Exercise 3

The objective of this exercise is to understand, practice and apply the concept of one-step predictive current control to a three-phase grid system. The following topics are considered:

- Analysis of predictions to debug the controller and to ensure that it works as desired.
- Delay compensation.
- Operation during grid voltage asymmetries.
- Predictive direct power control.

The understanding developed in this exercise will be instrumental when developing more elaborate predictive controllers.

3.1 Predictive current control: Implementation

Consider the three-phase system shown in Fig. 1. A three-level neutral point clamped (NPC) converter with the total dc-link voltage V_{dc} is connected to a symmetrical three-phase grid. The neutral point potential is assumed to be zero. The transformer, distribution and transmission system are modeled by the reactance X and the resistance R. The rated values of the system are provided in Table 1 and the parameters of the system are summarized in Table 2.

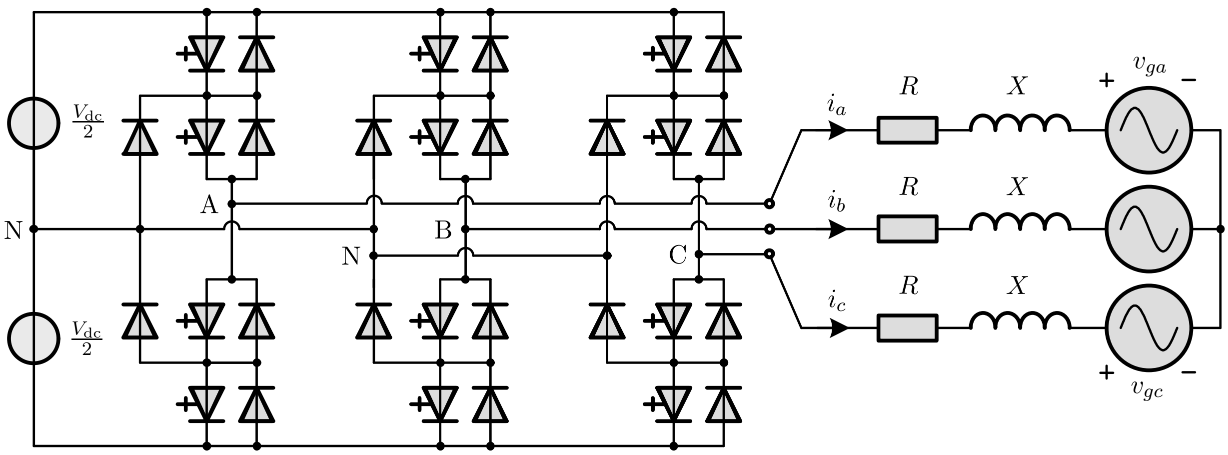

Figure 1: Grid-connected NPC converter system with a simplified grid representation

Table 1: Rated values of the grid-connected converter system

Parameter	Symbol	SI value
Voltage	V_R	3300 V
Current	I_R	1575 A
Apparent power	S_R	9 MVA
Angular grid frequency	ω_R	$2\pi 50$ rad/s

Table 2: SI and per unit (p.u.) parameters of the grid-connected converter system

Parameter	Symbol	SI value	Symbol	p.u. value
Grid frequency	f	50 Hz	ω	1 p.u.
System inductance	L	0.578 mH	X	0.150 p.u.
System resistance	R	18.15 mΩ	R	0.015 p.u.
Dc-link voltage	V_{dc}	5.2 kV	V_{dc}	1.930 p.u.

1. Preparation: Download the Matlab/Simulink files from the course webpage. Start Matlab and open the Simulink model threePhase_grid.slx. A few comments are provided hereafter:

   - The file predCurrCtrl_ini.m is called when the simulation is started, see Model Settings / Model Properties / Callbacks / InitFcn. The parameter and controller settings are provided in this file.
   - The yellow blocks, which execute the current controller (Matlab function named currCtrl) and power controller (Matlab function named powerCtrl), need to be completed. Note that the orange block provides the input to a multiport switch, which selects the controller: input "1" activates the power controller, while input "2" activates the current controller.
   - The power electronic system is simulated at the short sampling interval of 5 μs, while the controller is run at the (often much) longer sampling interval $T_s > 5\,\mu$s. To achieve this, several rate transition blocks are used in the Simulink model.
   - We distinguish between continuous-time quantities, which depend on the time t, and discrete-time quantities, which are a function of the time step k.

2. Recall from Exercise 1 that the continuous-time system model in the stationary reference frame and the per unit system is given by

$$\boldsymbol{v}_{c,\alpha\beta} = R\boldsymbol{i}_{\alpha\beta} + X\frac{\mathrm{d}\boldsymbol{i}_{\alpha\beta}}{\mathrm{d}t} + \boldsymbol{v}_{g,\alpha\beta}$$

   with the converter voltage $\boldsymbol{v}_{c,\alpha\beta} = [v_{c\alpha}\ v_{c\beta}]^T$, the current $\boldsymbol{i}_{\alpha\beta} = [i_\alpha\ i_\beta]^T$ and the grid voltage $\boldsymbol{v}_{g,\alpha\beta} = [v_{g\alpha}\ v_{g\beta}]^T$.

   Derive the discrete-time prediction model

$$\boldsymbol{i}_{\alpha\beta}(k+1) = \boldsymbol{A}\boldsymbol{i}_{\alpha\beta}(k) + \boldsymbol{B}_1\boldsymbol{u}_{abc}(k) + \boldsymbol{B}_2\boldsymbol{v}_{g,\alpha\beta}(k)$$

   using forward Euler discretization. Recall that $\boldsymbol{u}_{abc} = [u_a\ u_b\ u_c]^T$ denotes the three-phase switch position. Derive the numerical values of $\boldsymbol{A}$, $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ for $T_s = 100\,\mu$s.

3. Set the switch to "2" and implement the predictive current controller in the function currCtrl. Adopt the per unit system and use a prediction horizon of one step. To enumerate the set of three-phase switch positions, the simplest approach is to use three nested loops.

3.2 Predictive current control: Debugging

After having implemented the current controller, it will be debugged by analyzing its predictions and the minimization procedure.

1. Test the predictive controller with the settings $T_s = 100\,\mu$s and the penalty on switching $\lambda_u = 10 \cdot 10^{-3}$. What is the average switching frequency per device? At the very end of the Matlab function that implements the predictive current controller, compute the predicted current at the next time step when applying the chosen switch position. Provide the predicted current in the stationary reference frame and the per unit system, delay it by the controller sampling interval and compare it with the measured current. What do you observe?

2. Analyze the predictions in detail at one time step. To do so, use the script predCurrCtrl_analysis.m, which calls the predictive current controller in the static m-script currCtrl.m. For this part, copy your predictive current controller implementation from Step 3.1 into the m-script. Set a break-point in the predictive current controller after the minimization step. Analyze the predicted currents, the cost and the chosen switch position. Vary $\boldsymbol{u}(k-1)$ and $\boldsymbol{i}(k)$. What do you observe?

3.3 Predictive current control: Delays

In the following, the impact of delays on the controller behavior is investigated and a simple method to compensate for it is implemented. Open the Simulink model threePhase_grid_delay.slx. A delay of one controller sampling interval has been added to the input of the controller (see the red block). The current controller is provided in two versions (see the yellow blocks): one without delay compensation (Matlab function named currCtrl) and one with delay compensation (Matlab function named currCtrl_delay). Again the orange block provides the input to a multiport switch, which selects the controller: input "1" activates the controller without the delay compensation, while input "2" activates the version that compensates for the time delay.

1. Copy the current controller you designed in Step 3.1 (i.e., the controller which does not account for the delay, see function currCtrl) and set the switch input to "1". Run this controller with the settings $T_s = 100\,\mu$s and the penalty on switching $\lambda_u = 10 \cdot 10^{-3}$. What do you observe?

2. Set the switch to "2" and complete the predictive controller that compensates for the delay (Matlab function named currCtrl_delay) based on currCtrl. Due to the one-step delay in the system, the measurements and references in the interface section are shifted by one time step. To compensate for this, add a delay compensation to the predictive controller before running the simulations again. What do you observe?

3. Compare the predicted versus the actual currents with each other—both with and without delay compensation. Furthermore, do this for the case without a delay in the system (and a controller without delay compensation). What do you observe during steady-state operation? What do you observe during current reference steps?

3.4 Predictive current control: Grid voltage asymmetries

Last, we focus on the grid voltage. Open again the Simulink model threePhase_grid.slx (without a delay) and use the original predictive current controller (without delay compensation). Use the settings $T_s = 100\,\mu$s and $\lambda_u = 5 \cdot 10^{-3}$.

1. Simulate an asymmetry in the grid voltage by modifying the parameter relPhaseAmpl in the initialization file predCurrCtrl_ini.m. For example, set phase a to 0.5. How does the controller react to grid voltage asymmetries? Do these asymmetries affect the current waveforms?

2. The real and reactive power in the stationary orthogonal reference frame and the per unit system are defined as

$$P = v_{g\alpha} i_\alpha + v_{g\beta} i_\beta \,, \tag{1a}$$
$$Q = v_{g\alpha} i_\beta - v_{g\beta} i_\alpha \,. \tag{1b}$$

   Add a block to the Simulink model that computes the real and reactive power components. State the real and reactive power in the per unit system. What is the impact of grid voltage asymmetries on the real and reactive power?

3. Set the switch to "1" and implement the predictive power controller (powerCtrl). When doing this, the main consideration is to modify the objective function you used for the predictive

current controller. Instead of tracking the current reference, track the real and reactive power by penalizing the deviation of the predicted real and reactive power at the next time step from their references. This results in a predictive power controller.

What is the impact of grid voltage asymmetries on the real and reactive power? What is the impact on the three-phase currents? Is this outcome desirable? Distinguish between the case when one phase voltage is reduced and when it is increased. What happens when one phase voltage is very large, say above an amplitude of 1.2 p.u.?