**MATLAB PSAT Simulator Documentation**

**initial\_failures.m**

Implements function: initial\_failures

Arguments:

nl: Number of lines in data model

nf: Number of FACTS in data model

np: Number of PMU in data model

nd: Number of decision support entities in data model

ml: Max concurrent line failures

mf: Max concurrent FACTS failures

mp: Max concurrent PMU failures

mc: Max concurrent communication failures

md: Max conccurrent decision support failures

mg: Max number of concurrent GPS signals affected by noise

noise: duration of noise in microseconds

Output:

Returns structure with fields:

line

facts

pmu

ds\_facts

pmu\_ds

ds

gps

sv

Dependencies:

None

**cascade.m**

Implements function: cascade

Arguments:

file\_name: Name of PSAT .m data file without the ‘.m’

initial\_failures: Structure created by function: initial\_failures

Output:

failure\_seq/file\_name.txt: Text representation of output

failuer\_seq/file\_name.mat: Stores failure\_seq structure created by cascade.m

Dependencies:

data/file\_name.m: PSAT .m data file for power grid

nn/file\_name.mat: Neural network file

**identify\_dep.m**

Implements function: identify\_dep

Arguments:

file\_name: Name of PSAT .m data file from cascade.m without the ‘.m’

method: one of “qis”, “cor”, or “rdc”

Output:

Returns matrix structure: dependency matrix

Dependencies:

failure\_seq/file\_name.mat: Output from cascade.m

data/file\_name\_comps.mat: Number of components in input system

**plot\_graph.m**

Implements function: plot\_graph

Arguments:

file\_name: Name of PSAT .m data file from cascade.m without the ‘.m’

d: Structure output from identify\_dep.m

legends: Boolean value to determine if graph has legends

pdf: Boolean value to determine if pdf output should be generated

Output:

Prints a graph to the screen

/graph\_file\_name.pdf: Optional pdf output

Dependencies:

data/file\_name\_comps.mat: Number of components in input system

**Working input examples:**

Addpath ‘your\_path\psat\’

init = initial\_failures(20, 3, 3, 1, 1, 1, 1, 0, 1)

cascade(“ieee14\_smart”, init)

dqis = identify\_dep(“ieee14\_smart”, “qis”)

dcor = identify\_dep(“ieee14\_smart”, “cor”)

drdc = identify\_dep(“ieee14\_smart”, “rdc”)

plot\_graph(“ieee14\_smart”, dqis, 1, 1)

plot\_graph(“ieee14\_smart”, dcor, 1, 1)

plot\_graph(“ieee14\_smart”, drdc, 1, 1)