Axial flux motor optimization example

Create a copy of *"[MotorXP-AFM files]\CustomScripts\runTSEMO_MXPAFM_emrax228.m"* with the name *runTSEMO_MXPAFM_18s16p_test.m*.

Create a copy of *"[MotorXP-AFM files]\CustomScripts\ evalDesigns_emrax228.m"* with the name *evalDesigns_test.m*.

Make the following changes to *runTSEMO_MXPAFM_18s16p_test.m*:

Line 25	`no_outputs = 3;  % number of objectives`
Line 26	`labelObjective = {'Weight, grams', 'Losses, watts', 'Magnet losses, watts'}; % objectives`

Lines 28-34	`bounds = [ % lower  upper`
	`200  250;  % inner diameter bounds [mm]`
	`25  43;  % slot width bounds [mm]`
	`20  42;  % stator height bounds [mm]`
	`15  40;  % magnet spacing bounds [el.deg]`
	`5  15;  % yoke height bounds [mm]`
	`5  15  % magnet height bounds [mm]`

Line 35	`1.5  2.5];  % air gap [mm]`
Line 40	`y_refdesign=[25829 2435.9];  % reference design weight (grams) and losses (watts) @(50 kW, 1600 RPM) - reference design red x on Pareto plot`

Make the following changes to *evalDesigns_test.m*:

Line 1	change function name to *evalDesigns_test*
Line 8	`motorProps=setParamafm(motorProps,'Coil fill factor',0.35);  % keep the same coil fill factor for all the designs`
Line 127	`settingsMagnetostatic=setParamafm(settingsMagnetostatic,'Mechanical speed',1600);  % RPM`
Line 130	`settingsMagnetostatic=setParamafm(settingsMagnetostatic,'RMS supply current',2000);  % initial current value`
Lines 132-133	`targetTorque = 50000/(1600*2*pi/60);  % N*m` `initCurrent = 2000;`

Lines 183-188	`settingsMagnetostatic=setParamafm(settingsMagnetostatic,'Simulation time','One electrical period');`
	`nPolePairs = getParamafm(motorProps,'Number of pole pairs');`
	`nSlots = getParamafm(motorProps,'Number of slots');`
	`settingsMagnetostatic = setParamafm(settingsMagnetostatic, 'Number of points', setNumberOfPoints('Low accuracy',nPolePairs,nSlots));`
	`% settingsMagnetostatic=setParamafm(settingsMagnetostatic,'Time from',0);`
	`% settingsMagnetostatic=setParamafm(settingsMagnetostatic,'Time to',T1/6);`
	`% 1/6th of electrical period`

Line 202	`magnetLoss = inf*ones(nDesigns,1);`
Line 217	`magnetLoss(n) = Results.MagnetLoss;`

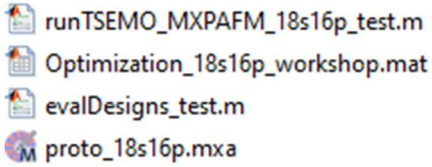
Line 224	`Objectives = [1000*motorWeight motorLoss_norm magnetLoss];`
Line 213	`motorLoss(n) = Results.Ps + Results.Piron_hyst + Results.Piron_eddy + Results.MagnetLoss + Results.OtherECloss;`

Lines 218-221	`if motorLoss_norm(n)>6000`
	`% punish algorithm for losses > 6000W`
	`motorWeight(n) = 50;`
	`end`

Line 364	`airGap = designParams(i,7); % air gap`
Line 365	`outerDiameter = 350; % motor outer diameter`
Line 381	`motorProps=setParamafm(motorProps,'Air gap',airGap);` `% air gap`

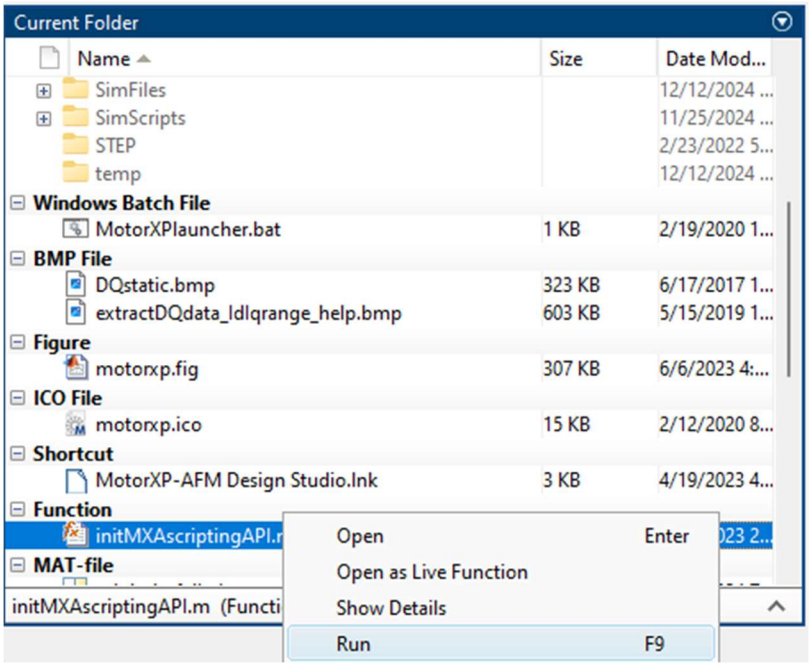
- Adding minimization of magnet losses as third optimization objective
- Adding air gap as an optimization variable
- Changing simulation time to one electrical period

All files related to this AFM optimization example can be found in "Workshop2024\day1\AFM":



To view the optimization results:

First run file *initMXAscriptingAPI.m* from the MotorXP-AFM folder



Then type in the MATLAB Command Window (correct the path to the file):