

Healing Touch for Power Systems (HITS)

Version 2.0

Zhijun Qin,
Qinfei Long,
Jiazuo Hou,
Zhicheng Xu,
Yunhe Hou

The University of Hong Kong

Insert appropriate Auto Text License Entry. If license is copyright, please delete.

SOFTWARE DESCRIPTION

The Healing Touch for System (HITS) version 2.0 is a MATLAB toolbox that can establish a system restoration plan following a complete or partial outage, also can detect the vulnerable line set. It is updated based on the previous version, making the whole restoration process more visualized.

Description

The HITS is a MATLAB toolbox that can establish a system restoration plan following a complete or partial outage and detect the vulnerable line set. Users can configure the characteristics of generating units, loads, transmission lines, and transformers. After the data are loaded, the restoration process can be established automatically. The users can also control the restoration process by setting milestones and priority of each generating unit and load. The plan can be displayed in MATLAB Command Window and some key results can be shown in the GUI result interface, and all the results are stored in a structural variable named “Grid”. Moreover, the new version enables users to detect the vulnerable line set in system and generate the influence graph matrix, indicating the system vulnerability.

Benefits and Value

Value and benefits provided by the Generic Restoration Milestones are:

- Establishing of a system restoration planning from complete or partial outage.
- Evaluating a system restoration plan and checking constraints during restoration process.
- Detecting the vulnerable line set in system and generating the influence graph matrix.
- Displaying a system restoration plan in MATLAB Command Window and some key results can be shown in the GUI result interface.

HITS Development

Below is the highlight of the change since version 2.0 of HITS.

- Providing a GUI interface exhibiting the simulation results, convenient for users’ understanding.

- Detecting the vulnerable line set in system and generating the influence graph matrix based on a series of $N - 1$ cascading failure tests.
- Some bugs have been fixed.

Requirements

HITS 2.0 can run on Windows XP, Windows Vista and Windows 7 (32-Bit and 64-Bit) and on MATLAB 7.0 or higher. (PS: Much updated MATLAB version may not be used for HITS. With experience, the available higher version is MATLAB 2013.)

ABSTRACT

The developed HITS program is a MATLAB toolbox that can establish a system restoration plan following a complete or partial outage and detect the vulnerable line set. The plan can be displayed in MATLAB Command Window and some key results are shown in the GUI result interface, and all the results are stored in a structural variable named “Grid”. Users can also control the restoration process by setting milestones and priority for each generating unit and load.

This user’s manual provides:

- Instructions for installing and running HITS,
- Detailed explanation for the operation of the program.

It is assumed that the user is familiar with Microsoft (MS) Windows operations and MATLAB environment. The first section provides an introduction to HITS. Section 2 gives the installation instruction for the toolbox. Section 3 describes how to use HITS.

Keywords

Power System Restoration

1 INTRODUCTION TO HITS

Power system restoration is well recognized as one of the most important tasks for electric power grids. Following a power outage, dispatchers in the control center work with the field crews to re-establish the generation and transmission systems and then to pick up load and restore service. It is reported that the impact of a blackout increases with the duration of its restoration. This program utilizes “generic restoration milestones” and is able to establish a system restoration plan following a complete or partial outage and detect the vulnerable line set. This power system restoration plan can be established automatically and interactively. The following data (*.txt, *.xls, *.xlsx, *.m) are required:

- Topology and parameters of a power grid
- Initial state before system restoration
- Characteristics of each generating unit
- Characteristics of each load
- Characteristics of each branch
- Characteristics of each transformer

The program has the following functional features:

- The program establishes a system restoration plan following a complete or partial outage and detect the vulnerable line set.
- Editable file (*.txt, *.xls, *.xlsx, *.m) allows the user to create new data files, enter or modify the input parameters. Data can also be input into tables of the GUI main interface by the keyboard.
- Calculation results are displayed in MATLAB Command Window and stored in a structural variable named “Grid”.

2 BEFORE USING HITS

This section describes the computer system requirements and installation procedures needed to use the HITS program.

2.1 Installation of HITS at Client Site

2.2 If You Experience Difficulties Accessing the Application

2.3 Minimum Computer Requirements

A PC with the following system requirements:

- Windows XP, Windows Vista, Windows 7 (32-Bit and 64-Bit) and Windows 10 (32-Bit and 64-Bit)
- 128 Megabytes RAM
- Monitor Resolution: 1024×768

MATLAB version requirements:

- MATLAB 7.0 or higher

MATLAB toolbox requirements:

- MATPOWER 7.0

2.4 Installation of HITS

The HITS is installed by copying the “HITS_V2.0” folder (with its contents) to your system hard driver (e.g., C:). Also, add the whole subfolders in HITS_V2.0 to MATLAB search path.

2.5 Uninstall HITS program

To uninstall the HITS program:

Delete folder “HITS_V2.0” from your hard driver.

3 OVERVIEW: THE HITS USERS INTERFACE

3.1 Directory Structure

After users have installed the HITS successfully, they will find five folders and seven *.m files under the setup directory as shown in Figure 3-1.

- GridData: contains the power system grid data files and their parsing files.
- GriData1: contains the power system grid data displayed on the tables of the main GUI interface.
- GRM1: contains the key functions of HITS.
- Initialization: contains the HITS variables initialization file and launch information file.
- OPF: contains the optimal power flow files.
- StepValue: update the variables step by step.
- GRMs_FileNameDef.m: defines data files' names and function files' names.
- Random_Edge_Cascading: the cascading failure simulation.
- Relationship_Graph.m: the influence graph simulation.
- hits.m: the main algorithm file.
- hits.fig: the main GUI file.
- hits1.fig: the result GUI file.
- hits2.fig: the 'about' information GUI file.

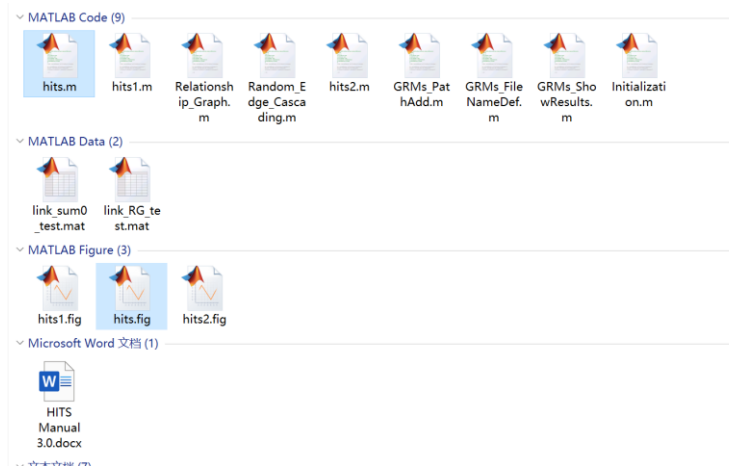


Figure 3-1 Program Files

3.2 Sample Data

The sample data files:

- RTS24_area.txt
- RTS24_bus.txt
- RTS24_criticalLoad.txt
- RTS24_dispatchableLoad.txt
- RTS24_gen.txt
- RTS24_line.txt
- RTS24_tran.txt
- case24_ieee_rts.m

can be found under the “...\HITS_V2.0\GridData\RTS24” directory. The following chapter will show how to use the HITS program with the sample data.

3.3 Launching HITS

(1) “hits.m” is the main file of HITS, users can start HITS program by double clicking “hits.m” in folder, then clicking “Run” in MATLAB EDITOR view. Or the users can launch the MATLAB first, set the MATLAB current path to HITS setup directory, input *hits* in the command window, then press “Enter” to start HITS.

(2) The users can also start from the graphic user interface by clicking in MATLAB HOME view or input GUIDE in the command window and press “Enter” to open Existing GUI named hist.fig, hist1.fig and hist2.fig. hist.fig is the main file, while the other two are the sub-files. Then users can click the “Run” button . Figure 3-2, 3-3, 3-4, 3-5 show the details.

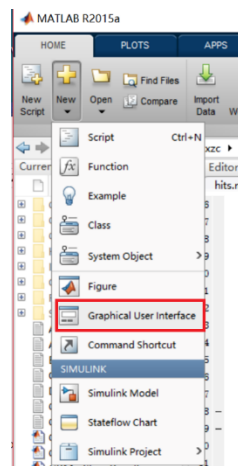


Figure 3-2 Open the *.fig file from graphic user interface---1

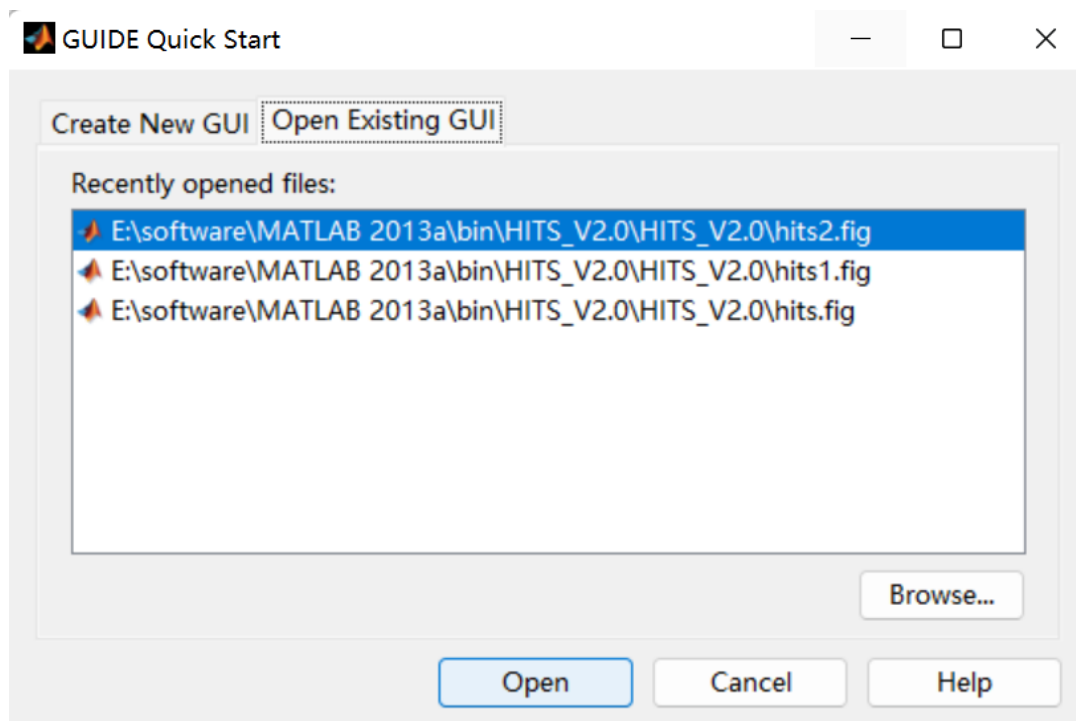


Figure 3-3 Open the *.fig file from graphic user interface---2

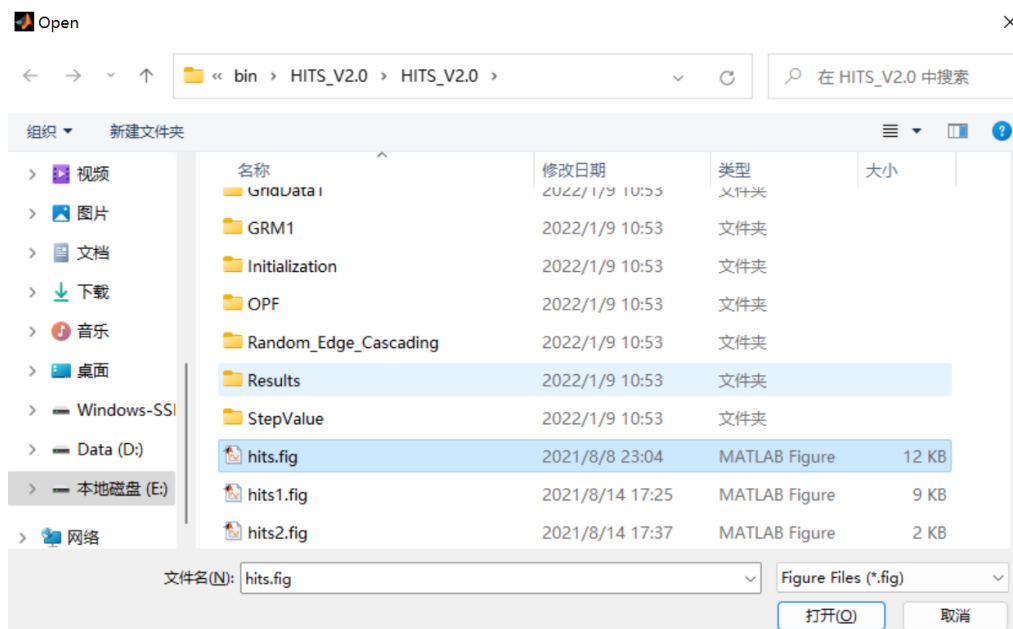


Figure 3-4 Open the *.fig file from graphic user interface---3

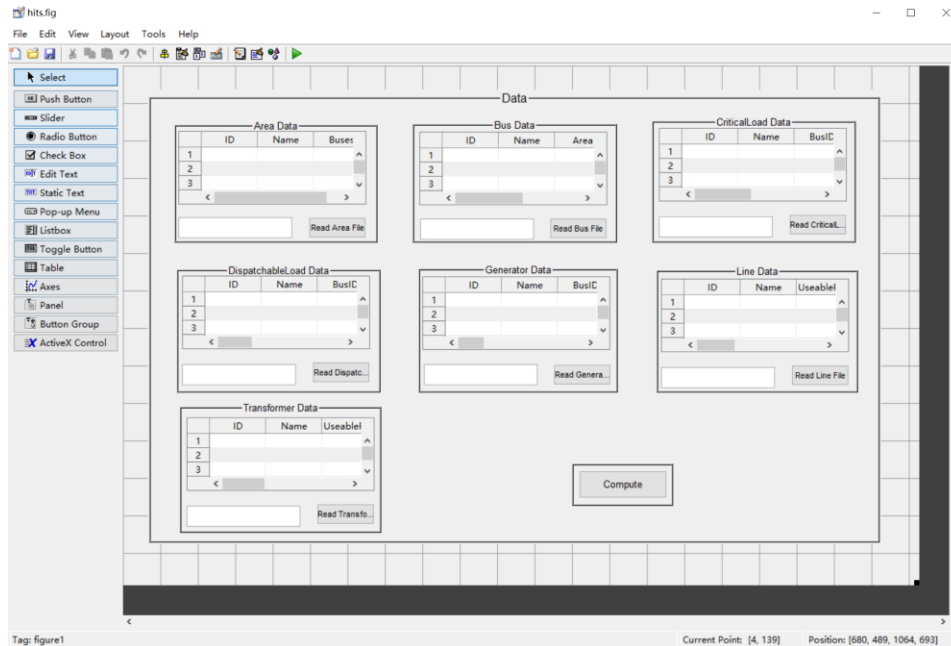


Figure 3-5 The editable main interface

Once users launch the program, the main interface of HITS version 2.0 is popped out, as shown in Figure 3-6.

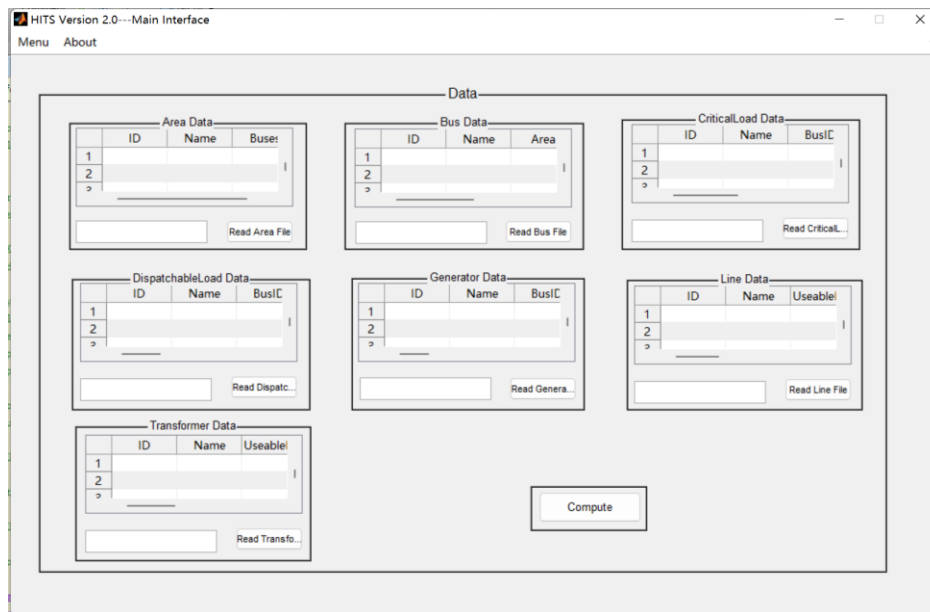


Figure 3-6 HITS Launching View

The data used for the program can be loaded in the main interface in the above figure. The data can be loaded from the folder named “...\GridData\RT24” or any folders users created, or users can just input or modify them on the tables of the main interface directly. The loaded data interface is shown in Figure 3-7 and the loaded MATPOWER system data in Figure 3-8.

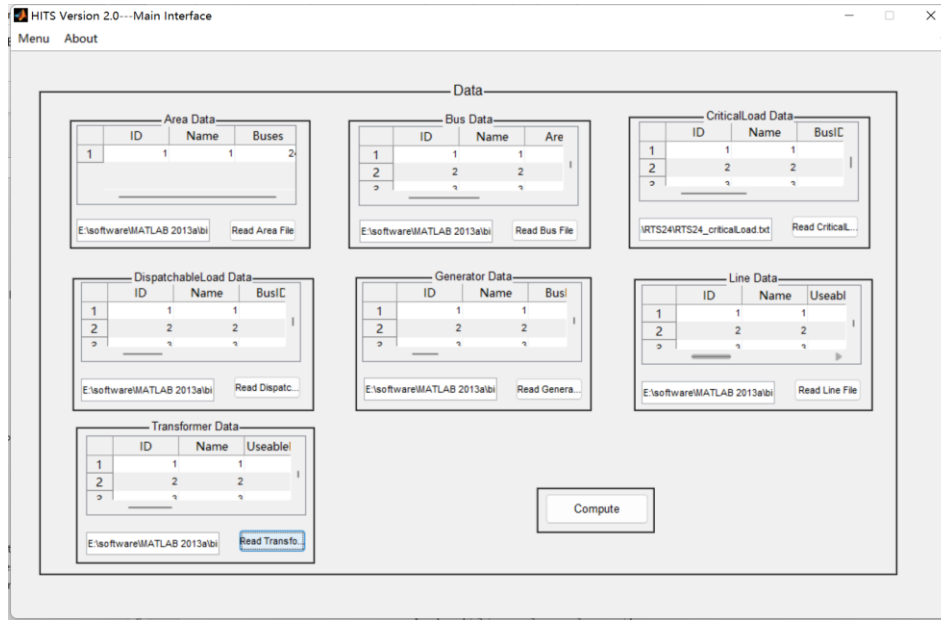


Figure 3-7 Data loaded on the main interface

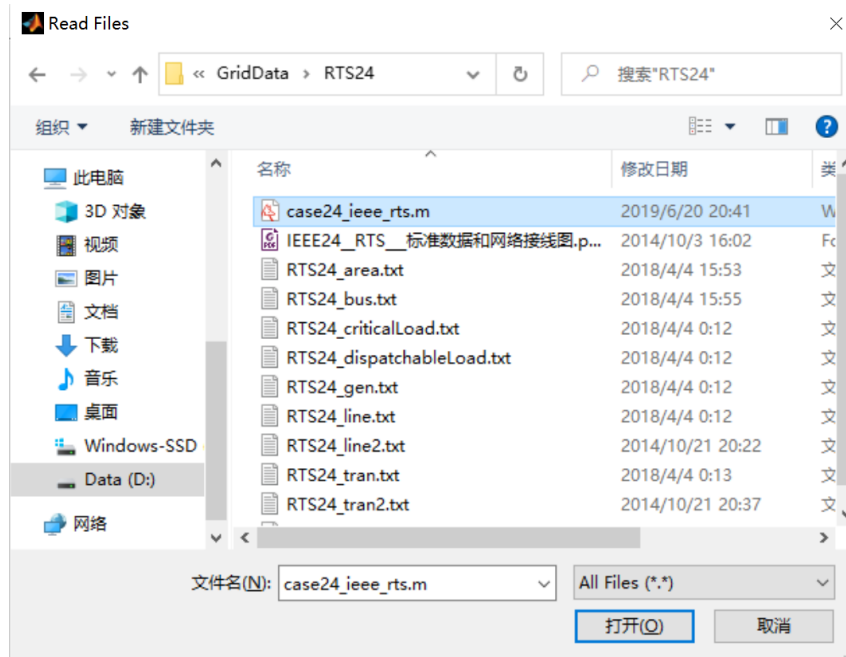


Figure 3-8 MATPOWER system data loaded on the main interface

Users can also change the parameters, create new data files, and even modify the program codes by themselves.

Once the data are loaded in the main interface, then by clicking the “Compute” button, the data in the main interface tables will be restored in the folder named “GridData1”. Also, there is a window pops up to load a related system file (MATPOWER format) if user wants to apply the influence graph function for vulnerable lines of system. Then the whole restoration plan is established.

The key results (including the Generator Start Time, Generator Start Order, Total Time of Each Step, Voltage and voltage’s phase angle of the BUS, voltage profiles of the last step, influence graph matrix, line weight rank) are shown in the results interface, shown in Figure 3-9, and the whole main restoration plan can also be obtained from the Command Window. But if user do not load the MATPOWER system file, the key results show in Figure 3-10, without influence graph matrix and line weight rank. There is also a popup window giving users reminder.

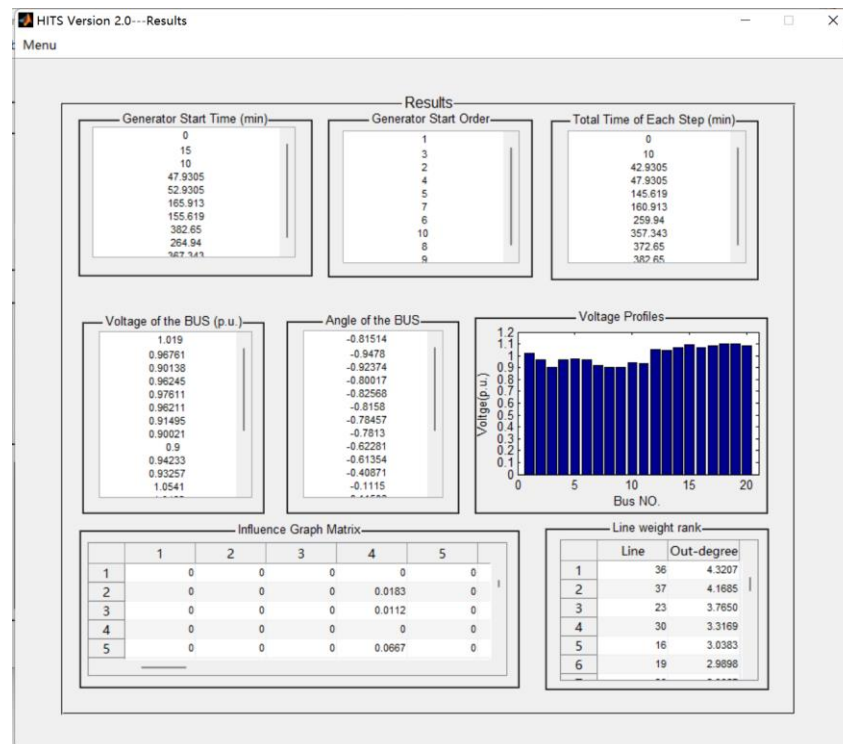


Figure 3-9 Results interface with loaded MATPOWER system file

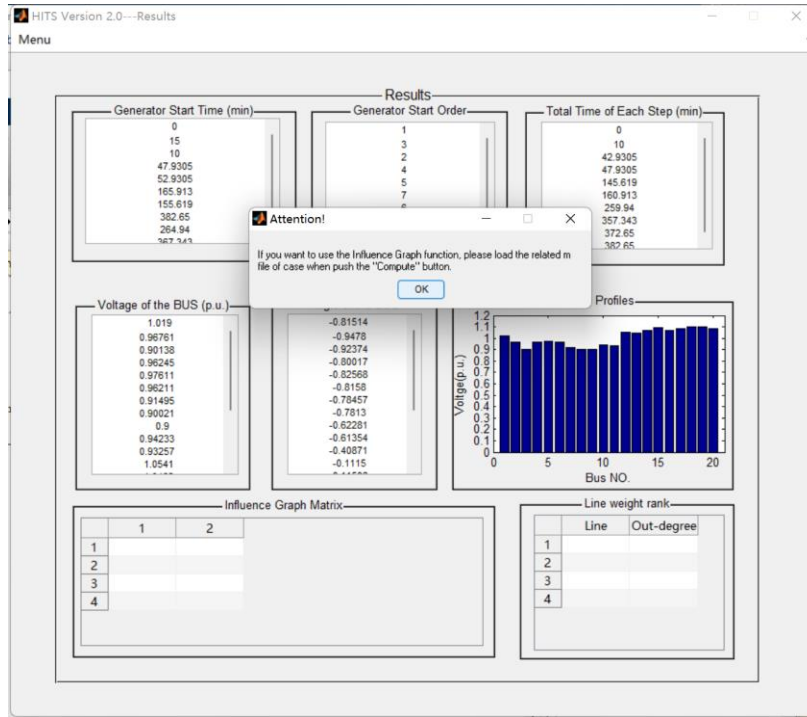


Figure 3-10 Results interface without loaded MATPOWER system file

3.4 Save for the data

The data displayed in the Results interface can be restored by clicking ‘Save’ in the main interface, as shown in Figure 3-11.

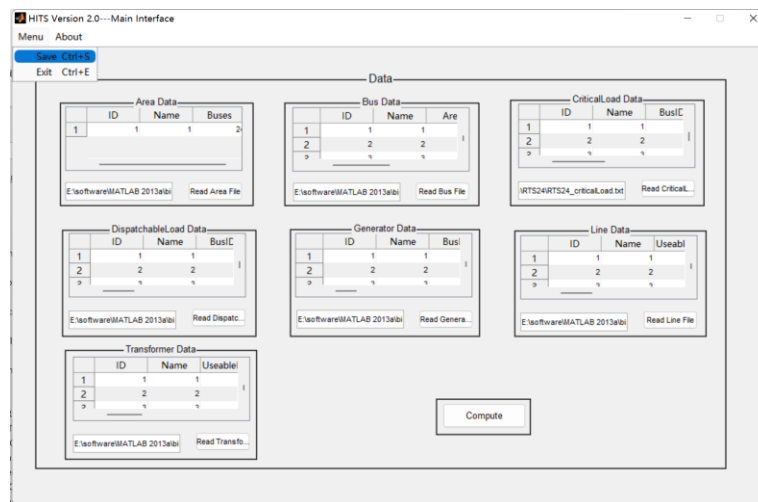


Figure 3-11 Save for data

3.5 About HITS_V2.0

By clicking the ‘about’ in the main interface, shown in Figure 3-12

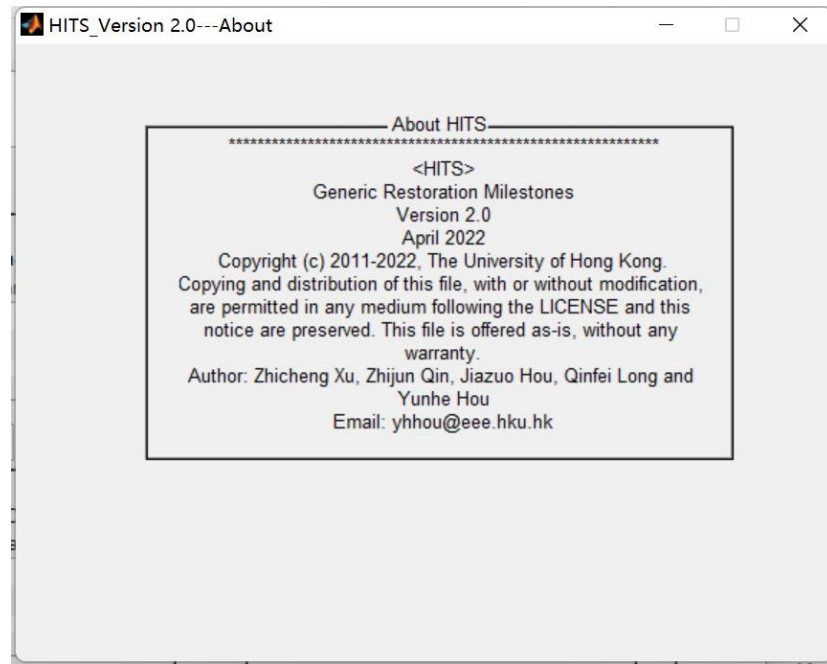


Figure 3-12 The ‘About’ interface of the GUI

The following chapters will show how the HITS works, how the data defines.

4 CREATING CUSTOM DATA

4.1 Data Location

As with the Sample Data, Custom Data is also placed under the path: “...\GRMs\GridData\RT24” as shown in Figure 4-1.

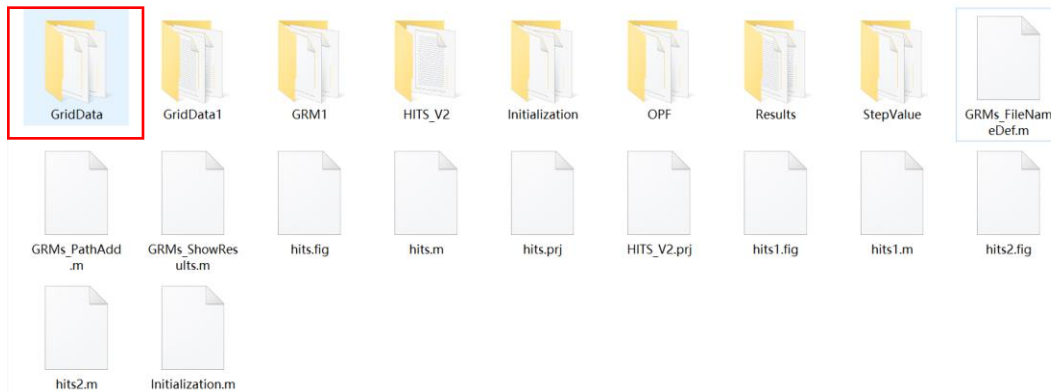


Figure 4-1 Data Location View

Users can create a new folder for storing new data files, as shown in Figure 4-2.

MyData_1	2018/6/22 10:57	文件夹	
RTS24	2018/6/21 12:51	文件夹	
GRMs_GetData_Area.m	2018/4/17 11:28	M 文件	1 KB
GRMs_GetData_Bus.m	2018/4/17 10:53	M 文件	1 KB
GRMs_GetData_CriticalLoad.m	2018/4/17 10:43	M 文件	1 KB
GRMs_GetData_DispatchableLoad.m	2018/4/17 10:54	M 文件	1 KB
GRMs_GetData_Generator.m	2018/4/17 10:54	M 文件	1 KB
GRMs_GetData_Line.m	2018/4/17 10:54	M 文件	1 KB
GRMs_GetData_Transformer.m	2018/4/17 10:54	M 文件	1 KB
GRMs_Preprocessing.m	2014/11/5 20:42	M 文件	3 KB

Figure 4-2 Create New Folder

4.2 Introduction of Data Format

In order to successfully establish the restoration process, users need to create the *Area*, *Bus*, *Critical Load*, *Dispatchable Load*, *Generator*, *Transformer*, *Line* and MATPOWER system files. Each data file has a corresponding data format, just as follows.

- **Area Data File Format**

The area data are as follows.

ID,	Name,	Buses
1,	Area01,	24

ID (int): Area ID;

Name (string): Area Name;

Buses (int): Total Number of Buses in Current Area.

- **Bus Data File Format**

The bus data are as follows.

ID,	Name,	Area,	V,	Delta
1,	bus001,	1,	1,	0
2,	bus002,	1,	1,	0
3,	bus003,	1,	1,	0
4,	bus004,	1,	1,	0
5,	bus005,	1,	1,	0
6,	bus006,	1,	1,	0
7,	bus007,	1,	1,	0
8,	bus008,	1,	1,	0

ID (int): Bus ID;

Name (string): Bus Name;

Area (int): Which Area the Current Bus Belongs;

V(float): Initial Voltage Amplitude;

Delta(float): Initial Voltage Angle.

- **Critical Load Data File Format**

The Critical Load data are as follows.

ID,	Name,	BusID,	P,	Q,	Priority
1,	1,	19,	0.15,	0.08,	1
2,	2,	14,	0.53,	0.19,	1
3,	3,	9,	0.40,	0.1,	1

ID (int): Critical Load ID;

Name (string): Critical Load Name;

BusID (int): Which Bus the Current Critical Load Connected;

P(float): Active Power of Current Critical Load;

Q(float): Reactive Power of Current Critical Load;

Priority(int): Priority of Current Critical Load.

- **Dispatchable Load Data File Format**

The dispatchable Load data follows.

ID,	Name,	BusID,	P,	Pmin,	Pmax,	Q,	Qmin,	Qmax,	Blocks
1,	1,	17,	1.53,	0,	1.53,	0.68,	0,	0.68,	2
2,	2,	11,	1.3725,	0,	1.3725,	0.64,	0,	0.64,	3
3,	3,	4,	1.0,	0,	1.0,	0.3,	0,	0.3,	3
4,	4,	10,	1.5,	0,	1.5,	0.5,	0,	0.5,	3

ID (int): Dispatchable Load ID;

Name (string : Dispatchable Load Name;

BusID (int): Which Bus the Current Dispatchable Load Connected;

P(float): Active Power of Current Dispatchable Load;

Pmin(float): Minim Active Power;

Pmax(float): Maximum Active Power;

Q(float): Reactive Power of Current Dispatchable Load;

Qmin(float): Minim Reactive Power;

Qmax(float): Maximum Reactive Power;

Blocks(int): The Number of Current Dispatchable Load can be divided;

- **Generator Data File Format**

The generator data are as follows.

ID,	Name,	BusID,	Capacity,	BlackStartFlag,	RampingRate,	StartRequirement,	TimeToCrank(min),	Pmin,	Pmax,	MinOutputRate,	Priority
1,	1,	22,	3.85,	1,	3.84,	0,	0,	0,	999,	0.001,	1
2,	2,	21,	3.00,	0,	1.20,	0.066,	30,	0,	999,	0.1,	1
3,	3,	18,	6.60,	0,	3.46,	0.132,	30,	0,	999,	0.3,	1
4,	4,	16,	6.00,	0,	1.57,	0.12,	100,	0,	999,	0.5,	1
5,	5,	15,	2.52,	0,	1.50,	0.30,	120,	0,	999,	0.5,	1
6,	6,	13,	1.35,	0,	0.30,	0.027,	160,	0,	999,	0.3,	1
7,	7,	23,	3.00,	0,	1.20,	0.06,	100,	0,	200,	0.7,	1
8,	8,	7,	3.00,	0,	1.00,	0.09,	120,	0,	210,	0.5,	1
9,	9,	2,	3.45,	0,	1.48,	0.12,	30,	0,	240,	0.3,	1
10,	10,	1,	3.02,	0,	1.20,	0,	0,	0,	999,	0.1,	1

ID (int) : Generator ID;

Name (string) : Generator Name;

BusID (int): Which Bus the Generator Connected;

Capacity(float): Capacity of Current Generator;

BlackStartFlag(int): Black Start Flag of Current Generator;

RampingRate(float): Ramping Rate of Current Generator;

StartRequirement(float): Active Power needed When the Current Generator Start Working;

TimeToCrank(float): The Time from Generator Start Working to Start Cranking;

Pmin(float): Minim Active Power;

Pmax(float): Maximum Active Power;

MinOutputRate(float): Minim Output Rate.

- **Line Data File Format**

The line data are as follows.

ID,	Name,	UseableFlag,	I,	J,	R,	X,	B,	Time
1,	1,	1,	1,	2,	0.003,	0.014,	0.061,	5
2,	2,	1,	1,	3,	0.055,	0.211,	0.057,	5
3,	3,	1,	1,	5,	0.022,	0.085,	0.023,	5
4,	4,	1,	2,	4,	0.033,	0.127,	0.034,	5
5,	5,	1,	2,	6,	0.05,	0.192,	0.052,	5
6,	6,	1,	3,	9,	0.031,	0.119,	0.032,	5
7,	7,	1,	4,	9,	0.027,	0.104,	0.028,	5
8,	8,	1,	5,	10,	0.023,	0.088,	0.024,	5
9,	9,	1,	6,	10,	0.014,	0.061,	2.459,	5
10,	10,	1,	7,	8,	0.016,	0.061,	0.017,	5
11,	11,	1,	8,	9,	0.043,	0.165,	0.045,	5
12,	12,	1,	8,	10,	0.043,	0.165,	0.045,	5
13,	13,	1,	11,	13,	0.006,	0.048,	0.100,	5
14,	14,	1,	11,	14,	0.005,	0.042,	0.088,	5
15,	15,	1,	12,	13,	0.006,	0.048,	0.100,	5
16,	16,	1,	12,	23,	0.012,	0.097,	0.203,	5
17,	17,	1,	13,	23,	0.011,	0.087,	0.182,	5
18,	18,	1,	14,	16,	0.005,	0.059,	0.082,	5
19,	19,	1,	15,	16,	0.002,	0.017,	0.136,	5
20,	20,	1,	15,	21,	0.003,	0.024,	0.206,	5

ID (int): Line ID;

Name (string): Line Name;

UseableFlag (int): Initialization state of Line;

I(int): Left Bus Number of Line;

J(int): Right Bus Number of Line;

R(float): Line Resistance;

X(float): Line Reactance;

B(float): Line Acceptance;

Time(float): Line charging time.

- **Transformer Data File Format**

The transformer data are as follows.

ID,	Name,	UseableFlag,	I,	J,	R,	X,	K,	Time
1,	1,	1,	3,	24,	0.002,	0.084,	1,	5
2,	2,	1,	9,	11,	0.002,	0.084,	1,	5
3,	3,	1,	10,	11,	0.002,	0.084,	1,	5
4,	4,	1,	9,	12,	0.002,	0.084,	1,	5
5,	5,	1,	10,	12,	0.002,	0.084,	1,	5

ID (int): Transformer ID;

Name (string): Transformer Name;

UseableFlag (int): Initialization state of Transformer;

I(int): Left Bus Number of Transformer;

J(int): Right Bus Number of Transformer;

R(float): Transformer Resistance;

X(float): Transformer Reactance;

K(float): Transformer Tap ratio;

Time(float): Transformer charging time.

- **MATPOWER System File Format**

The MATPOWER System data are defined in MATPOWER **data** folder.

```

function mpc = case24_ieee_rts
%CASE24_IEEE_RTS Power flow data for the IEEE RELIABILITY TEST SYSTEM.
% Please see CASEFORMAT for details on the case file format.
%
% This system data is from the IEEE RELIABILITY TEST SYSTEM, see
%
% IEEE Reliability Test System Task Force of the Applications of
% Probability Methods Subcommittee, "IEEE reliability test system,"
% IEEE Transactions on Power Apparatus and Systems, Vol. 98, No. 6,
% Nov./Dec. 1979, pp. 2047-2054.
%
% IEEE Reliability Test System Task Force of Applications of
% Probability Methods Subcommittee, "IEEE reliability test system-96,"
% IEEE Transactions on Power Systems, Vol. 14, No. 3, Aug. 1999,
% pp. 1010-1020.
%
% Cost data is from Web site run by Georgia Tech Power Systems Control
% and Automation Laboratory:
%
%      http://pscal.ece.gatech.edu/testsys/index.html
%
% MATPOWER case file data provided by Bruce Wollenberg.
%
% MATPOWER

```

5 INTRODUCTION OF KEY FUNCTIONS

- hits.m: The main file.
- hits.fig: The main GUI file.
- hits1.fig: The result GUI file.
- hits2. Fig: The 'about' information GUI file.
- GRMs_FileNameDef.m: Used for file name definitions, including the data file name and the file name of some of the sub-function files.

- GRMs_PathAdd.m: Used to automatically add a path without having to add subdirectories manually.
- GRM1_LoopInit.m: As the initialization file of the main loop before GRM1 calculation, the variables which are the key to the loop are initialized.
- GRM1_StepInit.m: After a successful start of a generator, ready to start the next unit to do the initialization. Once a generating unit start successfully, step increase in the value of 1.
- GRM1_UpdateDepth.m: Update the search-depth D. In the current version, if all the generator sets cannot start at the current search-depth, then the step value continue the same while the search-depth D is enlarged, and the starting search of the generator sets is continued. If search-depth D has been expanded to the entire grid, and this stage can not successfully start the generator, then that the entire black start failed.
- GRM1_WaitForStart.m: Find the unit to be started under the current search dimension, schedule the load and the critical load.
- GRM1_SortGenerators.m: Prioritize the crews to be started.
- GRM1_CrankInit.m: Initialization before attempting to start a generator.
- GRM1_EnergizePath.m: The uncharged lines can be charged according to the generators, loads, and energized buses found in the search range, and multiple lines can be charged at the same time.
- GRM1_OPF.m: The main function of OPF.
- GRM1_CrankJudgment.m: Determine whether the current generator can be successfully started, and update some of the loop variables.
- GRM1_CalculateTime.m: Calculate the returned generator power based on the OPF, calculate the time of each generator, and return the total time.
- Random_Edge_Cascading.m: Using cascading failure simulator to obtain the influence graph matrix and line weight rank.
- Relationship_Graph.m: On the base of simulated cascading failure database generated by Random_Edge_Cascading.m, this file can calculate the related influence graph matrix and detect the vulnerable line set including lines' outdegree and rank.

6 SOFTWARE FLOW CHART

