

### Introduction

ANDES is a symbolic-numeric tool for power system analysis developed by CURENT. (Source code: [GitHub](#))  
ANDES is run with `andes command option(s)`. Please refer to **All Commands** section for available commands. Next, look up the corresponding section for options.

### All Commands

<code>run</code>	Run a simulation routine.
<code>plot</code>	Plot figures from output files.
<code>prepare</code>	Run the symbolic-to-numeric preparation.
<code>misc</code>	Miscellaneous functions.
<code>-v {10, 20, 30, 40, 50}</code>	Logging verbosity. 10-DEBUG, 20-INFO, 30-WARNING, 40-ERROR, 50-CRITICAL.

### andes run options

<code>filename</code>	Input file name in relative or absolute path.
<code>-r {tds, eig}</code>	Routine to run. Options are <code>tds</code> (time-domain simulation) and <code>eig</code> (eigenvalue analysis).
<code>-p</code>	Search path for cases.
<code>INPUT_PATH</code>	
<code>-o OUTPUT_PATH</code>	Path for output files.
<code>--convert [CONVERT]</code>	Convert input files to FORMAT. The default is <code>xlsx</code> .
<code>--profile</code>	Enable Python cProfiler.
<code>-n</code>	Disable all outputs.

### andes plot options

<code>filename</code>	One output file name to plot.
<code>x</code>	X-axis variable index. The default is 0 (time).
<code>y</code>	Y-axis variable index (or indices). Supports Python indexing.
<code>-s</code>	Save figure to a PNG file.
<code>-c</code>	Convert to a <code>csv</code> file.
<code>--xmin</code>	X-axis minimum tick XMIN.
<code>--xmax</code>	X-axis maximum tick XMAX.
<code>--ymin</code>	Y-axis minimum tick YMIN.
<code>--ymax</code>	Y-axis maximum tick YMAX.
<code>--dpi</code>	Set figure DPI.
<code>-g</code>	Grid on.
<code>-d</code>	Disable LaTeX.

### andes plot options (cont)

`-n` Disable figure showing.

### andes misc options

<code>--save-config</code>	Save configurations to a file.
<code>--edit-config</code>	Edit an existing configuration file.
<code>--clean</code>	Clean all output files.

### Basic Command-Line Interface (CLI) Examples

Run power flow for `case14.xlsx` in the current directory:

```
andes run case14.xlsx
```

Run time-domain simulation for `case14.xlsx`:

```
andes run case14.xlsx -r tds
```

Run TDS for `case14.xlsx` to simulate 40 seconds:

```
andes run case14.xlsx -r tds --tf 40
```

Run eigenvalue analysis for `case14.xlsx`:

```
andes run case14.xlsx -r eig
```

Plot TDS variables with indices 2 and 4 against time:

```
andes plot case14 _ou t.npy 0 2 4
```

Plot TDS variables with indices 2, 4, ..., 20 against time:

```
andes plot case14 _ou t.npy 0 2:21:2
```

### Advanced CLI Examples

Save ANDES configuration file to home directory:

```
andes misc --save
```

Edit ANDES configuration file in home directory:

```
andes misc --edit
```

Batch run power flow for all files with the `.xlsx` extension:

```
andes run *.xlsx
```

Run TDS with verbose outputs for debugging:

```
andes -v 10 run case14.xlsx -r tds
```

Run TDS with profiling and print the results:

```
andes run case14.xlsx -r tds --profile
```

