Case Study 7: Feedback Linearization Control

This case study contains an application implementing the feedback linearization control which aims at maximizing the power captured by WECS. In order to drive the system to the optimal operating point using linear techniques (pole placement), a linearizing control input is computed. This application uses the mdl-file named `feedback_lin_ctrl.mdl`; it is described below.

Scope: Evaluation by numerical simulation of the control method aiming at maximizing the captured power for a low-power fixed-pitch PMSG-based wind turbine by using a linear controller built over a (exact) linearized system. The key feature of this method is to find a transform from the original state variables, denoted by $\mathbf{x}$, to new ones, denoted by $\mathbf{z}$. The objective is to maintain the tip speed ratio at its optimal value, which implies the use of a rotational speed loop. The control parameters imposing the dynamic of the linear I/O mapping are k_1, k_2 and k_I to be found in the block entitled `Linear I/O Control`.

Application analysis: The file `data_feedback_lin.m` contains the system data. The mdl-file is organized as follows. The plant is placed central-upper, the controller blocks are in the bottom-right corner of the diagram. The wind velocity synthesizer, placed in the upper-left corner, allows using a pre-computed non-stationary wind speed sequence (selection 4) for feeding the WECS; other sequences (including constant wind) are also available. At bottom-left corner one can see, from up to down: the system data loading button, the scopes and some computation/display blocks.
The plant is modelled using the simplified aerodynamic model (which provides the wind torque), the motion equation of the rigid drive train and the *d-q* modelling of the PMSG (providing i_d, i_q and the electromagnetic torque).

The controller is structured as follows. The linear control system is situated bottom-right, it provides the control component, u_v, and is fed by wind velocity and some of the new state variables; the control providing the linearizing input to the plant (bottom) is fed by the two Lie derivatives and the control component, u_v. The state transform and Lie derivatives computation are situated in the upper-right corner of the diagram.

Running the application:
Step 1: Load system parameters.
Step 2: Select from the menu "Simulation → Parameters" a convenient time horizon and proceed with the simulation.
Step 3: Select the pre-computed sequence of wind velocity (selection 4).
Step 4: Dynamical system behaviour can be overviewed by opening the scopes (upper-left corner of the application file). Operating point position and/or its standard deviation around the ORC can be plotted using the LSS mechanical power and rotational speed data (vectors `pwrl` and `omgl`) saved in the workspace from the associated scopes. Check the control performance by opening the power coefficient (`cp`) and the tip speed (`lam`) scopes, also.
Step 5: Change the wind velocity to constant value (selection 1) and change the wind mean value in steps of 1 m/s. The system dynamic behaviour can be seen on the HSS rotational speed (`omgh`) scope; the disturbance rejection can be assessed on the tip speed ratio or the power coefficient scopes.

Step 6: Change the control parameters, k_1, k_2 and/or k_I (from the block entitled `Linear I/O Control`) and re-evaluate the system behaviour. If the system becomes too slow, the operating point excursion around the ORC is larger (worse tracking performance). The dynamical behaviour can change significantly for some values of the control parameters. Try, for example, $k_1 = 3500$, $k_2 = 150$ and $k_I = 40000$, for step changes in the wind velocity.

Remarks: The intermediary parameters used for system modelling, state variable transform and Lie derivatives computation can be found in the file `data_feedback_lin.m`.

The PMSG is considered to be voltage-controlled by a rectifier-chopper pair, whose influence on the PMSG is represented by the equivalent resistance experienced by the generator. In the diagram `Rch` plays the role of R_S from the textbook.