Frequency measurement with gridlabd

Matthieu Stephant, 25/08/17

This short document presents quickly some indications to measure frequency with gridlabd (using delta-mode). It summarizes the necessary elemements to this measurement, according to what I have understood, and all the useful information that I have found. However, the delta-mode seems to be still under development today, so check regularly the gridlabd documentation and the forum.

The frequency measurement requires to set the file into delta-mode and to add some parameters. The deltamode enables measurement with a scale below 1 second. To enable the f4requency to move (and simulate a mirco-grid), a generator, for example diesel generator, has to be linked to the swing node.

Here is a list of the element that should be added in your file to run it into delta-mode:

- **Settings of the glm:** (the values are example of reasonable values but can be changed)
  - **remove** `#set minimum_timestep=60;` it is incompatible with the deltamode
  - **add** `#define rotor_convergence=0.0001` (example of reasonable value)
  - **add** `#set deltamode_timestep=100000000 //100 ms`
  - **add** `#set deltamode_maximumtime=60000000000 //1 minute`
  - **add** `#set deltamode_iteration_limit=10 //Iteration limit`

- **Modules:** the order matters for deltamode, the following modules have to be added after calling module `tape;`
  - **add module powerflow:**

```
module powerflow {
 enable_subsecond_models true;  //Enables deltamode for the
powerflow module
```

```
 deltamode_timestep 10.0 ms;    //Good timestep value for
electromechanical diesels
 solver_method NR;           //NR MUST be used for
deltamode, otherwise it won't work
 all_powerflow_delta true;   //Saves having to put "flags
DELTAMODE;" on every powerflow object
 default_maximum_voltage_error 1e-3; //by default 10e-9,
but this value is used only to validate tests
 nominal_frequency 50; // in Hz, by default 60 Hz
}
```

- **add module generators:**

```
module generators {
 enable_subsecond_models true;  //Enables deltamode for
generator module
 deltamode_timestep 10.0 ms; //Desired timestep for the
generator module models
}
```

- **Power flow:**
  - each power flow object (node, meter, load, transformers,…) has to be set into deltamode by **adding** `flags DELTAMODE;` (if you have a doubt if you should add this for an objet, put it as it seems to not affect the simulation even if this flag is useless for the considered object)
  - Frequency can be measured at the nodes. However, if you want to measure other properties such as power, energy, voltage and current, you should use meters instead of nodes. Meters inherit from the node class but have more measurement properties.
  - ***Node/Meter:***
    - **add** `flags DELTAMODE;`
    - **add** `frequency_measure_type PLL; //see documentation to know more about this type of measurem`**`ent`**
    - **add** a recorder to measure frequency
    - **example of meter**

```
object meter:633 {
 name 633;
 phases "ABCN";
 flags DELTAMODE;
```

```
 frequency_measure_type PLL;
 voltage_A 2401.7771;
 voltage_B -1200.8886-2080.000j;
 voltage_C -1200.8886+2080.000j;
 nominal_voltage 2401.7771;

 object recorder {                                    flags
DELTAMODE;   //Must explicitly set the recorder to do
deltamode
 file meters_outputs/meter633.csv;
 interval 1;
 limit 100000000000;
 property measured_angle_A, measured_angle_B,
measured_angle_C, measured_frequency_A,
measured_frequency_B, measured_frequency_C,
measured_frequency, measured_power.real,
measured_power.imag, measured_power_A.real,
measured_power_A.imag, measured_power_B.real,
measured_power_B.imag, measured_power_C.real,
measured_power_C.imag;
 };
}
```

- ***SWING node/meter:***

  To enable the frequency to move, it is necessary to link a generator to the swing node. For example:

  - ```
    object diesel_dg {
    parent 650; // parent node
    name Gen_Bus;
    Rated_V 4160;   // adapt to your voltage
    Rated_VA 10.0 MVA;  //default value
    flags DELTAMODE;
    Gen_type DYN_SYNCHRONOUS;
    Exciter_type SEXS;                  //Give it voltage
    control
    Governor_type DEGOV1;               //Make it try to
    maintain frequency
    rotor_speed_convergence ${rotor_convergence};
    }
    ```
    ```

- ***Other power flow objects, recorders and players:***
  - **add** `flags DELTAMODE;`

- ***Inverters and solar panels:***
  Here is an example of a standard setup for a solar panel coupled to an inverter, with constant production (no climate object)

```
object inverter {
 name solar_inv_2;
 phases AS;    // adapt to your configuration
 parent trip_meter2;
 inverter_efficiency .95;
 rated_power 6000;
 generator_mode CONSTANT_PF;
 generator_status ONLINE;
 inverter_type PWM;
 flags DELTAMODE;
 dynamic_model_mode PID;
 inverter_convergence_criterion 0.001;
 power_factor 1.0;
 object solar {
  name solar_2;
  phases AS;
  generator_status ONLINE;
  generator_mode SUPPLY_DRIVEN;
  panel_type SINGLE_CRYSTAL_SILICON;
  area 250 ft^2;
  tilt_angle 47.0;
  efficiency 0.135;
  };
 }
```

- **<u>How to run the file in deltamode:</u>**

  Even if you have set up all the previous parameters, it is possible that the simulation runs without entering into deltamode: it will run without error, but the frequency measured at the node will be still equal to the nominal frequency, and values will be measured each second. To make the file enter into deltamode, something has to change with an interval inferior to one second: consumption of a load, status of a switch… So you have to use a player file. For

example, you can use a switch somewhere in your configuration and set its status in a player file:

- ```
  object switch:671692 {
      phases "ABCN";
   name 671-692;
      from meter:671;
      to meter:692;
      object player {
      flags DELTAMODE;
      file inputs/switch_status.csv;
      property status;
      };
  }
  ```
- player file `switch_status.csv` in the folder `inputs/`

2010-01-01 12:00:00.00 PST	CLOSED
2010-01-01 12:00:00.01 PST	CLOSED

(for a simulation starting at 2010-01-01 12:00:00)

With this player, you don’t actually modify the status of the switch but you make the simulation runs into deltamode from 12:00:00.01.

Note: According to the gridlabd team, the deltamode is still under development, so the information here might change. Moreover, at the present time (August 2017), the deltamode was very few explained on the gridlab-d documentation, but some additional information may be added by the gridlabd team.
  ```