

Notes Week 47

Dynamics independency of CityLearn 2023:

A choice has to be made to utilize pre-computed or pre-measured load demands from .csv building files to satisfy some pre-calculated indoor dry bulb temperature. It is mentioned that the heating and cooling loads are ideal, yet the given pre-calculated temperatures still indicate a temperature delta with the ideal temperature (the setpoint).

(ideal load refers to the energy that must be provided by an energy system to meet a control setpoint e.g. an air conditioner providing cooling energy to meet a cooling temperature setpoint of 22C in a room)

Found in the building class documentation (likewise for heating demand):

```
property cooling_demand_without_partial_load: numpy.ndarray
    Total building space ideal cooling demand time series in [kWh].
```

This is the demand when cooling_device is not controlled and always supplies ideal load

```
property indoor_dry_bulb_temperature_without_partial_load: numpy.ndarray
    Ideal load dry bulb temperature time series in [C].
```

This is the temperature when cooling_device and heating_device are not controlled and

When using the pre-measured indoor temperatures and heating/cooling loads, the heating and cooling loads have to be satisfied exactly in action space to keep consistent with building temperatures. If it is chosen to not use these pre-calculated actions for heating/cooling loads, the pre-measured temperatures can not be used. Therefore, if it is desired to not satisfy indoor temperatures like the pre-measured data one should choose to not use the pre-measured data.

Using the pre-measured data would change the action space to only charging and providing energy by energy storage devices, cooling and heating actions will be

taken from the .csv data.

Dynamics independent variables:

1. Neighborhood level variables:
 - Calendar type variables:
 - Month
 - Hour
 - Day type
 - Daylight saving status
 - Weather type variables:
 - Outdoor dry bulb temp (0h, 6h, 12h, 24h)
 - Outdoor relative humidity (0h, 6h, 12h, 24h)
 - Diffuse solar irradiance (0h, 6h, 12h, 24h)
 - Direct solar irradiance (0h, 6h, 12h, 24h)
 - District level carbon intensity (district = neighborhood)
2. Building specific variables:
 - Average unmet cooling setpoint difference
 - Indoor relative humidity
 - Non shiftable load
 - Solar generation
 - Electricity pricing (0h, 6h, 12h, 24h)
 - Indoor dry bulb temperature set point

Dynamics dependent variables:

1. Neighborhood level variables:
 - None
2. Building specific variables:
 - Cooling storage state of charge
 - Heating storage state of charge
 - Domestic hot water storage state of charge
 - Electrical storage state of charge
 - Net electricity consumption
 - Cooling device coefficient of performance
 - Heating device coefficient of performance
 - Cooling electricity consumption

- Heating electricity consumption
- Domestic hot water electricity consumption

Either taken from .csv file or calculated during runtime:

1. Neighborhood level variables:
 - None
2. Building level variables:
 - Indoor dry bulb temperature
 - Cooling demand
 - Heating demand
 - Domestic hot water demand
 - Indoor dry bulb temperature difference to set point
 - Power outage boolean (?)

Actions

Actions are real numbers between $[-1.0, 1.0]$ that prescribes the proportion of a storage device capacity that is to be charged or discharged. Here, -1.0 would mean to completely discharge the energy stored from full storage to completely empty. Likewise, 0.5 would charge the storage device halfway full.

The lower and upper bounds for the cooling storage, heating storage and domestic hot water storage actions are set to $(+/-) 1/\text{maximum_demand}$ for each respective end use, as the energy storage device can't provide the building with more energy than it will ever need for a given time step. . For example, if cooling_storage capacity is 20 kWh and the maximum cooling_demand is 5 kWh, its actions will be bounded between $-5/20$ and $5/20$. These boundaries should speed up the learning process of the agents and make them more stable compared to setting them to -1 and 1 .

- Cooling storage
- Heating storage
- Domestic hot water storage
- Electrical storage

Furthermore, we have actions that indicate the use of heating and cooling devices to satisfy heating and cooling needs or charging the storage devices, these can be read from csv files.

- Cooling device
- Heating device

In the building object documentation, we see an ***apply_actions*** function, where it is written:

Update cooling and heating demand for next timestep and charge/discharge storage devices.

The order of action execution is dependent on polarity of the storage actions. If the electrical storage is to be discharged, its action is executed first before all other actions. Likewise, if the storage for an end-use is to be discharged, the storage action is executed before the control action for the end-use electric device. Discharging the storage devices before fulfilling thermal and non-shiftable loads ensures that the discharged energy is considered when allocating electricity consumption to meet building loads. Likewise, meeting building loads before charging storage devices ensures that comfort is met before attempting to shift loads.

```
apply_actions(cooling_device_action: Optional[float] = None, heating_device_action: Optional[float] = None)
```

Parameters:

cooling_device_action (float, default: np.nan) – Fraction of cooling_device nominal capacity to be used for cooling

heating_device_action (float, default: np.nan) – Fraction of heating_device nominal capacity to be used for heating

cooling_storage_action (float, default: 0.0) – Fraction of cooling_storage capacity to be used for cooling

heating_storage_action (float, default: 0.0) – Fraction of heating_storage capacity to be used for heating

dhw_storage_action (float, default: 0.0) – Fraction of dhw_storage capacity to be used for heating

electrical_storage_action (float, default: 0.0) – Fraction of electrical_storage capacity to be used for charging

Supported Control Architectures

CityLearn supports centralized, decentralized-independent and decentralized-coordinated control architectures. In the centralized architecture, 1 agent controls

all storage, cooling and heating devices i.e. provides as many actions as storage, cooling and heating devices in the district. In the decentralized-independent architecture, each building has its own unique agent and building agents do not share information i.e. each agent acts in isolation and provides as many actions as storage, cooling and heating devices in the building it controls. The decentralized-coordinated architecture is similar to the decentralized-independent architecture with the exception of information sharing amongst agents.

Here we see the trade off between the centralized control and execution and its large action and state space, and the less intense action space of decentralized-coordinated architecture.