

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

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: pathos, pandas, tables, ray, boto3
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',
'/processed_results_2021-02-09T11:59:26.345803',
'/processed_results_2021-02-09T15:57:01.325820',
'/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',
'/results_2021-02-09T15:57:01.325820',
'/results_2021-02-09T18:46:33.073363']

```

```

[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',
'/exceptions_2021-02-09T15:57:01.325820',
'/exceptions_2021-02-09T18:46:33.073363']

```

```

[8]: # Copy a results_ key from the above keys to select the experiment
experiment_results_key = 'results_2021-02-09T18:46:33.073363' # Or select last_
↳ result: experiment_results_keys[-1]
experiment_timestamp = experiment_results_key.strip('results_')
exceptions_key = 'exceptions_' + experiment_timestamp
experiment_timestamp

```

```

[8]: '2021-02-09T18:46:33.073363'

```

```

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

```

```

[9]:
      cdp_metrics optimal_values \
3240120 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_... {}
3240121 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_... {}
3240122 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_... {}
3240123 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_... {}
3240124 {'cdp_count': 1, 'open_cdp_count': 1, 'closed_... {}

      sim_metrics timedelta cumulative_time timestamp blockheight \
3240120 {} 3600 5184000 2017-03-02 0
3240121 {} 3600 5184000 2017-03-02 0
3240122 {} 3600 5184000 2017-03-02 0
3240123 {} 3600 5184000 2017-03-02 0
3240124 {} 3600 5184000 2017-03-02 0

      eth_price liquidity_demand liquidity_demand_mean \
3240120 300.0 0.0 0.0
3240121 300.0 0.0 0.0
3240122 300.0 0.0 0.0
3240123 300.0 0.0 0.0
3240124 300.0 0.0 0.0

      cdps eth_collateral \
3240120 open time locked drawn ... 87837.975477
3240121 open time locked drawn ... 87837.975477
3240122 open time locked drawn ... 87837.975477
3240123 open time locked drawn ... 87837.975477
3240124 open time locked drawn ... 87837.975477

      eth_locked eth_freed eth_bitten principal_debt \
3240120 154827.528922 66989.553445 0.0 1.221319e+07
3240121 154827.528922 66989.553445 0.0 1.221319e+07
3240122 154827.528922 66989.553445 0.0 1.221319e+07
3240123 154827.528922 66989.553445 0.0 1.221319e+07
3240124 154827.528922 66989.553445 0.0 1.221319e+07

      rai_drawn rai_wiped rai_bitten accrued_interest \
3240120 2.048538e+07 8.272184e+06 0.0 9603.895316
3240121 2.048538e+07 8.272184e+06 0.0 9603.895316
3240122 2.048538e+07 8.272184e+06 0.0 9603.895316
3240123 2.048538e+07 8.272184e+06 0.0 9603.895316
3240124 2.048538e+07 8.272184e+06 0.0 9603.895316

      interest_dripped interest_wiped interest_bitten w_1 w_2 \
3240120 0 0 0 3.480488e-08 0.0
3240121 0 0 0 3.480488e-08 0.0
3240122 0 0 0 3.480488e-08 0.0
3240123 0 0 0 3.480488e-08 0.0

```

3240124		0		0		0	3.480488e-08	0.0
---------	--	---	--	---	--	---	--------------	-----

	w_3	system_revenue	stability_fee	market_price	market_price_twap	\
3240120	0.0	0.0	1.585490e-10	2.157619	2.15762	
3240121	0.0	0.0	1.585490e-10	2.157619	2.15762	
3240122	0.0	0.0	1.585490e-10	2.157619	2.15762	
3240123	0.0	0.0	1.585490e-10	2.157619	2.15762	
3240124	0.0	0.0	1.585490e-10	2.157619	2.15762	

	target_price	target_rate	eth_return	eth_gross_return	\
3240120	1.488012	-1.401514e-12	0.0	1.0	
3240121	1.488012	-1.401514e-12	0.0	1.0	
3240122	1.488012	-1.401514e-12	0.0	1.0	
3240123	1.488012	-1.401514e-12	0.0	1.0	
3240124	1.488012	-1.401514e-12	0.0	1.0	

	expected_market_price	expected_debt_price	error_star	\
3240120	2.22051		3.14	-0.000002
3240121	2.22051		3.14	-0.000002
3240122	2.22051		3.14	-0.000002
3240123	2.22051		3.14	-0.000002
3240124	2.22051		3.14	-0.000002

	error_star_integral	market_slippage	RAI_balance	ETH_balance	\
3240120	-553.008363	9.970634e-09	1.221319e+07	87838.062495	
3240121	-553.008363	9.970634e-09	1.221319e+07	87838.062495	
3240122	-553.008363	9.970634e-09	1.221319e+07	87838.062495	
3240123	-553.008363	9.970634e-09	1.221319e+07	87838.062495	
3240124	-553.008363	9.970634e-09	1.221319e+07	87838.062495	

	UNI_supply	uniswap_oracle	\
3240120	10000000.0	<models.system_model_v3.model.parts.uniswap_or...	
3240121	10000000.0	<models.system_model_v3.model.parts.uniswap_or...	
3240122	10000000.0	<models.system_model_v3.model.parts.uniswap_or...	
3240123	10000000.0	<models.system_model_v3.model.parts.uniswap_or...	
3240124	10000000.0	<models.system_model_v3.model.parts.uniswap_or...	

	simulation	subset	run	substep	timestep	events
3240120	0	24	5	14	1440	NaN
3240121	0	24	5	15	1440	NaN
3240122	0	24	5	16	1440	NaN
3240123	0	24	5	17	1440	NaN
3240124	0	24	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    0        4        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
* Number of MC runs: 5
* Timestep duration: 0.004 seconds
* Control parameters: ['kp', 'ki']
* Number of parameter combinations: 25
* Expected experiment duration: 12.0 minutes / 0.2 hours
```

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

```
[13]: df = post_process_results(df_raw, params, set_params=['ki', 'kp', 'L',
↳ 'liquidity_ratio'])
df
```

```
[13]:
```

	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_...
...
180120	3240052	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
180121	3240070	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
180122	3240088	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
180123	3240106	{'cdp_count': 1, 'open_cdp_count': 1, 'closed_...
180124	3240124	{'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
...
180120	{}	{}	3600	5169600
180121	{}	{}	3600	5173200
180122	{}	{}	3600	5176800
180123	{}	{}	3600	5180400
180124	{}	{}	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
...
180120	2017-03-01 20:00:00	0	300.000000	-0.0
180121	2017-03-01 21:00:00	0	300.000000	-0.0
180122	2017-03-01 22:00:00	0	300.000000	0.0
180123	2017-03-01 23:00:00	0	300.000000	0.0
180124	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
...	...
180120	0.0000
180121	0.0000
180122	0.0000
180123	0.0000
180124	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	
...	
180120	None	87837.975477	
180121	None	87837.975477	
180122	None	87837.975477	
180123	open time	locked	drawn ... 87837.975477
180124	open time	locked	drawn ... 87837.975477

	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	
...	
180120	154827.528922	66989.553445	0.0	1.221319e+07	2.048538e+07	
180121	154827.528922	66989.553445	0.0	1.221319e+07	2.048538e+07	
180122	154827.528922	66989.553445	0.0	1.221319e+07	2.048538e+07	
180123	154827.528922	66989.553445	0.0	1.221319e+07	2.048538e+07	
180124	154827.528922	66989.553445	0.0	1.221319e+07	2.048538e+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	
...	
180120	8.272184e+06	0.0	9575.989424	0	
180121	8.272184e+06	0.0	9582.965891	0	
180122	8.272184e+06	0.0	9589.942362	0	
180123	8.272184e+06	0.0	9596.918837	0	
180124	8.272184e+06	0.0	9603.895316	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	
...	
180120	0	0	3.443432e-08	0.0	0.0	
180121	0	0	3.443432e-08	0.0	0.0	
180122	0	0	3.480488e-08	0.0	0.0	
180123	0	0	3.480488e-08	0.0	0.0	
180124	0	0	3.480488e-08	0.0	0.0	

	system_revenue	stability_fee	market_price	market_price_twap	\
0	0.0	1.585490e-10	3.140000		0.00000
1	0.0	1.585490e-10	3.140000		0.00000
2	0.0	1.585490e-10	1.525000		0.00000
3	0.0	1.585490e-10	3.133425		0.00000
4	0.0	1.585490e-10	3.133425		0.00000
...	
180120	0.0	1.585490e-10	2.157620		2.15762
180121	0.0	1.585490e-10	2.157620		2.15762
180122	0.0	1.585490e-10	2.157619		2.15762
180123	0.0	1.585490e-10	2.157619		2.15762
180124	0.0	1.585490e-10	2.157619		2.15762

	target_price	target_rate	eth_return	eth_gross_return	\
0	3.140000	0.000000e+00	0.000000		0.0
1	2.165517	0.000000e+00	0.020168		1.0
2	2.165517	0.000000e+00	0.000000		1.0
3	2.165517	0.000000e+00	0.000000		1.0
4	2.165517	0.000000e+00	0.000000		1.0
...	
180120	1.488012	-1.372091e-12	0.000000		1.0
180121	1.488012	-1.386934e-12	0.000000		1.0
180122	1.488012	-1.386934e-12	0.000000		1.0
180123	1.488012	-1.386934e-12	0.000000		1.0
180124	1.488012	-1.401514e-12	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	
...	

180120	2.220509	3.14	-0.000002
180121	2.220509	3.14	-0.000002
180122	2.220510	3.14	-0.000002
180123	2.220510	3.14	-0.000002
180124	2.220510	3.14	-0.000002

	error_star_integral	market_slippage	RAI_balance	ETH_balance \
0	0.000000	0.000000e+00	1.000000e+07	106777.606153
1	5652.000000	0.000000e+00	1.450000e+07	73708.353400
2	16932.000000	5.143311e-01	1.012023e+07	105703.331685
3	28166.000000	-1.054705e+00	1.012023e+07	105703.331685
4	39354.000000	0.000000e+00	1.012023e+07	105703.331685
...
180120	-565.008434	9.864478e-09	1.221319e+07	87838.064239
180121	-562.008473	9.864479e-09	1.221319e+07	87838.063807
180122	-559.008399	9.864479e-09	1.221319e+07	87838.063369
180123	-556.008324	9.970634e-09	1.221319e+07	87838.062932
180124	-553.008363	9.970634e-09	1.221319e+07	87838.062495

	UNI_supply	uniswap_oracle \
0	10000000.0	None
1	10000000.0	None
2	10000000.0	None
3	10000000.0	None
4	10000000.0	None
...
180120	10000000.0	None
180121	10000000.0	None
180122	10000000.0	None
180123	10000000.0	<models.system_model_v3.model.parts.uniswap_or...
180124	10000000.0	<models.system_model_v3.model.parts.uniswap_or...

	simulation	subset	run	substep	timestep	events \
0	0	0	1	0	0	NaN
1	0	0	1	18	1	NaN
2	0	0	1	18	2	NaN
3	0	0	1	18	3	NaN
4	0	0	1	18	4	NaN
...
180120	0	24	5	18	1436	NaN
180121	0	24	5	18	1437	NaN
180122	0	24	5	18	1438	NaN
180123	0	24	5	18	1439	NaN
180124	0	24	5	18	1440	NaN

	eth_collateral_value	collateralization_ratio	ki \
0	4.553000e+07	1.450000	-5.000000e-09

1	4.644826e+07	1.479244	-5.000000e-09
2	3.177754e+07	1.450000	-5.000000e-09
3	3.177754e+07	1.450000	-5.000000e-09
4	3.177754e+07	1.450000	-5.000000e-09
...
180120	2.635139e+07	1.450000	-2.000000e-10
180121	2.635139e+07	1.450000	-2.000000e-10
180122	2.635139e+07	1.450000	-2.000000e-10
180123	2.635139e+07	1.450000	-2.000000e-10
180124	2.635139e+07	1.450000	-2.000000e-10

	kp	liquidation_ratio	target_price_scaled
0	2.000000e-07	1.45	4.553000
1	2.000000e-07	1.45	3.140000
2	2.000000e-07	1.45	3.140000
3	2.000000e-07	1.45	3.140000
4	2.000000e-07	1.45	3.140000
...
180120	5.000000e-06	1.45	2.157617
180121	5.000000e-06	1.45	2.157617
180122	5.000000e-06	1.45	2.157617
180123	5.000000e-06	1.45	2.157617
180124	5.000000e-06	1.45	2.157617

[180125 rows x 56 columns]

```
[14]: %%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

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

param_sweep = generate_parameter_sweep(params)
```

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

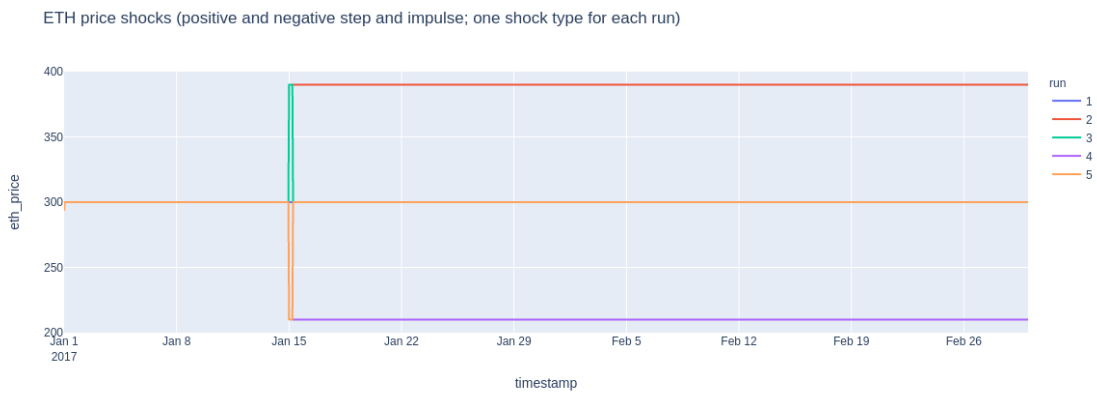
df_control_parameters = df_control_parameters.drop_duplicates(subset=['kp', '
↳ 'ki'])
df_control_parameters
```

```
[16]:      subset      kp      ki
0         0  2.000000e-07 -5.000000e-09
1441      1  2.000000e-07 -3.000000e-09
2882      2  2.000000e-07 -1.000000e-09
```

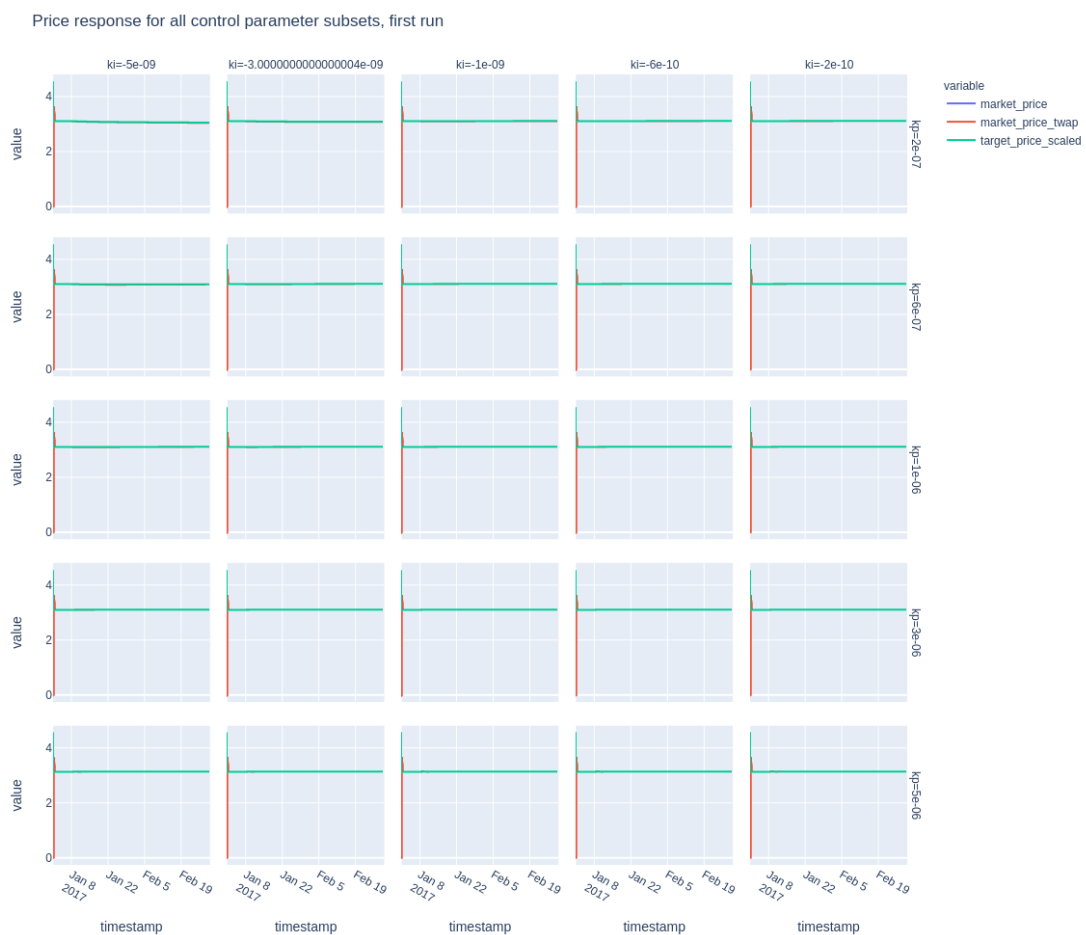
4323	3	2.000000e-07	-6.000000e-10
5764	4	2.000000e-07	-2.000000e-10
7205	5	6.000000e-07	-5.000000e-09
8646	6	6.000000e-07	-3.000000e-09
10087	7	6.000000e-07	-1.000000e-09
11528	8	6.000000e-07	-6.000000e-10
12969	9	6.000000e-07	-2.000000e-10
14410	10	1.000000e-06	-5.000000e-09
15851	11	1.000000e-06	-3.000000e-09
17292	12	1.000000e-06	-1.000000e-09
18733	13	1.000000e-06	-6.000000e-10
20174	14	1.000000e-06	-2.000000e-10
21615	15	3.000000e-06	-5.000000e-09
23056	16	3.000000e-06	-3.000000e-09
24497	17	3.000000e-06	-1.000000e-09
25938	18	3.000000e-06	-6.000000e-10
27379	19	3.000000e-06	-2.000000e-10
28820	20	5.000000e-06	-5.000000e-09
30261	21	5.000000e-06	-3.000000e-09
31702	22	5.000000e-06	-1.000000e-09
33143	23	5.000000e-06	-6.000000e-10
34584	24	5.000000e-06	-2.000000e-10

6 Simulation Analysis

```
[17]: 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'
)
```



```
[18]: 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:

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

[19]: 3.14

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

```
[20]: 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'])
```

[20]: Empty DataFrame
Columns: [subset, kp, ki]
Index: []

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

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

[21]: Empty DataFrame
Columns: [subset, kp, ki]
Index: []

Create dataframe of stable simulation scenarios.

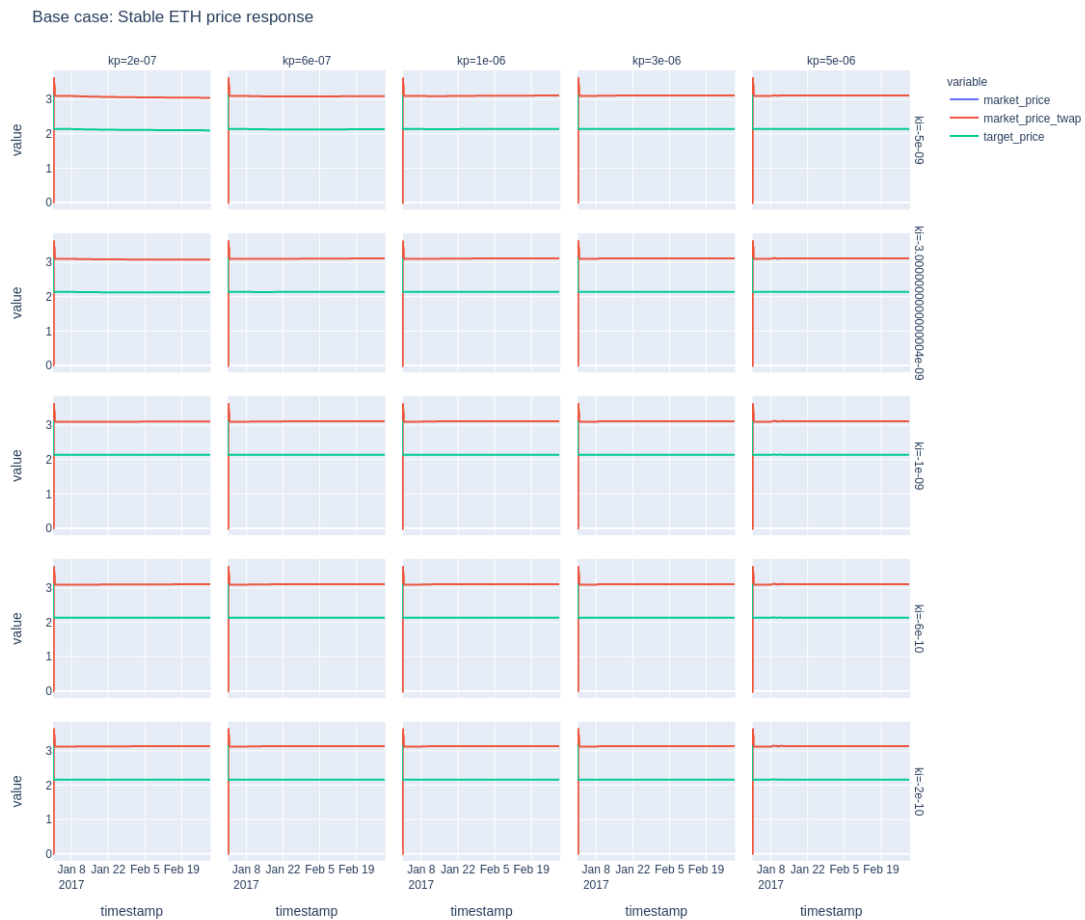
Stability is defined as: 1. The market price and scaled target price remaining within 0.1x and 10x the starting price, for all timesteps

```
[22]: 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()
```

[22]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
 17, 18, 19, 20, 21, 22, 23, 24])

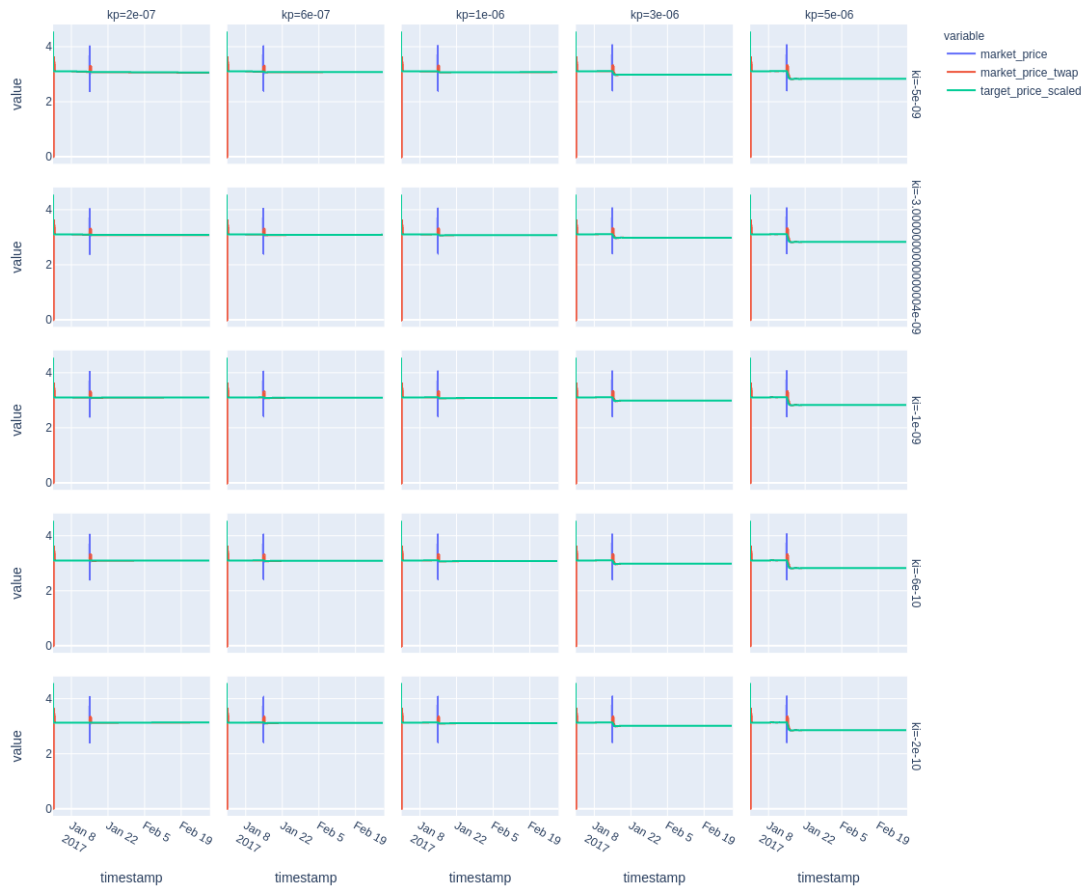
```
[23]: fig = px.line(
    df_stable_price.query('run == 1'),
    title="Base case: Stable ETH price response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price"],
    facet_col="kp",
    facet_row="ki",
    facet_col_wrap=2,
    height=1000
```

```
)
# 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()
```



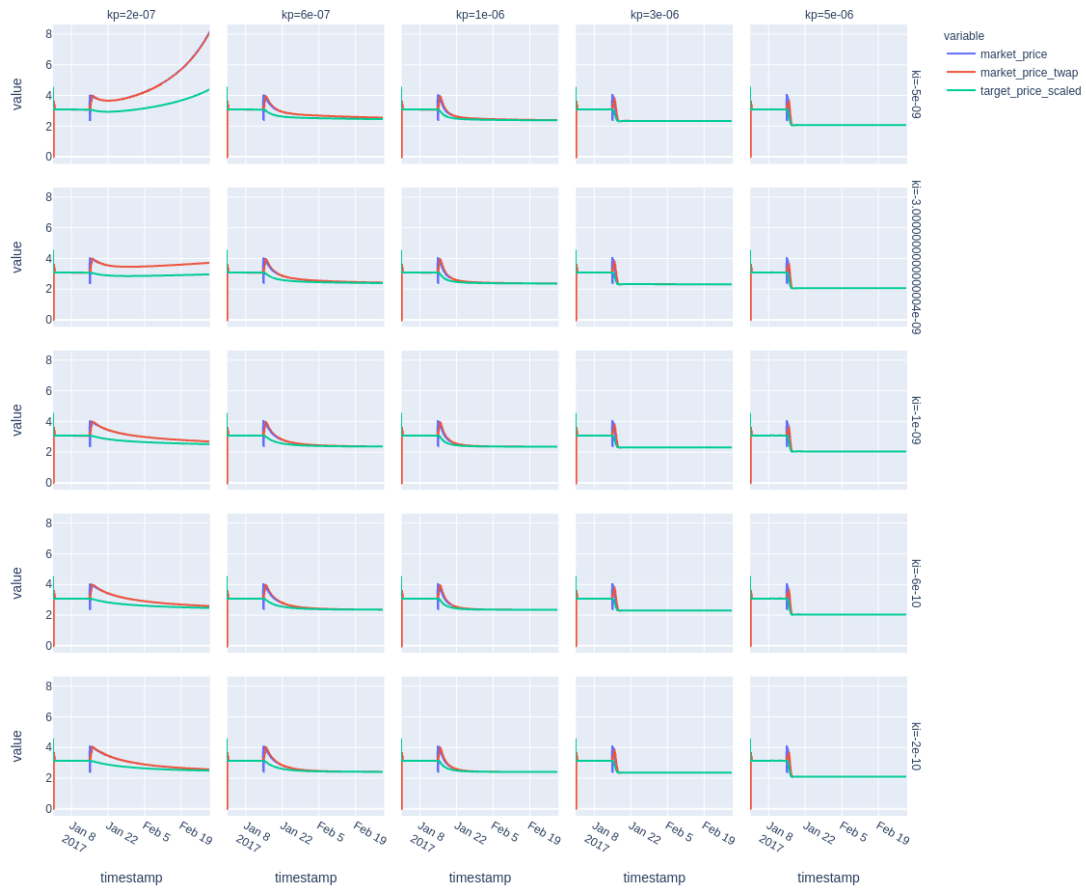
```
[24]: fig = px.line(
    df_stable_price.query('run == 2'),
    title="ETH price 30% step response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price_scaled"],
    facet_col="kp",
    facet_row="ki",
    facet_col_wrap=2,
    height=1000
)
fig.show()
```

ETH price 30% step response

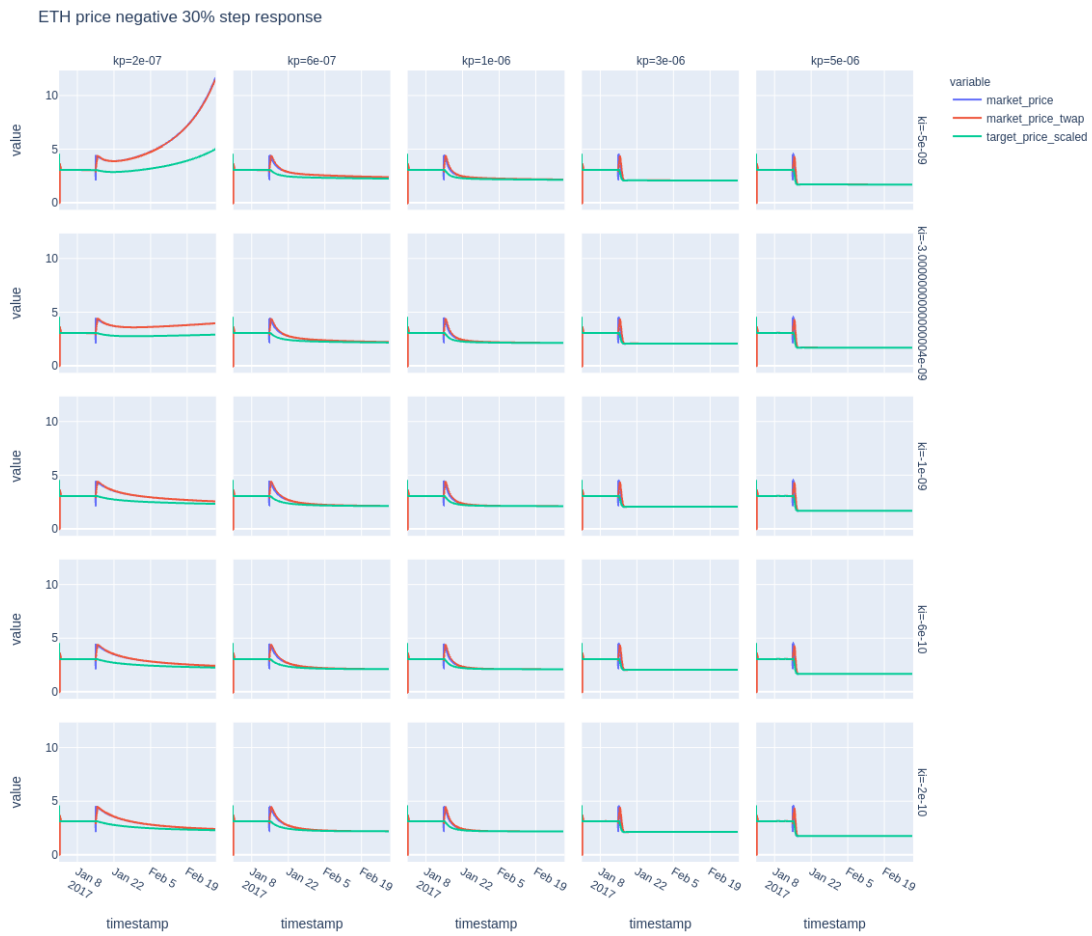


```
[25]: fig = px.line(
    df_stable_price.query('run == 3'),
    title="ETH price 30% impulse response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price_scaled"],
    facet_col="kp",
    facet_row="ki",
    facet_col_wrap=2,
    height=1000
)
fig.show()
```


ETH price 30% impulse response

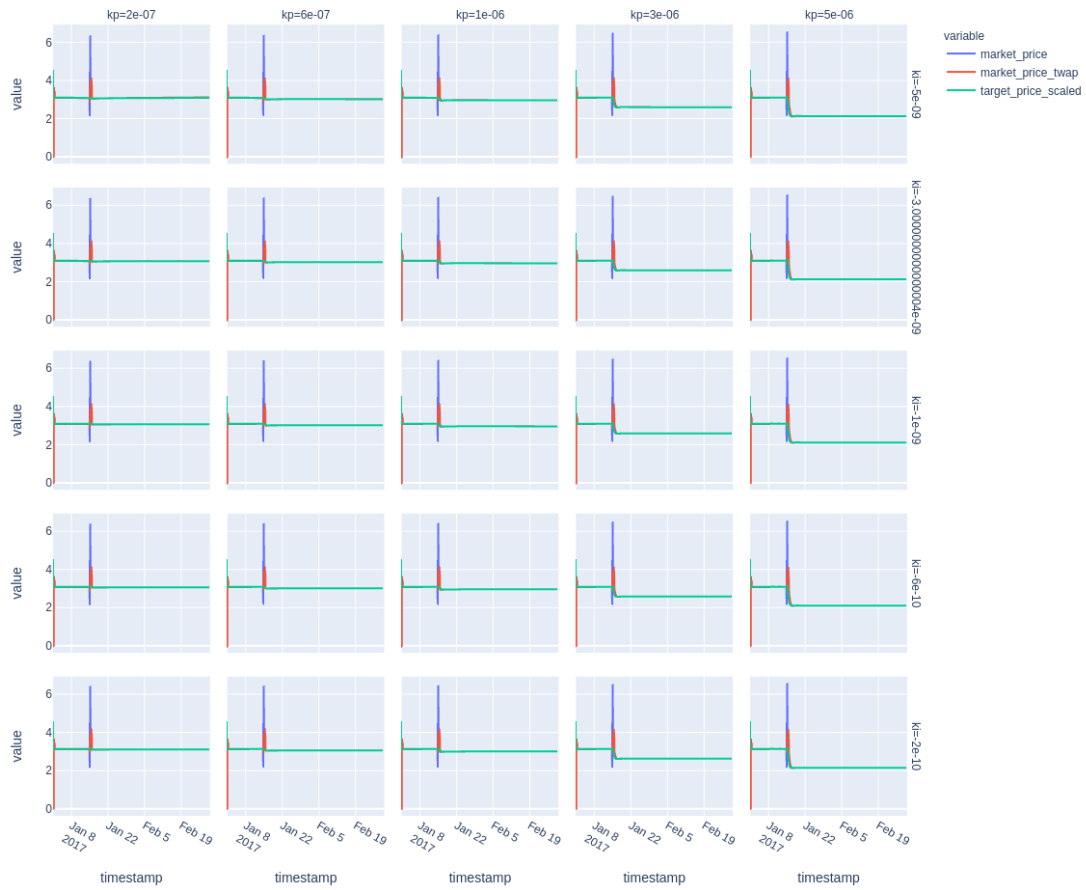


```
[26]: fig = px.line(
    df_stable_price.query('run == 4'),
    title="ETH price negative 30% step response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price_scaled"],
    facet_col="kp",
    facet_row="ki",
    facet_col_wrap=2,
    height=1000
)
fig.show()
```

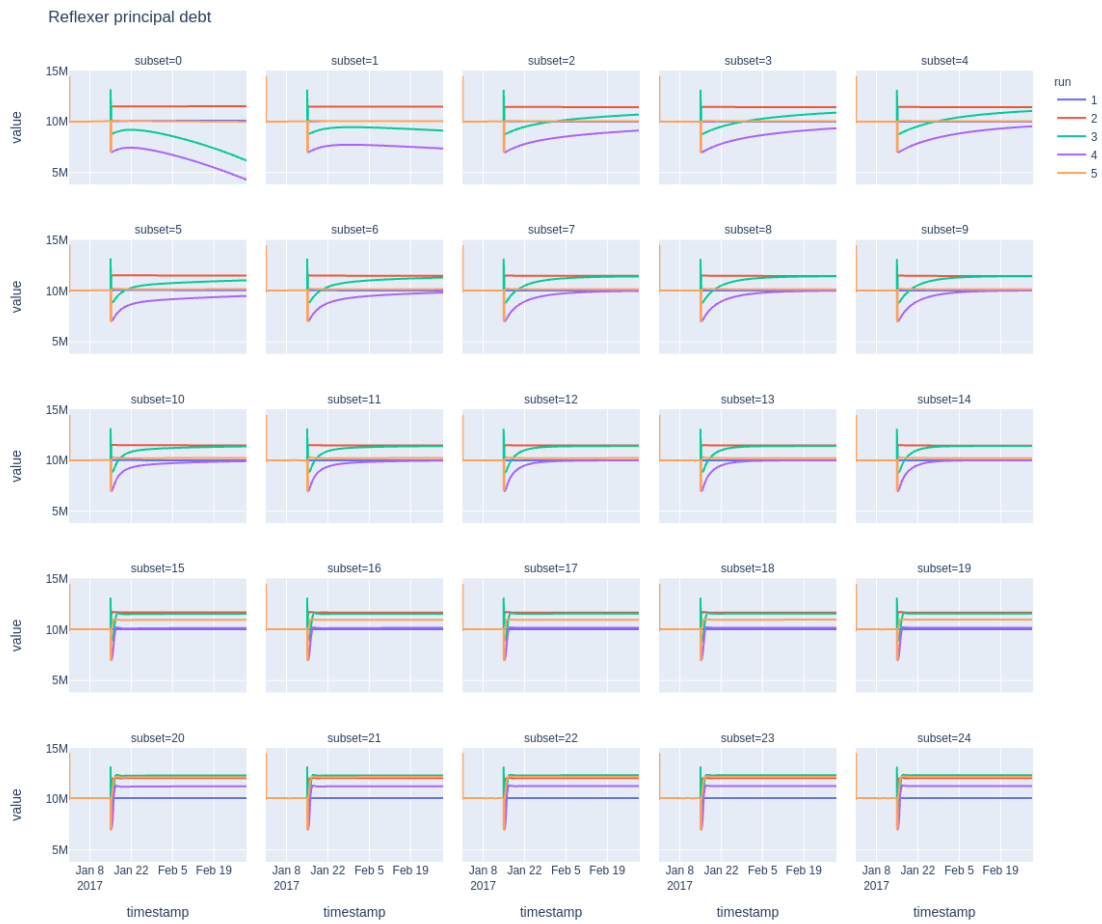


```
[27]: fig = px.line(
    df_stable_price.query('run == 5'),
    title="ETH price negative 30% impulse response",
    x="timestamp",
    y=["market_price", "market_price_twap", "target_price_scaled"],
    facet_col="kp",
    facet_row="ki",
    facet_col_wrap=2,
    height=1000
)
fig.show()
```

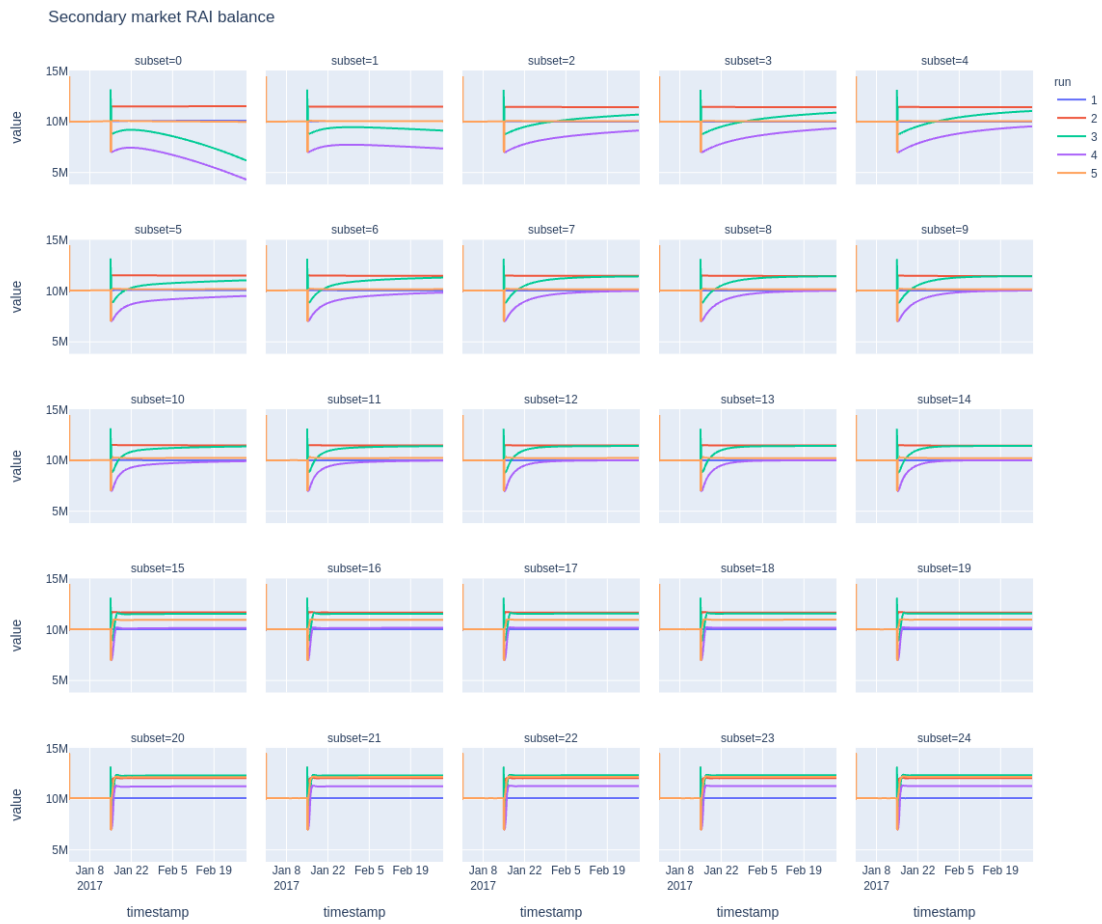
ETH price negative 30% impulse response



```
[34]: fig = px.line(
    df_stable_price,
    title="Reflexer principal debt",
    x="timestamp",
    y=["principal_debt"],
    color='run',
    facet_col="subset",
    facet_col_wrap=5,
    height=1000
)
fig.show()
```



```
[35]: fig = px.line(
    df_stable_price,
    title="Secondary market RAI balance",
    x="timestamp",
    y=["RAI_balance"],
    color='run',
    facet_col="subset",
    facet_col_wrap=5,
    height=1000
)
fig.show()
```



```
[36]: fig = px.line(
    df_stable_price,
    title="Reflexer ETH collateral",
    x="timestamp",
    y=["eth_collateral"],
    color='run',
    facet_col="subset",
    facet_col_wrap=5,
    height=1000
)
fig.show()
```



```
[37]: fig = px.line(
    df_stable_price,
    title="Secondary market ETH balance",
    x="timestamp",
    y=["ETH_balance"],
    color='run',
    facet_col="subset",
    facet_col_wrap=5,
    height=1000
)
fig.show()
```



```
[38]: df_stable_price.plot(
    x='timestamp',
    y=['collateralization_ratio'],
    title='Collateralization ratio',
    facet_col="subset",
    facet_col_wrap=5,
    height=1000
)
```

