MATGMD

Gandhali Prakash Juvekar
Texas A&M University
Department of Electrical and Computer Engineering

Contents

0.1 Introduction

0.1.1 Background

MATGMD [1] is a collection of MATLAB® function files for studying Geomagnetic Disturbance (GMD). MATGMD is the result of an inspiration drawn from MATPOWER which is a tool for Steady-State Operations, Planning, and Analysis [1]. This tool for researchers and educators as well as the industry. It is easy to use and modify. MATGMD is designed to give the best performance possible while keeping the code simple to understand. The tool can be found at:
https://github.com/gandhalijuvekar/MATGMD

It focuses on extracting the Geomagnetically Induced Currents (GICs) at transformer neutrals. It uses a step-by-step procedure for obtaining these currents and also provides the ability to obtain the injection currents at each bus, the GIC DC neutral voltage at every node (substation/bus), the substation neutral GICs, the ratios of the transformer winding currents to the transformer neutral GICs and effective GICs that can be used to obtain losses in transformers as a result of GICs.

The initial need for a MATLAB based tool for GMD was born out of the requirements of a GIC-inclusive state estimator design for GMD events. MATGMD can be used for mitigation and transient stability studies for GMD events which makes it a crucial tool for analyzing effects of GMD on the bulk power system and modeling the event from the power system perspective. This is the first version release of the tool and it continues to be developed and improved.

0.1.2 Citing MATGMD

We request that publications derived from the use of MATGMD explicitly acknowledge that fact by citing reference [2].

G. P. Juvekar and K. Davis, "MATGMD: A tool for Enabling GMD Studies in MATLAB," in Texas Power and Energy Conference (TPEC), 2019. Yet to appear.

0.2 Getting Started

0.2.1 System Requirements

To use MATGMD you will need:

- MATLAB version 7 (R14) or later[1], or
- GNU Octave version 3.4 or later[2]

For the hardware requirements, please refer to the system requirements for the version of MATLAB[3] or Octave that you are using. In this manual, references to MATLAB usually apply to Octave as well.

0.2.2 Installation

Installation and use of MATGMD requires familiarity with the basic operation of MATLAB, including setting up your MATLAB path.

1. Download the zipped file MATGMD.zip from https://github.com/gandhalijuvekar/MATGMD which is a repository.
2. Unzip the downloaded file. Move the resulting directory to the location of your choice. These files should not need to be modified, so it is recommended that they be kept separate from your own code.
3. These files can be replicated in any folder of your choice and can be used to perform GMD studies accordingly.
4. Test files have been given as examples to explain the sequence of use for the function files.

[1] MATLAB is available from The MathWorks, Inc. (http://www.mathworks.com/). Matlab is a registered trademark of The MathWorks, Inc.

[2] GNU Octave [2] is free software, available online at http://www.gnu.org/software/octave/.

[3] http://www.mathworks.com/support/sysreq/previous_releases.html

0.2.3 Running a Simulation

The primary functionality of MATGMD is to obtain system parameter values for GMD events focusing primarily on GICs. This involves:

1. preparing the input data defining all of the relevant power system dc parameters required for GMD studies like dc line conductance, transformer high and low side dc resistance, substation location information, etc.,
2. calling the appropriate function files to run the simulation, and
3. viewing and accessing the results that are printed to the screen and/or saved in output data structures or files.

0.2.4 Preparing Case Input Data

The input data for the case to be simulated are specified in a set of data matrices packaged as the fields of the input function file. A standard input file of type .m should include the following mandatory data matrices and the data in these matrices should be ordered in columns as mentioned below for every system component:

- **Xmr**: transformer dc information - high side and low side nominal voltage (kV), high side and low side bus number, high side and low side dc winding resistance, autotransformer? (1/0), wye-wye? (1/0), Substation number. Additionally, data like the transformer scalar parameter 'k' can also be added if there is knowledge about it.
- **Bus_dc**: bus dc information - Bus number, Substation number, Nominal voltage (kV).
- **Substn**: substation dc information - Substation number, grounding resistance, Latitude, Longitude.
- **Lines**: transmission line dc information - From bus, To bus, dc line conductance, Circuit number.

Optionally, **Ckt_n**: circuit numbers for combined transformer and transmission line branches, can be an output field depending on its necessity. For a uniform electric field analysis, **E**: directional magnitudes of the uniform electric field in the north and east direction, is also an output field.

GMD analysis using MATGMD for a non-uniform electric field requires the electric field data to be given as an input through an M-file. The output fields for this file should be

- **Ematrix**: electric field matrix with directional magnitudes (north and east directions) of the electric field in zones defined by latitudes and longitudes.

Lat	Lon: -88.000	Lon: -86.000	Lon: -84.000	Lon: -82.000
35.000	0.500, 1.000	0.500, 1.000	1.000, 1.000	1.000, 1.000
34.000	0.500, 1.000	0.500, 1.000	1.000, 1.000	1.000, 1.000
33.000	0.500, 1.000	0.500, 1.000	1.000, 1.000	1.000, 1.000
32.000	0.500, 1.000	0.500, 1.000	1.000, 1.000	1.000, 1.000

Figure 1: Non-uniform electric field case as it appears on PowerWorld using .b3d files

- **del**: the step size for changing the latitudes and longitudes at which the electric field is defined.
- **lmin**: the vector with the minimum value of the latitude and longitude.
- **lmax**: the vector with the maximum value of the latitude and longitude.

```
function [Ematrix, del, lmin, lmax] = EPRI20Eip
%% input
%GIC3DMatrixRow
%Lon:  -88.000  Lon:  -86.000   Lon:  -84.000   Lon:  -82.000
Ematrix = [
    0.500, 1.000    0.500, 1.000    1.000, 1.000    1.000, 1.000
    0.500, 1.000    0.500, 1.000    1.000, 1.000    1.000, 1.000
    0.500, 1.000    0.500, 1.000    1.000, 1.000    1.000, 1.000
    0.500, 1.000    0.500, 1.000    1.000, 1.000    1.000, 1.000
    ];

%% step size
%    ydel xdel
del = [1; 2];

%% min values
%    ymin  xmin
lmin = [32; -88];

%% max values
%    ymax  xmax
lmax = [35; -82];
```

Figure 2: Non-uniform electric field input case on MATLAB

The function file, **ext2int_dc.m** can be used to renumber the buses in a consecutive manner if the **Bus_dc** matrix indicates that the buses in the system are not numbered consecutively.

0.3 System Modeling

0.3.1 G matrix

Let N^b be the number of buses for a transmission system. These buses do not include the ones which are connected to the generators grounded via the nearest substation. Let N^s be the number of substations in the system considered. Together, the buses and substations form a set of nodes, $N : N^b \cup N^s$. Let the set of branches between the nodes, n and m where, $(n,m) \subseteq N^b \times N^b$, be denoted by $\mathscr{L}$. This includes transmission lines and transformer branches which are the autotransformer series windings. Let the set of transformer windings, which include the high side winding for a delta–wye transformer, the common winding for an autotransformer and both the windings for a wye–wye transformer, between the bus nodes and substation nodes, n and m where, $(n,m) \subseteq N^b \times N^s$, be denoted by $\mathscr{X}\mathscr{L}$. Together, $\mathscr{L}$ and $\mathscr{X}\mathscr{L}$ constitute the edges, ε represented as $(n,m) \subseteq N \times N$. Let the equivalent conductance between nodes n and m for any $(n,m) \in \varepsilon$ be $g_{nm} = g_{mn}$. Similarly, grounding conductance of the substation neutral can be represented using g_{nn} for $n \in N^s$. The symmetric and real matrix, G which is an $N \times N$ matrix can be constructed using these notations. These conductance quantities account for all the three phases in parallel. The elements of G can be given by:

$$G_{mn} = \begin{cases} -g_{mn}, & \text{if } (n,m) \subseteq \varepsilon \\ \sum_{v \in N_n} g_{nv}, & \text{if } n = m \subseteq N^b \\ g_{nn} + \sum_{v \in N_n} g_{nv}, & \text{if } n = m \subseteq N^s \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

The MATLAB function file for the dc conductance matrix is **G_mat.m**. The above responses for each condition are implemented in the code for the function file that gives the output matrix **g**. This matrix is similar to the reactance matrix **Y**, but with substation nodes included and the dc conductance values of the system used.

0.3.2 Injection currents

The potential induced due to the electric field at every transmission line is a voltage source that can be modeled as:

$$V_{nm} = E^N L^N_{nm} + E^E L^E_{nm} \tag{2}$$

where, the dc potential that is induced is V_{nm} (V), the distance for the line between buses n and m in the north and east directions are L^N_{nm} and L^E_{nm} (km), respectively. Using the Norton's equivalent, this induced voltage can be converted to a dc current injection at each node connecting these lines and is given by:

$$I_{nm} = V_{nm}/R_{nm} \tag{3}$$

where, the dc injection current to bus n is I_{nm} (Amps) and the dc resistance on the line between buses n and m is R_{nm} (ohms). The equivalent current injection I_{mn} to bus m equals to $-I_{nm}$ (Amps). The summation of all the current injections at bus n is the overall injection current at a bus n given by:

$$I_n = \sum_{(n,m)\in L} I_{nm} = \sum_{(n,m)\in L} \frac{E^N}{R_{nm}} L^N_{nm} + \frac{E^E}{R_{nm}} L^E_{nm} \tag{4}$$

The injected current is simply $I_n = 0$, $\forall n \in N^s$ because the substations are not directly connected by transmission lines. Stacking eqn. 4 for all the N nodes:

$$\mathbf{I} := [I_1, ..., I_N]^T \tag{5}$$

The MATLAB function file for the computation of the injection currents for a uniform electric field is **InjC.m** and for a non-uniform electric field is **VarEFV.m**. The above described equations are implemented in the code for the function file mentioned. The calculations for the injection currents involve a simple matrix multiplication between the directional electric field values and the directional lengths of the line for a uniform electric field case. There is an elaborate process for the non-uniform electric field case which involves breaking down the lines into various parts corresponding to the electric field zone they fall in. Then, for every part of the line, the directional length is multiplied with the directional electric field and summed to be later divided by the dc resistance of that line. Directional lengths for each line can be computed by using the location coordinates and the distance formula [?]:

$$L^N = (111.113 - 0.56 * cos(2\phi))\Delta lat \tag{6}$$

$$L^E = (111.5065 - 0.1872 * cos(2\phi)) * cos(\phi)\Delta long \tag{7}$$

where, L^N and L^E are the North and East directional lengths (km). Here, the difference in the latitudinal and longitudinal coordinates of the from and to substations are given by Δlat and $\Delta long$ respectively. The value of ϕ is given as,

$$\phi = \frac{latA + latB}{2}$$

0.3.3 DC Node Voltages

Calculating the dc node voltages is a vital intermediary step for the computation of the system GICs and related values. Let the dc voltage at any bus or substation n be V_n (Volts). We get $\mathbf{V} := [V_1, V_2, .., V_N]^T$ by concatenating all the nodes. From the dc power flow model, **G** and eqn. 5, we get:

$$\mathbf{V} = \mathbf{G^{-1}I} \tag{8}$$

The MATLAB function file that implements the equation for the computation of the dc node voltages is **Vdcnodes.m**

0.3.4 Substation and Transformer GICs

The GIC flowing at every substation n can be obtained by:

$$I_n^s = g_{nn}V_n, \forall n \in N^s \tag{9}$$

At the transformer level, the high-side and low-side transformer GICs can be obtained. The current flowing through the wye side of a delta–wye transformer, either side for a wye–wye transformer or the common winding of an autotransformer, from bus n to substation m via the transformer in-between, is given by:

$$I_{nm}^T = g_{nm}(V_n - V_m), \forall n \in N^b, m \in N^s \tag{10}$$

The current flowing through the series winding for an autotransformer is given by:

$$I^T_{nm} = g_{nm}(V_n - V_m), \forall n \in N^b, m \in N^b \tag{11}$$

where the high-side bus and the low-side bus are denoted by n and m respectively. The transformer neutral current in the summation of the high side and the low side currents calculated using eqn. 10 and 11. The neutral current, I^T_n is given by:

$$I^T_n = (I_H + I_L) * 3, \tag{12}$$

where, the per phase high-side and the low-side currents for a transformer are I_H and I_L respectively. Additionally, r_h and r_l correspond to the ratios of the high side and the low side currents to the transformer neutral current respectively:

$$r_h = \frac{I_H * 3}{I^T_n}, r_l = \frac{I_L * 3}{I^T_n} \tag{13}$$

The MATLAB function file that implements the equations for the computation of various GIC values is **XmrI.m**

0.3.5 Effective GICs

The additional reactive power loss (Q_{loss}) in a transformer during a GMD are linearly dependent on the value of the effective GIC for the corresponding transformer. These effective GICs depend on the transformer turns ratio, a. Let t be the transformer under consideration. The effective GIC per phase $I_{eff,t}$ for the corresponding transformer is given by:

$$I_{eff,t} = | I_{H,t} + \frac{I_{L,t}}{a_t} |$$

where, $I_{H,t}$ and $I_{L,t}$ are the transformer high side and low side currents per phase respectively and a_t is the transformer turns ratio. The equation can also be expressed as:

$$I_{eff,t} = | I^T_n (r_h + \frac{r_l}{a_t}) | \tag{14}$$

The MATLAB function file that implements the equation for the computation of effective GIC values is **Igiceff.m**

0.3.6 Test case file

The outputs of the test files can be customized as per the need of the user. The study that needs to be performed decides the kind of parameters or values that need to be extracted from the tool since various GMD studies target unique system values.

The test files need to declare three inputs:

1. The name of the input file pertaining to the dc information of the system,
2. the name of the electric field input file if the field is non-uniform. If the field is uniform, enter 'nil' as the case name, and
3. a flag called 'isuniform' that informs if the electric field is uniform or not.

These inputs are then send to the function file **runGMD.m** which contains all the function files that help compute the dc conductance matrix, injection currents, dc node voltages, neutral currents and their ratios, and the effective GICs. The outsput of this function file can be altered as needed.

The function files **loadGMD.m** and **loadEfield.m** load the input matrices corresponding to the dc system components and the non-uniform electric field respectively.

Here is an example of a test file

```
function testEPRI20DC_E
%% define named indices into data matrices
[HKV, LKV, HSB, LSB, H_RES, L_RES, AUT, YY, TR_S, CN_TR, KVAL, INEU] = idx_xmr;

%% input file
caseGMD = 'EPRI20_DC4.m';
caseEfield = 'EPRI20Eip_2';
isuniform = 0;

%% Run GMD
[et, In, rt, Igeff] = runGMD(caseGMD, caseEfield, isuniform);

%% output
[Xmr, Bus_dc, Substn, Lines] = loadGMD(caseGMD);
```

Figure 3: An example test file

REFERENCES

[1] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, “Matpower: Steady-state operations, planning, and analysis tools for power systems research and education,” *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, Feb 2011.

[2] G. P. Juvekar and K. Davis, “Matgmd: A tool for enabling gmd studies in matlab,” in *Texas Power and Energy Conference (TPEC)*, Feb. 2019.