

PLECS CONFERENCE 2022

Real-Time Simulation of Power Electronics

September 20 • 8:30 to September 21 • 16:30

Technopark Zurich Technoparkstrasse 1 • 8005 Zurich • Switzerland Phone +41 44 533 51 00 • Email <u>info@plexim.com</u>

Integration of PLECS Circuital Models into the Open-Source Design Suite syreDrive

gianmario.pellegrino@polito.it

Politecnico di Torino, Turin, Italy



About the speaker

Gianmario Pellegrino is Professor of Power Converters, Electrical Machines and Drives at Politecnico di Torino.

He is constantly engaged in research projects with the industry in several fields of application. He is the co-founder of the open-source project SyR-e (<u>https://github.com/SyR-e</u>) and of the Power Electronics Innovation Center (<u>https://www.peic.polito.it/</u>) of Politecnico di Torino.

He was a visiting fellow at Aalborg University, Denmark, the University of Nottingham, UK, and the University of Wisconsin-Madison, USA.

He has 60+ IEEE journal papers, five patents and nine Best Paper Awards.

He is and IEEE Fellow and the recipient of the 8th Grand Nagamori Award in 2022.







This is the SyR-e team of the PEIC at PoliTO, a group of young e-motor design and e-drive control experts

Gratitude goes to the group for their contributions to the material presented today



S. Ferrari, PhD

Team Leader

Power Electronics In

olitecnico Torino















syre team Gruppo · 5 partecipanti

PLECS CONFERENCE 2022 @

Today's Content

Introduction

- Politecnico di Torino
- PEIC: Power Electronics Innovation Center

SyR-e and syreDrive

- SyR-e geography
- Main tools for design and modelling
- Case study: Tesla Model 3 rear axle IPM machine
- CCG and VBR approaches
- Direct and inverse flux maps
- Computational time comparison

Conclusion





Politecnico di Torino

Technical School for Engineers founded in 1959, Politecnico di Torino since 1906

Home to Galileo Ferraris, pioneer of electrical engineering

35000 BSc and MSc, 800 PhD students

1000 Faculty members900 Administrative and Technical staff

Budget (2020): 263 M€ (62% State, 12% student fees, 26% projects)

Tuition fee: 0 - 2600€, depending on family income and merit



PEIC: the Power Electronics Innovation Center

The Inter-Departmental Center dedicated to Power Electronics, from the Si-SiC-GaN device to the final application, established in 2017

- 20+ faculty, 2 technicians, 25 PhD students
- Main fields of application: Transportation, Energy, Industry and Home App.
- TRL4 demonstrators, support to higher-TRL prototypes

litecnico

• E-motor drive tests up to 20.000 rpm, 500 kW pk





http://www.peic.polito.it/



PEIC: the Power Electronics Innovation Center

Since 2017

litecnico

- 2 new IEEE Fellows (5 total)
- 1 Nagamori and 1 Grand Nagamori awardees
- Several best paper awards
- 10+ patent applications

Opportunities of collaboration

- Funded and co-funder PhD grants
- Research contracts
- EU funded projects: Horizon Europe, MSCA, Clean Aviation





http://www.peic.polito.it/



eDrives testing



Test -eDrive facility (video)

- Experimental flux maps, cold and hot motor operating conditions (PoliTo benchmark methods)
- Efficiency maps
- HBM Data recorders and torque sensor T12HP
- Automotive test rig: 150 kW, 200Nm, 20,000 rpm, controlled cooling conditions (0-85 °C)







Today's Content

Introduction

- Politecnico di Torino
- PEIC: Power Electronics Innovation Center

SyR-e and syreDrive

- SyR-e geography
- Main tools for design and modelling
- Case study: Tesla Model 3 rear axle IPM machine
- CCG and VBR approaches
- Direct and inverse flux maps
- Computational time comparison

Conclusion



https://github.com/SyR-e



SyR-e



https://github.com/SyR-e

Synchronous Reluctance evolution

Open-source design tool for e-machines and their control, Matlab and <u>FEMM</u> based

Features

- FEAfix: FEA-calibrated design equations
- Fast FEA model manipulation
- Preliminary design of e-machine complement for CAD suites (Ansys, Altair, Simcenter, JMAG)
- syreDrive control simulation model
 - $\,\circ\,$ No need for time consuming FEA co-simulation in Simulink
 - Open-source C control code
 - **NEW** PLECS model generation



Gratitude to Dave Meeker of FEMM



ECCE 2017, tutorial with David Meeker and Francesco Cupertino



Global Impact of SyR-e



8,000+ downloads in 86 countries, since September 2014

Used by partner- (GE Avio, Eldor Corporation, Volvo Cars) and non-partner- companies and academia







New Public Release

v3.4 on GitHub, September 2022

New SyR-e release with ICEM'22 updates!

- Improved (x,b) design plane (ICEM22 paper)
- Improved scaling rules (ECCE22 paper)
- New demo TeslaModel3_custom
- 3rd GUI syrmDesignExplorer
- syreDrive improved
- Improved structural simulation
- Preliminaries of Induction Motor

PLECS model generation not yet public





SyR-e Geography



Politecnico di Torino

Power Electronics Innovation Center

GUI_Syre_MMM

Magnetic Model Manipulation

Main	Scaling & Skewing	Torque-Speed	syreDrive	Waveform	Thermal				
Mode	Is Loaded						Sy Magnetic Mad	R-e	inulati
\checkmark	dq Flux Map	Load	Plot	Save		Print	Magnetic Mod	erman	ipulati
	dqt Flux Məp	Load	Plot	Save			Load	ave	FileChec
	dq Iron Loss Map	Load	Plot	Save			New Sa	ve As	Close all
•	AC Loss Model	Load	Plot	Save					
							Motor Ratings		
Contr	ol Traiectories						Motor name	TesiaMod	del3_custo
	Control Trajectories	Load	Plot	Save		Print	Pathname	D:\syre w	orking cop
	Method:				Evaluate		Motor type	PM	
							Rated power [W]	523370.0	/995
Induc	tance and Anisotropy Maj	ps					Rated speed [rpm]	10835.60	046
	dq inductance Map	Eval	Plot	Save			Rated current [Apk]	1403.7	5
							DC link voltage [V]	231	P
Curre	nt Angle Curves			Steady-S	State Short C	Sircuit	PM temperature	80	•
	Current levels	1	Plot		Evaluate		Stack length [mm]	134	Er
Invers	e Model						Turns in series per phase	12	
	Inverse dg Flux Map	Eval	Plot	Save			inertia [kg m^2]	0.04143	35
		Eval	Plot	Save					
	Inverse dqt Flux Map								

syre notorExamples Axis type SR 🔻 461.2402 18100 num sneed imm 1414.2136 0.0054794 120 vinding length (mm) 126.7903 mhor of 3nhaco col



 $\mathbf{2}$

 $t \; [ms]$

3

1

0



syreDrive Phase currents, PWM ripple

NEW





syreDrive: Control Simulation

[1] A. Varatharajan, D. Brunelli, S. Ferrari, P. Pescetto and G. Pellegrino, "syreDrive: Automated Sensorless Control Code Generation for Synchronous Reluctance Motor Drives," 2021 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD), Modena, Italy, 2021, pp. 192-197.

The Simulink and now PLECS model of the e-drive is automatically generated

Floating-point ANSI-C control code is also provided: hand-written, fully parametrized

The tool is meant for

olitecnico

- \circ Control code simulation, development and debug ightarrow rapid prototyping and HiL
- PWM current waveforms evaluation for FEM re-evaluation of e-motor loss
- Simulation of fault conditions



PWM waveforms and loss

PWM loss computed at 100Nm, 5000rpm

- Additional loss (except PM) is 172W (+14%)
- Mostly on the stator (eddy current term)

PM loss explodes due to PWM:

olitecnico Torino

Power Electronics Innovation

Segmentation and refinement with transient FEA



[2] G. Pellegrino and S. Ferrari, "Design, Identification and Simulation of PM Synchronous Machines for Traction," Tutorial Notes, ICEM 2022, Valencia







syreDrive tab

syreDrive tab of GUI_Syre_MMM

	GUI_Syre	_MMM										-	
PLECS model createdPLECS model of the e-driveCopy and paste a template PLECS file	Main S Model Setu Flux	caling & Skewing up Model type (A maps model (c Control type (1	Torque-Speed	SyreDrive Converter of Inter	Waveform data ON threashold mal resistance [C Dead time	Thermal d [V] 0 0hm] 0.00 [us] 1	001	S Magnetic Mo Load	yR-e del Man Save	TileChec Close a	tion =k II	Syr	e
 Calibrated with SyR-e data of the e-driv 	/e Sensorless	s control					Мо	or Ratings					
 Main features Circuital motor model Flux-map based 2D (dq) or 3D (dq-theta) maps time-average or instantaneous PWM 	Off	On	Low speed region (HF High speed region Position e	 Voltage Injecti Injected signal Demodulation rror estimation 	Ourrent			Motor nam Pathnam Motor typ Rated power [V Rated speed [rpn Rated current [Api DC link voltage [\ PM temperatur	TeslaMod e TeslaMod e C:\syre\m e PM /] 523370.0 n] 10835.6 k] 1403.7 /] 231 e 80	del3_cust notorExar 0995 046 75	om mples\ Rated torqu Maximum speed Max curren Phase resistance Winding tempe	is type SR e [Nm] 46 d [rpm] 1 t [Apk] 14 [Ohm] 0.0 erature	 ▼ 1.2402 8100 4.2136 054794 120
 Discrete-time control Open ANSI-C control code Torque control, speed control, current control Sensorless control of SyR machines 			Create F	PLECS Model	RUN	N Simulink Mod	del	Stack length [mn urns in series per phas Inertia [kg m^/	134 e 12 2] 0.0414	35	End winding lengtl	n [mm] 12	3.7903 1



PLECS Model Folder

Content of the new folder TeslaModel3_custom_ctrl_PLECS

The generated folder contains

- The PLECS model
- Mat files imported from SyRe

• Source C code

olitecnico Torino

Power Electronics Innovation

- Motor_ctrl.c (main C-script)
- User_functions library

📣 MJ	ATLAB R2021b - academic use									_	
н	OME PLOTS	APPS	EDITOR	V	IEW		1 🖥 🛱 🦻 🗗	🕐 💿 Search D	Documentation	۹ م	Gianmario
New	Open Save ⊖ Print ▼	Go To	 	Refact	% ‰ ‰ or 📔 💀 🐼	🖏 Profiler 📝 Analyze	Run Section Bre	ak Avance Run	Step Stop		
	FILE	motorExa	NAVIGATE	el3 custo	CODE	ANALYZE	SECTION		RUN		
Current Folder		THE COLEME	Q	Z Edite	or - C:\syre\moto	rExamples\Tes	laModel3_custom_ctrl_PLE0	CS\Motor_ctrl.c *			⊙×
Name		:+ c:-	o Data M	Mot	or_ctrl.c * × · // // /*	+				*/	
	G	It SIZ	e Date M	2	/* File: Moto	r_ctrl.c			*/	,	
SimMatFiles			14/09/2	3	<pre>/* the parame /* CTRL_TYPE</pre>	ter CTRL_TYF = 0 - Curre	PE decides the type of ent control	control	*/	*/	
InvFluxMapdq.mat	C	97.	14/09/2	5	/* CTRL_TYPE	= 1 - Flux	control			*/	
InvFluxMapdqt.mat	0	36.	14/09/2	7	/* CTRL_TYPE	= 3 - Speed	d control			*/	
Parameters.mat	() 1 K	(B14/09/2	8	/* /* Motor type	:		*/	*/		Ow
Η UserSettings.mat	() 1 K	B 14/09/2	10	/*				*/		
🖃 🚞 User_functions			09/09/2	12							
🖃 📒 Inc			09/09/2	13 14	3 isabc.a 4 isabc.b	-	= InputSignal(0,0); = InputSignal(0,1);				
📓 MotorData.h	(41.	14/09/2	15	isabc.c	-	= InputSignal(0,2);				
📓 User Constants.h	0) 2 K	B09/09/2	16 17	vdc	-	= inputSignal(0,3);				
🥁 User data types.h	0) 2 K	B09/09/2	18 19	theta_mec n ref in	_meas =	<pre>= InputSignal(0,4); = InputSignal(0,5);</pre>	// encode // rpm	er		
With the second	0) 3 K	B09/09/2	20	T_ext		= InputSignal(0,6);	// refere	ence torque		
Will User Macrosh	0) 2 k	B09/09/2	21 22	isdq_ext. isdq_ext.	a = q =	= InputSignal(0,7); = InputSignal(0,8);	// Currer // Currer	nt reference nt reference		
User MotorControl h	Ċ) 1 k	B09/09/2	23 24	Reset		= InputSignal(0.9):	// Black	button		
Wilser MotorControl Fi	unctions h) 2 k	B09/09/2	25	Go	-	= InputSignal(0,10);	// Blue bu	utton		
Willser Variables h		2 2 1	BNQ/NQ/2	26	Ctrl_type	-	= InputSignal(0,11);				
			00/00/2	28 29	inj_waveform	orm =	<pre>= InputSignal(0,12); = InputSignal(0,13);</pre>	// Inject // Curren	ted waveform (s nt or Flux demo	sinusoidal o oduation	r square
Ucer MeterCentrel E	unctions c (10	09/09/2	30	HS_ctrl	-	= InputSignal(0,14);	// High s	speed position	error estim	ation te
User_WotorControl_Functions.c		> 19.	09/09/2	31 32	SS_on accel	-	= InputSignal(0,15); = InputSignal(0,16);	// Sensor // speed	rless ON or OF acceleration i	∽pm/s	
init_sim_PLECS.m			BU9/U9/2	33	Quad_Maps		<pre>= InputSignal(0,17); = InputSignal(0,18);</pre>				
Motor_ctrl.c		16.	14/09/2	35	th0	-	= InputSignal(0,19);				
motorModel.mat		21.	14/09/2	36 37	switch(St	ate){					
TeslaModel3_custom_M	otor_ctrl.plecs	J 47.	14/09/2	38	(259	FRROR					
·			'	40	/	/ Variables	Init				
TI-A	Andal2 sustam Mator -t-ll-		model file)	41	p	wm_stop	= 1;	_			-
Testaw	nouelo_custom_wotor_ctrl.piec	5 (PLECS	niouer nie)			Zoom: UT			aadar fila		27 Col 2
						20011 01	I-0 CRLF	C++ Source of f	leader me	Lr	57 COLS



E-Drive Model Overview

di Torino

Power Electronics Innovation Cent

PLECS model generated by syreDrive



Control Subsystem

Triggered sub-system executed at $T_s = T_{PWM}$

One-step actuation delay

C-Script control code with user-defined library calls

C source code

 automatically calibrated using the machine parameters existing in SyRe, same as for the e-motor model

• Open access

Copy of hand-written code (not a generated code)

The control code is portable to dSPACE and ARM based MCUs (e.g. SMT32)

Content of the triggered sub-system Digital Control





User-Defined Motor Control Library

C-script code declarations

The header files of a User_functions folder are included in the Code declarations section of the C-script block

Besides data types, constants and variables definitions, the file MotorData.h contains the motor parameters for control (LUTs for MTPA, flux maps LUTs for flux observer, etc ..)

tecnico

C-Script parameters: TeslaModel3_custo – 🛛	×
Setup Code	
Code declarations	\sim
<pre>1 #include "math.h" 2 #include "User_functions\Inc\User_data_types.h" 3 #include "User_functions\Inc\User_Variables.h" 4 #include "User_functions\Inc\User_Constants.h" 5 #include "User_functions\Inc\User_Macros.h" 6 #include "User_functions\Inc\User_MotorControl_Funct 7 #include "User_functions\Src\User_MotorControl_Funct 9 10 11 12 </pre>	
OK Cancel Apply Help)

User-Defined Motor Control Library

C-script code initialization

Once again, the variables are initialized according to the parameters of the e-drive (current measurements A/D resolution, acceleration rate ..) defined in SyR-e

All such variables are editable directly in PLECS

N C-Script parameters: Tesla	Model3_custo.	–		Х
Setup Code				
Start function code				\sim
1 pwm_stop = 1;				
2 counter = 0; 3				
4 offset_current_a	= 2120;			
5 offset_current_b	= 2120;			_
6 offset_current_c	= 2120;			
8 n ref in	= 0.0f;			
9 omega ref in	= 0.0f;			
10 omega ref ramp	= 0.0f;			
11 accel	= 1000;	// rpm/		
12 omega_elt_meas	= 0.0f;			
13 omega_elt_meas_f	= 0.0f;			
ОК	Cancel	Apply	Hel	р



Motor Control Function

The function Motor_ctrl.c contains the FOC source code

Current vector control, with several options

- $^{\rm o}$ Torque control / speed control / i_d , i_q control
- Sensorless control (for SyR machines)

olitecnico i Torino

Power Electronics Innovation

LUT-based MTPA (Maximum Torque per Ampere) law is included, from SyRe data

CODE ANAL CODE ANAL

CODE ANALYZE SECTION Step Stop RUN -----Control Type -----// 224 225 switch (Ctrl_type){ 226 227 case 0: //CurrentControl 228 isdq ref.d = isdq ext.d; 229 isdq ref.q = isdq ext.q; 230 break: 231 232 case 2: //TorqueControl 233 ReadLut(&ID REF[0], fabs(T ext), TMAX, TMIN, DT, INV DT, &isdq ref.d); 234 ReadLut(&IO REF[0], fabs(T ext), TMAX, TMIN, DT, INV DT, &isdq ref.q); 235 switch (Quad_Maps){ 236 case 0: 237 isdq_ref.d = sgn(T_ext)*isdq_ref.d; 238 break; 239 240 case 1: 241 isdq ref.d = sgn(T ext)*isdq ref.d; 242 break; 243 244 case 2: 245 isdq_ref.q = sgn(T_ext)*isdq_ref.q; 246 break; 247 Zoom: ... UTF-8 CRLF C++ source or header file Ln 1 Col 1

Motor ctrl.c function for torgue and speed control

Х

1 🖓 🕤 🖸

_

Inverter Subsystem

Based on library **3-ph 2-level Voltage Source Inverter** of PLECS

- Instantaneous or time-averaged switches
- Dead-time modeled also in the average model
- Sub-cycle average model type is comfortably used for either instantaneous and average simulation

The circuital model is compatible with simulation of fault conditions (uncontrolled converter, active short circuit, ph-to-ph short, etc ..)

Content of the Inverter Model subsystem





Sub-Cycle Average

- Switched: Ideal switches are used to model the individual power semiconductors. The control inputs are instantaneous logical gate signals

- Sub-cycle average: The component is modeled with controlled voltage and current sources. The control inputs are the relative on-times of the semiconductors with values between 0 and 1. The gate signals can be either instantaneous (using only values 0 and 1) or time-averaged.

Full circuital, non idealized, approach

Comfortable: flexible use in instantaneous and average models

Content of the Inverter Model subsystem ▶ TeslaModel3_custom_Motor_ctrl/Inverter Model * Х File Edit View Simulation Format Coder Window Help Dabc 🗸 🗲 Dabc Enable **Tabc** Command Configuration Block Parameters: TeslaModel3 custom Motor ctrl/Inverter Mo... X 3-Phase Voltage Source Inverter (mask) (link) This power module implements a 3-phase 2-level voltage source inverter. It offers two configurations: - Switched: Ideal switches are used to model the individual power semiconductors. The control inputs are instantaneous logical gate signals - Sub-cycle average: The component is modeled with controlled voltage and current sources. The control inputs are the relative on-times of the semiconductors with values between 0 and 1. The gate signals can be either instantaneous (using only values 0 and 1) or time-averaged. g 🗲 Configuration \sim Sub-cycle average Parameters available Semiconductor symbol: IGBT \sim Assertions: \sim On OK Cancel Apply Help



Insight on the e-motor model

MSc Thesis by A. Bojoi in early 2022 on LUTbased circuital models of synchronous emachines

Two modelling approaches were compared and tested in Simulink and PLECS

- VBR: voltage behind reactance
- CCG: controlled current generators



[3] A. Bojoi, "Advanced Dynamic Model of E-motor for Control Rapid Prototyping [MSc Thesis]", Politecnico di Torino, 2022, <u>https://webthesis.biblio.polito.it/22088/1/tesi.pdf</u>



What is PLECS recommending?

The Reference PLECS library models are:

Permanent Magnet SM

- VBR or Rotor Reference Frame (CCG) approaches
- \circ Constant parameters ($L_d,L_q,$ PM flux linkage), no LUTs
- $\circ i_d$, i_q are the state variables

Non-excited SM

• VBR approach based on flux map LUTs

PLECS library models





VBR or CCG approaches

Voltage Behind Reactance:

olitecnico i Torino

Power Electronics Innovation

- 3-phase RLE circuit (abc coordinates)
- Controlled voltage source E and inductance L



- no LUTs: constant L_d , L_q , PM flux linkage
- i_d , i_q are the state variables
- Non-excited SM uses VBR with flux-map LUTs

Controlled Current Generators

- Controlled current sources
- State equations integrated into the C-script



syreDrive motor model

Custom model inspired to PLECS and Simulink library models

After comparison of VBR and CCG approaches, CCG was selected for use in syreDrive, for reason of better computational time

Work is in progress; we might change decisions for the future

Content of the Motor Model subsystem





Motor Subsystem

Controlled current generators (CCG) approach

Phase currents are calculated by manipulation of current generators' voltages, using the motor parameters and flux maps

Flux linkages in *dq* frame as state variables

Inverse flux maps need (flux in, current out)

olitecnico i Torino

Power Electronics Innovation

$$i_d = f(\lambda_d, \lambda_q)$$

$$i_q = g(\lambda_d, \lambda_q)$$

Content of the Circuital Model subsystem





Limits of the CCG approach

Direct flux maps

$$\lambda_d = f'(i_d, i_q)$$
$$\lambda_q = g'(i_d, i_q)$$

This is the output of experiments or FEM



Inverse flux maps

Power Electronics Innovation

olitecnico Torino

$$i_d = f(\lambda_d, \lambda_q)$$

$$i_q = g(\lambda_d, \lambda_q)$$

This is obtained via numerical manipulation

Example

The use of the inverse flux maps limits the i_d , i_q domain of identification of the machine under test

- This can be overcome with FEA
- Strong limitation with experimental flux maps

The figure is exaggerated: with the 2nd quadrant only, the limitation is within 70%





VBR approach

Voltage Behind Reactance (VBR) approach

Phase quantities (abc), state equations solved within the circuit

Direct flux maps used for back-EMF calculation

Incremental inductance maps used for adjusting the inductance term

5x 2D-LUTs instead of 2x 2D-LUTs of CCG

• Direct flux maps (2x) - λ_d , λ_q

olitecnico Torino

• Incremental inductance maps (3x) - l_d , l_q , l_{dq}





VBR model of the motor

Comparison of CCG and VBR

A standardized speed and torque cycle of 1.0s is used as reference in the following

Computational time comparison

- VBR takes 2.3-2.5x longer than CCG in avg mode
- 3.6-4.5x longer in intantaneous mode

The most accurate dq-theta approach comes at a limited extra computational cost

Windows laptop, 6-core Intel i7-10750H CPU, 2.60GHz, 16GB RAM

litecnico





Dq-theta model

3-dimensional representation of flux maps, including the rotor phase angle input

$$\lambda_d = f'(i_d, i_q, \mathbf{\theta})$$
$$\lambda_q = g'(i_d, i_q, \mathbf{\theta})$$

- dq maps are called the fundamental model
- dq-theta includes space harmonic effects, giving birth to back-emf and torque undulation

The 3D maps are natively retrieved from FEA simulations (just by manipulation)

Same approach used with experimental flux maps, using raw data points (e.g. from HBM data recorder) [4] S. Ferrari, G. Dilevrano, P. Ragazzo and G. Pellegrino, "The dq-theta Flux Map Model of Synchronous Machines," 2021 IEEE Energy Conversion Congress and Exposition (ECCE), 2021







Waveforms example

Torque reversal, **dq** model

Peak torque reversal is shown at 2000 rpm

- FOC control with MTPA id, iq references
- Sampled phase current waveforms

olitecnico Torino

Power Electronics Innovation

This example uses the **dq model of the machine** (no ripple effect), based on inverse dq flux maps (CCG)

Under proper current control, currents are sinusoidal, and torque is smooth



Torque reversal, **dq-theta** model

Peak torque reversal is shown at 2000 rpmFOC control with MTPA id, ig references

Sampled phase current waveforms

The **dq-theta model** of the machine is used this time, including the flux linkage and torque harmonic effects

Currents no longer sinusoidal, torque ripple is evident

High-fidelity model for model-based control design, HiL, digital twin, ..





Comparison to Simulink

The same 1.0s duty cycle is used

Computational time comparison (CCG)

- The instantaneous PLECS model is way faster (2.75 with dq, 3.2 with dq-theta)
- AVG models are comparable

Besides faster computation, PLECS shows better consistency w.r.t. changes between PWM and AVG and between dq and dq-theta

Simulation settings

Instantaneous: variable step solver, Tmax 2us, average: variable step solver, Tmax 10us. Switching frequency 10 kHz, single sampling, single update





Summary: CCG vs VBR and PLECS vs SImulink

The respective VBR models are included in the comparison

Trends of longer simulation time by Simulink are confirmed

olitecnico Torino

Also, the trend of less consistency is confirmed: unexpectedly, the AVG model is faster with VBR than CCG (contrary to the assumption of more time related to more LUTs)



Conclusion

syreDrive, the SyR-e approach to control simulation for PM synchronous machines was presented, using the IPM motor drive of Tesla Model3 as reference case

SyR-e is about to include PLECS models generation, beyond Simulink ones

The CCG and VBR approaches were compared, and advanced LUT-based modelling features have been presented, such as the dq-theta approach

Computational time comparison was shown and commented

Inverse flux maps dictate a tradeoff between current domain (VBR is best) and time (CCG is best)

The dq-theta model adds accuracy at limited cost

Work is in progress

We invite you to try syreDrive and collaborate with us!



References

Politecnico di Torino

[1] A. Varatharajan, D. Brunelli, S. Ferrari, P. Pescetto and G.
Pellegrino, "syreDrive: Automated Sensorless Control Code
Generation for Synchronous Reluctance Motor Drives," 2021 IEEE
Workshop on Electrical Machines Design, Control and Diagnosis
(WEMDCD), Modena, Italy, 2021, pp. 192-197.

[2] G. Pellegrino and S. Ferrari, "<u>Design, Identification and</u> <u>Simulation of PM Synchronous Machines for Traction</u>," Tutorial Notes, ICEM 2022, Valencia

[3] A. Bojoi, "Advanced Dynamic Model of E-motor for Control Rapid Prototyping [MSc Thesis]", Politecnico di Torino, 2022, <u>https://webthesis.biblio.polito.it/22088/1/tesi.pdf</u>

[4] S. Ferrari, G. Dilevrano, P. Ragazzo and G. Pellegrino, "The dqtheta Flux Map Model of Synchronous Machines," 2021 IEEE Energy Conversion Congress and Exposition (ECCE), 2021



Thank you!

Questions are very welcome

gianmario.pellegrino@polito.it

www.peic.polito.it



