Massachusetts Institute of Technology

BOPTEST webinar

Revisiting MPC for HVAC control using BOPTEST

Sicheng (James) Zhan
Postdoctoral associate, MIT Building Technology

26/2/5

At the intersection of architectural engineering and computational science

2016, BS in Energy and Environment Systems Engineering, Zhejiang University

2017, MS in Civil Engineering, Carnegie Mellon University

2022, PhD in the Built Environment, National University of Singapore

2022-24, Research fellow, National University of Singapore

2024-, Postdoctoral Associate in Building Technology, Massachusetts Institute of Technology

Research interest

Building energy simulation, optimal control, digital twin, carbon reduction, energy flexibility, scientific machine learning, optimization

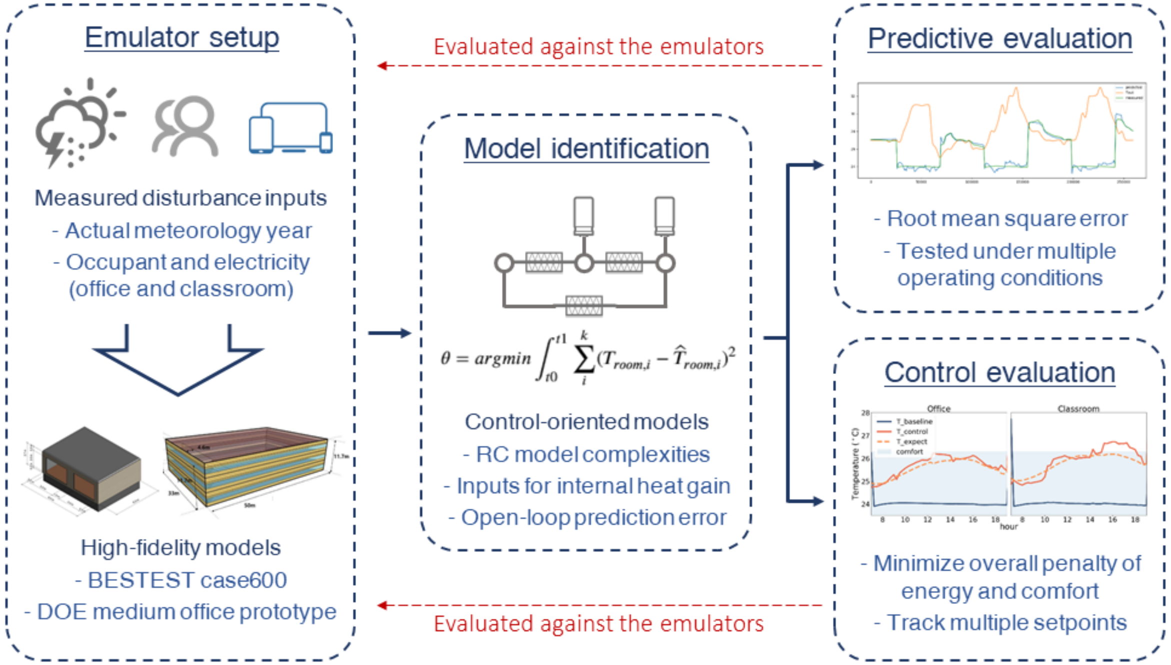
My history with BOPTEST and Modelica Buildings Library

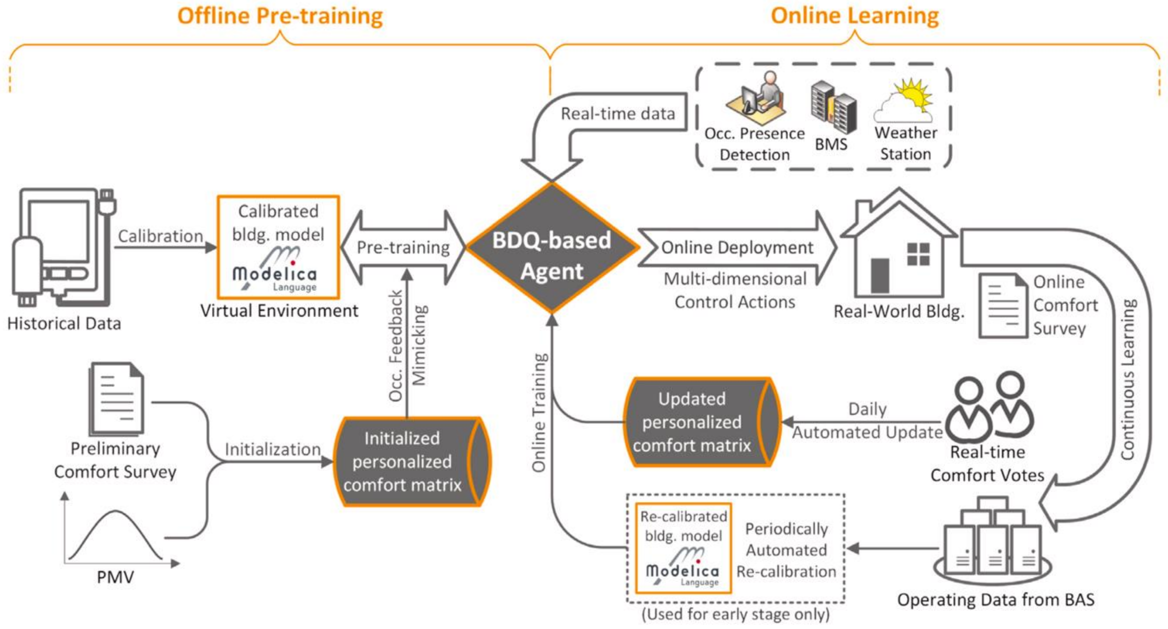
BS 2019, IBPSA Project 1, BOPTEST workshop

My PhD, data requirements for MPC, simple/complex testbed in Modelica (Thank you Dave!!)

Other work at NUS ideas-lab, calibrated MBL model for RL training

BOPTEST as control and prediction testbed





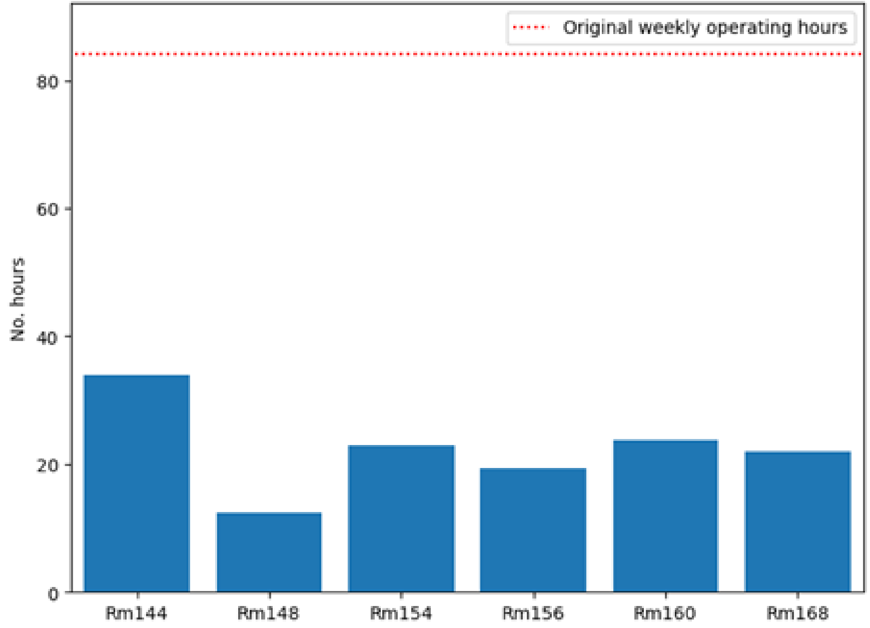
Occupant-centric control in campus buildings

BOPTEST role - validate control methods

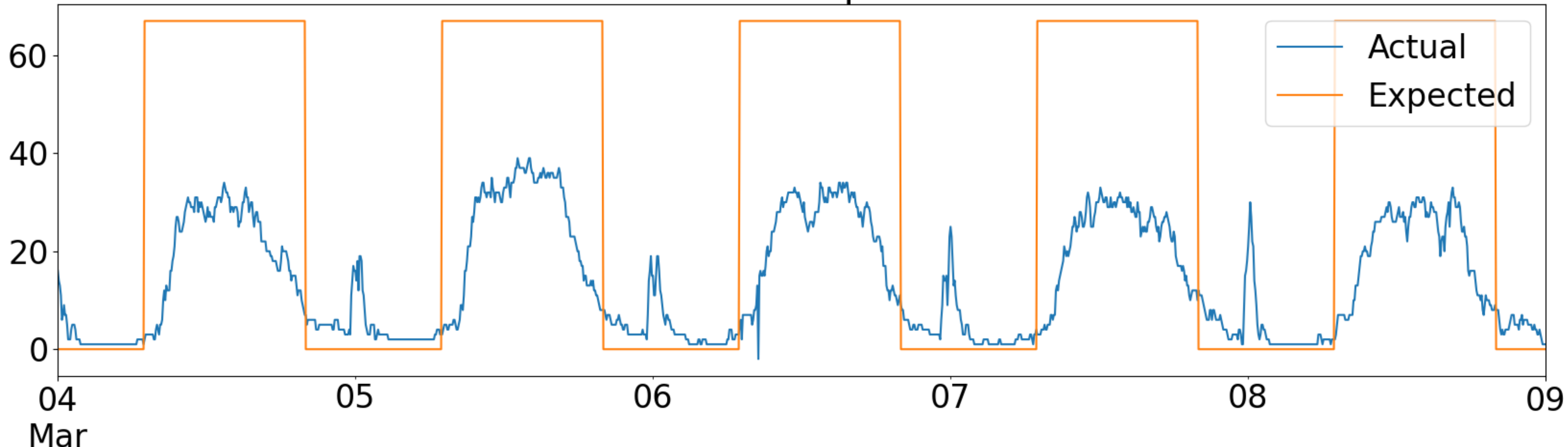
Significant energy waste in campus buildings

- Every room on campus was conditioned from 7am to 7pm (or 11pm, sometimes 7 days a week)
- On average, classes are only scheduled during 26.8% of these hours during semesters
- At maximum around half of the offices were occupied

Number of hours for each room



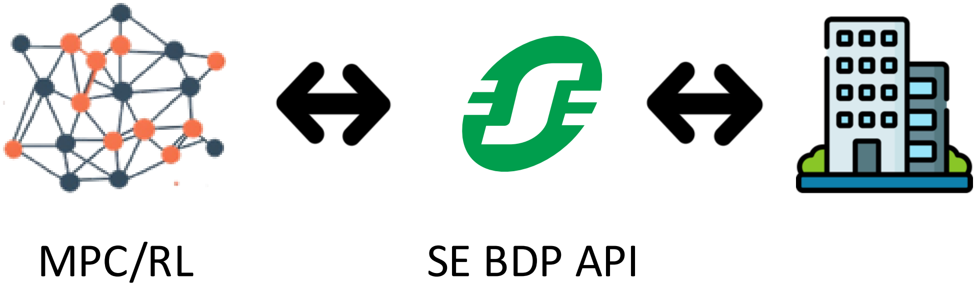
Number of occupied offices

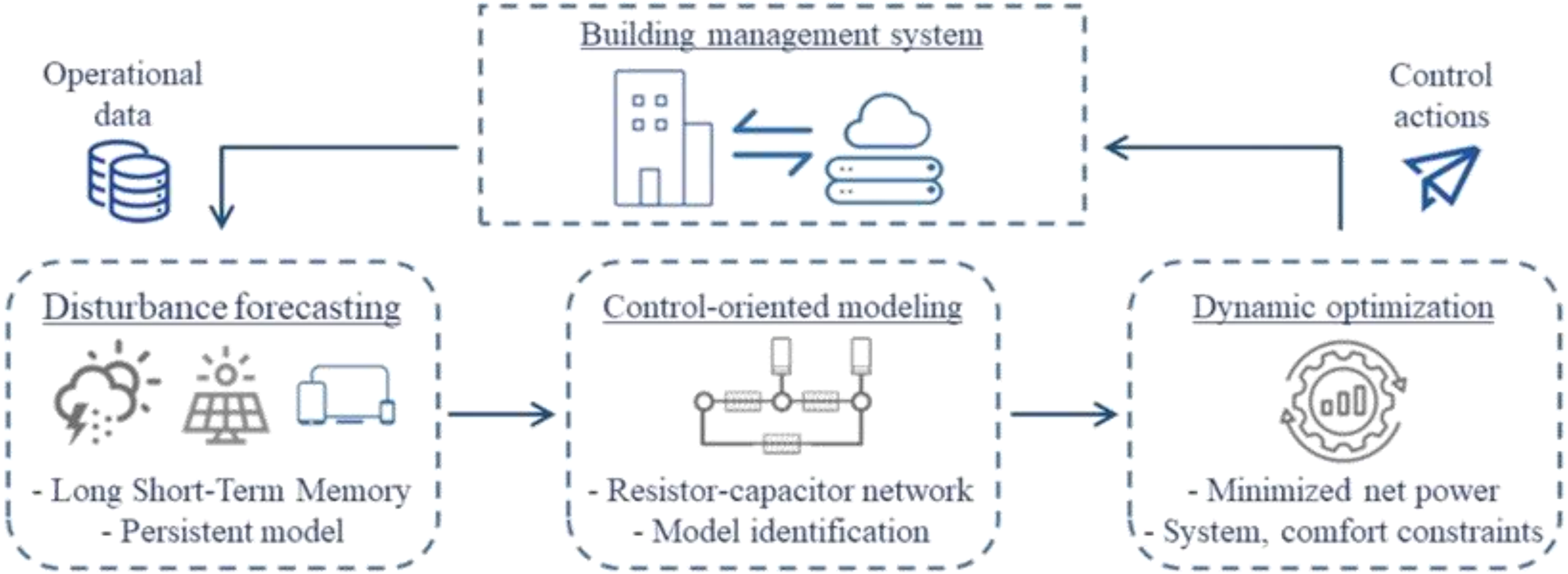


Pilot test of MPC

- Thermal response models for optimal predictive control

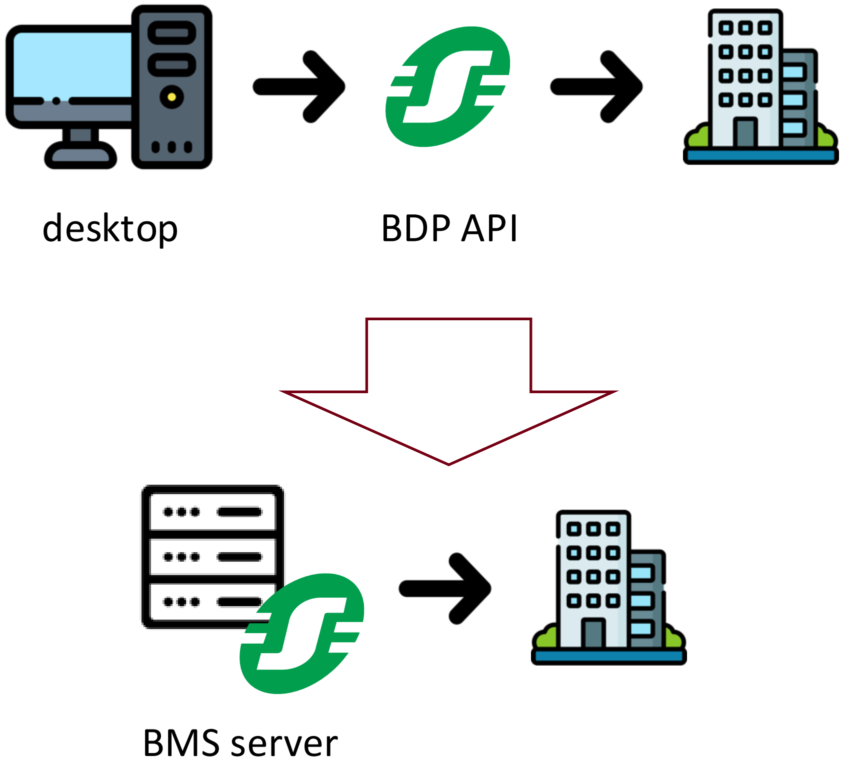
- Optimizing temperature setpoints in real-time
  - Minimize energy consumption given occupant-centric comfort constraints





Challenges to scaling up the optimal control

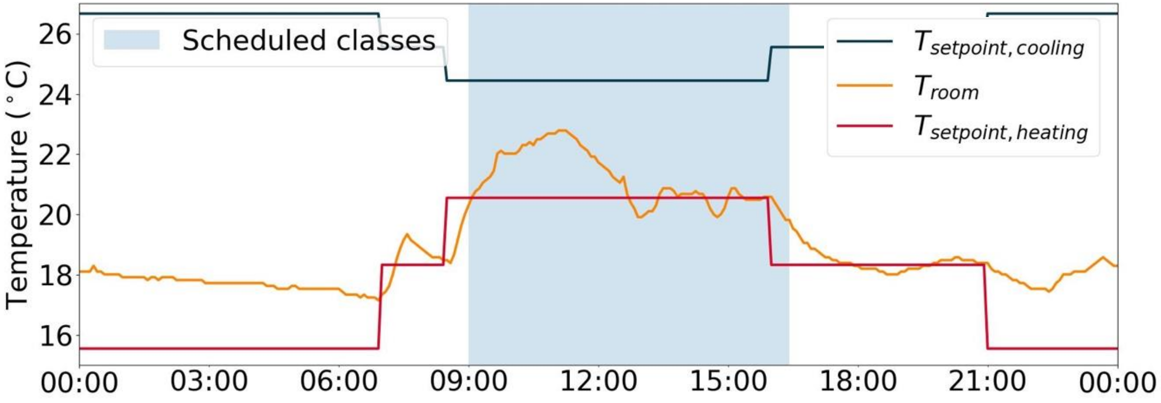
- Computational resource (maintenance)
  - From 1 building to 100 buildings
  - Data exchange costs and risks
- Sensor/data availability
  - Existing outdated infrastructure
  - Lack of historical data (additional data server needed)
- Do we need MPC to save energy?

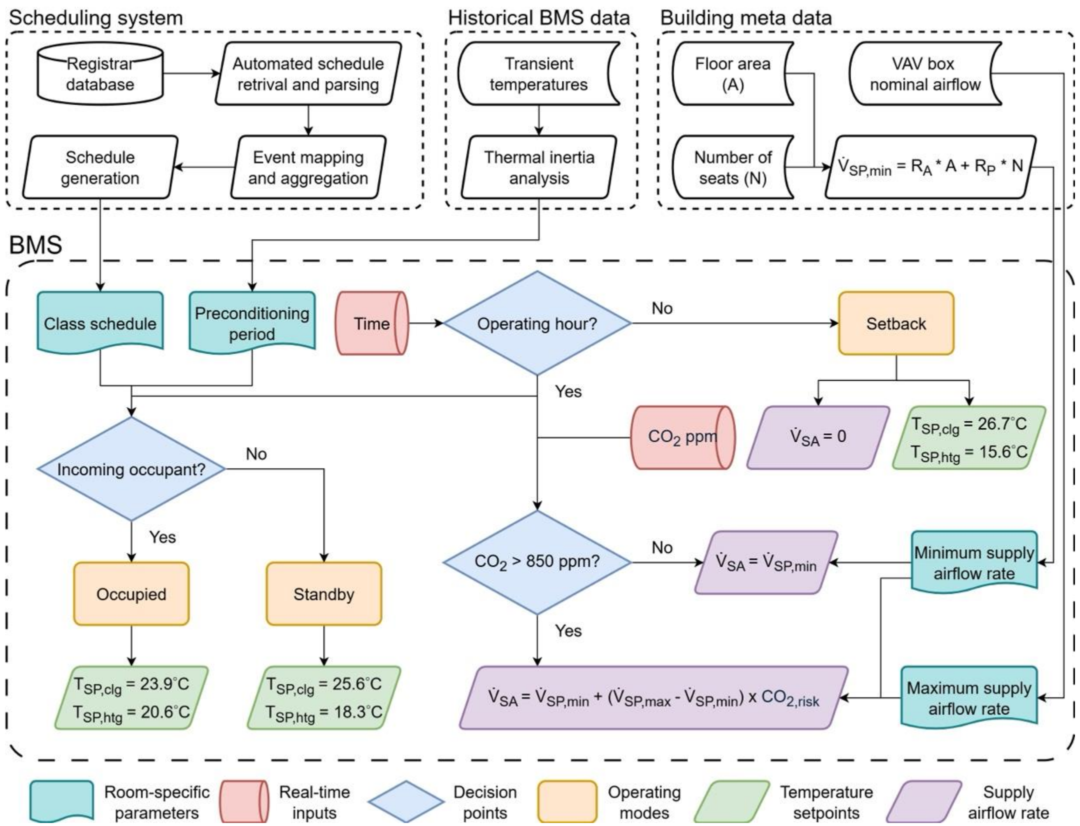


Rule-based occupant-centric control for classrooms

- Operating schedule informed by class registration
- Pre-conditioning set by historical data
  - Start from X hours before the first class
  - Comfortable environment when peoples come in
  - More free-floating during unoccupied operating hours

Example of schedule-based control

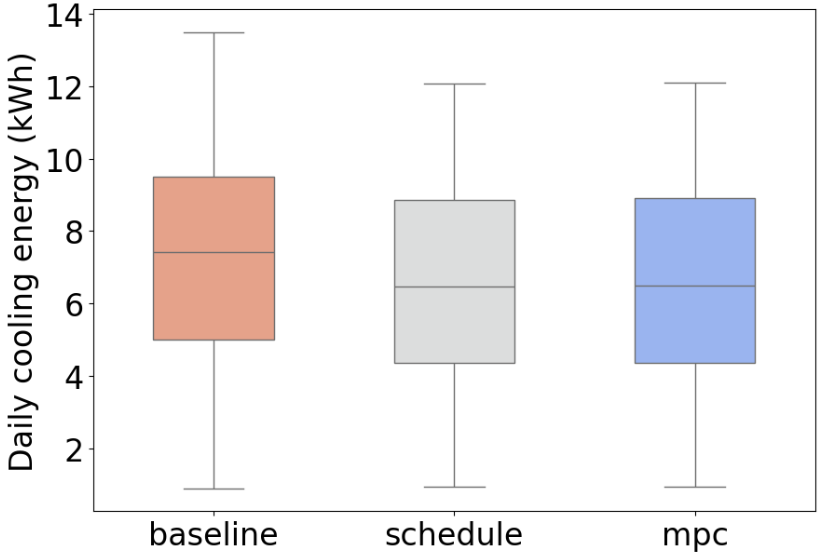


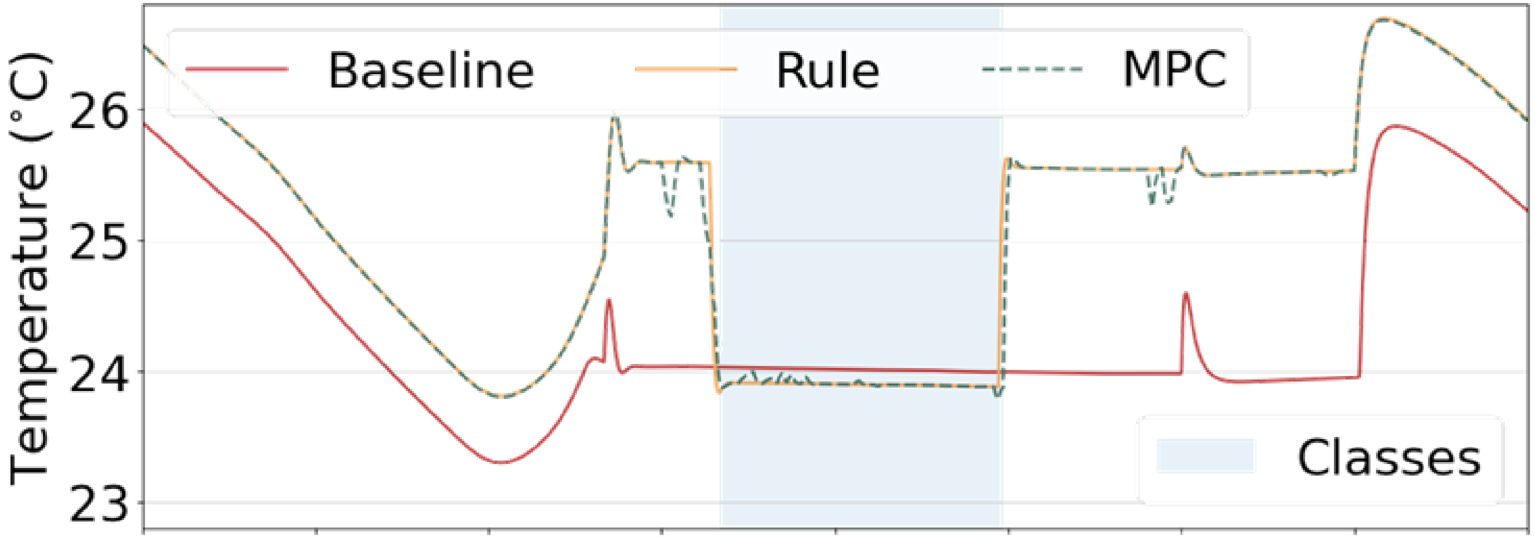


Performance comparison using BOPTEST (bestest_air)

- Comparison of control performance
  - Baseline: static day-to-day schedule
  - Schedule: mode switching based on actual schedules
  - MPC: minimize energy with occupancy-based comfort constraint

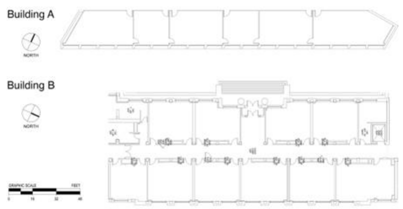
- Close energy saving of 11.2% (schedule) and 10.8% (MPC)
  - Given the same information, only minor differences in control actions



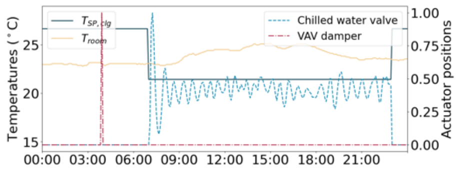


Significant savings by avoiding unnecessary HVAC usage

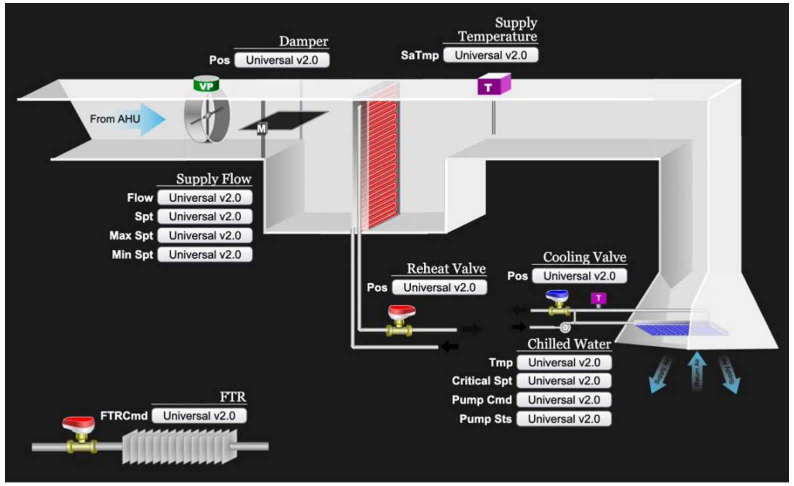
Class schedules



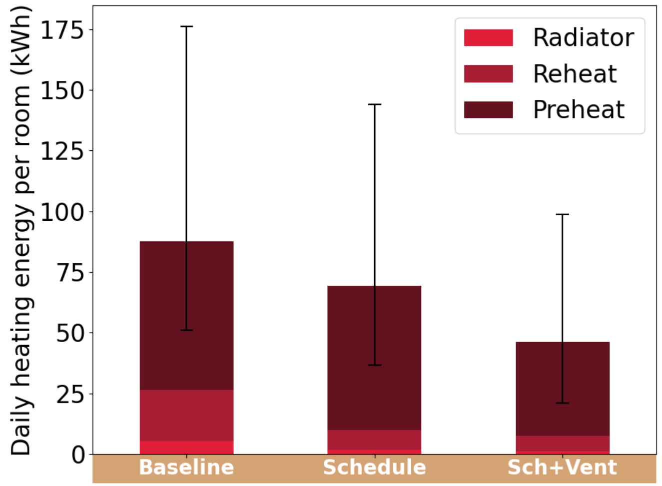
Drawings



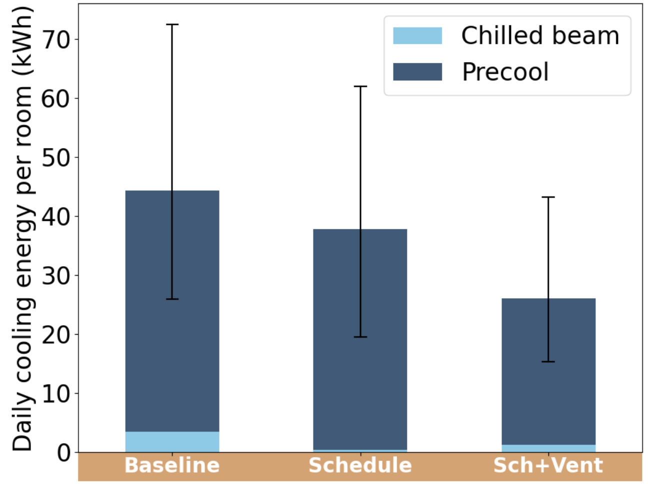
Historical data



Building management system



	Baseline	Schedule	Sch+Vent
Total kWh	87.66	69.34	46.05
Saved kWh	/	18.32	41.61
Saved ratio	/	**0.21**	**0.47**



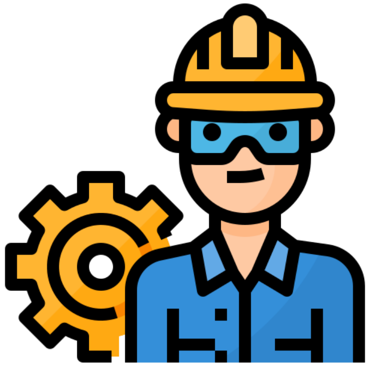
	Baseline	Schedule	Sch+Vent
Total kWh	44.27	37.75	25.98
Saved kWh	/	6.52	18.29
Saved ratio	/	**0.15**	**0.41**

Physics-informed models for MPC

BOPTEST role - customized prediction and control tests

Field implementation of optimal HVAC control

V.S.



What research papers highlight

- X% higher predictive accuracy
- Y kWh energy saving based on simulation/experiments
- Z% reduction in data requirements/ computational costs

What facility managers care about

- What's the payback period?
- Is it easy to set up, with few additional requests?
- **Does it offer robust control under all conditions?**

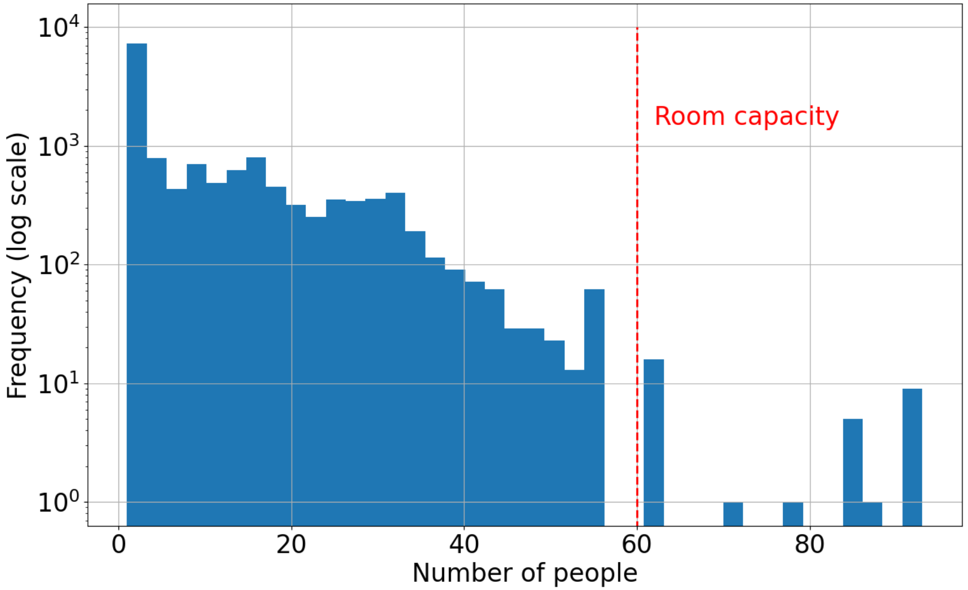
Overlooked corner cases

- Unseen, unexpected, or extreme situations that a system might encounter, potentially causing errors or failures in its operation (e.g. autonomous driving)

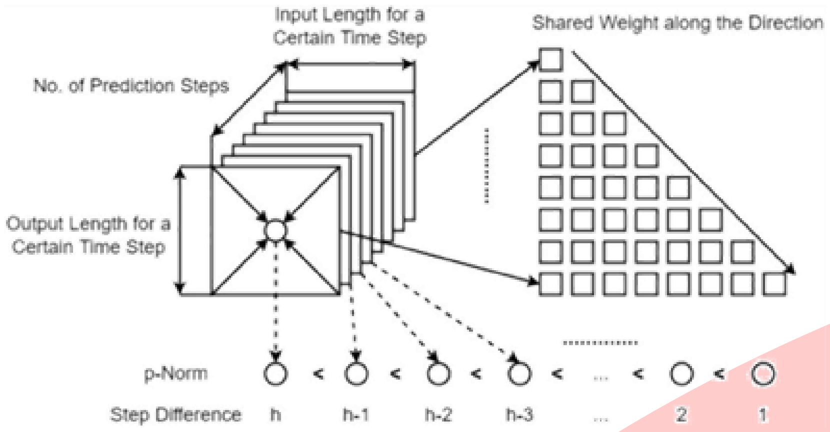
- For occupant-centric control, previous results focused on average KPI over a relatively long period

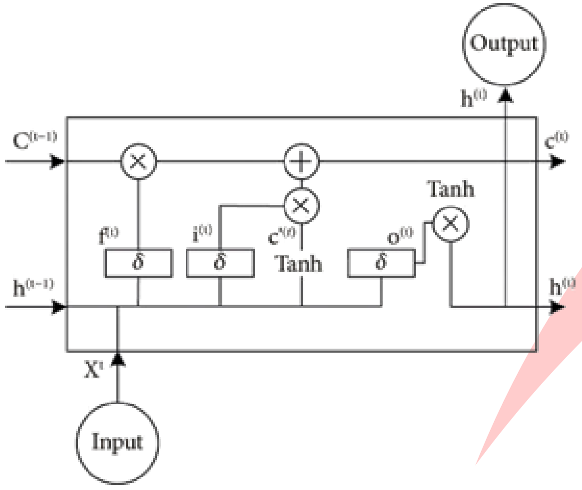
- More attention is needed for the robustness in unseen and extreme scenarios (e.g. weather, occupant)

- Informed decision-making requires scalable and extrapolatable predictive model



Integrating domain knowledge with machine learning





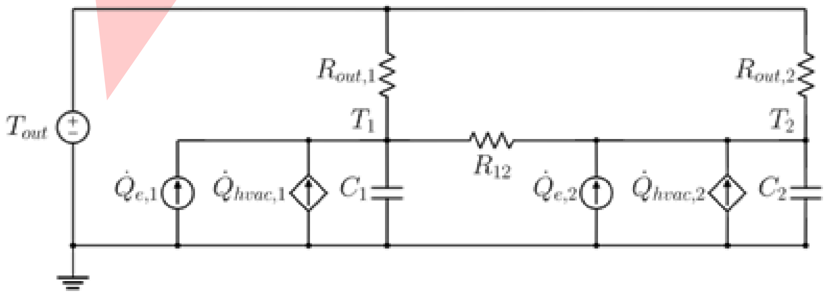
LSTM

PINN

PCNN

RC

What's the proper level of physics-integration?



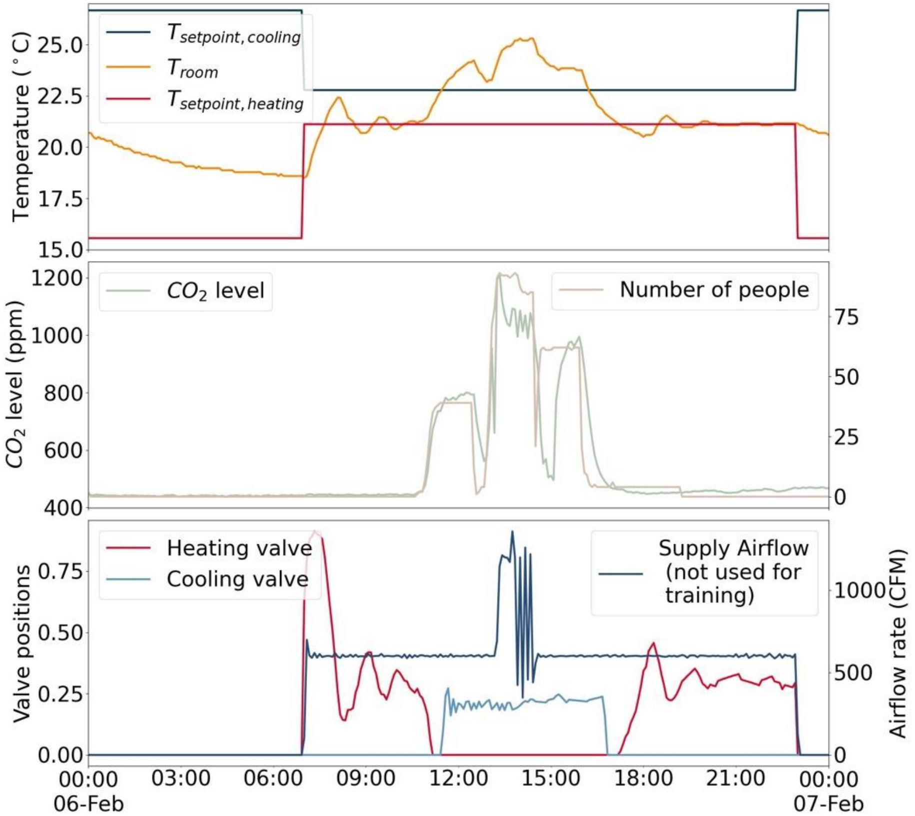
$$\mathcal{L}_{PINN} = \mathcal{L}_{MSE} + \lambda \mathcal{L}_{grad},$$

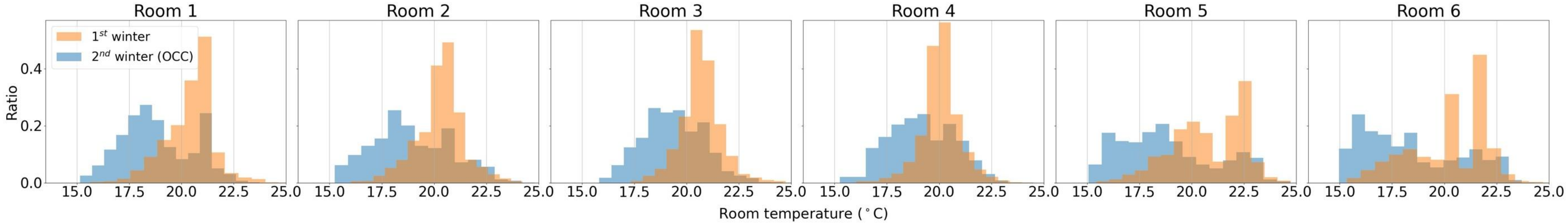
$$\mathcal{L}_{grad} = \frac{1}{l} \sum_{k=0}^{l-1} \left[\frac{1}{m} \sum_{z=1}^{m} g_k^z \right],$$

$$g_k^z = \sum_{y=1}^{m} ReLU\left(-\frac{\partial \hat{T}_l^z}{\partial u_k^y}\right) + ReLU\left(-\frac{\partial \hat{T}_l^z}{\partial T_k^{out}}\right)$$

Data and prediction tests

- Training and first testing dataset from 2023 winter: standard operation with static operating schedules
- Second testing dataset from 2024 winter: occupant-centric control driven by class schedules
- Corner case definition: temperature spike caused by an unexpected large group around noon



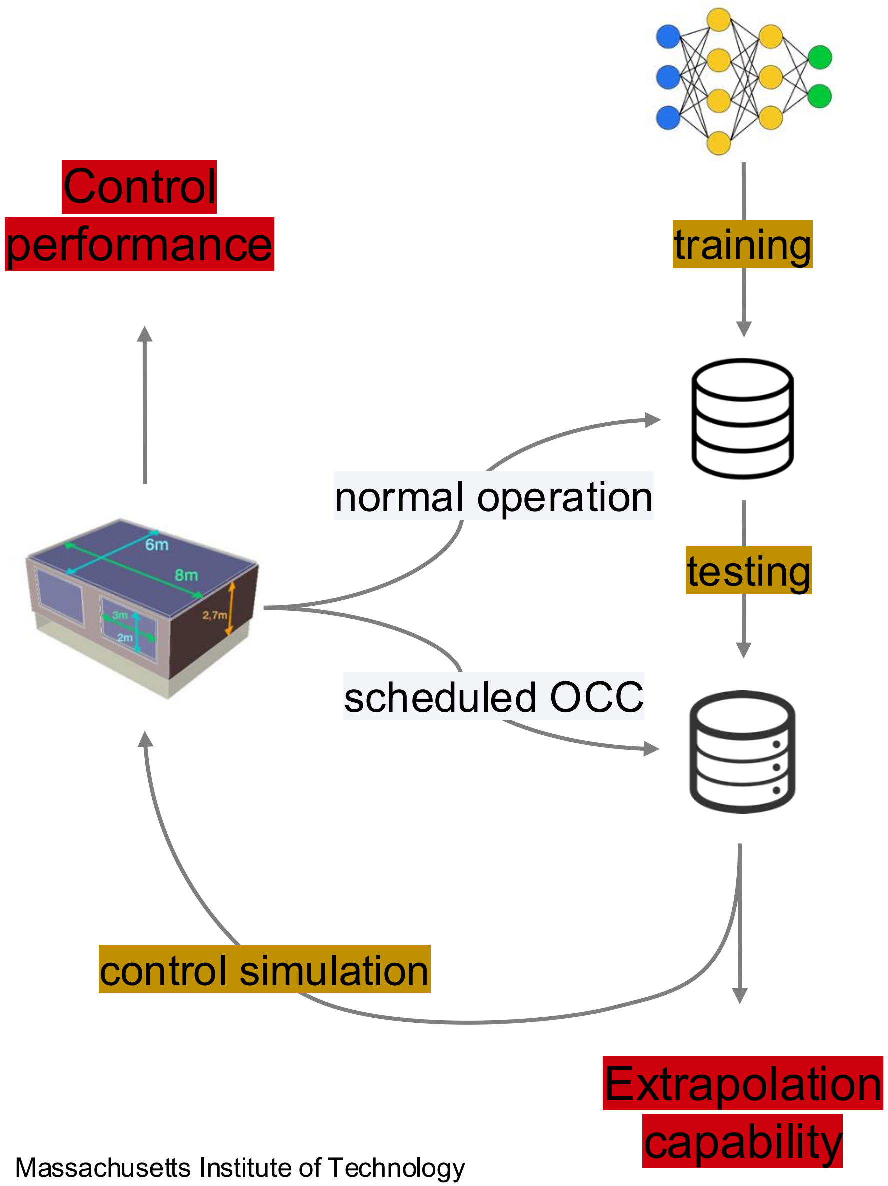


24 hour open-loop prediction

- PCNN had slightly larger fitting error, performed similarly in standard tests, but significantly better in extrapolation

Model	Mean absolute errors (°C)								
	Training[a]			1st case (standard)			2nd case (extrapolation)		
	by room	*std*	**mean**	by room	*std*	**mean**	by room	*std*	**mean**
LSTM	0.6, 0.65, 0.77, 0.88, 1.29, 0.79	*0.23*	**0.83**	0.69, 0.7, 0.82, 1.11, 1.25, 0.81	*0.21*	**0.89**	1.04, 0.99, 1.17, 0.88, 2.18, 1	*0.44*	**1.21**
PINN	0.54, 0.82, 0.62, 0.51, 1.44, 1.02	*0.33*	**0.82**	0.66, 0.93, 0.7, 0.76, 1.36, 0.95	*0.23*	**0.89**	0.99, 0.88, 0.95, 0.58, 2.19, 1.13	*0.51*	**1.12**
PCNN	0.7, 0.57, 0.66, 0.56, 1.08, 1.59	*0.37*	**0.86**	0.76, 0.65, 0.74, 0.74, 1.16, 1.22	*0.22*	**0.88**	0.69, 0.92, 0.6, 0.64, 1.38, 1.01	*0.27*	**0.87**
RC	1.46, 1.41, 1.19, 1.16, 1.59, 1.73	*0.2*	**1.42**[b]	1.34, 1.53, 1.31, 1.14, 1.48, 1.59	*0.15*	**1.39**	1.78, 1.97, 1.9, 1.62, 1.6, 1.82	*0.13*	**1.78**

Bridging prediction and control in BOPTEST



- Physics-informed models are usually hard to train
- Physics-integration could reduce training data requirement?
  - More physical model may require more “physical” data (e.g. non-ideal setpoint tracking)
- Only beneficial in tasks with long-enough horizons

Thank you!

Sicheng Zhan

szhan@mit.edu