Non-Myopic Neuro Symbolic AI for Smart Infrastructure

Abhishek Dubey, Vanderbilt University

(Several Collaborators, over the years – indicated through out)

Supported in part by the National Science Foundation, Siemens, Nissan, CT, Cisco, DARPA, NASA, AFRL and Department of Energy.

What is Smart Infrastructure?

Modern cities as large-scale cyber-physical systems

A modern city is a cyber-physical system: thousands of sensors, embedded computers, communication networks, and human operators continuously interact across transportation and energy infrastructure. "Smart" infrastructure means closing the loop — using real-time data to make decisions that look ahead, not just react.

Sensing

Traffic detectors, smart meters, GPS, cameras, SCADA — continuous streams of real-world state

Networks

Communication fabric linking sensors to computation. Latency, bandwidth, and reliability constrain decisions

Computation

Embedded controllers and cloud systems that process data, predict, and plan in real time

Human Operators

Dispatchers, engineers, and city managers who set policies, override decisions, and manage exceptions

The challenge: most current decision systems are myopic — they react to immediate problems without planning for long-term consequences or cross-system effects.

Learning Based Approaches for Planning

Machine learning for perception, prediction, and data-driven decision making

Demand Forecasting

Deep learning models predict passenger demand, energy load, and traffic flow from historical patterns and real-time streams.

Disruption Detection

Anomaly detection networks identify incidents, equipment failures, and unusual patterns in sensor data within seconds.

State Estimation

Neural networks fuse noisy, incomplete sensor data into coherent estimates of system state across infrastructure domains.

Policy Learning

Reinforcement learning discovers control policies (signal timing, dispatch, scheduling) that outperform hand-tuned rules.

Strengths

- Learns complex patterns from data
- Scales to complex input spaces (images, time series, graphs)
- Fast inference: real-time predictions at city scale

Limitations

- No inherent notion of constraints, rules, or safety guarantees
- Myopic: optimizes immediate reward without long-term lookahead
- Black box: hard to verify, explain, or compose across domains

Neural methods excel at perception and prediction, but infrastructure decisions require reasoning about constraints, feasibility, and long-term consequences.

Symbolic Methods

Rule-based reasoning, constraint satisfaction, and formal planning

Constraint Satisfaction

Enforce hard constraints: vehicle capacity limits, ADA compliance, grid frequency bounds, safety zones.

Rule-Based Reasoning

Encode domain expertise as logical rules: if a bus route is disrupted, activate predefined alternative service patterns. Transparent and auditable.

Formal Planning

Mixed-integer optimization. Plan sequences of actions with provable properties over finite horizons.

Compositional Logic

Reason about how systems interact: energy dispatch affects EV charging, which affects grid load, which affects transit electrification.

Strengths

- Guarantees: safety constraints are never violated
- Explainable: every decision has a traceable reasoning chain
- Composable: domain-specific reasoners combine cleanly

Limitations

- Brittle: manually authored rules can't cover every situation
- No learning: can't improve from data or adapt to novel patterns
- Combinatorial explosion: planning over large action spaces is costly

Symbolic methods provide guarantees and structure, but struggle with the scale, noise, and novelty of real-world infrastructure data.

The Neuro-Symbolic Approach

Learning functions+ symbolic reasoning + non-myopic planning = decisions that are fast, safe, and far-sighted

Neural

Perceive state, predict demand, estimate uncertainty, learn from data at scale

+

Symbolic

Enforce constraints, reason about rules, compose across domains, provide guarantees

=

Non-Myopic

Plan ahead with uncertainty, balance short-term needs with long-term utility, while closing the loop

Deployed Systems: Chattanooga & Nashville

Real-world non-myopic neuro-symbolic decision making in action

Vehicle-to-Grid Coordination

Rolling-horizon scheduling of EV charging/discharging. Non-myopic planning avoids peak demand spikes while meeting driver needs.

Microtransit Scheduling

On-demand routing with look-ahead assignment. Constraint-aware reasoning ensures ADA compliance and capacity limits.

School Bus Operations

Routing across hundreds of buses with time-window constraints, safety rules, and rolling replanning.

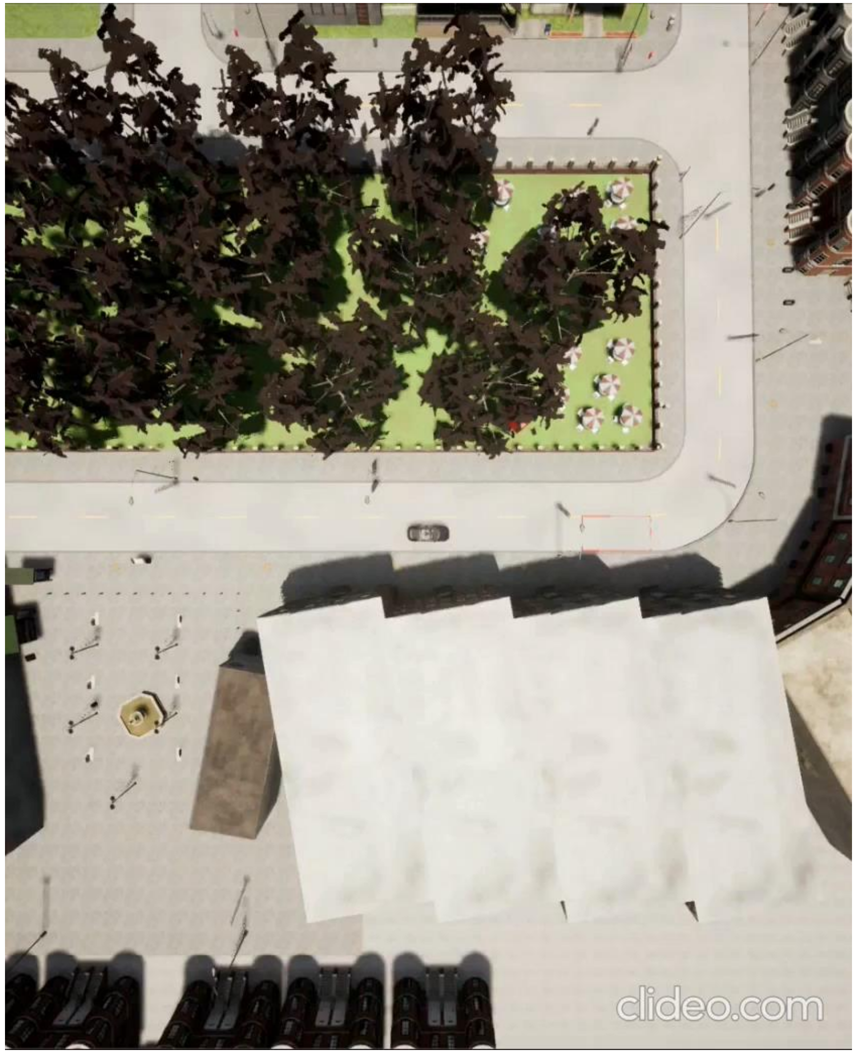
Transit Disruption Management

Plans alternative service under disruption-duration uncertainty. Anticipates cascading passenger displacement.

All systems: rolling planning + constraint-aware reasoning + uncertainty modeling.

What are the key Challenges

Why is this hard: complexity grows combinatorially across time, uncertainty, and constraints, not just state size.



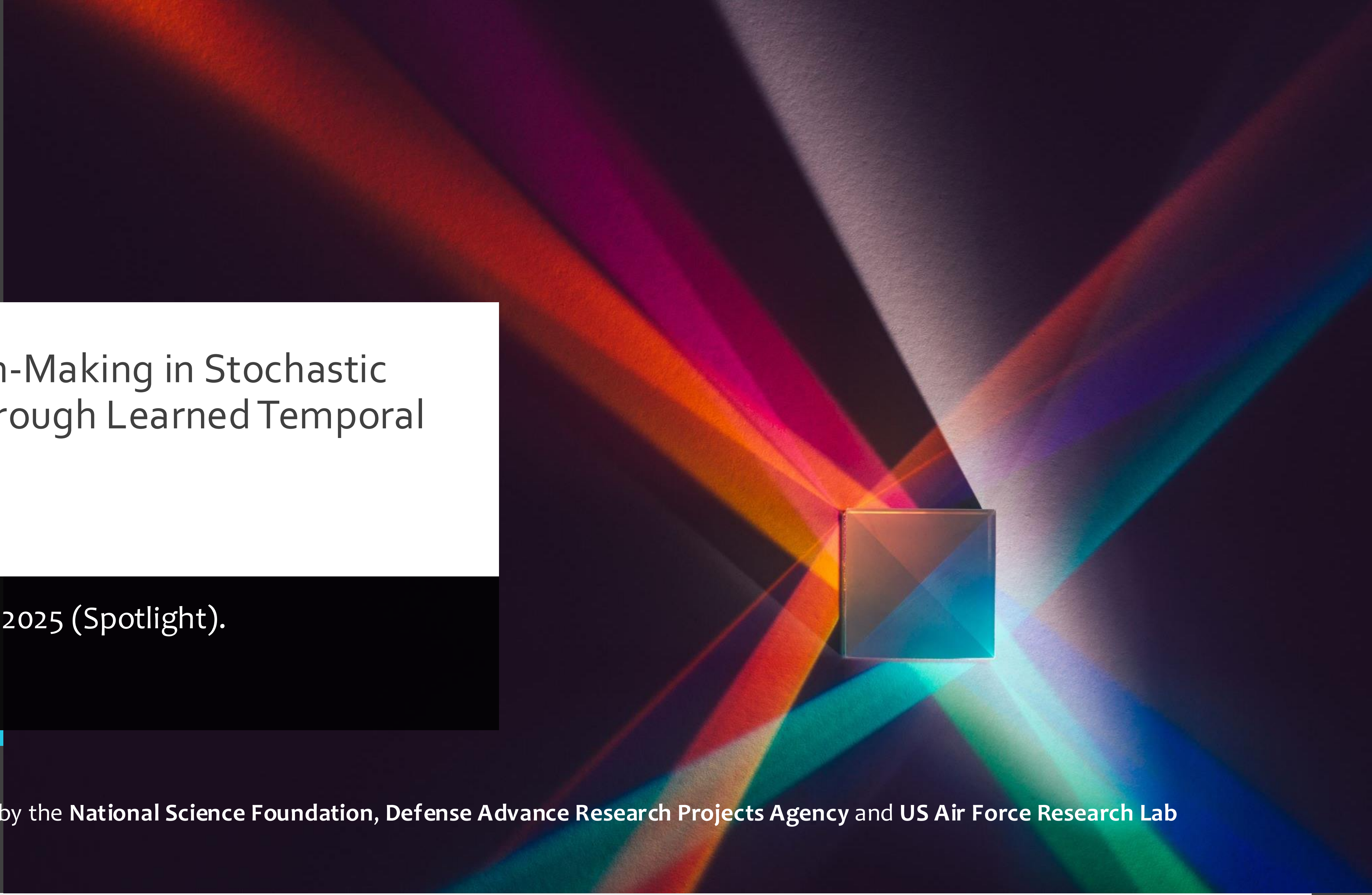
- **High-dimensional states & actions**: Continuous dynamics + discrete dynamics. Large Action Space – often continuous (e.g., EV SoC × charger assignment, fleet routing)
- **Long horizons**: Decisions affect future feasibility and value (e.g., today's charging impacts demand charge at the end of month.)
- **Partial observability & uncertainty**: Incomplete, multi-modal futures (e.g., uncertain arrivals, demand, disruptions)
- **Hard physical & safety constraints**: Limits cannot be violated (e.g., battery physics, grid capacity, vehicle limits)
- **Real-time Bounds and Computation Constraints:** Often the systems and end-to-end decisions must be taken online under strict scheduling and resource constraints.
- **Non-stationarity and Variance of Settings**: Dynamics evolve over time (e.g., seasonal loads, behavior drift, incidents).

Overview of some recent work

- **Persistent & Constraint-Aware Planning (P-V2B)- (Dubey et al. – ICCPS 2026 (under review)):** A Learned Value Function (neural) approximates long horizon returns and guides a Symbolic control that rigorously enforces hard constraints and safety bounds for Vehicle to building
- **Human-in-the-loop Negotiation (CONSENT for V2B problems) (Sen et al. – AAMAS 2026):** Combines Neural preference learning (predicting user flexibility) with Symbolic mechanism design (logic/rules) to guarantee incentive compatibility and voluntary participation.
- **Evaluation Framework (NS-Gym) (Keplinger et al. – NeurIPS 2025):** Decouples agent logic from environmental drift, allowing rigorous stress-testing of neuro-symbolic planners against specific types of non-stationarity (e.g., sudden shift vs. gradual drift)

Overview of some recent work

- **Act as You Learn: Adaptive MCTS (ADA-MCTS) (Luo et al. – AAMAS 2024):** Integrates Bayesian learning of environmental dynamics (inference) with symbolic tree search (planning), transitioning from pessimistic minimax search in unknown regions to optimistic MCTS as certainty increases.
- **Particle Filter for High-Fidelity Belief Representation ((Zhang et al. – NeurIPS 2025; ICAPS 2025) :** Reducing particle degeneration by using annealed importance sampling, bounded box reachability propagation, and Correlation and temporal consistency aware projections to prevent mode collapse. MCTS is then used to do planning.
- **Scalable Planning via Learned Abstraction (Luo et al. – ICLR 2025 (SpotLight)) :** Uses a Neural VQ-VAE to compress continuous dynamics into discrete latent codes, enabling Symbolic MCTS to search efficiently in the abstract space without suffering from the curse of dimensionality.



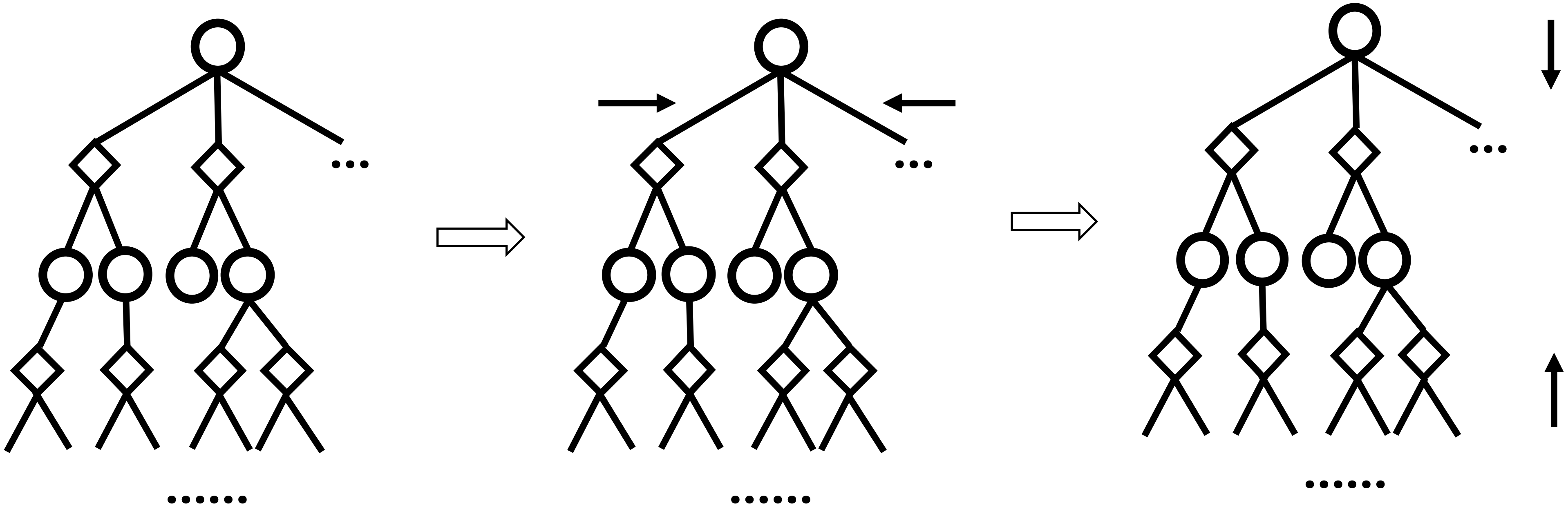
Scalable Decision-Making in Stochastic Environments through Learned Temporal Abstraction.

Baiting Luo et al. ICLR 2025 (Spotlight).

This work is supported by the **National Science Foundation, Defense Advance Research Projects Agency** and **US Air Force Research Lab**

Core Challenge: High Dimensionality

Challenge: Making decisions in high-dimensional continuous action spaces
- Action space is continuous and high dimensional.
- Environment randomness adds complexity.
- The planning horizon compounds the computational burden.
- Hypothesis: Can we compress the data (similar to macro actions) that helps with search – especially why data is available that can be used to learn transitions.



Macro actions[1] to counter long planning horizon

1. McGovern, A., & Sutton, R. S. (1998). *Macro-Actions in Reinforcement Learning: An Empirical Analysis*.

VQ Discretization: Encoder and State Conditioned Decoder

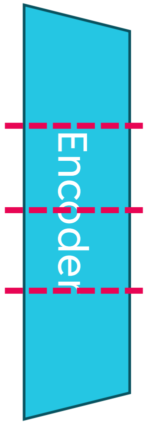
Vanilla discretization:

1. Lose granularity for decision makings.
2. The size of discretized action sets is still large and computationally challenging to iterate over.
3. Use Vector Quantised-Variational AutoEncoder (VQ-VAE)

$$x_1 = (s_1, a_2, a_3, a_4),\ x_2 = (s_2, a_2, a_2, a_8) \dashrightarrow z_e(x_1),\ z_e(x_2)$$

$$x_3 = (s_3, a_1, a_2, a_3),\ x_4 = (s_4, a_1, a_2, a_3) \dashrightarrow z_e(x_3),\ z_e(x_4)$$

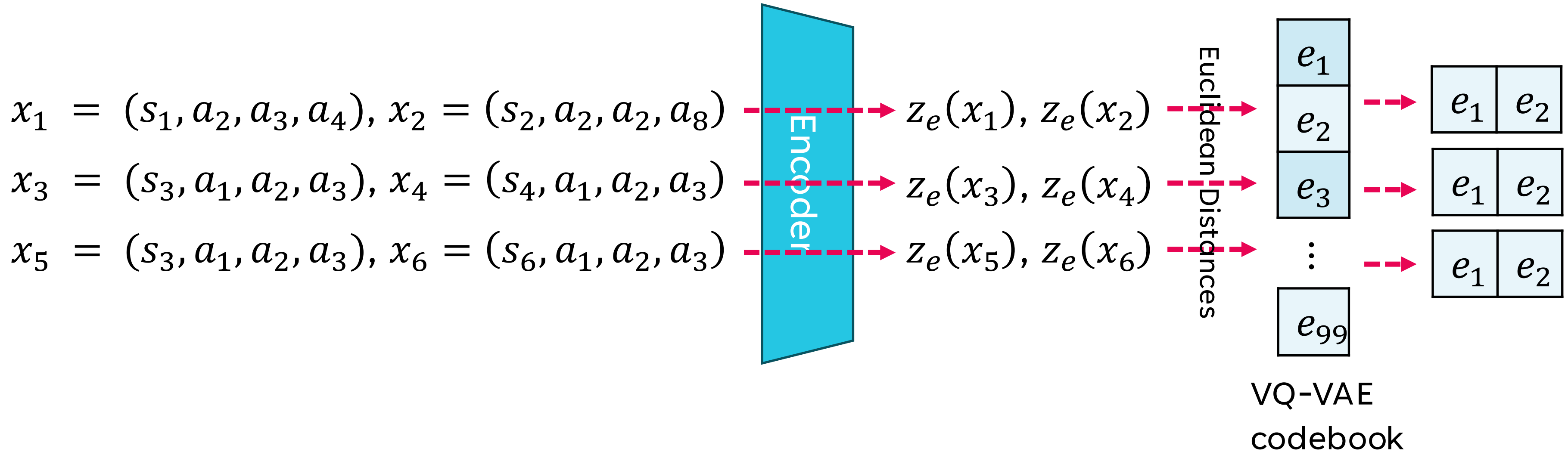
$$x_5 = (s_5, a_1, a_2, a_3),\ x_6 = (s_6, a_1, a_2, a_3) \dashrightarrow z_e(x_5),\ z_e(x_6)$$



- **Encoding**: During the training phase, the VQ-VAE[1] processes trajectory data and maps state-macro action pairs into discrete latent codes, while handling stochasticity of return estimates through a masking approach.
- Discrete Latent Space: The encoder maps the input data to a discrete latent space using a codebook, a collection of learnable vectors (codes). Each data point is represented by the closest vector in this codebook.
- State conditioned decoding and Reconstruction: The state conditioned decoder (using relatively few discrete actions while maintaining high granularity) then learns to reconstruct the original data from these latent codes.

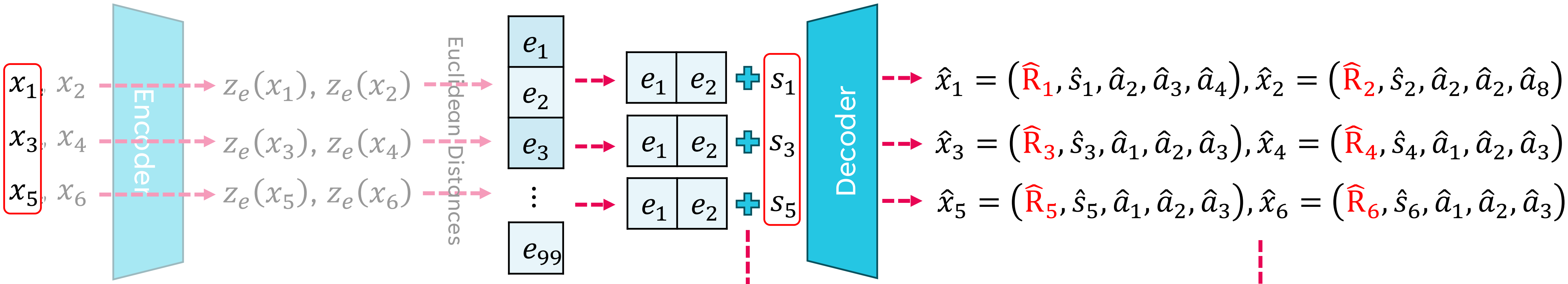
1. van den Oord, A., Vinyals, O., & Kavukcuoglu, K. (2017). *Neural Discrete Representation Learning*. In NeurIPS 2017.

VQ Discretization: Encoder and State Conditioned Decoder



- **Encoding**: During the training phase, the VQ-VAE processes trajectory data and maps state-macro action pairs into discrete latent codes, while handling stochasticity of return estimates through a masking approach.
- **Discrete Latent Space**: The encoder maps the input data to a discrete latent space using a codebook, a collection of learnable vectors (codes). Each data point is represented by the closest vector in this codebook.
- **State conditioned decoding and Reconstruction**: The state conditioned decoder (using relatively few discrete actions while maintaining high granularity) then learns to reconstruct the original data from these latent codes.

VQ Discretization: Encoder and State Conditioned Decoder



The key technique[1] to maintain granularity with a small codebook

We reconstructed return to go estimates here.

$x_7 = (-10, s_7, a_1, a_2, a_3), x_8 = (R_8, s_6, a_2, a_2, a_3)$ → e_7 e_{12}

$x_9 = (10, s_7, a_1, a_2, a_3), x_{10} = (R_{10}, s_3, a_2, a_2, a_3)$ → e_9 e_{11}

The same action would lead to very different outcomes but treated as different actions during planning!

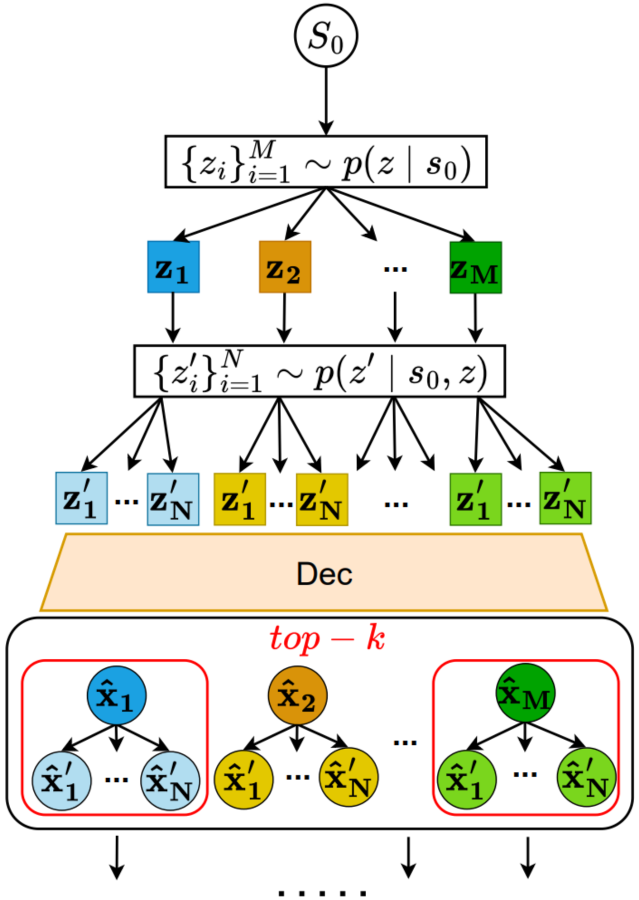
- **State conditioned decoding and Reconstruction**: The state conditioned decoder learns to reconstruct the original data from these latent codes.

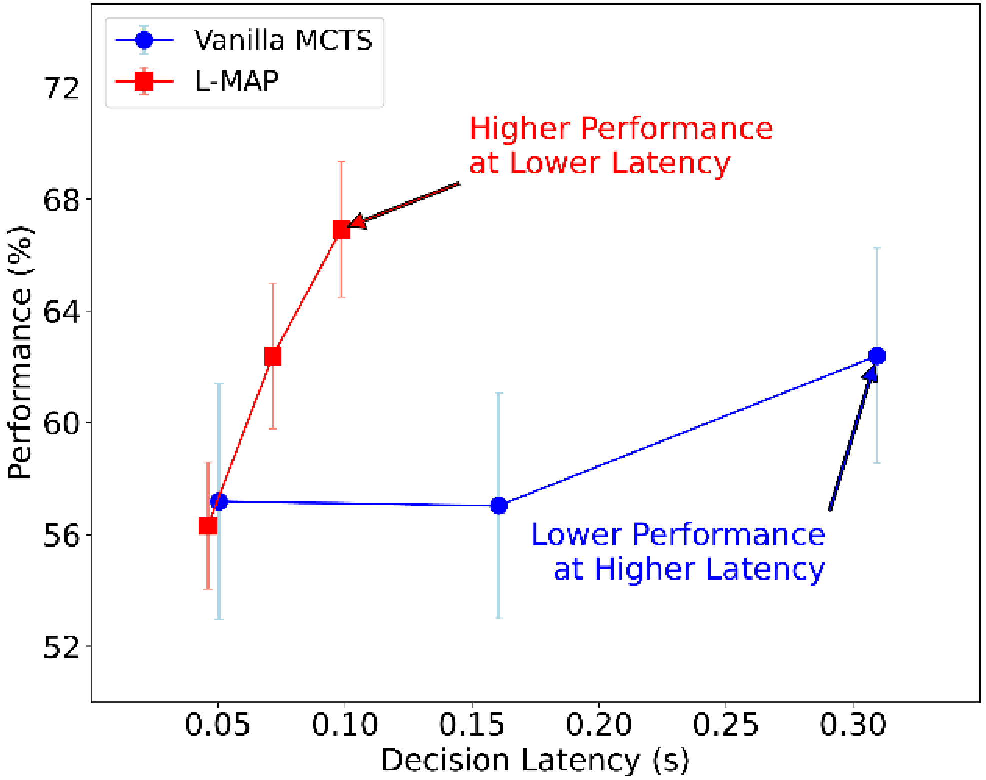
1. Jiang, Z., Zhang, T., Janner, M., Li, Y., Rocktäschel, T., Grefenstette, E., & Tian, Y. *Efficient Planning in a Compact Latent Action Space*. ICLR 2023.

Pre-Construct Latent Search Space: improve planning efficiency

However, even **a discrete codebook with a few hundred entries** for planning still falls short under tight time constraints.

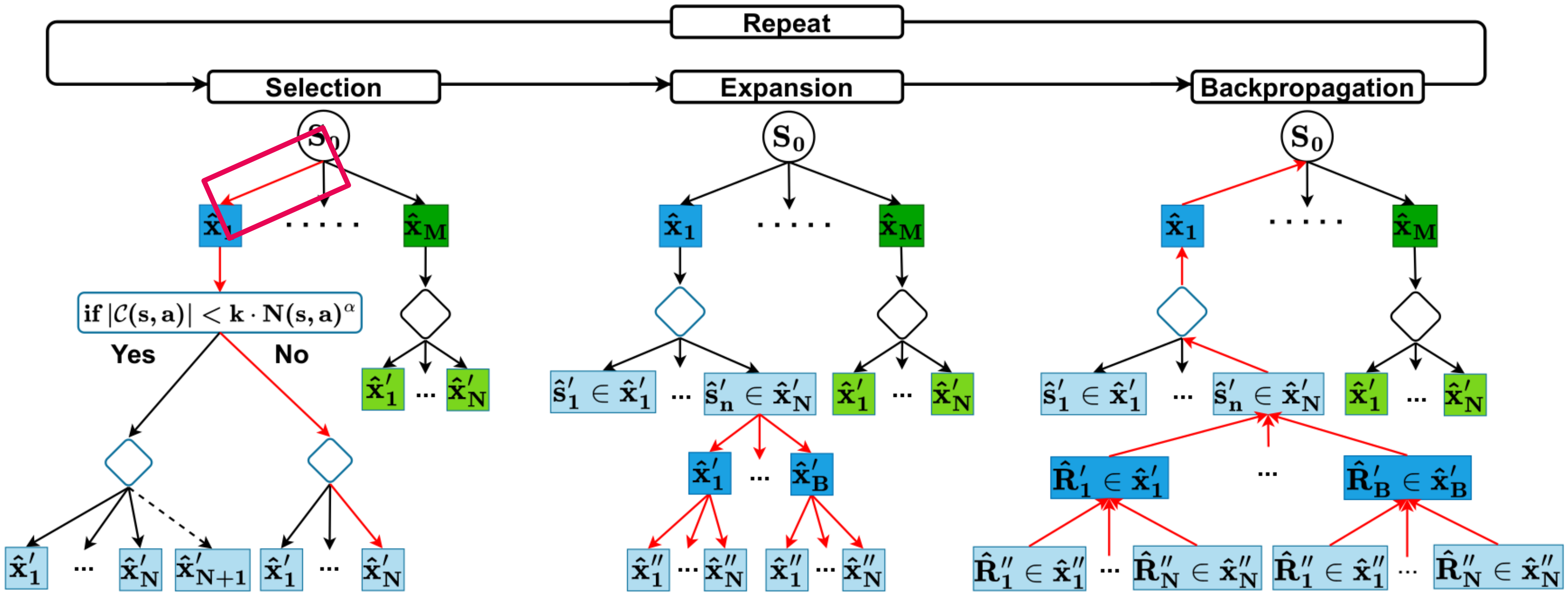
- **Sequence model:** A sequence model (we used causal Transformer but not limited to it) is trained to model sequences of latent codes. It serves two purposes in our framework:
  - Acts as a prior policy to sample promising macro-actions based on the current state
  - Functions as a transition model to predict next latent codes, which encode future states and returns





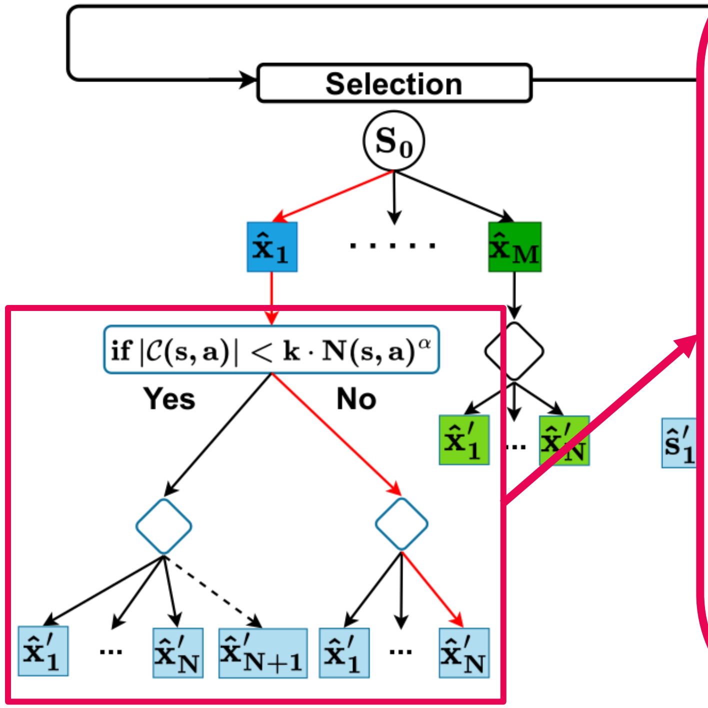
We sample multiple sequence and evaluate promising macro-actions in parallel

Finally we search with Monte Carlo Tree Search

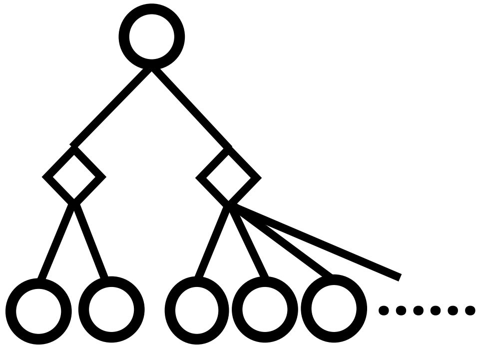


- Selection: Choose latent macro-actions using UCT[1] (Upper Confidence Bounds for Trees)
- Progressive Widening: Balances between exploring existing search space and adding new branches
- Expansion: Sample multiple potential macro-actions and their outcomes in parallel
- Backpropagation: Update estimated values based on reconstructed returns

Progressive widening is used to manage explosion



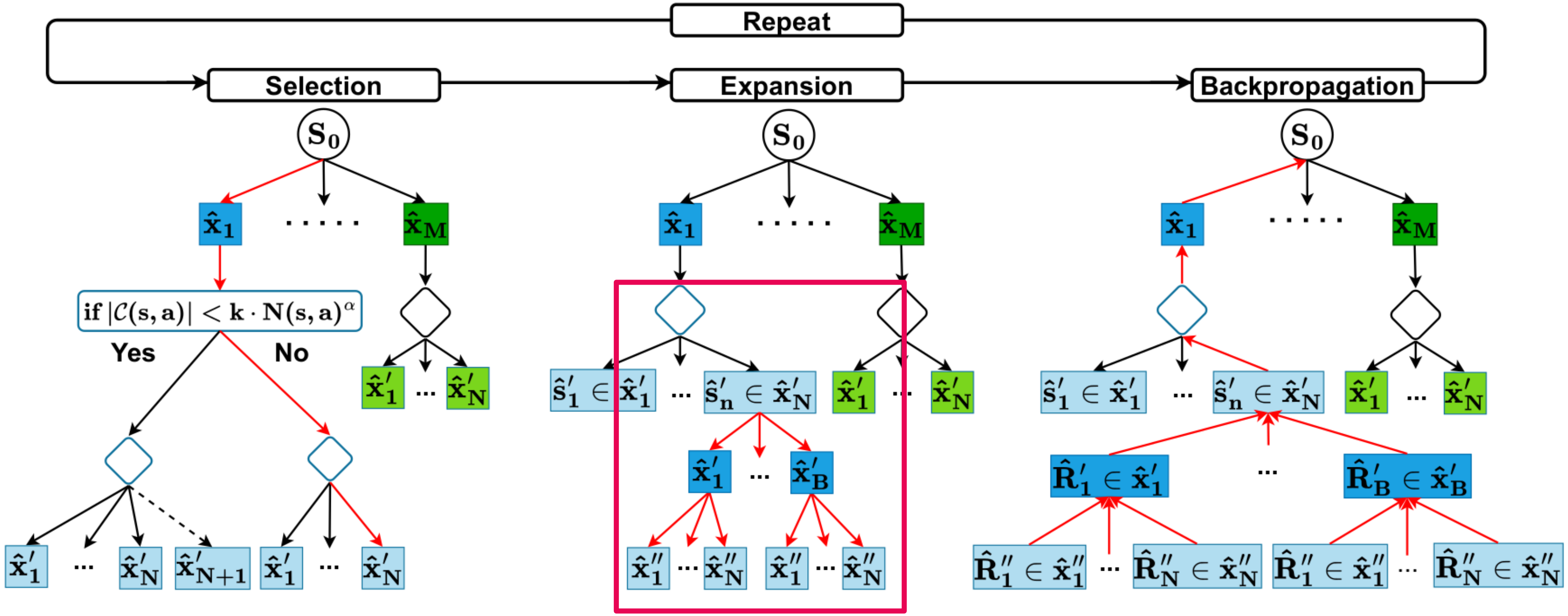
Why is progressive widening[1] necessary?



1. Without progressive widening, the branching factor explodes, the tree gets stuck at shallow depth.
2. The preconstructed search space is underutilized as the planner keeps exploring new branches.

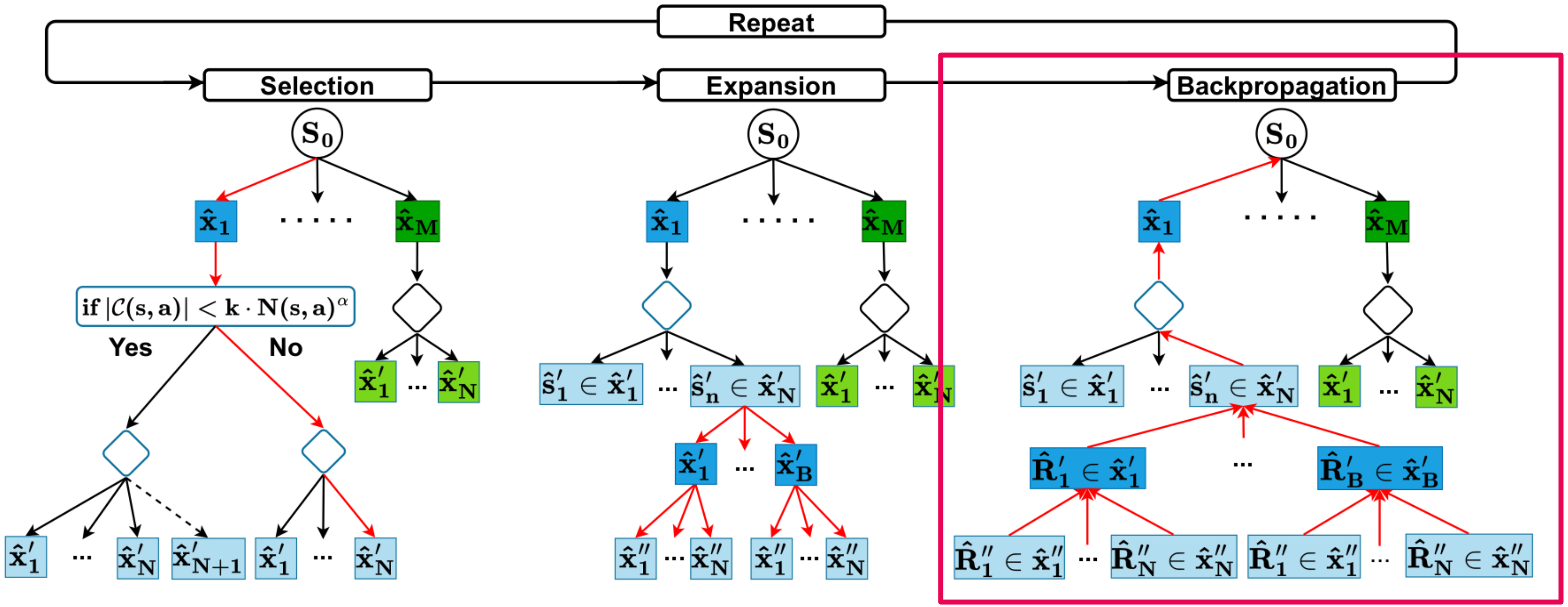
- Selection: Choose latent macro-actions using UCT (Upper Confidence Bounds for Trees)
- Progressive Widening: Balances between exploring existing search space and adding new branches
- Expansion: Sample multiple potential macro-actions and their outcomes in parallel
- Backpropagation: Update estimated values based on reconstructed returns

Expansion is done in parallel



- Selection: Choose latent macro-actions using UCT (Upper Confidence Bounds for Trees)
- Progressive Widening: Balances between exploring existing search space and adding new branches
- Expansion: Sample multiple potential macro-actions and their outcomes in parallel
- Backpropagation: Update estimated values based on reconstructed returns

Finally backpropagation and selection



- Selection: Choose latent macro-actions using UCT (Upper Confidence Bounds for Trees)
- Progressive Widening: Balances between exploring existing search space and adding new branches
- Expansion: Sample multiple potential macro-actions and their outcomes in parallel
- Backpropagation: Update estimated values based on reconstructed returns

L-MAP Results:

Handles stochastic continuous control with strong returns
– On Stochastic MuJoCo, **consistently higher normalized return** than prior model-based planners and generally better than strong model-free baselines.

Dataset collected with different qualities of behavior policies

Categories of methods

			Model-Based			Model-Free	
Dataset Type	**Env**	**L-MAP**	**TAP**	**TT**	**1R2R**	**CQL**	**IQL**
Medium-Expert-Mod	Hopper	106.11 ± 2.16	40.86 ± 5.42	56.10 ± 3.33	52.19 ± 8.37	**106.17 ± 2.16**	60.61 ± 3.46
Medium-Expert-Mod	Walker2D	**93.43 ± 1.41**	91.40 ± 1.42	80.93 ± 2.60	56.48 ± 7.51	91.44 ± 1.44	86.66 ± 1.84
Medium-Mod	Hopper	55.07 ± 3.06	43.64 ± 2.25	44.49 ± 2.47	**65.24 ± 3.31**	49.92 ± 3.00	56.00 ± 3.60
Medium-Mod	Walker2D	52.94 ± 1.57	44.46 ± 1.82	43.61 ± 2.15	**65.16 ± 2.84**	49.38 ± 2.02	48.82 ± 2.31
Medium-Replay-Mod	Hopper	**52.30 ± 2.65**	38.10 ± 3.22	37.85 ± 1.19	22.82 ± 2.08	40.53 ± 1.52	49.12 ± 3.38
Medium-Replay-Mod	Walker2D	51.44 ± 1.65	43.49 ± 2.27	27.43 ± 3.33	**52.23 ± 2.22**	40.24 ± 1.67	40.77 ± 2.72
Medium-Expert-High	Hopper	66.93 ± 3.46	37.31 ± 3.66	58.04 ± 3.60	37.99 ± 2.71	**68.03 ± 3.94**	44.83 ± 2.58
Medium-Expert-High	Walker2D	**97.18 ± 2.08**	91.09 ± 2.78	50.01 ± 3.51	32.38 ± 4.55	83.18 ± 3.70	68.61 ± 3.33
Medium-High	Hopper	**55.32 ± 3.56**	43.93 ± 2.66	41.26 ± 5.53	33.99 ± 0.92	45.21 ± 2.97	49.69 ± 2.47
Medium-High	Walker2D	**68.87 ± 2.21**	52.20 ± 2.76	59.84 ± 5.03	32.13 ± 4.51	61.49 ± 3.24	47.53 ± 3.05
Medium-Replay-High	Hopper	58.05 ± 3.36	48.69 ± 2.97	39.24 ± 2.16	**68.25 ± 3.78**	51.70 ± 3.09	43.27 ± 2.78
Medium-Replay-High	Walker2D	**65.87 ± 3.07**	55.15 ± 3.29	16.55 ± 2.17	65.63 ± 3.41	50.33 ± 3.88	45.13 ± 2.38
Mean		**68.63**	52.53	46.28	48.71	61.47	53.42

In general, L-MAP produces higher mean value

Beam search is not sufficient for stochastic environment

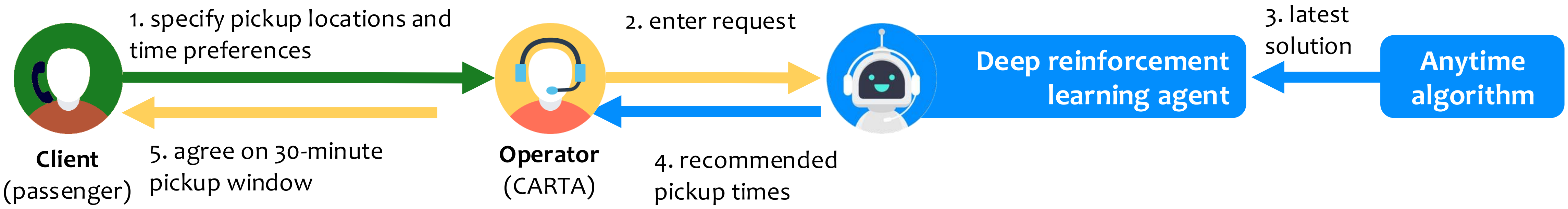
L-MAP

Without temporal abstraction

Real-world Applications

This work is supported by the **National Science Foundation, Defense Advance Research Projects Agency** and **US Air Force Research Lab**

Negotiated Pick up and Delivery Process -- Example



Smart EV charging positively impacts on buildings & power grids

VEHICLE-TO-GRID

Why Vehicle-to-Grid?

High Variable Costs

- **Time-of-Use rates:** Energy usage prices depend on time of day (high during afternoon).
- **Demand Charges:** A fee on the *highest power drawn* in a billing cycle (>30% of bill).

EV as Battery Asset

- A typical EV has ~60 kWh battery capacity.
- With bidirectional chargers, each parked EV becomes an **energy reserve**

User requirements

- Each user has a **Required SoC** and **Requested Departure.**
- To ensure reliability, the system must **meet these requirements** precisely.

Vehicle-to-X: A Spectrum of Opportunities

V2B

Building peak shaving

V2H

Home backup power

V2G

Grid frequency regulation

How Our Work Advances the SOTA

Prior SOTA   Our Work

Differentiating Dimension	Prior SOTA*
Algorithm paradigm *Core method for action selection*	LP / MIP offline, static
Domain-aware training *MILP or domain heuristics guide learning*	✗ pure solver / RL
Online EV arrival uncertainty *Decisions without knowing future arrivals*	~ forecast-based
Cross-day coordination *Actions today affect next day's cost*	✗ greedy / daily
Heterogeneous charger fleet *Mix of uni- & bidirectional, varying power rates*	✗ Homogeneous assumed
Multi-day billing horizon *Explicit 30-day demand charge optimization*	~ day-ahead only
Formal incentive guarantees *Strategy-proof & budget-feasible mechanism*	✗

How Our Work Advances the SOTA

Prior SOTA | Our Work

Differentiating Dimension	Prior SOTA*	RL-V2B AAMAS '25
Algorithm paradigm *Core method for action selection*	LP / MIP offline, static	DDPG continuous RL
Domain-aware training *MILP or domain heuristics guide learning*	✗ pure solver / RL	✓ MILP policy guidance
Online EV arrival uncertainty *Decisions without knowing future arrivals*	~ forecast-based	~ model-free RL
Cross-day coordination *Actions today affect next day's cost*	✗ greedy / daily	✗ no explicit model
Heterogeneous charger fleet *Mix of uni- & bidirectional, varying power rates*	✗ Homogeneous assumed	✓ Multiple types, rates
Multi-day billing horizon *Explicit 30-day demand charge optimization*	~ day-ahead only	~ monthly reward
Formal incentive guarantees *Strategy-proof & budget-feasible mechanism*	✗	✗

How Our Work Advances the SOTA

Prior SOTA | Our Work

Our Works →

Differentiating Dimension	Prior SOTA*	RL-V2B AAMAS '25	DG-MCTS ICCPS '25
Algorithm paradigm *Core method for action selection*	LP / MIP offline, static	DDPG continuous RL	DG-MCTS online tree search
Domain-aware training *MILP or domain heuristics guide learning*	✗ pure solver / RL	✓ MILP policy guidance	✓ domain-pruned tree
Online EV arrival uncertainty *Decisions without knowing future arrivals*	~ forecast-based	~ model-free RL	✓ online w/ pruning
Cross-day coordination *Actions today affect next day's cost*	✗ greedy / daily	✗ no explicit model	✗ no explicit model
Heterogeneous charger fleet *Mix of uni- & bidirectional, varying power rates*	✗ Homogeneous assumed	✓ Multiple types, rates	✓ Multiple types, rates
Multi-day billing horizon *Explicit 30-day demand charge optimization*	~ day-ahead only	~ monthly reward	~ monthly reward
Formal incentive guarantees *Strategy-proof & budget-feasible mechanism*	✗	✗	✗

Institute for Software Integrated Systems

Papers available at scopelab.ai

VANDERBILT UNIVERSITY

How Our Work Advances the SOTA

Prior SOTA | Our Work

		Our Works →		
Differentiating Dimension	**Prior SOTA***	**RL-V2B AAMAS '25**	**DG-MCTS ICCPS '25**	**P-V2B HSCC '26**
Algorithm paradigm *Core method for action selection*	LP / MIP offline, static	DDPG continuous RL	DG-MCTS online tree search	MC-MPC + VF neuro-symbolic
Domain-aware training *MILP or domain heuristics guide learning*	✗ pure solver / RL	✓ MILP policy guidance	✓ domain-pruned tree	✓ MC sampling + NN
Online EV arrival uncertainty *Decisions without knowing future arrivals*	~ forecast-based	~ model-free RL	✓ online w/ pruning	✓ MC over futures
Cross-day coordination *Actions today affect next day's cost*	✗ greedy / daily	✗ no explicit model	✗ no explicit model	✓ Long term value function
Heterogeneous charger fleet *Mix of uni- & bidirectional, varying power rates*	✗ Homogeneous assumed	✓ Multiple types, rates	✓ Multiple types, rates	✓ Multiple types, rates
Multi-day billing horizon *Explicit 30-day demand charge optimization*	~ day-ahead only	~ monthly reward	~ monthly reward	✓ cross-day value fn
Formal incentive guarantees *Strategy-proof & budget-feasible mechanism*	✗	✗	✗	✗

How Our Work Advances the SOTA

Prior SOTA | Our Work

Differentiating Dimension	Prior SOTA*	RL-V2B AAMAS '25	DG-MCTS ICCPS '25	P-V2B HSCC '26	CONSENT AAMAS '26
		Our Works →			
Algorithm paradigm *Core method for action selection*	LP / MIP offline, static	DDPG continuous RL	DG-MCTS online tree search	MC-MPC + VF neuro-symbolic	MC-MPC + Negotiation mechanism design
Domain-aware training *MILP or domain heuristics guide learning*	✗ pure solver / RL	✓ MILP policy guidance	✓ domain-pruned tree	✓ MC sampling + NN	~ MPC-structured
Online EV arrival uncertainty *Decisions without knowing future arrivals*	~ forecast-based	~ model-free RL	✓ online w/ pruning	✓ MC over futures	✓ realtime MPC
Cross-day coordination *Actions today affect next day's cost*	✗ greedy / daily	✗ no explicit model	✗ no explicit model	✓ Long term value function	✗
Heterogeneous charger fleet *Mix of uni- & bidirectional, varying power rates*	✗ Homogeneous assumed	✓ Multiple types, rates	✓ Multiple types, rates	✓ Multiple types, rates	✓ Multiple types, rates
Multi-day billing horizon *Explicit 30-day demand charge optimization*	~ day-ahead only	~ monthly reward	~ monthly reward	✓ cross-day value fn	✓ cross-day via MPC
Formal incentive guarantees *Strategy-proof & budget-feasible mechanism*	✗	✗	✗	✗	✓ proven theoretical guarantees

OPTIMUS: The Simulation Foundation

IEEE SmartComp 2024 • Talusan, Sen, Pettet et al.[1]

What is OPTIMUS?

A **discrete-event simulation** platform built on real Nissan data that provides a shared sandbox for benchmarking V2B optimization algorithms.

- Generative models from real EV & building data
- Modular: swap policies, charger configs, grid events
- Built-in MILP, MCTS, RL, and greedy solvers
- Web dashboard for non-technical users
- API for real-time charger integration

Architecture

Data Collection
↓
Generative Models
↓
Policy Engine
↓
Dashboard UI + API

[1] *OPTIMUS: Discrete Event Simulator for Vehicle-to-Building Charging Optimization.* SMARTCOMP 2024, J.P. Talusan, R. Sen, A. Pettet, A. Kandel, Y. Suzue, L. Pedersen, A. Mukhopadhyay, A. Dubey (2024).– Simulator to evaluate V2B optimization algorithms at scale.

Paper Link

Reinforcement Learning for V2B

AAMAS 2025 • Liu, Sen, Talusan et al.[5] • Best Paper Nominated • Acceptance: 24.5%

Enhanced DDPG Framework

- **Handles sparse rewards** arriving at end of monthly billing period
- **Action masking:** domain-specific constraints ensure feasible charger actions
- **MILP-driven policy guidance** during training for faster convergence
- **Handles heterogeneity** in chargers (uni- & bidirectional) with varying rates
- **Validated on 9 months** of real Nissan operational data

Key Results

Significant cost savings (>4%)

over heuristic and traditional RL baselines across all months, meeting 100% of EV charging requirements

Scalable approach

One of the first **general, scalable solutions** for V2B energy management: handles real-world heterogeneity and long-term reward horizons

[5] Reinforcement Learning-based Approach for Vehicle-to-Building Charging with Heterogeneous Agents and Long Term Rewards. AAMAS 2025, F. Liu, R. Sen, J.P. Talusan, A. Pettet, A. Kandel, Y. Suzue, A. Mukhopadhyay, A. Dubey

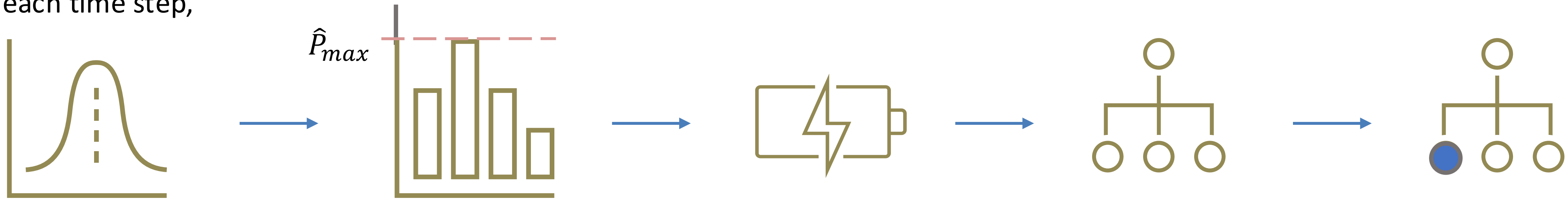
Paper Link

Online Decisions under Uncertainty: DG-MCTS

Handles Uncertainty, Plans Non-Myopically, Provides near-optimal actions, Scales Efficiently

IEEE/ACM ICCPS 2025 • Sen, Zhang, Liu et al.[6] • Acceptance: 26%

At each time step,



1. Gather exploration samples (EVs and building loads)

2. **Estimate** peak power (using MILP in exploration samples).

3. **Pruned actions** (domain heuristics + power gap actions)

4. MCTS **evaluates actions**, simulates with trickle charging policy.

5. **Select best action** from averaged root-parallel trees.

Extension: **Decentralized DG-MCTS** scales linearly, for small performance tradeoff

Key Results:

- DG-MCTS **lowest in 6 of 8 months** (~3 % cheaper than RL)
- Lower price increase under more uncertainty (building load & EV user behavior)
- SoC requirements are kept flexible

[6] Online Decision-Making Under Uncertainty for Vehicle-to-Building Systems. ICCPS 2025. R. Sen, Y. Zhang, F. Liu, J.P. Talusan, A. Pettet, Y. Suzue, A. Mukhopadhyay, A. Dubey

P-V2B: Neuro-Symbolic Control with Persistence

IEEE/ACM ICCPS 2025 • Sen, Liu, Talusan et al.[7.] • Acceptance 25%

Key Insight:
- Employees return daily, exploit this **persistence** to plan multi-day charging strategies.
- **Enables Over-charges** during low demand, enabling deeper discharge during peak days.

MC-MPC Layer *(Short-horizon symbolic optimizer)*

- **Monte Carlo sampling** over N samples of EVs & building load
- Solves MILP per step across N samples, till end of day
- Embeds neural VF via PWL for tractable real-time actions
- Guarantees feasibility and all user SoC requirements

Neural Value Function (VF) *(Long-horizon learned predictor)*

- Input: {day, overcharge user clusters} at end of day
- Predicts monthly peak power; trained offline on MILP
- Online fine-tuning via replay buffer; surrogate aligns VF with MC-MPC behavior over time
- Enables cross-day coordination (proactive overcharging) that no single-day solutions can achieve

Annual Performance

Policy	Peak (kW)	Bill ($/mo)	vs Ours
MC-MPC+VF (Ours)	**156 ± 3**	**$10,238**	—
D-RL (DDPG)	171 ± 9	$10,392	**+1.5%**
D-MCTS	173 ± 4	$10,439	**+2.0%**
D-LLF heuristic	184 ± 4	$10,557	**+3.1%**
Fast Charging	199 ± 4	$10,732	**+4.8%**

Formal Guarantee

The persistent formulation provably achieves lower or equal cost than any daily-decomposed policy

[7] Sen, Rishav, et al. "P-V2B: A Neuro-Symbolic Framework for Leveraging User Persistence in Vehicle-to-Building Charging." *Accepted to ICCPS* (2026).

Paper Link

CONSENT: Human-Centered Negotiation

Moving beyond pure optimization: negotiating with real humans to align building and driver incentives.

AAMAS 2026 • Sen, Liu, Talusan et al.[8] • Acceptance: 25%

Novelty: Option generation & Cost evaluation for options

Features:

- Explicitly models **user flexibility** (departure time, SoC).
- **Reward heterogeneity** in user flexibility (minor deviations in SoC, departure times).

Solution:

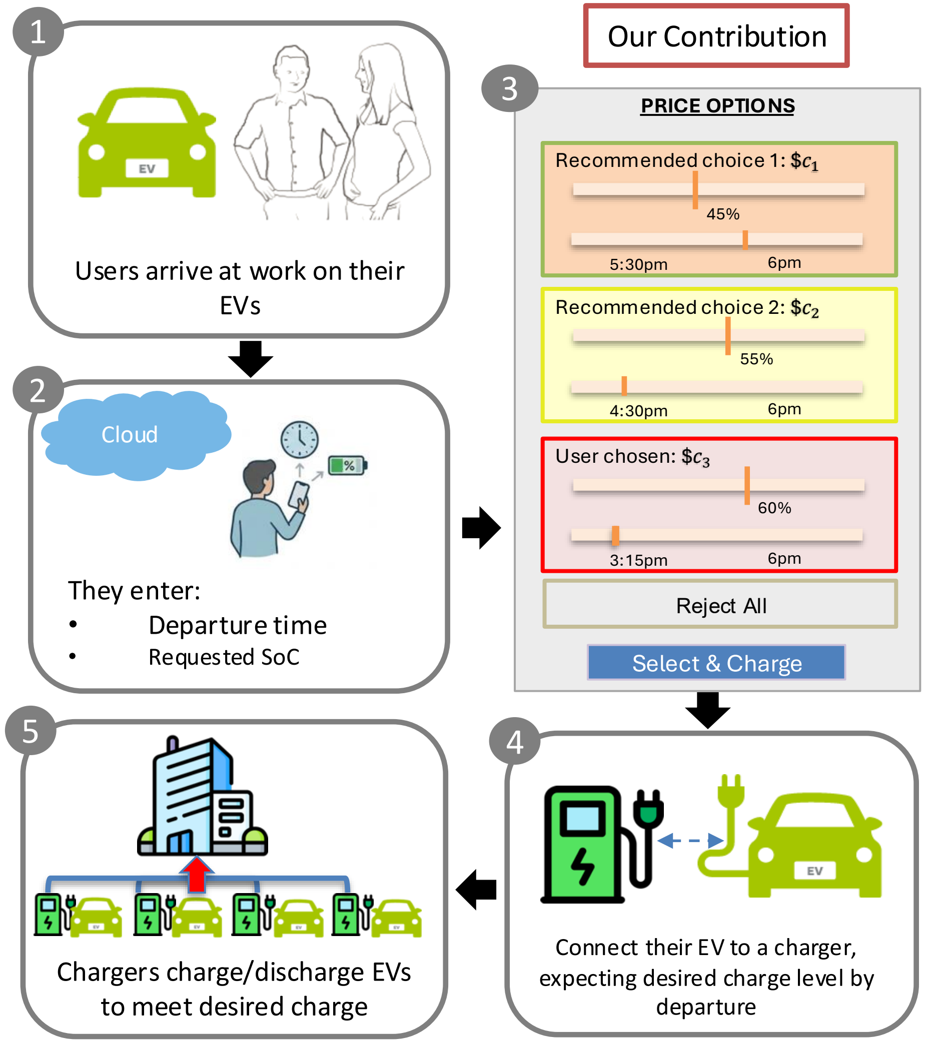
- Stochastic MPC charging & Negotiation policies (interdependent)

Guarantees:

- Strategy-proof (cannot be gamed)
- Budget-feasible (incentives are capped at expected savings)
- Voluntary participation (Every user benefits from participating)

Results:

- Operator cost ↓ **~6%**; User cost ↓ **~22%**.
- Sustains **~78% participation** of test users.



[8] Sen, Rishav, et al. "CONSENT: A Negotiation Framework for Leveraging User Flexibility in Vehicle-to-Building Charging under Uncertainty." *Accepted to AAMAS* (2026).

Institute for Software Integrated Systems

VANDERBILT UNIVERSITY

What's Next

PRIMARY OPEN PROBLEMS

01

Grid Integration

V2G Aggregation & Demand Response

- **Single-building DR signals**
  Respond to utility demand response events in real time while guaranteeing user SoC: no existing framework handles persistent users under DR.
- **Multi-building aggregation**
  Coordinate fleets across buildings as a virtual power plant; requires decentralized control that preserves cross-day coupling and fairness.
- **Multi-stakeholder incentives**
  Utility, operator, and user objectives conflict: mechanism design for three-way incentive alignment is formally open.

02

AI / Forecasting

Foundation Models for V2X Forecasting

- **Zero-shot arrival & load prediction**
  LLMs and time-series foundation models (e.g., TimesFM, Moirai) could generalize across buildings without site-specific retraining.
- **Embedding behavioral persistence**
  Current models treat time series as stationary; encoding user identity across sessions is an open architectural challenge.
- **Forecast-to-control pipeline**
  Tight integration of foundation model outputs into MC-MPC scenario sets — calibration and uncertainty quantification remain unsolved.

SUPPORTING CHALLENGES

- **Sim-to-Real Gap**
- **Standards & Adoption**
- **Battery Degradation**
- **Open Benchmark Dataset**

NS-Gym: Benchmarking Decisions Under Non-Stationarity

Four Key Questions

1 What changes?

Transition dynamics, rewards, observations

2 How does it change?

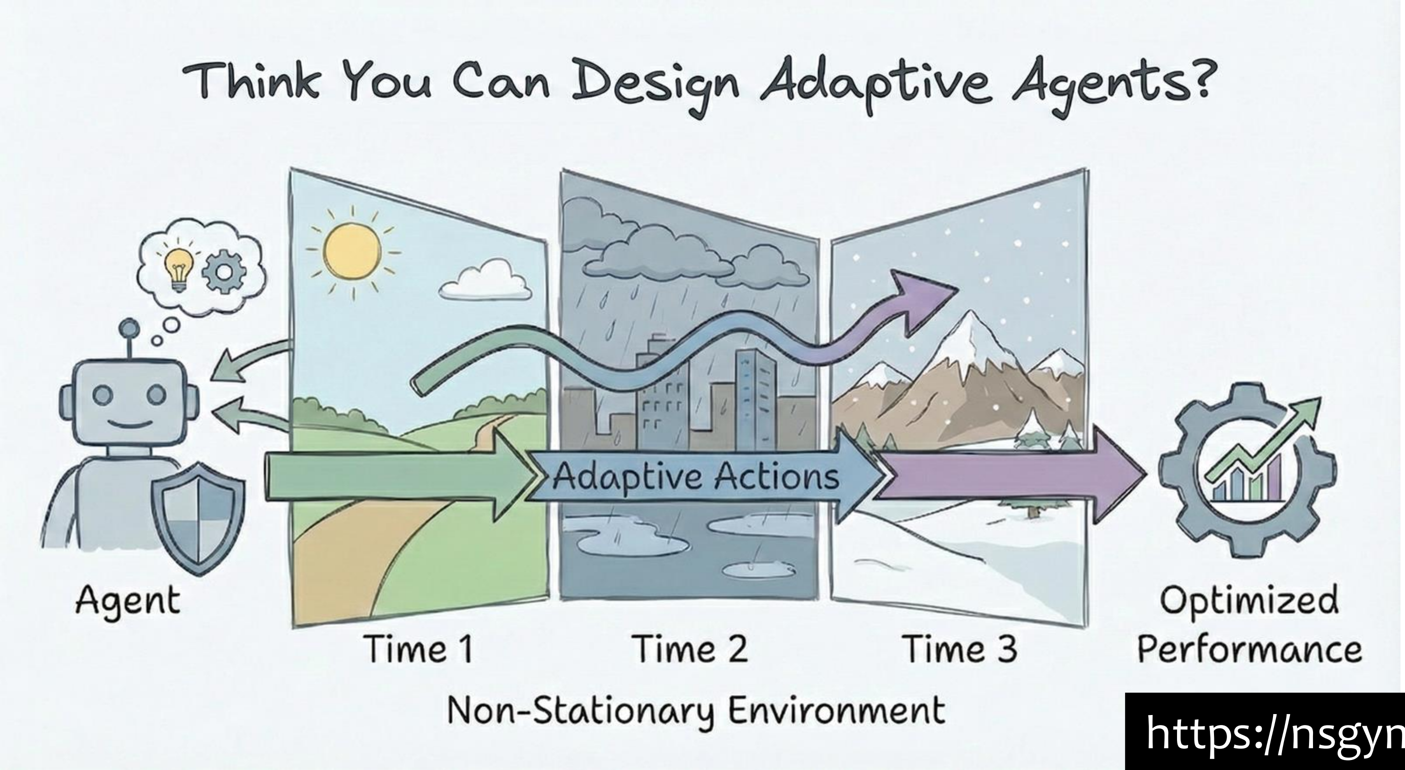
Gradual drift, abrupt shifts, periodic cycles

3 Does the agent detect it?

Runtime monitoring and change detection

4 Does the agent know it?

Notification vs. unaware adaptation



Transport Applications

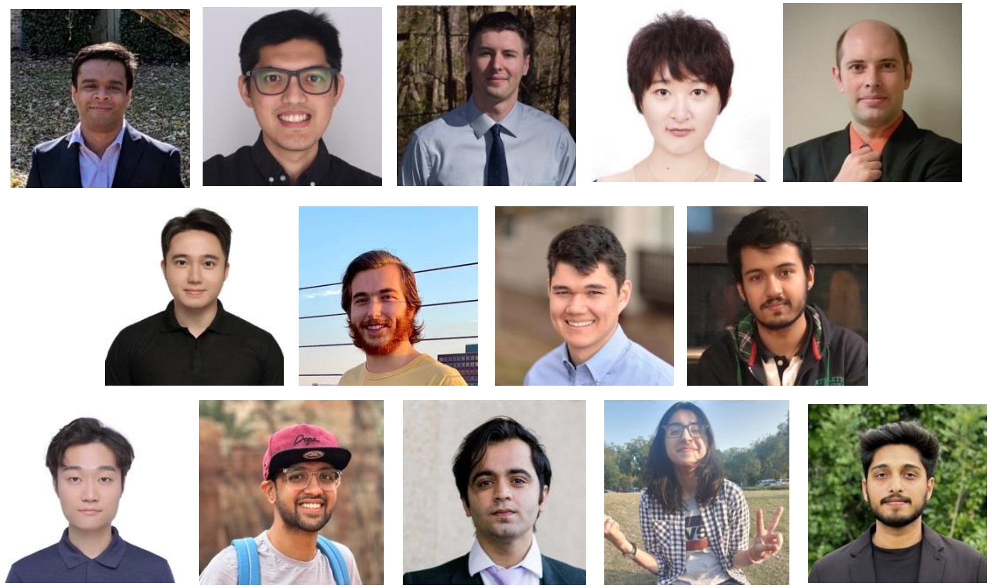
- Transit headway under demand drift
- EV charging with stochastic arrivals
- Emergency dispatch under load shifts
- School bus routing under disruptions

Hosting AAMAS 2026 Competition and CPS Week Tutorial - Evaluating Decision Agents under Non-Stationarity

https://nsgym.io/aamas2026_competition.html

Scopelab.ai Team and Collaborators

Current Team and a non-exhaustive list of partners and collaborators



Ayan Mukhopadhyay, JP Talusan, David Rogers, Fangqi Liu, Michael Yuhas
Baiting Luo, Jacob Buckelew, Nathaniel Keplinger, Rishav Sen
Yunuo Zhang, Samir Gupta, Ammar Zulqarnain, Agrima Khana, Vakul Nath

Cornell University, Pennsylvania State University, Nashville Metro Office,
Williamson County School District
WeGo Public Transit, Chattanooga Area Regional Transportation Authority
Nissan, Nashville Metro ITS, Nashville Fire Deparment