

Purpose

This document is meant to explain PST additions and alterations created to accommodate AGC.

Area Definitions

To enable area calculations, each bus in the **bus** array must be assigned to an area in the **area_def** array. An example **area_def** array is shown below.

```
1  %% area_def data format
2  % should contain same number of rows as bus array (i.e. all bus areas defined)
3  % col 1 bus number
4  % col 2 area number
5  area_def = [ ...
6              1  1;
7              2  1;
8              3  1;
9              4  1;
10             10 1;
11             11 2;
12             12 2;
13             13 2;
14             14 2;
15             20 1;
16             101 1;
17             110 2;
18             120 2];
```

It should be noted that rows may not have to be in the same order as the bus array (untested). The **area_def** array is automatically placed into the global **g** structure.

```
1  >> g.area
2  ans =
3      area_def: [13x2 double]
4      n_area: 2
5      area: [1x2 struct]
```

Each area currently contains values that may be relevant to AGC calculations. The **calcAreaVals** function is used to calculate and store such values. An example of what is stored in the **g.area.area(x)** structure is shown below.

```
1  >> g.area.area(2)
2  ans =
3      number: 2
4      areaBuses: [6x1 double]
5      macBus: [2x1 double]
```

```
6      macBusNdx: [3 4]
7      loadBus: [4x1 double]
8      loadBusNdx: [8 9 12 13]
9      genBus: [2x1 double]
10     genBusNdx: [6 7]
11     totH: [1x4063 double]
12     aveF: [1x4063 double]
13     totGen: [1x4063 double]
14     totLoad: []
15     icA: [1x4063 double]
16     icS: []
17     exportLineNdx: [11 12]
18     importLineNdx: []
19     n_export: 2
20     n_import: []
```

It should be noted that `icS` represents a placeholder for a scheduled interchange value and the `totLoad` is a field for collected total load. The collection of actual running load values may prove more complicated as the bus array does not seem to be updated every step, only a reduced Y matrix used in the `nc_load` function called from `i_simu`.

Line Monitoring

Power flow on a line must be calculated each step as AGC requires actual area interchange for the ACE calculation. The previously existing `line_pq` function performed this task, but was not fully implemented into the simulation to allow calculation during execution. This minor oversight has been resolved, however the `lmon_con` array is still used to define monitored lines.

```
1  %% Line Monitoring
2  % Each value corresponds to an array index in the line array.
3  % Complex current and power flow on the line will be calculated and logged during simulation
4
5  %lmon_con = [5, 6, 13]; % lines between bus 3 and 101, and line between 13 and 120
6
7  lmon_con = [3,10]; % lines to loads
```

Line monitoring data is collected in the `g.lmon` field of the global variable.

```
1  >> g.lmon
2  ans =
3      lmon_con: [3 10]
4      n_lmon: 2
5      busFromTo: [2x2 double]
6      line: [1x2 struct]
```

Each `g.lmon.line` entry contains the following fields:

```

1  >> g.lmon.line(2)
2  ans =
3      busArrayNdx: 10
4      FromBus: 13
5      ToBus: 14
6      iFrom: [1x4063 double]
7      iTo: [1x4063 double]
8      sFrom: [1x4063 double]
9      sTo: [1x4063 double]

```

Weighted Average Frequency

An average weighted frequency is calculated for the total system and for each area if there are areas detected. The calculation involves a sum of system inertias that may change with generator trips. The current algorithm does not account for tripped generators, but was designed to incorporate this feature in the future.

In a system with N generators, M areas, and N_M generators in area M , the `calcAveF` function performs the following calculations for each area M :

$$H_{tot_M} = \sum_i^{N_M} MV A_{base_i} H_i$$

$$F_{ave_M} = \left(\sum_i^{N_M} Mach_{speed_i} MV A_{base_i} H_i \right) / H_{tot_M}$$

Then system total values are calculated as

$$H_{tot} = \sum_i^M H_{tot_M}$$

$$F_{ave} = \left(\sum_i^M F_{ave_M} \right) / M$$

If $M = 0$, `calcAveF` performs

$$H_{tot} = \sum_i^N MV A_{base_i} H_i$$

$$F_{ave} = \left(\sum_i^N (Mach_{speed_i} MV A_{base_i} H_i) \right) / H_{tot}$$

Automatic Generation Control

Under development...

- Kundur 4 machine system packaged with PST
- Constant Z load model
- All machines, exciters, and govcs identical
- PSS on gen 1 and 3
- SVC on bus 101
- Event: +50 MW (0.5 PU) step of load on bus 4 at $t=1$

Initial simulation results (No AGC)

