Efficient Scheduling of Smart Building Energy Systems using AI Planning

Houssam Hajj Hassan*, Jun Ma*, Georgios Bouloukakis*, Roberto Yus†, Ajay Kattepur‡
*Télécom SudParis, Institut Polytechnique de Paris, France
{houssam.hajj_hassan, jun_ma, georgios.bouloukakis}@telecom-sudparis.eu
†Dept. of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, USA, ryus@umbc.edu
‡Ericsson AI Research, India, ajay.kattepur@ericsson.com

***Abstract*—Buildings account for a significant share of global energy consumption, with Heating, Ventilation, and Air Conditioning (HVAC) systems being responsible for up to 60% of a building's energy usage. For this purpose, existing scheduling and control solutions can be used to design more sustainable energy systems and limit their environmental impact. However, these approaches mainly consider HVAC, ignoring other energy systems in buildings such as lighting control and plug loads. In addition, these solutions have to be customized for a specific building instance, hindering portability across different application domains. This paper presents a holistic approach for efficiently scheduling smart building energy systems through AI planning methodologies. AI planning enables decoupling domain knowledge from problem representations, enhancing portability and allowing for straightforward runtime adaptation when needed. We evaluate our approach in a smart office setting and show how AI planning enables reducing energy consumption by up to 30%.**

***Index Terms*—Energy Efficiency, Energy Systems Scheduling, AI Planning, Adaptation**

I. Introduction

Buildings consume a significant amount of energy, accounting for approximately one-third of global energy consumption [1], [2]. The environmental impact of energy consumption in buildings is increasingly recognized, leading to a growing demand for sustainable building practices. Improving energy efficiency in buildings can help reduce global energy consumption and associated greenhouse gas emissions, leading to a more sustainable and environmentally-friendly future. The Heating, Ventilation, and Air Conditioning (HVAC) system stands out as one of the most energy-intensive components within buildings, accounting for 38% of total energy consumption worldwide [2], [3], but reaching up to 60% in more extreme climate zones [2]. Nonetheless, it is crucial to acknowledge the influence of other energy systems within buildings, including lighting control, plug loads from electrical appliances and equipment, and building automation systems.

Optimizing these energy systems can yield a significant reduction, up to 60%, in overall energy consumption [4]. Moreover, the proliferation of sensors and Internet of Things (IoT) devices provides monitoring capabilities in buildings, which can be exploited to offer tailored and real-time optimization solutions according to the observed situations. Therefore, smart buildings can greatly benefit from scheduling solutions of energy systems operations to achieve more sustainable practices. For instance, motion sensors and video cameras may provide occupancy states of rooms and working spaces in office buildings; thermostats and humidity sensors can be used for thermal comfort purposes in residential buildings; and smart plugs can capture information about the energy consumption of electric equipment. Building administrators typically want to schedule the operation of energy systems in buildings for different purposes, e.g., meeting load demands, ensuring occupants' thermal comfort or reducing energy consumption. Currently, this process requires domain expertise and is performed for each energy system independently. This task becomes even more complicated when we consider the dynamicity of today's buildings. For instance, modern office buildings operate in a hybrid mode where employees work from home on certain days of the week, attend the office on other days, and occasionally gather in the office all together for specific events.

Designing sustainable and energy-efficient systems in smart buildings has gained significant attention recently. Most of the existing state-of-the-art solutions focus on the optimization and control of HVAC systems to reduce energy consumption [5]–[7]. The use of Artificial Intelligence (AI) and Machine Learning (ML) techniques is becoming more prominent lately, especially for buildings that support real-time monitoring with IoT devices and sensors [8]–[12]. However, such solutions largely ignore energy systems besides HVAC, and thus fail to capitalize on an *all-inclusive framework* to exploit the full potential of energy savings that can be achieved in buildings. In addition, data-driven approaches heavily rely on accurate and reliable data input. Any deviation from normal behavior during runtime results in sub-optimal solutions, necessitating the retraining of utilized models in such dynamic buildings. Finally, most existing solutions are custom-made, tightly coupled with the specific buildings' properties and characteristics, hindering their portability in different building types or different situations.

This paper presents the first approach, up to the authors' knowledge, that leverages AI planning [13] to offer a holistic solution for efficient scheduling of energy systems operations in smart buildings. In particular, AI planning enables the definition of generic domain knowledge including possible scheduling actions within a single AI-domain model that may be used for any smart building of the same type (e.g., airport buildings). Subsequently, the unique properties of a particular building, including spatial and contextual information, are encapsulated within an AI-problem file. An AI planner uses a domain model and problem file to generate an efficient energy system schedule that minimizes energy consumption. This allows defining a *single instance* of the domain model that may be paired with different problem files representing different buildings or building situations. The main contributions of this paper can be summarized as follows:

– Enabling cross-building portability of scheduling solutions by separating domain knowledge from problem representations.
– Enabling runtime adaptation of generated schedules through re-planning capabilities.

The rest of this paper is organized as follows. Section II provides an overview of the state-of-the-art for designing energy-efficient systems, as well as existing approaches relying on AI planning. In Section III, we present an overview of our proposed approach, and then go into the details of our solution in Section IV. Section V shows the experimental evaluation of our solution, and Section VI concludes the paper.

II. Related Work

This section provides an overview of the literature concerning the scheduling of energy systems to enhance operational efficiency. It then provides a comparison of these works against our proposed solution.

The intersection of Information and Communication Technology (ICT) and energy efficiency has garnered increasing attention in recent years [14]. Given its substantial energy consumption, optimizing the energy efficiency of HVAC systems in buildings has become a priority, and several strategies have been developed for this purpose. Some prevalent occupancy-based approaches include setting back the thermostat during unoccupied hours and at times when the building is vacant [5]. Other works focus on saving energy using systems such as lighting control, window shading, and controlling plug loads [4]. For instance, [9] controls lighting systems by estimating the behaviors of occupants in residential buildings. Other existing approaches control HVAC systems based on thermal comfort feedback provided by occupants [6]. More sophisticated solutions include the application of Model Predictive Control (MPC) and Adaptive Predictive Control Strategies (APCS) [8]. Such strategies enable scheduling HVAC operations efficiently by relying on their capability of forecasting dynamic future conditions [15]. For instance, [10]–[12] utilize occupant number prediction to adjust HVAC and save energy. Reinforcement learning (RL) is another technique used as a scheduling solution for HVAC systems, as described in [16]. These algorithms learn from experience and can adapt to changes in the environment, making them a suitable approach for HVAC systems. RL is specifically utilized to optimize energy efficiency and thermal comfort. Other scheduling solutions for HVAC systems include rule-based scheduling, heuristic scheduling, and genetic algorithms [5]. These approaches have been shown to be effective in various types of buildings and environments and have been the subject of extensive academic research.

Finally, Digital Twin (DT) approaches for buildings offer significant potential for enhancing HVAC performance by providing a dynamic and real-time simulation environment [7]. In [17], authors propose the application of DT to an energy recovery ventilation unit to improve the operational efficiency of the HVAC system. Additionally, [18] proposes a DT framework for HVAC systems to improve thermal provision ability and energy efficiency.

Despite the above efforts, there has been a relative lack of focus on cross-building portability of the developed solutions and dynamic scheduling of energy systems. Current data-driven methods often overlook the potential for a generic, adaptable building model, leading to the need for extensive, scenario-specific modifications. Such prolonged and frequent adjustments not only increase the complexity but also contribute to additional usage of computational resources and energy resulting from training and running AI/ML models [19]. Thus, this work provides a solution for scheduling of energy systems (including HVAC) using AI Planning that focuses on cross-building portability and runtime adaptation. Efficient schedules for different runtime situations of buildings will also be considered (e.g., scale up rooms, occupants etc.).

III. Overview

This section provides two motivating scenarios to highlight the needs and challenges associated with providing efficient scheduling solutions for energy systems operations. Then, it introduces how we leverage AI planning for generating schedules for efficient energy systems operations.

A. Smart Building Energy Systems Scheduling

We consider the case of a smart office building, where employees adopt a hybrid working mode (remote and on-site). This means that the number of employees present in the building in any given day may change, depending on the number of people opting to work from home. For such cases, current practices require generating energy systems operations schedules on a day-to-day basis. Smart technologies can be used for generating such schedules, especially that they have the potential to achieve energy savings that go up to 60% [4]. Indeed, IoT devices and data can be exploited to get insights, analyze patterns and provide more efficient schedules for the various energy systems that co-exist in the building. For example, in office buildings, rooms' occupancy patterns can be learned from sensor data to efficiently schedule HVAC systems, control lighting, and provide the needed electric load according to the number of occupants and activities taking place in different rooms throughout the day. Note that we consider that data provided by devices is accurate, and handling faulty sensing situations is out of the scope of the paper.

However, in smart buildings, providing a holistic solution for scheduling energy systems operations is not a trivial task and requires addressing multiple challenges. First, even though there exist approaches for automating the scheduling of energy systems operations, they usually target HVAC systems only, and hence do not capture the potential efficiencies that can be achieved by integrating and coordinating all energy systems within a building. Second, buildings can be dynamic environments where space properties are constantly changing. For example, the number of occupants in rooms can change on an hourly or daily basis. In addition, the carbon intensity may change depending on the electricity provider [20]. This requires providing scheduling solutions that are flexible enough to enable frequent re-configuration when certain properties of buildings change, in a timely and cost-effective manner. Finally, most existing solutions have to be tailored to one specific instance of a building. This happens either by training machine learning models on a representative dataset of a building, or by embedding buildings' properties and characteristics into custom-made solutions. This hinders the portability of such approaches, and solutions may not be readily applied for different systems and in different situations. For instance, solutions targeted towards office buildings may not be directly applied to airport buildings, and may necessitate re-training models, revising some aspects to take additional needs into consideration.

B. Power Generation Scheduling of Hydropower Plants

Another case we consider is a hydropower plant, where appropriate power generation scheduling - referring to the strategic planning of when and how much electricity to generate - is fundamental to reducing resource waste and increasing equipment lifespan. However, providing a comprehensive solution for hydropower plant scheduling is not straightforward and involves multiple challenges. Firstly, seasonal changes and weather conditions, as well as the highly nonlinear nature of the rainfall-runoff process makes it unfeasible to rely solely on historical data and real-time weather forecasts to predict variability of water inflow in extreme weather which is essential for estimation of energy generation. In addition, the fluctuations in electricity demand, environmental impact concerns and maintenance requirements significantly influence the plant's ability for consistent power generation and require adapting the schedules according to specific situations. Existing approaches often focus on specific aspects of the operation [21], such as optimizing generation based on water inflow prediction or volume of water in reservoirs, without integrating other considerations. In addition, many current solutions are customized for specific plants, trained based on private and local historical data.

C. Solution overview

To address the aforementioned challenges, this paper presents an approach for efficient scheduling of energy systems operations in smart buildings by leveraging AI Planning methodologies. AI Planning is a model-based technology devoted to decision making, which can be used in a variety of application domains. Traditionally, robotics [22] has been a paradigmatic application area, but other uses including flexible

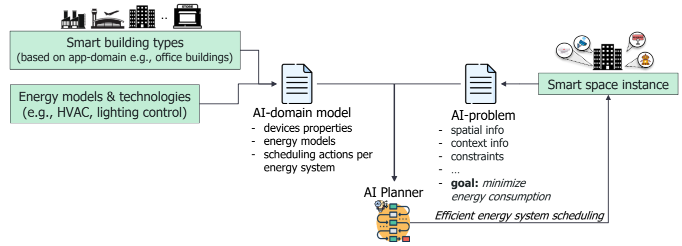

Fig. 1: Cross-building AI planning for efficient scheduling of energy systems.

manufacturing [23], agriculture and process management [24], and adaptive management of IoT Data flows [25] can be found in the literature. There are very few works utilizing AI planning for enhancing energy efficiency. In [26], ontologies are used for user activity recognition, and planning based on Hierarchical Task Networks is used for controlling device operations using sensors and actuators. [27] uses AI planning for controlling operations in network systems while reducing energy consumption.

The novelty of using AI planning for scheduling energy systems operations lies in the separation of domain knowledge representation and problem instances. In particular, as shown in Figure 1, AI planning defines a domain file – also called *AI-domain model* – that contains generic information to represent any smart building type (e.g., airport buildings). These include general properties of IoT devices that may be deployed, energy models used to evaluate energy consumption [28], and actions for scheduling the operations of various energy systems. Because it encapsulates generic information, the domain file is defined once per building type. Then, the problem file – also called *AI-problem* – captures information related to a specific building instance under a specific situation (e.g., a typical working day in a university building). The file includes spatial information such as the number and type of rooms in a building, context information provided by IoT data (temperatures inside rooms, occupancy levels in rooms, etc.), constraints that have to be taken into account (e.g., temperature bounds for thermal comfort), and other properties of the building instance. In addition, the problem file includes defining a goal state, e.g., reducing energy consumption or carbon footprint. We explain in detail how the domain and problem files are designed in Section IV.

The domain and problem files are then fed to an AI planner [29] that generates a plan consisting of a schedule for the various energy systems, aiming to minimize energy consumption. For this purpose, the planner leverages algorithms based on forward/backward chaining search and/or heuristics (e.g., A^* algorithm) to find a plan that optimizes metrics defined in the problem file. The specific algorithm used for solving the planning problem depends on the planner; for example, the LPG [30] planner relies on local search to find plans, while Metric-FF [31] is built on top of the Fast-Forward planning system [32]. To provide schedules for different building instances, only the problem file that contains buildings' properties needs to be modified. Consequently, the primary advantages of using AI planning over other approaches for scheduling energy systems operations include: (i) providing an energy-efficient schedule that integrates and coordinates all energy systems by defining appropriate actions that can be performed, (ii) enhancing portability through decoupling of domain knowledge and problem representations, and (iii) enabling adaptation under different situations through re-planning.

IV. Efficient Scheduling of Energy Systems

This section presents the planning methodology for efficiently scheduling energy systems in smart buildings. While this involves using state-of-the-art AI planners as a black box to provide schedules for efficient energy systems operation, defining the AI domain knowledge and problem files is not a trivial task, since we have to capture context information, runtime behavior and goals. We start first by providing an overview of automated planning systems. Then, we show how domain models and problem instances are created for generating schedules in energy systems.

A. AI Planning

An AI planning system ("planner", for short) takes a problem formalisation, or model, as input and uses some problem solving technique, such as heuristic search, propositional satisfiability, or other, to work out its solution. [29]. The descriptive models used by planning systems are called *planning domains*. These include a description of the planning environment, i.e., *states*, and *actions* that can be taken by the planner in order to reach a certain *goal*. In addition, a *cost* can be associated with one or more actions. Formally, we define a planning domain as follows:

Definition 1: **Planning Domain.** A planning domain is a state transition system $\Sigma = (S, \mathcal{A}, \gamma, \mathcal{C})$, where:

- S is a finite set of states of the system. These refer to the states of the energy systems under consideration (e.g., temperature setpoints, luminosity levels).
- $\mathcal{A}$ is a set of actions that may be performed by an agent. Actions are used to alter systems based on conditions and effects (e.g., turning ON/OFF HVAC systems, increasing maximum plug load).
- $\gamma : S \times \mathcal{A} \to S$ is the state transition function. If $\gamma(s, \alpha)$ is defined then action α is applicable to state s, with $\gamma(s, \alpha)$ being the predicted outcome. For example, turning ON the cooling system in a specific room may lead to a decreased temperature in that room.
- $\mathcal{C} : S \times \mathcal{A} \to [0, \infty)$ is a cost function with the same domain as γ. It can represent a cost function minimizing monetary cost, energy consumption or other parameters that have to be optimized.

Given a planning domain, we can then define one or more planning problems $P = (\Sigma, s_0, G)$ where Σ is a state-transition domain, s_0 is the initial state, and G is a set of ground literal goals. The goal state is typically the desired final state of the system, for example meeting certain energy demands, or achieving energy savings while taking into account people's thermal comfort preferences. We propose a planning problem to generate a schedule for the operations of energy systems aiming to reduce energy consumption. For the sake of clarity and conciseness, we only consider temperature and light control systems in the provided examples. However, note that the definition is generic enough to cover any energy system. To define the planning problem, we first model a planning domain (defined earlier) and an initial state s_0, where all energy systems are turned off. s_0 also includes information about building properties to be taken into account when scheduling energy systems, such as the number of occupants in each room r_i at each time t, $occ_{r_i}(t)$, the inside temperature $\theta^{in}_{r_i}(t)$ inside each room, the outside temperature $\theta^{out}(t)$, and the lighting levels $\phi_{r_i}(t)$ inside rooms at a given time. In addition, constraints such as inside temperature bounds θ_{min} and θ_{max} and minimum lighting level ϕ_{min} are defined. The goal state G is one where all constraints are met

$$\begin{array}{ll} [1] & \theta^{in}_{r_i}(t) >= \theta_{min} \\ [2] & \theta^{in}_{r_i}(t) <= \theta_{max} \\ [3] & \phi_{r_i}(t) >= \phi_{min} \end{array} \tag{1}$$

An AI planner takes as input a planning domain and a planning problem, and finds a solution consisting of a set of actions that transform the system from the initial state to reach the final desired goal state.

Definition 2: **Plan.** A plan is a finite set of actions:

$$\pi = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle \tag{2}$$

where the plan's length $|\pi|$ is n, and its *cost* is the sum of the action costs: $cost(\pi) = \sum_{i=1}^{n} cost(a_i)$. A plan π is applicable to a state $s_0 \in S$ if there are states $s_1, s_2, \ldots, s_n$ such that $\gamma(s_{i-1}, \alpha_i) = s_i$ for $i = 1, \ldots, n$. In this case, $\gamma(s_0, \alpha_\pi) = s_n$ (with α_π being the last action in plan π). A solution for P is a plan π' such that $\gamma(s_0, \alpha_1) \ldots \gamma(s_m, \alpha'_\pi)$ satisfies G.
In our case, the planner generates a plan $\pi = \langle \alpha_1, \alpha_2, ..., \alpha_n \rangle$ where the total energy consumption ϵ_{total} is minimized, and all goal conditions are met. π is a schedule consisting of a series of actions to configure energy systems at several timesteps, according to the initial situation of the building, defined in s_0. Techniques used to generate plans from the initial state to the goal state include: (i) graph based search techniques, (ii) state-transition systems, (iii) constraint solvers that make use of symbolic predicates, constraints and effects, (iv) heuristic approximations such as removing negative predicates. Comparison of scale, benchmarking and speed of AI planning solvers has been evaluated within the International Planning Competition held bi-annually since 2000[1]. The planners (classical, temporal, uncertainty tracks) have been compared based on the scale of problems, heuristic planning accuracy and time taken to solve benchmark problems.

[1] https://www.icaps-conference.org/competitions/

B. Scheduling Energy Systems with AI Planning

To express planning domains and problems, we use the *Planning Domain Definition Language* (PDDL) [29]. PDDL is an action-centered language that enables defining states (also called predicates), and actions with preconditions and effects. PDDL divides the definition of a planning problem into two parts: the *domain* defines the state variables and actions, while the *problem* defines the initial state of the environment and the goal conditions, as well as one or more metric to be optimized. The same domain description may be paired with multiple problem instances, with varying grounded objects, initial conditions, goals, and cost functions.

Listing 1: PDDL Domain File

```
1  (:types room sensor window system property -object
2          hvac window_shading lighting_control -system
3          temperature_sensor light_sensor -sensor
4          occupancy -property))
5  (:predicates
6      (hvac_on ?h -hvac)
7      (open ?ws -window_shade)
8      (light_on ?l -lighting_control)
9      (temperature_set ?h -hvac)
10     (occupied ?r -room)
11     ...)
12 (:functions
13     (number_occupants ?o -occupancy)
14     (temperature_inside ?r -room)
15     (temperature_outside)
16     ...
17     (lighting_level ?r -room)
18     (plug_load ?r -room)
19     (energy_consumption)
20 (:action temperature_setting_cooling_21
21     :parameters (?o -occupancy)
22     :precondition (and (>=(time)0)
23         (=(occupancy_type ?o)1)
24         (>(temperature_outside) (max_temperature))
25         (>=(number_occupants ?o)4)
26         (<=(number_occupants)6))
27     :effect (and (assign (temperature_setting) 21)
28         (increase (energy_consumption) 6000))
29 (:action light_setting_500
30     :parameters (?o -occupancy ?ws -window_shading)
31     :precondition (and (>=(time)0)
32         (=(occupancy_type ?o)2)(not (open ?ws))
33     :effect (and (assign (lighting_level) 500)
34         (open ?ws) (increase (energy_consumption) 2000))
35 (:durative-action hvac_cooling_off
36     :parameters (?o -occupancy)
37     :duration (=? duration 1)
38     :condition (and
39         (at start (on)) (at start(cooling))
40         (at start (>=(temp_inside)(min_temp)))
41     :effect (and (at start (not(on)))
42         (at end (increase (time)1))
43         (at end (increase(temp_inside)
44                 (+(*(*(nb_occupants)0.5)-1)2))))
```

1) Defining domain models: As shown in Listing 1, the domain file specifies the type of objects that exist in the smart building. In particular, we define rooms, sensors that may be installed in rooms, and the *energy systems* for which we provide a schedule – in this case HVAC, window shading, and lighting systems (Lines 1–4). We then define *predicates*, which are propositions that represent the state of the environment and may either be true or false. For example, we can define predicates that reflect the state of the HVAC system (whether it's turned ON or OFF) or the state of rooms' occupancy (Lines 5–10). In addition, the numerical planning subset of PDDL allows introducing *state variables* whose values are rational numbers. These are instances of *functions* whose values can be modified by actions (Lines 12–19). In our domain definition, functions are used to model and control temperatures inside and outside rooms, the level of lighting in rooms, and plug loads throughout a smart building.

We next define the *actions* that can be taken by the AI planner to schedule energy systems (Lines 20–44). Actions in PDDL are characterized by *parameters*, *preconditions* that must be satisfied prior to executing the action, and *effects*. We first define actions for selecting the initial configuration of different systems at the beginning of the schedule (HVAC temperature setpoint, lighting levels, etc.). For instance, Lines 20–28 show an action to set up the temperature for HVAC systems depending on the outside temperature, inside temperature, and number of occupants inside a room. Similarly, the action defined in Lines 29–34 is used by the planner to set up the lighting and window shadings. Note how the effect of each action includes increasing the value of the `energy_consumption` function. As described later, this function is used as a cost function to be optimized by the planner when searching for an efficient schedule.

We further define *durative actions* to provide a schedule for controlling energy systems. These are a special type of actions characterized by the fact that they are executed over a specific duration, rather than instantly. In addition, preconditions and effects of such actions are annotated with special *durative tags* to specify when a proposition holds: (i) a proposition can hold at the start of the interval (time point when the action starts)—i.e. keyword `at start`; (ii) a proposition can hold at the end of the interval (time point when the action effects are asserted)—i.e. keyword `at end`; (iii) a proposition should hold over the entire interval (invariant over the action duration)—i.e. keyword `over all`. For example, Lines 35–44 in Listing 1 show a durative action for turning OFF the cooling system after the temperature inside a room has reached a specified threshold. Note how this condition should hold at the *start* of the action (Line 40). The effects of this action — increasing the time and the inside temperature — characterized with the `at end` keyword, will take place *after* the action has been executed (Lines 42 – 43).

2) Defining problem instances: In the PDDL problem file, we instantiate objects and define the initial state of the smart building.

Listing 2: PDDL Problem File

```
(:init
    (=(temperature_outside) 27)
    (=(temperature_inside) 27)
    (=(energy_consumption) 0)
    (=(min_temperature) 20)
    (=(max_temperature) 23)
    (=(number_occupants ?o) 2))
(:goal
    (and
    (all_done)
    (>=(temperature_inside) 20)
    (<=(temperature_inside) 24)))
(:metric minimize (energy_consumption))
```

As shown in Listing 2, the problem file includes initializing values for the outside and inside temperature, the number of occupants in the room, and thermal comfort bounds. Such information can be provided by the different sensors deployed in a smart building via querying a smart building's data management system. The `energy_consumption` function is initialized to 0, with each action increasing its value by specified increments, as described earlier. Our goal is to provide a schedule for the whole scheduling period, while keeping the temperature inside the rooms within bounds that take into account the thermal comfort of occupants (Lines 8–12). In addition, we further specify a *metric* to be minimized. In our case, we want the planner to provide a schedule that minimizes the total energy consumption. Hence, the AI planner will choose actions that result in the minimal value for the `energy_consumption` function, all the while satisfying the conditions specified in the goal state. Note that a planning horizon can be specified to provide schedules tailored to different use cases. For instance, in office buildings, schedules are typically needed only during operation hours (e.g., from 8 a.m. to 7 p.m.), whereas in airports, planning is required continuously and may be triggered periodically (e.g., every few hours) as needed.

A part of an example output using PDDL planners such as LPG [33] is provided below:

```
0.0010: (TEMPERATURE_SETTING_21 O1) [D:1.0000; C:0.1000]
0.0012: (TURN_COOLING_ON R1) [D:1.0000; C:5500.0000]
1.0013: (HVAC_IS_COOLING R1) [D:1.0000; C:4000.0000]
2.0015: (HVAC_COOLING_OFF R1) [D:1.0000; C:0.1000]
3.0017: (TURN_COOLING_ON R1) [D:1.0000; C:5500.0000]
```

The plan indicates the temperature setpoint for the HVAC system, and turning ON and OFF the HVAC according to intervals defined in the domain file.

C. Enabling Cross-Building Scheduling of Smart Energy Systems

A distinguishing feature of AI planning is the separation of domain models and problem instances. This enables domain knowledge to be captured and modeled in one domain file that includes all possible actions that can be taken by the planner, regardless of the specific problem instances. In contrast, in data driven approaches that leverage machine learning for optimizing energy consumption, it is necessary to train models for each building instance. Even in cases where a generic model can used for different buildings, there is still a need for fine-tuning to guarantee an optimal performance of the energy system. Hence, the separation provided by AI planning between domain knowledge and building representation proves to be particularly advantageous, since it allows to effectively decouple the general domain knowledge from the particular problem instances; we encapsulate in the domain file the general rules, constraints, and actions that can be taken by the planner for controlling energy systems in any smart building (e.g., setting temperatures, scheduling systems).

On the other hand, the problem file includes details about a particular smart building instance in a specific situation. We can thus define multiple problem files for different smart buildings, or for the same building under different conditions, without needing to change the existing domain model. For example, the same domain file shown in Listing 1 may be coupled with two different problem files that represent different types of buildings (e.g., office building vs. elderly care facility). The different characteristics of these buildings will be captured in the problem file (Listing 2) which can also include different metrics to optimize for: the thermal comfort of occupants may be more important in the elderly care facility than in the office building. In addition, the same domain model can be used with problem files that represent buildings in different climate zones.

Hence, the primary advantage of using AI planning is enabling portability of domain knowledge of all energy systems by encapsulating all possible configuration actions in one domain file, and instantiating different problem to handle different situations in buildings. This also allows for quick runtime adaptation, e.g., in case of an emergency situation, which would otherwise require re-training Machine Learning models or re-modeling optimization solutions. As part of our future work, we plan to define problem files that capture different needs of building administrators and country regulations. Using weighted metrics, plans can be generated to optimize for multiple objectives, such as energy consumption, occupants' thermal comfort, and carbon footprints.

V. Experimental Evaluation

This section provides the experimental evaluation of our approach. We first present the experimental setup used throughout the evaluation, and how we generate energy scheduling plans in V-A. We then evaluate the generated schedules against current practices in V-B, and showcase the adaptation capabilities of our approach in V-C. We finally show in V-D how our models can be adapted for defining actions depending on building administrators' specific needs.

A. Experimental Setup

We evaluate our approach by generating plans for scheduling the HVAC system in an office building. For this purpose, we use Co-zyBench [7], a benchmarking platform for evaluating the performance of HVAC control systems in terms of occupants' thermal comfort and energy efficiency. As shown in Figure 2, Co-zyBench consists of the following Digital Twins (DTs): (i) the *Building DT* for modeling and simulating a building and its HVAC system; and (ii) the *Occupant DT* for modeling and simulating occupants' trajectories in the building. A co-simulation middleware integrates these DTs with our AI planner to provide HVAC schedules.

1) Building DT: To create the Building DT, Co-zyBench integrates EnergyPlus[2] for simulating the HVAC system. We use the DT to represent an office space from *The Office*, a popular US sitcom. The space comprises three office rooms, a conference room, a restroom, a breakroom and a kitchen, with a gross area of $268m^2$ (see Figure 2). The office rooms are shared by people who are assigned to work in specific rooms,

[2] https://energyplus.net/

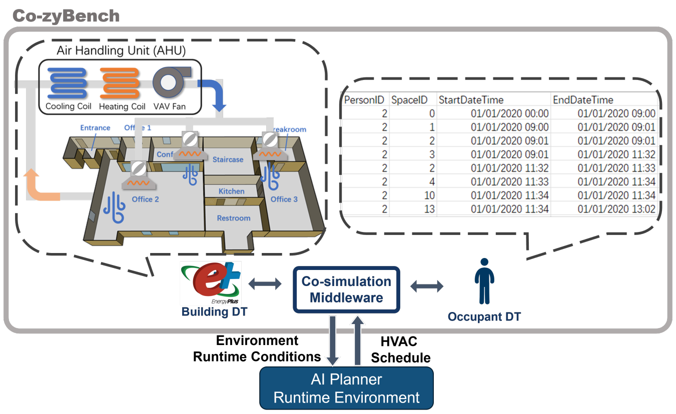

Fig. 2: Runtime scheduling via Co-zyBench co-simulations.

TIME	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	Boss, Sellers				
9:30			Boss, Accountants, Supplier Manager		
10:00		Sellers		Sellers	
10:30	Sellers, Marketing, Sales Representative				Sellers, Marketing, Sales Representative,
11:00					
11:30					
12:00 - 14:00	Lunch Time				
14:00					All
14:30	Accounts		Sellers, Marketing, Sales Representative,		
15:00					
15:30					
16:00					

TABLE I: Weekly schedule for the office occupants.

while common rooms are shared among all occupants. For the HVAC system, we implement a variable air volume (VAV) system model, a widely used system in commercial buildings, which can effectively provide thermal comfort while consuming less energy than other systems. VAV mainly consists of an outdoor air (OA) system, an air handling unit (AHU), sensors, and VAV boxes for each room. The OA system blends return air and outdoor air, which is then conditioned by the AHU before being distributed to the rooms. The temperature of the supply air is monitored to adjust the coils and VAV fan of the AHU. Finally, the supply air blows to each room through VAV boxes that are equipped with dampers for air column control and a reheat coil to precisely control the supply air temperature.

2) Occupant DT: To accurately simulate natural and dynamic human movement within the space, we create the Occupant DT component using SmartSPEC [34], a tool for generating realistic people trajectories in a smart space. This is achieved by employing a semantic model that encompasses the space, its occupants, and various events. The core concept involves simulating occupants' movements as they navigate to different events occurring within the building. We create profiles for 18 individuals representing various job roles such as director, seller, accountant, etc. Each person is assigned a unique personality, influencing their behavior in attending events, and thus their trajectories. This is done by varying the probabilities for each individual of being early, late, or absent for each event. We categorize possible events into two types: *static* events with time-fixed schedules such as meetings, and *dynamic* events that can occur at spontaneously such as going to the break room or to the restroom. Figure 2 contains the meeting schedules that we created for guiding people's trajectories in the office building. The work day starts at around 9:00 and ends after 18:00, people spend most of their working time attending a "working event" that occurs in their assigned offices. Their weekly schedule involves two types of meetings happening in the conference room: small meetings, held by some members of one or two teams mainly in the morning lasting 30 to 60 minutes; and large meetings, held by some members of two or more teams lasting around 90 minutes. For example, sellers and marketers have a small meeting every two days for at most one hour in the morning, all the staff have a large meeting at 14:00 for at most two hours every Friday. Besides, we model a lunchtime event from 12:00 to 14:00 and limit the amount of time spent for lunch to around one hour in the kitchen, the break room, or the outside.

3) Climate zone setup: The location for the office space is set in Paris, France, categorized under Climate Zone 4A-Mixed Humid by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). To evaluate our approach, our experiments focus on the months with the highest cooling demand (June, July, August). For the whole year's climate data, we use the latest available Typical Meteorological Year (TMY) weather reports derived from the recent 15 years (2007-2021) from [35] (with a minimum and maximum temperature of -5.0-34.5$°C$). Accordingly, the office building's construction is modeled to meet the specifications for the climate of Paris, adhering to ASHRAE standards such as 6.8 cm of insulation in the walls and 21 cm of insulation in the roof. Additionally, we include the heat generated by indoor equipment and people located in a particular space of the building in the computation. Note that we only evaluate the approach for cooling over a period of 3 summer months because of the growing need for cooling, especially with temperatures rising due to climate change.

4) Scheduling scenarios: To generate HVAC control schedules, we use PDDL to create a domain file that contains possible actions that could be applied to the HVAC system (e.g., turning it ON/OFF, changing the temperature). We then receive inputs from the building and occupant simulator about the current indoor and outdoor temperatures, as well as the number of occupants, and use them to model the building properties in the problem file. The AI Planner then crafts an HVAC control schedule with the primary goal of minimizing energy consumption, while maintaining indoor temperatures within the comfortable range of 21–24$°C$. The HVAC operating schedules include operation from 8:00 to 20:00 daily, and actions may be performed every 30 minutes. To accommodate potential inaccuracies in predicting future conditions, we update the schedules at 8:00, 14:00, and 16:00 every day. Moreover, we continually monitor deviations in the simulation environment. If there's a change exceeding 3$°C$

in temperature or a fluctuation of more than 3 people in occupancy, we trigger re-planning to generate a new schedule.

B. Evaluating the Generated Schedules

We first assess our approach's effectiveness in reducing energy consumption compared to control algorithms that maintain a fixed temperature of $22°C$ throughout the day (Fixed-22). We also monitor the temperature deviation from the optimal comfort range of 21-24$°C$ to ensure that our approach does not cause thermal discomfort for occupants. Figure 3 shows the energy consumption per month when using schedules generated by the AI planner to regulate the HVAC system, compared to a fixed setpoint of $22°C$. Our approach manages to achieve a reduction of energy consumption of up to 35% per month. In July, the hottest month in Paris, AI planning manages to reduce energy consumption from 531 kWh to 337 kWh, while keeping an average temperature deviation under $1°C$ per day. To have a better understanding of how this is achieved, we plot in Figure 4 the variation in the inside and outside temperatures for one day. We can see that as opposed to having a constant temperature throughout the day, schedules generated with AI planning lead to fluctuations in inside temperature. These are the results of the actions taken every 30 minutes by the AI planner (turning on/off the HVAC, changing temperature setpoints, etc.). However, the planner ensures that the temperature does not go outside the thermal comfort bounds defined in the problem file, as mentioned in Section IV. The frequency at which actions are executed is determined by observing the time interval it generally takes for the indoor temperature to naturally rise beyond the comfort range (for cooling settings). This time interval varies from building to building due to factors such as local climate, fenestration, and building materials' thermal mass. This information can be obtained through historical temperature data of a building. In our scenario, historical data demonstrated that this time interval is over 30 minutes. The duration of each action can be specified in the AI domain file (Listing 1).

Figure 5 shows the carbon emissions of both approaches throughout one day, with each data point representing the emissions during a simulation time step of two minutes. We also plot the carbon intensity during that day. This is a measure of how much CO2 emissions are produced per kilowatt hour of electricity consumed, for one single day. For this purpose, we rely on electricity data[3] to set the carbon intensity values for one day in France. Note that carbon emissions = energy consumption × carbon intensity. Therefore, we can see that the AI Planner is capable of reducing carbon emissions by 33.7%. This decrease demonstrates the system's efficiency in energy use but also highlights its contribution to environmental sustainability by mitigating the carbon footprint. Carbon intensity can be incorporated in the domain model and problem representations as an additional metric to optimize for. As different countries rely on different methods for electricity generation, the planner can take such metrics into account while trying to generate a schedule for more efficient energy systems.

[3]https://www.electricitymaps.com/

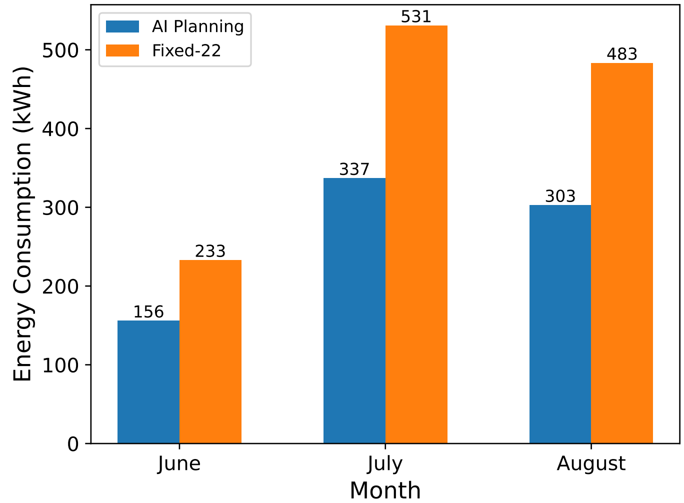

Fig. 3: Total energy consumption (per month) for cooling.

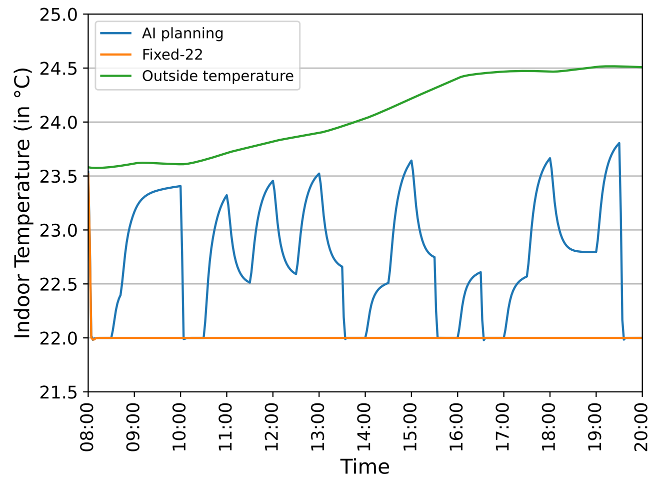

Fig. 4: Inside temperature variations in a single office.

C. Enabling Adaptation in Dynamic Situations

In the wake of the global pandemic, the trend of working from home has gained significant momentum. This shift towards remote work has led to lower occupancy in spaces like office rooms and substantial energy waste. Simply turning OFF the HVAC in these areas is not a viable solution as it negatively impacts thermal comfort for individuals entering these rooms. For example, turning off the HVAC in seldom-used break rooms can lead to discomfort for people wishing to stay in the room, and they will not stay long enough, waiting until the temperature is adjusted to a comfortable level, resulting in failure to achieve desired thermal comfort.

To address this issue, our approach enables generating schedules on a day-to-day basis, depending on the number

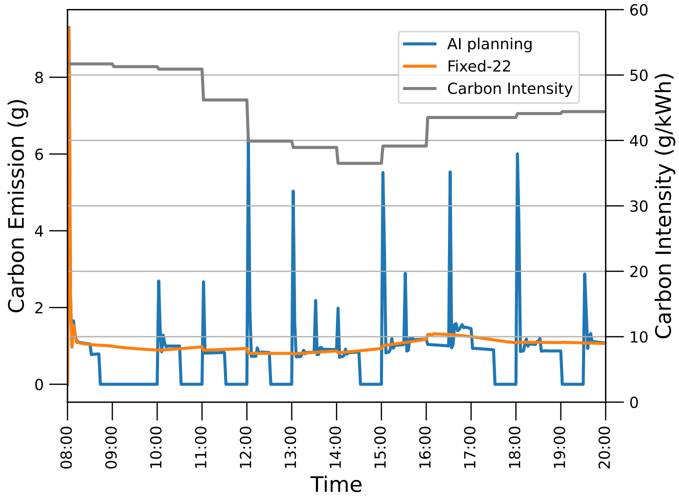

Fig. 5: Carbon emissions for a single day.

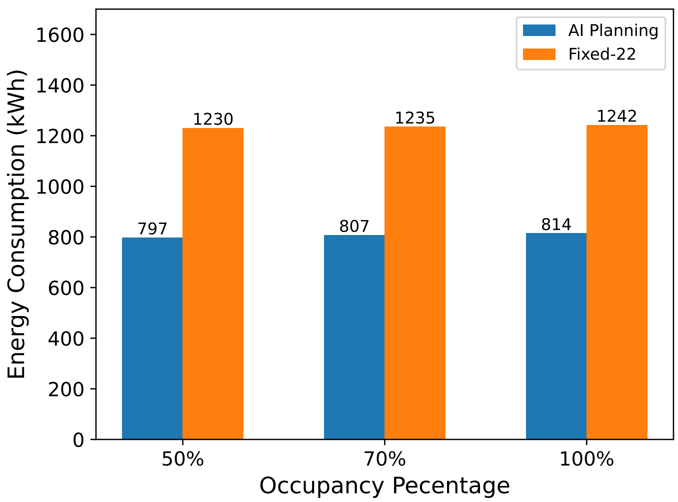

Fig. 6: Energy consumption with different occupancy levels.

of occupants in each space[4]. We conduct a set of experiments to evaluate the ability of our AI Planner to generate plans that adapt to the number of occupants present in rooms. We use the same experimental setup described in Section V-A, but we consider different levels of presence of employees, ranging from 50%, to 100%. The occupants present in the building are randomly selected at the beginning of each day.

Figure 6 shows the energy consumption in kWh across the scenarios. Once again, we can see that AI planning manages to adapt to the changing circumstances and achieve energy savings of more than 30% compared to setting a fixed temperature of $22°C$. However, energy savings are not significant when fewer people are present in the building, compared to having full occupancy levels. This happens because of the enforced constraints related to the temperature comfort bounds.

[4]Note that in order to keep the temperature in the room within the range of thermal comfort, one strategy includes not turning off the HVAC system completely, even when there are no occupants in the room.

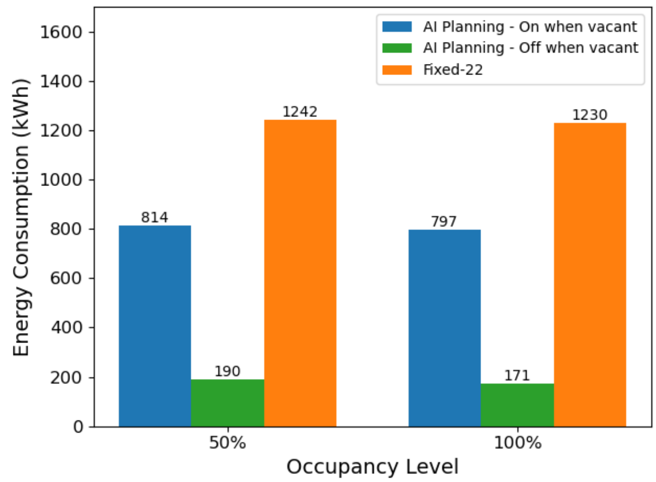

Fig. 7: Energy consumption with HVAC ON & HVAC OFF for vacant rooms.

D. *Improving Efficiency in Energy-Constrained Settings*

We now consider the case where the building administrator wishes to turn off the HVAC system when rooms are vacant. This could happen for various reasons, e.g., achieving more reduction of energy consumption, or reducing the load on the electric system of the building during peak demand periods. For this purpose, we only need to add one action which consists of turning OFF the HVAC when there are no occupants in rooms. The planner will then be able to include this action in the generated schedule when the needed conditions are met.

Figure 7 shows the energy consumption across the scenarios of turning the HVAC ON or OFF in empty rooms, as well as Fixed-22 which doesn't consider occupant numbers. As expected, turning off the HVAC in vacant rooms results in a 77% decrease in energy consumption compared to not turning off the HVAC in vacant rooms. To see this strategy's impact on the inside temperature, we plot in Figure 8 the variations of the inside temperature throughout one day. The figure reveals that the temperature exceeds $24°C$, especially in the afternoon. For example, at 17:30 when people enter a room, the indoor temperature reaches $24.5°C$ after an unoccupied period of 2 hours and the occupants have to experience discomfort and wait till the temperature decreases. This happens as a result of turning OFF the HVAC system when rooms are vacant; as a consequence, it takes more time for the HVAC system to regulate the temperature when employees occupy the room again. Hence, in such cases, there is a trade-off between energy consumption and occupants' thermal comfort.

VI. Conclusion

This paper presents an approach for efficiently scheduling energy systems in smart buildings using AI planning. This is achieved by leveraging PDDL to represent domain models as building types (e.g., office building), incorporating actions for controlling energy systems, and problem instances for describing specific situations and properties of smart buildings.

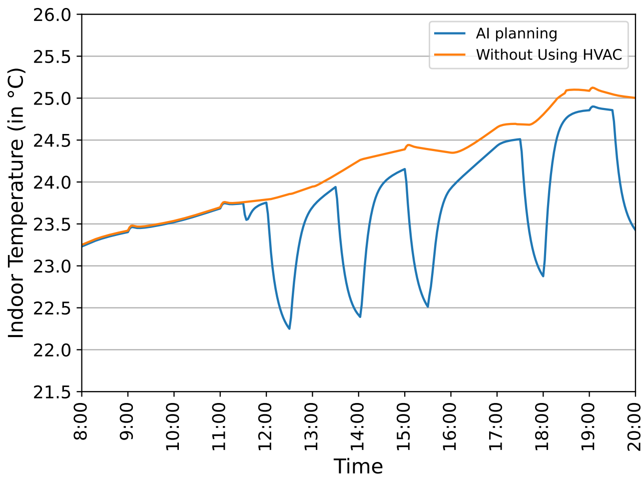

Fig. 8: Inside temperature variations in a single office – HVAC OFF in vacant rooms.

An AI planner then generates a schedule aiming to optimize metrics such as energy consumption and carbon footprints. We evaluate our approach in a realistic smart office setting and manage to reduce energy consumption by 30% compared to current practices, which fix a temperature setpoint for HVAC systems throughout the day.

In our future work, we aim to expand our approach to include optimization for various needs of building administrators and compliance with country regulations. These include carbon footprints and occupants' thermal preferences. In addition, we plan to validate our approach using diverse scenarios at different climate zones including beyond HVAC systems, other smart energy systems such as lighting control, plug loads and more. We also aim to apply our solution in real-life settings for hydropower plants in the context of the Di-Hydro EU project[5]. Finally, we shall investigate more powerful model representations, such as PDDL+, for enabling precise modeling and control of energy systems. These modifications would further enhance the effectiveness of the proposed approach and contribute to a more energy-efficient future.

Acknowledgements

This work is partially supported by the Horizon Europe project DI-Hydro under grant agreement number 101122311, the Energy4Climate Interdisciplinary Center (E4C), which is in part supported by 3rd Programme d'Investissements d'Avenir [ANR-18-EUR-0006-02], and the China Scholarship Council (CSC).

References

[1] N. Luo, Z. Wang, D. Blum, C. Weyandt, N. Bourassa, M. A. Piette, and T. Hong, "A three-year dataset supporting research on building energy management and occupancy analytics," *Scientific Data*, vol. 9, no. 1, p. 156, 2022.

[2] M. González-Torres, L. Pérez-Lombard, J. F. Coronel, I. R. Maestre, and D. Yan, "A review on buildings energy information: Trends, end-uses, fuels and drivers," *Energy Reports*, vol. 8, pp. 626–637, 2022.

[3] Y. Khan, V. R. Khare, J. Mathur, and M. Bhandari, "Performance evaluation of radiant cooling system integrated with air system under different operational strategies," *Energy and Buildings*, vol. 97, pp. 118–128, 2015.

[4] L. Yu, S. Qin, M. Zhang, C. Shen, T. Jiang, and X. Guan, "A review of deep reinforcement learning for smart building energy management," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 12 046–12 063, 2021.

[5] M. Esrafilian-Najafabadi and F. Haghighat, "Occupancy-based hvac control systems in buildings: A state-of-the-art review," *Building and Environment*, vol. 197, p. 107810, 2021.

[6] W. Jung and F. Jazizadeh, "Human-in-the-loop hvac operations: A quantitative review on occupancy, comfort, and energy-efficiency dimensions," *Applied Energy*, vol. 239, pp. 1471–1508, 2019.

[7] M. Jun, P. Dimitrije, Y. Roberto, and B. Georgios, "Co-zybench: Using co-simulation and digital twins to benchmark thermal comfort provision in smart buildings," in *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2024.

[8] M. Gholamzadehmir, C. Del Pero, S. Buffa, R. Fedrizzi *et al.*, "Adaptive-predictive control strategy for hvac systems in smart buildings–a review," *Sustainable Cities and Society*, vol. 63, p. 102480, 2020.

[9] S. Ahmadi-Karvigh, B. Becerik-Gerber, and L. Soibelman, "Intelligent adaptive automation: A framework for an activity-driven and user-centered building automation," *Energy and Buildings*, vol. 188, pp. 184–199, 2019.

[10] A. Yayla, K. S. Świerczewska, M. Kaya, B. Karaca, Y. Arayici, Y. E. Ayözen, and O. B. Tokdemir, "Artificial intelligence (ai)-based occupant-centric heating ventilation and air conditioning (hvac) control system for multi-zone commercial buildings," *Sustainability*, vol. 14, no. 23, p. 16107, 2022.

[11] A. Javed, H. Larijani, A. Ahmadinia, R. Emmanuel, M. Mannion, and D. Gibson, "Design and implementation of a cloud enabled random neural network-based decentralized smart controller with intelligent sensor nodes for hvac," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 393–403, 2016.

[12] M. Aftab, C. Chen, C.-K. Chau, and T. Rahwan, "Automatic hvac control with real-time occupancy recognition and simulation-guided model predictive control in low-cost embedded system," *Energy and Buildings*, vol. 154, pp. 141–156, 2017.

[13] M. Ghallab, D. Nau, and P. Traverso, *Automated planning and acting*. Cambridge University Press, 2016.

[14] A. Rugeviciute, V. Courboulay, and L. M. Hilty, "The research landscape of ict for sustainability: harnessing digital technology for sustainable development," in *2023 International Conference on ICT for Sustainability (ICT4S)*. IEEE, 2023, pp. 97–107.

[15] Y. Yao and D. K. Shekhar, "State of the art review on model predictive control (mpc) in heating ventilation and air-conditioning (hvac) field," *Building and Environment*, vol. 200, p. 107952, 2021.

[16] S. Sierla, H. Ihasalo, and V. Vyatkin, "A review of reinforcement learning applications to control of heating, ventilation and air conditioning systems," *Energies*, vol. 15, no. 10, p. 3526, 2022.

[17] C. Vering, P. Mehrfeld, M. Nürenberg, D. Coakley, M. Lauster, and D. Müller, "Unlocking potentials of building energy systems' operational efficiency: application of digital twin design for hvac systems," *16th International Building Performance Simulation Association (IBPSA)*, pp. 1304–1310, 2019.

[18] X. Xie, J. Merino, N. Moretti, P. Pauwels, J. Y. Chang, and A. Parlikad, "Digital twin enabled fault detection and diagnosis process for building hvac systems," *Automation in Construction*, vol. 146, p. 104695, 2023.

[19] M. Gutiérrez, M. Á. Moraga, and F. García, "Analysing the energy impact of different optimisations for machine learning models," in *2022 international conference on ICT for sustainability (ICT4S)*. IEEE, 2022, pp. 46–52.

[20] https://www.electricitymaps.com/.

[21] J. Bernardes Jr, M. Santos, T. Abreu, L. Prado Jr, D. Miranda, R. Julio, P. Viana, M. Fonseca, E. Bortoni, and G. S. Bastos, "Hydropower operation optimization using machine learning: A systematic review," *AI*, vol. 3, no. 1, pp. 78–99, 2022.

[22] F. Ingrand and M. Ghallab, "Deliberation for autonomous robots: A survey," *Artificial Intelligence*, vol. 247, pp. 10–44, 2017.

[23] B. Wally, J. Vyskočil, P. Novák, C. Huemer, R. Šindelář, P. Kadera, A. Mazak, and M. Wimmer, "Production planning with iec 62264

[5] https://dihydro-project.eu/

and pddl," in *2019 IEEE 17th international conference on industrial informatics (INDIN)*, vol. 1. IEEE, 2019, pp. 492–499.
[24] A. Marrella, "Automated planning for business process management," *Journal on data semantics*, vol. 8, no. 2, pp. 79–98, 2019.
[25] H. H. Hassan, G. Bouloukakis, A. Kattepur, D. Conan, and D. Belaïd, "Planiot: A framework for adaptive data flow management in iot-enhanced spaces," in *18th Symposium on Software Engineering for Adaptive and Self-Managing Systems*, 2023.
[26] I. Georgievski, T. A. Nguyen, and M. Aiello, "Combining activity recognition and ai planning for energy-saving offices," in *2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing*, 2013, pp. 238–245.
[27] K. Awahara, S. Izumi, T. Abe, and T. Suganuma, "Autonomous control method using ai planning for energy-efficient network systems," in *2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications*, 2013, pp. 628–633.
[28] M.-J. Li and W.-Q. Tao, "Review of methodologies and polices for evaluation of energy efficiency in high energy-consuming industry," *Applied Energy*, vol. 187, pp. 203–215, 2017.
[29] P. Haslum, N. Lipovetzky, D. Magazzeni, C. Muise, R. Brachman, F. Rossi, and P. Stone, *An introduction to the planning domain definition language*. Springer, 2019, vol. 13.
[30] A. Gerevini and I. Serina, "Lpg: A planner based on local search for planning graphs with action costs." in *Aips*, vol. 2, 2002, pp. 281–290.
[31] J. Hoffmann, "Extending ff to numerical state variables," in *ECAI*, vol. 2. Citeseer, 2002, pp. 571–575.
[32] ——, "Ff: The fast-forward planning system," *AI magazine*, vol. 22, no. 3, pp. 57–57, 2001.
[33] A. Gerevini and I. Serina, "Lpg: A planner based on local search for planning graphs with action costs." in *Aips*, vol. 2, 2002, pp. 281–290.
[34] A. Chio, D. Jiang, P. Gupta, G. Bouloukakis, R. Yus, S. Mehrotra, and N. Venkatasubramanian, "Smartspec: Customizable smart space datasets via event-driven simulations," in *2022 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2022, pp. 152–162.
[35] D. B. C. Lawrie, Linda K, "Development of Global Typical Meteorological Years (TMYx)," 2019. [Online]. Available: http://climate.onebuilding.org