

Power Injection

October 23, 2023

1 Power Injection

This example is to elaborate how to get the bus injected power in ANDES, and how to inspect them during the simulation, as an answer for [Discussion #471](#).

1.1 Variables of Interest

```
[ ]: import numpy as np
import pandas as pd

import andes

import datetime
```

```
[ ]: print(f"Last run: {datetime.datetime.now()}\nANDES version: {andes.
↪ __version__}")
```

```
Last run: 2023-10-23 10:21:24.544021
ANDES version: 1.8.10.post18+g4d1e7315
```

```
[ ]: andes.config_logger(stream_level=20)
```

Here we use IEEE 14-bus case as an example.

Load the case, run power flow, and initialize the TDS.

```
[ ]: sa = andes.load(andes.get_case("ieee14/ieee14_full.xlsx"),
                    no_output=True, setup=True)

sa.PFlow.run()

_ = sa.TDS.init()
```

Working directory:

```
"/Users/jinningwang/Documents/work/andes/icebar/injectedpower"
> Loaded config from file "/Users/jinningwang/.andes/andes.rc"
> Loaded generated Python code in "/Users/jinningwang/.andes/pycode".
Generated code for <PQ> is stale.
Numerical code generation (rapid incremental mode) started..

Generating code for 1 models on 8 processes.
```

```

Saved generated pycode to "/Users/jinningwang/.andes/pycode"
> Reloaded generated Python code of module "pycode".
Generated numerical code for 1 models in 0.3124 seconds.
Parsing input file
"/Users/jinningwang/Documents/work/andes/andes/cases/ieee14/ieee14_full.xlsx"...
Input file parsed in 0.1119 seconds.
System internal structure set up in 0.0234 seconds.
-> System connectivity check results:
    No islanded bus detected.
    System is interconnected.
    Each island has a slack bus correctly defined and enabled.

```

```

-> Power flow calculation
    Numba: Off
    Sparse solver: KLU
    Solution method: NR method
Power flow initialized in 0.0040 seconds.
0: |F(x)| = 0.5605182134
1: |F(x)| = 0.006202200332
2: |F(x)| = 5.819382825e-06
3: |F(x)| = 6.957087684e-12
Converged in 4 iterations in 0.0035 seconds.
Initialization for dynamics completed in 0.0254 seconds.
Initialization was successful.

```

Initialize variables to store bus injected power from Line, SynGen, and Load, respectively.

```
[ ]: p_inj_line = np.zeros(sa.Bus.n)
     p_inj_syg = np.zeros(sa.Bus.n)
     p_inj_load = np.zeros(sa.Bus.n)
```

In this case we only have SynGen in dynamic generators. If RenGen occurs, similar method can be applied to include it.

```
[ ]: syg_idx = []
     for mdl in sa.SynGen.models.values():
         syg_idx += mdl.idx.v

     syg_bus = sa.SynGen.get(src='bus', attr='v', idx=syg_idx)

     load_idx = []
     for mdl in sa.StaticLoad.models.values():
         load_idx += mdl.idx.v

     load_bus = sa.StaticLoad.get(src='bus', attr='v', idx=load_idx)
```

Here, a for loop is used to iterate through all buses in the system.

Note that this is only for demonstration purpose, and can be inefficient for large cases.

In model Line, attribute a1 and a2 are ExtAlgeb objects that will be summed to target Bus variable a for active power calculation. The attribute e of ExtAlgeb is the injected value.

Similarly, for a model connected to a bus such as SynGen or Load, there is usually an ExtAlgeb named a that is connected to the bus variable a.

```
[ ]: for bus_idx in sa.Bus.idx.v:
    # get the location of bus device
    bus_loc = sa.Bus.idx2uid(bus_idx)

    # find the Line idx given "from bus"
    # NOTE: method `find_idx` returns incomplete idx if multiple matches occur
    inj_line_idx = []
    for line_idx in sa.Line.idx.v:
        if sa.Line.get(src='bus1', attr='v', idx=line_idx) == bus_idx:
            inj_line_idx.append(line_idx)
    line_loc = sa.Line.idx2uid(inj_line_idx)
    line_e = sa.Line.get(src='a1', attr='e', idx=inj_line_idx)
    p_inj_line[bus_loc] += line_e.sum()

    # similar, find the Line idx given "to bus"
    inj_line_idx = []
    for line_idx in sa.Line.idx.v:
        if sa.Line.get(src='bus2', attr='v', idx=line_idx) == bus_idx:
            inj_line_idx.append(line_idx)
    line_loc = sa.Line.idx2uid(inj_line_idx)
    line_e = sa.Line.get(src='a2', attr='e', idx=inj_line_idx)
    p_inj_line[bus_loc] += line_e.sum()

    # get Dynamic Generator idx given "bus"
    inj_syg_idx = []
    for syg in syg_idx:
        if sa.SynGen.get(src='bus', attr='v', idx=syg) == bus_idx:
            inj_syg_idx.append(syg)
    syg_e = sa.SynGen.get(src='a', attr='e', idx=inj_syg_idx)
    p_inj_syg[bus_loc] += syg_e.sum()

    # NOTE: If DynLoad occurs, similar method can be used
    inj_load_idx = []
    for load in load_idx:
        if sa.StaticLoad.get(src='bus', attr='v', idx=load) == bus_idx:
            inj_load_idx.append(load)
    load_e = sa.StaticLoad.get(src='a', attr='e', idx=inj_load_idx)
    p_inj_load[bus_loc] += load_e.sum()
```

In the last, the total bus injected power can be summed up from the three components.

Note that the positive direction is defined “out from the bus”.

```
[ ]: p_inj_bus = p_inj_line + p_inj_syg + p_inj_load

p_inj = pd.DataFrame({'Bus': sa.Bus.idx.v,
                    'Line': p_inj_line, 'SynGen': p_inj_syg,
                    'Load': p_inj_load, 'Total': p_inj_bus})

p_inj.round(4)
```

```
[ ]:
   Bus  Line  SynGen  Load  Total
0    1    0.8143 -0.8143  0.000  -0.0
1    2    0.1830 -0.4000  0.217   0.0
2    3   -0.1000 -0.4000  0.500  -0.0
3    4   -0.4780  0.0000  0.478  -0.0
4    5   -0.0760  0.0000  0.076   0.0
5    6    0.1500 -0.3000  0.150  -0.0
6    7    0.0000  0.0000  0.000   0.0
7    8    0.3500 -0.3500  0.000  -0.0
8    9   -0.2950  0.0000  0.295  -0.0
9   10   -0.0900  0.0000  0.090  -0.0
10   11   -0.0350  0.0000  0.035   0.0
11   12   -0.0610  0.0000  0.061  -0.0
12   13   -0.1350  0.0000  0.135  -0.0
13   14   -0.2000  0.0000  0.200   0.0
```

1.2 Inspect the bus injected power during simulation

```
[ ]: sa = andes.load(andes.get_case("ieee14/ieee14_full.xlsx"),
                    pert='pert.py',
                    no_output=True, setup=False)

sa.add('Toggle', dict(model='Line', dev='Line_9', t=1))
sa.setup()
```

Working directory:

"/Users/jinningwang/Documents/work/andes/icebar/injectedpower"

> Loaded config from file "/Users/jinningwang/.andes/andes.rc"

> Reloaded generated Python code of module "pycode".

Generated code for <PQ> is stale.

Numerical code generation (rapid incremental mode) started..

Generating code for 1 models on 8 processes.

Saved generated pycode to "/Users/jinningwang/.andes/pycode"

> Reloaded generated Python code of module "pycode".

Generated numerical code for 1 models in 0.1526 seconds.

Parsing input file

"/Users/jinningwang/Documents/work/andes/andes/cases/ieee14/ieee14_full.xlsx"...

Input file parsed in 0.1085 seconds.

System internal structure set up in 0.0221 seconds.

```
[ ]: True
```

```
[ ]: sa.PFlow.run()  
  
sa.TDS.config.no_tqdm = True # disable progress bar  
sa.TDS.config.tf = 5 # set simulation time to 5 seconds  
  
_ = sa.TDS.init()
```

```
-> System connectivity check results:  
    No islanded bus detected.  
    System is interconnected.  
    Each island has a slack bus correctly defined and enabled.
```

```
-> Power flow calculation  
    Numba: Off  
    Sparse solver: KLU  
    Solution method: NR method  
Power flow initialized in 0.0056 seconds.  
    No islanded bus detected.  
    System is interconnected.  
    Each island has a slack bus correctly defined and enabled.
```

```
-> Power flow calculation  
    Numba: Off  
    Sparse solver: KLU  
    Solution method: NR method  
Power flow initialized in 0.0056 seconds.  
0: |F(x)| = 0.5605182134  
1: |F(x)| = 0.006202200332  
2: |F(x)| = 5.819382825e-06  
3: |F(x)| = 6.957087684e-12  
Converged in 4 iterations in 0.0033 seconds.  
Perturbation file "pert.py" loaded.  
Initialization for dynamics completed in 0.0465 seconds.  
Initialization was successful.
```

```
[ ]: sa.TDS.run()
```

```
-> Time Domain Simulation Summary:  
Sparse Solver: KLU  
Simulation time: 0.0-5 s.  
Fixed step size: h=33.33 ms. Shrink if not converged.  
  
t=0.03333, pinj_line=0.81427  
t=0.06667, pinj_line=0.81427  
t=0.10000, pinj_line=0.81427  
t=0.13333, pinj_line=0.81427
```

t=0.16667, pinj_line=0.81427
t=0.20000, pinj_line=0.81427
t=0.23333, pinj_line=0.81427
t=0.26667, pinj_line=0.81427
t=0.30000, pinj_line=0.81427
t=0.33333, pinj_line=0.81427
t=0.36667, pinj_line=0.81427
t=0.40000, pinj_line=0.81427
t=0.43333, pinj_line=0.81427
t=0.46667, pinj_line=0.81427
t=0.50000, pinj_line=0.81427
t=0.53333, pinj_line=0.81427
t=0.56667, pinj_line=0.81427
t=0.60000, pinj_line=0.81427
t=0.63333, pinj_line=0.81427
t=0.66667, pinj_line=0.81427
t=0.70000, pinj_line=0.81427
t=0.73333, pinj_line=0.81427
t=0.76667, pinj_line=0.81427
t=0.80000, pinj_line=0.81427
t=0.83333, pinj_line=0.81427
t=0.86667, pinj_line=0.81427
t=0.90000, pinj_line=0.81427
t=0.93333, pinj_line=0.81427
t=0.96667, pinj_line=0.81427
t=0.99990, pinj_line=0.81427
t=1.00000, pinj_line=0.81427
<Toggle Toggle_1>: Line.Line_9 status changed to 0 at t=1.0 sec.
t=1.00010, pinj_line=0.81427
t=1.03343, pinj_line=0.81396
t=1.06677, pinj_line=0.81355
t=1.10010, pinj_line=0.81345
t=1.13343, pinj_line=0.81342
t=1.16677, pinj_line=0.81341
t=1.20010, pinj_line=0.81341
t=1.23343, pinj_line=0.81342
t=1.26677, pinj_line=0.81343
t=1.30010, pinj_line=0.81344
t=1.33343, pinj_line=0.81346
t=1.36677, pinj_line=0.81347
t=1.40010, pinj_line=0.81348
t=1.43343, pinj_line=0.81349
t=1.46677, pinj_line=0.81350
t=1.50010, pinj_line=0.81350
t=1.53343, pinj_line=0.81350
t=1.56677, pinj_line=0.81349
t=1.60010, pinj_line=0.81349
t=1.63343, pinj_line=0.81348

t=1.66677, pinj_line=0.81347
t=1.70010, pinj_line=0.81346
t=1.73343, pinj_line=0.81345
t=1.76677, pinj_line=0.81344
t=1.80010, pinj_line=0.81343
t=1.83343, pinj_line=0.81342
t=1.86677, pinj_line=0.81341
t=1.90010, pinj_line=0.81341
t=1.93343, pinj_line=0.81340
t=1.96677, pinj_line=0.81339
t=2.00010, pinj_line=0.81339
t=2.03343, pinj_line=0.81339
t=2.06677, pinj_line=0.81338
t=2.10010, pinj_line=0.81338
t=2.13343, pinj_line=0.81337
t=2.16677, pinj_line=0.81337
t=2.20010, pinj_line=0.81337
t=2.23343, pinj_line=0.81336
t=2.26677, pinj_line=0.81336
t=2.30010, pinj_line=0.81336
t=2.33343, pinj_line=0.81335
t=2.36677, pinj_line=0.81335
t=2.40010, pinj_line=0.81334
t=2.43343, pinj_line=0.81334
t=2.46677, pinj_line=0.81334
t=2.50010, pinj_line=0.81334
t=2.53343, pinj_line=0.81333
t=2.56677, pinj_line=0.81333
t=2.60010, pinj_line=0.81333
t=2.63343, pinj_line=0.81333
t=2.66677, pinj_line=0.81333
t=2.70010, pinj_line=0.81333
t=2.73343, pinj_line=0.81333
t=2.76677, pinj_line=0.81332
t=2.80010, pinj_line=0.81332
t=2.83343, pinj_line=0.81332
t=2.86677, pinj_line=0.81332
t=2.90010, pinj_line=0.81332
t=2.93343, pinj_line=0.81332
t=2.96677, pinj_line=0.81331
t=3.00010, pinj_line=0.81331
t=3.03343, pinj_line=0.81331
t=3.06677, pinj_line=0.81330
t=3.10010, pinj_line=0.81330
t=3.13343, pinj_line=0.81330
t=3.16677, pinj_line=0.81330
t=3.20010, pinj_line=0.81329
t=3.23343, pinj_line=0.81329

t=3.26677, pinj_line=0.81329
t=3.30010, pinj_line=0.81329
t=3.33343, pinj_line=0.81328
t=3.36677, pinj_line=0.81328
t=3.40010, pinj_line=0.81328
t=3.43343, pinj_line=0.81328
t=3.46677, pinj_line=0.81328
t=3.50010, pinj_line=0.81328
t=3.53343, pinj_line=0.81328
t=3.56677, pinj_line=0.81328
t=3.60010, pinj_line=0.81328
t=3.63343, pinj_line=0.81328
t=3.66677, pinj_line=0.81328
t=3.70010, pinj_line=0.81328
t=3.73343, pinj_line=0.81328
t=3.76677, pinj_line=0.81328
t=3.80010, pinj_line=0.81328
t=3.83343, pinj_line=0.81328
t=3.86677, pinj_line=0.81328
t=3.90010, pinj_line=0.81328
t=3.93343, pinj_line=0.81328
t=3.96677, pinj_line=0.81328
t=4.00010, pinj_line=0.81327
t=4.03343, pinj_line=0.81327
t=4.06677, pinj_line=0.81327
t=4.10010, pinj_line=0.81327
t=4.13343, pinj_line=0.81327
t=4.16677, pinj_line=0.81327
t=4.20010, pinj_line=0.81327
t=4.23343, pinj_line=0.81327

Simulation to t=5.00 sec completed in 0.2530 seconds.

t=4.26677, pinj_line=0.81327
t=4.30010, pinj_line=0.81327
t=4.33343, pinj_line=0.81327
t=4.36677, pinj_line=0.81327
t=4.40010, pinj_line=0.81327
t=4.43343, pinj_line=0.81327
t=4.46677, pinj_line=0.81327
t=4.50010, pinj_line=0.81327
t=4.53343, pinj_line=0.81327
t=4.56677, pinj_line=0.81327
t=4.60010, pinj_line=0.81327
t=4.63343, pinj_line=0.81327
t=4.66677, pinj_line=0.81327
t=4.70010, pinj_line=0.81327
t=4.73343, pinj_line=0.81327
t=4.76677, pinj_line=0.81327


```
t=4.80010, pinj_line=0.81327  
t=4.83343, pinj_line=0.81327  
t=4.86677, pinj_line=0.81327  
t=4.90010, pinj_line=0.81327  
t=4.93343, pinj_line=0.81327  
t=4.96677, pinj_line=0.81327  
t=5.00000, pinj_line=0.81327
```

```
[ ]: True
```