Model-Based Reinforcement Learning and Its Application

Mingcong Cao

Preliminary Examination

15 Aug 2025

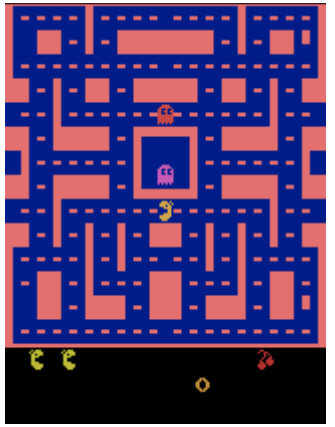
Committee Members: Grigorios Chrysos, Josiah Hanna, Umit Y. Ogras, Suat Gumussoy

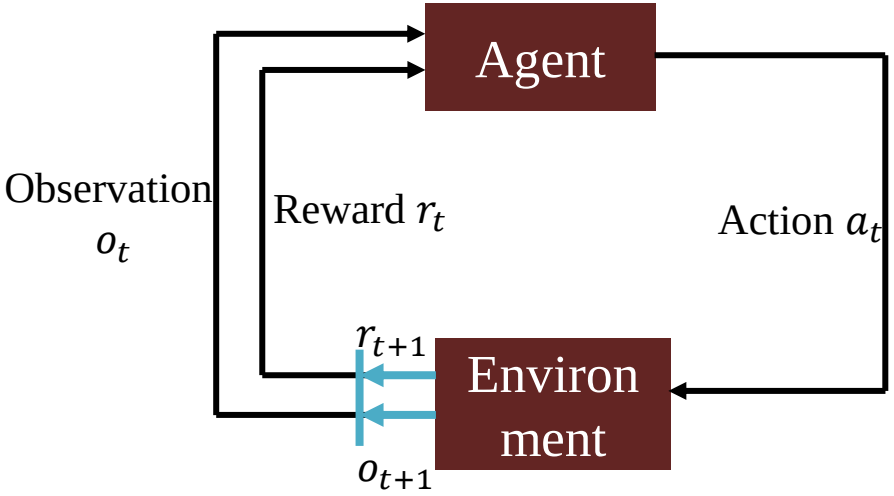
Agenda

- **Motivation**
- **Preliminary Work-1:**
  - Pact: Accurate power analysis and carbon emission tracking for sustainability
- **Preliminary Work-2:**
  - Mamba World Model for RL: Enhancing Long-term Memory in Model-based Reinforcement Learning
- **Ongoing and Proposed Work:**
  - State-Space Model-Based Reinforcement Learning for Dynamic Spectrum Access
  - Reinforcement Learning for Chip Design
- **Timeline**
- **Conclusions**

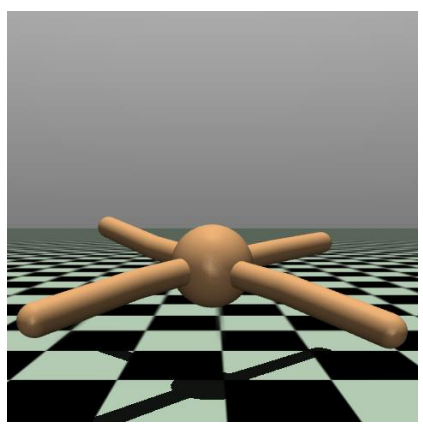
Motivation: Successes of Reinforcement Learning

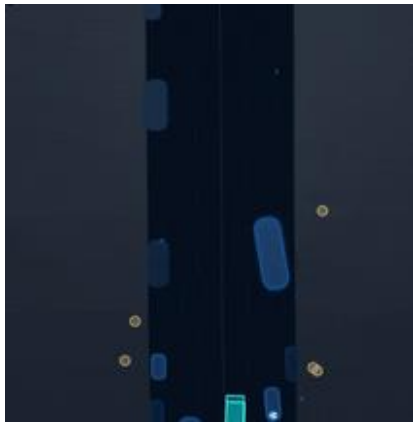
- **RL has demonstrated great success in multiple domains, for example:**
  - **Game:** Achieved superhuman performance in complex games
  - **Robotics:** Policies trained in simulation transfer successfully to real robots
  - **Driving:** Learns complex driving behaviors from interactions
  - **LLM Training:** Improve LLM reasoning through RL

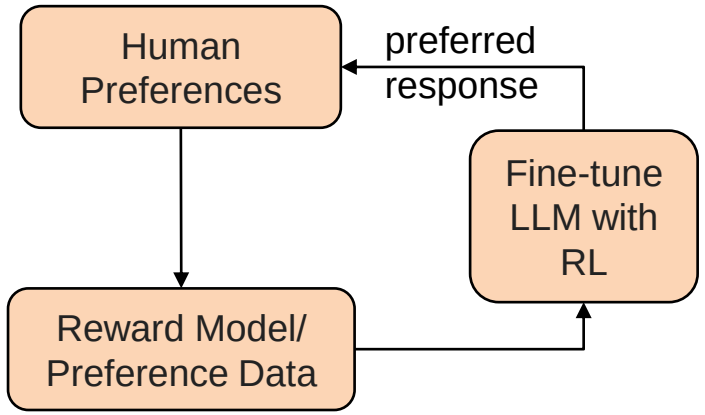




The RL perception-action-learning loop







Img Sources:
[1] Gymnasium Documentation. Accessed at: https://gymnasium.farama.org/environments/mujoco/
[2] Imitation Is Not Enough: Robustifying Imitation with Reinforcement Learning for Challenging Driving Scenarios. https://waymo.com/research/imitation-is-not-enough-robustifying-imitation-with-reinforcement-learning/

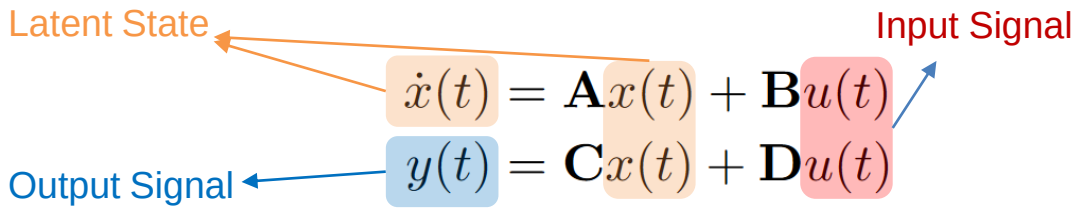
Limitation of RL in Practice - 1

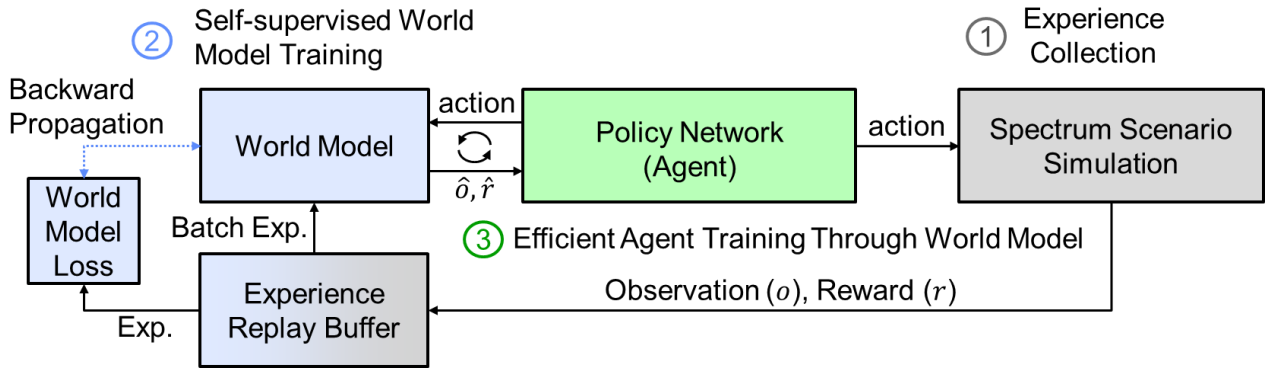
- **Sample efficiency**
  - RL agents often require millions of environment interactions to achieve high performance
  - Data collection is **expensive**, **time-consuming**, and sometimes **unsafe** (e.g., communication system simulation, robotics, autonomous driving)
  - Training on a large dataset or experience replay is not energy efficient
- **Solution: Model-based RL**
  - Learns a predictive model of the environment's dynamics from limited data
  - Training RL through imagined

Apply practical sample-efficient RL Algorithms to achieve high performance and energy efficiency

Limitation of RL in Practice – 2

- **Partially observable**
  - The Spectrum usage cannot be fully observed since the receive channels are limited
- **Complexity of action space**
  - High-dimensional with constraints
- **Expensive simulation**
  - The simulation is more than 100 times slower than real-world time.
- **Solution we explore:**
  - SSM agent with long-term memory
  - World model
  - Multiple action heads





Agenda

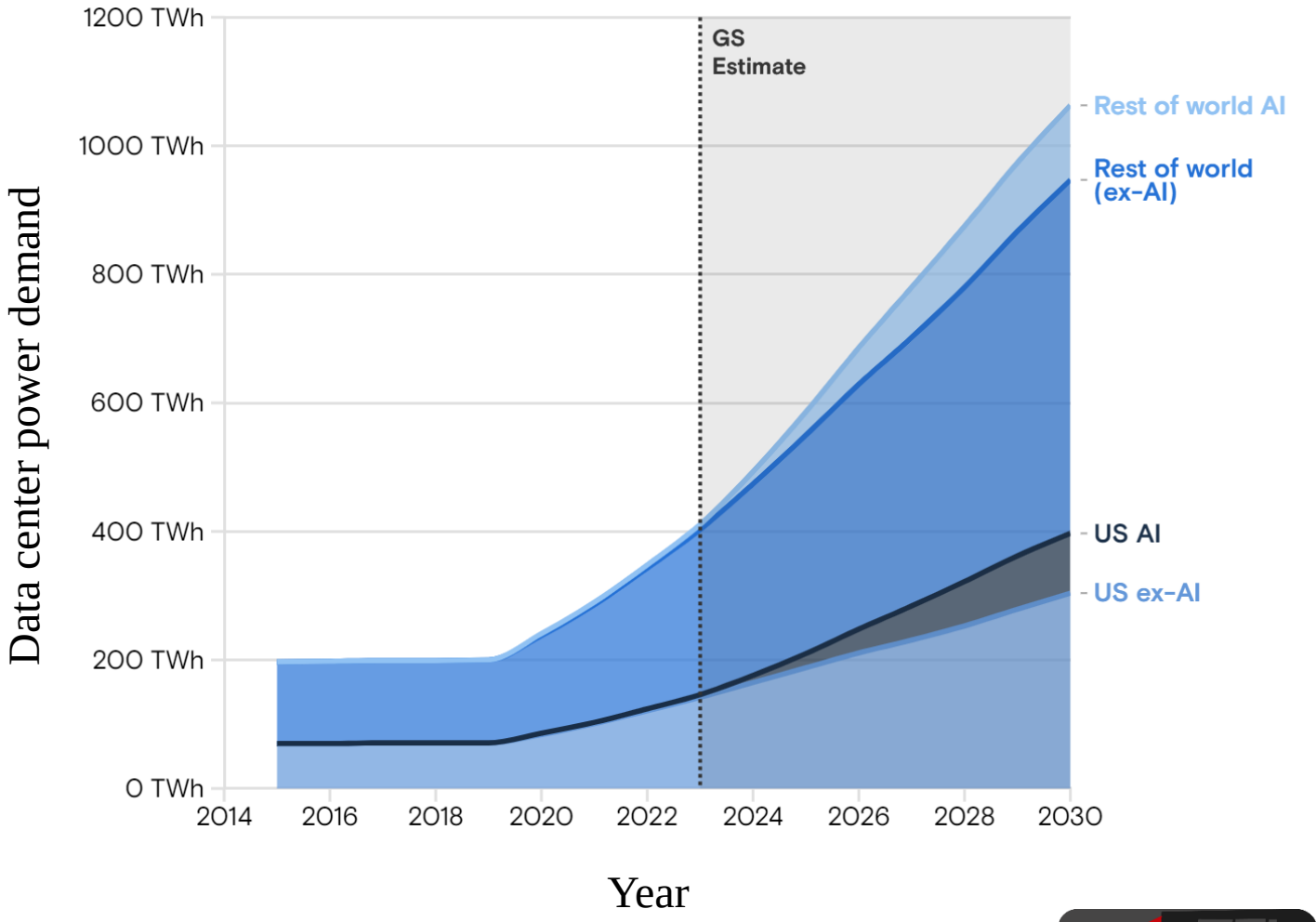
- Motivation
- **Preliminary Work-1:**
  - Pact: Accurate power analysis and carbon emission tracking for sustainability
    Aditya Ukarande[1], Toygun Basaklar[2], **Mingcong Cao**, Umit Y. Ogras
    [1]Joined nVIDIA in Summer 2024, [2]Joining Apple in Fall 2024
- **Preliminary Work-2:**
  - Mamba World Model for RL: Enhancing Long-term Memory in Model-based Reinforcement Learning
- **Ongoing and Proposed Work:**
  - State-Space Model-Based Reinforcement Learning for Dynamic Spectrum Access
  - Reinforcement Learning for Chip Design
- **Timeline**
- **Conclusions**

Overview

- Datacenter Power Demand and AI Power Consumption
- Carbon Footprint Measurement and Prediction Tools
- **PACT: Accurate Power Analysis and Carbon Emission Tracking**
  - Power Measurement
  - Dataset Generation
  - Power, Energy, and CO_2 Emission Modeling
- Experimental Evaluation
- Conclusions

Rising Data Center Power Demand

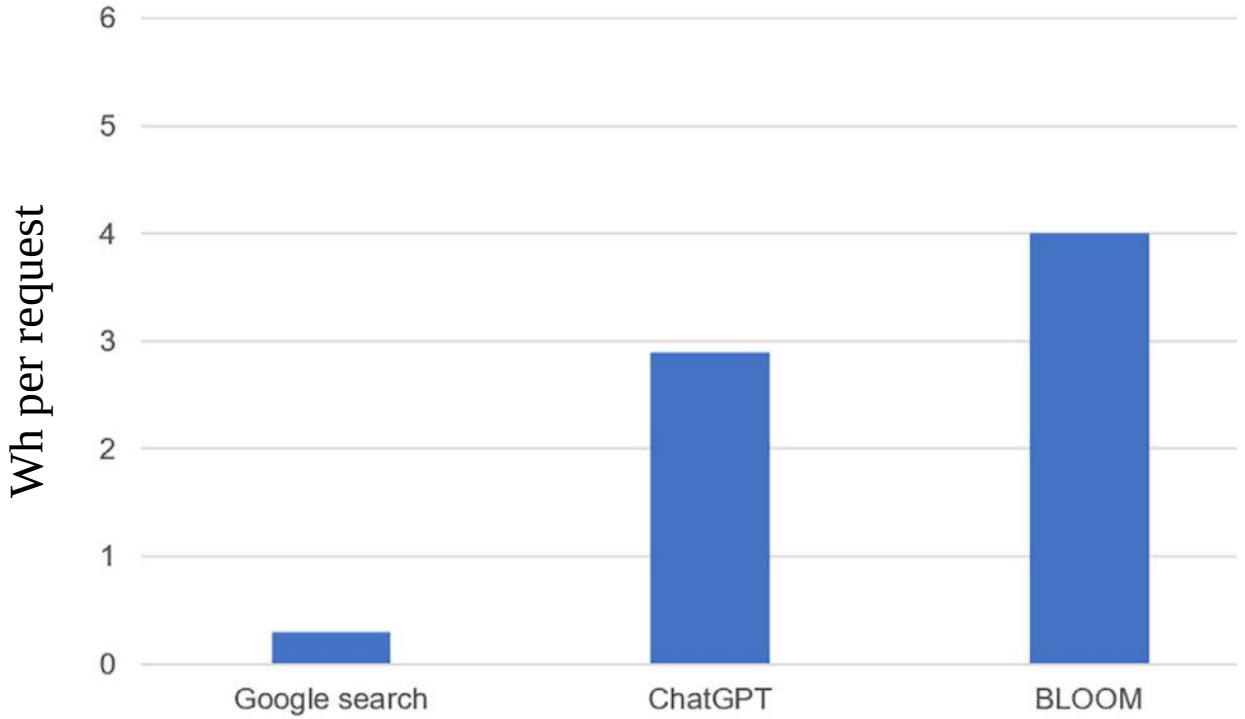
- **Trend**
  - Data center power demand expected to grow by **160%** in 2030 vs. 2023 levels
  - AI expected to represent about **19%** of data center power demand in 2028
- **Causes**
  - Rising data demand
  - Slowing down power efficiency gains



Source: Goldman Sachs Research, "GS SUSTAIN: Generational Growth: AI, data centers and the coming US power demand surge." 2024
Generational Growth: AI, data centers and the coming US power demand surge | Goldman Sachs

Carbon Footprint of AI Workloads

- GPT-3, OpenAI's 175 billion parameter model, reportedly used 1,287 MWh to train, while DeepMind's 280 billion parameter model used 1,066 MWh. This is about 100 times the energy used by the average US household in a year.
- **The inference energy cost of ChatGPT is about 10x higher than a Google search.**
- **Assuming full AI adoption, Google's AI alone could consume as much electricity as a country such as Ireland (29.3 TWh per year).**



Estimated energy consumption per request

de Vries, The growing energy footprint of artificial intelligence, Joule (2023), https://doi.org/10.1016/ j.joule.2023.09.004

Need for Energy & Carbon Footprint Prediction

- **Increasing energy and the resulting carbon footprint concerns**
- **Strong motivation for tools that can**
  - Evaluate the energy consumption
  - Predict carbon usage
  - Co-optimize energy and carbon footprint with accuracy

What is the cost of getting 0.1% higher accuracy?

- There are two main classes of approaches
  - **Measurement-based approaches:** Require running the workload & collecting data
  - **Prediction approaches:** Predict energy and carbon footprint based on model and hardware parameters

Existing Carbon Footprint Measurement Tools - 1

- There are two commonly used energy and carbon footprint prediction tools
  - **CodeCarbon** [1]
  - **CarbonTracker** [2]
- Most accurate prior work, but they did not accurately model power consumption by components other than CPU and GPU
  - **CodeCarbon** makes a conservative estimate (as a constant value) for everything besides CPU and GPU
  - **CarbonTracker** ignores the rest of the components

Average Error on Different Machine Learning Workloads

	CarbonTracker	CodeCarbon	PACT (ours)
Energy Error	22.8 – 37.9%	20.2 – 24.0%	0.56 – 0.57%
Carbon Error	34.6 – 38.2%	19.6 – 29.3%	0.56 – 0.57%

[1] CodeCarbon. https://github.com/mlco2/codecarbon, 2020.
[2] L. F. W. Anthony, B. Kanding, and R. Selvan. Carbontracker: Tracking andpredicting the carbon footprint of training deep learning models. ICML Workshopon Challenges in Deploying and monitoring Machine Learning Systems, July2020. arXiv:2007.03051.

PACT Methodology

Proposed PACT methodology consists of three major steps

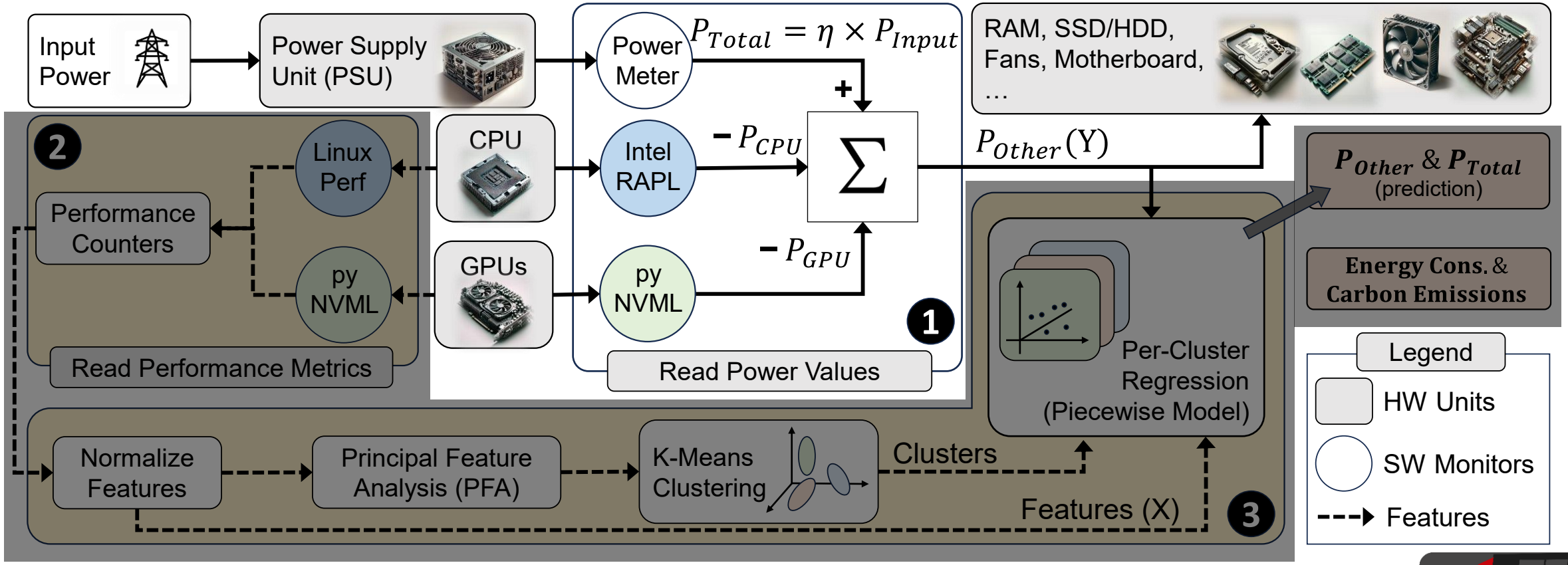
1. **Accurate power measurements using Power Supply Unit (PSU)**
2. **Dataset generation by running different classes of AI workloads**
   - Computer vision (CV)
   - Natural language processing (NLP)
   - Reinforcement learning (RL)

   And Stress tests
3. **Average power consumption/energy modeling and carbon footprint prediction**

Picture of server

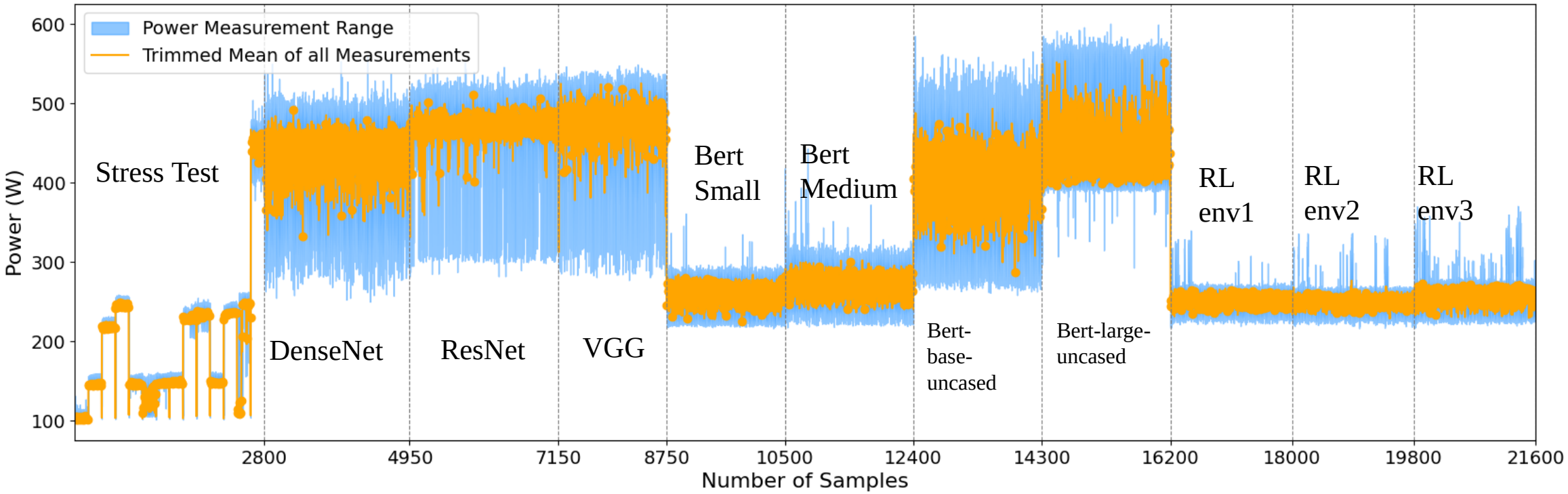
1. Power Measurement

- **Total power $(P_{total})$is measured using a special power supply unit**
- **CPU and GPU power measurement from the manufacturer's tool (Intel RAPPL, nVIDIA SMI)**
- **Power consumption of remaining components is found as: $P_{other} = P_{total} - P_{GPU} - P_{CPU}$**



1. Power Measurement (cont'd)

- **Power measurements are noisy**
- **Each run is repeated five times, and the average values are found**



Total Power Measurement and Trimmed Mean of Five Runs

2. Dataset Generation

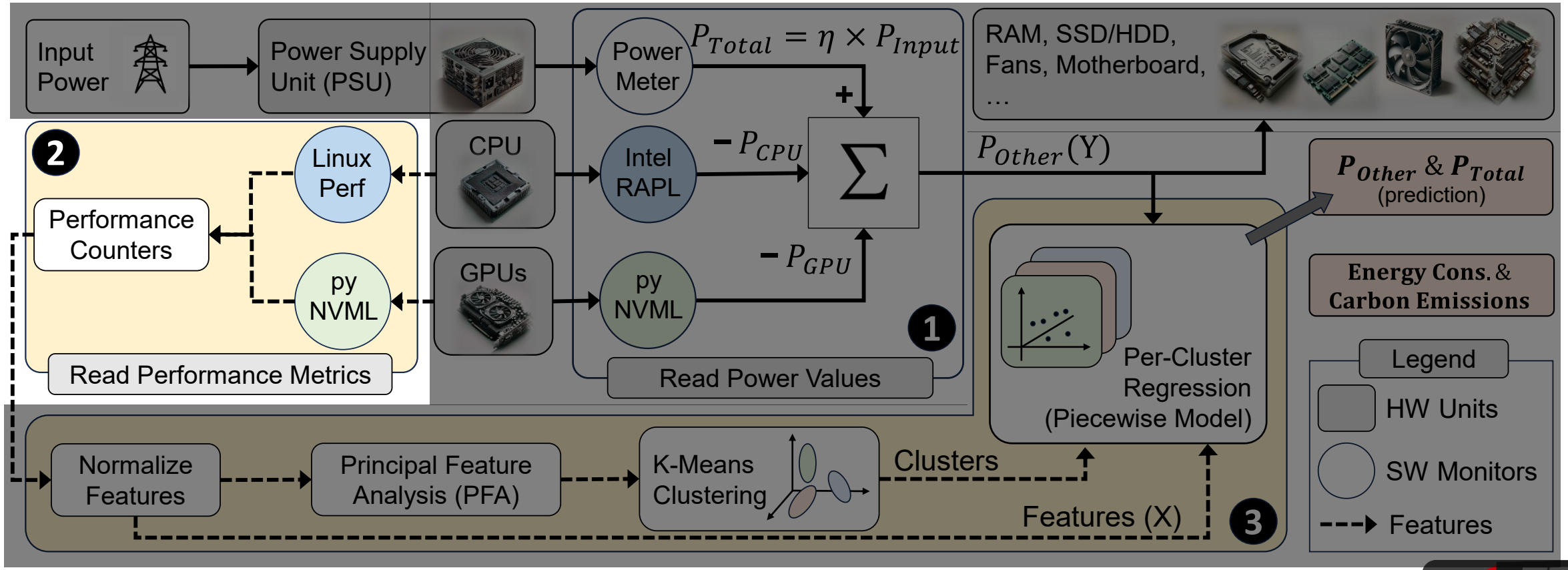
- **Power consumption is a function of architectural events**
  – which in turn are functions of the workload and hardware
- **We analyzed performance counters using *perf* and CPU/GPU utilizations**
- **We found that the following events have the highest correlations with power consumption**

Linux perf	cache-misses, page-faults, CPU-cycles, CPU-migrations
pyNVML	GPU-SM-utilization (%), GPU-memory-utilization (%)

• Dataset is generated by running both stress tests and ML benchmarks while collecting the performance counters and power measurements

2. Dataset Generation (cont')

- **A piecewise linear regression model is fitted on the dataset**
- **This model is further used to estimate energy consumption and carbon emissions**



2. Dataset Generation – Stress Tests

- **Capture idle system behavior for 200 seconds to establish a baseline**
- **Then, stress the CPU, I/O, RAM, and SSDs/HDDs for 200 seconds by initiating 1, 25, and 50 threads through the Linux stress [1] tool**
  - Captures workload behavior encountered in real-life scenarios, primarily
    - Arithmetic operations,
    - File I/Os,
    - Memory allocations and deallocations, and
    - disk read/writes
- **Use a GPU stress-testing tool [1] to perform operations on the GPU**
  - These tests comprehensively simulate a mix of operations, ensuring our evaluation accurately reflects practical usage scenarios

[1] PACT Repository. https://anonymous.4open.science/r/TCAP-4DF4.

2. Dataset Generation – AI Workloads

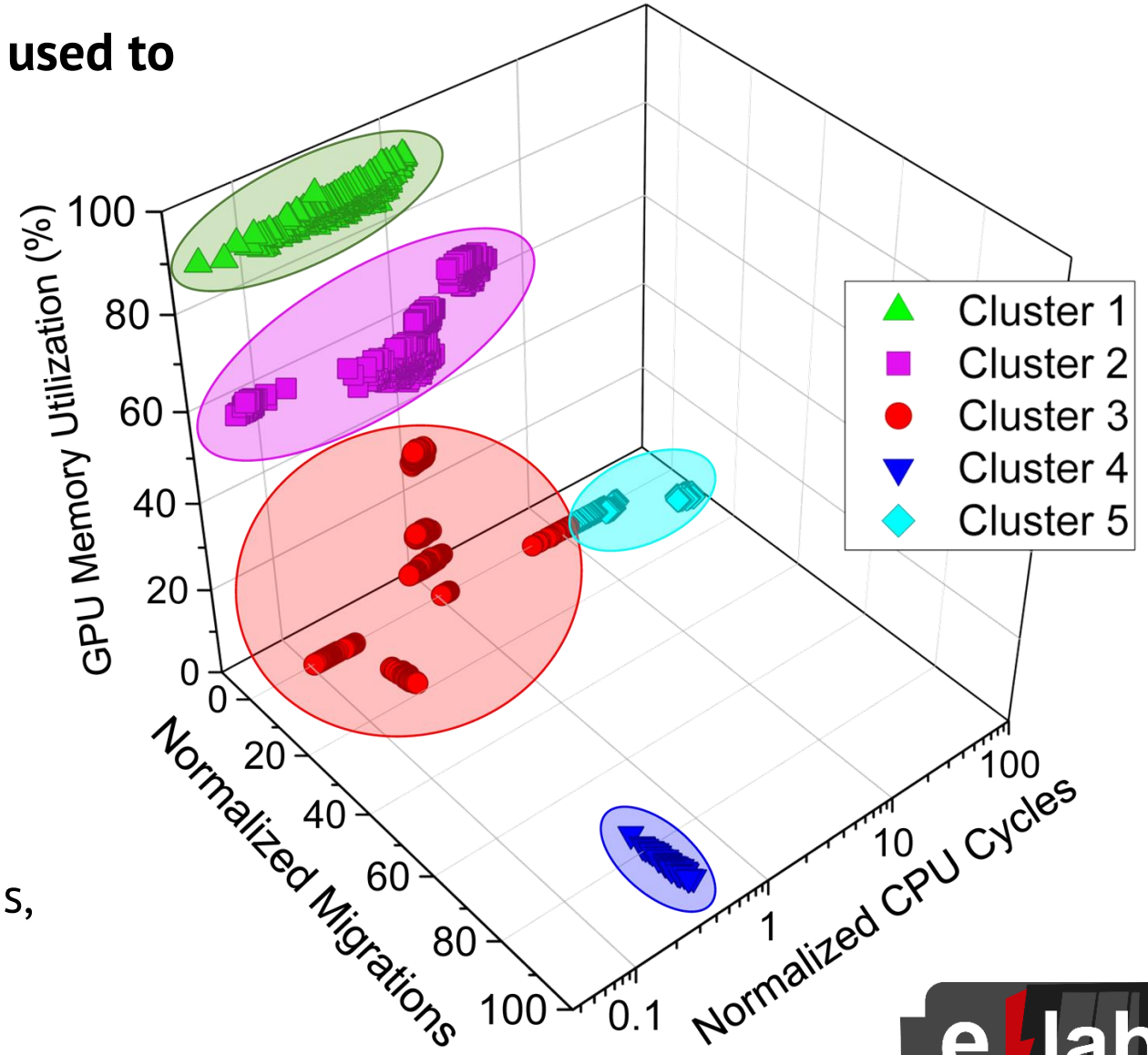
- **State-of-the-art models from key domains like CV, NLP, and RL**
- **Computer Vision domain**
  - DenseNet, VGG, and ResNet on the CIFAR10 classification task
- **Natural Language Processing**
  - BERT family of models, including bert-small, bert-medium, bert-base-uncased, and bert-large-uncased, using the SST-2 text classification datasets
- **Reinforcement Learning**
  - Soft-actor-critic algorithm, evaluating across various environments such as cheetah-run, finger-turn-hard, and humanoid-run

3. Power Consumption/Energy Modeling - Preprocessing

- **Our objective is to predict average power values based on the performance counters, while PSU outputs instantaneous power consumption**
- **Additionally, Linux perf-based performance counters are collected at a 100× larger sampling frequency than power measurements**
- **These disparities are addressed by a step-by-step preprocessing approach**
- **Moving Average Filter:**
  – 20-second window size set empirically aligned with CPU and GPU average power tracking tools
- **Normalization of Performance Counters:**
  – averaged over 100 samples and then standardized them by scaling between 0 and 100
- **Trimmed Mean:**
  – perform 5 runs and average after removing the highest and lowest measurements

3. Power Consumption/Energy Modeling - Clustering

- **Principal feature analysis and K-Means are used to find 5 clusters using 3 selected features**
- **Five distinct clusters were observed across all experiments**
  - **Cluster 1:** High GPU, low-moderate CPU utilization. Rare migrations
  - **Cluster 2:** Moderate-high GPU utilization, moderate CPU utilization and migrations
  - **Cluster 3:** Low-moderate CPU/GPU utilization and migrations
  - **Cluster 4:** Low GPU utilization, moderate CPU utilization, significant migrations
  - **Cluster 5:** Low GPU utilization and migrations, high CPU utilization



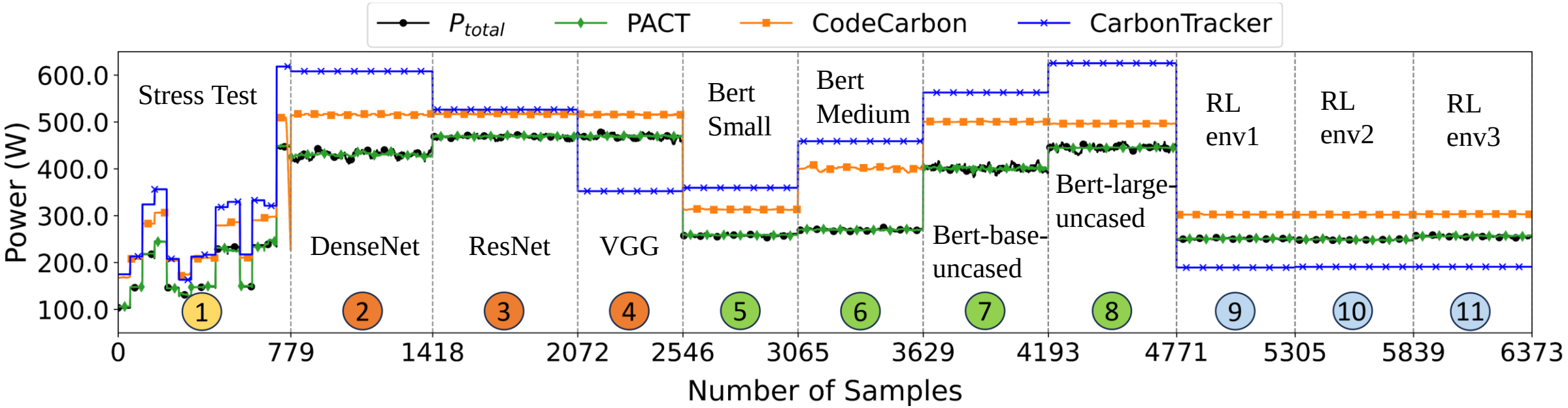
3. Power Consumption/Energy Modeling (cont')

- **Average power consumption: An independent linear regression model is trained to fit P_{other} for each cluster using all the features collected**
- **Energy consumption:**
  - P_{other} is added to the CPU and GPU power consumption to find the total power
  - Next, we divide P_{total} by the PSU's efficiency factor (η) to calculate the average power drawn from the socket
  - This value is then multiplied by the measurement duration to calculate the total energy
- **Carbon emission:**
  - This total energy value is multiplied by the most recent carbon intensity value, CI (kg/kWh), specific to the region [1] to obtain the carbon emission

[1] Electricity Maps. https://app.electricitymaps.com.

Power Consumption at a Function of Time & Workload

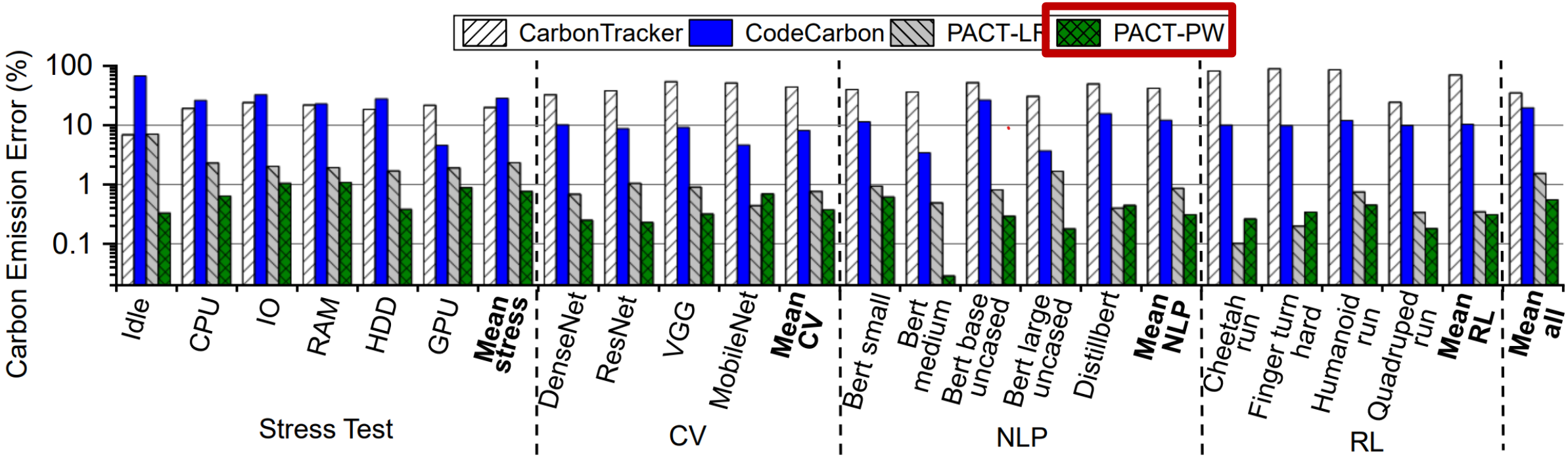
- **PACT tracks the measured power accurately for all workloads during all phases**



- CodeCarbon and CarbonTracker over- or under-estimate at different phases since they assume fixed overheads

Carbon Emission Estimates

- **Low Error:** PACT consistently performs well across **all ML workloads**, with energy and carbon emission errors **less than 1%**
- **High Adaptivity:** PACT consistently performs excellently on 2 servers we tested, out-performing state-of-the-art tools



Conclusions

- **Sustainability:** Existing literature often **overlooks the sustainability** of ML algorithms.
- **System-wide:** PACT is a comprehensive **system-wide power analysis and carbon emission tracking** methodology considering all system components.
- **Accuracy:** PACT demonstrates **superior accuracy** across various workloads
  - less than 1% average error in energy consumption and carbon emission estimation
  - In comparison, existing tools have 20.2% to 38.3% error
- **Adaptivity :**This work focused on AI/ML models, but the proposed methodology also **applies to other types of workloads**.
- **Development:** PACT provides a valuable tool for **guiding sustainable development** and responsible implementation of ML algorithms.

Agenda

- Motivation
- Preliminary Work-1:
  - Pact: Accurate power analysis and carbon emission tracking for sustainability
- **Preliminary Work-2:**
  - Mamba World Model for RL: Enhancing Long-term Memory in Model-based Reinforcement Learning
- Ongoing and Proposed Work:
  - State-Space Model-Based Reinforcement Learning for Dynamic Spectrum Access
  - Trajectory Classification Based on Topological Features for Stability Analysis and Sample-Efficient RL
- Timeline
- Conclusions

Overview and Related Work

- **Motivation**
  - Deep RL algorithms still suffer from low sample efficiency
  - While Model-based RL algorithms offer high sample efficiency, they are hindered by two key challenges:
    - Low training speed
    - They cannot handle long-term memory dependencies
- **Related work on model-based RL**
  - The backbones for the world model include LSTM, GRU, Transformer, and Mamba

Attribute	DreamerV3 [1]	STORM [2]	DRAMA [3]	MWM (ours)
Sequence model	GRU	Transformer	Mamba	Mamba
State update	Autoregressive	Past 16 frames	Past 16 frames	Autoregressive
Parallel training	No	Yes	Yes	Yes

[1] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering Diverse Domains through World Models," Apr. 17, 2024, *arXiv*: arXiv:2301.04104 .
[2] W. Zhang, G. Wang, J. Sun, Y. Yuan, and G. Huang, "STORM: Efficient Stochastic Transformer based World Models for Reinforcement Learning".
[3] W. Wang, I. Dusparic, Y. Shi, K. Zhang, and V. Cahill, "Drama: Mamba-Enabled Model-Based Reinforcement Learning Is Sample and Parameter Efficient," May 16, 2025, *arXiv*: arXiv:2410.08893.

Related Work

- **Dreamer V3 is based on Gated Recurrent Unit**
  - Non-linear operation makes it not parallelizable
  - Hard to retain long term memory in hidden state
- **STORM is Transformer based world model**
  - Linear growing KV cache significantly limit its context length

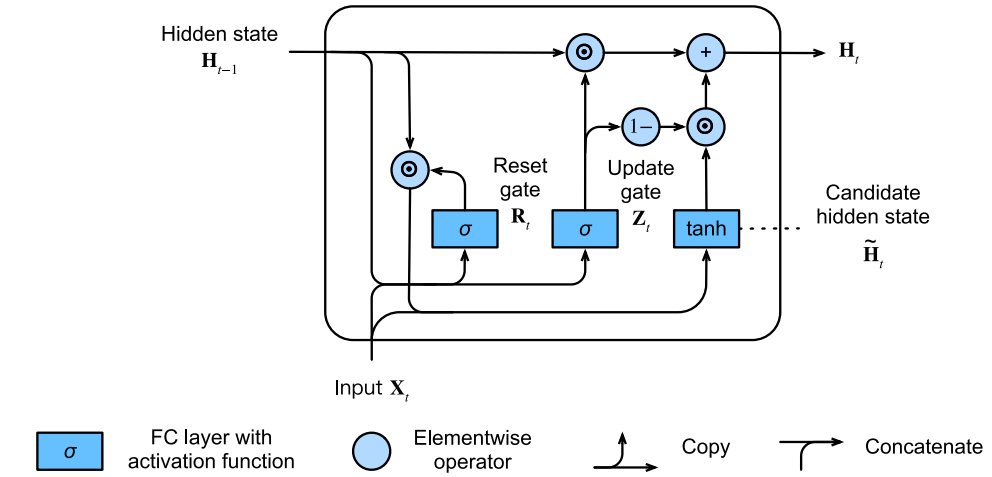
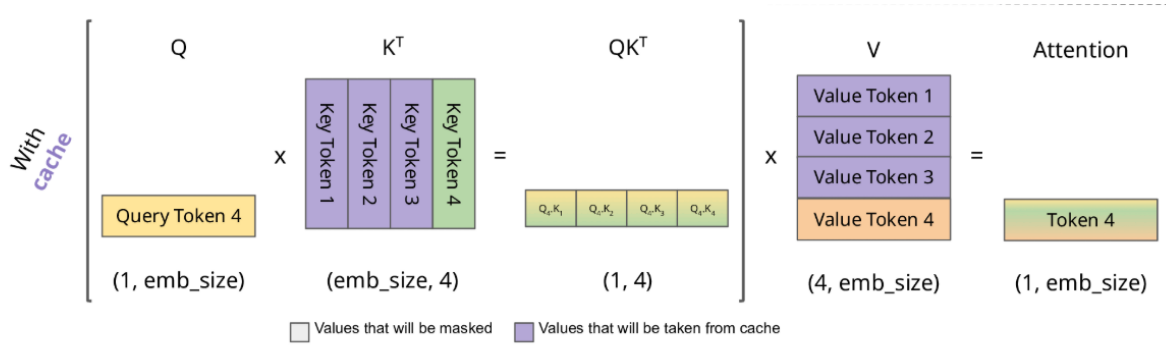


Illustration of GRU architecture

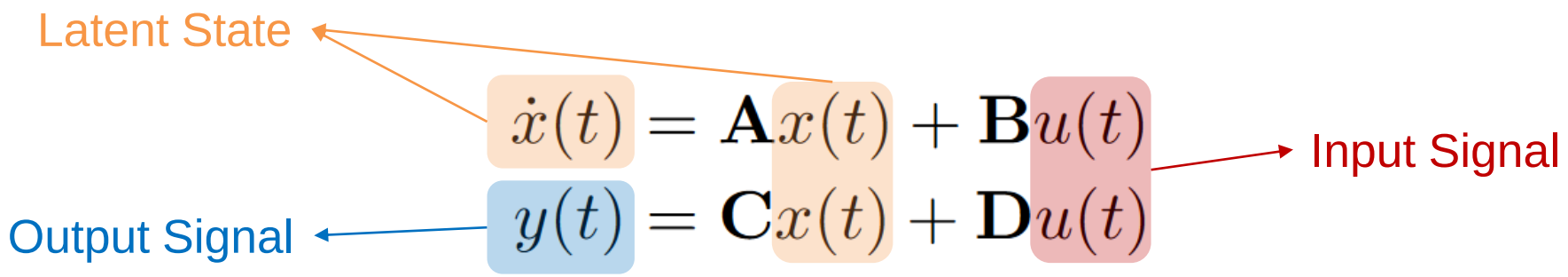


Linear growing KV Cache

[1] Dive into Deep Learning. Accessed at: https://d2l.ai/chapter_recurrent-modern/gru.html
[2] Transformers KV Caching Explained. Accessed at: https://medium.com/@joaolages/kv-caching-explained-276520203249

Background: Mamba Architecture

- Continuous-time linear SSMs are defined by the following equations:



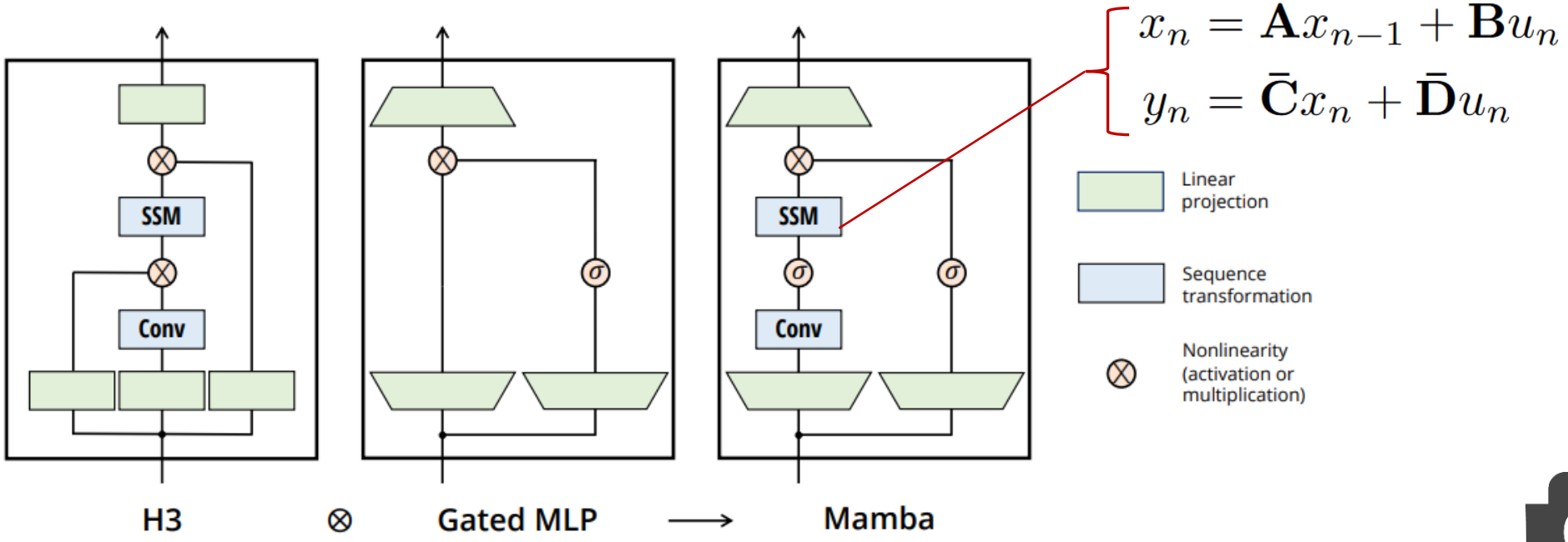
where $\mathbf{A} \in \mathbb{R}^{P\times P}$ is the state matrix, $\mathbf{B} \in \mathbb{R}^{P\times U}$ is the input matrix, $\mathbf{C} \in \mathbb{R}^{M\times P}$ is the output matrix and $\mathbf{D} \in \mathbb{R}^{M\times U}$ is the feedthrough matrix.

- We can discretize the SSM with a learnable fixed step size Δ to obtain the discrete representation:

$$x_n = \bar{\mathbf{A}}x_{n-1} + \bar{\mathbf{B}}u_n$$
$$y_n = \bar{\mathbf{C}}x_n + \bar{\mathbf{D}}u_n$$

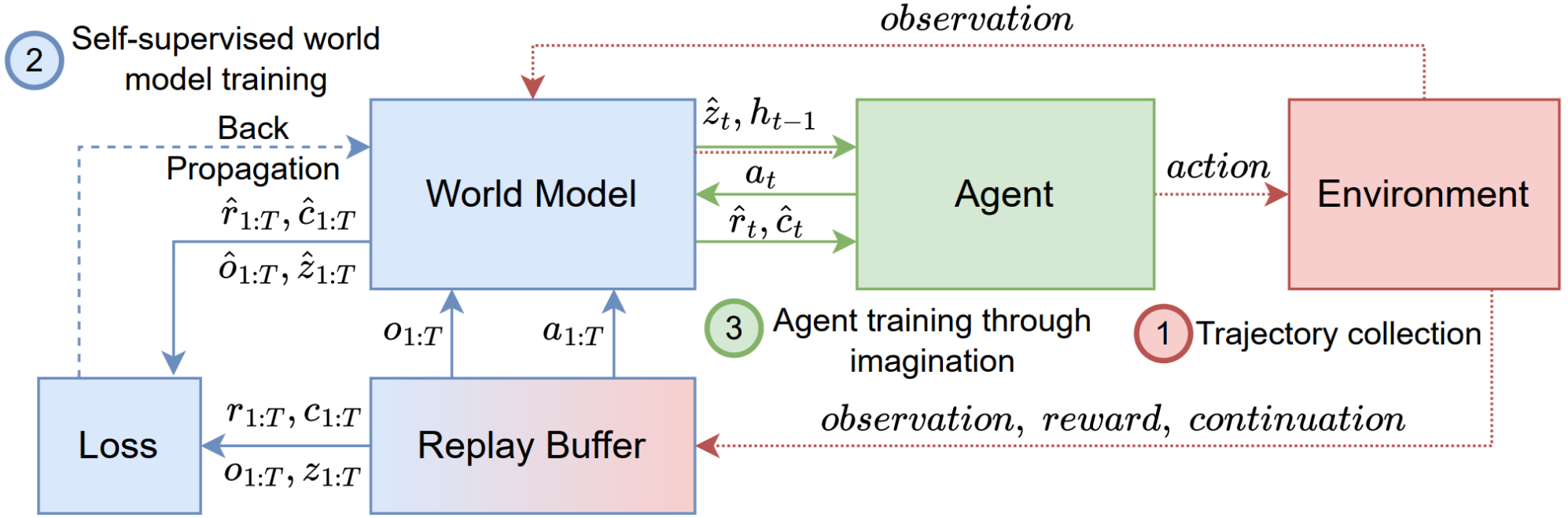
Background: Mamba Architecture (Cont.)

- **Mamba is a sequence-to-sequence model that combines a linear system as its recurrent unit with projections and non-linear activations**
- **Properties of Mamba:**
  - Linear time and memory complexity during inference
  - Superior long-term memory
  - Hardware-friendly parallel scan for training



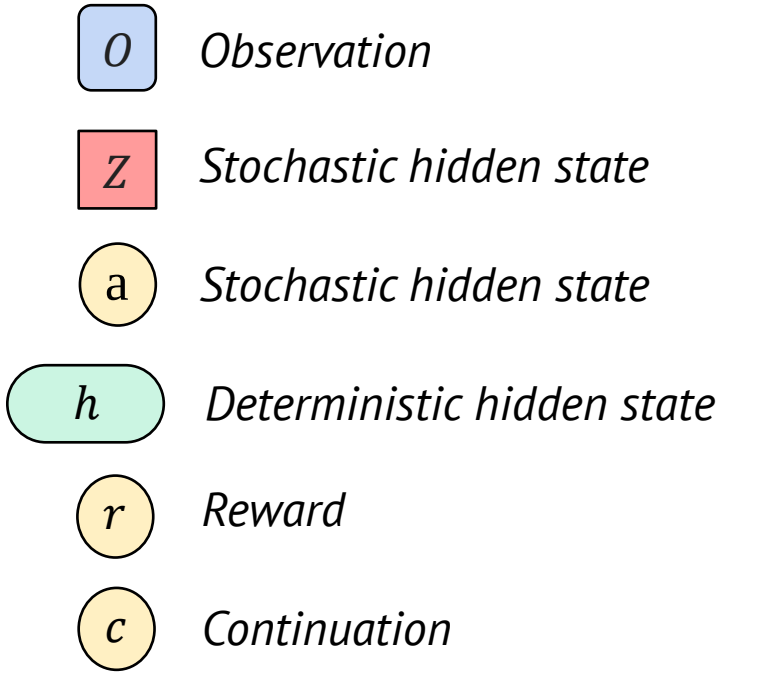
Overview of World Model for RL Training

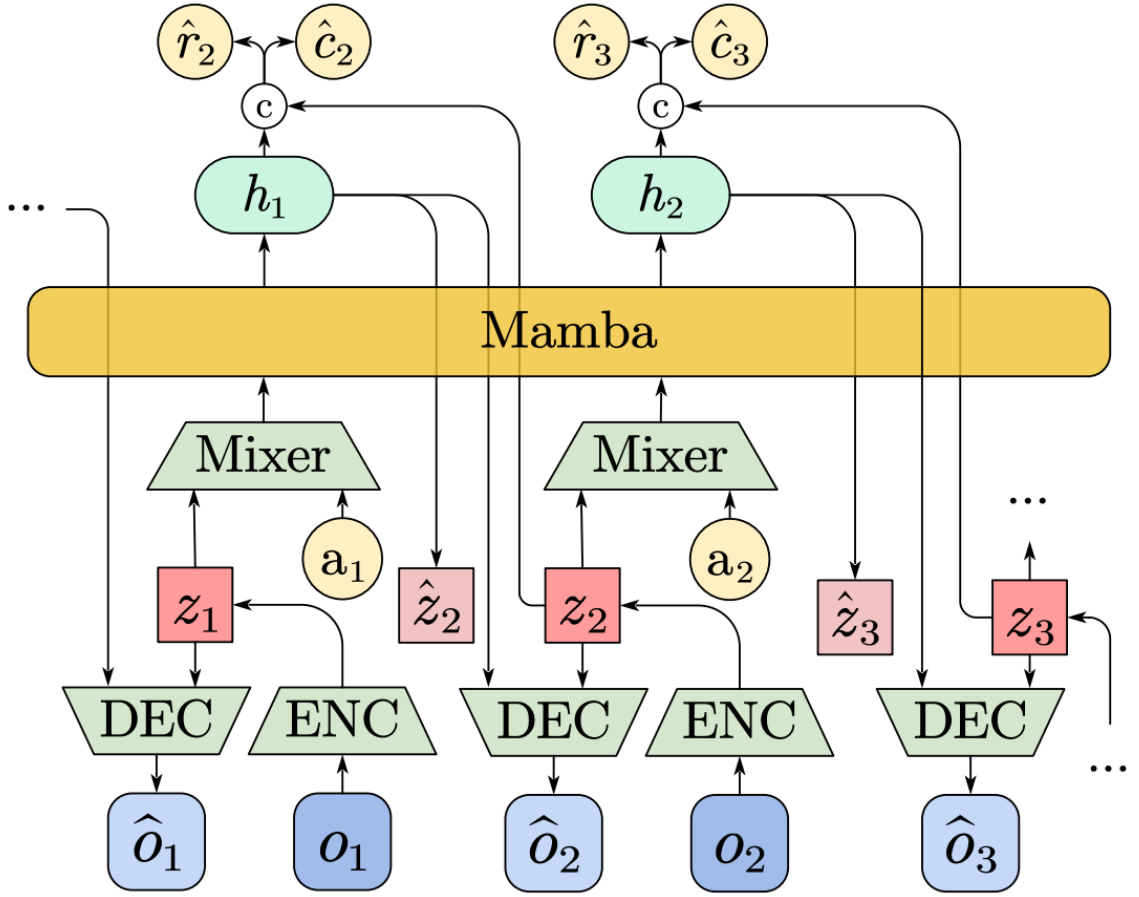
- **A full iteration of world model training includes three steps**
  1. The agent gathers experience with the real environment
  2. A batch of trajectories with a fixed length is sampled to update the world model
  3. The agent is trained on-policy with the world model for a fixed horizon



World Model Architecture

- **The world model is trained to predict the next observation given the current observation and action**





World Model Loss with entropy regularization

- **World model loss includes the following terms:**

$$\mathcal{L}_t^{\text{pred}}(\phi) \doteq -\ln p_\phi\left(o_t \mid z_t, h_{t-1}\right) - \ln p_\phi\left(r_t \mid z_t, h_{t-1}\right) - \ln p_\phi\left(c_t \mid z_t, h_{t-1}\right)$$

1. Log likelihood of predictions

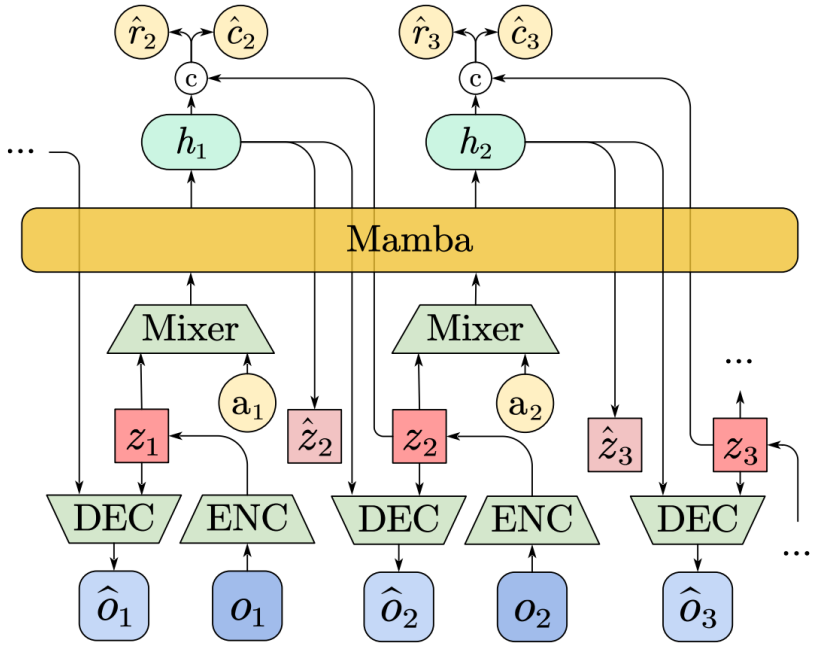
$$\mathcal{L}_t^{\text{dyn}}(\phi) \doteq \max\left(1,\ \mathrm{KL}\left[\mathrm{sg}\left(q_\phi(z_t \mid o_t)\right) \,\middle\|\, p_\phi(\hat{z}_t \mid h_{t-1})\right]\right)$$

$$\mathcal{L}_t^{\text{rep}}(\phi) \doteq \max\left(1,\ \mathrm{KL}\left[q_\phi(z_t \mid o_t) \,\middle\|\, \mathrm{sg}\left(p_\phi(\hat{z}_t \mid h_{t-1})\right)\right]\right)$$

2. 3. Pulling posterior and predicted distribution together

$$\mathcal{L}_t^{\text{ent}}(\phi) \doteq H\left(q_\phi(\cdot \mid o_t)\right).$$

4. Proposed Entropy term



- **The world model loss is a weighted sum of individual loss terms:**

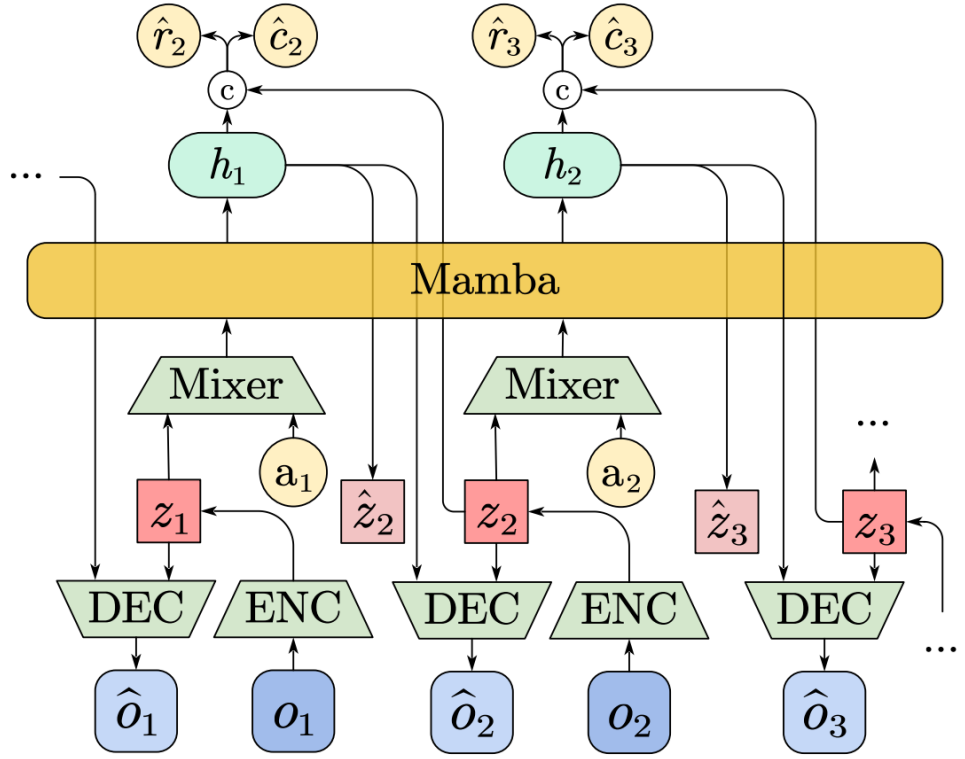
$$\mathcal{L}(\phi) \doteq \mathbb{E}_{q_\phi}\left[\sum_{t=1}^{T}\left(\mathcal{L}_t^{\text{pred}}(\phi) + \beta_1\mathcal{L}_t^{\text{dyn}}(\phi) + \beta_2\mathcal{L}_t^{\text{rep}}(\phi) - \eta\mathcal{L}_t^{\text{ent}}(\phi)\right)\right]$$

η is dynamically scaled to target the entropy of 50 nats, ~45% of uniform distribution

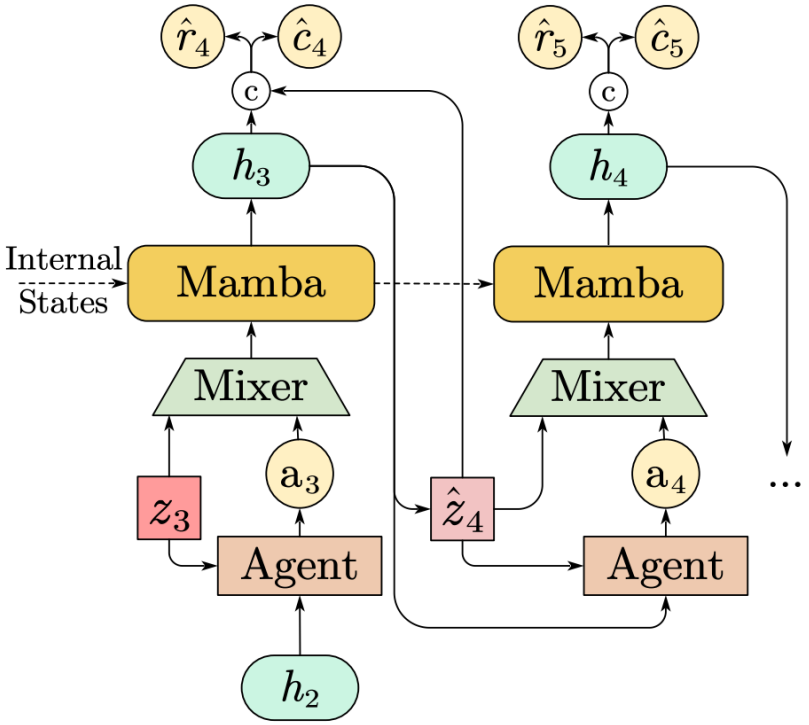
Entropy term is proposed to avoid overfit to memorize deterministic transitions and enhance imagination

World Model Architecture

- **In the agent training stage, the agent collect rollout and use on-policy algorithm REINFORCE**



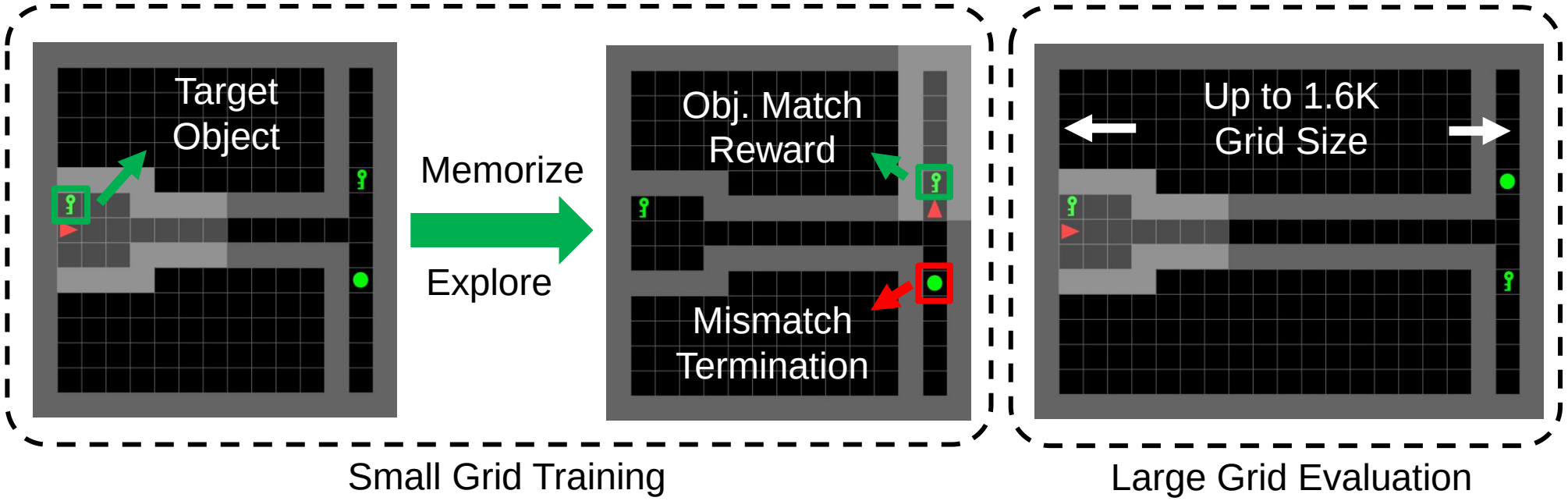
Parallel Training



Inference

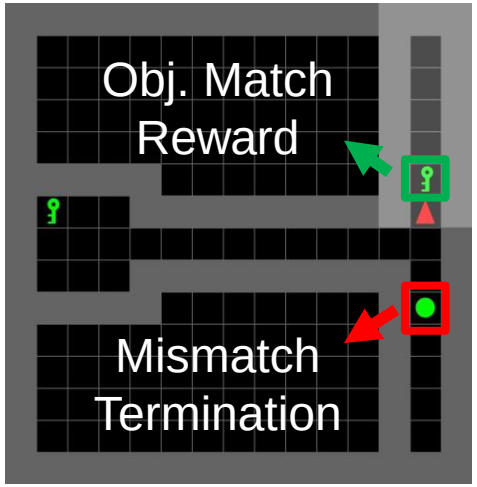
Long-term Memory Test: Grid World T Maze

- **The world model is trained on small grids to accelerate training and evaluated on large grids up to 1600 in length**
- **Reward:** $1 - 0.9 \times (\frac{step}{max\ step})$ if object match, otherwise zero

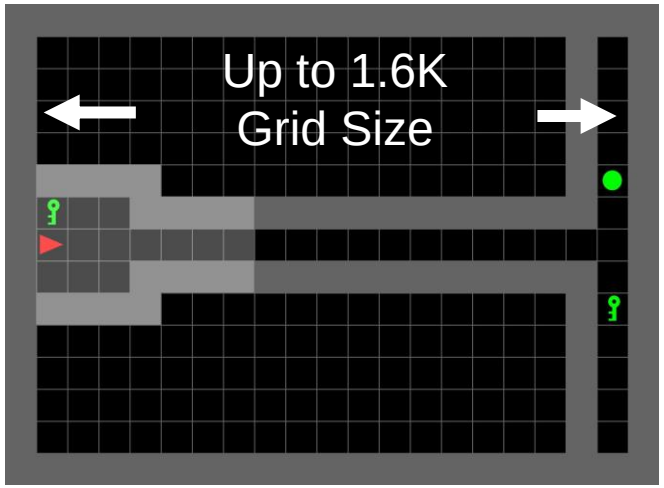


Long-term Memory Test: Grid World T Maze (cont.)

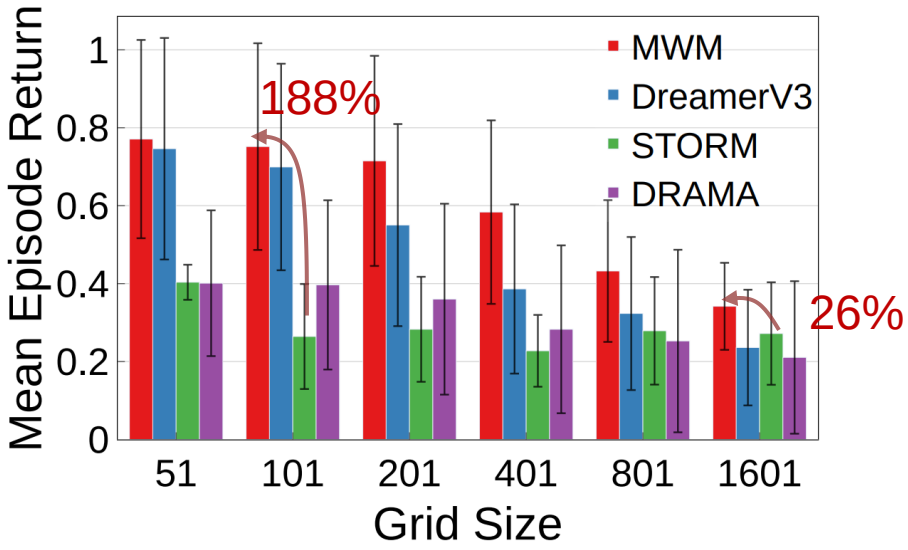
- For training efficiency, we use a compact 6.7M-parameter MWM model, the XS version of DreamerV3 with 8M parameters, 20M STORM model, and 7M DRAMA model



Small Grid Training



Large Grid Evaluation



Atari 100k Benchmark

Game	Random	Human	SimPLe	TWM	IRIS	DreamerV3	STORM	DRAMA	MWM
Alien	228	7128	617	675	420	**1118**	984	820	562
Amidar	6	1720	74	122	143	97	**205**	131	163
Assault	222	742	527	683	**1524**	683	801	539	1063
Asterix	210	8503	1128	1116	854	1062	1028	**1632**	1043
Bank Heist	14	753	34	467	53	398	641	137	**995**
Battle Zone	2360	37188	4031	5068	13074	**20300**	13540	10860	10330
Boxing	0	12	8	78	70	**82**	**80**	78	**83**
Breakout	2	30	16	20	**84**	10	16	7	11
Chopper Command	811	7388	979	1697	1565	**2222**	1888	1642	1003
Crazy Climber	10780	35829	62584	71820	59234	**86225**	66776	**83931**	63146
Demon Attack	152	1971	208	350	**2034**	577	165	201	181
Freeway	0	30	17	24	**31**	0	0	15	3
Frostbite	65	4335	237	1476	259	**3377**	1316	785	1364
Gopher	258	2413	597	1675	2236	2160	**8240**	2757	6125
Hero	1027	30826	2657	7254	7237	**13354**	11044	7946	8401
James Bond	29	303	101	362	463	**540**	509	372	310
Kangaroo	52	3035	51	1240	838	2643	4208	1384	**5380**
Krull	1598	2666	2204	6349	6616	8171	8413	**9693**	8722
Kung Fu Master	256	22736	14862	24555	21760	**25900**	**26182**	23920	16892
Ms Pacman	307	6952	1480	1588	1588	1521	**2673**	2270	2008
Pong	-21	15	13	**19**	15	-4	11	15	**18**
Private Eye	25	69571	35	87	100	3238	**7781**	90	-148
Qbert	164	13455	1289	3331	746	2921	**4522**	796	2308
Road Runner	12	7845	5641	9109	9615	**19230**	17564	14020	14373
Seaquest	68	42055	683	774	661	**962**	525	497	463
Up N Down	533	11693	3350	15982	3546	**46910**	7985	7387	5106
Human Mean	0	100	33	96	105	**125**	**122** [1]	105	116
Human Median	0	100	13	51	29	49	**58**	27	30

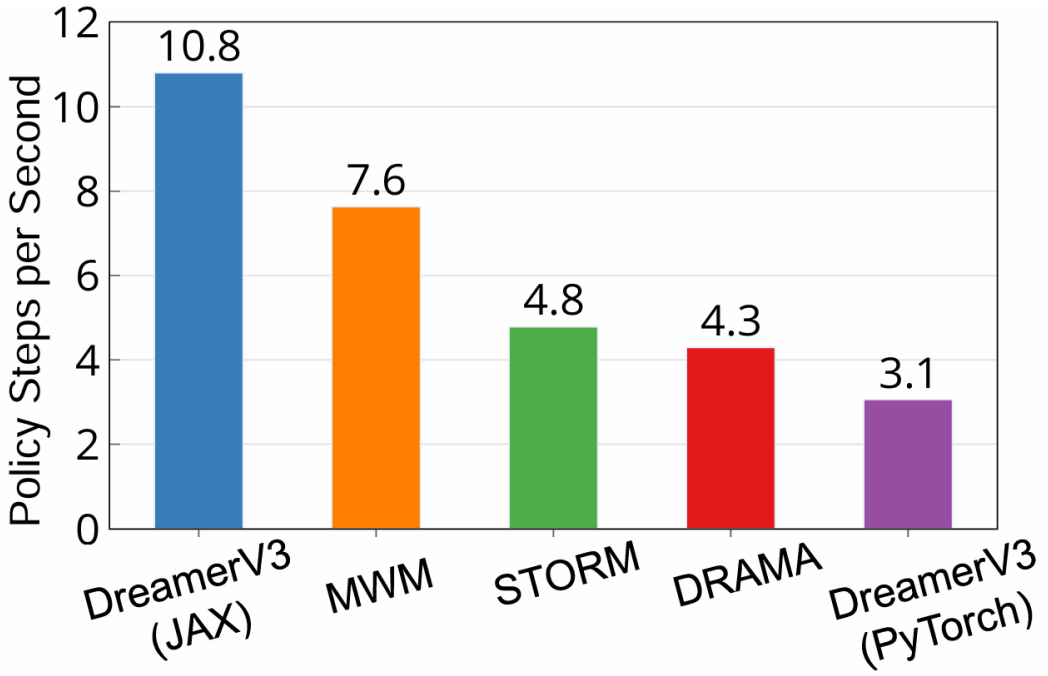
- **Norm score calculation:**

$$\frac{eval - random}{human - random}$$

- **Parameters:**
  - MWM (ours) 19.6M
  - STORM 20M
  - DreamerV3 200M
  - DRAMA 7M (slower)
    - DRAMA uses sample strategy similar to priority replay to enhance performance

Training Speed Evaluation

- **We leverage the intermediate states of world model training to speed up imagination**
- **Our MWM trains the fastest among all the PyTorch-based world models**



Ablation Study of Entropy Regularization

- **Conducted on a subset of environments where world model RL performs strongly.**
- **The entropy loss $\mathcal{L}^{ent}$ shows significant improvement across multiple environments — from complex tasks like Krull to simpler tasks such as Boxing.**

Game	Random	Human	MWM w/o $\mathcal{L}^{\text{ent}}$	MWM
Assault	222	742	**1043**	**1063**
Asterix	210	8503	**1007**	**1043**
BankHeist	14	753	**988**	**995**
Boxing	0	12	**80**	**83**
CrazyClimber	10780	35829	51812	**63146**
Gopher	258	2413	3506	**6125**
Kangaroo	52	3035	4662	**5380**
Krull	1598	2666	8032	**8722**
Pong	-21	15	**18**	**18**
Human Mean	0	100	238	**270**
Human Median	0	100	154	**179**

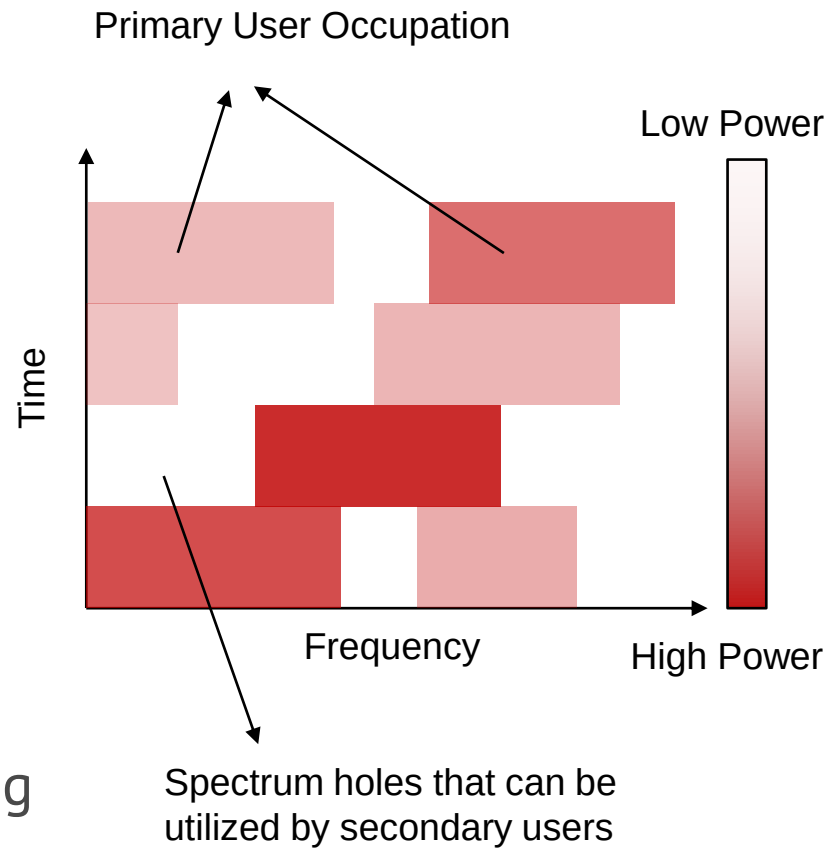
Some videos here might be helpful

Agenda

- Motivation
- Preliminary Work-1:
  - Pact: Accurate power analysis and carbon emission tracking for sustainability
- Preliminary Work-2:
  - Mamba World Model for RL: Enhancing Long-term Memory in Model-based Reinforcement Learning
- **Ongoing and Proposed Work:**
  - State-Space Model-Based Reinforcement Learning for Dynamic Spectrum Access
  - Reinforcement Learning for Chip Design
- Timeline
- Conclusions

Dynamic Spectrum Access: Motivation

- **Problem: Low Spectrum Utilization**
  - Current spectrum allocation is static and inflexible
  - Spectrum holes in time/frequency are unused
  - Growing spectrum demand + limited resources
- **Cognitive Radio**
  - Cognitive Radio (CR) allows Secondary Users (SUs) to use unoccupied spectrum (spectrum holes)
- **Challenges in PU–SU Coexistence**
  - Primary Users (PUs) do not disclose their transmission behavior to Sus
  - Unknown and dynamic spectrum usage complicates sensing and opportunistic access

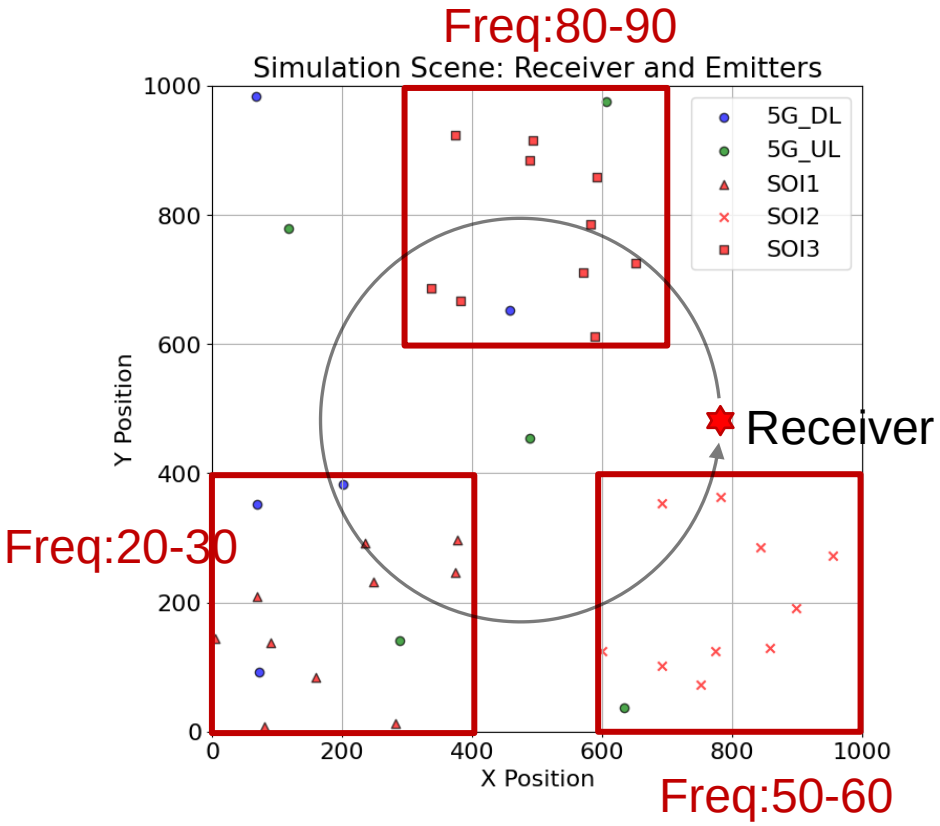


Overview Intelligent Resource Management (IRM)

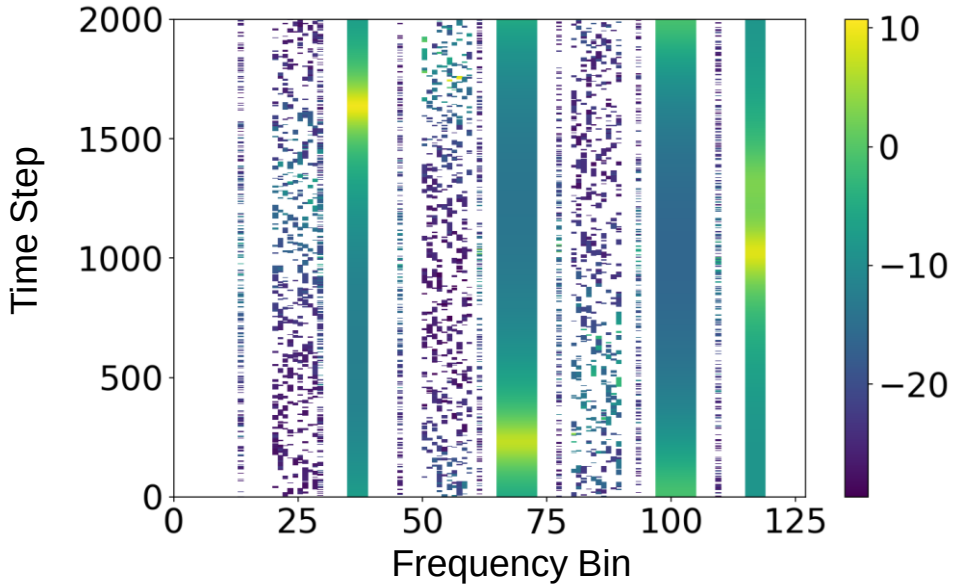
- **The limited receive channel and processing resources need to be managed with intelligent resource management**
  - Solving for the optimal solution for real-world environments is impossible since the spectrum usage is highly dynamic and unknown to SUs
  - The real state of the system is only partially observable
- **Formulate the problem as a Partially Observable Markov Decision Process**
  - The agent receives the partially observed spectrum and forms a hidden state
  - Based on the hidden state, it intelligently configures the antenna and filter configurations to gain more information about the environment

IRM Simulation Setup

- **5G Downlink, 5G Uplink, and Signal of Interest (SOI) are scattered in the scene**
- **The receiver is moving counterclockwise in the scene**

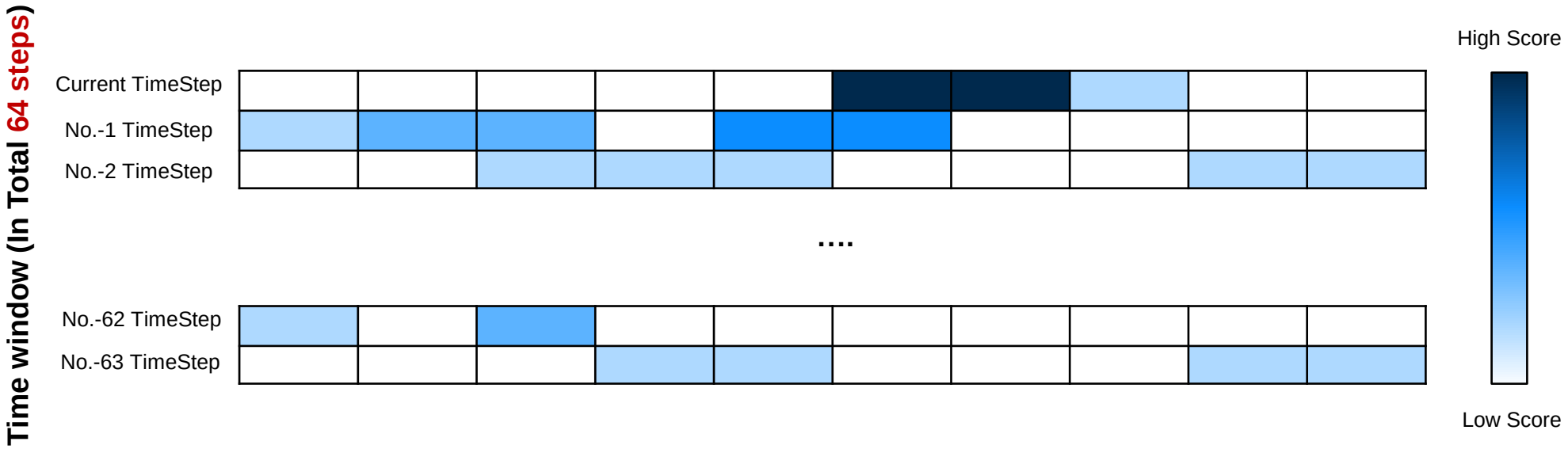


Spectrogram of all emitter signals (without noise)



IRM State and Action Space

- **State Space is a snapshot of the past detection results. The sums of the scores of the previously detected signals are put into the corresponding frequency bins.**
  - Detection Score depends on whether the signals (Env ground truth) are observed or not. If observed, it will add a certain amount of score, depends on the importance of the signal (e.g., SOI weights more)
- **State Space also includes the current time step and the last time a band was observed.**
- **The action of the RL agent controls the center frequency and bandwidth of each antenna group**



IRM Reward

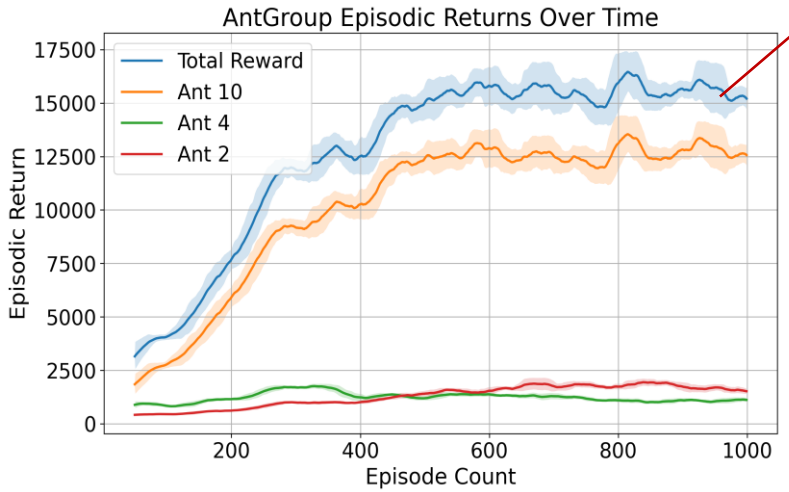
- **The reward function at time step t is the summation of the weights of all the detected signals**
- **Let $\mathcal{D}_t$ denote all the detected signals, then reward can be written as:**

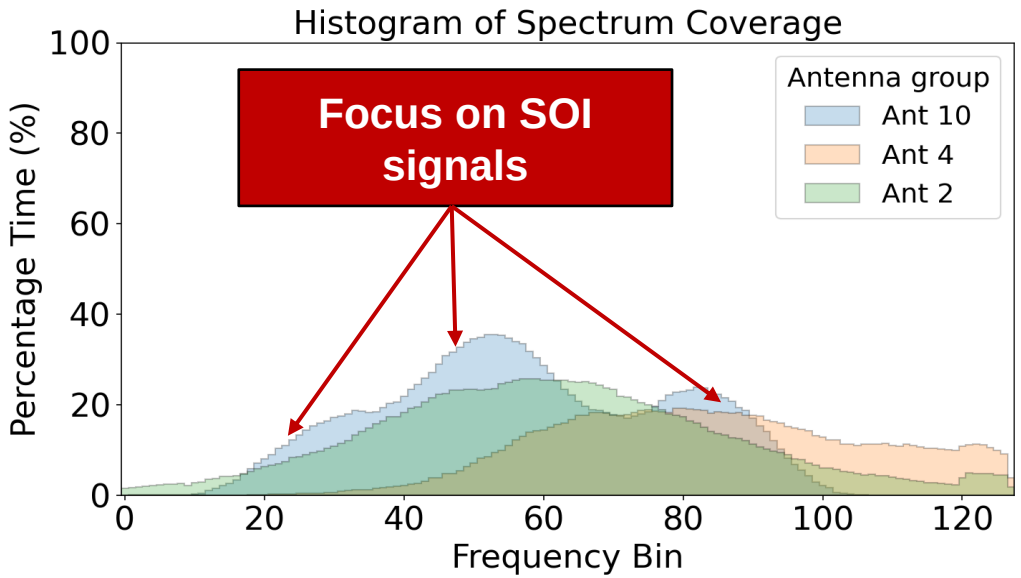
$$r_t = \sum_{s \in \mathcal{D}_t} \text{weight}(s), \qquad \text{where weight}(s) = \begin{cases} 1 & \text{if } s \text{ is 5G Downlink} \\ 5 & \text{if } s \text{ is 5G Uplink} \\ 10 & \text{if } s \text{ is SOI} \end{cases}$$

Preliminary Results

- **RL agent is trained using PPO with three antenna groups containing 10, 4, and 2 antennas**
  - Converge around 500 episodes, achieving 2.7x larger reward than linear scan
  - The 10-antenna group contributes to about 80% of the detection reward, and is able to focus on a larger number of SOIs
  - The smaller groups facilitate exploration

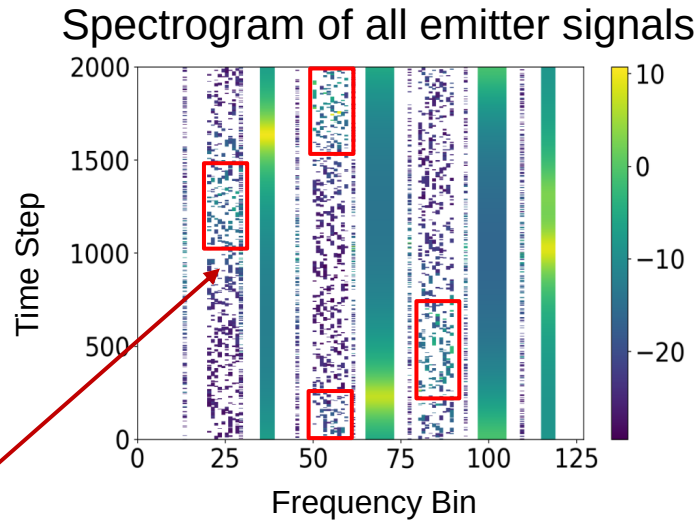
2.7x larger than linear scan reward (5595)

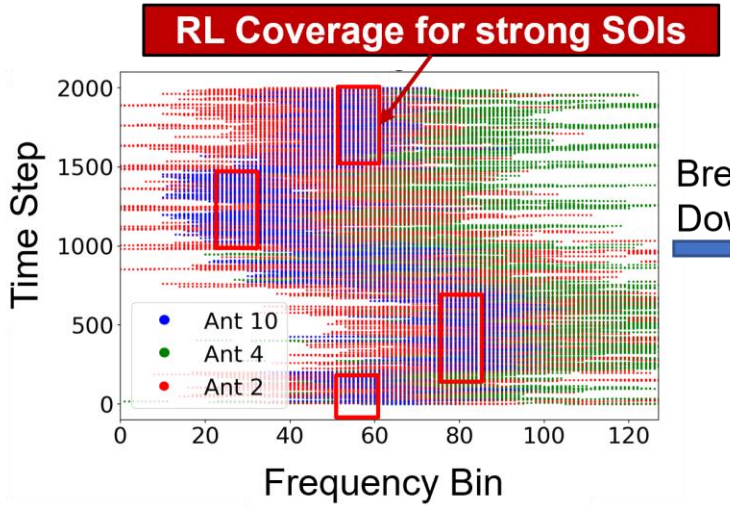




Explainable RL agent behaviors

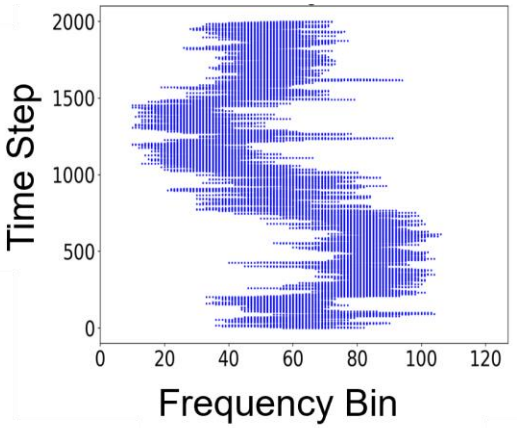
- Experiments show that **RL can *leverage the locality* of SOIs**, shifting its focus to where the signal power is the strongest
- **RL can also efficiently explore and find *all three groups* of SOIs**



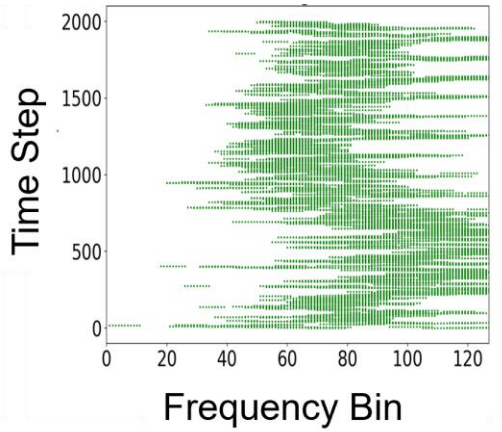


Action Coverage of All 3 Antenna Groups

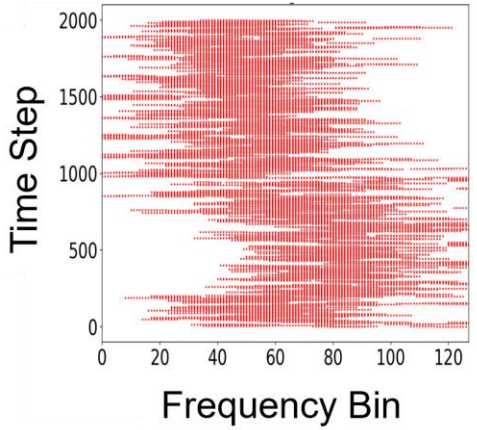
Break Down



Coverage of the 10-Antenna Group



4-Antenna Group



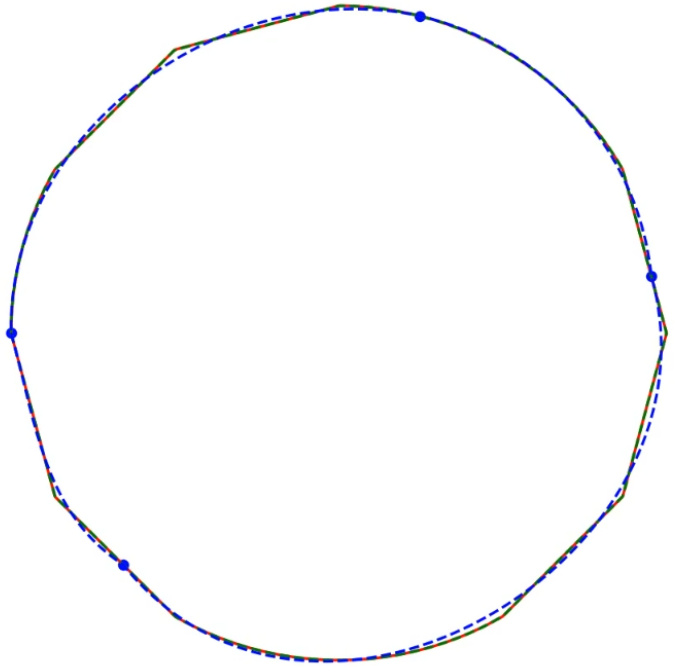
2-Antenna Group

Agenda

- **Motivation**
- **Preliminary Work-1:**
  - Pact: Accurate power analysis and carbon emission tracking for sustainability
- **Preliminary Work-2:**
  - Mamba World Model for RL: Enhancing Long-term Memory in Model-based Reinforcement Learning
- **Ongoing and Proposed Work:**
  - State-Space Model-Based Reinforcement Learning for Dynamic Spectrum Access
  - Reinforcement Learning for Chip Design
- **Timeline**
- **Conclusions**

Bezier Fitted Polygon for Chip Design

- **Masks are generated for further photolithography and etching in the foundry**
- **Millions of points are sampled, with unnecessary points that create significant overhead**
  - Partition into convex shapes
  - Intelligently down sample the convex shape with Reinforcement Learning
  - Reconstruct by fitting a Bezier curve



Example of fitted curve of convex shape

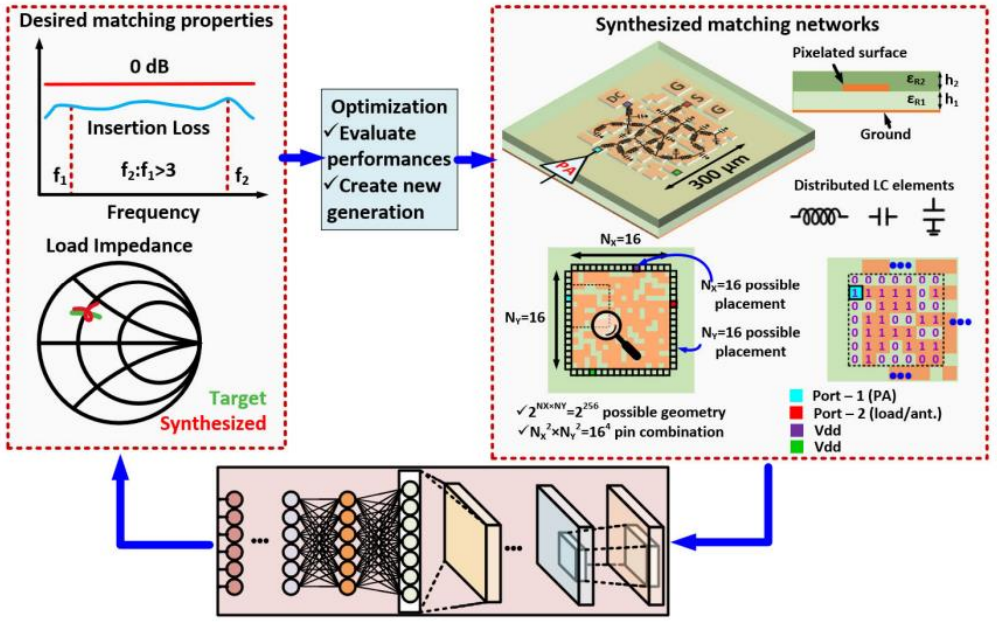
State: Points on polygon, down sampled points

Reward: $r_t = \Delta IOU - C$, where C is constant for step penalty

Action: Next point to choose from

Reinforcement Learning for RFIC design

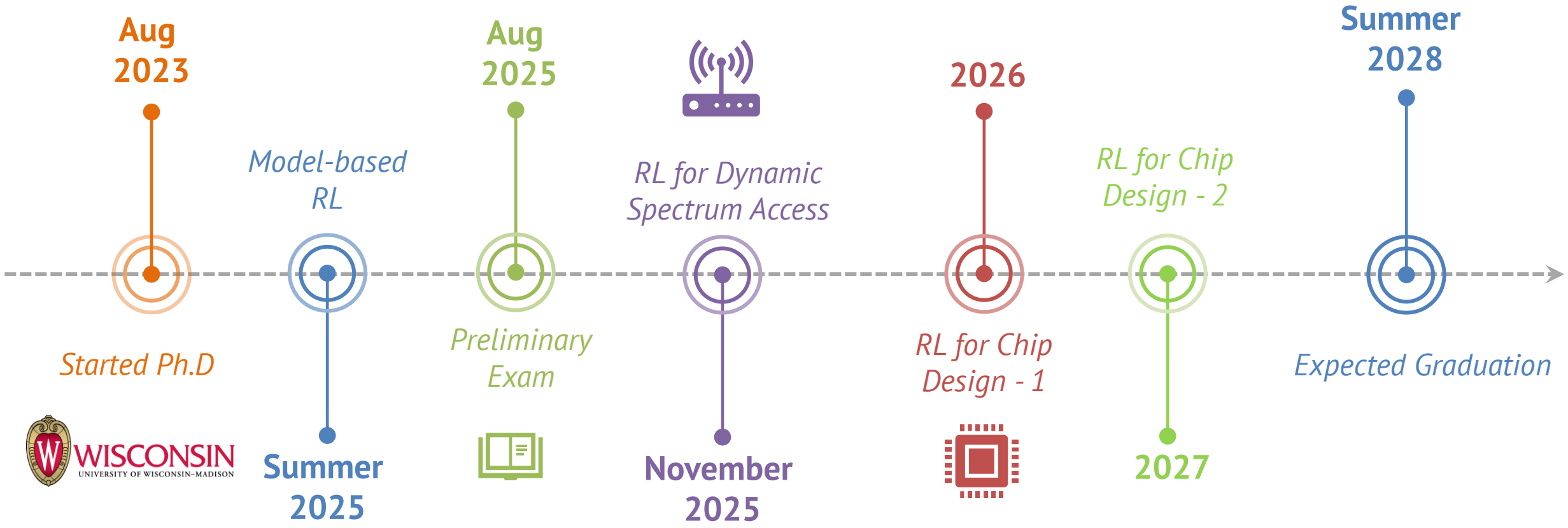
- **Emerging methods use machine learning-driven surrogate models and optimization to achieve desired properties**
- **World models for RFIC design offer key advantages:**
  - They couple world model learning with optimization
  - Avoids over-fitting and unnecessary surrogate model learning
  - Flexible, integrated pipeline for diverse applications



Deep-CNN-based surrogate model that provide rapid feed back for optimization process [1]

[1] E. A. Karahan, Z. Liu, and K. Sengupta, "Deep-Learning-Based Inverse-Designed Millimeter-Wave Passives and Power Amplifiers," *IEEE Journal of Solid-State Circuits*, vol. 58, no. 11, pp. 3074–3088, Nov. 2023, doi: 10.1109/JSSC.2023.3276315.

Timeline


Aug
2023
Started Ph.D
WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON
Model-based
RL
Summer
2025
Aug
2025
Preliminary
Exam
RL for Dynamic
Spectrum Access
November
2025
2026
RL for Chip
Design - 1
RL for Chip
Design - 2
2027
Summer
2028
Expected Graduation