

# notebook-experiment-shocks

February 9, 2021

## 1 Experiment Analysis: Controller parameter stability search

Perform shocks of ETH price to test controller parameter stability, without stochastic processes.

- See `experiments/system_model_v3/experiment_shocks.py`

## 2 Setup and Dependencies

```
[1]: # Set project root folder, to enable importing project files from subdirectories
from pathlib import Path
import os

path = Path().resolve()
root_path = str(path).split('notebooks')[0]
os.chdir(root_path)

# Force reload of project modules, sometimes necessary for Jupyter kernel
%load_ext autoreload
%autoreload 2

# Display framework versions for easy debugging
%pip show cadCAD
%pip show radcad
```

Name: cadCAD

Version: 0.4.23

Summary: cadCAD: a differential games based simulation software package for research, validation, and Computer Aided Design of economic systems

Home-page: <https://github.com/cadCAD-org/cadCAD>

Author: Joshua E. Jodesty

Author-email: [joshua@block.science](mailto:joshua@block.science)

License: LICENSE.txt

Location: /home/bscholtz/workspace/reflexer/venv/lib/python3.8/site-packages

Requires: fn, funcy, pandas, pathos

Required-by:

Note: you may need to restart the kernel to use updated packages.

Name: radcad

Version: 0.5.4

Summary: A cadCAD implementation, for dynamical systems modelling & simulation  
Home-page: None  
Author: Benjamin Scholtz  
Author-email: ben@bitsofether.com  
License: None  
Location: /home/bscholtz/workspace/reflexer/venv/lib/python3.8/site-packages  
Requires: boto3, pandas, ray, pathos, tables  
Required-by:  
Note: you may need to restart the kernel to use updated packages.

```
[2]: # Import all shared dependencies and setup
from shared import *

import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
# import plotly.io as pio
# pio.renderers.default = "png"
from pprint import pprint
```

```
[3]: # Update dataframe display settings
pd.set_option('display.max_columns', 100)
pd.set_option('display.max_rows', 50)
```

### 3 Load Results

Using the experiment logs, select the experiment of interest from the specific HDF5 store file (these datasets are very large, and won't be committed to repo):

```
[4]: # experiment_results = 'experiments/system_model_v3/experiment_controller_sweep/
    ↳ experiment_results.hdf5'
experiment_results = 'experiments/system_model_v3/experiment_shocks/
    ↳ experiment_results.hdf5'
```

```
[5]: experiment_results_keys = []
with pd.HDFStore(experiment_results) as store:
    experiment_results_keys = list(filter(lambda x: "results" in x, store.
    ↳ keys()))
    exceptions_keys = list(filter(lambda x: "exceptions" in x, store.keys()))
```

```
[6]: # A list of all experiment result keys
experiment_results_keys
```

```
[6]: ['/processed_results_2021-02-08T19:20:25.053328',
      '/processed_results_2021-02-08T19:45:15.538863',
      '/processed_results_2021-02-08T20:05:35.827550',
      '/processed_results_2021-02-08T22:54:43.005952',
```

```

'/processed_results_2021-02-08T23:20:54.834451',
'/processed_results_2021-02-09T09:43:01.154807',
'/results_2021-02-08T19:11:14.829535',
'/results_2021-02-08T19:20:25.053328',
'/results_2021-02-08T19:45:15.538863',
'/results_2021-02-08T20:05:35.827550',
'/results_2021-02-08T22:54:43.005952',
'/results_2021-02-08T23:20:54.834451',
'/results_2021-02-09T09:43:01.154807',
'/results_2021-02-09T11:44:24.150805',
'/results_2021-02-09T11:48:54.447990',
'/results_2021-02-09T11:56:55.060726',
'/results_2021-02-09T11:59:26.345803']

```

```

[7]: # A list of all experiment result exception keys
exceptions_keys

```

```

[7]: ['/exceptions_2021-02-08T19:11:14.829535',
      '/exceptions_2021-02-08T19:20:25.053328',
      '/exceptions_2021-02-08T19:45:15.538863',
      '/exceptions_2021-02-08T20:05:35.827550',
      '/exceptions_2021-02-08T22:54:43.005952',
      '/exceptions_2021-02-08T23:20:54.834451',
      '/exceptions_2021-02-09T09:43:01.154807',
      '/exceptions_2021-02-09T11:44:24.150805',
      '/exceptions_2021-02-09T11:48:54.447990',
      '/exceptions_2021-02-09T11:56:55.060726',
      '/exceptions_2021-02-09T11:59:26.345803']

```

```

[8]: # Copy a results_ key from the above keys to select the experiment
experiment_results_key = 'results_2021-02-09T11:59:26.345803' # Or select last_
    ↳ result: experiment_results_keys[-1]
experiment_timestamp = experiment_results_key.strip('results_')
exceptions_key = 'exceptions_' + experiment_timestamp
experiment_timestamp

```

```

[8]: '2021-02-09T11:59:26.345803'

```

```

[9]: df_raw = pd.read_hdf(experiment_results, experiment_results_key)
df_raw.tail()

```

```

[9]:
518415 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...      \
518416 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...      \
518417 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...      \
518418 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...      \
518419 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...      \

```

	sim_metrics	timedelta	cumulative_time	timestamp	blockheight	\
518415	{}	3600	5184000	2017-03-02	0	
518416	{}	3600	5184000	2017-03-02	0	
518417	{}	3600	5184000	2017-03-02	0	
518418	{}	3600	5184000	2017-03-02	0	
518419	{}	3600	5184000	2017-03-02	0	

	eth_price	liquidity_demand	liquidity_demand_mean	\
518415	300.0	0.0	0.0	
518416	300.0	0.0	0.0	
518417	300.0	0.0	0.0	
518418	300.0	0.0	0.0	
518419	300.0	0.0	0.0	

				cdps	eth_collateral	\
518415	open	time	locked	drawn	...	127527.405834
518416	open	time	locked	drawn	...	127527.405834
518417	open	time	locked	drawn	...	127527.405834
518418	open	time	locked	drawn	...	127527.405834
518419	open	time	locked	drawn	...	127527.405834

	eth_locked	eth_freed	eth_bitten	principal_debt	rai_drawn	\
518415	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
518416	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
518417	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
518418	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
518419	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	

	rai_wiped	rai_bitten	accrued_interest	interest_dripped	\
518415	4.285738e+06	0.0	6900.562442	0	
518416	4.285738e+06	0.0	6900.562442	0	
518417	4.285738e+06	0.0	6900.562442	0	
518418	4.285738e+06	0.0	6900.562442	0	
518419	4.285738e+06	0.0	6900.562442	0	

	interest_wiped	interest_bitten	w_1	w_2	w_3	system_revenue	\
518415	0	0	0.0	0.0	0.0	0.0	
518416	0	0	0.0	0.0	0.0	0.0	
518417	0	0	0.0	0.0	0.0	0.0	
518418	0	0	0.0	0.0	0.0	0.0	
518419	0	0	0.0	0.0	0.0	0.0	

	stability_fee	market_price	market_price_twap	target_price	\
518415	1.585490e-10	4.547664	4.547664	3.14	
518416	1.585490e-10	4.547664	4.547664	3.14	
518417	1.585490e-10	4.547664	4.547664	3.14	

518418	1.585490e-10	4.547664	4.547664	3.14
518419	1.585490e-10	4.547664	4.547664	3.14

	target_rate	eth_return	eth_gross_return	expected_market_price	\
518415	0.0	0.0	1.0	4.680219	
518416	0.0	0.0	1.0	4.680219	
518417	0.0	0.0	1.0	4.680219	
518418	0.0	0.0	1.0	4.680219	
518419	0.0	0.0	1.0	4.680219	

	expected_debt_price	error_star	error_star_integral	\
518415	3.14	-1.407664	-1.231163e+06	
518416	3.14	-1.407664	-1.231163e+06	
518417	3.14	-1.407664	-1.231163e+06	
518418	3.14	-1.407664	-1.231163e+06	
518419	3.14	-1.407664	-1.231163e+06	

	secondary_market_slippage	RAI_balance	ETH_balance	UNI_supply	\
518415	0	8.402860e+06	127377.935231	10000000.0	
518416	0	8.402860e+06	127377.935231	10000000.0	
518417	0	8.402860e+06	127377.935231	10000000.0	
518418	0	8.402860e+06	127377.935231	10000000.0	
518419	0	8.402860e+06	127377.935231	10000000.0	

	uniswap_oracle	simulation	subset	\
518415	<models.system_model_v3.model.parts.uniswap_or...	0	3	
518416	<models.system_model_v3.model.parts.uniswap_or...	0	3	
518417	<models.system_model_v3.model.parts.uniswap_or...	0	3	
518418	<models.system_model_v3.model.parts.uniswap_or...	0	3	
518419	<models.system_model_v3.model.parts.uniswap_or...	0	3	

	run	substep	timestep	events
518415	5	14	1440	NaN
518416	5	15	1440	NaN
518417	5	16	1440	NaN
518418	5	17	1440	NaN
518419	5	18	1440	NaN

Get experiment exceptions, tracebacks, and simulation metadata for further analysis:

```
[10]: exceptions_df = pd.read_hdf(experiment_results, exceptions_key)
exceptions_df.head()
```

```
[10]: exception traceback simulation run subset timesteps \
0      None      None      0      0      0      1440
1      None      None      0      0      1      1440
2      None      None      0      0      2      1440
```

```

3      None      None      0      0      3      1440
4      None      None      0      1      0      1440

```

```

                                parameters \
0  {"0":{"debug":false,"raise_on_assert":true,"fr...
1  {"0":{"debug":false,"raise_on_assert":true,"fr...
2  {"0":{"debug":false,"raise_on_assert":true,"fr...
3  {"0":{"debug":false,"raise_on_assert":true,"fr...
4  {"0":{"debug":false,"raise_on_assert":true,"fr...

                                initial_state
0  {'cdp_metrics': {}, 'optimal_values': {}, 'sim...
1  {'cdp_metrics': {}, 'optimal_values': {}, 'sim...
2  {'cdp_metrics': {}, 'optimal_values': {}, 'sim...
3  {'cdp_metrics': {}, 'optimal_values': {}, 'sim...
4  {'cdp_metrics': {}, 'optimal_values': {}, 'sim...

```

```
[11]: # Print the first 5 exceptions - indicating failed simulations
pprint(list(exceptions_df['exception'][:5]))
```

```
[None, None, None, None, None]
```

## 4 Post Process Results

```
[12]: from experiments.system_model_v3.post_process import post_process_results
from experiments.system_model_v3.experiment_shocks import params,
      ↪SIMULATION_TIMESTEPS
```

```

* Number of timesteps: 1440 / 60.0 days
* Timestep duration: 0.004 seconds
* Control parameters: ['kp', 'ki', 'controller_enabled', 'rescale_target_price']
* Approx. number of values per parameter: 1
* Number of parameter combinations: 4
* Expected experiment duration: 0.384 minutes / 0.0064 hours

```

Remove substeps, add `set_params` to dataframe, and add post-processing columns:

```
[34]: df = post_process_results(df_raw, params, set_params=['ki', 'kp',
      ↪'liquidation_ratio', 'controller_enabled', 'rescale_target_price'])
df
```

```
[34]:      index      cdp_metrics \
0         0      {}
1        18 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
2        36 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
```

3	54	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
4	72	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
...	...	...
28815	518347	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
28816	518365	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
28817	518383	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
28818	518401	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
28819	518419	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...

	optimal_values	sim_metrics	timedelta	cumulative_time	\
0	{}	{}	0	0	
1	{}	{}	3600	3600	
2	{}	{}	3600	7200	
3	{}	{}	3600	10800	
4	{}	{}	3600	14400	
...	...	...	...	...	
28815	{}	{}	3600	5169600	
28816	{}	{}	3600	5173200	
28817	{}	{}	3600	5176800	
28818	{}	{}	3600	5180400	
28819	{}	{}	3600	5184000	

	timestamp	blockheight	eth_price	liquidity_demand	\
0	2017-01-01 00:00:00	0	294.069151	1.0	
1	2017-01-01 01:00:00	0	300.000000	0.0	
2	2017-01-01 02:00:00	0	300.000000	-0.0	
3	2017-01-01 03:00:00	0	300.000000	0.0	
4	2017-01-01 04:00:00	0	300.000000	-0.0	
...	...	...	...	...	
28815	2017-03-01 20:00:00	0	300.000000	-0.0	
28816	2017-03-01 21:00:00	0	300.000000	-0.0	
28817	2017-03-01 22:00:00	0	300.000000	0.0	
28818	2017-03-01 23:00:00	0	300.000000	0.0	
28819	2017-03-02 00:00:00	0	300.000000	0.0	

	liquidity_demand_mean	\
0	1.0000	
1	0.5000	
2	0.2500	
3	0.1250	
4	0.0625	
...	...	
28815	0.0000	
28816	0.0000	
28817	0.0000	
28818	0.0000	
28819	0.0000	

				cdps	eth_collateral	\
0				None	154827.528922	
1				None	154827.528922	
2				None	105925.122117	
3				None	105925.122117	
4				None	105925.122117	
...				...	...	
28815				None	127527.405834	
28816				None	127527.405834	
28817				None	127527.405834	
28818	open	time	locked	drawn	...	127527.405834
28819	open	time	locked	drawn	...	127527.405834

	eth_locked	eth_freed	eth_bitten	principal_debt	rai_drawn	\
0	154827.528922	0.000000	0.0	1.000000e+07	1.000000e+07	
1	154827.528922	0.000000	0.0	1.450000e+07	1.450000e+07	
2	154827.528922	48902.406805	0.0	1.012023e+07	1.450000e+07	
3	154827.528922	48902.406805	0.0	1.012023e+07	1.450000e+07	
4	154827.528922	48902.406805	0.0	1.012023e+07	1.450000e+07	
...	...	...	...	...	...	
28815	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
28816	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
28817	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
28818	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	
28819	154827.528922	27300.123088	0.0	8.402860e+06	1.268860e+07	

	rai_wiped	rai_bitten	accrued_interest	interest_dripped	\
0	0.000000e+00	0.0	0.000000	0	
1	0.000000e+00	0.0	5.707764	0	
2	4.379765e+06	0.0	11.484158	0	
3	4.379765e+06	0.0	17.260556	0	
4	4.379765e+06	0.0	23.036956	0	
...	...	...	...	...	
28815	4.285738e+06	0.0	6881.362098	0	
28816	4.285738e+06	0.0	6886.162180	0	
28817	4.285738e+06	0.0	6890.962265	0	
28818	4.285738e+06	0.0	6895.762352	0	
28819	4.285738e+06	0.0	6900.562442	0	

	interest_wiped	interest_bitten	w_1	w_2	w_3	\
0	0	0	0.000000e+00	0.0	0.0	
1	0	0	8.276258e+00	0.0	0.0	
2	0	0	4.723892e-06	0.0	0.0	
3	0	0	2.696510e-12	0.0	0.0	
4	0	0	0.000000e+00	0.0	0.0	
...	...	...	...	...	...	



28815	0	0	0.000000e+00	0.0	0.0
28816	0	0	8.881784e-16	0.0	0.0
28817	0	0	0.000000e+00	0.0	0.0
28818	0	0	8.881784e-16	0.0	0.0
28819	0	0	0.000000e+00	0.0	0.0

	system_revenue	stability_fee	market_price	market_price_twap	\
0	0.0	1.585490e-10	3.140000	0.000000	
1	0.0	1.585490e-10	3.140000	0.000000	
2	0.0	1.585490e-10	1.525000	0.000000	
3	0.0	1.585490e-10	3.133425	0.000000	
4	0.0	1.585490e-10	3.133425	0.000000	
...	...	...	...	...	
28815	0.0	1.585490e-10	4.547664	4.547664	
28816	0.0	1.585490e-10	4.547664	4.547664	
28817	0.0	1.585490e-10	4.547664	4.547664	
28818	0.0	1.585490e-10	4.547664	4.547664	
28819	0.0	1.585490e-10	4.547664	4.547664	

	target_price	target_rate	eth_return	eth_gross_return	\
0	3.140000	0.0	0.000000	0.0	
1	2.165517	0.0	0.020168	1.0	
2	2.165517	0.0	0.000000	1.0	
3	2.165517	0.0	0.000000	1.0	
4	2.165517	0.0	0.000000	1.0	
...	...	...	...	...	
28815	3.140000	0.0	0.000000	1.0	
28816	3.140000	0.0	0.000000	1.0	
28817	3.140000	0.0	0.000000	1.0	
28818	3.140000	0.0	0.000000	1.0	
28819	3.140000	0.0	0.000000	1.0	

	expected_market_price	expected_debt_price	error_star	\
0	3.140000	3.14	0.000000	
1	3.231683	3.14	3.140000	
2	1.569093	3.14	3.140000	
3	3.224303	3.14	3.140000	
4	3.224444	3.14	3.140000	
...	...	...	...	
28815	4.680218	3.14	-1.407664	
28816	4.680218	3.14	-1.407664	
28817	4.680218	3.14	-1.407664	
28818	4.680218	3.14	-1.407664	
28819	4.680219	3.14	-1.407664	

	error_star_integral	secondary_market_slippage	RAI_balance	\
0	0.000000e+00	0	1.000000e+07	

1	5.652000e+03	0	1.450000e+07
2	1.693200e+04	0	1.012023e+07
3	2.816600e+04	0	1.012023e+07
4	3.935400e+04	0	1.012023e+07
...	...	...	...
28815	-1.231115e+06	0	8.402860e+06
28816	-1.231127e+06	0	8.402860e+06
28817	-1.231139e+06	0	8.402860e+06
28818	-1.231151e+06	0	8.402860e+06
28819	-1.231163e+06	0	8.402860e+06

	ETH_balance	UNI_supply \
0	106777.606153	10000000.0
1	73708.353400	10000000.0
2	105703.331685	10000000.0
3	105703.331685	10000000.0
4	105703.331685	10000000.0
...	...	...
28815	127377.935231	10000000.0
28816	127377.935231	10000000.0
28817	127377.935231	10000000.0
28818	127377.935231	10000000.0
28819	127377.935231	10000000.0

	uniswap_oracle	simulation	subset \
0	None	0	0
1	None	0	0
2	None	0	0
3	None	0	0
4	None	0	0
...	...	...	...
28815	None	0	3
28816	None	0	3
28817	None	0	3
28818	<models.system_model_v3.model.parts.uniswap_or...	0	3
28819	<models.system_model_v3.model.parts.uniswap_or...	0	3

	run	substep	timestep	events	eth_collateral_value \
0	1	0	0	NaN	4.553000e+07
1	1	18	1	NaN	4.644826e+07
2	1	18	2	NaN	3.177754e+07
3	1	18	3	NaN	3.177754e+07
4	1	18	4	NaN	3.177754e+07
...	...	...	...	...	...
28815	5	18	1436	NaN	3.825822e+07
28816	5	18	1437	NaN	3.825822e+07
28817	5	18	1438	NaN	3.825822e+07

28818	5	18	1439	NaN	3.825822e+07
28819	5	18	1440	NaN	3.825822e+07

	collateralization_ratio	ki	kp	liquidation_ratio \
0	1.450000	-1.000000e-09	0.000001	1.45
1	1.479244	-1.000000e-09	0.000001	1.45
2	1.450000	-1.000000e-09	0.000001	1.45
3	1.450000	-1.000000e-09	0.000001	1.45
4	1.450000	-1.000000e-09	0.000001	1.45
...	...	...	...	...
28815	1.450000	-1.000000e-09	0.000001	1.45
28816	1.450000	-1.000000e-09	0.000001	1.45
28817	1.450000	-1.000000e-09	0.000001	1.45
28818	1.450000	-1.000000e-09	0.000001	1.45
28819	1.450000	-1.000000e-09	0.000001	1.45

	controller_enabled	rescale_target_price	target_price_scaled
0	True	True	4.553
1	True	True	3.140
2	True	True	3.140
3	True	True	3.140
4	True	True	3.140
...	...	...	...
28815	False	False	4.553
28816	False	False	4.553
28817	False	False	4.553
28818	False	False	4.553
28819	False	False	4.553

[28820 rows x 58 columns]

```
[36]: %%capture
# Save the processed results to the same HDF5 store file
df.to_hdf(experiment_results, key=f'processed_results_{experiment_timestamp}')
```

## 5 Control Parameters

```
[37]: from radcad.core import generate_parameter_sweep

param_sweep = generate_parameter_sweep(params)
```

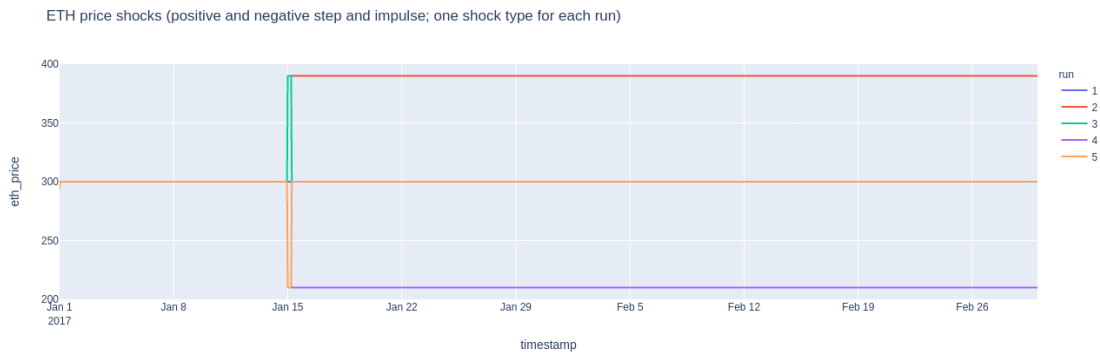
```
[38]: df_control_parameters = df[['subset', 'kp', 'ki']]

df_control_parameters = df_control_parameters.drop_duplicates(subset=['kp',
↳ 'ki'])
df_control_parameters
```

```
[38]: subset      kp      ki
      0      0  0.000001 -1.000000e-09
```

## 6 Simulation Analysis

```
[39]: df.query('subset == 0')[['timestamp', 'eth_price', 'run']].plot(
      title="ETH price shocks (positive and negative step and impulse; one shock_
      ↳type for each run)",
      x='timestamp',
      y='eth_price',
      color='run'
    )
```



```
[ ]: fig = px.line(
      df.query('run == 1'),
      title="Price response for all control parameter subsets, first run",
      x="timestamp",
      y=["market_price", "market_price_twap", "target_price_scaled"],
      facet_col="ki",
      facet_row="kp",
      height=1000
    )
fig.show()
```

Get the initial target price to test stability conditions:

```
[44]: initial_target_price = df['target_price'].iloc[0]
      initial_target_price
```

```
[44]: 3.14
```

Find all controller constant subsets where the price goes to zero:

```
[45]: df_market_price_zero = df.query("market_price <= 0.1*@initial_target_price")
df_market_price_zero[['subset', 'kp', 'ki']].drop_duplicates(subset=['kp', 'ki'])
```

```
[45]: Empty DataFrame
Columns: [subset, kp, ki]
Index: []
```

Find all controller constant subsets where the price goes to infinity:

```
[46]: df_market_price_infinity = df.query("market_price > 10*@initial_target_price")
df_market_price_infinity[['subset', 'kp', 'ki']].drop_duplicates(subset=['kp', 'ki'])
```

```
[46]: Empty DataFrame
Columns: [subset, kp, ki]
Index: []
```

Create dataframe of stable simulation scenarios.

Stability is defined as: 1. The market price and scaled target price remaining within 0.1x and 10x the starting price, for all timesteps

```
[47]: df['stable_price'] = False
df.loc[df.eval("""
0.1*@initial_target_price < market_price <= 10*@initial_target_price and 0.
↳ 1*@initial_target_price < target_price_scaled <= 10*@initial_target_price
"""), 'stable_price'] = True
df_stable_price = df.groupby("subset").filter(lambda x: all(x.query('timestep >=
↳ 24*2'))['stable_price'])) # and x['timestep'].max() == SIMULATION_TIMESTEPS
df_stable_price['subset'].unique()
```

```
[47]: array([0, 1, 2, 3])
```

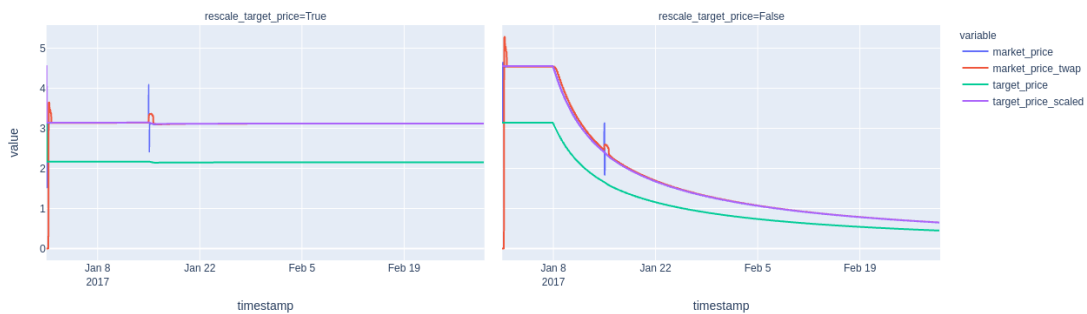
```
[63]: fig = px.line(
    df_stable_price.query('run == 1 and controller_enabled == True'),
    title="Base case: Stable ETH price response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price", "target_price_scaled"],
    facet_col="rescale_target_price",
    facet_col_wrap=2,
)
# fig.for_each_annotation(lambda a: a.update(text = f"kp={param_sweep[int(a.
↳ text.split('=')[-1])]['kp']} ki={param_sweep[int(a.text.
↳ split('=')[-1])]['ki']}"))
fig.show()
```

Base case: Stable ETH price response



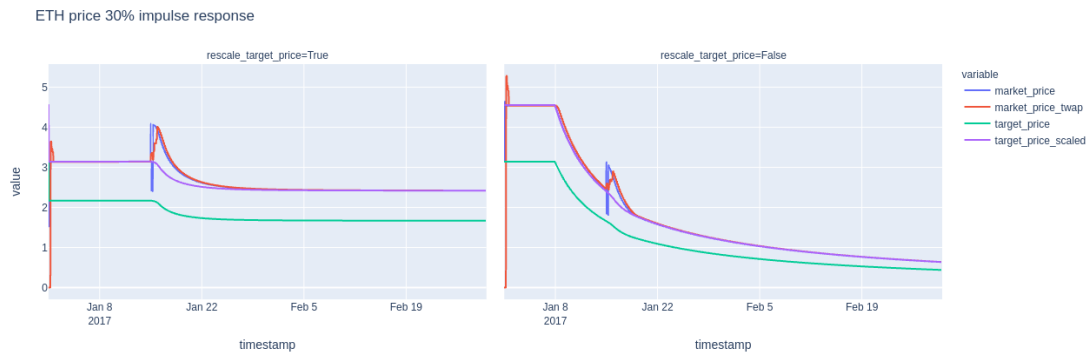
```
[64]: fig = px.line(
    df_stable_price.query('run == 2 and controller_enabled == True'),
    title="ETH price 30% step response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price",
    ↪ "target_price_scaled"],
    facet_col="rescale_target_price",
    facet_col_wrap=2,
)
fig.show()
```

ETH price 30% step response

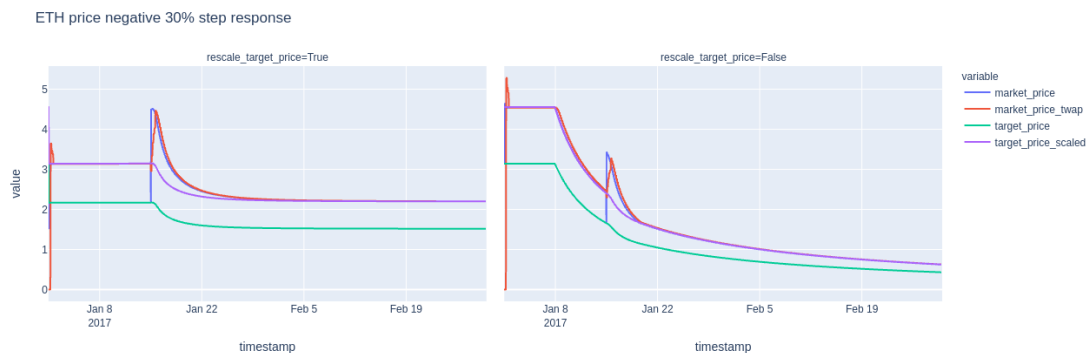


```
[65]: fig = px.line(
    df_stable_price.query('run == 3 and controller_enabled == True'),
    title="ETH price 30% impulse response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price",
    ↪ "target_price_scaled"],
    facet_col="rescale_target_price",
    facet_col_wrap=2,
)
```

```
)
fig.show()
```



```
[66]: fig = px.line(
    df_stable_price.query('run == 4 and controller_enabled == True'),
    title="ETH price negative 30% step response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price", "target_price_scaled"],
    facet_col="rescale_target_price",
    facet_col_wrap=2,
)
fig.show()
```

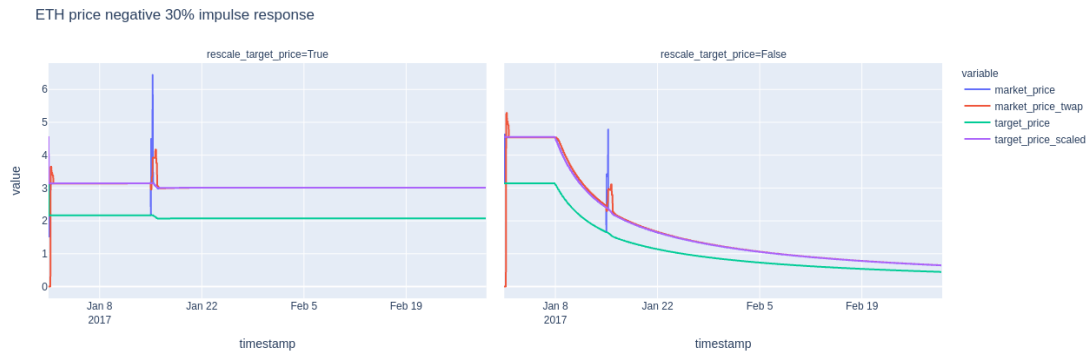


```
[67]: fig = px.line(
    df_stable_price.query('run == 5 and controller_enabled == True'),
    title="ETH price negative 30% impulse response",
    x="timestamp",
```

```

y=["market_price", "market_price_twap", "target_price",
↪ "target_price_scaled"],
facet_col="rescale_target_price",
facet_col_wrap=2,
)
fig.show()

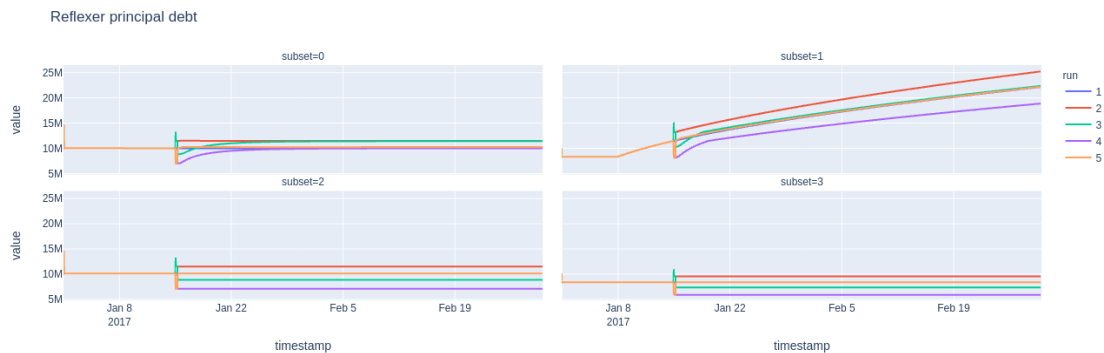
```



```

[68]: fig = px.line(
df_stable_price,
title="Reflexer principal debt",
x="timestamp",
y=["principal_debt"],
color='run',
facet_col="subset",
facet_col_wrap=2
)
fig.show()

```

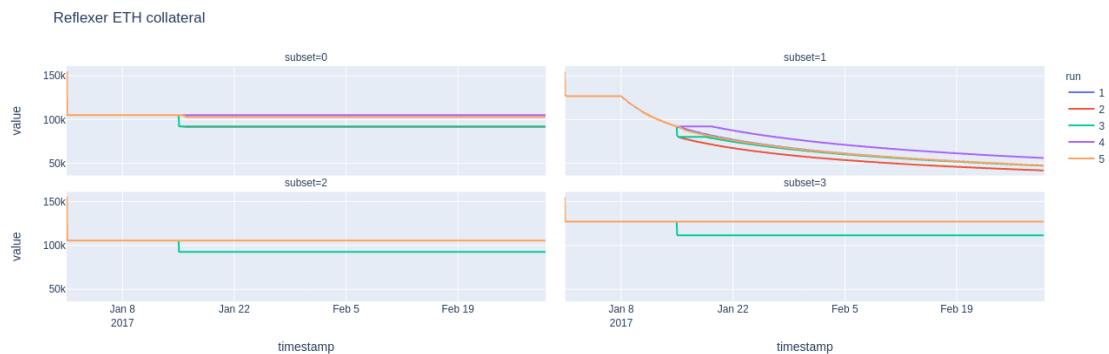




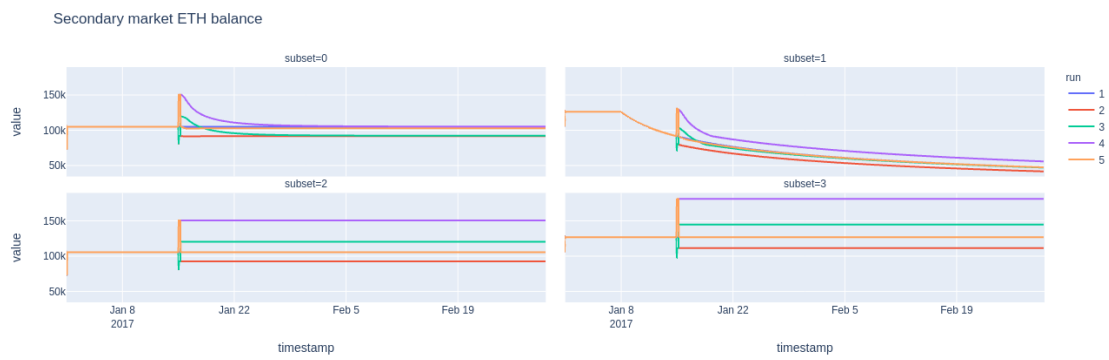
```
[69]: fig = px.line(
    df_stable_price,
    title="Secondary market RAI balance",
    x="timestamp",
    y=["RAI_balance"],
    color='run',
    facet_col="subset",
    facet_col_wrap=2
)
fig.show()
```



```
[70]: fig = px.line(
    df_stable_price,
    title="Reflexer ETH collateral",
    x="timestamp",
    y=["eth_collateral"],
    color='run',
    facet_col="subset",
    facet_col_wrap=2
)
fig.show()
```



```
[71]: fig = px.line(
    df_stable_price,
    title="Secondary market ETH balance",
    x="timestamp",
    y=["ETH_balance"],
    color='run',
    facet_col="subset",
    facet_col_wrap=2
)
fig.show()
```



```
[32]: df_stable_price.plot(
    x='timestamp',
    y=['collateralization_ratio'],
    title='Collateralization ratio',
    facet_col="subset",
)
```

