Multiobjective Deep Reinforcement Learning for Grid-Interactive Energy-Efficient Buildings (MODRLC)

National Renewable Energy Laboratory (NREL), University of Colorado Boulder (CU Boulder), QCoefficient Inc.
PI: Andrey Bernstein, Senior Researcher/Group Manager (NREL)
Andrey.Bernstein@nrel.gov

Project Summary

Timeline:

Start date: July 1st, 2019

Planned end date: June 30th, 2022

Key Milestones

1. Proof of concept in small-scale simulation environment.; June 30th, 2020
2. Proof of concept in large-scale HPC simulation; June 30th, 2021
3. Demonstration in real commercial building; June 30th, 2022

Budget:

Total Project $ to Date:

- DOE: $714k
- Cost Share: $120k

Total Project $:

- DOE: $1.5M
- Cost Share: $375k

Key Partners:

NREL
CU Boulder
QCoefficient Inc

Project Outcome:

- Learning-based building controllers scalable to buildings of different sizes and types without building-by-building customization.
- Avoid the expertise and cost of detailed engineering models.
- Optimally balance building-centric, grid-serving, and resiliency-oriented objectives.
- Enable the vision of grid-interactive efficient buildings (GEBs), enhancing renewables integration, decarbonization, and equity.

Team

NREL
NATIONAL RENEWABLE ENERGY LABORATORY

Energy systems optimization and control:
- Dr. Andrey Bernstein; PI, senior researcher/group manager
- Dr. Yue Chen; control systems researcher

Commercial and residential buildings modelling and operation:
- Dr. Xin Jin; senior researcher
- Dr. Rohit Chintala, researcher

Computational science and machine learning:
- Dr. Peter Graf, principal researcher
- Dr. Xiangyu Zhang, researcher

University of Colorado Boulder

Energy systems optimization and control:
- Prof. Emilano Dall'Anese
- Ana Ospina

Commercial and residential buildings modelling and operation:
- Prof. Gregor Henze
- Dr. Thibault Marzullo

QCoefficient

Vince Cushing, CTO

Challenge and Approach

Buildings are Major Player in Energy Sector

Residential and commercial buildings accounted for at least 28% of total U.S. end-use energy consumption in 2019.
(U.S. Energy Information Administration)

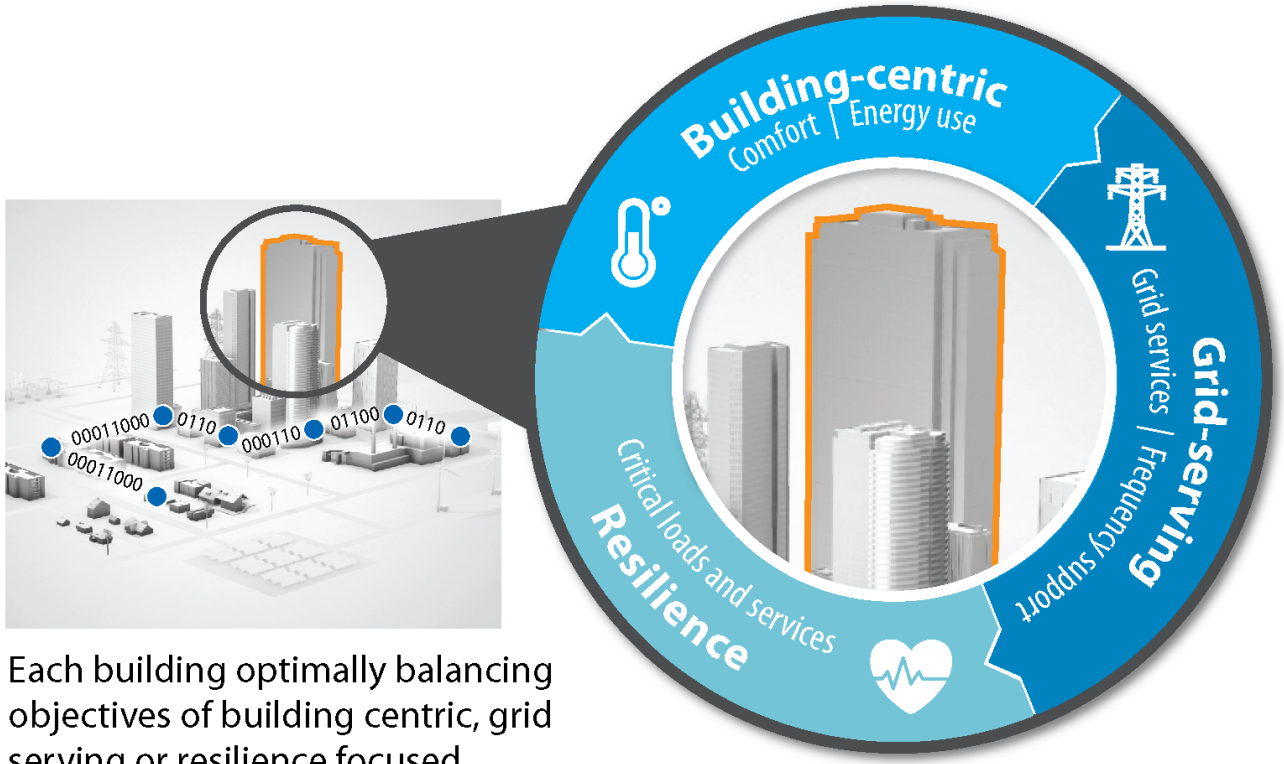
Commercial buildings could save almost one-third in annual energy costs by using smart management with modern sensors and controls.
(BTO Sensor and Control Technologies R&D Overview, 2018)

We Need Building Controls Help Meet Multiple Critical Objectives!

Building-centric and People-centric
Reduce costs and energy losses, increase comfort, support equity

Grid-serving
Provide grid services at multiple time scales, supporting renewables integration and decarbonization

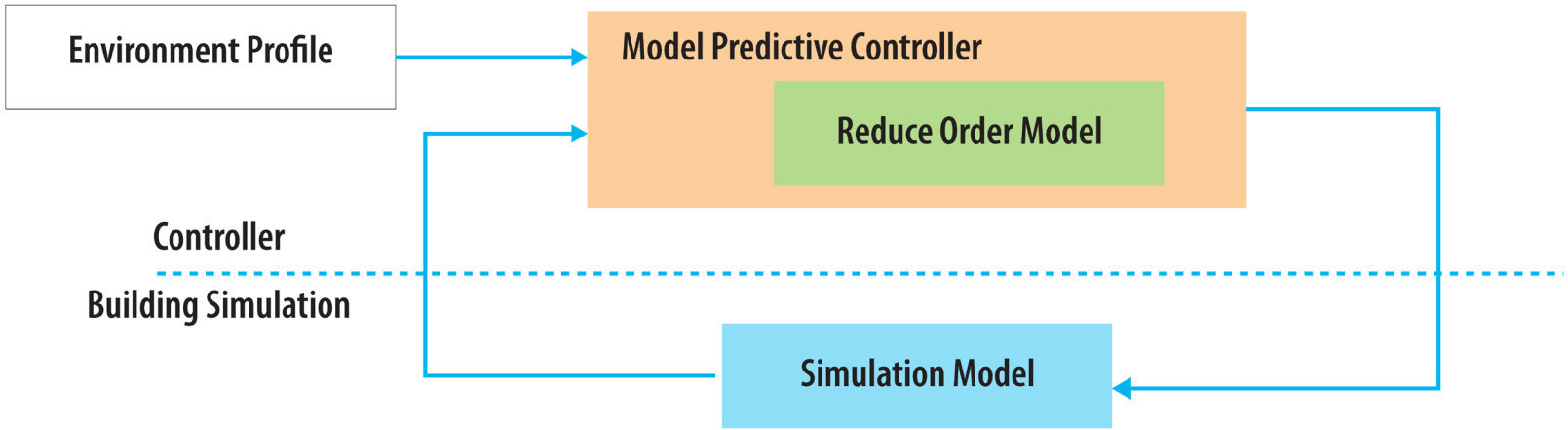
Resilience-centric
Ensure building survivability during natural disasters



Each building optimally balancing objectives of building centric, grid serving or resilience focused operation

Every Building Needs a Model

To control a building for interactivity, efficiency, and grid services, we need to know how the building operates. In other words, **we need to know its model**.

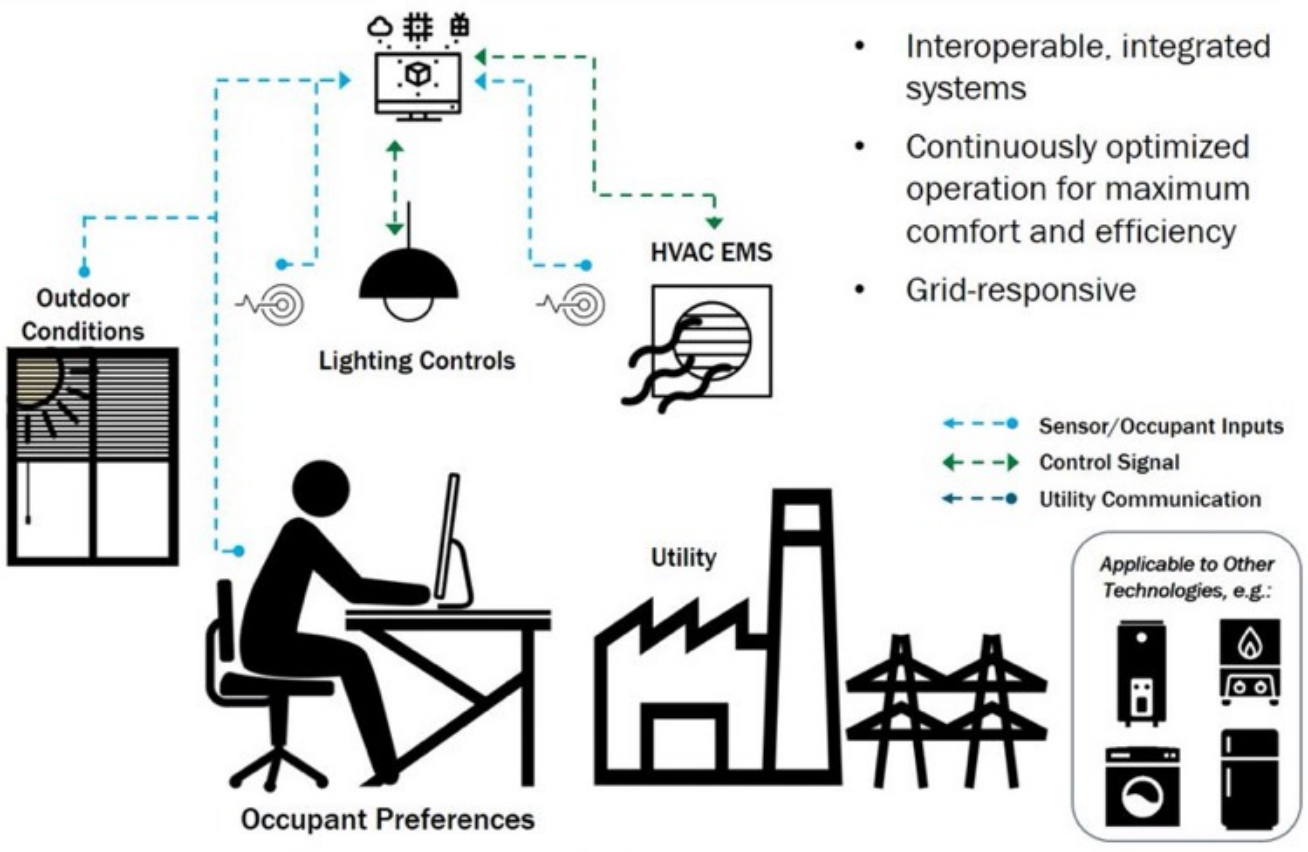


The standard approach uses a high-detail testing model, and a reduced-size model for the controller. This is **Model Predictive Control**.

But Every Building Tells a Different Story…

Buildings have different

- Occupancy patterns
- Hardware
- Grid connections
- Comfort preferences
- Etc.



- Interoperable, integrated systems
- Continuously optimized operation for maximum comfort and efficiency
- Grid-responsive

This fact makes high-detail building models difficult and time-consuming to develop

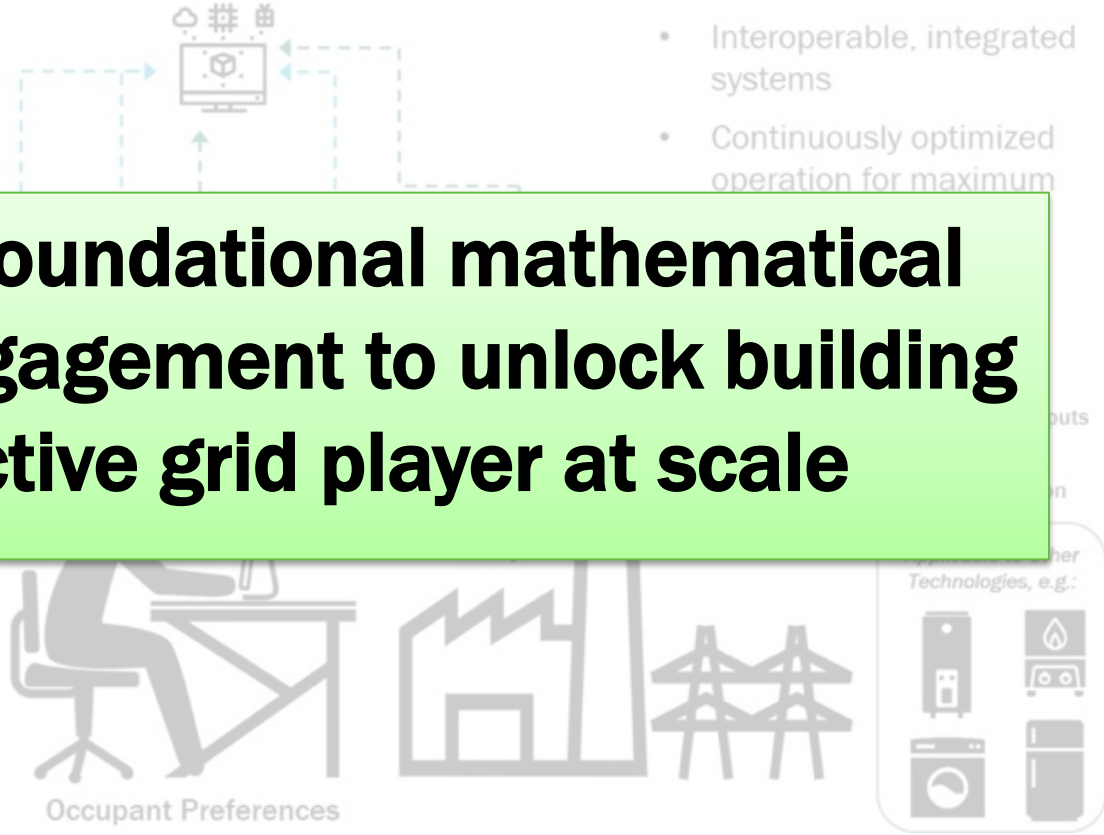
But Every Building Tells a Different Story…

Buildings have different

- Occupancy patterns

- Interoperable, integrated systems
- Continuously optimized operation for maximum

We need a combination of foundational mathematical research and stakeholder engagement to unlock building capabilities as an interactive grid player at scale



This fact makes high-detail building models difficult and time-consuming to develop

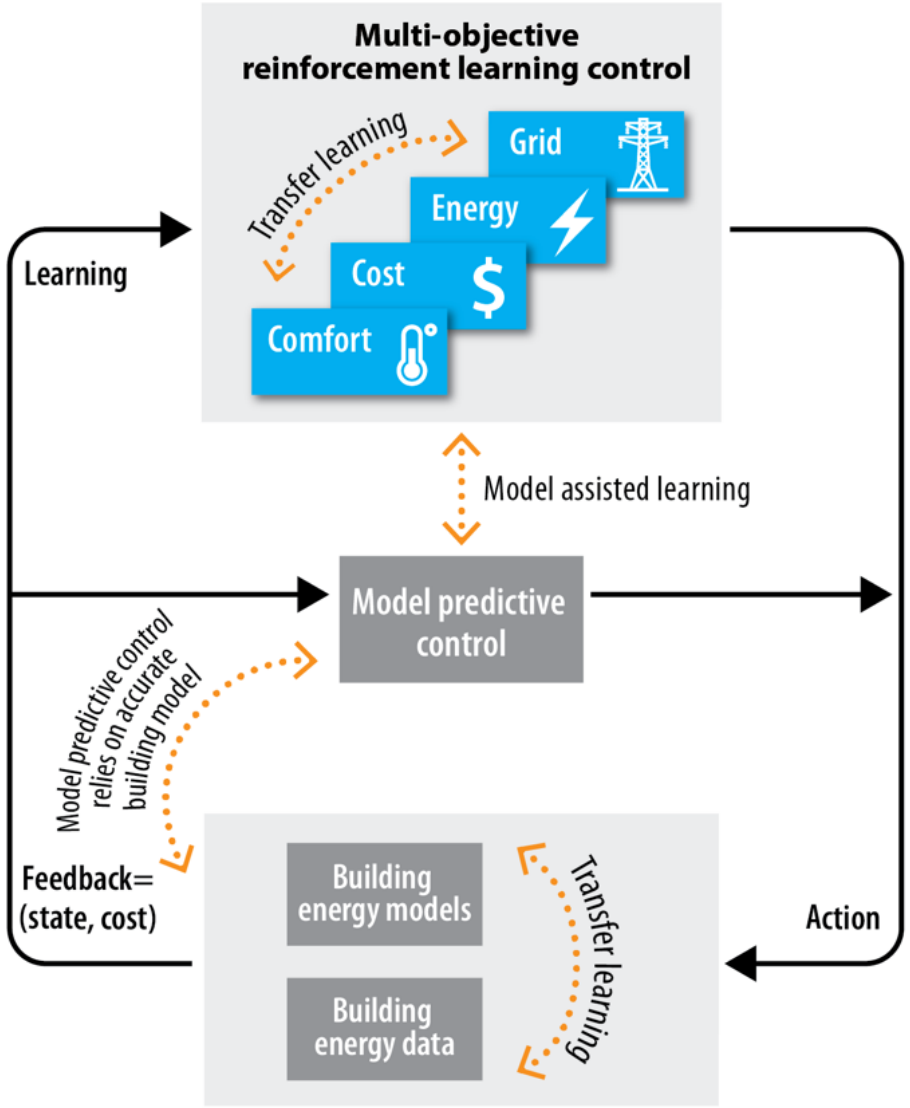
A Solution to Building-by-Building Customization

"Multi-objective deep reinforcement learning control (MODRLC)"

- We can **listen** to buildings using sensor networks and **learn** building patterns with advanced machine learning (ML).
- Unlike detailed models created by engineers, **ML-based approaches are scalable and adaptable to a wide range of buildings.**

Multi-Objective Deep Reinforcement Learning Control (MODRLC)

A balance of machine learning and model predictive control can efficiently and optimally control any building



Mathematical Formulation: Optimal Control

$$\min_{\{u_k\}_{k=0}^{K-1}} \quad \sum_{k=0}^{K-1} C(x_k, u_k) + V(x_K) \qquad \text{(MPC1)}$$

$$\text{s.to:} \quad x_{k+1} = f(x_k, u_k), \quad k \geq 0 \qquad \text{(MPC2)}$$

$$u_k \in \mathcal{U}, \quad k \geq 0 \qquad \text{(MPC3)}$$

$$x_k \in \mathcal{X}, \quad k \geq 0 \qquad \text{(MPC4)}$$

x – state of the building
u – control action
$C(x, u)$ – cost for action u in state x
K – control horizon
$V(x)$ – final cost
$f(x, u)$ – building's dynamical model
$\mathcal{U}, \mathcal{X}$ – constraints on action and state

Standard Approach: Model Predictive Control

$$\min_{\{u_k\}_{k=0}^{K-1}} \quad \sum_{k=0}^{K-1} C(x_k, u_k) + V(x_K) \qquad \text{(MPC1)}$$

$$\text{s.to:} \quad x_{k+1} = f(x_k, u_k), \quad k \geq 0 \qquad \text{(MPC2)}$$

$$u_k \in \mathcal{U}, \quad k \geq 0 \qquad \text{(MPC3)}$$

$$x_k \in \mathcal{X}, \quad k \geq 0 \qquad \text{(MPC4)}$$

- Predict/estimate (if needed) $\mathcal{U}, \mathcal{X}, C(x, u), V(x), f(x, u)$.
- Solve (MPC), obtain the optimal sequence $u_1, \ldots, u_K$.
- Implement $u(\tau) = u_1$ at all $\tau \in [t, t + \Delta t)$.
- $t \leftarrow t + \Delta t$ and repeat.

Standard Approach: Model Predictive Control

$$\min_{\{u_k\}_{k=0}^{K-1}} \quad \sum_{k=0}^{K-1} C(x_k, u_k) + V(x_K) \qquad \text{(MPC1)}$$

$$\text{s.to:} \quad x_{k+1} = f(x_k, u_k), \quad k \geq 0 \qquad \text{(MPC2)}$$

$$u_k \in \mathcal{U}, \quad k \geq 0 \qquad \text{(MPC3)}$$

$$x_k \in \mathcal{X}, \quad k \geq 0 \qquad \text{(MPC4)}$$

Challenge 1: Need building model *f* – building-by-building customization.

Challenge 2: Multiple objectives – the cost *C(x,u)* is of the form

$$C(x, u) = \sum_{i=1}^{N} w_i c_i(x, u)$$

where the weights w_i may change during the operation.

Standard Approach: Model Predictive Control

$$\min_{\{u_k\}_{k=0}^{K-1}} \quad \sum_{k=0}^{K-1} C(x_k, u_k) + V(x_K) \qquad \text{(MPC1)}$$

$$\text{s.to:} \quad x_{k+1} = f(x_k, u_k), \quad k \geq 0 \qquad \text{(MPC2)}$$

$$u_k \in \mathcal{U}, \quad k \geq 0 \qquad \text{(MPC3)}$$

$$x_k \in \mathcal{X}, \quad k \geq 0 \qquad \text{(MPC4)}$$

Challenge 1: Need building model f – building-by-building customization.

Addressing Building-By-Building Customization

To solve the building-by-building customization challenge, we explore two alternative approaches:

1. Deep Reinforcement Learning (DRL)
   - Interact with the building, its model, or use previous operation data, to directly learn the optimal policy $u^* = \pi^*(s)$.
2. MPC using Gaussian Processes (MPC-GP)
   - Decompose:

$$f(s,u) = \underbrace{h(s,u)}_{\text{known (e.g., linear RC model)}} + \underbrace{g(s,u)}_{\text{unknown error}}$$

   - Model *g(s,u)* as Gaussian Process (GP), and learn it from data/interacting with building or model.

RL vs MPC-GP

DRL:

- No need for model, learn optimal policy directly.
- Powerful non-linear approximation tools, e.g., Deep Neural Networks.
- Might need a lot of exploratory data.
- Hard to impose constraints.

MPC-GP:

- Can use prior model information explicitly, less training data needed.
- Can impose constraints explicitly.
- Needs accurate forecasts
- Might be too conservative

Impact

Enabling Technology: GEBs, Renewables, Equity

- **GEB enabler:**
  - scalable to buildings of different sizes and types without building-by-building customization
  - avoids the expertise and cost of detailed engineering models
  - optimally balances building-centric, grid-serving, and resiliency-oriented objectives
- **Enhances renewable integration and decarbonization:**
  - adaptive to different technologies, including inverter-based resources (PV, batteries) and electric vehicle charging equipment
  - adaptive to different environments (e.g., microclimates)
- **Equity enabler**: considers human-in-the-loop and adaptive to population preferences and needs

Smart Building Control Fully Enabled Across Industry

Comparable effectiveness
Early results suggest learning-based controller approaches performance of the ideal model-based controller

Wide-scale energy industry impact
Fast adoption of technology in a sector that accounts for 40% of U.S. energy use

Grid interactivity and equity enabler
Algorithms' architecture unlocks grid connectivity, enhancing resilience and equity

Proof of concept in real building
The benefits of MODRLC will be illustrated in a **field demonstration** in New York City – see more details below

Progress

Summary for Year 1 and 2

- RL algorithms development and evaluation
  - Multi-objective RL framework and algorithms developed
  - Reduced order state-space model (ROM) developed for five-zone building
  - OpenAI Gym learning environment implemented for ROM
- GP-based algorithms development and evaluation
  - GP-based MPC formulation developed
  - Online GP-MPC algorithms developed
- Advanced Control Test Bed (ACTB) development
  - Development of the ACTB prototype and validation of the RL interface
  - Exploration of transfer learning and imitation learning for RL
- Field demonstration
  - Demonstration building in Manhattan selected
  - Control problem formulated and data obtained

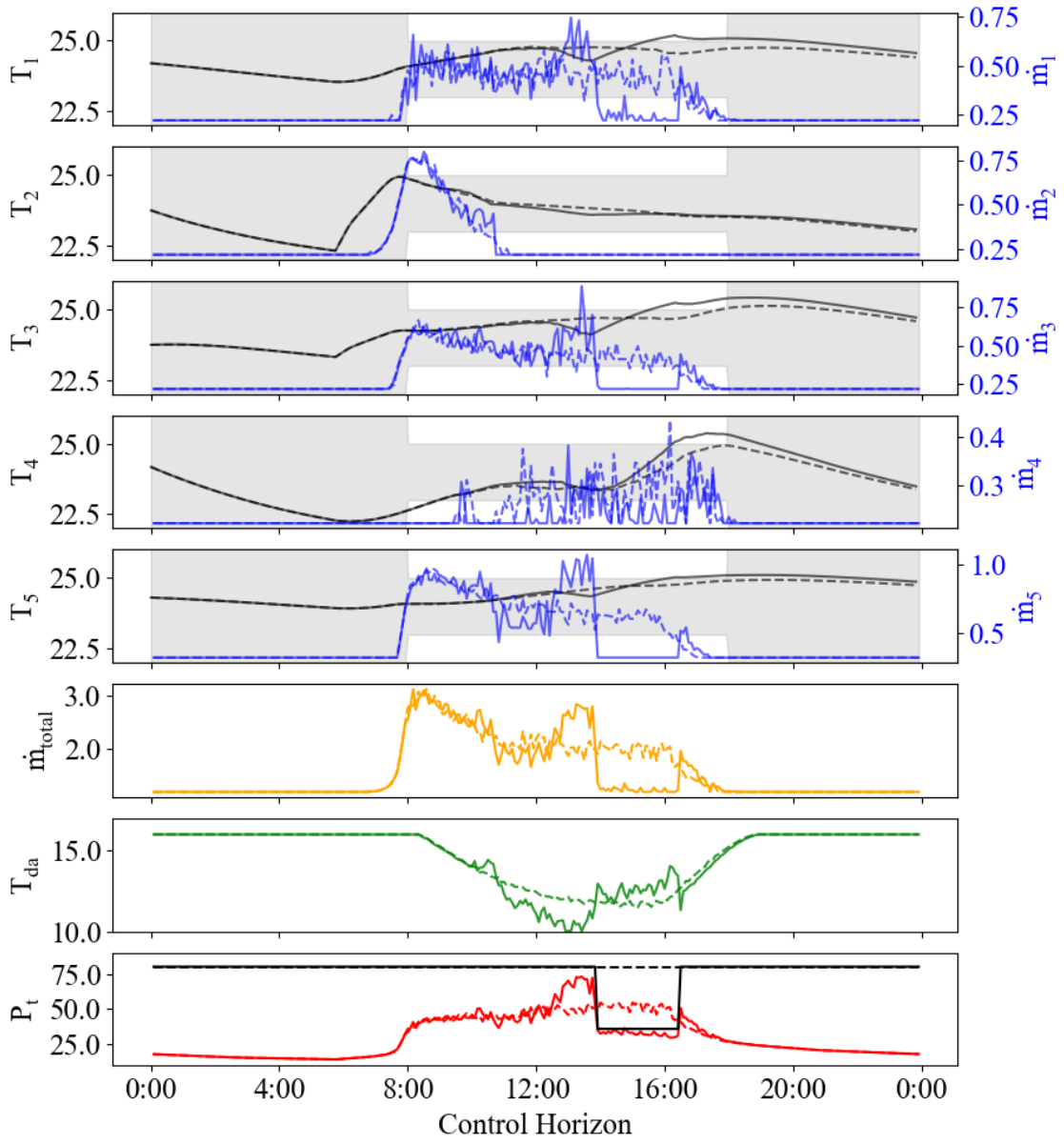
RL Controller Design

Developed a novel **two-stage global-local RL policy** search method that combines the advantages of two types of RL algorithms, in order to achieve a faster policy convergence to a better solution*

	ES-RL (Zero Order Gradient Estimation)	Proximal Policy Optimization (Policy Gradient)
Pros	• Scalable • Back-propagation (BP) free, fast gradient estimation • Optimizing on the Gaussian smoothed objective, likely to avoid local optimum	• Consider KL divergence during policy update (stable policy improvement) • Gradient-based learning on original objective (better local search ability)
Cons	❖ Only converges to the vicinity of the global optimum/a good-performing local optimum.	❖ Slower learning due to BP and conservative update ❖ Not scalable ($\mathcal{O}(N^2)$ communication complexity of full gradient info) ❖ Prone to be trapped in local optimum
Uses	✓ Stage I: Global Search	✓ Stage II: Local Tuning

* X. Zhang, R. Chintala, A. Bernstein, P. Graf and X. Jin, Grid-Interactive Multi-Zone Building Control Using Reinforcement Learning with Global-Local Policy Search, arXiv preprintarXiv:2010.06718 (2020), 2021 American Control Conference (ACC), IEEE Transactions on Smart Grid.

Grid Services – Demand Response



Two control test cases during a single day:

- No DR event that day.
- DR event with 36 kW power limit.

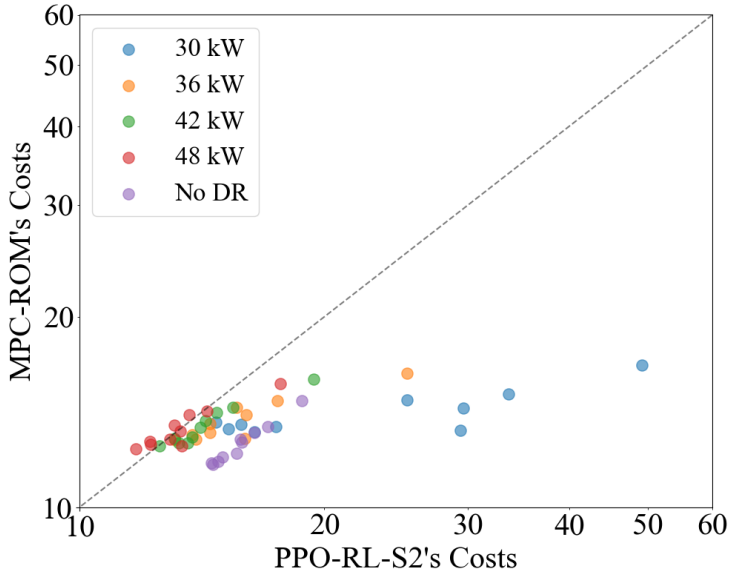
Control Performance:

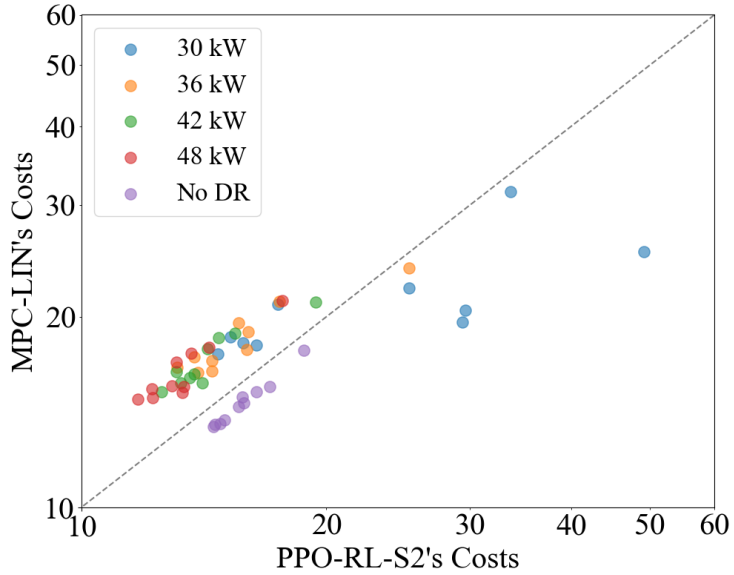
- All zone temperature are mostly kept within the comfort band, except for some short period during DR events.
- Grid requirement can be successfully met.
- Proactive actions are taken to prepare the building for the incoming DR event.
- Though not explicitly instructed, the RL controller learned to differentiate different zone for better control.

Grid Services – Comparison with MPC

MPC Baseline:

- *Non-Convex MPC (MPC-ROM)*: Perfect building model + Perfect exogenous inputs.
- *Convex MPC (MPC-LIN)*: Linearized building model + Perfect exogenous inputs.





On average, RL costs 25.3% more when compared with the optimal controller.

On average, RL costs 4.16% less when compared with the Convex MPC.

PPO-RL-S2 might provide even more cost reduction (compared with 4.16%) since MPC is unlikely to receive perfect forecast in real life scenarios. In addition, when using RL controller, no forecasting module and on-demand computation are needed.

Resilience Services

Objective: Train an RL controller that can help sustaining the building under grid-disconnected mode for as long as possible, leveraging the PV generation and battery on-site

Results:

- Each entry : mean, median, max, min of the self-sustained duration of the test cases.
- Larger initial storage can give a longer self-sustained duration.
- If building is disconnected in early morning, the self-sustained duration will be longer due to the support from PV generation.
- Control optimality needs to be evaluated by comparing with optimization-based control.

Self-sustained duration (Hour)		Grid Disconnected Time							
		0:00	**3:00**	**6:00**	**9:00**	**12:00**	**15:00**	**18:00**	**21:00**
Initial Storage (kWh)	**200**	4.28, 4.33, 4.50, 4.00	4.47, 4.50, 4.67, 4.17	12.36, 12.00, 16.25, 11.00	11.63, 11.17, 15.92, 10.08	7.26, 6.83, 10.50, 6.00	3.96, 3.83, 5.42, 3.17	3.13, 3.08, 3.67, 2.83	3.73, 3.75, 4.17, 3.42
	350	18.74, 18.33, 22.67, 17.42	17.91, 17.42, 22.92, 16.17	17.17, 16.67, 23.08, 15.17	17.17, 16.67, 24.0, 14.83	12.30, 11.92, 17.17, 10.42	8.94, 8.75, 11.75, 7.58	8.88, 8.83, 10.08, 8.00	21.01, 22.33, 24.00, 9.25
	500	22.91, 23.08, 24.00, 21.33	22.64, 22.83, 24.00, 20.83	22.52, 22.75, 24.00, 20.67	23.54, 24.00, 24.00, 21.00	20.88, 24.00, 24.00, 16.00	17.62, 14.92, 24.00, 13.00	24.00, 24.00, 24.00, 24.00	24.00, 24.00, 24.00, 24.00

MPC based on Gaussian Processes (GP)

OBJECTIVE: Develop a learning predictive control that (i) continuously learn the temperature dynamics of the building; and, (ii) use the learned dynamics to solve a multi-objective predictive control problem.

LEARNING METHOD – GP

$$G(\mathbf{z}^*)|\mathcal{S}, \mathbf{z}^* \sim \mathcal{N}(\mu(\mathbf{z}^*), \sigma(G(\mathbf{z}^*)))$$

$$\mu(\mathbf{z}^*) := \mathbf{k}(\mathbf{z}^*)^\top (\mathbf{K}_N + \sigma^2 \mathbf{I})^{-1} \mathbf{y}$$

$$\sigma(\mathbf{z}^*, \mathbf{z}^*)^2 := k(\mathbf{z}^*, \mathbf{z}^*) - \mathbf{k}(\mathbf{z}^*)^\top (\mathbf{K}_N + \sigma^2 \mathbf{I})^{-1} \mathbf{k}(\mathbf{z}^*)$$

Gradient estimation $\hat{\nabla} J(\mathbf{m}^j, \mathbf{T}^{\mathrm{da},j}) = \frac{1}{2\gamma} \boldsymbol{\xi} (\hat{J}(\mathbf{u}_+^j) - \hat{J}(\mathbf{u}_-^j))$

GP-BASED ALGORITHM

Primal update:

$$\mathbf{m}^{j+1} = \mathrm{proj}_{\mathcal{M}} \left\{ \mathbf{m}^j - \alpha \left(\hat{\nabla} J(\mathbf{m}^j) + \boldsymbol{\lambda}^j \right) \right\}$$

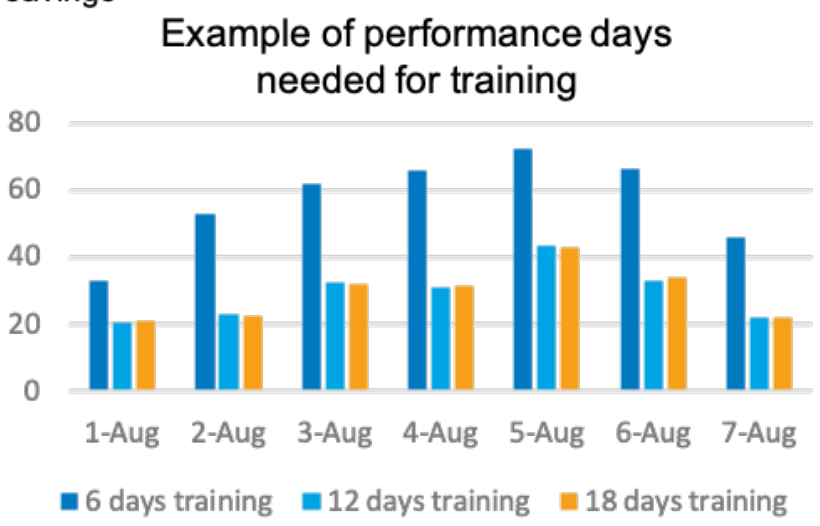
$$\mathbf{T}^{\mathrm{da},j+1} = \mathrm{proj}_{\mathcal{N}} \left\{ \mathbf{T}^{\mathrm{da},j} - \alpha \hat{\nabla} J(\mathbf{T}^{\mathrm{da},j}) \right\}$$

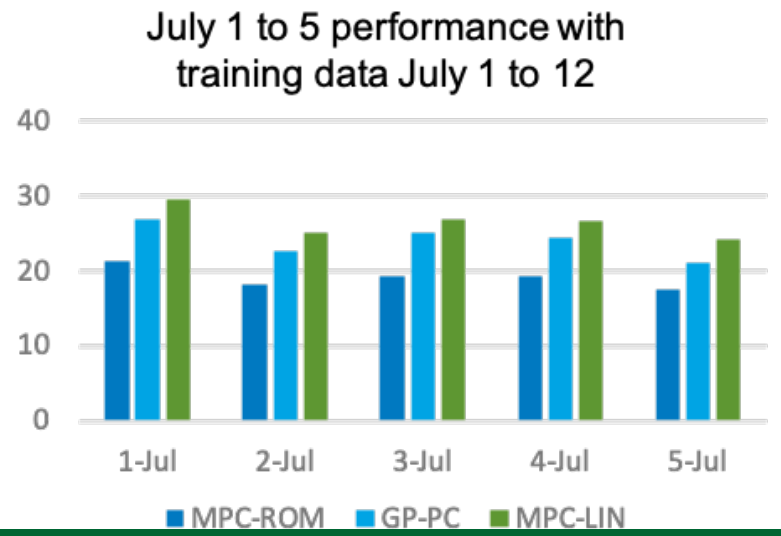
Dual update: $\boldsymbol{\lambda}^{t+1} = \left[\boldsymbol{\lambda}^t + \alpha \left(\mathbf{m}^{\mathrm{s}} - d\mathbf{1} \right) \right]^+$

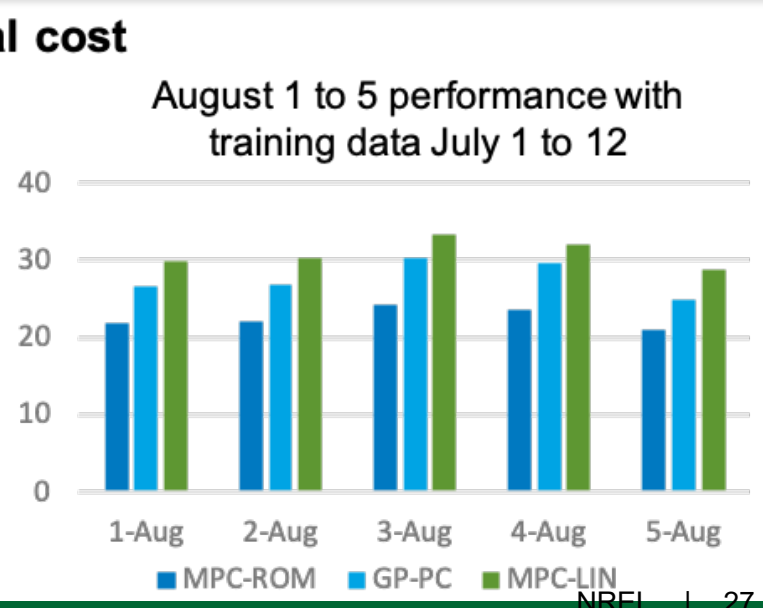
ADVANTAGES OF GP IN CONTROL:

- GPs provide a confidence level of the prediction through the variance.
- GPs require a small size of data sets in order to provide meaningful estimates (sample efficient method).
- Prior knowledge of the system can be incorporate to improve the learning process.
- GPs provide a closed form expression for the output mean and variance.

Objective function: 80% thermal discomfort & 20% energy savings







Advanced Controls Test Bed (ACTB)

Objectives:

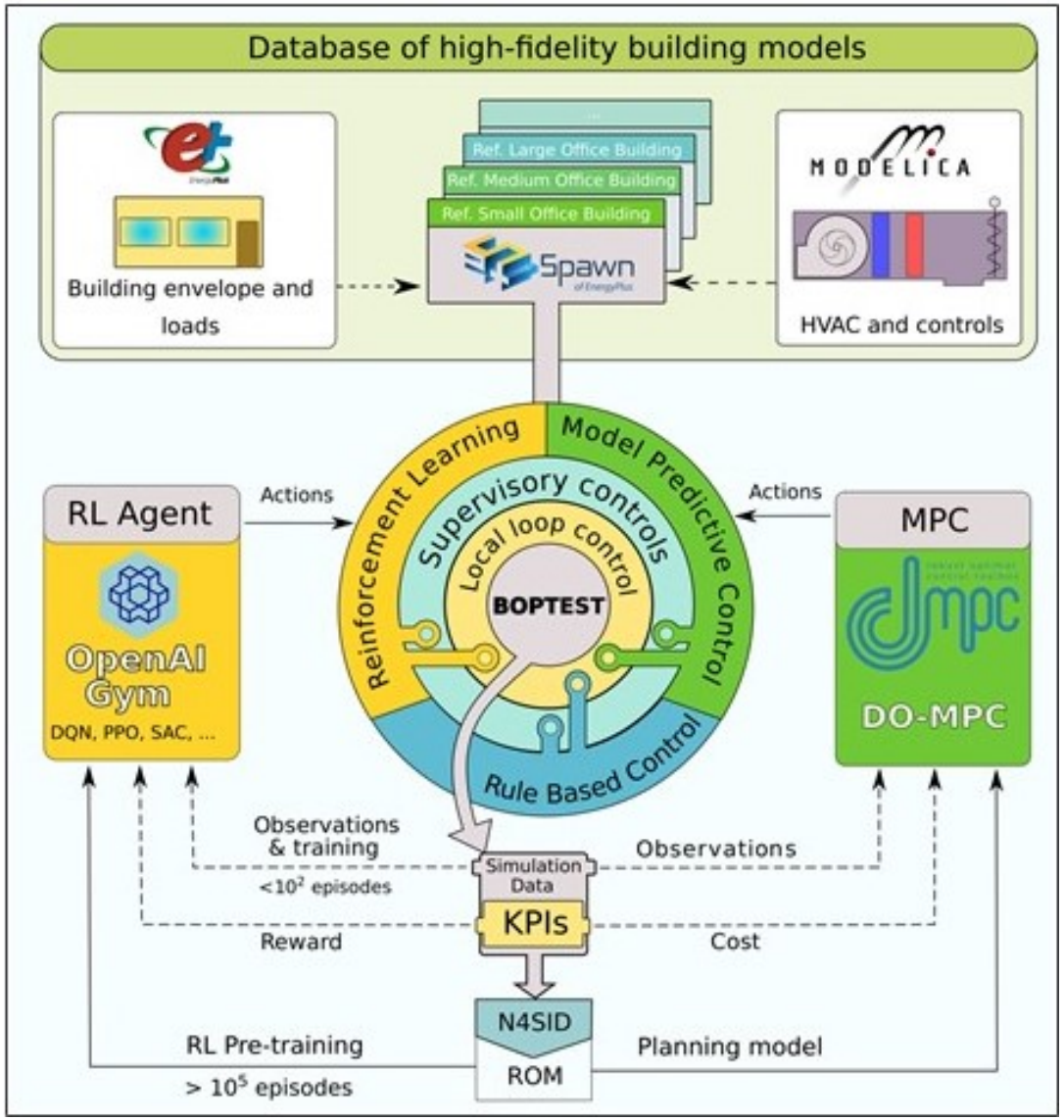
- To develop **an open-source controls test bed** that uses **high-fidelity building models**
- To provide interfaces to **machine learning and MPC libraries**
- To **explore transfer learning** using reinforcement learning controllers

Progress to date:

- Development of **the ACTB prototype** using IBPSA's BOPTEST framework
- Development and validation of **reinforcement learning capabilities** using OpenAI Gym
- Development of a **high-fidelity Spawn building model**

Current work:

- Development and validation of **MPC capabilities** using DO-MPC
- Development of **additional** Spawn building models
- Identification of **a real building** for exploring transfer learning



Stakeholder Engagement

Field Demonstration with QCoefficient

- Worked with QCoefficient to select a high-rise building in Manhattan for demonstration
- Obtained feedback on the control problem and update the formulation accordingly
- QCoefficient engaged with different stakeholders (**ConEd, NYSERDA, NYISO, and the NYDPU**) to promote the demonstration efforts
- The demonstration will help New York authorities to achieve their renewable integration and decarbonization goals
- NREL engaged **Schneider Electric** to obtain feedback and develop future partnership

Demonstration Building Details

- Number of floors: 40
- Total Building Area: 1.3 million sq-ft.
- Cooling System:
  - 4 electric chillers
  - 8 cooling towers

EnergyPlus Model:

- Building consolidated into 3 floors
- Each floor has 5 zones; core zone and 4 perimeter zones
- Supervisory control to regulate thermal comfort using a single setpoint temperature for each floor

Next steps:

- Develop reduced order model
- Develop RL and MPC
- Test in simulation
- Deploy in building

Remaining Project Work

Algorithms Development

RL

- RL algorithms with no exploration to facilitate actual deployment
- Transfer learning between buildings and model-to-building
- Multiagent RL for controlling a community of buildings (stretch)

MPC-GP

- MPC-GP simulations and analysis
- Primal-dual method for real-time implementation

ACTB Testbed and Field Demo

ACTB

- Development of MPC capabilities using DO-MPC
- Development of additional DOE Reference Commercial Buildings Spawn models
- Extension of the RL framework for parallel computing
- Identification of a real building, development of its Spawn model, and validation of the ACTB
- Exploration of multi-agent RL and new RL algorithms, e.g. CQL, BCQ, SAC
- Exploration of fully offline learning from building data using NN models to generate pseudo-data
- Develop novel efficient online algorithms with a guided exploration when building data is unavailable (new construction).

Field demo – see **Stockholders Engagement** part.

Thank You

NREL, CU Boulder, QCoefficient
PI: Andrey Bernstein
andrey.bernstein@nrel.gov

REFERENCE SLIDES

Project Budget

Budget History (Cost-to-Date)					
FY 2020 (past)		FY 2021 (to date)		FY 2022 (planned)	
DOE	Cost-share	DOE	Cost-share	DOE	Cost-share
$326k	80k	$714k	$120k	$1.5M	$375k

Project Plan and Schedule

Task	Description	Budget Period 1				Budget Period 2				Budget Period 3			
		Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
1	**MODRLC Algorithms Development**												
1.1.1	Develop multi-objective RL framework for building control		M										
1.1.2	Develop MODRLC algorithms using approximate and deep RL approaches			M									
1.1.3	Explore the role of models	M			M								
1.1.4	Extend the algorithms to online learning and other exploration schemes						M						
2	**Large-Scale Evaluation**												
2.2.1	Advanced control testbed (ACTB) development							M				M	
2.2.2	Large-scale validation using distribution feeder								M			M	
3	**Commercial Building Commissioning and Field Demonstration**												
1.3.1	Commissioning of a commercial building for demonstration					M		M					
2.3.1	Development of a baseline MPC controller for the selected building									M			
3.3.1	Field deployment and demonstration										M		M

Completed work
M Milestone
M Go/No-Go
M Milestone missed due to COVID-19
Work in progress

Approach

Hybrid approach:

$$\min_{\{u_k\}_{k=0}^{K-1}} \quad \sum_{k=0}^{K-1} C(x_k, u_k) + V(x_K) \qquad \text{(MPC1)}$$

$$\text{s.to:} \quad x_{k+1} = f(x_k, u_k), \quad k \geq 0 \qquad \text{(MPC2)}$$

$$u_k \in \mathcal{U}, \quad k \geq 0 \qquad \text{(MPC3)}$$

$$x_k \in \mathcal{X}, \quad k \geq 0 \qquad \text{(MPC4)}$$

- Use a GP-model instead of *f*.
- Approximate *V(x)* using RL.
- Start with a large enough *K*, and progressively decrease it as the value function estimate *V* becomes more accurate.

Addressing Multiple Objectives

$$C(x,u) = \sum_{i=1}^{N} w_i c_i(x,u)$$

In our application, we have the following objectives ($N = 4$):

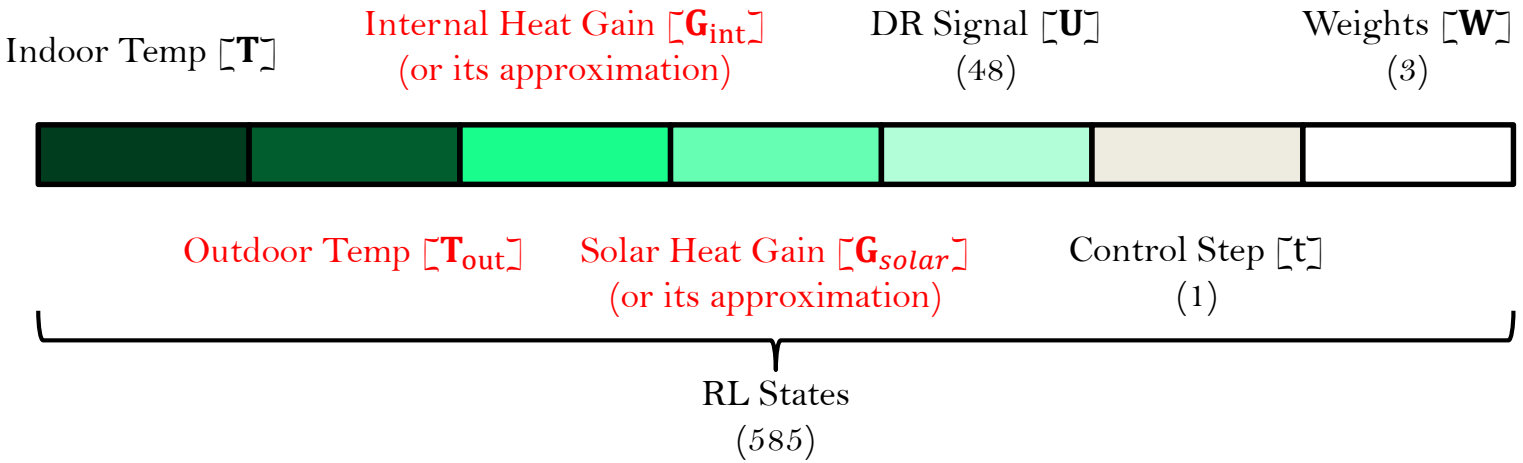
- Building comfort
- Minimizing energy consumption
- Providing flexibility for grid services
- Resilience objectives for the contingency situations

Extend the above framework: define augmented state $s = (x,w)$, and

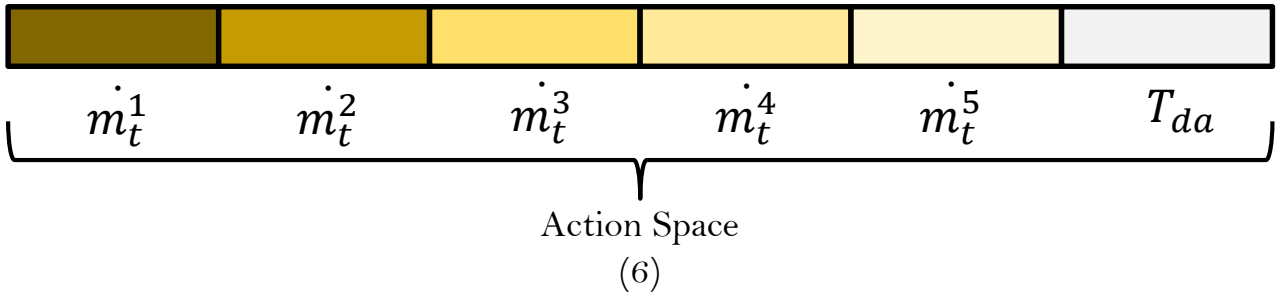
$$\widetilde{C}(s,u) := \sum_{i=1}^{N} w_i c_i(x,u) = C(x,u)$$

RL Controller Design

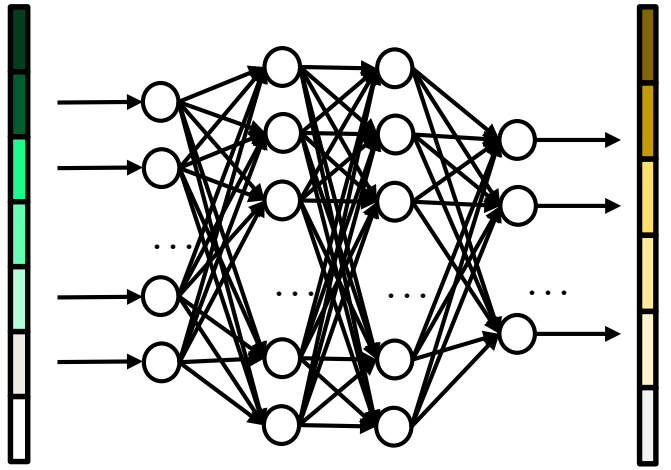
Input Vector (State)



Output Vector (Action)



Policy Network Structure



[585] [256, 128, 128, 64, 64, 32, 16] [6]

Progress: RL Algorithms

Reduced Order Model (ROM)

5 subsystem models developed to simulate the 5 zones of the building

Subsystem dynamics

$$x_{k+1}^i = A^i x_k + B^i u_k$$
$$T_{k+1}^i = C^i x_{k+1}$$

Inputs to the model

$$u_k^i = [T_k^{oa} - T_k^i, Q_k^{hvac}, Q_k^{sol}, Q_k^{int}, T_k^{sur} - T_k^i]$$
$$Q_k^{hvac} = \dot{m}_k^i \, (T_k^{da} - T_k^i)$$

$$i \in \{1,2,3,4,5\}, k = t, t+1, \dots, t+n_p-1$$

- T^{oa} - outside air temperature
- T^i- temperature of i^{th} room
- T^{sur}- temperature of surrounding rooms
- $Q^{hvac}, Q^{sol}, Q^{int}$- Heat sources
- $\dot{m}^i$ - mass flow rate of i^{th} room
- T^{da}- discharge air temperature
- k – optimization time step
- t – current time step
- n_p – optimization horizon
- t_s – duration of time step
- P^{ch}, P^{fan} – fan and chiller power
- w_e, w_{comf}, w_{dr} – objective function weights
- f_{dis} – function evaluating discomfort
- f_{dr} - function evaluation demand response

MPC objective function

$$w_e \cdot \sum_{k=t}^{t+n_p-1} \left(P_k^{ch} + P_k^{fan}\right) \cdot t_s + w_{comf} \sum_{k=t}^{t+n_p-1} \sum_{i}^{n} f_{dis}\left(\hat{T}_k^i\right) + w_{dr} \cdot \sum_{k=t}^{t+n_p-1} f_{dr}\left(P_k^{ch}, P_k^{fan}, P_k^{dr-ref}\right)$$

Progress: RL Algorithms

Control Variables

Inputs to the model

$$u_k^i = [T_k^{oa} - T_k^i, Q_k^{hvac}, Q_k^{sol}, Q_k^{int}, T_k^{sur} - T_k^i]$$

$$\boldsymbol{Q_k^{hvac} = \dot{m}_k^i\,(T_k^{da} - T_k^i)}$$

control variables

state

- T^{oa} - outside air temperature
- T^i- temperature of i^{th} room
- T^{sur}- temperature of surrounding rooms
- $Q^{hvac}, Q^{sol}, Q^{int}$- Heat sources
- $\dot{m}^i$ - mass flow rate of i^{th} room
- T^{da}- discharge air temperature
- k – optimization time step
- t – current time step
- n_p – optimization horizon
- t_s – duration of time step
- P^{ch}, P^{fan} – fan and chiller power
- w_e, w_{comf}, w_{dr} – objective function weights
- f_{dis} – function evaluating discomfort
- f_{dr} - function evaluation demand response

Bilinear constraint in the optimization problem → **Non-convex, hard optimization**

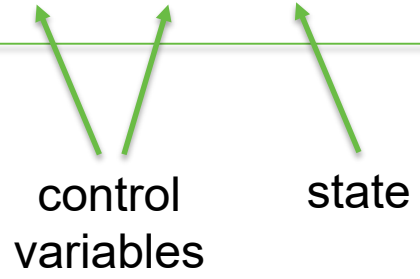
Progress: RL Algorithms

Two MPC Formulations

Inputs to the model

$$u_k^i = [T_k^{oa} - T_k^i, Q_k^{hvac}, Q_k^{sol}, Q_k^{int}, T_k^{sur} - T_k^i]$$

$$\boldsymbol{Q_k^{hvac} = \dot{m}_k^i \, (T_k^{da} - T_k^i)}$$



Optimal Controller: Non-convex MPC

- Input vector is a function of predicted temperatures
- Q_k^{hvac} is a bilinear input.

Progress: RL Algorithms

Two MPC Formulations

Inputs to the model

$$u_k^i = [T_k^{oa} - T_k^i, Q_k^{hvac}, Q_k^{sol}, Q_k^{int}, T_k^{sur} - T_k^i]$$

$$\boldsymbol{Q_k^{hvac} = \dot{m}_k^i \, (T_k^{da} - T_k^i)}$$

control variables

state

Optimal Controller: Non-convex MPC

- Input vector is a function of predicted temperatures
- Q_k^{hvac} is a bilinear input.

Baseline: Convex MPC

- Based on first-order Taylor series expansion of the non-linear model

Progress: Grid Services – Demand Response

Environment: Based on training data (i.e., weather/occupancy profile) from 07/01 to 07/31, demand response (DR) events will be triggered according to a certain distribution.

The objective: to train an RL controller that properly controls the building to satisfy both requirements from building and grid.

DR Settings:

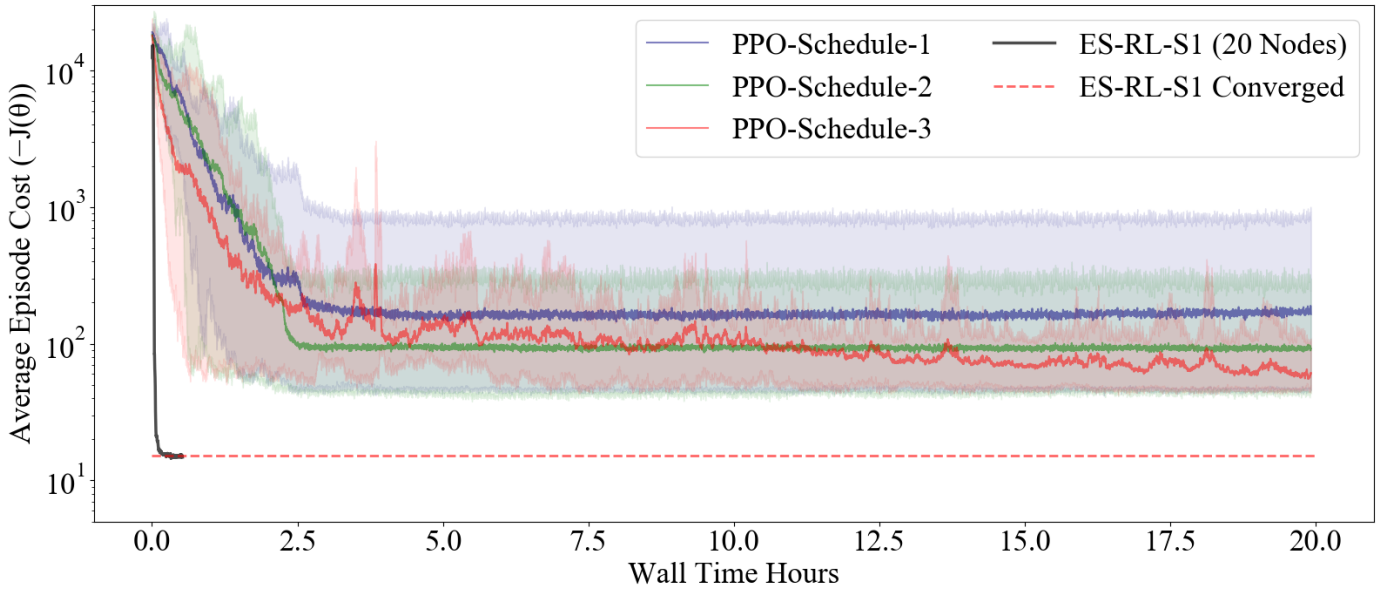
- Random occurrence: 50% of the training days have a DR event in a day, other 50% do not.
- The DR event starts any time between 11AM and 6PM, with a duration of T hours, $T \sim U(2, 4)$
- Each DR event will have a power limit D, which represents the building load's upper bound.
- D is inversely proportional to T and is bounded between 30kW and 50kW. (Building peak demand is ~70kW)
- Controller will be notified 4 hours before a DR event begins.

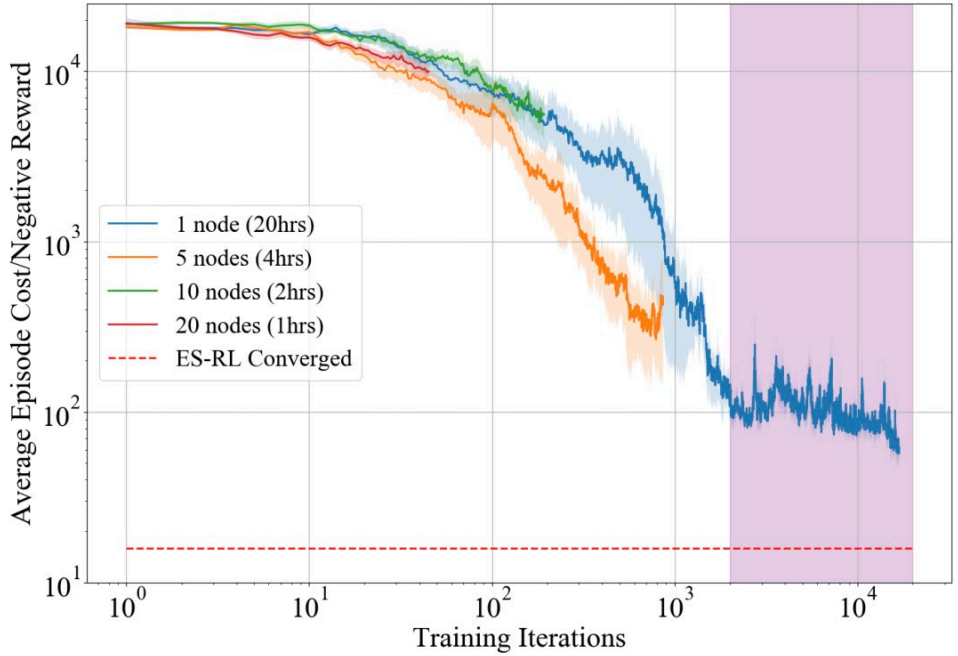
Multi-objective Settings: During DR events, weights will change accordingly:

- During normal hours $w_{comf} = 0.7\ w_e = 0.2, w_{dr} = 0.1$
- During DR event hours $w_{comf} = 0.5, w_e = 0,\ w_{dr} = 0.5$

Progress: Grid Services – Demand Response – Global Search

Computing platform: RL controller training is conducted on the NREL high-performance computing platform (Eagle).



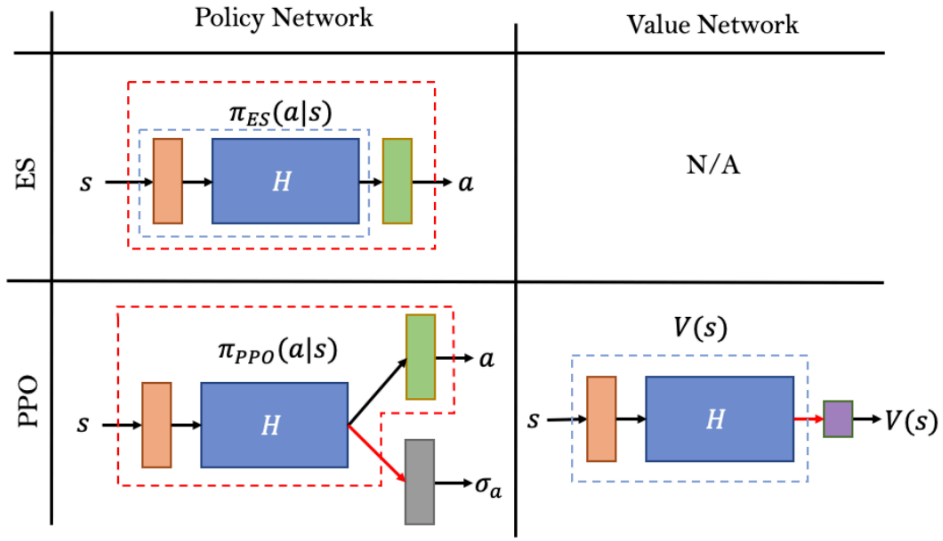


- ES-RL can be scaled to 20 HPC nodes, converging to a better optimum in 30 minutes of training (consuming 20*0.5 = 10 node-hour computing resource). In contrast, PPO is trapped in poorer local optima after consuming 20 node-hour of computing resources.
- PPO is not scalable (i.e., using more HPC nodes does not provide benefit such as faster convergence). So, PPO in the left figure uses one HPC node when training.

Globally searched policy, referred to as *ES-RL-S1*, will be passed to the second stage for fine-tuning.

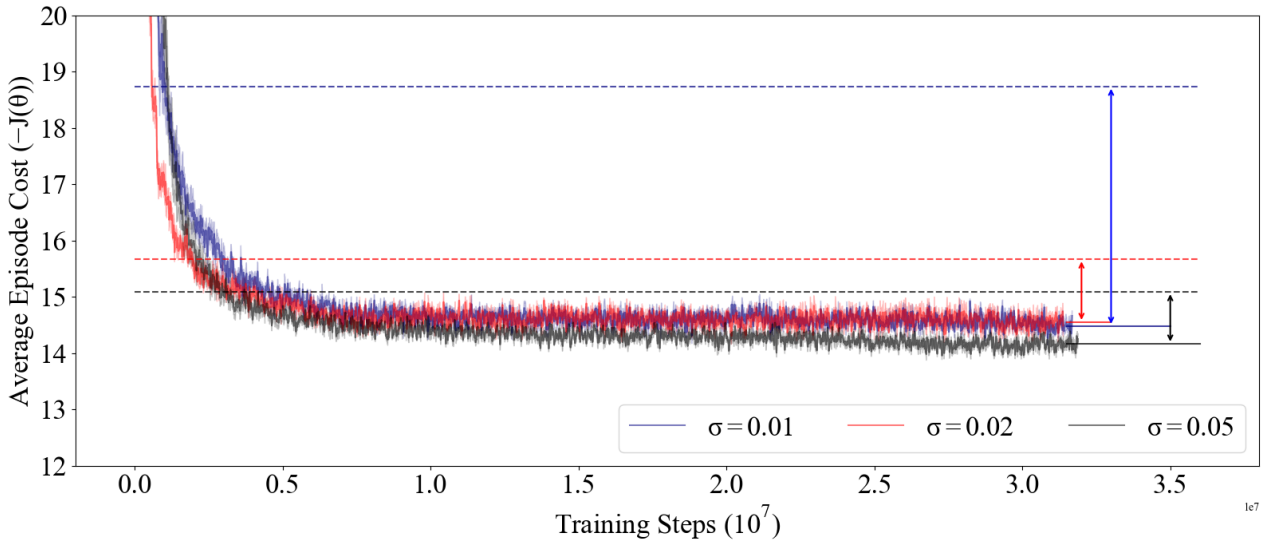
Progress: Grid Services – Demand Response – Local Tuning

Stage II policy fine-tuning using PPO.



- Due to the change of algorithms between stages, knowledge learned in ES-RL-S1 needs to be transferred to the PPO learning stage. We leverage the weight-copying warm-starting.

Locally tuned policy, referred to as *PPO-RL-S2*, will be used for building control.



- Fine-tuning three best performing ES-RL-S1 control policies. Solid curves show the PPO-RL-S2 learning progress and dash lines indicates the performance of the ES-RL-S1 predecessors.

TABLE V
SECOND STAGE POLICY TUNING IMPROVEMENT (COST REDUCTION)

σ_i	Average Episodic Cost at Policy Convergence		
	ES-RL-S1	*PPO-RL-S2*	*Improvement (%)*
0.01	18.74	14.48	22.73%
0.02	15.67	14.55	7.15%
0.05	15.09	14.17	6.49%

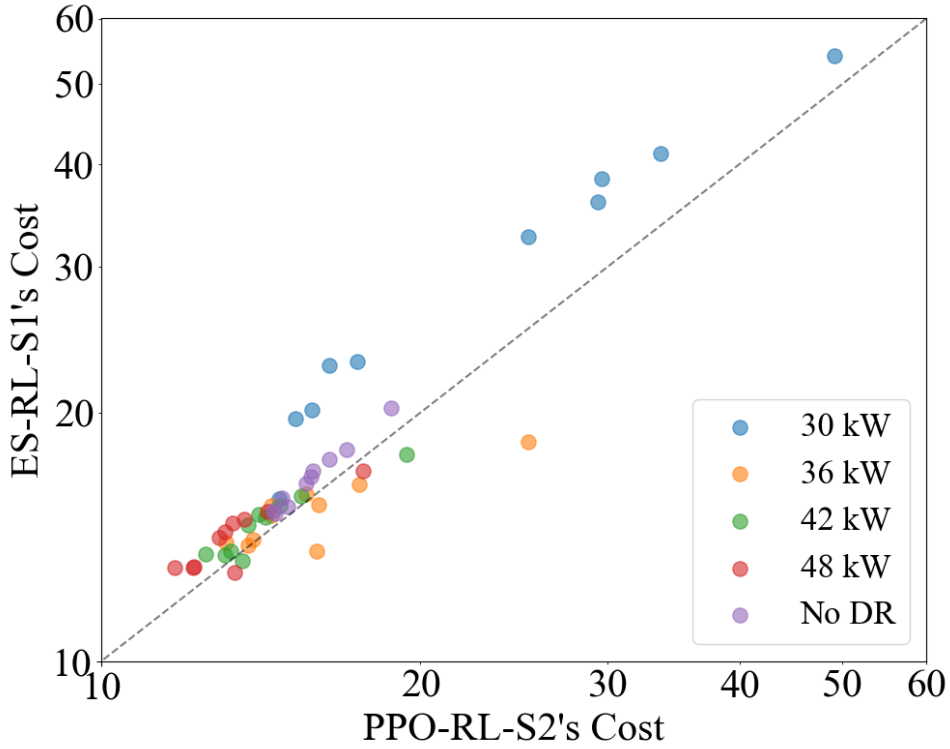
Grid Services – Demand Response

The objective: Train an RL controller that properly controls the building to satisfy both requirements from building and grid.

For each of the 10 testing days, we consider five DR scenarios with different power limit:

$$P_k^{dr-ref} = \{30, 36, 42, 48, No\ DR\}$$

PPO-RL-S2 brings a 7.55% cost reduction when compared with ES-RL-S1.



Daily cost of Global (ES-RL-S1) and local (PPO-RL-S2) algorithms

Progress: Resilience Services

Objective:
To train an RL controller that can help sustaining the building under grid-disconnected mode for as long as possible, leveraging the PV generation and battery on-site.

Assumption for training:
- Building can be disconnected at any time in a day, emulating the randomness of grid-level fault.
- During outage, power consumed by the building comes from in-building PV and battery:
  $P_{battery} + P_{pv} = P_{AirConditioner} + P_{Other}$
  P_{Other} is the power of **90% of the assets in the building**.
- During outage, use a larger comfort band.
- PV generation profile is given as exogenous data.
- Battery initial energy is sampled from a Gaussian distribution.
- Training episode terminates if
  - the energy in battery is depleted; or
  - the building has successfully self-sustained for 24 hours.

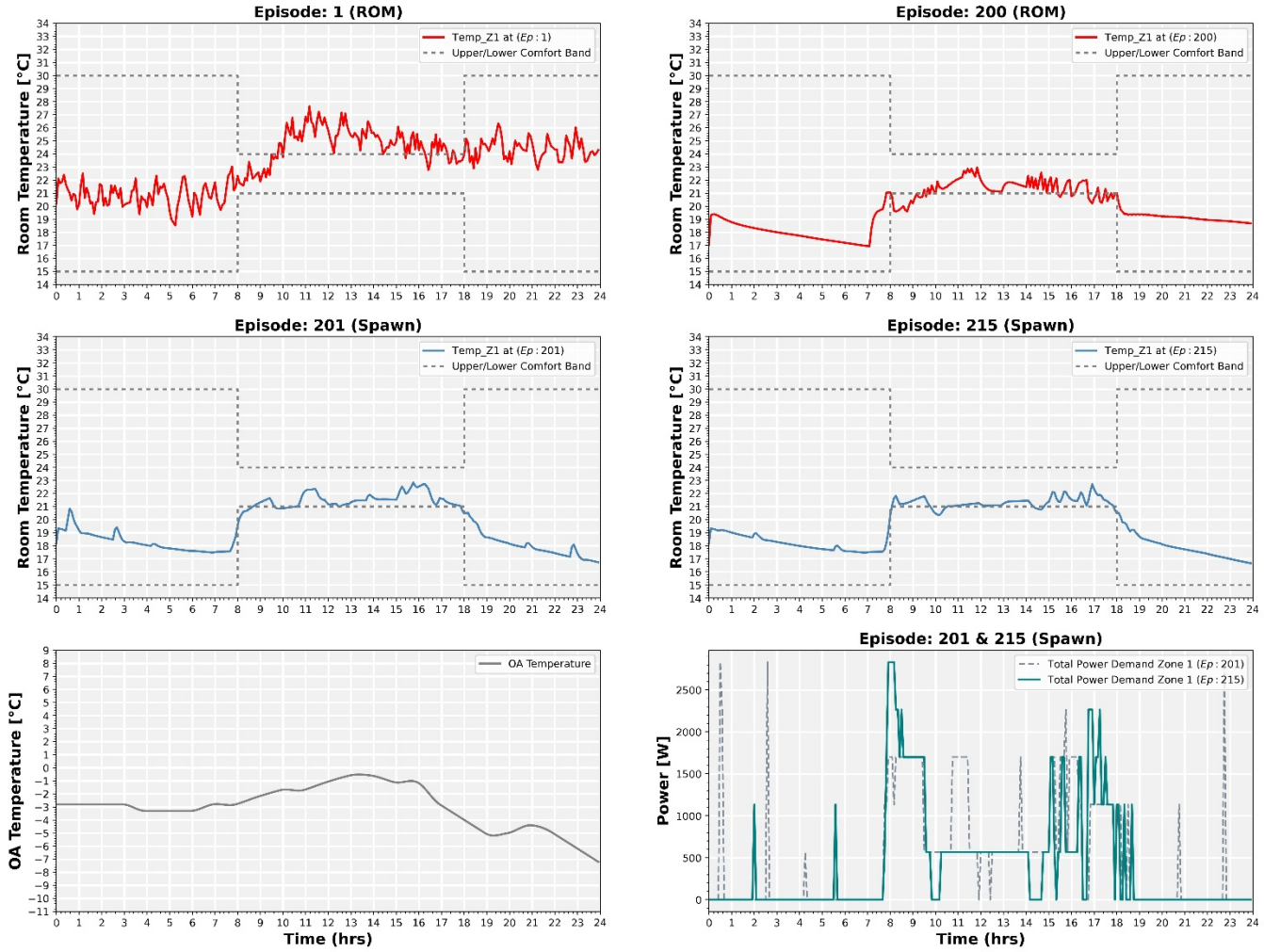
Progress: Advanced Controls Test Bed (ACTB)

Reinforcement learning test cases

No.	Cases	Algorithm	Zones	Note
1a.	Low-Level Heating	DQN	1 Zone	Pre-trained with ROM
1b.	Low-Level Heating	PPO	1 Zone	Pre-trained with ROM
2a.	High-Level Heating	DQN	1 Zone	Trained only on Spawn
2b.	High-Level Heating	PPO	1 Zone	Trained only on Spawn
2c.	High-Level Heating	DQN	1 Zone	Hybrid offline-online learning with RBC data
2d.	High-Level Heating	PPO	1 Zone	Uses Imitation learning from RBC
3a.	Low-Level Cooling DR	DQN	5 Zones	Pre-trained with ROM
3b.	Low-Level Cooling DR	PPO	5 Zones	Pre-trained with ROM
3c.	Low-Level Cooling DR	PPO	5 Zones	Uses Imitation learning from Heuristic Rules

Progress: Advanced Controls Test Bed (ACTB)

Example 1: Case 1a, low-level heating control with ROM pre-training

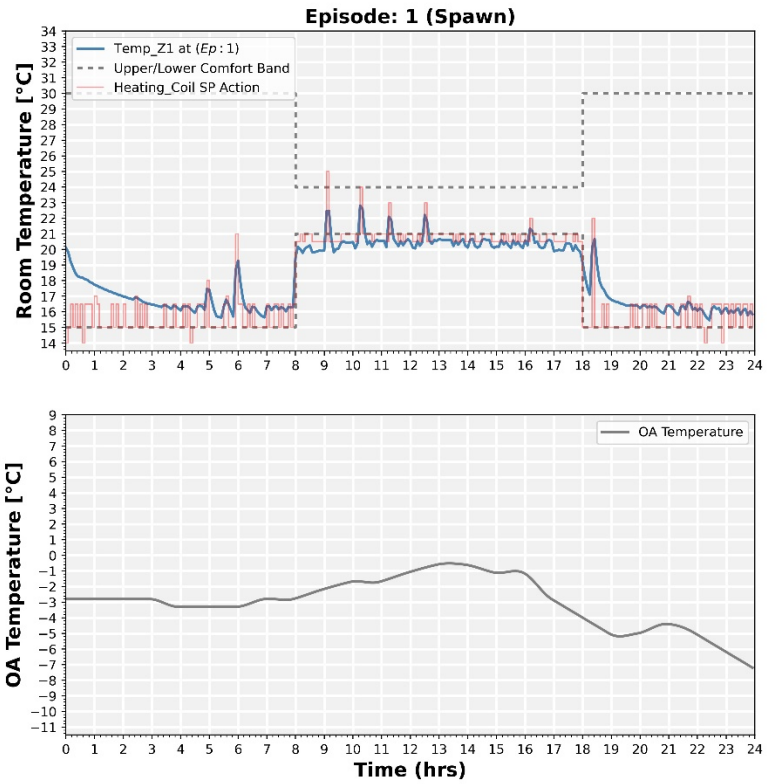


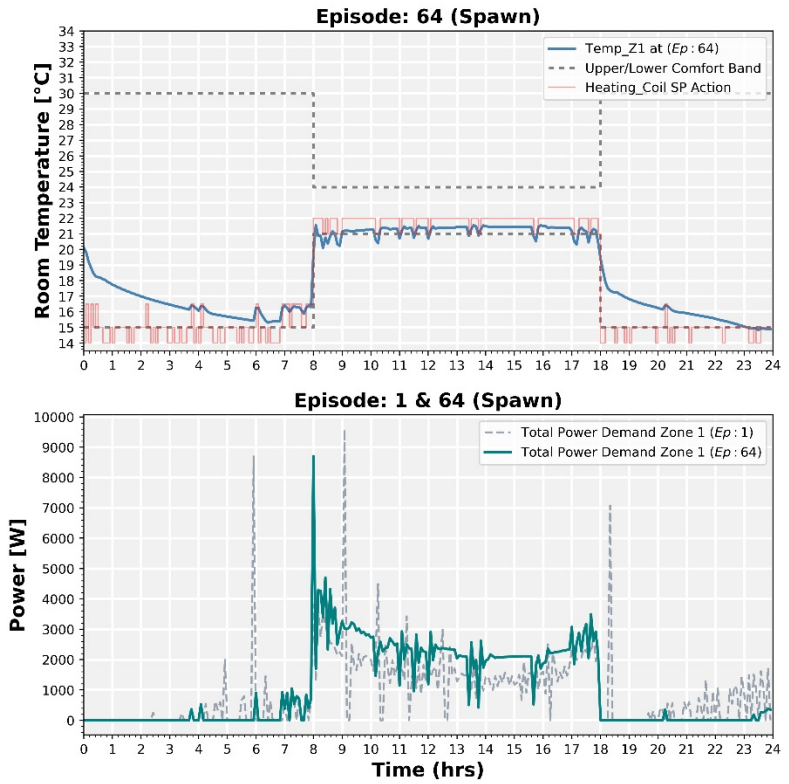
- Pre-training of a DQN agent on a **reduced-order model** for 200 episodes
- Continued training of the agent on the more **complex** Spawn model, which exhibits **different system dynamics**
- Each episode requires under 5 minutes to train using the ROM, compared to over 1 hour for the high-fidelity Spawn model.

Outcome: Pre-training saves considerable training time. In practical applications, pre-training could **cut down the time** during which an RL controller under-performs in a real building.

Progress: Advanced Controls Test Bed (ACTB)

Example 2: Case 2d, high-level heating control, imitation learning from RBC



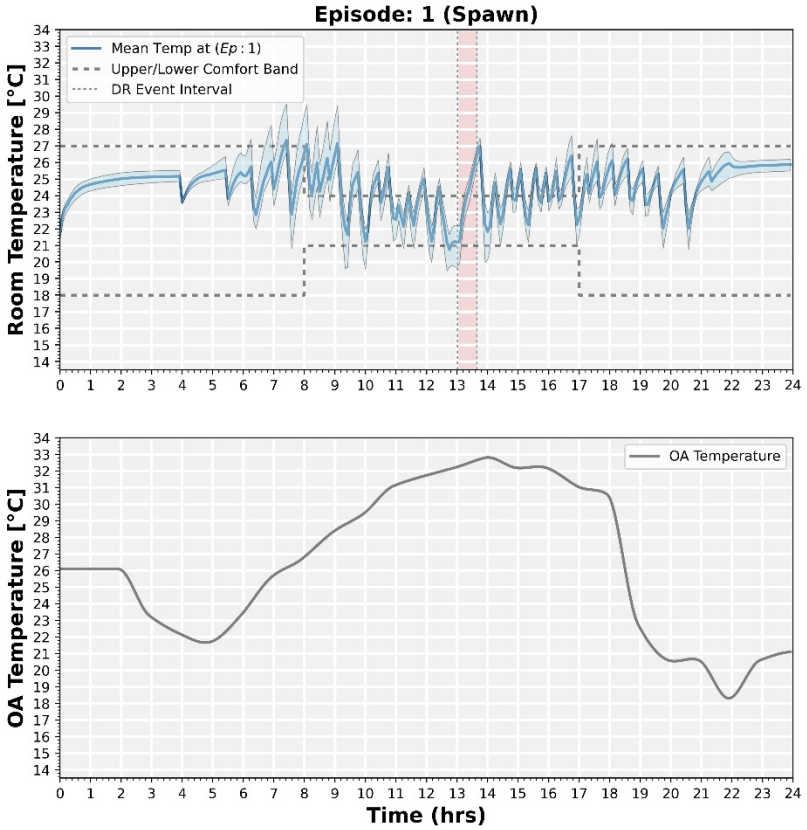


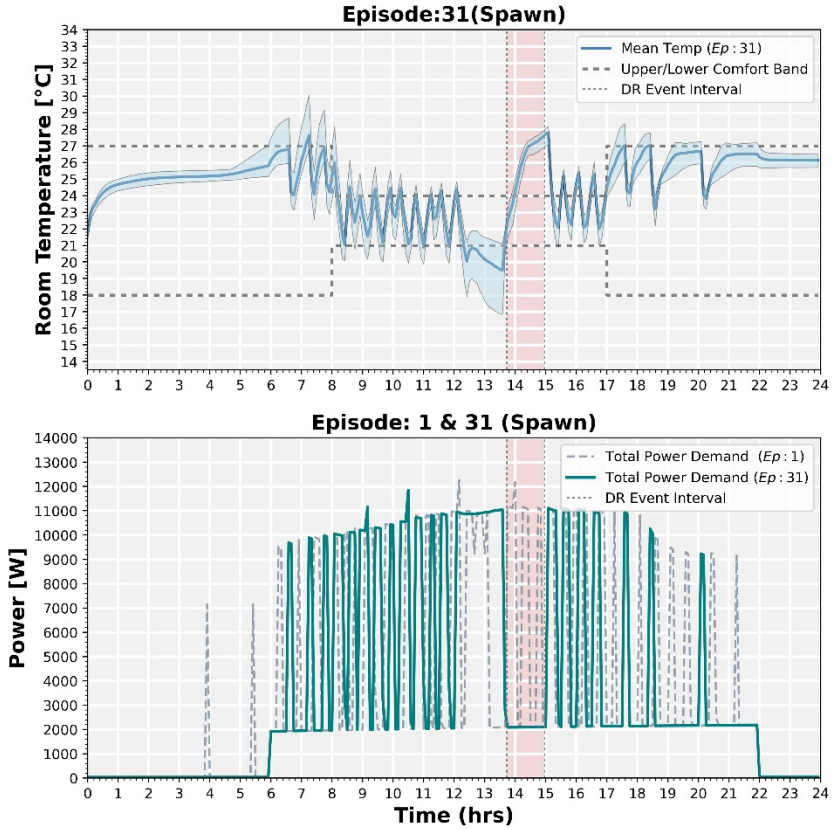
- The PPO agent's memory buffer is **pre-populated with experience** from an RBC controller. As the model starts learning, this experience is gradually replaced by the one it gathers from interacting with the Spawn model.
- Imitation learning **saves considerable training time** as it allows the agent to start its training in a state that is **closer to the optimal behavior.**

Outcome: Historical data can be used to **improve training performance** with imitation learning.

Progress: Advanced Controls Test Bed (ACTB)

Example: Case 3c, high-level cooling control, imitation learning from heuristic rules during a DR event





- The PPO agent's memory buffer is **pre-populated** with data generated using **heuristic rules** (e.g.: cooling on if $T_{room} > T_{cooling}$).
- As in the previous case, training is **considerably faster** using imitation learning.

Outcome: The usage of heuristic rules renders it possible to train an agent **in the absence of historic data** using imitation learning