Purdue University School of Electrical and Computer Engineering

Power and Energy Devices and Systems Area

Power Magnetic Material Toolbox

Version 1.0, February 2010

Coordinator: S.D. Sudhoff (sudhoff@purdue.edu)

I. INTRODUCTION

The Power Magnetic Material Toolbox is a Matlab based code containing routines to support the design of power electromagnetic and electromechanical devices such as inductors, transformers, motors, and generators. The routines encapsulate many of the electrical, thermal, and magnetic properties of conductors and magnetic materials. Specifically, the routines in the toolbox contain information on American Wire Gauge (AWG) dimensions [1]; material properties of copper, aluminum, silver, and gold for use as conductors [2]-[6]; material properties of select rare earth Neodymium-Iron-Boron (NdFeB) and Samarium Cobalt (SmCo) permanent magnets; and parameters of select ferrites and silicon steels [7],[8] as characterized using the method in [9]. Furthermore, routines used to calculate the DC and AC resistance of a wire [10], the loss density of a variety of ferromagnetic and ferromagnetic materials based on the Modified Steinmetz Equation (MSE) [11], and the permeability as a function of flux density [9] are included. The information from these routines is directly compatible with Purdue’s Magnetic Equivalent Circuit Toolbox and Thermal Equivalent Circuit Toolbox.

II. MATERIAL TOOLBOX ROUTINES

In this section, the routines in the Power Magnetic Material Toolbox will be explained. The routines are given below in alphabetical order:

1. awg_catalog
   Function call: [WP] = awg_catalog(n)
   Purpose: Assigns wire parameters based on the AWG.
   Inputs: n – wire number
   Outputs: WP – data structure of wire parameters
   - WP.desc – gauge of wire (AWG)
   - WP.area – wire area (m^2)
   - WP.diam – wire diameter (m)

2. awg_floor
   Function call: [WP] = awg_floor(rw)
   Purpose: Assigns wire parameters by determining the AWG with the closest radius less than the desired wire radius.
   Inputs: rw – desired wire radius
   Outputs: WP – data structure of wire parameters
   - WP.desc – gauge of wire (AWG)
   - WP.area – wire area (m^2)
   - WP.diam – wire diameter (m)
   - WP.rad – wire radius (m)

3. awg_round
   Function call: [WP] = awg_round(rw)
   Purpose: Assigns wire parameters by determining the AWG with radius closest to the desired wire radius.
   Inputs: rw – desired wire radius (m)
   Outputs: WP – data structure of wire parameters
   - WP.desc – gauge of wire (AWG)
   - WP.area – wire area (m^2)
   - WP.diam – wire diameter (m)
   - WP.rad – wire radius (m)

4. conductor_catalog
   Function call: [CP] = conductor_catalog(n)
   Purpose: Assigns conductor parameters.
   Inputs: n – conductor number
   - n=1 – copper
   - n=2 – aluminum
   - n=3 – silver
   - n=4 – gold

   Outputs: CP – data structure of conductor parameters
   - CP.desc – conductor material description
   - CP.t0 – temperature at which conductivity characterized (°C)
   - CP.sigma0 – conductivity at t0 (1/(Ohm-m)

CP.sigmac	–	same as CP.sigma0 (1/(Ohm-m)) (retained for compatibility)
CP.alpha	–	temperature coefficient of resistivity (1/K)
CP.jmax	–	recommended maximum current density (A/m^2)
CP.row	–	mass density (Kg/m^3)
CP.k	–	thermal conductivity (W/(K-m))

5. ferrite_catalog

Function call: [FP] = ferrite_catalog(n)

Purpose: Assigns ferrite parameters.

Inputs:	n	–	ferrite number
	n=1	–	MN8CX
	n=2	–	MN60LL
	n=3	–	MN67
	n=4	–	MN80C
	n=5	–	3C90
Outputs:	FP	–	data structure of ferrite parameters
	FP.desc	–	ferrite material description
	FP.Msat	–	saturated magnetization (T)
	FP.mur	–	relative permeability in linear region
	FP.Blim	–	recommended limit on B to avoid saturation (T)
	FP.row	–	mass density (kg/m^3)
	FP.BH	–	data structure of BH curve parameters
	FP.BH.m	–	magnetization coefficients (T)
	FP.BH.n	–	exponents
	FP.BH.h	–	field intensity breakpoints (A/m)
	FP.MSE	–	data structure of MSE loss parameters
	FP.MSE.alpha	–	frequency exponent
	FP.MSE.beta	–	flux density exponent
	FP.MSE.kh	–	hysteresis loss coefficient (J/m^3)
	FP.MSE.ke	–	eddy current loss coefficient ($J\text{-}s/m^3$)
	FP.muB	–	data structure of mu(B) parameters
	FP.muB.mur	–	initial relative permeability of anhysteretic curve
	FP.muB.a	–	vector of alpha coefficients (1/T)
	FP.muB.b	–	vector of beta exponential coefficients (1/T)
	FP.muB.g	–	vector of gamma exponential offsets (T)
	FP.muB.d	–	vector of delta coefficients
	FP.muB.e	–	vector of epsilon values
	FP.muB.z	–	vector of zeta values
	FP.muB.h	–	vector of eta values (1/T)
	FP.muB.t	–	vector of theta values

6. loss_density

   Function call: [Pld] = loss_density(B,pB,wr,SP)

   Purpose: Finds the loss density for a steel with a given flux density waveform. Note the flux density waveform must be over an integer number of cycles. It must also be evenly spaced and start and end on the same point in cycle. Model based on a modified version of the generalized Steinmetz model.

   Inputs:
   - B – B field at corresponding electrical rotor positions (T)
   - pB – derivative of B with respect to electrical rotor position (T/rad)
   - wr – electrical rotor speed (rad/s)
   - SP – data structure of steel parameters
     - SP.MSE.alpha – frequency exponent
     - SP.MSE.beta – flux density exponent
     - SP.MSE.kh – hysteresis loss coefficient (J/m^3)
     - SP.MSE.ke – eddy current loss coefficient ($J\text{-}s/m^3$)

   Outputs:
   - Pld – power loss density (W/m^3)

7. MSE_loss

   Function call: [Pld] = MSE_loss(B,qr,wr,SP)

   Purpose: Finds the loss density for a steel with a given flux density waveform. Model based on a modified version of the generalized Steinmetz model.

   Inputs:
   - B – B field as electrical rotor position is varied from 0 to 2π (T)
   - qr – electrical rotor position from 0 to 2π (rad)
   - wr – electrical rotor speed (rad/s)
   - SP – data structure of steel parameters
     - SP.MSE.alpha – frequency exponent
     - SP.MSE.beta – flux density exponent
     - SP.MSE.kh – hysteresis loss coefficient (J/m^3)
     - SP.MSE.ke – eddy current loss coefficient ($J\text{-}s/m^3$)

   Outputs:
   - Pld – power loss density (W/m^3)

8. muB

   Function call: [mu,pmu] = muB(MP,B)

   Purpose: Calculates permeability as a function of flux density.

   Inputs:
   - MP – data structure of material parameters
     - MP.muB.mur – initial relative permeability of anhysteretic curve
     - MP.muB.a[] – vector of alpha coefficients (1/T)
     - MP.muB.b[] – vector of beta exponential coefficients (1/T)
     - MP.muB.g[] – vector of gamma exponential offsets (T)
     - MP.muB.d[] – vector of delta coefficients
     - MP.muB.e[] – vector of epsilon values
     - MP.muB.z[] – vector of zeta values

	MP.muB.h[]	–	vector of eta values (1/T)
	MP.muB.t[]	–	vector of theta values
	B	–	vector of points at which to calculate permeability (T)
Outputs:	mu	–	permeability at points corresponding to B (H/m)
	pmu	–	derivative of mu with respect to B (H/(m-T))

9. pm_catalog

Function call:	[MP] = pm_catalog(n)		
Purpose:	Assigns permanent magnet parameters.		
Inputs:	n	–	permanent magnet number
	n=1	–	NdFeB-4SB
	n=2	–	NdFeB-30
	n=3	–	NdFeB-48
	n=4	–	SmCo-B15S
Outputs:	MP	–	data structure of permanent magnet parameters
	MP.desc	–	permanent magnet description
	MP.brm	–	residual flux density (T)
	MP.xpm	–	susceptibility
	MP.hmmn	–	minimum field intensity before the permanent magnet is demagnetized (A/m)
	MP.row	–	mass density (kg/m^3)

10. resistance

Function call:	[rdc,rac] = resistance(ac,lc,Np,f,sigma,ur)		
Purpose:	Determines the AC and DC resistances of a wire. Note the resistances are calculated based on an insulated, solid conductor within the wire. A wire may consist of multiple insulated, solid conductors.		
Inputs:	ac	–	cross-sectional area of an insulated, solid conductor within the wire (m^2)
	lc	–	length of conductor (m)
	Np	–	number of insulated, solid conductors in parallel that form the wire
	f	–	vector of frequencies (Hz)
	sigma	–	conductivity of conductor (1/(Ohm-m))
	mur	–	relative permeability of conductor
Outputs:	rdc	–	DC resistance of wire (Ohms)
	rac	–	vector of AC resistances of the wire as a function of frequency (Ohms)

11. steel_catalog

Function call: [SP] = steel_catalog(n)

Purpose: Assigns silicon steel parameters.

Inputs:	n	–	steel number
	n=1	–	M19
	n=2	–	M36
	n=3	–	M43
	n=4	–	M47
Outputs:	SP	–	data structure of silicon steel parameters
	SP.desc	–	silicon steel material description
	SP.Msat	–	saturated magnetization (T)
	SP.mur	–	relative permeability in linear region
	SP.Blim	–	recommended limit on B to avoid saturation (T)
	SP.row	–	mass density (kg/m^3)
	SP.k	–	thermal conductivity (W/(K-m))
	SP.c	–	specific heat capacity (J/(kg-K))
	SP.BH	–	data structure of BH curve parameters
	SP.BH.m	–	magnetization coefficients (T)
	SP.BH.n	–	exponents
	SP.BH.h	–	field intensity breakpoints (A/m)
	SP.MSE	–	data structure of MSE loss parameters
	SP.MSE.alpha	–	frequency exponent
	SP.MSE.beta	–	flux density exponent
	SP.MSE.kh	–	hysteresis loss coefficient (J/m^3)
	SP.MSE.ke	–	eddy current loss coefficient (J-s/m^3)
	SP.muB	–	data structure of mu(B) parameters
	SP.muB.mur	–	initial relative permeability of anhysteretic curve
	SP.muB.a	–	vector of alpha coefficients (1/T)
	SP.muB.b	–	vector of beta exponential coefficients (1/T)
	SP.muB.g	–	vector of gamma exponential offsets (T)
	SP.muB.d	–	vector of delta coefficients
	SP.muB.e	–	vector of epsilon values
	SP.muB.z	–	vector of zeta values
	SP.muB.h	–	vector of eta values (1/T)
	SP.muB.t	–	vector of theta values

III. INSTALLATION INSTRUCTIONS

The Power Magnetic Material Toolbox can be used in Matlab by adding the location of the ‘PMMT’ subdirectory of the toolbox to the Matlab path. To add the Power Magnetic Material Toolbox to the Matlab path, click on the *File* drop down menu from the Matlab startup screen and then click on *Set Path*. From there a new dialogue window will appear. Next, click on the *Add Folder* button. The specific location of the folder in the computer can now be selected.

IV. CONTRIBUTORS

The contributors for this Power Magnetic Material Toolbox include Purdue University faculty Prof. Scott D. Sudhoff; Purdue University graduate student Grant Shane and former graduate students James Cale and Brandon Cassimere.

V. ACKNOWLEDGMENTS

Sponsors for this work include the Office of Naval Research, grants N00014-08-1-0080 (Electric Ship Research and Development Consortium) and N00014-08-1-0397, and the U.S. Army, contract DAAB07-03-D-B009/0083.

VI. REFERENCES

[1] Herrington, D. E., Handbook of Electronic Tables and Formulas. Indianapolis, IN: Howard W. Sams & Co., 1959.

[2] Fink, D. G. (ed.), and H. W. Beaty (ed.), Standard Handbook for Electrical Engineers. 13th ed. New York: McGraw-Hill, Inc., 1993.

[3] Lide, D. E. (ed.), CRC Handbook of Chemistry and Physics. 84th ed. Boca Raton, FL: CRC Press, 2003.

[4] Serway, R. A., Principles of Physics. 2nd ed. Fort Worth, TX: Saunders College Pub., 1998.

[5] Stauffer, H. B., Engineer's Guide to the National Electric Code. Boston, MA: Jones and Bartlett Publishers, Inc., 2008.

[6] Young, H. D., University Physics. 7th ed. Reading, MA: Addison-Wesley, 1992.

[7] ASM International Metals Handbook, Properties and Selection: Nonferrous Alloys and Special-Purpose Materials. vol. 2, 10th ed., 1990.

[8] Mapes and Sprowl Steel, Elk Grove Village, IL, private communication, 2010.

[9] Shane, G. and S. D. Sudhoff, "Refinements in Anhysteretic Characterization and Permeability Modeling," *IEEE Transactions on Magnetics*, submitted for publication.

[10] Ramo, S., J. R. Whinnery, and T. Van Duzer., Fields and Waves in Communication Electronics. New York: John Wiley & Sons, Inc., 1965.

[11] Sudhoff, S. D., ECE 695 (Lecture Set 4: Modeling Magnetic Materials). West Lafayette, IN: Purdue University, Dept. of Electrical and Computer Engineering, 2009.

VII. LICENSE