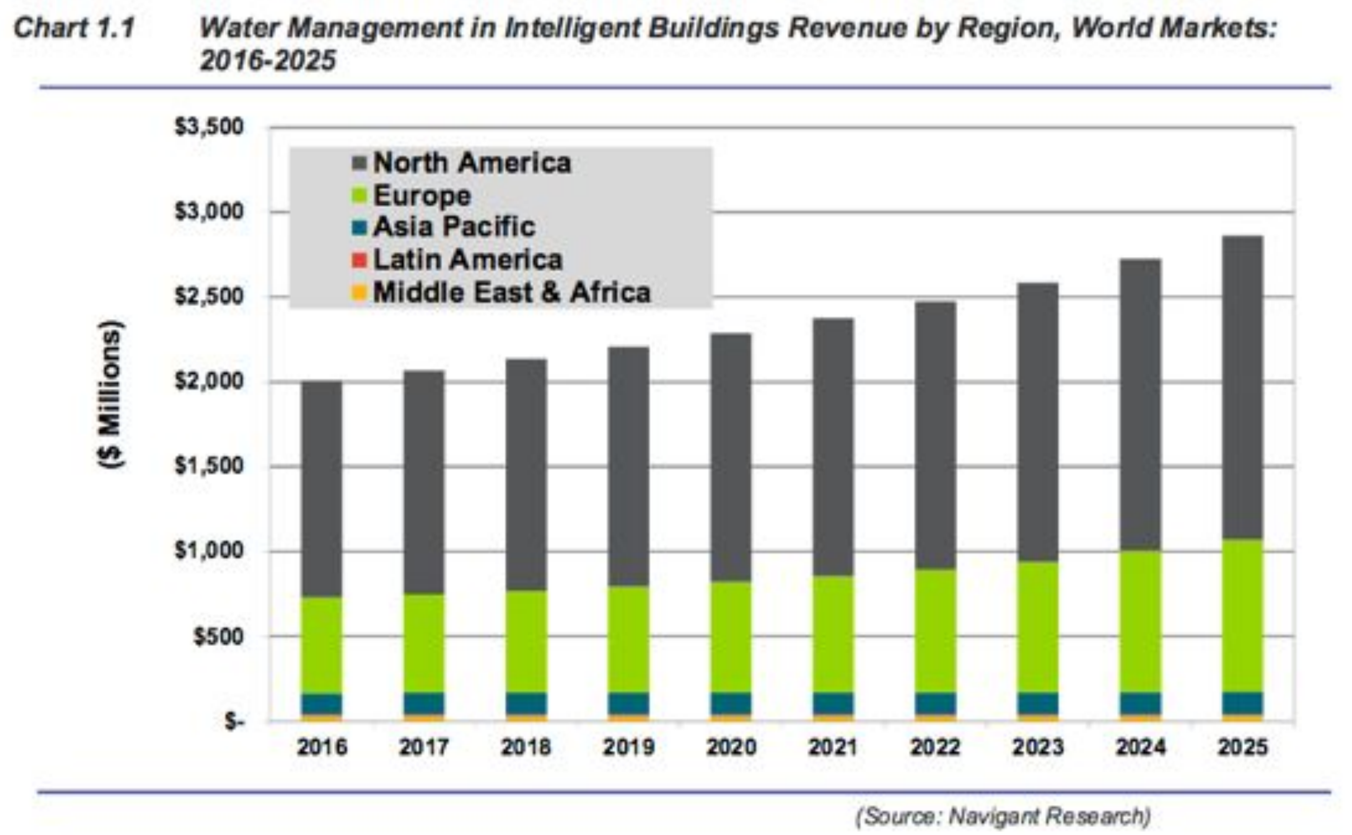
POWER GRID PROJECT

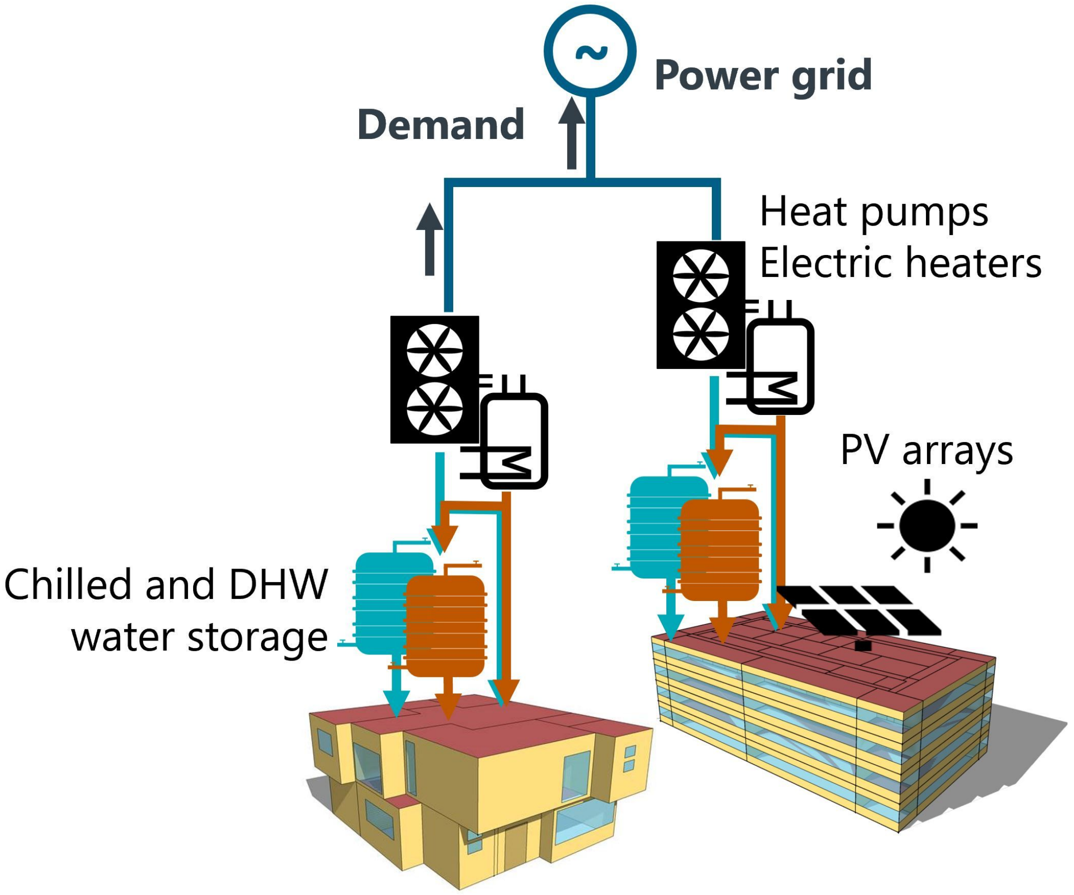
FELIPE BUCHBINDER, YOUNG KIM AND SHOTA TAKESHIMA

Smart building technologies are a huge and growing market. Investments in water management technologies alone is expected to grow from **$2.0 billion in 2016** to **$2.8 billion in 2025**

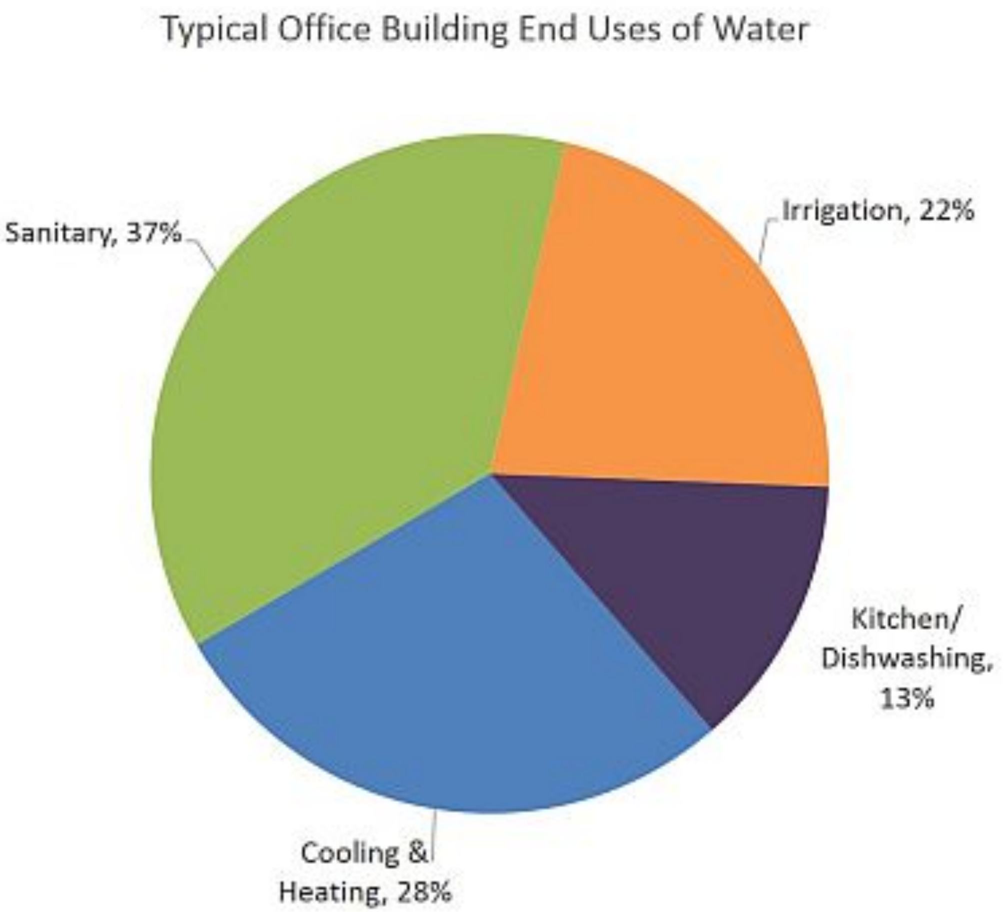


PROBLEM SETTING

- Buildings need to heat up or cool down.
- They do this by using hot or cold water.
- It costs electricity to warm or cool water.
- Future cost of electricity is affected by current demand and supply.
- If the cost of electricity is low, buildings can heat/cool more water than they need immediately and store it for future use.
- If the cost of electricity is high, buildings can use the hot/cold water they had previously stored.

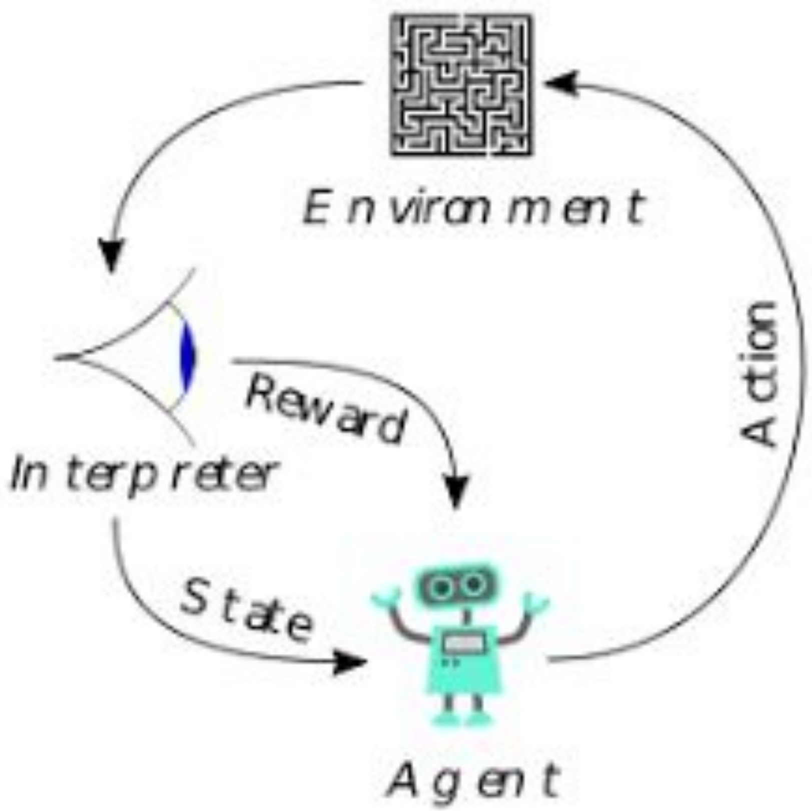


HEATING AND COOLING ACCOUNTS FOR 28% OF WATER CONSUMPTION IN OFFICE BUILDINGS

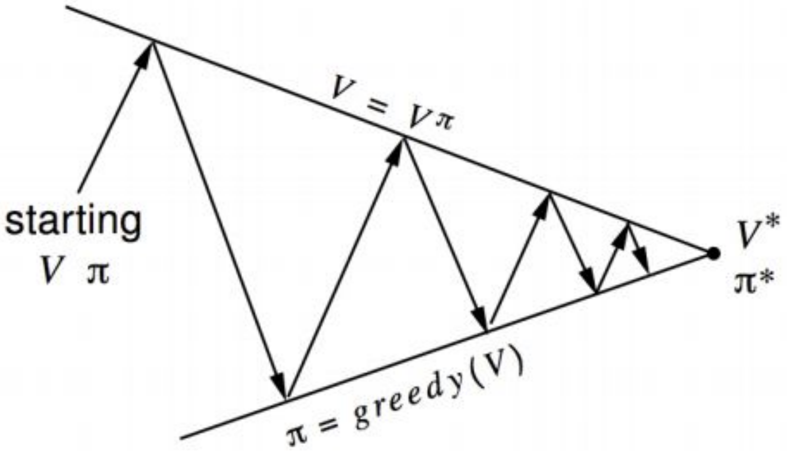


Source: United States Environmental Protection Agency. https://www.epa.gov/greeningepa/water-conservation-epa. Accessed: 11/02/2020

REINFORCEMENT LEARNING

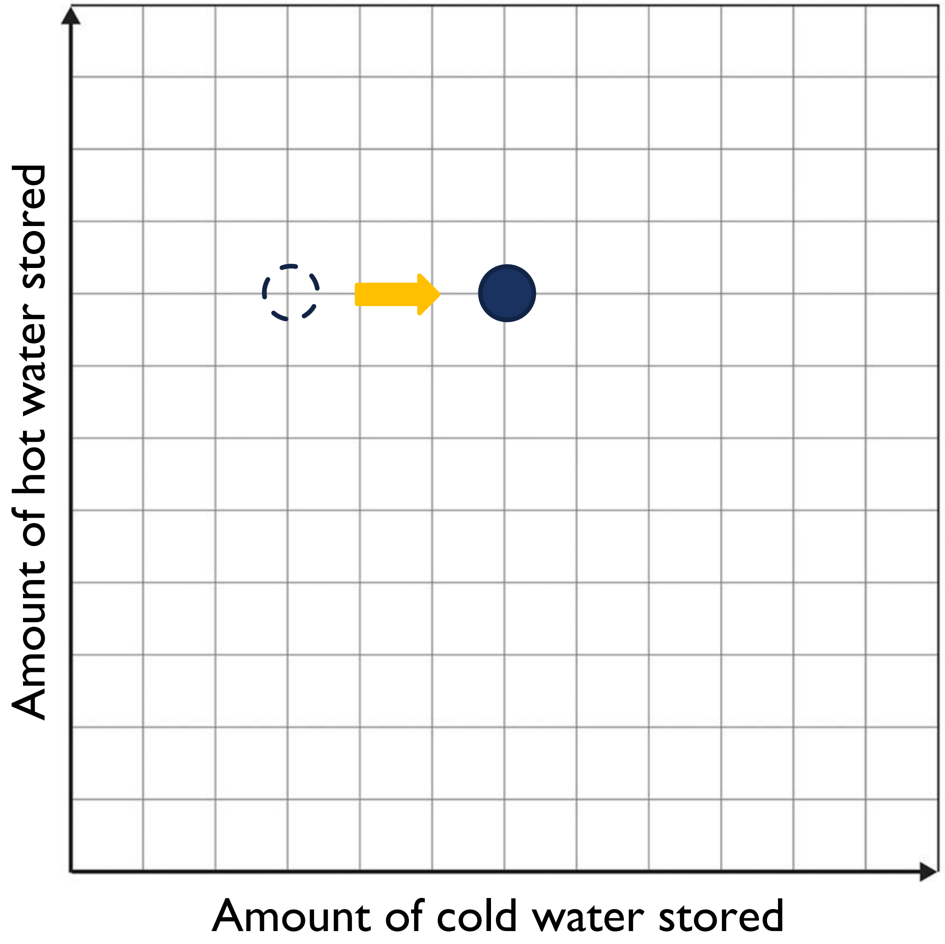


- Agent identifies current state. Takes action. Gets reward. Goes to new state.
- Goal is to maximize (discounted) value of future rewards (optimal policy).
- No knowledge of possible states or transition probabilities required (model free)
- Why use RL?



PROBLEM STATEMENT
AS A REINFORCEMENT LEARNING PROBLEM

- States: Amount of cold and hot water stored
- Actions: Increase / Lower stocks of cold / hot water
- Reward in terms of (negative) cost used to cool / heat building
- Goal: Find how much hot / cold water to use / store at each time in order to incur in the smallest cost (optimal policy)

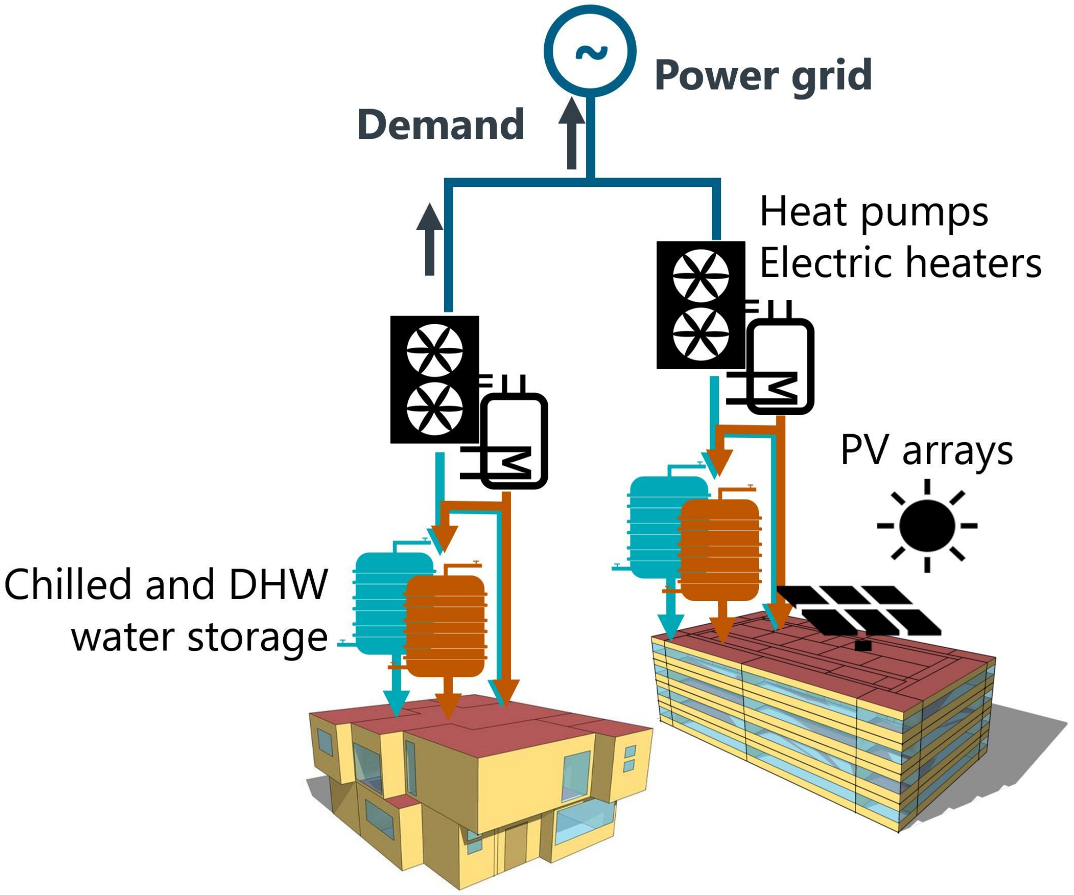


PROBLEM STATEMENT
AS A DEEP REINFORCEMENT LEARNING PROBLEM

- Knowing which action to take requires knowing the value of taking each action on each state (Q).
- But we have infinite states!
- Solution:
  1. encode states in terms of meaningful features
  2. using the states and actions we've seen during training period, fit a function that, given a state's features, returns the value of taking each action when in this state
  3. use this function to estimate the value of all states, even if we never visited them before

SIMULATOR ENVIRONMENT

- City Learn
  - “an open source OpenAI Gym environment for the implementation of Multi-Agent Reinforcement Learning (RL)”
  - “Its objective is to facilitate and standardize the evaluation of RL agents such that different algorithms can be easily compared with each other.”
- Each agent manages each building (possibly exchanging information)
- Optimize “glocally”



MARLISA: Multi-Agent Reinforcement Learning with Iterative Sequential Action Selection for Load Shaping of Grid-Interactive Connected Buildings

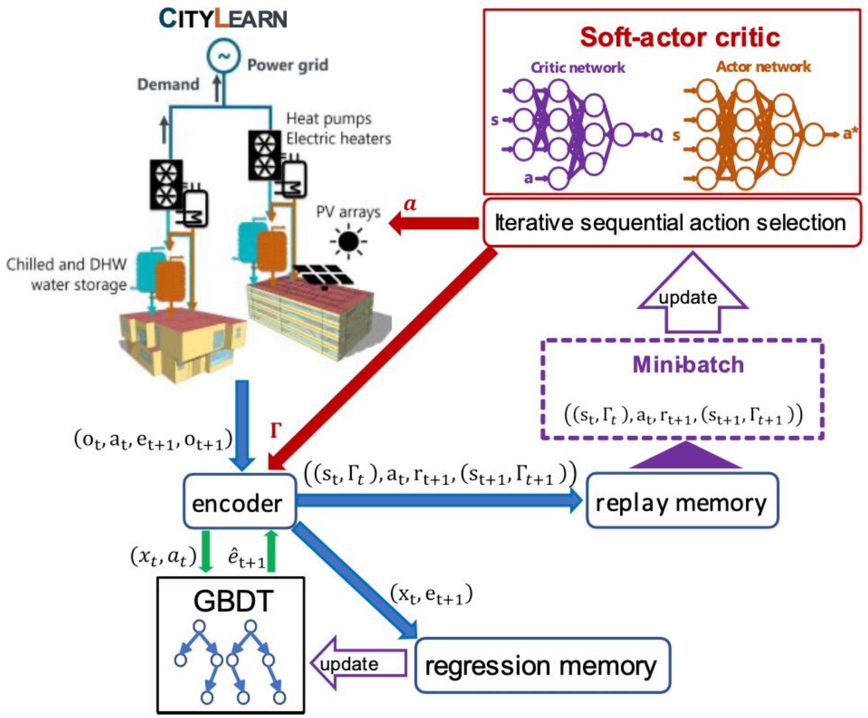


Figure 2 Simulation framework: integration of MARLISA into CityLearn

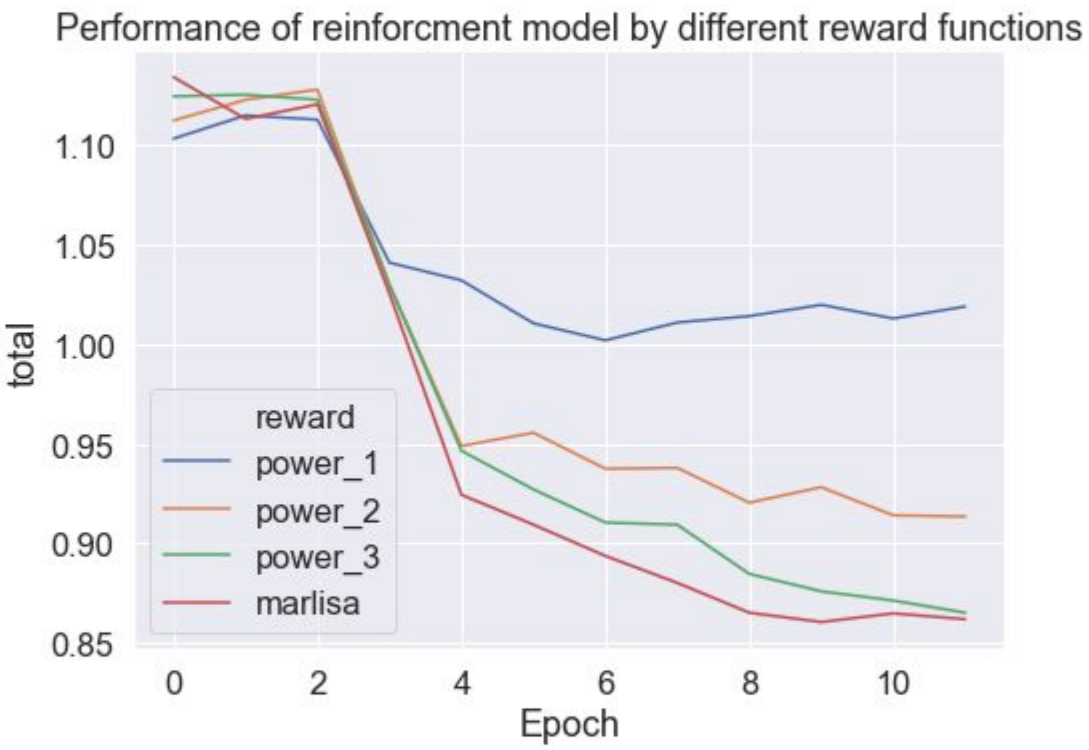
Table 2 Reward functions compared in this research

r_i^1	$min\{0, e_i\}$
r_i^2	$sign(e_i) \cdot min\{0, e_i\}^2$
r_i^3	$min\{0, e_i\}^3$
r_i^{MARL}	$-sign(e_i) \cdot e_i^2 \cdot \min\left\{0, \sum_{i=0}^{n} e_i\right\}$

r_i^{MARL}	$e_i > 0$	$e_i < 0$	$e_i = 0$
$\sum_{i=0}^{n} e_i \geq 0$	0	0	0
$\sum_{i=0}^{n} e_i < 0$	+	−	0

Source: Vázquez-Canteli, José & Henze, Gregor & Nagy, Zoltán. (2020). MARLISA for Load Shaping of Grid-Interactive Connected Buildings. 10.1145/3408308.3427604.

MULTIPLE AGENTS SIMULATIONS



Score

- ramping: sum(|e(t)-e(t-1)|), where e is the net non-negative electricity consumption every time-step.
- 1-load_factor: the load factor is the average net electricity load divided by the maximum electricity load.
- average_daily_peak: average daily peak net demand.
- peak_demand: maximum peak electricity demand
- net_electricity_consumption: total amount of electricity consumed

NEXT STEPS

- Try different techniques for encoder
- Try different techniques other than Gradient Boosting Decision Tree
- Incorporate volatility into reward function
- Incorporate pricing function into the simulation (optional)

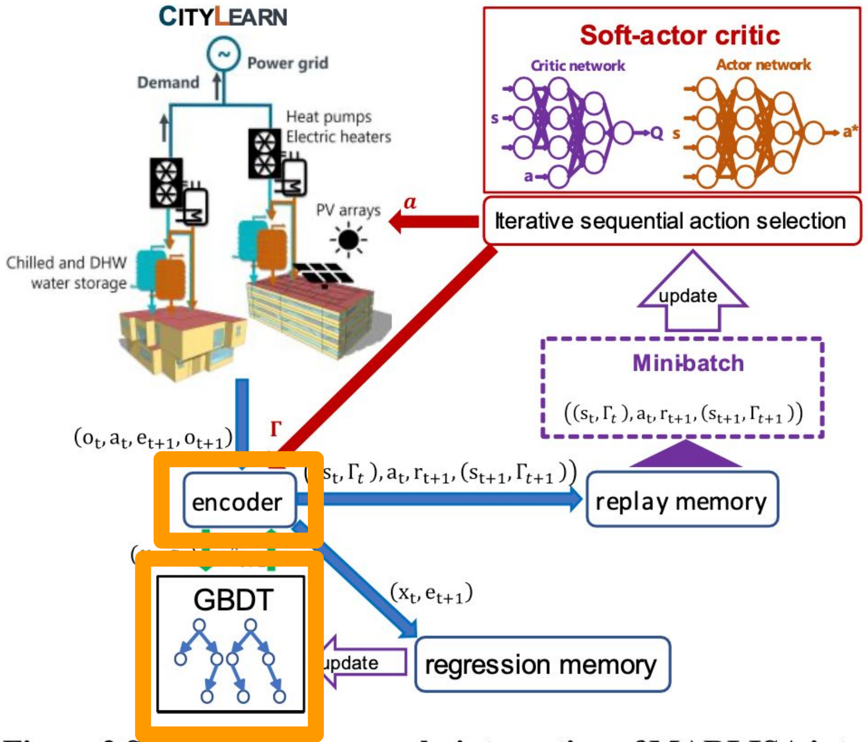


Figure 2 Simulation framework: integration of MARLISA into CityLearn

THANK YOU