

```
In [2]: %run plot_simulate_evoked.py
Warning: no DISPLAY environment variable.
--No graphics will be displayed.
Joblib will run 2 trial(s) in parallel by distributing trials over 2
jobs.
Warning: no DISPLAY environment variable.
--No graphics will be displayed.
Warning: no DISPLAY environment variable.
--No graphics will be displayed.
```

```
-----
_RemoteTraceback                                Traceback (most recent call
last)
```

```
_RemoteTraceback:
''''''
```

```
Traceback (most recent call last):
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
externals/loky/process_executor.py", line 463, in _process_worker
```

```
    r = call_item()
    ~~~~~
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
externals/loky/process_executor.py", line 291, in __call__
```

```
    return self.fn(*self.args, **self.kwargs)
    ~~~~~
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py", line 598, in __call__
```

```
    return [func(*args, **kwargs)
    ~~~~~
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/
network_builder.py", line 43, in _simulate_single_trial
```

```
    neuron_net = NetworkBuilder(net, trial_idx=trial_idx)
    ~~~~~
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/
network_builder.py", line 305, in __init__
```

```
    self._build()
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/
network_builder.py", line 316, in _build
```

```
    load_custom_mechanisms()
```

```
File "/opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/
network_builder.py", line 160, in load_custom_mechanisms
```

```
    raise FileNotFoundError(f'No .so or .dll file found in {mod_dir}')
FileNotFoundError: No .so or .dll file found in /opt/anaconda3/envs/
```

```
hnn/lib/python3.12/site-packages/hnn_core/mod
''''''
```

The above exception was the direct cause of the following exception:

```
FileNotFoundError                                Traceback (most recent call
last)
```

```
File ~/Downloads/plot_simulate_evoked.py:96
```

```

    93 from hnn_core import JoblibBackend
    95 with JoblibBackend(n_jobs=2):
----> 96     dpls = simulate_dipole(net, tstop=170., n_trials=2)
    98
#####
#####
    99 # Rather than reading smoothing and scaling parameters from
file, we recommend
    100 # explicit use of the :meth:`~hnn_core.dipole.Dipole.smooth`
and
    101 # :meth:`~hnn_core.dipole.Dipole.scale` methods instead. Note
that both methods
    102 # operate in-place, i.e., the objects are modified.
    103 window_len, scaling_factor = 30, 3000

```

File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/dipole.py:100, in simulate_dipole(net, tstop, dt, n_trials, record_vsec, record_isec, postproc)

```

    95 if postproc:
    96     warnings.warn('The postproc-argument is deprecated and
will be removed'
    97                   ' in a future release of hnn-core. Please
define '
    98                   'smoothing and scaling explicitly using
Dipole methods.',
    99                   DeprecationWarning)
--> 100 dpls = BACKEND.simulate(net, tstop, dt, n_trials, postproc)
    102 return dpls

```

File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/hnn_core/parallel_backends.py:558, in JoblibBackend.simulate(self, net, tstop, dt, n_trials, postproc)

```

    555 print(f"Joblib will run {n_trials} trial(s) in parallel by "
    556       f"distributing trials over {self.n_jobs} jobs.")
    557 parallel, myfunc = self._parallel_func(_simulate_single_trial)
--> 558 sim_data = parallel(myfunc(net, tstop, dt, trial_idx) for
    559                    trial_idx in range(n_trials))
    561 dpls = _gather_trial_data(sim_data, net=net,
n_trials=n_trials,
    562                          postproc=postproc)
    564 return dpls

```

File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/parallel.py:2007, in Parallel.__call__(self, iterable)

```

    2001 # The first item from the output is blank, but it makes the
interpreter
    2002 # progress until it enters the Try/Except block of the
generator and
    2003 # reaches the first `yield` statement. This starts the
asynchronous

```

```
2004 # dispatch of the tasks to the workers.
2005 next(output)
-> 2007 return output if self.return_generator else list(output)
```

```
File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py:1650, in Parallel._get_outputs(self, iterator,
pre_dispatch)
```

```
1647     yield
1649     with self._backend.retrieval_context():
-> 1650         yield from self._retrieve()
1652 except GeneratorExit:
1653     # The generator has been garbage collected before being
fully
1654     # consumed. This aborts the remaining tasks if possible
and warn
1655     # the user if necessary.
1656     self._exception = True
```

```
File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py:1754, in Parallel._retrieve(self)
```

```
1747 while self._wait_retrieval():
1748
1749     # If the callback thread of a worker has signaled that its
task
1750     # triggered an exception, or if the retrieval loop has
raised an
1751     # exception (e.g. `GeneratorExit`), exit the loop and
surface the
1752     # worker traceback.
1753     if self._aborting:
-> 1754         self._raise_error_fast()
1755         break
1757     # If the next job is not ready for retrieval yet, we just
wait for
1758     # async callbacks to progress.
```

```
File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py:1789, in Parallel._raise_error_fast(self)
```

```
1785 # If this error job exists, immediately raise the error by
1786 # calling get_result. This job might not exist if abort has
been
1787 # called directly or if the generator is gc'ed.
1788 if error_job is not None:
-> 1789     error_job.get_result(self.timeout)
```

```
File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py:745, in BatchCompletionCallback.get_result(self, timeout)
```

```
739 backend = self.parallel._backend
741 if backend.supports_retrieve_callback:
742     # We assume that the result has already been retrieved by
```

```
the
743     # callback thread, and is stored internally. It's just
waiting to
744     # be returned.
--> 745     return self._return_or_raise()
747     # For other backends, the main thread needs to run the
retrieval step.
748     try:
```

```
File /opt/anaconda3/envs/hnn/lib/python3.12/site-packages/joblib/
parallel.py:763, in BatchCompletionCallback._return_or_raise(self)
```

```
761     try:
762         if self.status == TASK_ERROR:
--> 763             raise self._result
764         return self._result
765     finally:
```

```
FileNotFoundError: No .so or .dll file found in /opt/anaconda3/envs/
hnn/lib/python3.12/site-packages/hnn_core/mod
```