東京大学
THE UNIVERSITY OF TOKYO

日本学術振興会
Japan Society for the Promotion of Science

独立行政法人
科学技術振興機構
Japan Science and Technology Agency

CREST

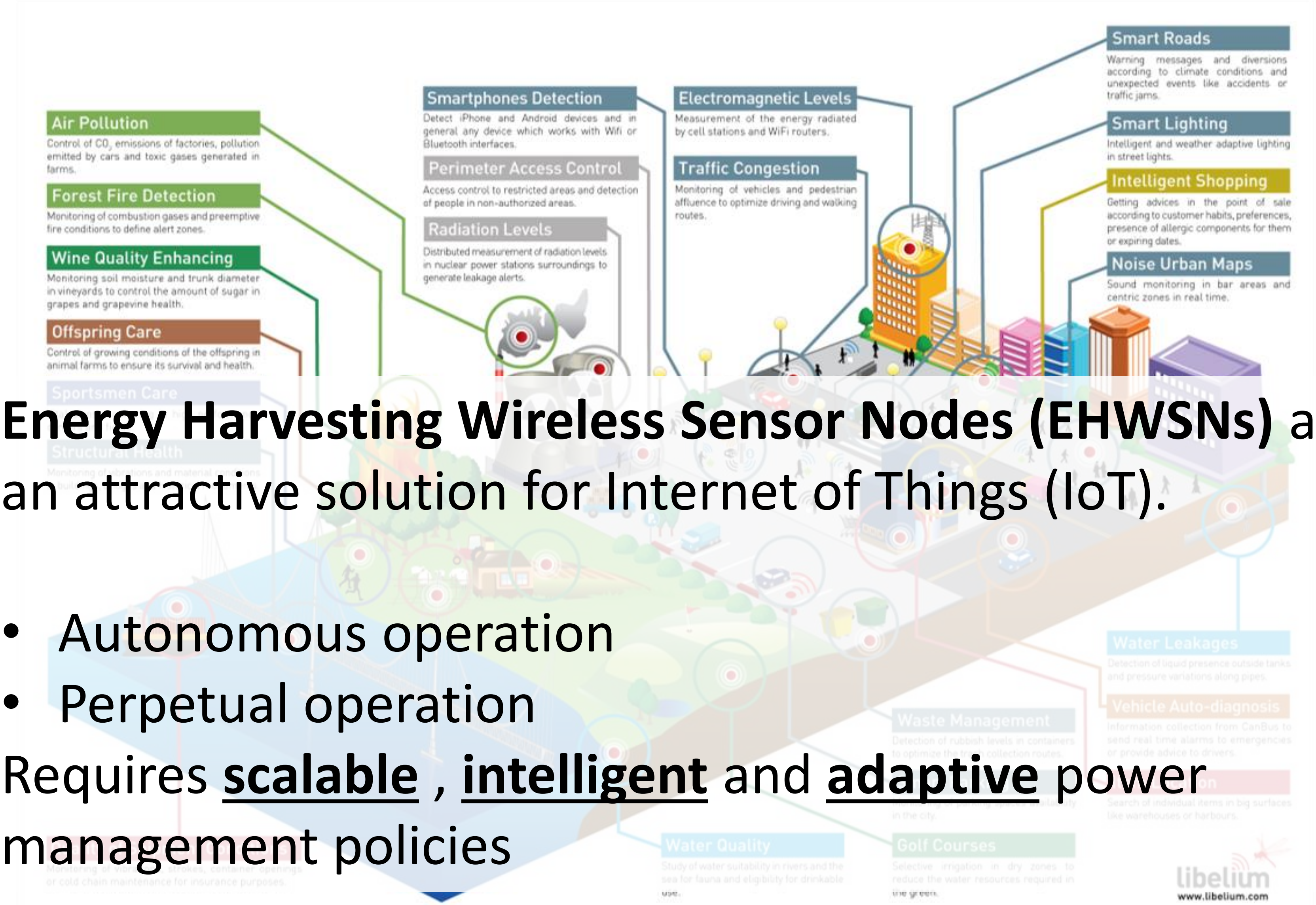
Power Management of Wireless Sensor Nodes with ***Coordinated Distributed Reinforcement Learning***

SHASWOT SHRESTHAMALI
MASAAKI KONDO
HIROSHI NAKAMURA

THE UNIVERSITY OF TOKYO

2 December 2019
ICCD 2019, Abu Dhabi

Wireless Sensor Networks for Internet of Things



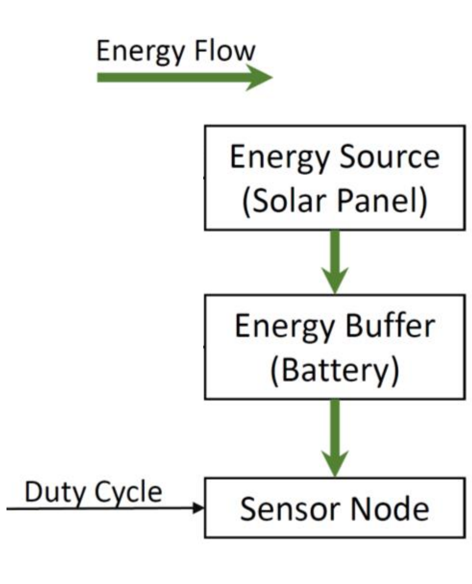
Energy Harvesting Wireless Sensor Nodes (EHWSNs) are an attractive solution for Internet of Things (IoT).

- Autonomous operation
- Perpetual operation

Requires **<u>scalable</u>** , **<u>intelligent</u>** and **<u>adaptive</u>** power management policies

http://www.libelium.com/wp-content/themes/libelium/images/content/applications/libelium_smart_world_infographic_big.png

ENO-RL System



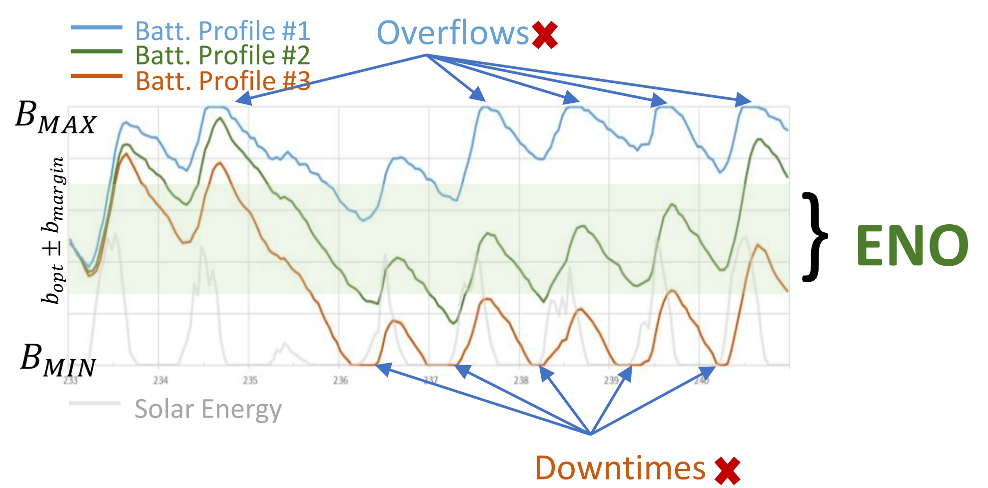
Energy Neutral Operation (ENO):

- harvested energy equals energy spent
- i.e., perpetual operation

ENO-RL OBJECTIVE

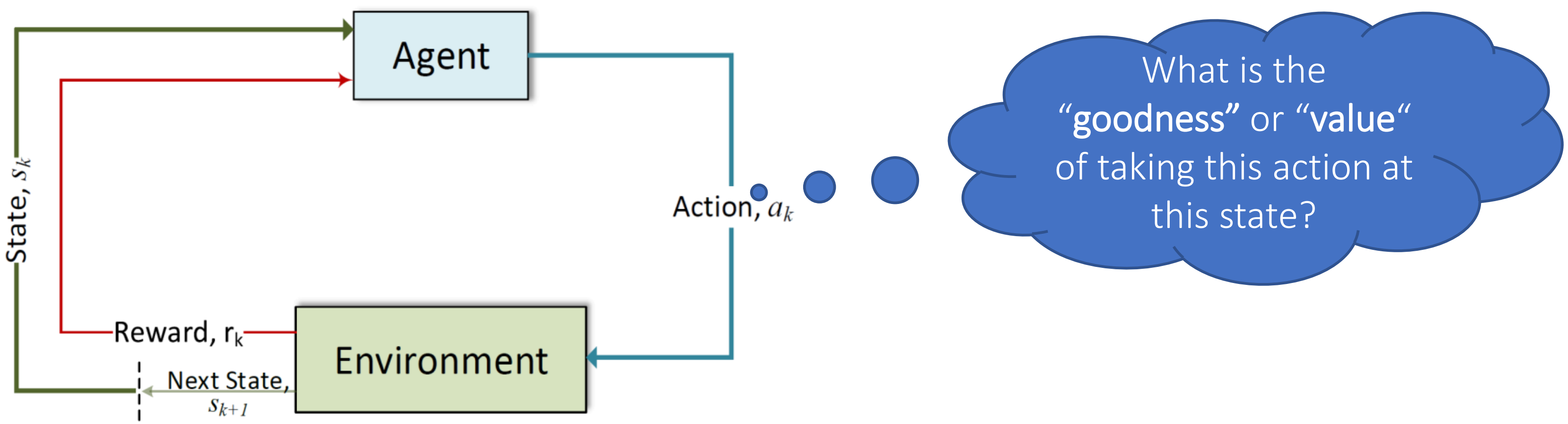
- Minimize battery *violations*
  - overflows (100% battery)
  - downtimes (0% battery)
- Maximize utility (duty cycle)
  - Sensor is always ON

- Solar EHWSN
- Duty cycle determines node energy consumption
- Hourly data
  - Tokyo: 1995-2018



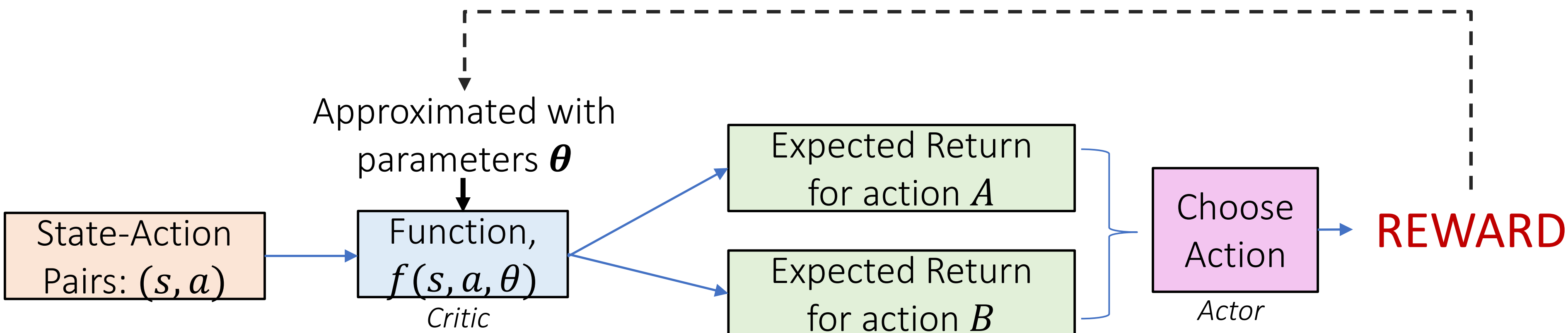
Reinforcement Learning (RL)

- Assuming a Markov Decision Process,
  1. **Perception:** What is the state of the agent?
  2. **Action:** Take an action according to a policy that maps states to actions
  3. **Reaction:** Receive a reward as feedback from the environment
  4. **Learning:** Learn from the reward to refine the policy.
- **OBJECTIVE**: Accumulate as much reward as possible



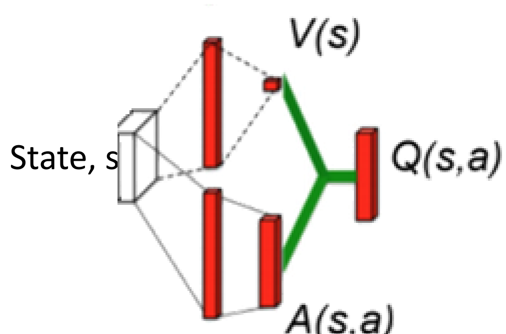
Deep Reinforcement Learning

- Use neural networks to predict the state-action value
- Learning via boot-strapping (better estimates from estimates)



Dueling Double Deep Q-Networks

Wang, Ziyu, et al. "Dueling network architectures for deep reinforcement learning." *arXiv preprint arXiv:1511.06581* (2015).

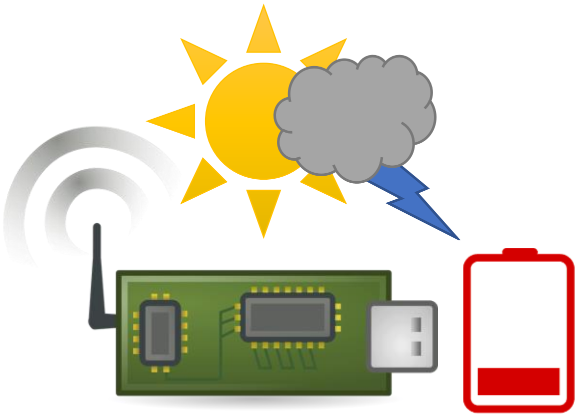


ϵ -greedy

- Always take greedy action
- With probability ϵ, take *random* action
- Start with high ϵ and decrease gradually (ϵ - annealing)

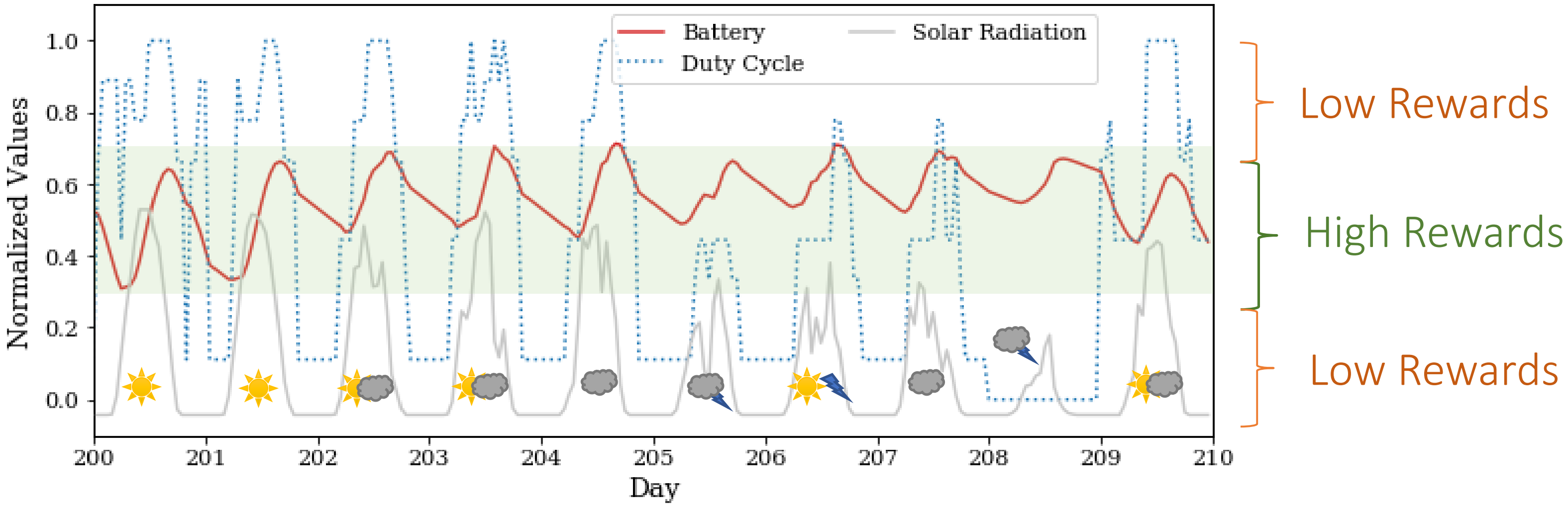
Single Agent ENO-RL: B-ENO

Basic-ENO: Naïve implementation of Deep RL for ENO-RL with a single agent.



State Space:
- Battery level
- Harvested Energy
- Energy Neutral Performance
- Weather Forecast

Action Space:
- Discrete Duty Cycle,
  - $D_{min} \leq d_t \leq D_{max}$



https://favpng.com/png_view/wireless-cliparts-wireless-sensor-network-internet-of-things-clip-art-png/JKvwgxaM

B-ENO: Performance

PERFORMANCE METRIC (Energy Neutral Operation)

ENO is achieved if there are less than 24 violations in 365 consecutive days

LEARNING TIME

(Time required to achieve ENO (1995-)

90,186 hours (~10 years)

UNREALISTIC SOLUTION

LEARNING PENALTY

(Violations committed during learning)

17,722 violations (~2 years)

OPERATION PENALTY

(Violations committed during greedy implementation from 1995-2018)

0 violations

Accelerating B-ENO

Can we decrease the learning *time* and *penalty* by leveraging

- the multiple nodes of the sensor network to
- simultaneously interact with the environment

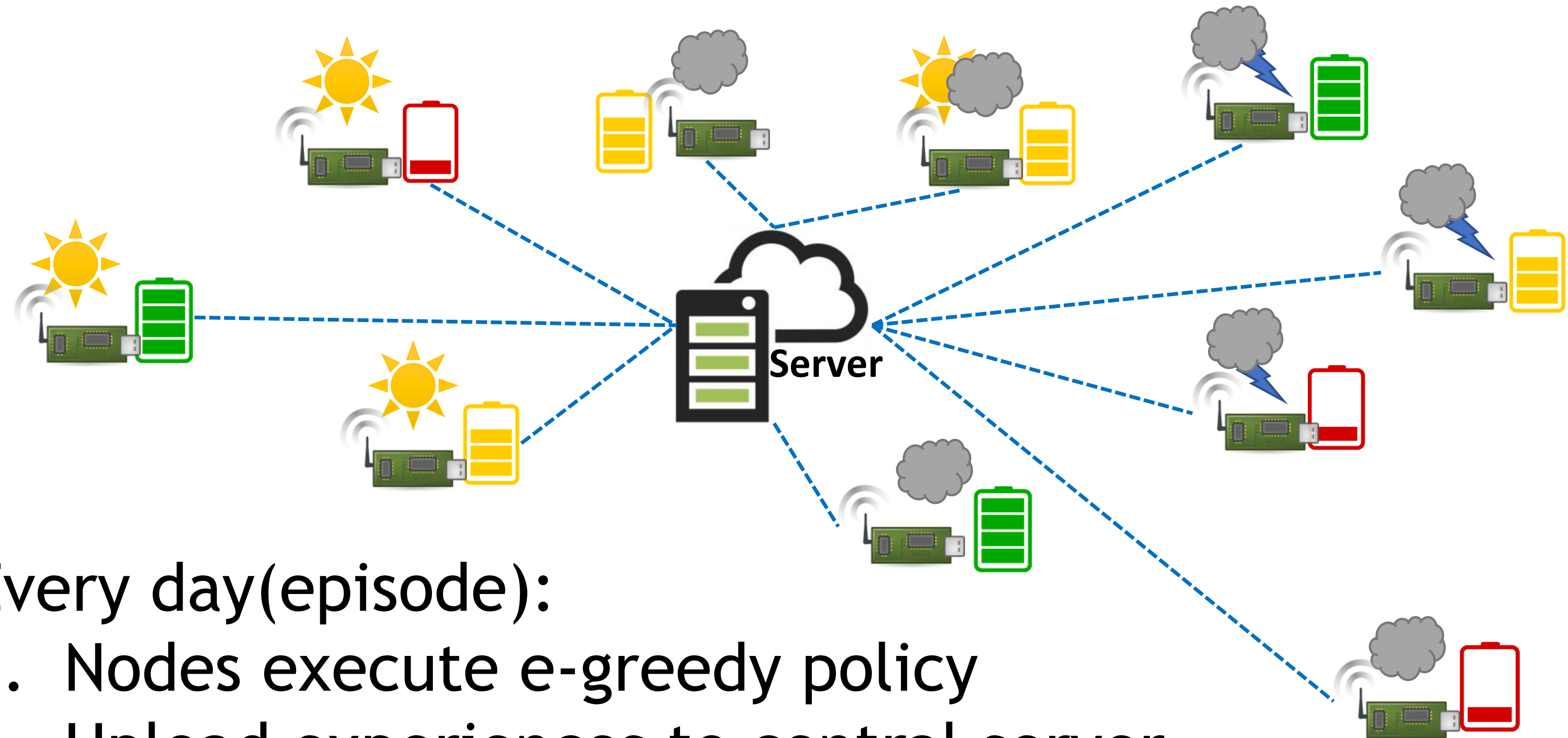
Distributed RL (DiRL)

- Leverage the collective experience to learn **better** and **faster**
- Explore **wider** and **faster**

```
experience => (present_state, action, reward, next_state)
```

Distributed RL (DiRL)

10 agents (nodes) and 1 central learner



Every day(episode):

1. Nodes execute e-greedy policy
2. Upload experiences to central server
3. Server executes N_l learning steps
4. Server broadcasts new policy to nodes

https://favpng.com/png_view/cloud-hosting-cliparts-web-server-cloud-computing-web-hosting-service-icon-png/BUiv4YG7

Challenges and Proposed Methods

1. Use Distributed RL (DiRL) to accelerate learning (**D**istributed-**ENO**)

   ☹ State space is not fully explored -> non-robust policies

2. **Partition** the state-space via coordinated exploration (**P**artitioned-**ENO**)

   ☹ Some agents face higher risks of violating ENO

3. Uniformly **distribute the risk** of exploration among nodes (**S**afe-**ENO**)

   ☹ Playing safe does not give best performance.

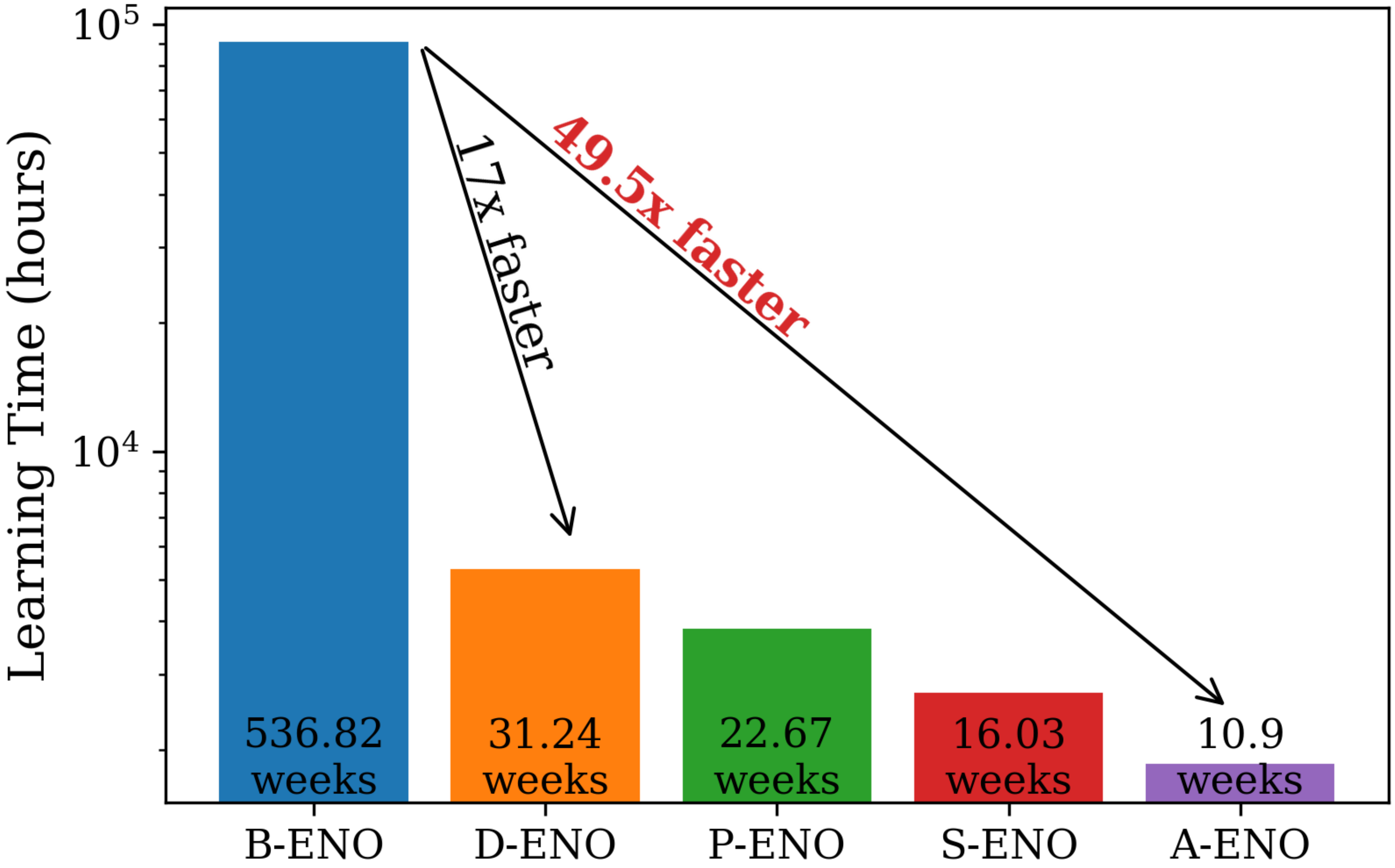
4. **Dynamically adapt** the exploration rate to tradeoff between learning time and learning cost (**A**daptive-**ENO**)

Comparative Analysis

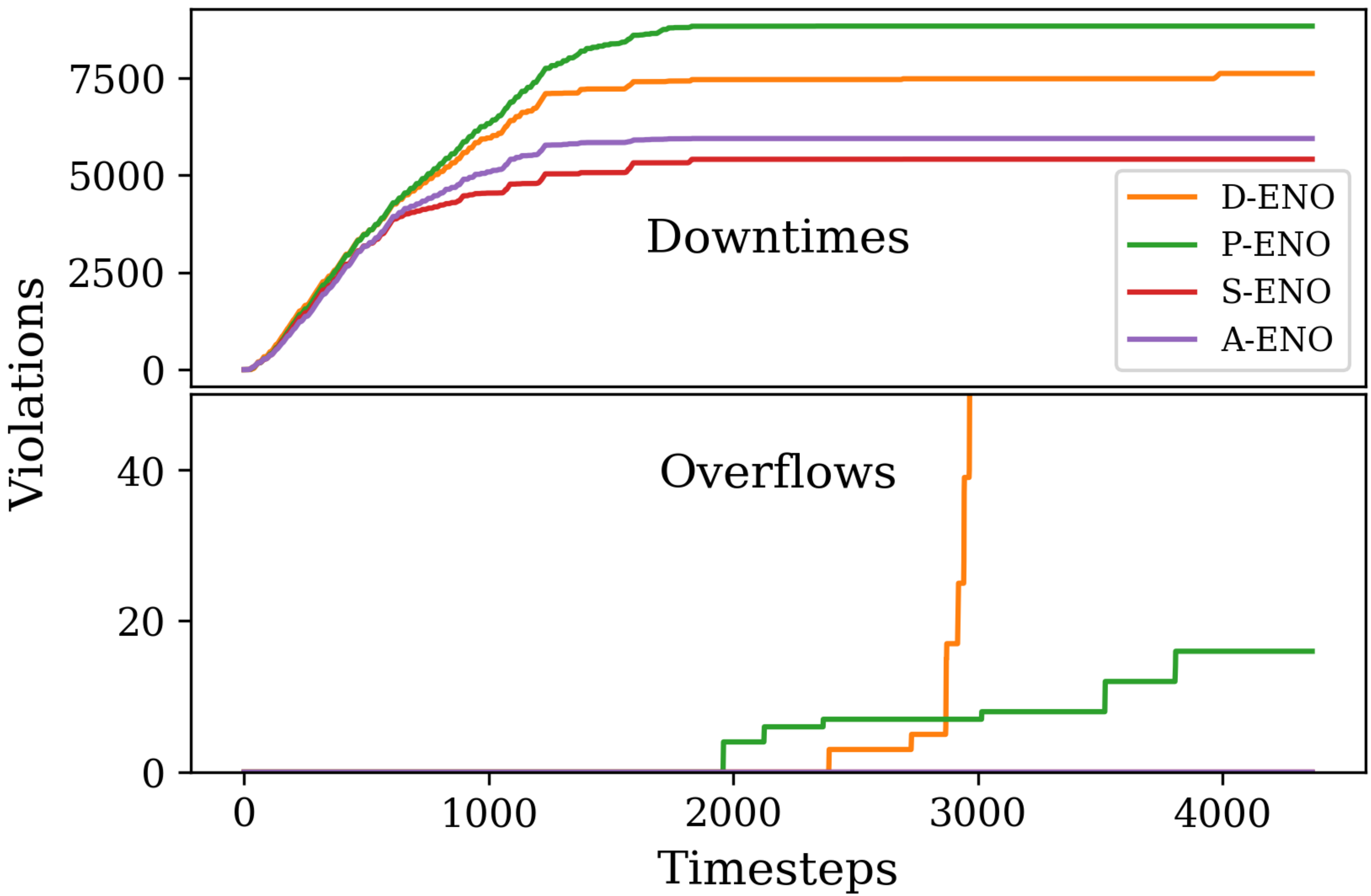
Algorithm	Learning Time (time to reach ENO)	Learning Penalty (# of violations)	Operation Penalty (# of violations)	Comments
B-ENO **(basic)**	10 yrs	17,722	0	Learning time and penalty too high
D-ENO **(naïve distributed)**	0.6 yrs	7,930	20	Insufficient state space exploration
P-ENO **(partitioned)**	0.4 yrs	8,817	16	State-space partitioning distributes risk non-uniformly
S-ENO **(safe)**	0.3 yrs	5,392	8	Tradeoff between learning time and penalty
A-ENO **(adaptive)**	0.2 yrs	5,921	0	Trading off learning costs dynamically

*Penalties are summed across all nodes for D,P,S,A-ENO

Comparison: Learning Time



Comparison: Learning Penalty

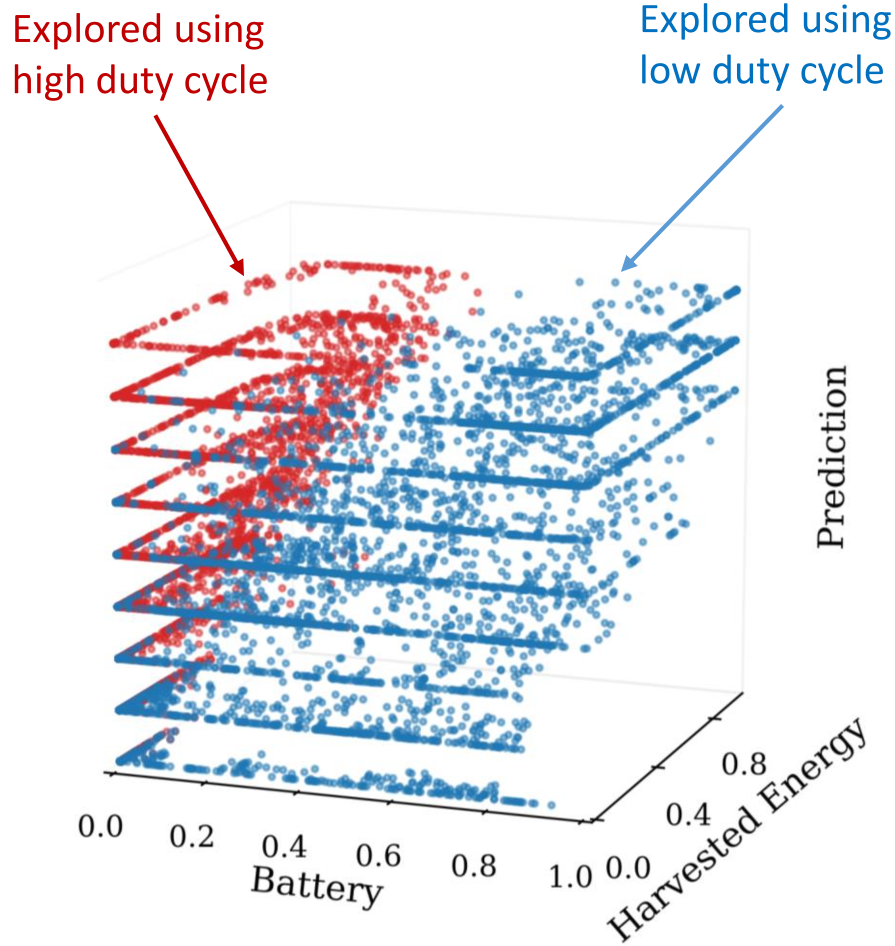


Partitioning the State Space

- High correlation between duty cycles (actions) and battery levels (states)

- Bias node duty cycles to explore different battery states

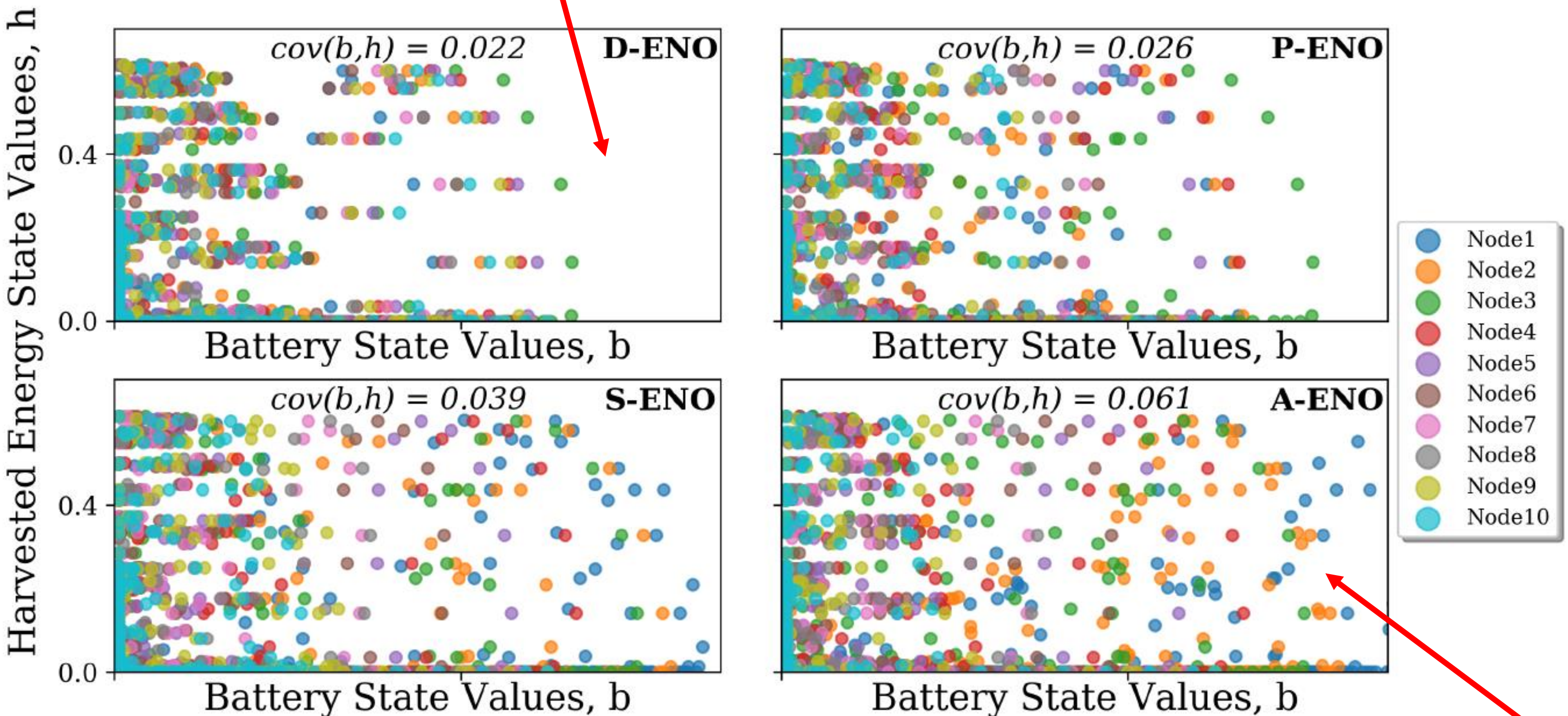
- Coordinate the nodes of DiRL to explore different regions of the vast problem state-space



Partition the state space using different exploratory actions.

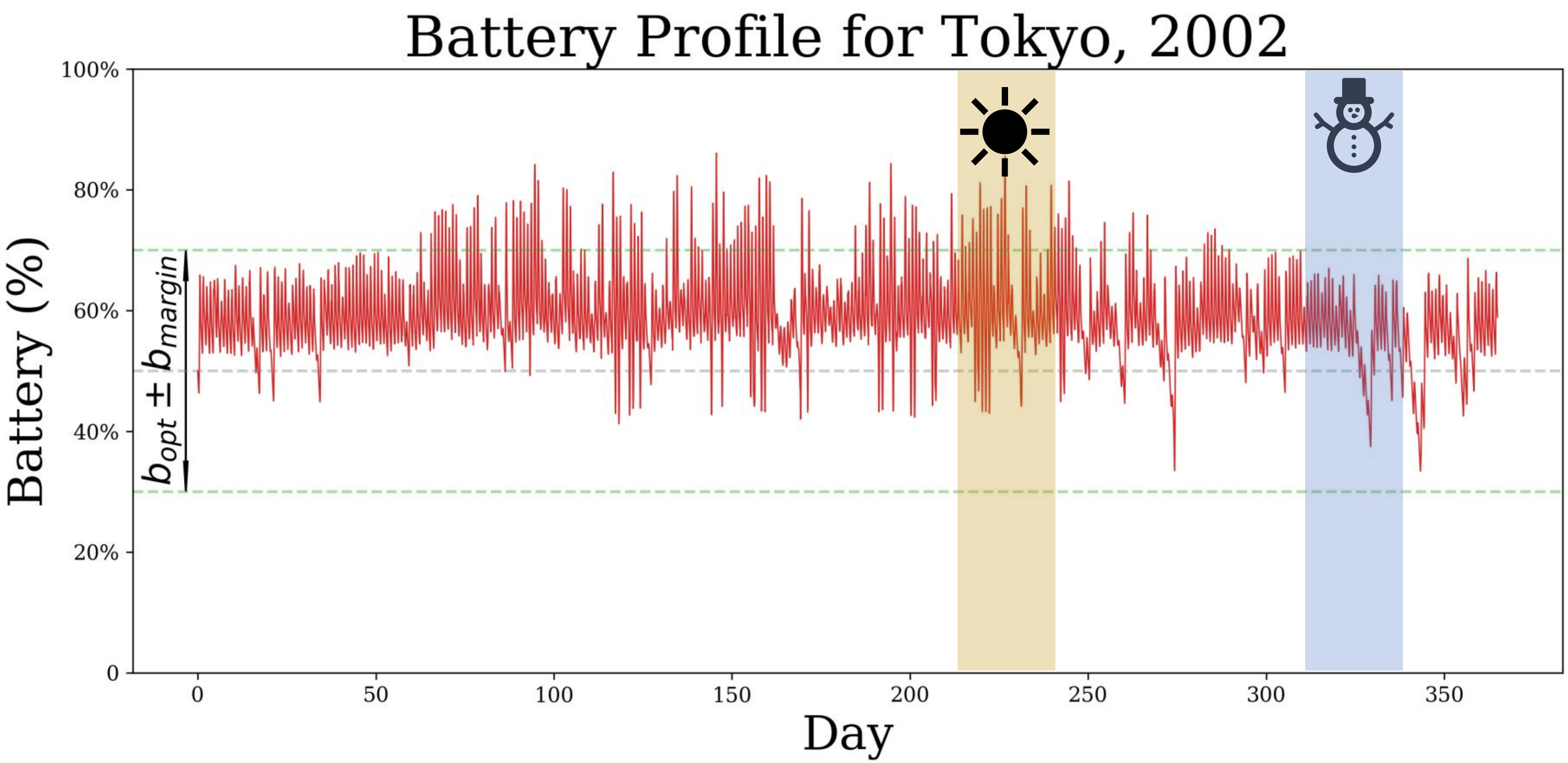
Comparison: Exploration

Less spread as a result of
naïve ϵ – greedy exploration.



More spread
in state space
resulting in
better learning

A-ENO: Year Run

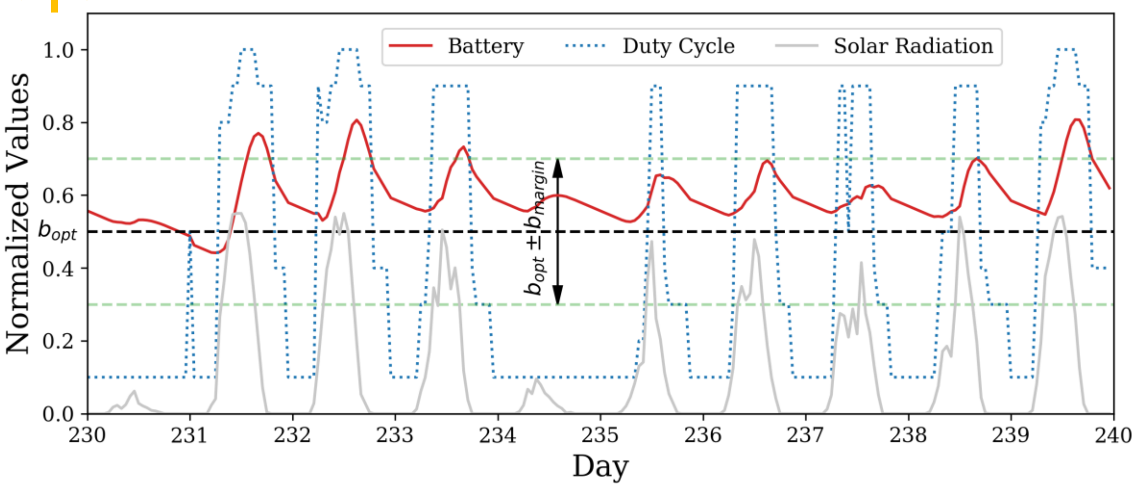


- ✓ Battery level is well within the required range.
- ✓ No violations despite seasonal and diurnal variations

A-ENO: Seasonal Adaptation

Summer

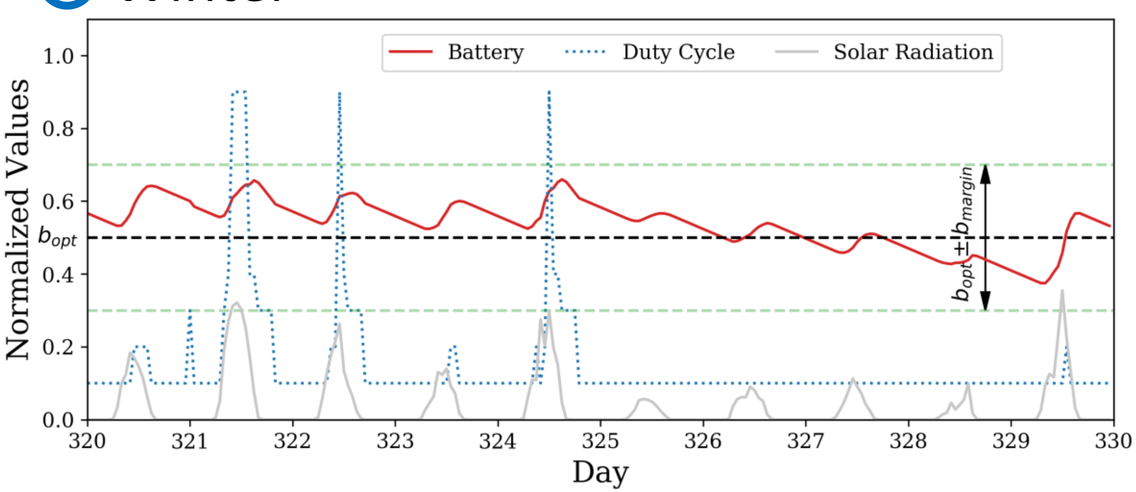
A-ENO: TOKYO,2002



- ✓ During summer, high duty cycles are used to maximize utility.
- ✓ Adaptation to sudden low energy days

Winter

A-ENO: TOKYO,2002



- ✓ During winter, lower duty cycles are used to save energy.
- ✓ Duty cycle is maximized when possible.

CONCLUSION

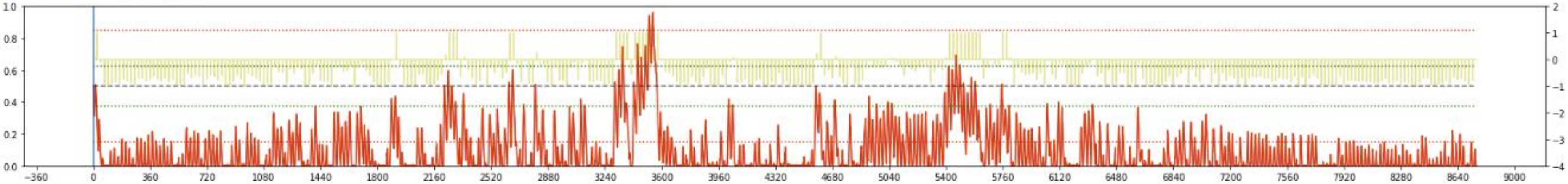
1. Non DiRL solutions are optimal but take impractically long to learn (B-ENO).
2. DiRL solutions learn faster but naïve implementations are sub-optimal (D-ENO).
3. Learning cost and time can be decreased by partitioning state space exploration (P-ENO).
4. Partitioning state space distributes risk non-uniformly which can be traded off for some performance loss (S-ENO).
5. Dynamically adjusting exploration rate trades off risk and performance and enhances learning (A-ENO).

Thank you

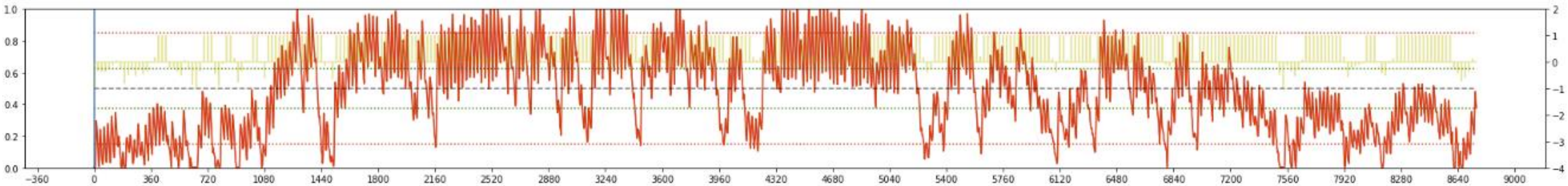
Any Questions or Comments

B-ENO: Learning

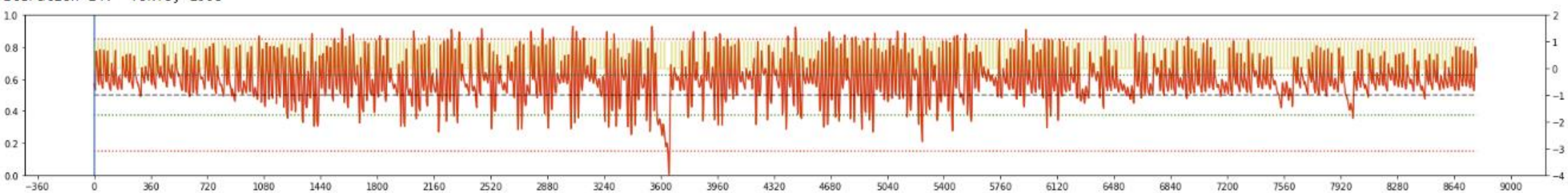
Iteration 0: TOKYO, 1995



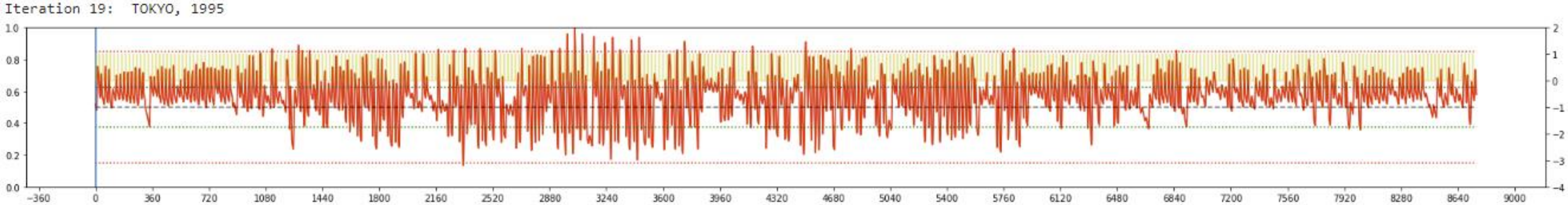
Iteration 7: TOKYO, 1995



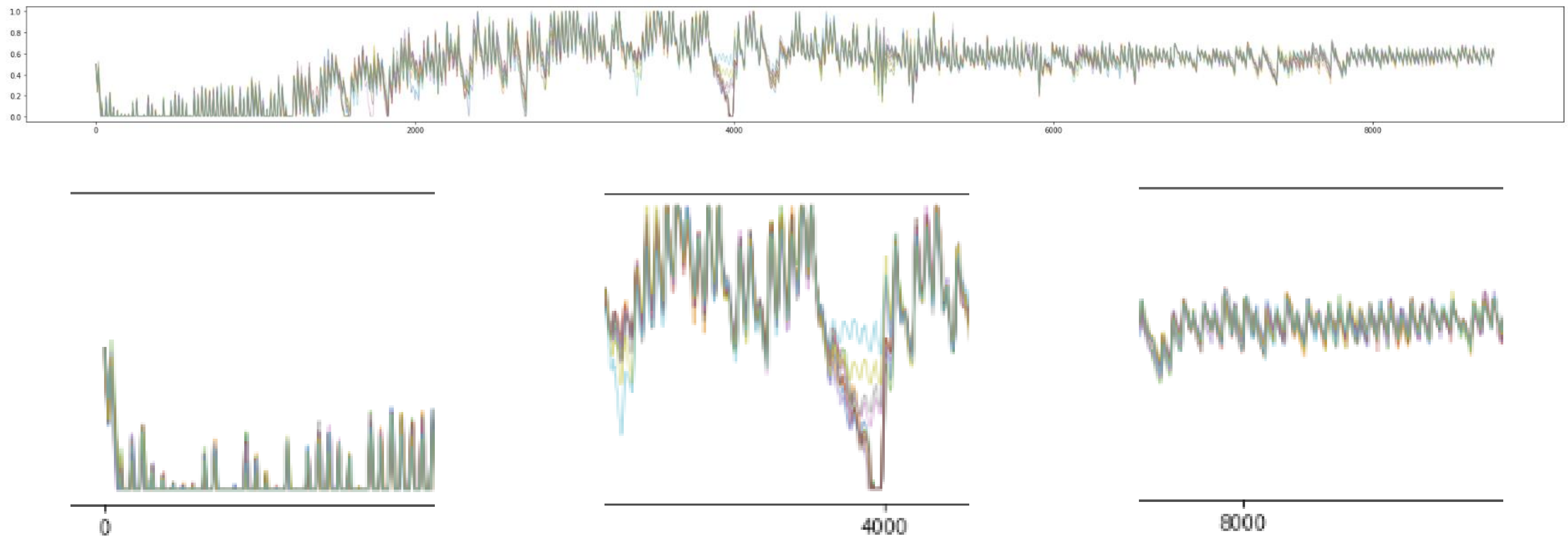
Iteration 14: TOKYO, 1995



Iteration 19: TOKYO, 1995



D-ENO: Learning



LEARNING TIME
5,249 hours (~0.6 years)

LEARNING COST (cumulative)
7,930 violations (~0.9 years)

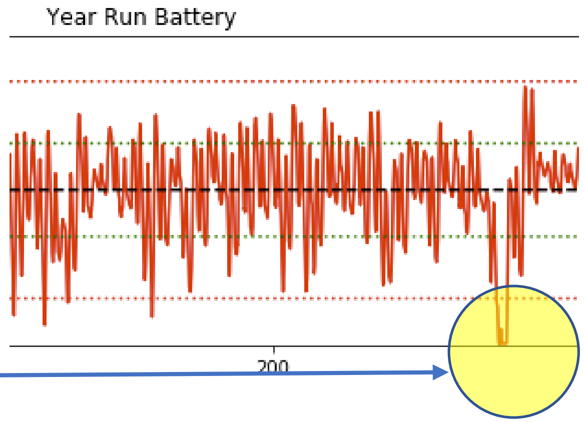
D-ENO: Testing

TOKYO

YEAR	AVG_RWD	VIOLATIONS DAY	VIOLATIONS BATT	EMPTY	FULL
1995	1.0	0	0	0	0
1996	1.0	0	0	0	0
1997	1.0	0	0	0	0
1998	1.0	0	0	0	0
1999	1.0	0	0	0	0
2000	1.0	0	0	0	0
2001	0.99	0	0	0	0
2002	1.0	0	0	0	0
2003	1.0	0	0	0	0
2004	0.98	2	20	20	0
2005	1.0	0	0	0	0
2006	0.99	0	0	0	0
2007	1.0	0	0	0	0
2008	1.0	0	0	0	0
2009	1.0	0	0	0	0
2010	1.0	0	0	0	0
2011	1.0	0	0	0	0
2012	1.0	0	0	0	0
2013	1.0	0	0	0	0
2014	0.99	0	0	0	0
2015	1.0	0	0	0	0
2016	0.99	0	0	0	0
2017	0.99	0	0	0	0
2018	0.99	0	0	0	0

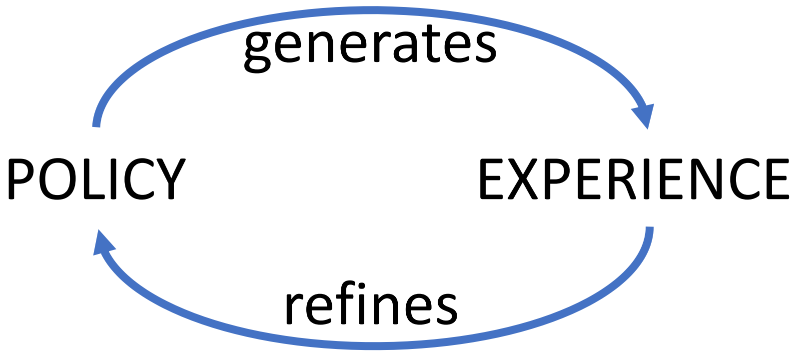
TOTAL Day Violations: 2.0
TOTAL Batt Violations: 20.0
TOTAL EMPTY Violations: 20.0
TOTAL FULL Violations: 0.0
****

TOKYO,2004



Challenges in Deep RL for ENO-RL

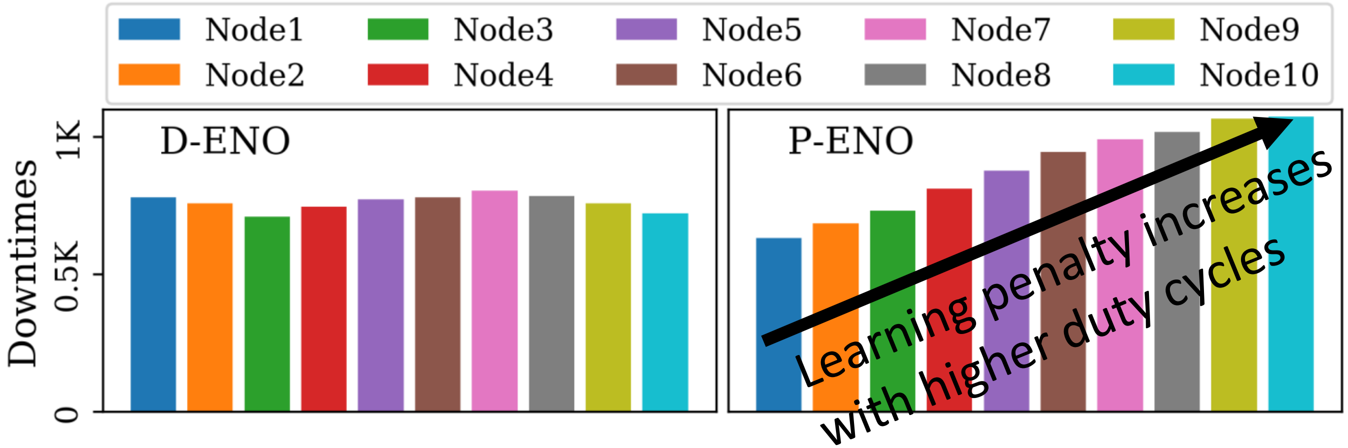
- Requires **LOTS** of training data
  - Longer training periods
  - Larger number of violations (downtimes and overflows)
- Unstable learning due to bootstrapping
  - Training should include a “correct” mix of positive and negative experiences.
  - Maximizing **EXPLORATION** of the state-space is critical
    - Unseen states may cause the network to destabilize
- Also, maximize utility
  - Exploration-exploitation tradeoff



GOALS:

- Converge to a robust policy
- Minimize learning time
- Maximize node utility (minimize violations during learning)

Safe Exploration: S-ENO



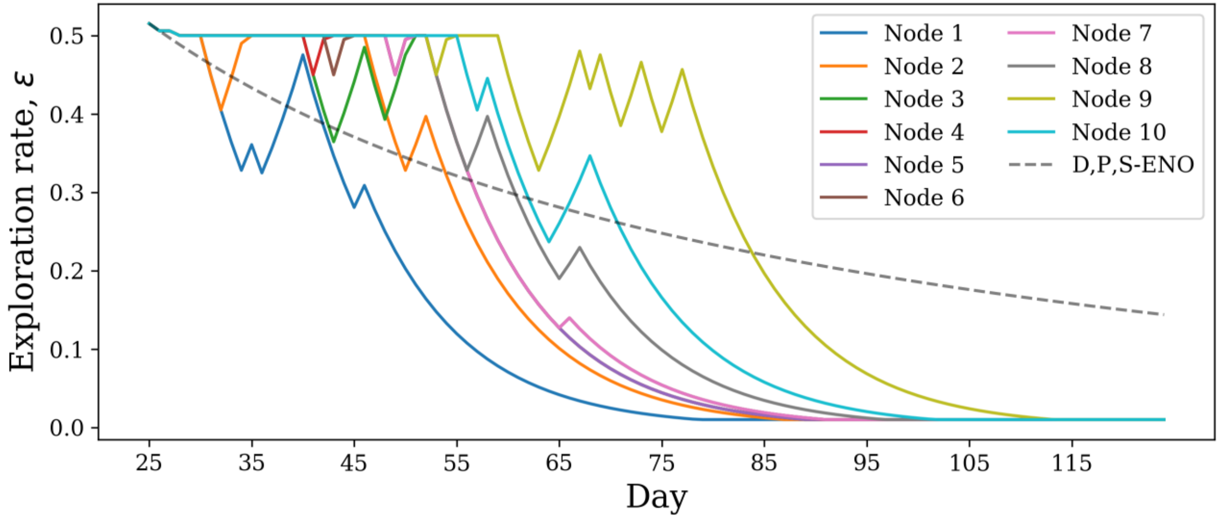
- High-duty cycle as non-greedy action -> more violations
- Low-duty cycle as non-greedy action -> less violations
- Change the preferred non-greedy action after every episode.

	Day 1	Day 2	Day 3	Day 4	Day 5	…
Node 1	d_1	d_2	d_3	d_4	d_5	…
Node 2	d_2	d_3	d_4	d_5	d_6	…
Node 3	d_3	d_4	d_5	d_6	d_7	…
Node 4	d_4	d_5	d_6	d_7	d_8	…
…	…	…	…	…	…	…

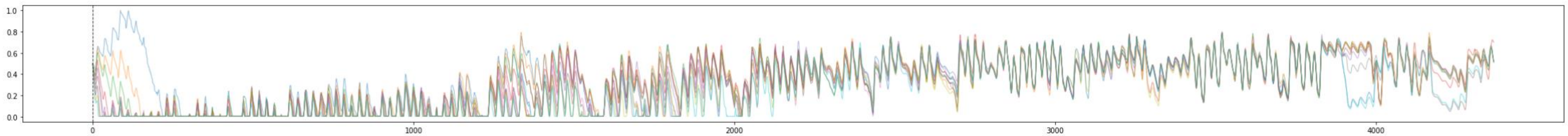
Adaptive exploration: A-ENO

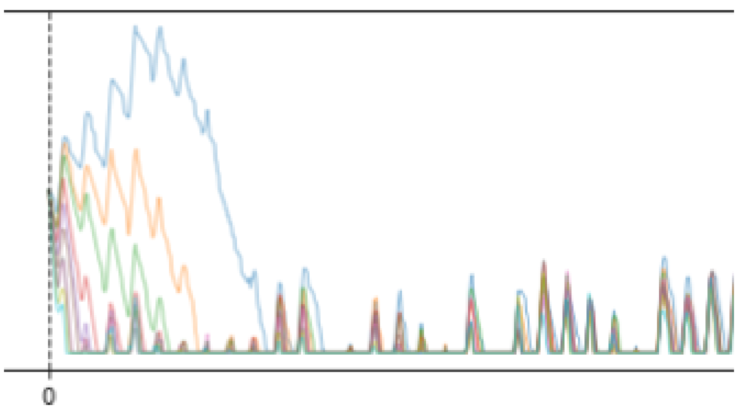
- Different nodes -> different environments
- Different environments -> different learning behavior
- Different learning behavior -> different annealing rates for ϵ.

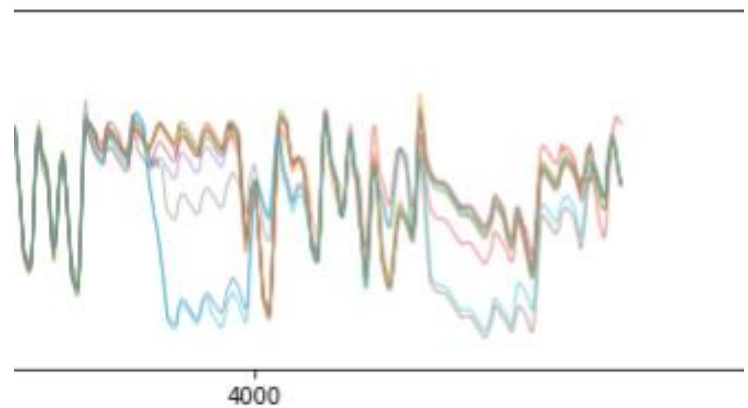
- Increase ϵ if reward is negative.
- Decrease ϵ if reward is positive.



Adaptive exploration: A-ENO







More diverse experiences in the beginning

Robust performance for anomalous states