SyR-e  
Magnetic Model Manipulation  
 ~   
GUI Quickstart Guide

|  |  |
| --- | --- |
|  | SyR-e **M**agnetic **M**odel **M**anipulation (MMM)  Version 1.0  Quickstart guide for Graphic User Interface  April, 2020 |

*This document is a “work in progress” written to give some basic guidelines on how to use the new GUI for the magnetic model manipulation in SyR-e. The tool is in continuous evolution and some of the procedure can change, some functions can be added, deleted or modified. We will try to update this guide, at least for the critical updates. As all the new tools, it is possible that some bugs or unclear section are presents. We encourage a constructive feedback on the critical aspects and the possible errors.*

Simone Ferrari ([simone.ferrari@polito.it](mailto:simone.ferrari@polito.it))

Gianmario Pellegrino ([gianmario.pellegrino@polito.it](mailto:gianmario.pellegrino@polito.it))

# Introduction

SyR-e (**Sy**nchronous **R**eluctance**-e**volution) is an open source tool developed in Matlab/Octave for the design, analysis and optimization of electric machines. SyR-e was first released in 2014 and all the functions was controlled from a Graphical User Interface (GUI) in Matlab. Almost all the features in SyR-e are related to the GUI, that helps order of the project. All the new features added in the years are linked to the GUI, following the structure rules. The only exception to this ruled organization is the functions to manipulate the magnetic model of the motors.

In the Spring 2020, during the Italian Covid-19 outbreak and the forced quarantine, the times are come: after some weeks of work, a second GUI of the SyR-e project, dedicated to the magnetic model manipulation born.

The name of this new project is MMM, standing for Magnetic Model Manipulation and replace all the old scripts and functions located in the folder *syreManipulateMM\*, maintaining almost the same theoretical background.

## Matlab version required

The new GUI is developed with Matlab AppDesigner, with the Matlab2019b version. Moreover, the function to be launched from command windows are under development.

# Getting Started

The new GUI is launched from the following command

GUI\_Syre\_MMM

It is composed from five tabs:

* **Main**: contains the command to load single models and do a quick elaboration of the flux maps
* **dqtMap**: contains all the procedures related to the dqtMap model
* **Scaling&Skewing**: it is possible to scale the magnetic model (change stack length, number of turns and 3D inductances) and obtain the magnetic model of the skewed motor
* **Torque-Speed**: allows the evaluation of performance on the torque-speed plane
* **syreDrive**: allows the interface with Simulink model (under development)

In addition, a table with the motor ratings and some crucial information on the motor is always visible.

The GUI is created with Matlab AppDesigner, Matlab2019b. The compatibility with older Matlab releases is ensured up to Matlab2016b.

## Data structure

The base data structure is called motorModel and contains several sub-structures for each model and elaboration. If one model is not available, the respective substructure is empty.

## New, load and save models

The button Load Model allows to load the same .mat file of the SyR-e project. If the structure motorModel is present, it will load also all the flux map data. Conversely, if motorModel structure is not present, all the information contained in dataSet are used to write the ratings of the motor. The single models (flux maps, iron loss, dqtMap, skin effects) must be loaded separately, with the dedicated buttons if not presents in the file.

Dealing with the save function, it is possible to save all the data in the same loaded file (Save Model button), or save a new project (Save Model As button).

It is also possible create a new model from scratch, that is useful if the original SyR-e file is not available, as for experimental results.

## Folder organization

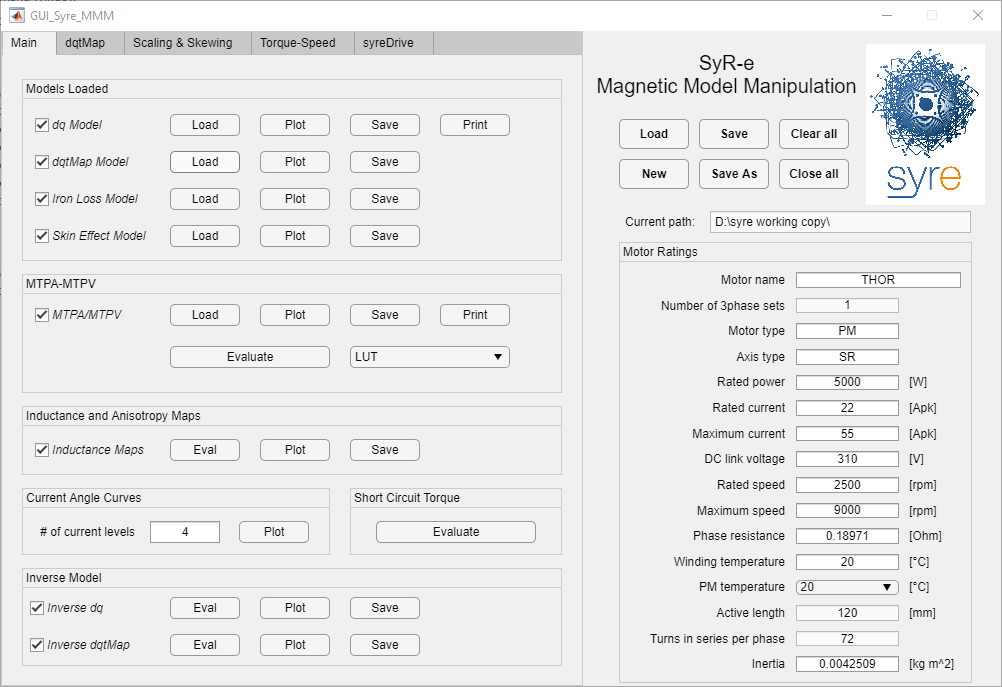
The GUI will create a new “MMM results\” folder in the motor results folder and all the models will be saved there. Inside the MMM results folder, there will be a cache folder for the magnetic models at different temperatures (useful for PM machines).

## Temperatures in the main data

There are two temperature data in the motor ratings tab: one for windings and one for PM. The former acts also on the phase resistance: if temperature change, phase resistance is update accordingly (the opposite is not true). Dealing with the PM temperature, it could be selected from a drop-down menu. Note that change PM temperature means change flux maps, so, if a new PM temperature is added, new maps must be loaded. The MMM works on just one map, so the maps at the other temperatures are saved in a cache folder.

# Tab Functionalities

## Main Tab



1. Main Tab

From the Main Tab it is possible to load the model and elaborate and plot the flux maps. Here each model or elaboration can have five functions:

* Load: allows to load the specified model or curves
* Elab: allows to perform some elaborations based on the loaded models
* Plot: allows to plot the selected model. At the end of the plot, a separate window will ask to save or not the pictures
* Save: allows to save the selected model, in the actual structure
* Print: print the C files for the control algorithm.
* Delete: by un-checking the respective checkbox.

### Models Loaded

Here it is shows which models are loaded. They are:

* *dq Model*: originally included in *fdfq\_idiq\_n256.mat*, it is the base flux maps, so flux linkage, torque, torque ripple rms and peak-to-peak function of the currents.
* *dqtMap Model*: it is the extended flux map modeling, so flux linkages and torque function of currents and rotor position .
* *Iron Loss Model*: iron loss model, saved as maps of loss terms function of the currents for a given speed. The model contains also the coefficients for the speed change.
* *Skin Effect Model*: it contains the coefficient for the Joule AC loss, function of frequency

### MTPA-MTPV

This elaboration allows to compute the control trajectory, stored in the AOA (Allowed Operating Area) substructure. At the moment, the two loci on the dq plane are:

* MTPA: Maximum Torque Per Ampere
* MTPV: Maximum Torque Per Voltage

But the computation functions are unified, and it is possible in the future to implement other locus, as the maximum power factor per ampere, or others. It is possible to use the interpolated or fitted loci, by selecting the proper option in the menu.

### Inductance and Anisotropy Maps

This function allows to compute the differential inductances on the plane and the anisotropy map, expressed as . The cross-saturation differential inductances are included in the computation.

### Current Angle Curves

It allows to plot quantities at constant current and function of the current angle , by elaborating the dq flux maps. The plotted quantities are:

* Torque
* Power factor
* Flux linkage amplitude
* Flux linkage angle
* Peak-to-peak torque ripple

The current levels are imposed in two ways:

* If the parameter is a number, it represents the number of current levels, up to the maximum current
* If the parameter is a vector, it represents the current levels that must be plotted, in per-unit of the rated current

No further models are saved from this elaboration

### Short Circuit Torque

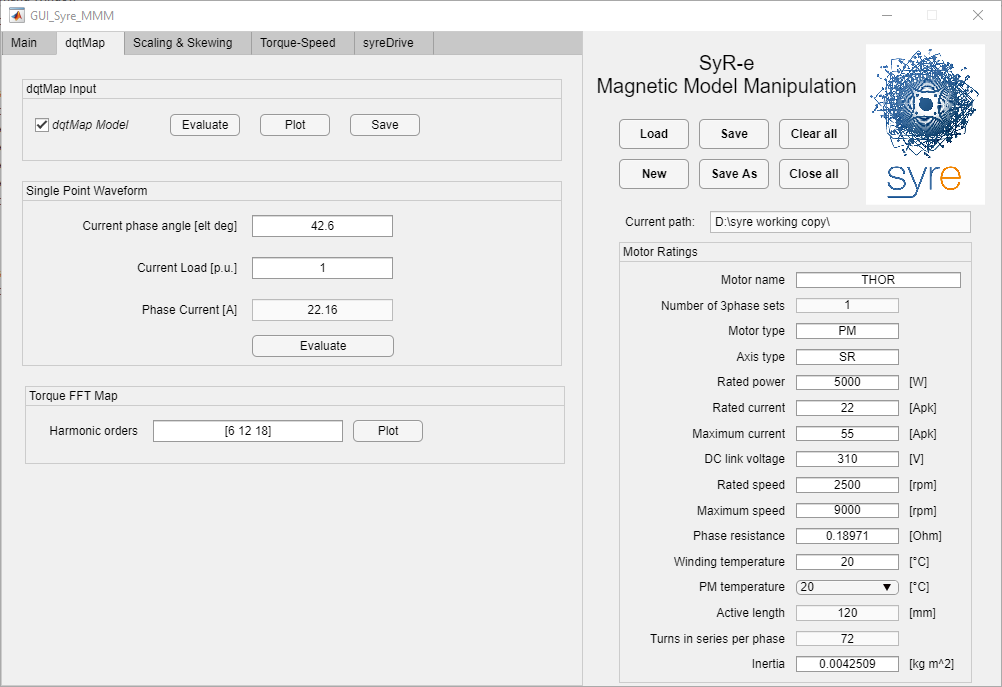
This button computes the steady-state short circuit torque from the flux maps at different rotor speeds, from zero to the maximum speed defined in the main data. The results (figures and a data structure) are saved in the respective folder.

### Inverse Model

The inverse model is when currents and torque are expressed function of the flux linkages . Here two models can be selected, the 2D model (with average values) and the 3D model, based on the dqtMap structure.

The inversion of the model is imposed to reduce the flux linkages limits to have a fully-defined rectangle. Moreover, it is possible to extend the limits by extrapolating or by having NaN grid (just 2D)

## dqtMap Tab



1. dqtMap Tab

The dqtMap Tab allows to manipulate the dqtMap model.

### dqtMap Inputs

This is almost a copy of the Main Tab controls, with the possibility also to compute the dqtMap model from the *F\_map* data.

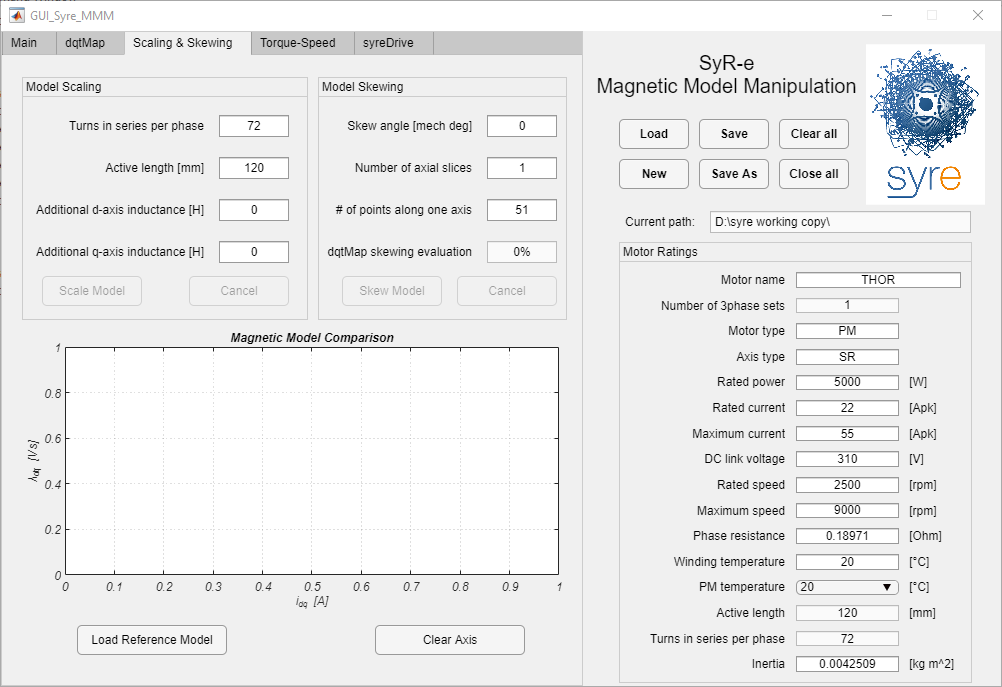
### Single Point Waveform

This function is similar to the Operating Point Evaluation in SyR-e, with the difference that no FEA simulation are executed and all the waveforms are extracted from the dqtMap Model.

### Torque FFT Harmonics

This allows to see the target torque ripple harmonic maps on the dq plane.

## Scaling & Skewing Tab



1. Scaling & Skewing Tab

This tab allows to edit the magnetic model of the motor, to obtain a scaled model, add 3D inductance effects and compute the skewed motor performances. It is divided into two section: scaling and skewing. If one of them is active, the other cannot be executed and the motor model cannot be saved (a Save Model As is required to save the new model). The modifications are done on the active model (PM temperature), but are extended to the other PM temperatures, when the model is saved.

On the bottom, there is a plot that shows the dq flux linkages before and after the modification. It is also possible to load a reference model (as experimental curves, for example).

### Model Scaling

The model scaling allows three modifications:

* Change the active stack length of the motor
* Change the number of turns in series per phase
* Add extra inductances on d and/or q axis

The model scaled are:

* *dq Model*
* *dqtMap Model*
* *Iron Loss Model*

All the other models are deleted and must be re-loaded/re-computed.

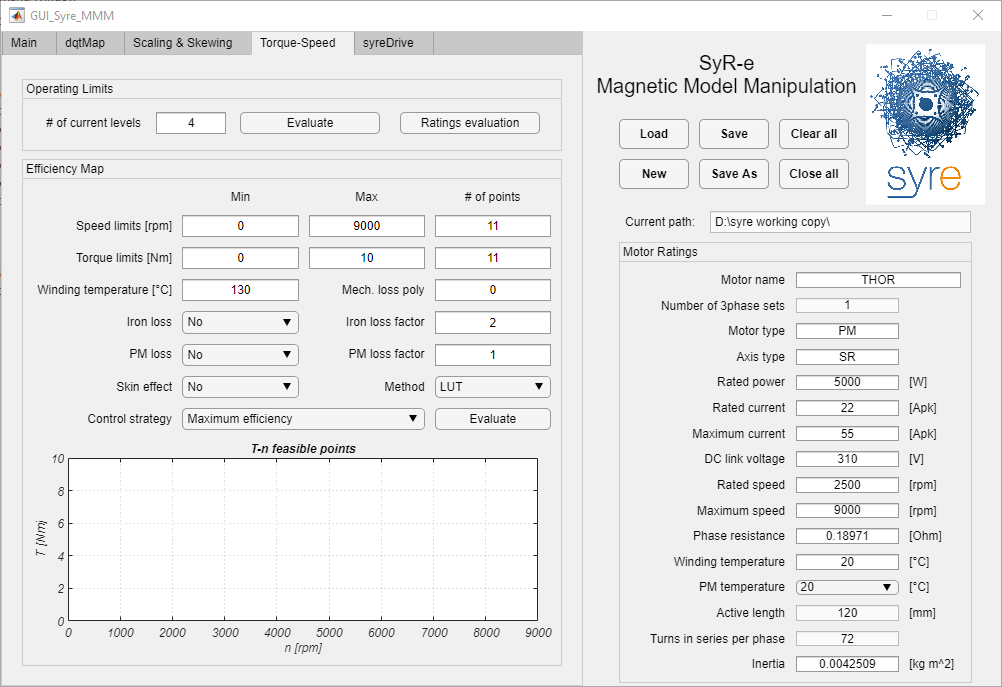
### Model Skewing

This option allows to skew the motor. The inputs are:

* Skew angle in mechanical degrees
* Number of axial slices
* Number of points along one axis

As for model scaling, just the dq, dqtMap and iron loss model are skewed. If the dqtMap is available, the torque ripple of the skewed machine is computed, otherwise it is set as NaN.

## Torque-Speed Tab



1. Torque-Speed Tab

This tab collects all the elaboration to evaluate the motor performance for different speeds. No additional models are saved in the motorModel, but elaboration results are saved in the respective folders.

### Operating Limits

This panel allows to compute the operating limits, from the dq model, with the selected voltage and current limits, saved in the motor ratings and neglecting the loss terms and the phase resistance. The input “# of current levels” works as for the current angle curves. The outputs of the elaboration are:

* Torque-speed profiles
* Power-speed profiles
* Power factor-speed profiles
* Phase current-speed profiles
* Line voltage-speed profiles
* Control trajectories on the dq plane

With the “Rating Evaluation” button it is possible to compute base speed and rated power from the current and voltage limits

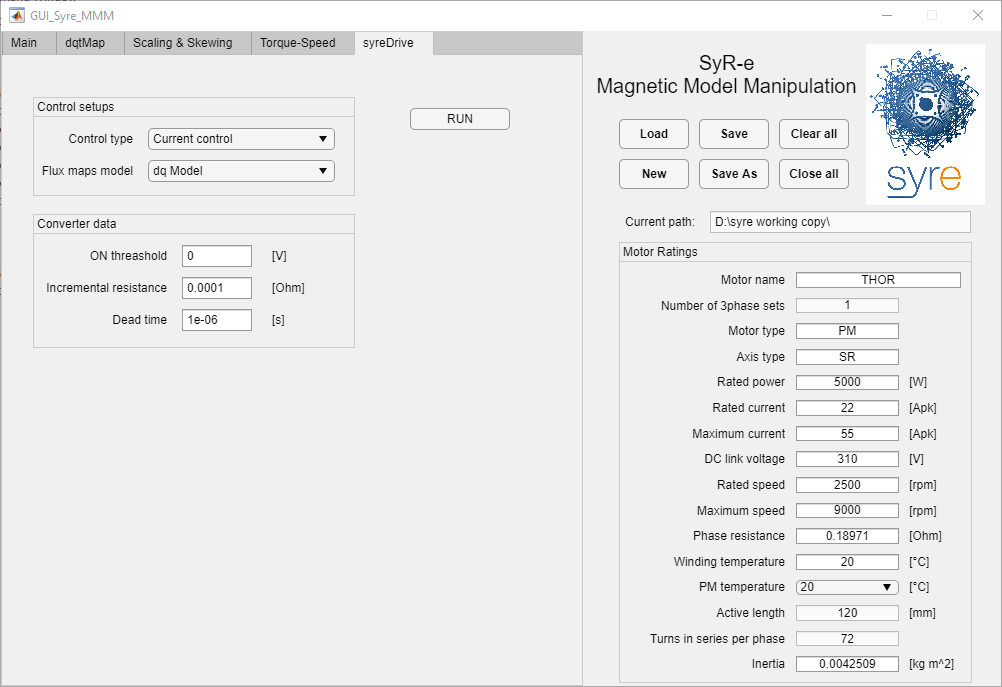
### Efficiency Map

This section groups all the functionalities of the MaxTw.m function. Here a regular grid on the torque-speed plane is imposed, with current and voltage limits. The efficiency map and control trajectories are computed according to two strategies (maximum efficiency or MTPA) and it is possible account or neglect iron loss, skin effect and mechanical loss.

The plot in the tab shows the advance of the computation and the feasible/unfeasible points (colored respectively in green and red). Also in this case, no model are loaded in the motorModel structure, but the results are eventually saved in a .mat file.

## SyR-e Drive

SyR-e drive tab offers the opportunity to export a motor model to Simulink and simulate the drive system, with different control strategies. At the moment is a work-in-progress functionality and it will be updated in future releases.



1. syreDrive Tab