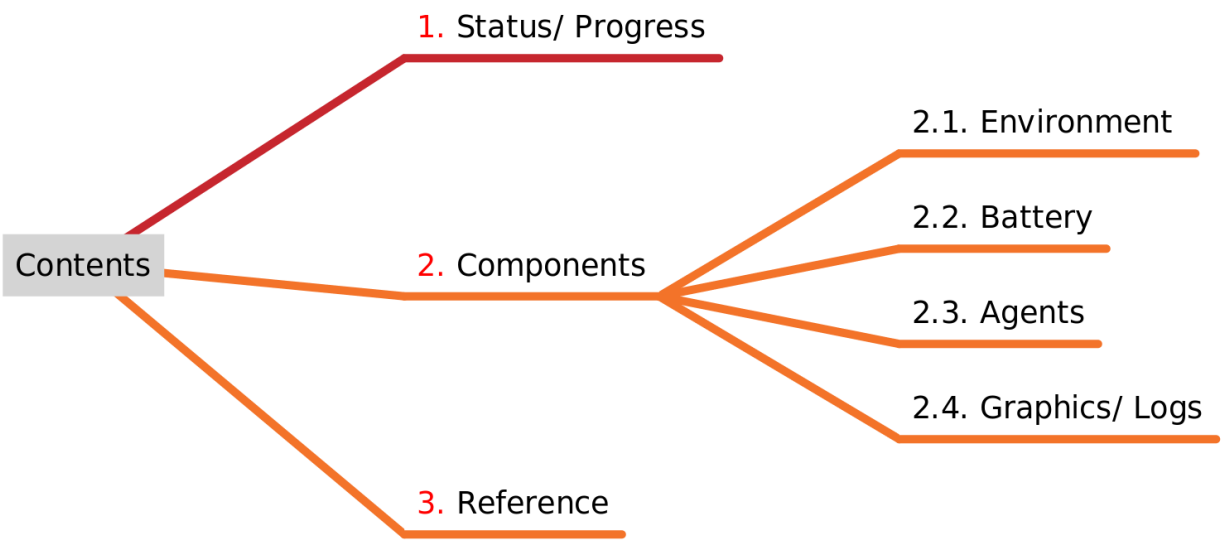
Training Prosumer Agents with Reinforcement Learning.

>>> **Biweekly Report 7.** ($2^{nd} June - 26^{th} June : 2024$)



---

1. Status/ Progress

Current Iteration

- [x] Search params, train/eval, and sample action iteration.
- [x] collection of observations

Next Iteration (Plan)

- [ ] performance metrics (policy based) and rule based actions comparisons
- [ ] collection of observations

- [ ] Documenting different components with graphics

---

2. Components

Taking feedback from weekly catchups into account and updated understanding of the system following changes were made to different components.

2.1. Environment Development

2.1.1 **Train/Eval/Test Env**

- the pv power sign is inverted.

2.1.2 **Rewards Estimation**

- the reward calculation is updated according to the change in sign of pv power.

2.2. Battery Module

- no change were made to the battery module.

2.3. Agent

- Algorithm in use:
  - Proximal Policy Optimization (PPO)
  - Soft Actor Critic (SAC)
  - Twin Delayed Deep Deterministic Policy Gradient (TD3)
- Different parameters for initialization(`model.init()`) and learning(`model.learn()`) were added and experimented, the results are shown in the graphics/logs section below.

2.4. Graphics/ Logs

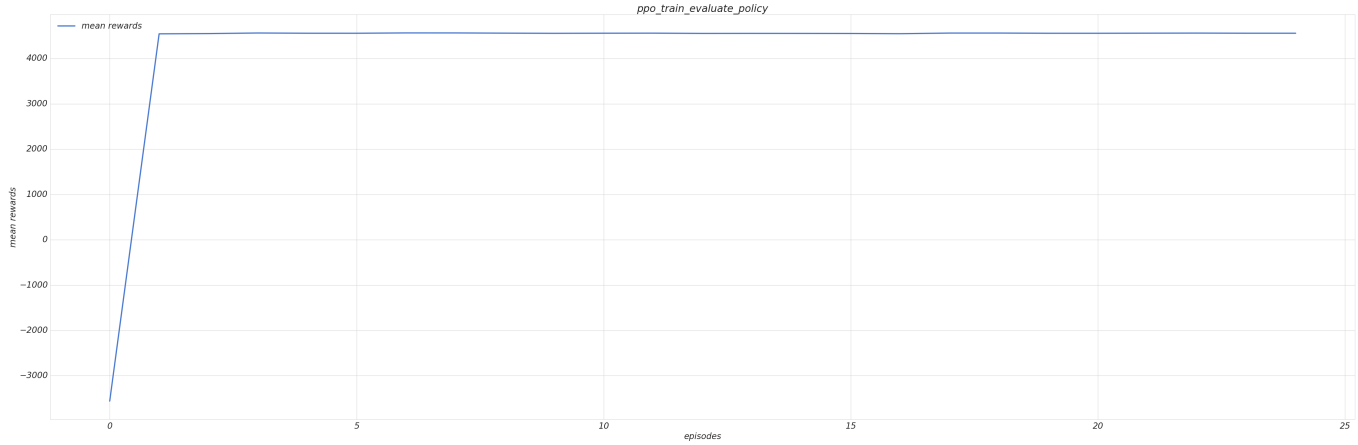
parameters config

```
data_config["train_test_split_ratio"] = 0.9 # (train 90% test 10%)
env_config: dict = {
    ...,
    "observation_window": int(4096),
    "num_envs": 1,
    ...,
}
policy_config: dict = {
    "policy_nw": "MlpPolicy",
    "reset_num_timesteps": False,
    "num_train_eval_cycles": 25,
    "num_retrain_eval_cycles": 25,
    "num_eval_episodes": 3,
    "num_test_episodes": 5,
    "train_timesteps": data_config["observation_window_train"],
    "retrain_timesteps": data_config["observation_window_train"],
}
ppo:
    - learning_rate: 0.0004,
    - gamma: 0.99,
    - n_steps: 4096,
    - clip_range: 0.25,
    - batch_size: 64,
    - net_arch: (pi, vf) [64, 32, 16],

sac:
    - learning_rate: 0.009,
    - gamma: 0.99,
    - tau: 0.074,
    - net_arch: (pi, qf) [64, 32, 16],

td3:
    - learning_rate: 0.003,
    - gamma: 0.99,
    - tau: 0.072,
    - net_arch: (pi, qf) [64, 32, 16],
```

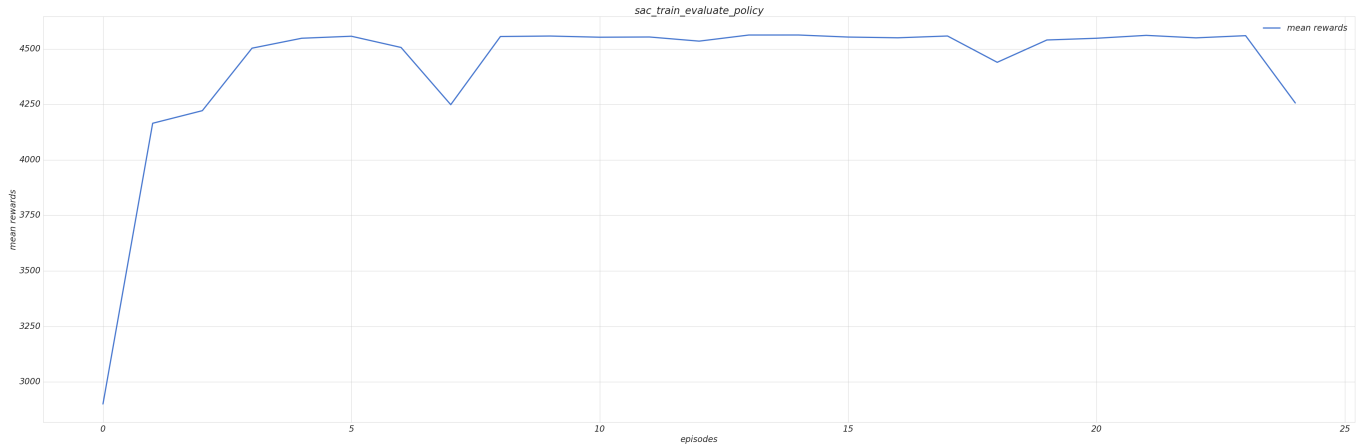
During Learning/ Evaluation

Evaluation of Mean Rewards per iteration : 25, with each model.

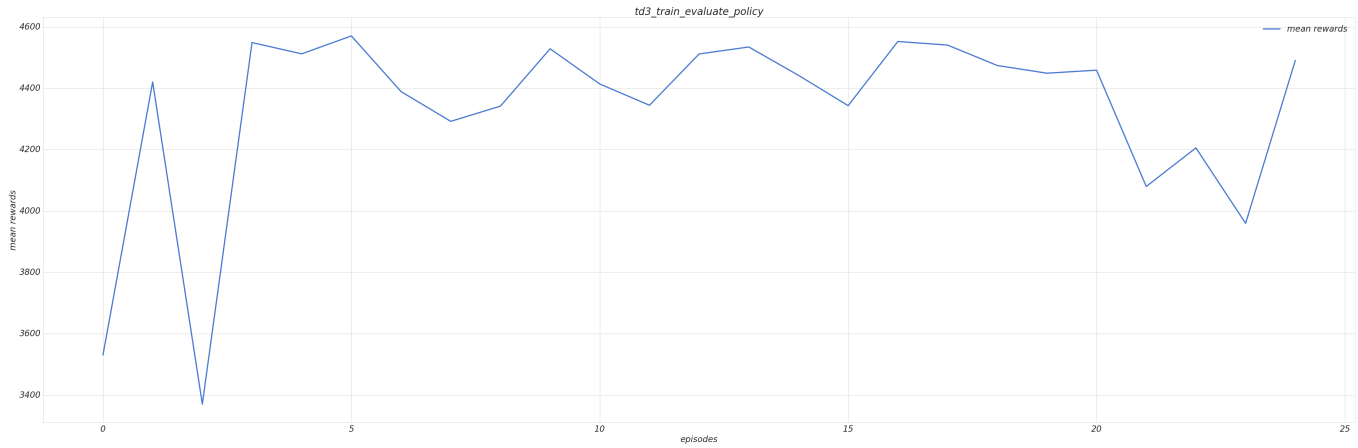
PPO



SAC

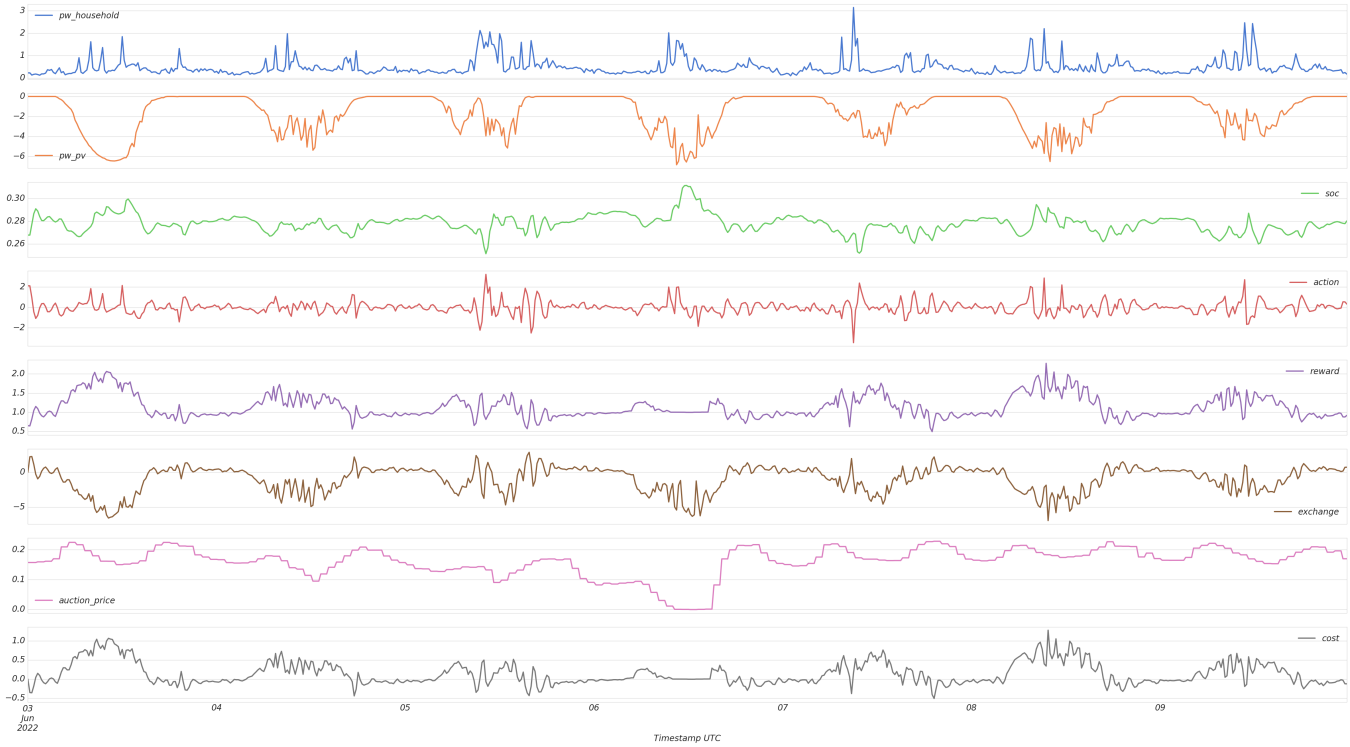


TD3

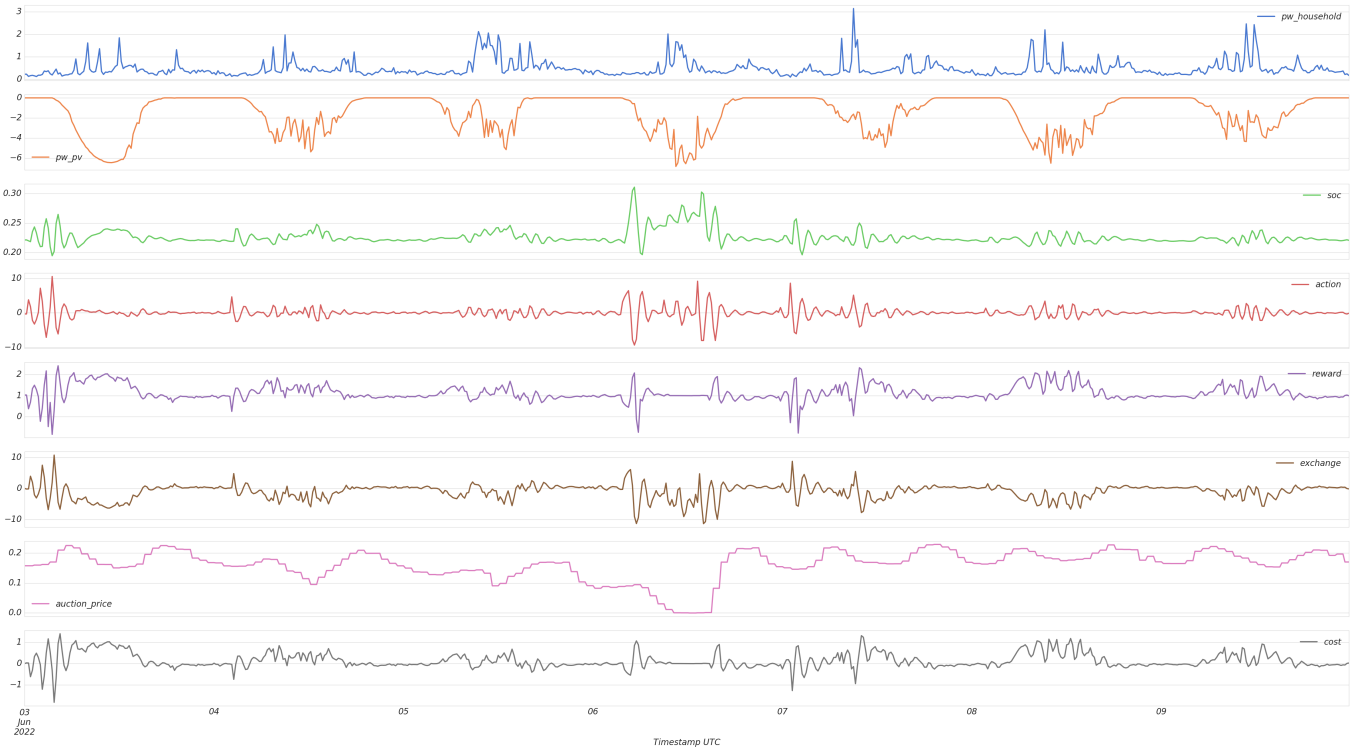


During Sampling Actions from trained policy (deterministic)

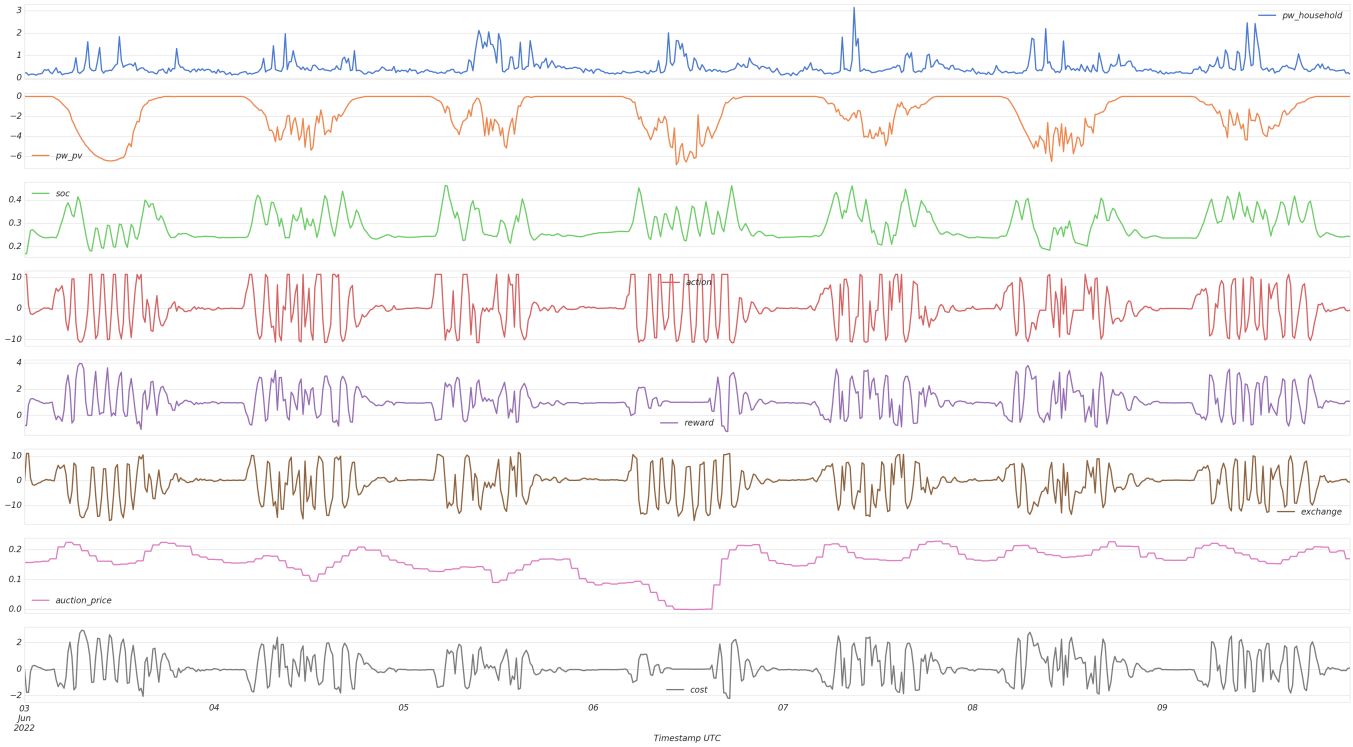
PPO


pw_household
pw_pv
soc
action
reward
exchange
auction_price
cost
03
Jun
2022
04
05
06
07
08
09
Timestamp UTC

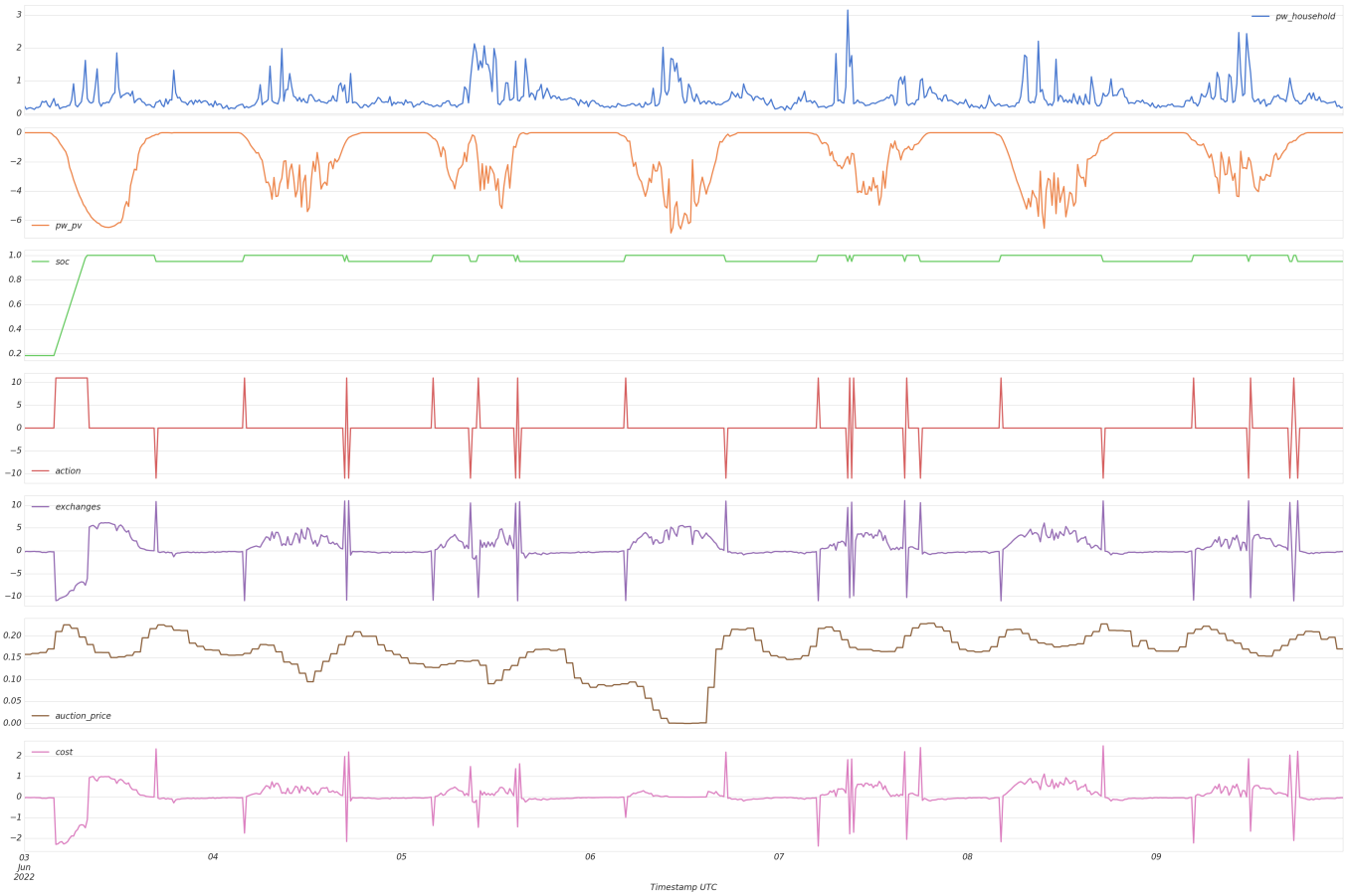
SAC


pw_household
pw_pv
soc
action
reward
exchange
auction_price
cost
03
Jun
2022
04
05
06
07
08
09
Timestamp UTC

TD3


pw_household
pw_pv
soc
action
reward
exchange
auction_price
cost
03
Jun
2022
04
05
06
07
08
09
Timestamp UTC

RBC


pw_household
pw_pv
soc
action
exchanges
auction_price
cost
03
Jun
2022
04
05
06
07
08
09
Timestamp UTC

Total Cost Comparison

Initial comparison with rule based vs rl models(PPO, SAC, TD3) was made on first 7 days of timesteps of test set as shown in the table below.

Models	**PPO**	**SAC**	**TD3**	**RBC**
Total Cost	2068.596	2068.223	2177.3386	2069.1265
