advances and trends. *IEEE Transactions on Industrial Electronics*, 64(2):935–947, Feb 2017. ISSN 1557-9948. doi: 10.1109/TIE.2016.2625238.

[9] M. Merzoug. Comparison of field-oriented control and direct torque control for permanent magnet synchronous motor (PMSM). *International Journal of Electrical and Computer Engineering*, 2(9):1797–1802, 2008.

[10] F. Korkmaz et al. Comparative performance evaluation of FOC and DTC controlled PMSM drives. In *4th International Conference on Power Engineering, Energy and Electrical Drives*, pages 705–708, May 2013. doi: 10.1109/PowerEng.2013.6635696.

[11] R. Souad. Comparison between direct torque control and vector control of a permanent magnet synchronous motor drive. In *2008 13th International Power Electronics and Motion Control Conference*, pages 1209–1214, Sep. 2008. doi: 10.1109/EPEPEMC.2008.4635433.

[12] F. Wang et al. Model predictive control for electrical drive systems-an overview. *CES TRANSACTIONS ON ELECTRICAL MACHINES AND SYSTEMS*, 1(3):219–230, Sep 2017.

[13] Z. Ma et al. Fpga implementation of model predictive control with constant switching frequency for PMSM drives. *IEEE Transactions on Industrial Informatics*, 10(4):2055–2063, Nov 2014.

[14] V. Asadzadeh et al. Selective harmonic elimination in direct torque controlled permanent magnet synchronous motor drive. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 38(1):153–166, 2019.

[15] A. Wijenayake. Modeling and analysis of permanent magnet synchronous motor by taking saturation and core loss into account. In *Proceedings of Second International Conference on Power Electronics and Drive Systems*, volume 2, pages 530–534 vol.2, May 1997. doi: 10.1109/PEDS.1997.627416.

[16] B. Stumberger et al. Evaluation of saturation and cross-magnetization effects in interior permanent-magnet synchronous motor. *IEEE Transactions on Industry Applications*, 39(5):1264–1271, Sep. 2003. ISSN 1939-9367. doi: 10.1109/TIA.2003.816538.

[17] J. D. Barros et al. Fast-predictive optimal control of NPC multilevel converters. *IEEE Transactions on Industrial Electronics*, 60(2):619–627, 2013. doi: 10.1109/TIE.2012.2206352.

[18] S. Vazquez et al. Model predictive control: A review of its

applications in power electronics. *IEEE Industrial Electronics Magazine*, 8(1):16–31, 2014. doi: 10.1109/MIE.2013.2290138.