MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

AI-Driven Scenario Discovery: Diffusion Models and Multi-Armed Bandits for Building Control Validation

Tang, Wei-Ting; Vinod, Abraham P.; Germain, François G; Paulson, Joel A.; Laughman, Christopher R.; Chakrabarty, Ankush

TR2025-132 September 10, 2025

Abstract

A critical component of model predictive control (MPC) in building energy management systems is the ability to reject exogenous disturbances such as occupant-induced heat loads, appliance loads, ambient temperature, and solar radiation. During the design phase of model predictive control, engineers typically select a limited set of disturbance scenarios through handcrafting or sampling from simple distributions. However, this approach often fails to capture a representative set of scenarios that elicit diverse closed-loop behaviors across the full operational envelope. This work addresses this limitation by proposing a combinatorial multi-armed bandit (CMAB) framework for systematically discovering representative disturbance scenarios using real building data. We formulate the scenario selection problem as a diversity maximization task, where the reward function quantifies the behavioral diversity of closed-loop responses through information-theoretic criteria such as dynamic time warping distance. The proposed approach treats the building simulation environment as a black-box system, making it applicable to complex, proprietary, or non-differentiable building models commonly encountered in practice. To address data scarcity challenges, we extend the framework by incorporating time-series generative models, specifically diffusion-based networks, to synthetically augment limited real datasets. Experimental validation using a commercial net-zero energy building demonstrates that synthetic data augmentation significantly enriches the diversity of discovered scenarios compared to using real data alone, as evidenced through principal component analysis and uniform manifold approximation projections. The CMAB algorithm successfully identified representative scenarios that revealed controller vulnerabilities not detected by conventional selection methods, leading to practical improvements in HVAC system design. The approach scales linearly with the number of scenarios and bandit iterations, making it computationally feasible for grid-interactive building energy management applications.

Energy and Buildings 2025

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

AI-Driven Scenario Discovery: Diffusion Models and Multi-Armed Bandits for Building Control Validation

Wei-Ting Tang[b], Abraham P. Vinod[a], François Germain[a], Joel A. Paulson[b], Christopher R. Laughman[a], Ankush Chakrabarty[a,*]

[a] *Mitsubishi Electric Research Laboratories, Cambridge, MA, USA.*
[b] *Department of Chemical and Biomolecular Engineering, The Ohio State University, Columbus, OH, USA*

Abstract

A critical component of model predictive control (MPC) in building energy management systems is the ability to reject exogenous disturbances such as occupant-induced heat loads, appliance loads, ambient temperature, and solar radiation. During the design phase of model predictive control, engineers typically select a limited set of disturbance scenarios through handcrafting or sampling from simple distributions. However, this approach often fails to capture a representative set of scenarios that elicit diverse closed-loop behaviors across the full operational envelope. This work addresses this limitation by proposing a combinatorial multi-armed bandit (CMAB) framework for systematically discovering representative disturbance scenarios using real building data. We formulate the scenario selection problem as a diversity maximization task, where the reward function quantifies the behavioral diversity of closed-loop responses through information-theoretic criteria such as dynamic time warping distance. The proposed approach treats the building simulation environment as a black-box system, making it applicable to complex, proprietary, or non-differentiable building models commonly encountered in practice. To address data scarcity challenges, we extend the framework by incorporating time-series generative models, specifically diffusion-based networks, to synthetically augment limited real datasets. Experimental validation using a commercial net-zero energy building demonstrates that synthetic data augmentation significantly enriches the diversity of discovered scenarios compared to using real data alone, as evidenced through principal component analysis and uniform manifold approximation projections. The CMAB algorithm successfully identified representative scenarios that revealed controller vulnerabilities not detected by conventional selection methods, leading to practical improvements in HVAC system design. The approach scales linearly with the number of scenarios and bandit iterations, making it computationally feasible for grid-interactive building energy management applications.

Keywords: Generative AI, Diffusion models, Multi-arm bandits, Model predictive control, Synthetic time-series, Grid-interactive efficient buildings, Data augmentation

Nomenclature

*Corresponding author
Email addresses: tang.1856@osu.edu (Wei-Ting Tang), second.author@example.com (Abraham P. Vinod), germain@merl.com (François Germain), paulson.82@osu.edu (Joel A. Paulson), laughman@merl.com (Christopher R. Laughman), achakrabarty@ieee.org (Ankush Chakrabarty)

$\mathbb{W}$	Disturbance space	GEB	Grid-interactive efficient building
$\mathcal{I}$	Scoring function	GPU	Graphics processing unit
$\mathcal{M}$	Black-box model	HP	Heat pump
$\mathcal{S}$	Super arm set	MAB	Multi-arm bandit
r	Reward	MAE	Mean absolute error
S	Super arm	MASE	Mean absolute scaled error
T	Length of time for time-series data	MBRL	Model-based reinforcement learning
t	Time index	MPC	Model predictive control
T_i	Number of times arm i been played	PCA	Principal component analysis
W	Disturbance	PV	Photovoltaic
BES	Building energy system	TSGN	Time-series generative network
BESS	Battery energy storage system	UCB	Upper confidence bound
CPU	Central processing unit	UMAP	Uniform manifold approximation & projection
CUCB	Combinatorial upper confidence bound		
DDPM	Denoising diffusion probabilistic model		
FFT	Fast Fourier transform		

1. Introduction

Model predictive control (MPC), which solves model-based optimization problems in real time to account for constrained multivariable dynamic systems, has proven a good solution to automated building energy management, for general building energy systems (BES); this is evident from the experiments reported in [1, 2, 3]. MPC is designed to achieve near-optimality and safety, supported by well-established stability theory, under the assumptions of a high-quality predictive model and a reasonably narrow range of possible disturbances [4]. However, BESs are subject to complex exogenous disturbances, such as solar radiation effects, occupant-driven thermal loads, plug loads, and ambient conditions [5, 6], many of which exhibit significant variability due to human behavior. Such wide disturbance variations can degrade the closed-loop control performance of standard MPC. Although feedback mechanisms offer inherent robustness, relying solely on feedback may be insufficient when rejecting time-varying large-magnitude disturbance patterns. To combat disturbances, stochastic MPC (SMPC) and robust MPC (RMPC) have been extensively studied for BES applications [5, 7, 8, 9, 10, 11]: these methods are especially powerful owing to their ability to explicitly account for, and protect against, uncertainty.

Incorporating disturbance rejection into MPC via RMPC or SMPC typically requires assumptions about the disturbance set, manifesting in clear deterministic bounds or probability distributions over possible disturbance realizations. However, closed-loop performance is highly sensitive to the quality of these assumptions. RMPC often leads to conservative designs because disturbance ranges must cover variations linked to seasonality, occupancy behavior, and lifestyle [12]. To capture all such variability, excessively conservative uncertainty sets may be needed. SMPC can mitigate this conservatism by leveraging probabilistic models, but standard assumptions such as Gaussianity are often imposed for computational tractability [13], which are not reflective of the true disturbance distribution.

With measured data collected from the target GEB, designers could, in principle, make more informed assumptions by analyzing the empirical disturbance distribution. However, using realistic distributions often leads to complex SMPC optimization problems, motivating the use of a finite set of disturbance scenarios (i.e., fixed-length time-series samples) for controller design and validation. In practice, scenario sets are frequently handcrafted, often limited to a small number of typical or extreme cases, such as mean profiles or worst-case disturbances. An open and important question is how to *automatically* construct a representative set of disturbance scenarios that induce "novel" or "unusual" closed-loop dynamics, capturing diverse performance behaviors beyond those observed previously. Extracting such a set from historical disturbance data could significantly improve controller evaluation and robustness validation. Prior work [14] applied Bayesian active learning techniques, specifically InfoBAX [15], combined with synthetic disturbance generation via RAFT-VG [16], to discover disturbance scenarios corresponding to the best- and worst-case closed-loop performance. While this approach demonstrated efficient identification of extreme outcomes, it primarily focused on finding top- and bottom-k performance scenarios rather than capturing the full diversity of possible closed-loop behaviors. In contrast, the present work aims to identify a broader range of diverse scenarios systematically and efficiently.

In this work, we develop a general framework to *efficiently* identify representative disturbance scenarios that elicit diverse closed-loop behaviors, formulating the problem within a multi-armed bandit (MAB) framework [17, 18]. Efficiency is critical because closed-loop building simulations are often computationally expensive. In MAB problems, an agent sequentially selects actions (arms) to optimize cumulative rewards under uncertainty. We extend this setup to combinatorial MABs (CMABs) [19], allowing selection of multiple disturbance scenarios (or arms) per round. Given a legacy MPC law for a target GEB and access to a GEB simulation model, we introduce a model-agnostic CMAB algorithm that sequentially selects a set of finite disturbance scenarios to maximize an information-theoretic criterion measuring closed-loop behavioral diversity (such as pairwise dynamic time warping (DTW) distance). Disturbance scenarios are constructed as fixed-length time-series (e.g., 24-hour profiles) encompassing ambient temperature, solar irradiance, and internal heat gains. Two major challenges commonly arise in practice: (i) the combinatorial nature of the disturbance scenario space and (ii) the opacity of the building simulation environment, where the disturbance-to-performance mapping is often a black-box

due to proprietary or non-differentiable components, such as commercial MPC solvers [20].

Furthermore, we extend our framework to address small-data settings, where the historical disturbance data for the target GEB may be insufficient. In such cases, we leverage time-series generative models (TSGMs) to synthetically augment the disturbance dataset. Conventional models such as nonlinear autoregressive models with exogenous inputs (NARX) [21] may be unsuitable for complex, long-range correlated building profiles. More advanced approaches, including generative adversarial networks (GANs) [22], variational autoencoders (VAEs) [23], and their hybridizations [16, 24, 25, 26, 27, 28], have been proposed to better capture the intricacies of building energy profiles while mitigating known limitations such as mode collapse in GANs [29] or underfitting in VAEs [30]. As an illustrative example, this paper applies a recent time-series diffusion network, Diffusion-TS [31], to generate realistic disturbance scenarios. Diffusion models [32, 33] have surpassed GANs in image and text generation tasks by offering more stable training and avoiding adversarial collapse [29]. Their application to time-series generation has gained momentum [34, 35, 36], particularly for tasks such as imputation, forecasting, and unconditional generation. Diffusion-TS, a non-autoregressive denoising diffusion probabilistic model (DDPM), has demonstrated strong performance across multiple synthetic data tasks, motivating its use here to enhance exploration in small-data regimes.

The major *contributions* of this work are as follows:

1. We introduce a novel problem formulation for identifying representative disturbance scenarios via a CMAB approach, which, to our knowledge, has not been previously explored in the context of BES and GEB.
2. We develop and apply an efficient CMAB algorithm to solve the proposed scenario selection problem.
3. We propose a practical small-data extension by incorporating synthetic disturbances generated through state-of-the-art time-series generative models (TSGMs).
4. We demonstrate through extensive simulation experiments that synthetic data augmentation significantly improves the diversity and representativeness of discovered disturbance scenarios, supporting more robust closed-loop controller evaluation.
5. We provide empirical evidence that the representative scenarios obtained can shed light on the performance of the legacy closed-loop system that had been designed only using nominal scenarios; we demonstrate that our CMAB-based representative scenarios can inform redesign, adding practical value in building energy management systems.

A schematic overview of the proposed framework is provided in Figure 1. The main steps, as enumerated in the contributions, include: using, or augmenting with synthetic data, a dataset of real disturbance scenarios to assess MPC (or its variants) performance on a closed-loop simulation model of a GEB. With the help of CMABs, we will efficiently determine a subset of disturbances that result in the most diverse (according to some time-series distance metric) output trajectories. The subset obtained with the highest reward will be deemed the set of representative scenarios.

The remainder of this paper is organized as follows. Section 2 summarizes the relevant background and motivation for the work. Section 3 describes the proposed methodology and supporting theory. Section 4 presents our results on a realistic GEB system and we provide concluding remarks in Section 5.

2. Motivation

2.1. Problem Statement

Dynamic simulators of buildings, including GEBs, can be represented by the mathematical model

$$y_{0:t} = \mathcal{M}(x_0; u_{0:t}, w_{0:t}) + v_{0:t}, \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ denotes an internal state representation of the building dynamics and $u \in \mathbb{R}^{n_u}$ denotes control inputs or setpoints to the system determined by some control policy; e.g. switching patterns of power electronic components, thermostat settings, or HVAC inputs such as compressor speed. The variable $w \in \mathbb{R}^{n_w}$ denotes exogenous disturbance variables affecting the grid-interactive building such as ambient conditions, solar radiation, cloud cover, or occupant-induced heat loads/plug loads. The outputs $y \in \mathbb{R}^{n_y}$ of the mathematical model $\mathcal{M}$ are assumed to be available for measurement, or reliably estimated (but often noisy, modeled by the additive white Gaussian noise $v \sim \mathcal{N}(0, \Sigma)$ for some covariance matrix $\Sigma = \Sigma^\top \succ 0$).

We assume that these outputs contain elements that are indicative of the closed-loop performance. The notation $(\cdot)_{0:T}$ indicates a sequence of vectors, with sequence length $T+1$ where each element of the sequence represents the vector at time t on the range $\{0, 1, \ldots, T\}$; that is $y_{0:T} \in \mathbb{R}^{n_y \times (T+1)}$. The modeling equation (1) implies that, given an initial state of the system x_0, a control trajectory $u_{0:T}$, and exogenous disturbance trajectory $w_{0:T}$, simulating the dynamics yields a trajectory of corresponding outputs $y_{0:T}$.

Executing a simulation depends solely on the structure of the model $\mathcal{M}$. In modern simulation environments such as EnergyPlus or Modelica, it is not uncommon for $\mathcal{M}$ to contain components that are non-trivial to analyze using analytical methods. For instance, software blocks may contain proprietary information, or exhibit high modeling complexity (e.g., finite-element models for airflow or refrigerant flow dynamics). These complex blocks render analysis of $\mathcal{M}$ difficult, often impractical, to handle in a computationally efficient manner for design or analysis. In such cases, it is reasonable to make as few assumptions on the structure of $\mathcal{M}$ as possible and treat the model as a 'black-box' function that can be simulated to acquire quantitative metrics of performance. This obviates the

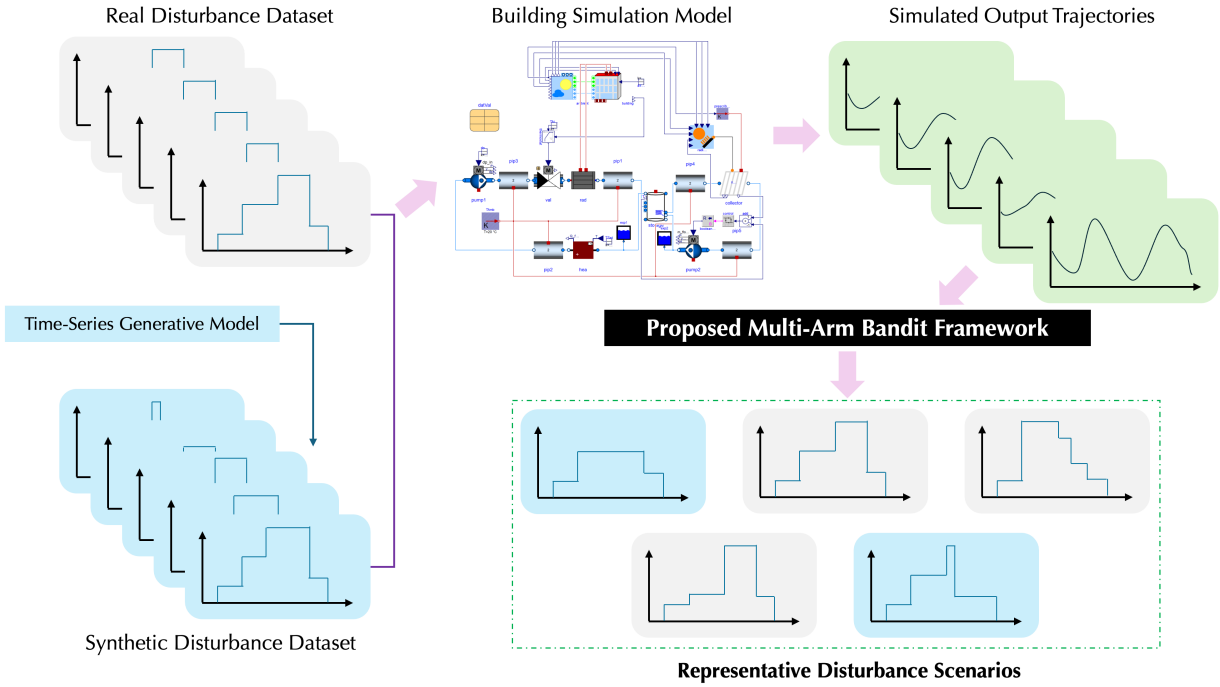

Figure 1: Schematic overview of the proposed framework: identifying representative disturbance scenarios using combinatorial multi-armed bandits (CMABs) and supporting small-data settings with synthetic augmentation via diffusion-based time-series generative models (TSGMs).

need to assume that we can directly access the structure of the model, for instance to take gradients, or directly handle the underlying system of equations.

Before we formalize the problem statement, we introduce the following definition.

Definition 1 (Disturbance Scenario)**.** *Let*

$$\mathcal{W}_T \subset \mathbb{R}^{n_w \times (T+1)}$$

be the set of all admissible disturbance trajectories of length $T+1$*, and let* π_W *be a probability distribution on* $\mathcal{W}_T$*. A disturbance scenario of length* T *is a sample* $w_{0:T} \sim \pi_W$*.*

Definition 1 explicitly places a joint distribution π_w over entire disturbance trajectories, capturing any temporal correlations or scenario-level structure (for example, diurnal weather cycles or scheduled occupancy patterns). This framing allows us to pose the central question of this work: *Given a closed-loop black-box building dynamics simulator* $\mathcal{M}$*, how do different length-*T *disturbance scenarios* $w_{0:T} \sim \pi_W$ *influence the closed-loop system trajectories?*

In practice, controllers are often designed for a nominal (or worst-case) disturbance scenario but must be validated under a variety of realistic conditions: varying ambient weather, internal loads, or occupancy profiles, in order to assess robustness. Accordingly, a systematic, model-agnostic algorithm for identifying disturbance scenarios that span the space of possible output behaviors is critical both for performance assessment using digital twins and for informing the redesign or adaptation of control strategies.

Based on Definition 1, we define a collection of $K \in \mathbb{N}$ disturbance scenarios by

$$W_{K,T} \triangleq \left\{ w_{0:T}^{(k)} \right\}_{k=1}^{K},$$

where $w_{0:T}^{(k)}$ is the k-th length-T disturbance scenario in the collection $W_{K,T}$. Then, for some initial state $x_0^{(k)}$, and control sequence $u_{0:T}^{(k)}$, we can simulate output trajectories

$$Y_{K,T} \triangleq \left\{ y_{0:T}^{(k)} \right\}_{k=1}^{K}, \text{where } y_{0:T}^{(k)} = \mathcal{M}\left(x_0^{(k)}, u_{0:T}^{(k)}, w_{0:T}^{(k)}\right). \quad (2)$$

Definition 2 (Diversity)**.** *Given a distance measure* $d_Y : \mathcal{Y} \times \mathcal{Y} \to [0, \infty)$ *between trajectories, the diversity of the collection* $Y_{K,T}$ *in* (2) *is defined as*

$$\mathcal{D}(Y_{K,T}) \triangleq \frac{2}{K(K-1)} \sum_{1 \le i < j \le K} d_Y\left(y_{0:T}^{(i)}, y_{0:T}^{(j)}\right).$$

Definition 2 quantifies the span of the collection of outputs based on a chosen distance metric d_Y. If we choose $\mathcal{I}$ to be a scoring function that quantifies the diversity of a collection of output signals $Y_{K,T}$, then we can obtain a *representative set of disturbance scenarios* by maximizing the diversity of these output trajectories. That is,

$$W_{K,T}^{\star} = \underset{W_{K,T} \subset \mathcal{W}_{K,T}}{\operatorname{argmax}} \; \mathcal{I}\left(\mathcal{D}\left(Y_{K,T}\right)\right). \quad (3)$$

Here, $\mathcal{W}_{K,T}$ denotes the space of admissible length-T scenarios.

2.2. Proposed Solution

Several challenges arise in solving (3). First, the objective function is 'black-box'. This is because the map from $W_{K,T}$ to $Y_{K,T}$ requires a simulation through the black-box model $\mathcal{M}$. Therefore, acquiring gradients and using local information is computationally intractable. Second, the admissible space of solutions $\mathbb{W}_{K,T}$ is not straightforward to model such that (3) is tractable. We reiterate that $\mathbb{W}_{K,T}$ are disturbance time-series $w_{0:T}$ of length T, but in order for such disturbances to be domain-feasible, one cannot randomly select such disturbance trajectories; instead, they must be drawn from π_W, which does not admit a simple closed-form representation. While handcrafting nominal disturbance scenarios may be possible for certain kinds of disturbances such as solar radiation, it is not a scalable approach and requires considerable time and effort, along with customization for specific use-cases. Furthermore, some disturbances such as occupancy-driven heat-loads are much harder to design by hand, and susceptible to changing social patterns (e.g., pre- vs. post-pandemic) and technical advancements.

A more practical path forward is to use disturbance data directly; that is, partition the measured disturbance into disturbance scenarios on fixed time lengths, e.g. $T = 24$ hr. Such datasets are often available, because the disturbances in question, though hard to predict long-term, are measurable with satisfactory accuracy by sensors equipped in the building system. By fixing the length T, one can split a long trace of collected w data into the form $\mathbb{W}_{K,T}$, and the resulting diversity maximization problem (3) becomes a combinatorial problem. Such a combinatorial problem still needs to be solved efficiently in case the number of scenarios in $\mathbb{W}_{K,T}$ is large, especially because each simulation may take considerable time to execute to completion.

When available data is too small, we can resort to synthesizing artificial data using time-series generative networks (TSGNs) (see Fig. 1); in particular, we study the utility of time-series diffusion networks to construct synthetic disturbance scenarios that can be appended to a small-sized dataset collected from the building. A benefit of using generative models for this purpose is that they can generate an arbitrarily large number of scenarios as needed. For this paper, we empirically find the synthetic scenarios from our chosen diffusion-based TSGN to be both plausible/realistic and useful, including in regimes where the TSGN can only be trained on 1,000 real scenarios or less. Further investigation of trade-offs relating to the choice of TSGN method and/or training data are beyond the scope of this paper and left to future work. Since the TSGNs can generate very large amounts of synthetic data, one way of keeping the size of $\mathbb{W}_{K,T}$ manageable is by performing time-series clustering [37] on the synthetic samples (or indeed real samples if the amount of real data is very large), and consider the cluster centers as the elements of the search space. In either case, TSGNs provide an avenue to distill the complex search space $\mathbb{W}_{K,T}$ to a rich lattice of disturbance scenarios.

Once the search space $\mathbb{W}_{K,T}$ is fixed, we can pose (3) as a combinatorial black-box optimization problem. In this paper, we propose the use of multi-arm bandits (MABs) to obtain the representative set of disturbance scenarios. There are a few reasons for this. First, MABs are well established algorithms capable of handling combinatorial problems with complex objective functions [17] while being supported by theoretical guarantees on regret decay rates. Second, bandit algorithms perform well despite noise in the objective. In this work, there are two sources of noise: the first is the explicit sensor or process noise ν described in (1) which captures the model's inherent uncertainty and real-world sensor noise.

Another more intricate source of noise is if one uses pre-clustering to whittle down the set $\mathbb{W}_{K,T}$ to a manageable size. Since a cluster center may not directly represent a feasible disturbance trajectory, we propose randomly selecting a disturbance trajectory embedded within the cluster itself since such a disturbance trajectory will either be real, or synthesized based on a TSGN replicating the real disturbance time-series distribution. Since we rely on a random selection procedure of elements within a cluster, the simulated outputs will be different even when the same cluster center index is chosen twice. Such an approach inevitably introduces noise in evaluating the scoring function $\mathcal{I}$. To combat this, we propose the use of MABs that are effective in handling such stochasticity.

3. Methodology

The proposed workflow for selecting representative disturbance scenarios is shown in Figure 2. We frame the problem in this work as a CMAB problem, in which we apply a well-known CMAB algorithm: combinatorial upper confidence bound (CUCB). We consider two cases: (i) the big-data case, where we have access to a large number of disturbance scenarios, and (ii) the small-data case, a Diffusion-TS network generates synthetic disturbance data to augment the original dataset.

3.1. Finding Representative Scenarios with MABs

MAB is a special type of sequential decision-making framework [17], characterized by a set of *arms* (e.g., a particular action to a decision problem) and a *scoring function* or reward obtained by 'pulling' an arm (e.g., outcome observed upon executing the action). Among variants of MABs, we are specifically interested in combinatorial MABs (CMABs) [19, 38], where the reward is defined based on a subset of arms, rather than an individual arm. Unlike most of the typical MAB problems that aim to identify the best arm while incurring minimal regret, CMAB aims to identify a best set of arms with a fixed cardinality while incurring minimal regret. Here, regret formalizes the sub-optimality encountered while dealing with the exploration-exploitation dilemma inherent in such sequential decision making problems. As described in Section 2, we seek to identify a set

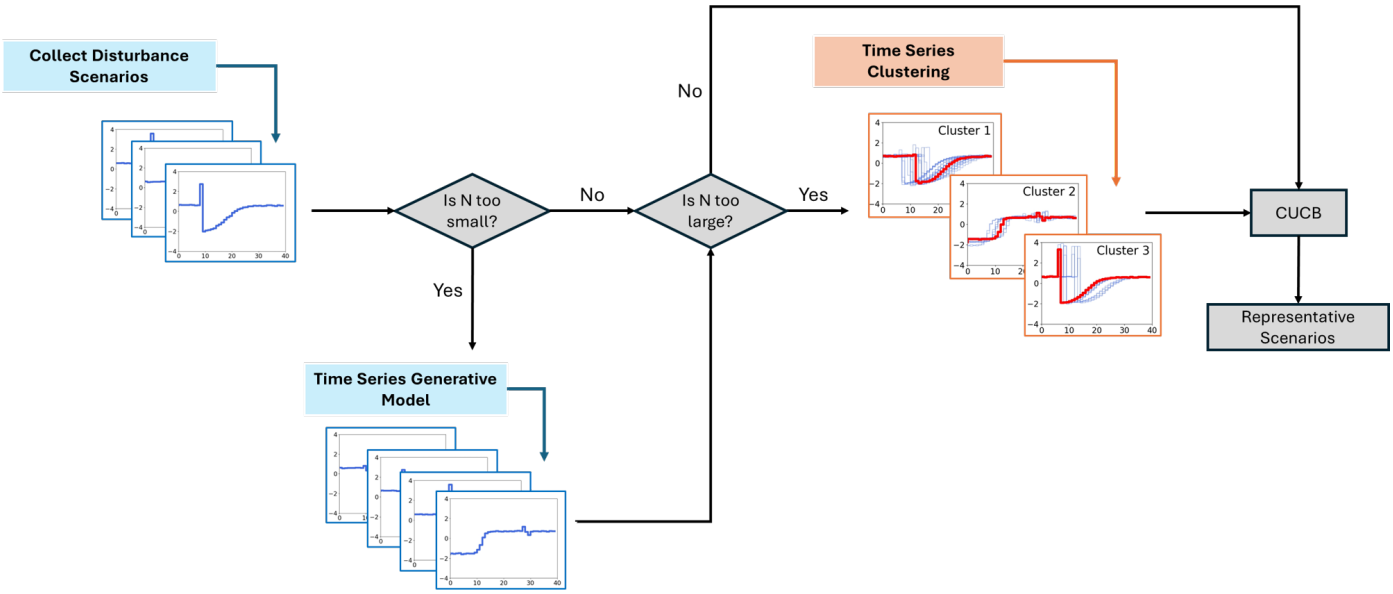

Figure 2: Workflow for the proposed solution framework.

of disturbance scenarios that "best" maximizes (3) while exploring combinatorially many subsets.

In our problem, the arms of the CMAB are the K scenarios in the disturbance dataset $W_{K,T}$. Given a subset of arms (a subset of arms is referred to as a *super arm*), we evaluate the scoring function $\mathcal{I}$ by executing a closed-loop GEB simulation with each of the selected scenarios and aggregating the outcomes or observables of those simulations. Typical CMAB algorithms are iterative in nature, where, in each round, the algorithms propose a potential subset candidate that solves (3), evaluates $\mathcal{I}$, and then repeats the process until a termination criterion is met. More concretely, to formulate problem (3) as a CMAB problem, each arm i is defined as a disturbance scenario $W_{i,T}$; and we suppose the total number of arms is N_s, where N_s is the number of samples drawn from the space $\mathbb{W}_{K,T}$ to make the action space finite. In practical terms, N_s is the size of the available dataset of disturbance scenarios. The corresponding output trajectory $y^{(k)}_{0:T}$ of arm k is a noisy sequence of observables obtained by executing a closed-loop GEB simulation with the k-th arm's associated disturbance scenario. Clearly, the total number of super arms is $\binom{N_s}{K}$, which for moderately large N_s is difficult to exhaustively search, especially when each closed-loop simulation is slow.

Let us now discuss the reward $\mathcal{I}$ for super arm S_t in round t, whose objective is to maximize diversity among the simulated outputs. We do this by computing the average pairwise DTW distance between all observed output trajectories. Specifically, by simulating all scenarios in the super arm S_t, we obtain the set of output sequences $Y^{(t)}_{K,T}$, whose k-th output sequence is $y^{(k)}_{0:T} = \mathcal{M}(x_0, u^{(k)}_{0:T}, w^{(k)}_{0:T})$. One can then compute the pairwise dynamic-time-warped (DTW) distance between each pair of these K output sequences, and average the DTW distances as an aggregated score. Given two trajectories $y^{(1)}_{0:T_1}$ and $y^{(2)}_{0:T_2}$), the DTW distance is given by

$$\mathrm{DTW}(y^{(1)}_{0:T_1}, y^{(2)}_{0:T_2}) = \min_{\pi \in \Pi_{T_1,T_2}} \left(\sum_{(s_1,s_2)\in\pi} \rho(y^{(1)}_{s_1}, y^{(2)}_{s_2}) \right),$$

where Π_{T_1,T_2} denotes the set of all valid alignment paths π between the indices of the two trajectories, and $\rho(\cdot,\cdot)$ is a local distance metric (e.g., Euclidean distance). Maximizing the average pairwise DTW distance maximizes the diversity of the time-series sequences.

To this end, a popular CMAB algorithm is CUCB, described in Algorithm 1, which has several desirable theoretical properties. The main is (roughly) that, for a user-defined threshold of sub-optimality, one can estimate how many rounds the CMAB needs to be iterated to achieve that level of sub-optimality [19].

CUCB extends the classical upper confidence bound algorithm for solving conventional MAB problems [39]. Let T_i denote the total number of times arm i has been played up to round t, and let $\hat{\mu}_i$ denote the empirical mean of all observed outcomes $\{\mathcal{I}_{i,R}\}$ for arm i up to round t, that is $\hat{\mu}_i = \sum_{R=1}^{t} \mathcal{I}_{i,R}/T_i$. CUCB defines the outcome expectation variable $\bar{\mu}_i$ for each arm i as the upper confidence bound (UCB) of the empirical outcome mean $\hat{\mu}_i$, given by

$$\bar{\mu}_i = \hat{\mu}_i + \sqrt{\frac{3\log t}{2\,T_i}}. \tag{4}$$

The UCB balances exploration and exploitation, enabling the algorithm to select arms with both high empirical means (large $\hat{\mu}_i$) and low play counts (small T_i). In the early stage, the algorithm is more "explorative" due to the smaller value of T_i; in the later stage, the algorithm is more "exploitive" due to the larger value of T_i.

Algorithm 1 CUCB Algorithm

Required: Super arm size K;
Admissible disturbance scenarios space $\mathbb{W}_{K,T}$;
Total number of CUCB rounds N_∞;
Number of times arm i has been played T_i

Time-series clustering:
if $|\mathbb{W}_{K,T}|$ is too large **then**
  Specify number of arms N_s
  TS-CLUSTER($\mathbb{W}_{K,T}, N_s$)
else
  $N_s \leftarrow |\mathbb{W}_{K,T}|$
end if
Initialization:
for each arm $i = 1, \ldots, N_s$ **do**
  Select an arbitrary super arm $S \in \mathcal{S}$ such that $i \in S$
  SIMULATION(S)
end for
for $t = N_s, \ldots, N_s + N_\infty$ **do**
  for each arm $i = 1, \ldots, N_s$ **do**
    Compute UCB:

$$\bar{\mu}_i = \hat{\mu}_i + \sqrt{\frac{3 \ln t}{2T_i}}$$

  end for
  Select the top-K arms with the highest UCB values as the super arm S_t
  SIMULATION(S_t)
end for
Select the representative super arm S^* as the one with the largest $\mathcal{I}$

function TS-CLUSTER($\mathbb{W}_{K,T}, N_s$)
  Perform time-series clustering algorithm on $\mathbb{W}_{K,T}$ to generate N_s clusters.
  return clustered data
end function

function SIMULATION(S)
  Perform GEB simulation for disturbances in S
  Calculate observed outcome $\mathcal{I}_{i,R}$ for each arm $i \in S$
  Calculate reward $\mathcal{I}$ for S
  Update T_i and $\hat{\mu}_i$ for all $i \in S$
end function

3.2. Generating Synthetic Data via Time-Series Diffusion Models

In this work, we investigate the benefits of supplementing real disturbance scenarios with synthetically-generated ones, especially in situations where the number N_s of available real scenarios is too small. As generator, we propose leveraging the recent Diffusion-TS model [31], a TSGN that can generate fixed-length time series. Diffusion-TS extends the denoising diffusion probabilistic models (DDPMs) proposed in [33]. Same as the DDPM, Diffusion-TS is based on diffusion, i.e., two core reciprocal processes: a forward noising process and a backward denoising process. For the forward process, independent Gaussian noise $\epsilon_t \sim \mathcal{N}(0, I)$ is gradually added to an input sample $w_0 \in \mathbb{R}^d$ drawn from an unknown data distribution to create a noised sequence $w_1, w_2, \cdots, w_T$ ending at $w_T \sim \mathcal{N}(0, I)$ and following the discrete Markov process

$$w_t = \sqrt{1 - \beta_t} w_{t-1} + \sqrt{\beta_t} \epsilon_t, \tag{5}$$

with $\{\beta_t \in (0, 1)\}_{t=1}^T$ a variance schedule pre-specified by the user. In this subsection, t denotes the time-step of the diffusion process, not the time-index of the discrete-time dynamics in the simulator. Using $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{k=1}^t \alpha_k$ and the reparameterization trick from [40], we can also express each sequence step as a noisy version of input x_0 following

$$w_t = \sqrt{\bar{\alpha}_t} w_0 + \sqrt{1 - \bar{\alpha}_t} \bar{\epsilon}_t, \tag{6}$$

with $\bar{\epsilon}_t \sim \mathcal{N}(0, I)$.

Conversely, the backward process should map a Gaussian noise sample $w_T \sim \mathcal{N}(0, I)$ back to an input sample w_0. However, performing this process exactly would require access to typically intractable quantities, such as the conditional distribution of w_{t-1} given w_t. Diffusion circumvents this issue by learning a parameterized model to approximate the backward process.

Diffusion-TS differs from the original DDPM framework [33] in predicting directly an input sample estimate, i.e., $\hat{w}_0(w_t, t, \theta)$, with θ the parameterization of the predictor model, and by changing the objective away from likelihood maximization. The training objective is instead the minimization of a distance between the input sample w_0 and its predicted estimate $\hat{w}_0(w_t, t, \theta)$ for any t. More precisely, Diffusion-TS proposes to minimize a weighted combination of the L_1 distances between a) original and estimated signals, and b) original and estimate signal Fourier transforms, computed via fast Fourier transform (FFT) [41]. Additionally, inspired by the performance gain obtained in DDPM from a similar modification, a weight w_t is added to down-weight, in relative terms, the loss corresponding to smaller diffusion step values t. The loss is ultimately expressed as

$$L_\theta = \mathbb{E}_{t, w_0} \left[\gamma_t \left(\lambda_1 \| w_0 - \hat{w}_0(w_t, t, \theta) \|_1^2 + \lambda_2 \| \mathsf{FFT}(w_0) - \mathsf{FFT}(\hat{w}_0(w_t, t, \theta)) \|_1^2 \right) \right],$$

where $\gamma_t = \frac{\lambda \alpha_t (1 - \bar{\alpha}_t)}{\beta_t^2}$ with λ a small constant, w_t is obtained through the forward process in (6), and λ_1, λ_2 are hyperparameters specified by user.

The predictor network for $\hat{w}_0$ is structured as a transformer-like encoder-decoder architecture with stacked blocks designed to better capture multi-scale patterns and correlations within time-series data [31]. Each of the D decoder blocks is built around a backbone combined a residual multi-head self-attention layer followed by a residual multi-head cross-attention layer. The resulting output, for j-th block, is then converted into three distinct time series. The first, "output" R_j, is fed to the next decoder block. The second, "trend synthesis" V_j, is a time series corresponding to the sum of a fitted small-degree polynomial with

the time-average of R_j. The third, "seasonality and error synthesis" S_j, is a time series corresponding to the superposition of a finite number of fitted phase-shifted frequency components. The final estimate is computed as the sum of the "synthesis" time series from all decoder blocks, and the time-series output from the last one:

$$\hat{w}_0(w_t, t, \theta) = R_D + \sum_{j=1}^{D} \left(V_j + S_j\right). \tag{7}$$

Here, we first train Diffusion-TS on the entire set of real disturbance scenarios we collected from a real building. Then, we generate any desired number of synthetic disturbance scenarios from our trained network as estimates $\hat{w}_0(w_T, T, \theta)$ for random Gaussian noise samples $w_T \sim \mathcal{N}(0, I)$. The hyperparameters of the diffusion model and solver that we use are provided in Table 1. In order to maintain realistic synthetic profiles, some amount of post-processing may have to be done, for instance ensure solar irradiance is zero during night hours corresponding to the day of the year, along with other smoothing and filtering as required, to ensure realistic rates-of-change or magnitudes.

Model Parameters	
Sequence length	96
Dimension of input/output features	3
Number of encoder blocks	4
Number of decoder blocks	4
Model hidden dimension	96 (24 × 4)
Number of attention heads	4
Number of diffusion sampling steps	100
Variance schedule $\{\beta\}$	cosine
Transformer feed-forward hidden dimension	384 (96 × 4)
Transformer attention dropout probability	0.0
Transformer residual dropout probability	0.0
Seasonality convolution kernel length	3
Seasonality convolution padding length	1
Optimizer Parameters	
Base learning rate	1.0e-5
Maximum number of epochs	10 000
Number of batches for gradient accumulation	2
Model save frequency (epochs)	1 000
EMA decay factor	0.995
EMA update interval (training steps)	10
Scheduler Parameters	
Learning-rate reduction factor	0.5
Scheduler patience (epochs)	300
Minimum learning rate	1.0e-5
Minimum change threshold	1.0e-1
Threshold mode	relative
Learning rate after warm-up	8.0e-4
Number of warm-up steps	100

Table 1: Combined hyper-parameter configuration used in our diffusion-transformer experiments.

3.3. Customization for Large Disturbance Datasets

Despite the potential of improving the optimal achievable reward for the CMAB problem by adding additional arms to the original arm set, it would also make the theoretical regret bound of the CUCB algorithm worse; c.f. [19]. In other words, more bandit rounds N_{CMAB} would be required to attain a good solution, which is not desirable because each simulation is expensive. This situation may arise if we greatly expand the number of admissible disturbance scenarios by incorporating synthetic disturbances.

To counteract this, one approach is to use time series k-means clustering [42] to partition disturbance scenarios into multiple clusters, where each cluster contains disturbance scenarios with similar (in some metric) trajectories. We then treat each cluster as an independent arm and apply the CUCB algorithm to select the K representative clusters with diverse output trajectories. With the clustering approach, the CUCB algorithm would randomly pick a disturbance scenario from the disturbance set of the selected cluster $i \in [N_s]$ to perform simulation, introducing randomness to the problem, which MABs are well-equipped to handle. An alternative to random selection is to choose the disturbance scenario closest to the cluster center, which is how we proceed in this paper. The expected reward $r_\mu(S_t)$ of playing super arm S_t can then be calculated using the output trajectories of each disturbance scenarios in super arm S_t.

Figure 2 illustrates the overall workflow of our proposed solution framework for discovering disturbance scenarios (clusters) that produce diverse simulation output trajectories. We begin by collecting disturbance scenarios from real data. If the number of disturbance scenarios is too small, we train a TSGM, such as the Diffusion-TS [31], on the real disturbance data set to generate synthetic disturbance scenarios. If the number of disturbance scenarios (including both real and synthetic data) is too large, we employ a time-series clustering algorithm to partition the scenarios into multiple clusters. The CUCB algorithm is then applied to the clustered disturbance scenarios. For comparison, we also perform CUCB directly on the original, non-clustered disturbance data. Ultimately, the CUCB algorithm selects a set of representative disturbances (or cluster centers) that best capture the diversity of simulation output trajectories.

4. Experimental Results

4.1. Grid-interactive building simulation model

4.1.1. Zone thermal dynamics

An air-to-air heat pump (HP) system comprises of outdoor and indoor heat exchangers, a single-speed compressor, and an expansion valve, facilitating efficient thermal regulation within a single-zone residential space. The system operates in two modes—cooling and heating—by modulating refrigerant flow to absorb or release heat within the indoor unit's heat exchanger, ensuring precise temperature control. This study presents control-oriented models for the building zone, HP, and thermostat, which collectively optimize system performance to maintain the desired indoor climate. The proposed thermal dynamic framework is applicable to both ducted and ductless HP configurations

by appropriately parameterizing system capacity and efficiency, offering a scalable solution for residential thermal management.

We model a single-zone space for the purposes of illustrating our approach. In particular, we use the 4-state RC model [43] given by,

$$C_{\mathsf{w}_{\mathsf{ext}}}\dot{T}_{\mathsf{w}_{\mathsf{ext}}} = \frac{3}{R_{\mathsf{wall}}}\left(T_{\mathsf{saw}} + T_{\mathsf{w}_{\mathsf{int}}} - 2T_{\mathsf{w}_{\mathsf{ext}}}\right) \tag{8a}$$

$$C_{\mathsf{w}_{\mathsf{int}}}\dot{T}_{\mathsf{w}_{\mathsf{int}}} = \frac{3}{R_{\mathsf{wall}}}\left(T_{\mathsf{w}_{\mathsf{ext}}} + T_{\mathsf{sin}} - 2T_{\mathsf{w}_{\mathsf{int}}}\right) \tag{8b}$$

$$\begin{aligned} C_{\mathsf{in}}\dot{T}_{\mathsf{in}} = {} & \frac{T_{\mathsf{amb}} - T_{\mathsf{in}}}{R_{\mathsf{wind}}} + \frac{T_{\mathsf{amb}} - T_{\mathsf{in}}}{R_{\mathsf{door}}} + \frac{T_{\mathsf{sin}} - T_{\mathsf{in}}}{R_{\mathsf{w}_{\mathsf{int}}}} \\ & + \frac{T_{\mathsf{itm}} - T_{\mathsf{in}}}{R_i} + \frac{T_{\mathsf{sar}} - T_{\mathsf{in}}}{R_{\mathsf{roof}}} - P_{\mathsf{hp}} + P_{\mathsf{int}} + P_{\mathsf{sol}_{\mathsf{in}}} \end{aligned} \tag{8c}$$

$$C_{\mathsf{itm}}\dot{T}_{\mathsf{itm}} = \frac{T_{\mathsf{in}} - T_{\mathsf{itm}}}{R_i} + \frac{T_{\mathsf{grnd}} - T_{\mathsf{itm}}}{R_{\mathsf{floor}}} + P_{\mathsf{sol}_{\mathsf{itm}}}, \tag{8d}$$

where the four-dimensional state includes the exterior wall temperature ($T_{\mathsf{w}_{\mathsf{ext}}}$), the interior wall temperature ($T_{\mathsf{w}_{\mathsf{int}}}$), the indoor air temperature of the zone (T_{in}), and the internal thermal mass (T_{itm}). The single-dimensional control input is the heat pump heating or cooling capacity (P_{hp}). The system is perturbed by three exogenous disturbances to the zone: 1) the ambient temperature T_{amb}, 2) the power influx due to solar irradiance I contributing to indoor air temperature and indoor thermal mass, and 3) the internal heat load P_{int} generated, for instance, by appliances, occupants, etc.

The thermal capacitances are given by $C_{\mathsf{w}_{\mathsf{ext}}}$, $C_{\mathsf{w}_{\mathsf{int}}}$, C_{in}, and C_{itm} corresponding to the states. The thermal resistances are R_{door}, R_{wall}, $R_{\mathsf{w}_{\mathsf{int}}}$, R_{wind}, R_i, along with the ground resistance R_{floor}, and the values of these parameters are provided in Table 2. Note that the ground temperature T_{grnd} does not matter if floor is well insulated, i.e. $R_{\mathsf{floor}} = \infty$.

Other variables in (8) may be expressed directly in terms of the state, input, and disturbance variables identified above. Specifically, (8) includes several additional temperatures are required to describe the thermal dynamics, which we describe next. In (8), T_{sin} is the wall surface temperature of the inner wall, T_{saw} and T_{sar} represent the combined effect of solar radiation, outdoor air temperature, and convective heat transfer on the exterior wall and roof outer surface respectively, and and T_{grnd} is the ground temperature. Here, T_{in} may be calculated by the conservation law while T_{saw} and T_{sar} are affine transformations of the disturbance variables,

$$(T_{\mathsf{w}_{\mathsf{int}}} - T_{\mathsf{sin}})/(R_{\mathsf{wall}}/3) = (T_{\mathsf{sin}} - T_{\mathsf{in}})/R_{\mathsf{w}_{\mathsf{int}}}, \tag{9a}$$

$$T_{\mathsf{saw}} = (\alpha_{\mathsf{w}_{\mathsf{ext}}})/(h_{\mathsf{w}_{\mathsf{ext}}})F_{\mathsf{w}_{\mathsf{ext}}}I + T_{\mathsf{amb}}, \tag{9b}$$

$$T_{\mathsf{sar}} = (\alpha_{\mathsf{w}_{\mathsf{roof}}})/(h_{\mathsf{w}_{\mathsf{roof}}})F_{\mathsf{w}_{\mathsf{roof}}}I + T_{\mathsf{amb}}, \tag{9c}$$

Additional heat inputs to the space due to solar radiation include:

$$P_{\mathsf{sol}_{\mathsf{wind}}} = A_{\mathsf{wind}}F_{\mathsf{wind}}\eta_{\mathsf{sol}}I, \tag{10}$$

which models the solar heat gain through windows; η_{sol} denotes the solar heat gain coefficient that abstracts the fraction of radiation that actually enters the building space. This total solar heat is then split into

$$P_{\mathsf{sol}_{\mathsf{in}}} = fP_{\mathsf{sol}_{\mathsf{wind}}}, \quad P_{\mathsf{sol}_{\mathsf{itm}}} = (1-f)P_{\mathsf{sol}_{\mathsf{wind}}} \tag{11}$$

which represent contributions towards heating the indoor air and the internal thermal mass, respectively. Some specific calculations involving view factors and window areas are left unwritten for brevity; we refer the reader to [43, 44] for further details.

Note that the heat pump capacity P_{hp} is an auxiliary variable introduced to simplify the model (15). In practice, P_{hp} relates to the compressor power in the physical world P_{comp} via the following piecewise linear function that separates the coefficients of performance in each operating mode, η_{heat} and η_{cool},

$$P_{\mathsf{comp}} = \begin{cases} P_{\mathsf{hp}}\eta_{\mathsf{heat}} & \text{if } P_{\mathsf{hp}} \geq 0, \\ P_{\mathsf{hp}}\eta_{\mathsf{cool}} & \text{otherwise.} \end{cases} \tag{12}$$

In addition to the heat generated by internal and external sources, the grid-interactive building has access to battery storage, and photovoltaic (PV) systems for harnessing solar energy. In particular, the PV and battery system offers an additional degree of control as we can control the charge and discharge of the battery. Specifically, we assume the battery has first-order charge/discharge dynamics given by

$$\dot{Q}_{\mathsf{bat}} = \eta_{\mathsf{bat}_{\mathsf{chg}}}P_{\mathsf{bat}_{\mathsf{chg}}} - \frac{P_{\mathsf{bat}_{\mathsf{dis}}}}{\eta_{\mathsf{bat}_{\mathsf{dis}}}}, \tag{13}$$

where both charge and discharge cycles operate at different efficiencies. Here, we require,

$$P_{\mathsf{bat}_{\mathsf{chg}}} = 0 \text{ or } P_{\mathsf{bat}_{\mathsf{dis}}} = 0, \tag{14}$$

pointwise-in-time to enforce mutually exclusive operating regimes.

Using a zero-order hold with sampling time Δt, we model the dynamics of the single-zone space system (including Q_{bat}) using the following discrete-time, perturbed, linear dynamics,

$$x_{t+1} = Ax_t + Bu_t + Fw_t, \tag{15}$$

with state $x_t = [T_{\mathsf{w}_{\mathsf{ext}}}, T_{\mathsf{w}_{\mathsf{int}}}, T_{\mathsf{in}}, T_{\mathsf{itm}}, Q_{\mathsf{bat}}] \in \mathbb{R}^5$, input $u_t = [P_{\mathsf{hp}}, P_{\mathsf{bat}_{\mathsf{chg}}}, P_{\mathsf{bat}_{\mathsf{dis}}}] \in \mathbb{R}^3$, disturbance $w_t = [T_{\mathsf{amb}}, I, P_{\mathsf{int}}] \in \mathbb{R}^3$, and matrices A, B, F with appropriate dimensions defined using (8), (9), (10), (11), and (13). Table 2 describes the simulation parameters used in this work.

4.2. Constraints on the system

We now briefly describe the physical constraints on the system (15) that must be satisfied for a desirable operation. Since we will use the discrete-time dynamics (15), we enforce these constraints at the discrete points in time.

Parameter	Value	Parameter	Value
$C_{\mathsf{w}_{\mathsf{ext}}}$	1.18×10^7	$C_{\mathsf{w}_{\mathsf{int}}}$	1.18×10^7
C_{in}	6.66×10^5	C_{itm}	1.63×10^7
R_{wall}	45×10^{-3}	R_{roof}	54×10^{-2}
R_{floor}	∞	R_{wind}	1×10^{-1}
R_{door}	19×10^{-2}	$R_{\mathsf{w}_{\mathsf{int}}}$	71×10^{-4}
R_i	22×10^{-4}	f	0.30
η_{heat}	3.50	η_{cool}	2.70
η_{sol}	0.19	$\eta_{\mathsf{ch}}, \eta_{\mathsf{dis}}$	0.93×10^{-6}
$F_{\mathsf{w}_{\mathsf{ext}} \to \mathsf{w}_{\mathsf{in}}}$	0.50	$F_{\mathsf{roof} \to \mathsf{in}}$	0.20
$F_{\mathsf{w}_{\mathsf{in}} \to \mathsf{in}}$	0.30	$\alpha_{\mathsf{w}_{\mathsf{ext}}}$	0.13
α_{roof}	0.05	h_{roof}	3.45
$h_{\mathsf{w}_{\mathsf{ext}}}$	20.0	$A_{\mathsf{w}_{\mathsf{ext}}}$	52.2
A_{wind}	6.10	A_{roof}	65.9
N_{pv}	25	A_{pv}	1.685 sq. m.
N_{oct}	50	T_{stc}	25.0 deg-C
η_{stc}	0.19	$\eta_{\mathsf{l\&t}}$	0.90
η_{T}	0.005	T_{noc}	25.0 deg-C

Table 2: Grid-interactive building simulation model parameters.

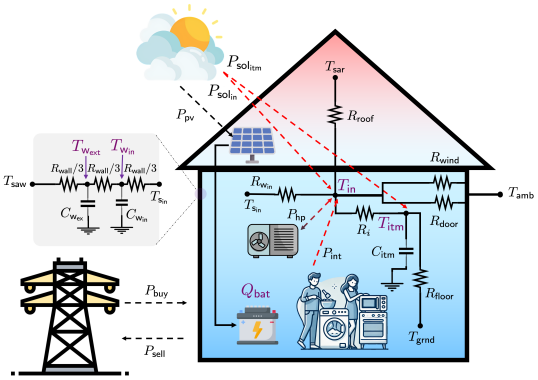

Figure 3: Grid-interactive building model schematic. States are shown in purple.

Recall that the measured variable from the battery system is the state-of-charge (SOC), given by

$$\mathsf{SOC} = \mathsf{Q}_{\mathsf{bat}}/\mathsf{Q}_{\mathsf{bat}}^{\max},$$

where $Q_{\mathsf{bat}}^{\max}$ is maximum energy stored in the battery. We enforce hard bounds on SOC by bounding Q_{bat},

$$Q_{\mathsf{bat}}^{\min} \le Q_{\mathsf{bat}} \le Q_{\mathsf{bat}}^{\max}. \tag{16}$$

Additionally, we place the following constraints on the inputs,

$$P_{\mathsf{hp}}^{\min} \le P_{\mathsf{hp}} \le P_{\mathsf{hp}}^{\max}, \tag{17a}$$
$$0 \le P_{\mathsf{bat}_{\mathsf{chg}}} \le P_{\mathsf{bat}_{\mathsf{chg}}}^{\max}(1 - s_{P_{\mathsf{bat}_{\mathsf{chg}}}}), \tag{17b}$$
$$0 \le P_{\mathsf{bat}_{\mathsf{dis}}} \le P_{\mathsf{bat}_{\mathsf{dis}}}^{\max} s_{P_{\mathsf{bat}_{\mathsf{chg}}}}, \tag{17c}$$
$$0 \le s_{P_{\mathsf{bat}_{\mathsf{chg}}}} \le 1. \tag{17d}$$

Note that (17b) and (17c) together enforce (14) exactly when $s_{P_{\mathsf{bat}_{\mathsf{chg}}}} \in \{0, 1\}$. To keep the formulation convex, we relax $s_{P_{\mathsf{bat}_{\mathsf{chg}}}} \in \{0, 1\}$ and impose an interval constraint on $s_{P_{\mathsf{bat}_{\mathsf{chg}}}} \in [0, 1]$ in (17d) instead.

Using an additional (slack) scalar variable $s_{T_{\mathsf{in}}} \ge 0$, we enforce a soft constraint on the indoor air temperature of the zone T_{in}, i.e., $T_{\mathsf{in}} \in [T_{\mathsf{in}}^{\min}, T_{\mathsf{in}}^{\max}]$ for some user-specified parameters $T_{\mathsf{in}}^{\min}, T_{\mathsf{in}}^{\max}$. Specifically, we enforce the following additional constraint on the system,

$$T_{\mathsf{in}}^{\min} - s_{T_{\mathsf{in}}} \le T_{\mathsf{in}} \le T_{\mathsf{in}}^{\max} + s_{T_{\mathsf{in}}}, \tag{18a}$$
$$s_{T_{\mathsf{in}}} \ge 0. \tag{18b}$$

By design, $s_{T_{\mathsf{in}}}$, when positive, is the margin of violation of the constraint $T_{\mathsf{in}} \in [T_{\mathsf{in}}^{\min}, T_{\mathsf{in}}^{\max}]$. We use the same slack variable $s_{T_{\mathsf{in}}}$ for all time instants.

4.2.1. Control objective

We formulate a convex objective J as a function of the state x and input u to balance the economic needs and desirable performance of the system. Specifically, the objective imposes two soft constraints on the system — minimize the use of compressor and minimize the margin of violation of temperature constraint (18). Additionally, the objective aims to minimize the power bought from the grid, which we formalize next.

Let P_{buy} denote the power to be bought from or sold to the grid at each time step, with

$$P_{\mathsf{buy}} \triangleq \underbrace{\left(P_{\mathsf{bat}_{\mathsf{chg}}} + P_{\mathsf{hp}}\right)}_{\text{power drawn from building}} - \underbrace{\left(P_{\mathsf{bat}_{\mathsf{dis}}} + P_{\mathsf{pv}}\right)}_{\text{power supplied to building}}. \tag{19}$$

Here, P_{pv} denotes the power output of the photovoltaic (PV) system. We model P_{pv} as a function of various system and environmental parameters,

$$P_{\mathsf{pv}} = N_{\mathsf{pv}} A_{\mathsf{pv}} \eta_{\mathsf{stc}} \eta_{\mathsf{l\&t}} I \left[1 - \left(\eta_{\mathsf{T}} \left(T_{\mathsf{amb}} + \frac{T_{\mathsf{noc}} - 20}{800} I - T_{\mathsf{stc}}\right)\right)\right]. \tag{20}$$

The total power generation depends on the number of PV panels, N_{pv}, and the area of each panel, A_{pv}, along with the panel efficiency under Standard Test Conditions (STC), η_{stc}, and a system loss efficiency factor, $\eta_{\mathsf{l\&t}}$. The incident solar irradiance, I, directly affects the power output, while temperature-dependent losses are accounted for through the term inside the brackets. These losses are influenced by the ambient temperature, T_{amb}, the Nominal Operating Cell Temperature (NOCT), T_{noc}, and the STC reference temperature, T_{stc}, with the efficiency reduction governed by the temperature coefficient, η_{T}. The NOCT parameter is used to estimate the actual cell temperature based on real-world irradiance conditions. This equation provides a comprehensive representation of PV power generation by incorporating both irradiance and temperature-dependent performance variations.

To achieve these preferences (minimizing power bought from the grid as well as enforcement of soft constraints),

we consider the following convex objective defined for a planning horizon N,

$$J(x,u) = \sum_{t=0}^{N} \max\left(\left(P_{\mathsf{buy}}\right)_t, 0\right) + \lambda_{\mathrm{HP}} \left\| \left(P_{\mathsf{hp}}\right)_{0:N-1} \right\|_1 + \lambda_{T_{\mathsf{in}}} s_{T_{\mathsf{in}}} + \lambda_{P_{\mathsf{bat}_{\mathsf{chg}}}} s_{P_{\mathsf{bat}_{\mathsf{chg}}}}, \quad (21)$$

where $\lambda_{\mathrm{HP}}, \lambda_{T_{\mathsf{in}}}, \lambda_{P_{\mathsf{bat}_{\mathsf{chg}}}} > 0$ are user-specified weights. Here, $\max\left(\left(P_{\mathsf{buy}}\right)_t, 0\right)$ corresponds to the power bought from the grid since instants where power is sold to grid will have $P_{\mathsf{buy}} < 0$, and $\left(P_{\mathsf{hp}}\right)_{0:N-1}$ denotes the sequence of P_{hp} inputs over the planning horizon N. In (21), we use ℓ_1-norm on P_{hp} to promote sparsity.

4.2.2. Predictive control algorithm

We use a receeding horizon control algorithm to control (15) subject to the physical constraints (16)–(18) and minimize the objective (21) Specifically, we solve the following (convex) optimization problem,

$$\begin{aligned} \text{minimize} \quad & \text{Cost } J \text{ in (21)}, \\ \text{subject to} \quad & \text{Dynamics (15)}, \\ & \text{SOC hard constraint (16)}, \\ & \text{Input constraints (17)}, \\ & T_{\mathsf{in}} \text{ soft constraint (18)}. \end{aligned} \quad (22)$$

Starting from an initial state x_0, we iteratively solve (22) for a sample disturbance sequence obtained from the generative model, apply the control input u_0, and repeat until a prespecified simulation window T. The convex MPC problem is implemented in Python using `CVXPY` [45] and solved with the `ECOS` solver [46], with a planning horizon of 32 time steps, which is equivalent to 8 hours.

4.3. Quality of diversity maximized scenarios

4.3.1. Real data

To begin with, we demonstrate the feasibility of our proposed solution framework for discovering diverse disturbance scenarios and output trajectories on the real disturbance dataset (number of disturbance scenarios $\mathbb{N} = 602$); all these scenarios are collected from the real building. Follow with the procedure shown in Figure 2, we partition the 602 disturbance scenarios into 50 clusters before implementing the CUCB algorithm to decrease the number of arm m. Our objective is to find the representative super arm containing $K = 5$ arms out of the 50 arms. We specify number of round $N_{\mathrm{CUCB}} = 500$ for the CUCB algorithm. Figure 4 shows the representative diverse disturbance scenarios and the corresponding simulation output trajectories recommended by CUCB on the real disturbance data set. From the left to the right column in Figure 4, we calculate the DTW distance with only the P_{buy} trajectory, the P_{pv} trajectory, and both output trajectories.

Figure 4 distils the full set of 602 real-world disturbance days into the five most informative scenarios, obtained with the CUCB combinatorial-bandit where diversity is quantified on, from left to right, P_{buy} alone, P_{pv} alone, and the joint trajectory $\{P_{\mathrm{buy}}, P_{\mathrm{pv}}\}$. In each column the upper three traces depict the exogenous drivers—global irradiance, internal heat gains, and ambient temperature—while the bottom two traces display the photovoltaic power export P_{pv} and the net grid exchange P_{buy}. When diversity is maximised on P_{buy} (left panel), the bandit favours combinations in which similar solar availability collides with contrasting temperature and load profiles, producing a wide spread in grid imports despite modest variation in P_{pv}. Conversely, maximising diversity on P_{pv} (centre panel) emphasises days with sharply different irradiance envelopes; P_{pv} now ranges from near-zero generation under persistent cloud cover to bell-shaped curves that clip at the inverter limit on clear days, while P_{buy} collapses into a narrow band because surplus PV largely offsets demand. The joint-metric selection (right panel) uncovers disturbance triplets that decouple the usual solar–grid anti-correlation: e.g. a cold yet intensely sunny winter day yields simultaneous peaks in both P_{pv} and the heating-driven P_{buy}, whereas a mild overcast shoulder season forces the building to rely almost exclusively on the grid. Taken together, Fig. 4 shows that, even within measured data, the bandit can isolate a compact scenario ensemble that stresses the controller along orthogonal operating axes—generation saturation, load-dominant import, and bidirectional power flow—thereby furnishing a rigorous basis for robustness assessment.

4.3.2. Benefits of synthetic data

As mentioned in Section 3.2, small number of arms m would limit the achievable optimal expectation reward for the CMAB problem. Creating synthetic disturbance scenarios enhances the diversity of the existing real disturbance dataset, potentially increasing the achievable optimal expected reward. To validate this, we visualize the projected disturbance time series in a 2D latent space using Principal Component Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP). Figure 5 compares the real disturbance data with 10k synthetic scenarios generated by Diffusion-TS. The results show that synthetic disturbances expand the covered latent region, demonstrating their ability to improve the diversity of the real disturbance dataset. With real data only, points cluster tightly around a single elongated manifold that reflects the seasonal co-variation of irradiance and temperature. Adding 10k diffusion-TS samples inflates that manifold in all directions: previously empty regions corresponding to low-temperature and low-irradiance winter anomalies, or to high-irradiance shoulder-season days with negligible internal gains, are now densely populated. Moreover, UMAP reveals new satellite clusters whose separation implies qualitatively distinct disturbance regimes: for instance, high-gain evening profiles decoupled from daytime solar, or abrupt temperature ramps uncorrelated with irradiance. This geometric expansion corroborates the performance gains reported in Fig. 6:

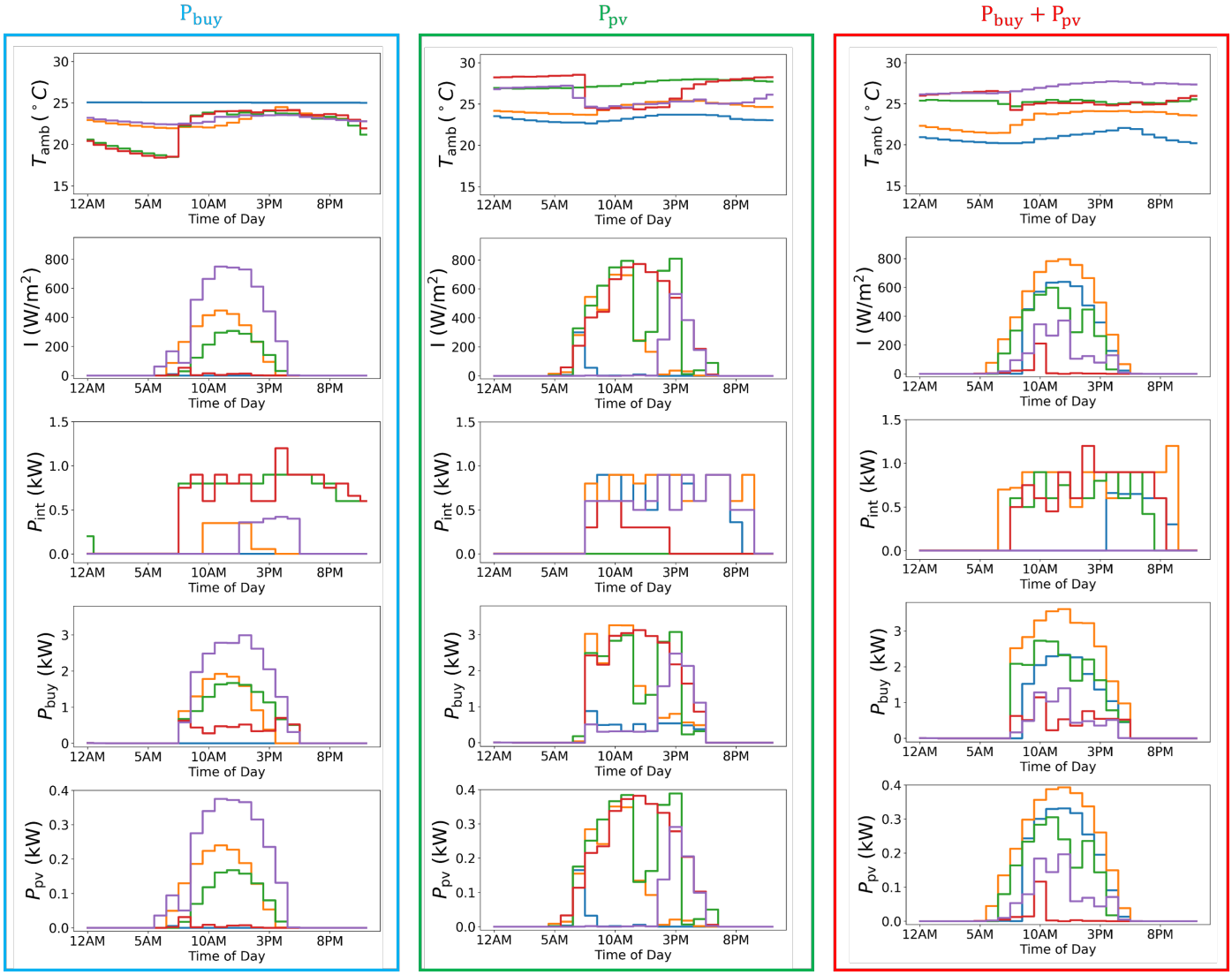

Figure 4: Diversity maximized scenarios for real disturbance data discovered by CUCB with DTW distance calculated using the P_{buy} (in W) output trajectory (Left), the P_{pv} (in W) output trajectory (Middle), and both P_{buy} and P_{pv} output trajectories (Right). The top 3 rows are the disturbance scenarios and the bottom 2 rows are the simulation output trajectories.

a larger, more isotropic latent footprint translates into more diverse closed-loop responses, thereby strengthening confidence in controller robustness across the full, and now better explored, operational envelope.

In this section, we further add 1,000 (denoted by 1k) synthetic disturbance scenarios generated by the Diffusion-TS model into the original dataset to investigate the potential improvement of the achievable reward for CUCB. Follow the procedure in Figure 2, we partition the total of 1,602 disturbance scenarios into 100 clusters to decrease the number of arm m. The objective is to find the super arm with representative $K = 8$ arms out of the 100 arms. We specify the number of round $N_{\text{CUCB}} = 2,000$ for the CUCB algorithm. Figure 6 shows the representative diverse disturbance scenarios and its corresponding output trajectories recommended from CUCB on the real+1k synthetic disturbance data set. The synthetic augmentation widens every disturbance dimension: extreme irradiance swings now include both prolonged low-sun periods and high-frequency transients; internal gains span evening-centric residential occupancy as well as daytime commercial profiles; and ambient temperature envelopes capture both heat-wave and cold-snap boundaries. These richer inputs propagate to the outputs: P_{pv} trajectories not only scale to higher peaks but also exhibit intermittent midday ramps characteristic of broken-cloud conditions, whereas P_{buy} stretches from sustained negative values—day-long export under surplus PV and muted demand—to deep import spikes driven by coincident low-solar, high-load episodes. Importantly, scenarios chosen on the joint diversity metric reveal operating points absent from the real-only set, such as a summer storm where solar production collapses abruptly while latent cooling loads remain high, forcing a rapid swing from export to import. Juxtaposition with Fig. 4 thus highlights the

value of generative augmentation: it exposes a broader envelope of coupled solar–temperature–load states, enabling the designer to probe controller performance under rare yet plausible extremes that historical measurements never recorded.

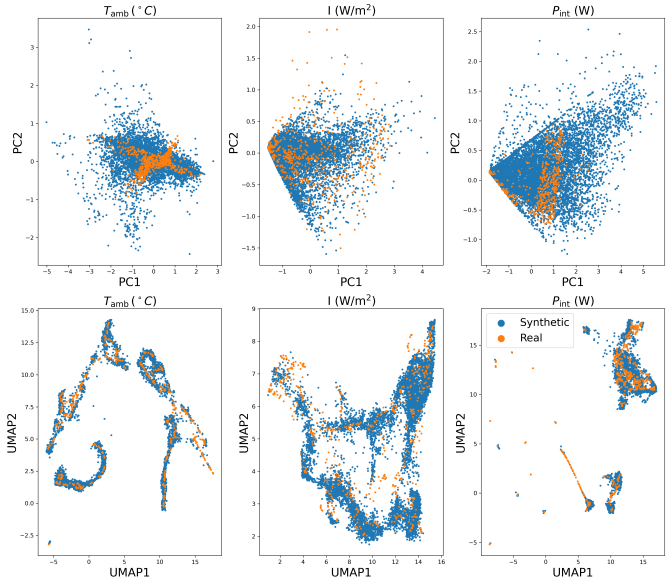

Figure 5: Projection of disturbance scenarios on 2-d latent space using PCA (top row) and UMAP (bottom row) for real and 10k synthetic disturbance data.

Furthermore, we want to understand the impact of adding synthetic data on different size of the real data set that we can access. Hence, we demonstrate the best achieved rewards defined in section 3.3 for CUCB on different size of the real data (20%, 60%, and 100% or the original data set) and different size of the real data + 1k synthetic data. For the case with only 20%, 60%, and 100% real data, we partition the disturbance scenarios into 10, 30, and 50 clusters. For the case with 20%, 60%, and 100% real data+1k synthetic data, we partition the disturbance scenarios into 70, 85, and 100 clusters. We set the number of representative scenario $K = 10$ for both real and real+1k synthetic data set. Results for the 20% and 60% real data are the mean of the randomly selected 5 data set. Achieved reward for different size of the real data are shown in Figure 7. As shown in Figure 7, incorporating synthetic data significantly enhances CUCB's achieved optimal reward. This improvement is particularly pronounced when the original real dataset is small, highlighting the benefits of synthetic data augmentation.

4.4. Comparative study with state-of-the-art

To demonstrate the benefit of framing the optimization problem shown in Eq. 3 as a CMAB problem and apply the CUCB algorithm for tackling it, we further compare the performance of the CUCB algorithm with two greedy approaches with and without clustering information. We consider both the real data set and the data set with real + 1k synthetic disturbance scenarios as the separate case studies. Again, we implement TS k-means clustering method to partition disturbance scenarios into different clusters to decrease the total number of arms. The objective for all baseline methods is to select the best-K representative clusters (arms) from the total m clusters (arms) to achieve the highest reward defined in section 3.3.

For the greedy approach without clustering information, we don't implement the clustering approach and treat each disturbance scenario as an individual arm. Hence, each arm i consists of single disturbance scenario, i.e. $\mathbb{N}_i = 1$. In every round of the algorithm, we perform simulations for all remaining disturbance scenarios in the disturbance data set $W_{\mathbb{N},T} \triangleq \left\{w_{0:T}^{(k)}\right\}_{k=1}^{\mathbb{N}}$ to obtain the simulation output trajectory set $Y_{\mathbb{N},T} \triangleq \left\{y_{0:T}^{(k)}\right\}_{k=1}^{\mathbb{N}}$. We select the i-th disturbance scenario whose corresponding simulation output trajectory $y_{0:T}^{i}$ has the largest mean pairwise DTW distance from all other output trajectories $y_{0:T}^{j}$, where $j \neq i \in \mathbb{N}$. The selected disturbance/outcome is removed from the disturbance/outcome set in each iteration, and the process continues until K disturbance scenarios are selected. Details for the greedy approach without clustering information are shown in Algorithm 2.

For the greedy approach with clustering information, we apply TS k-means clustering algorithm to partition disturbance scenarios into m clusters. We randomly select a disturbance scenario from all remaining m clusters and perform simulation to generate the simulation output trajectory set $Y_{m,T} \triangleq \left\{y_{0:T}^{(k)}\right\}_{k=1}^{m}$ at each iteration. We select the i-th cluster whose corresponding simulation output trajectory $y_{0:T}^{i}$ has the largest mean pairwise DTW distance from all other cluster's output trajectories $y_{0:T}^{j}$, where $j \neq i \in m$. We remove the selected cluster from the cluster set at every round. The algorithm stops when we sample K clusters. Detailed descriptions of the greedy approach with clustering information are provided in Algorithm 3.

Due to the intractability of finding the true optimal super arm S^* from the large number of possible combinations in the super arm set, the regret defined in section 3.1 cannot be computed directly. Therefore, we use the best achieved reward up to round t as the evaluation metric for all methods. Noted that all simulation considered in the experiments section is noise-free.

We firstly demonstrate results on real disturbance data set. We partition 602 disturbance scenarios into 50 clusters. We consider $K = 5, 10, 20$ representative clusters with diverse simulation outputs trajectories. For the CUCB algorithm, we set the number of rounds $N_{\text{CUCB}} = 500$. The optimal rewards achieved by greedy approaches and the CUCB algorithm are shown in Table 3. Next, we compare the performance of CUCB with the greedy approaches on the data set with real + 1k synthetic disturbance scenarios. We partition the 1,602 disturbance scenarios into 100 clusters. We set the number of rounds $N_{\text{CUCB}} = 1,500$ for the

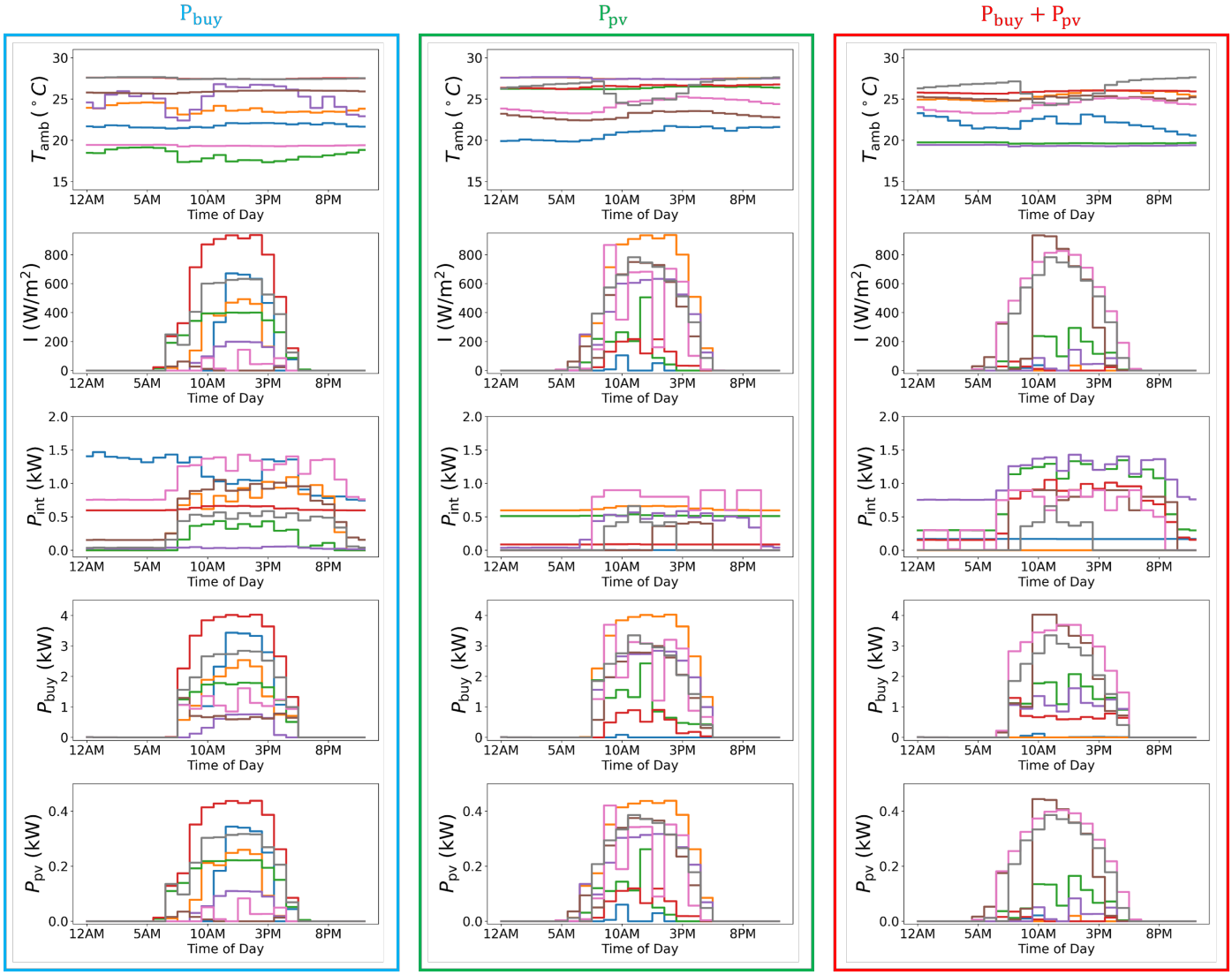

Figure 6: Diversity maximized scenarios for real + 1k synthetic data with DTW distance calculated using the P_{buy} (in W) output trajectory (Left), the P_{pv} (in W) output trajectory (Middle), and both P_{buy} and P_{pv} output trajectories (Right). The top 3 rows are the disturbance scenarios and the bottom 2 rows are the simulation output trajectories.

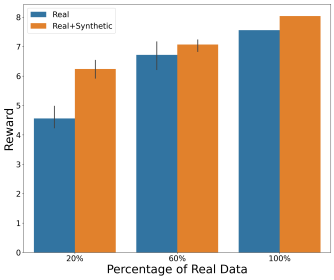

Figure 7: Best reward achieved by CUCB with different size of real disturbance data set. The blue bar shows the results with only $N\%$ of the original real dataset. The orange bar shows the results with $N\%$ of the original real dataset + 1k synthetic data. We observe that adding synthetic data can improve the achieved reward and has the greatest benefit while the real data size is small.

CUCB algorithm. Again, we consider $K = 5, 10, 20$ representative clusters. Optimal reward achieved by all methods are shown in Table 3. Noted that results for CUCB are the mean across 10 replicates, with each replicate differs from the initialization step. As presented in Table 3, CUCB consistently attains the highest reward across different values of K for both real and real+synthetic datasets. Furthermore, the inclusion of synthetic data leads to higher reward values across all methods, further reinforcing the advantages of synthetic data augmentation, as discussed in Section 4.3.2.

4.5. Ablation Studies

4.5.1. Alternative diversity metrics for performance evaluation

We investigate the impact of different diversity metrics on the output scenarios selected by CUCB. Specifically, we evaluate the Euclidean distance and Pearson correlation

Algorithm	Reward (K=5)	Reward(K=10)	Reward (K=20)
Proposed CUCB (Real)	7.18 (0.14)	7.43 (0.08)	6.86 (0.09)
Proposed CUCB (Real + 1k Synthetic)	**7.80 (0.21)**	**7.85(0.24)**	**7.82(0.13)**
Greedy with clustering (Real)	4.90 (1.96)	6.13 (1.49)	5.75 (0.75)
Greedy with clustering (Real + 1k Synthetic)	3.96 (0.58)	4.89 (0.57)	5.15 (0.38)
Greedy without clustering (Real)	0.02	0.03	0.04
Greedy without clustering (Real + 1k Synthetic)	1.41	1.99	2.63

Table 3: Best rewards achieved by different algorithms on the Real and Real + 1k Synthetic disturbance data set. Parenthetic numbers represents the standard deviation across 10 replicates.

Algorithm 2 Greedy approach without clustering information

Input: Number of disturbance scenario $\mathbb{N}$, number of representative disturbance scenarios K, disturbance data set $W_{\mathbb{N},T}$
for $k = \{0, \dots, K-1\}$ **do**
  Perform simulation for all disturbance scenarios in $W_{\mathbb{N},T}$ to obtain output trajectory set $Y_{\mathbb{N},T} \triangleq \left\{y_{0:T}^{(n)}\right\}_{n=1}^{\mathbb{N}}$
  for $i = 1, \dots, \mathbb{N}$ **do**
    Calculate the mean of pairwise DTW distance $\overline{DTW}_i$ between $y_{0:T}^i$ and all $y_{0:T}^j$ such that $i \neq j \in \mathbb{N}$
  end for
  Select $w_{0:T}^i$ such that the corresponding $y_{0:T}^i$ has the largest $\overline{DTW}_i$
  $W_{\mathbb{N},T} \setminus w_{0:T}^i$, $\mathbb{N} = \mathbb{N} - 1$
end for

Algorithm 3 Greedy approach with clustering information

Input: Number of clusters m, number of representative cluster K, admissible disturbance scenario space $\mathbb{W}_{K,T}$
while $k \leq K$ **do**
  Randomly select a disturbance scenario from each remaining cluster and perform simulation to get outcome trajectories set $Y_{m,T} \triangleq \left\{y_{0:T}^{(k)}\right\}_{k=1}^{m}$
  for $i = 1, \dots, m$ **do**
    Calculate the mean of pairwise DTW distance $\overline{DTW}_i$ between $y_{0:T}^i$ and all $y_{0:T}^j$ such that $i \neq j \in m$
  end for
  Select cluster i such that the corresponding output $y_{0:T}^i$ has the largest $\overline{DTW}_i$
  $k = k + 1$
  Remove cluster i, $m = m - 1$
end while

metrics as alternatives, which are two of the most common approaches for measuring time series similarity; c.f. [47]. To ensure a fair comparison, we fix the random seed across all steps, including the inherent randomness in the CUCB algorithm, for each diversity metric. The experiment is conducted on the real disturbance dataset with $K = 5$ and 500 CUCB iterations, under a noiseless setting.

As shown in Figure 8, different diversity metrics lead to different representative disturbance scenarios and corresponding output trajectories recommended by the CUCB algorithm. In general, both the DTW and Euclidean distance metrics produce diverse output scenarios that span a broad range of signal shapes and values. In contrast, while the Pearson correlation metric also yields diverse scenarios, the resulting value range is narrower compared to DTW and Euclidean distance. In particular, it fails to identify heat load scenarios as diverse as the others, and the peak solar irradiances, which result in the peak P_{buy} and P_{pv} being considerably higher for DTW and Euclidean approaches. This can be explained by realizing that the Pearson correlation is proportional to a normalized Euclidean distance [47], which means it captures differences in the shape of time series data rather than differences in magnitude.

This ablation study highlights the importance of selecting an appropriate diversity metric for the CUCB algorithm. The DTW distance metric is particularly effective in capturing both the shape and magnitude of time series data, making it a suitable choice for this application.

	Random	Closest-to-Center
Reward (K=5)	7.18 (0.14)	7.26 (0.10)
Reward (K=10)	7.43 (0.08)	7.64 (0.05)
Reward (K=20)	6.86 (0.09)	7.14 (0.13)

Table 4: Achieved reward for the CUCB algorithm using random and closest-to-center disturbance scenario selection. Format: mean (standard deviation) across 10 replicates.

4.5.2. Effect of random vs. deterministic intra-cluster scenario selection

If clustering is performed, and a cluster center is used as a proxy for an arm, then if that arm is pulled, there are two ways to ensure the selection of a realistic scenario from the cluster: (i) randomly select a scenario from the cluster, or (ii) select the scenario closest to the cluster center. The first approach is stochastic, while the second is deterministic. Here, we compare these two approaches in terms of their performance in CUCB. Note that choosing the actual cluster center as the representative scenario is not always a viable option, as the clustering method does not guarantee that the center is a valid scenario.

We compare the CUCB performance using both selection approaches on the real disturbance dataset. Rewards are averaged over 10 replicates, with the number of CUCB

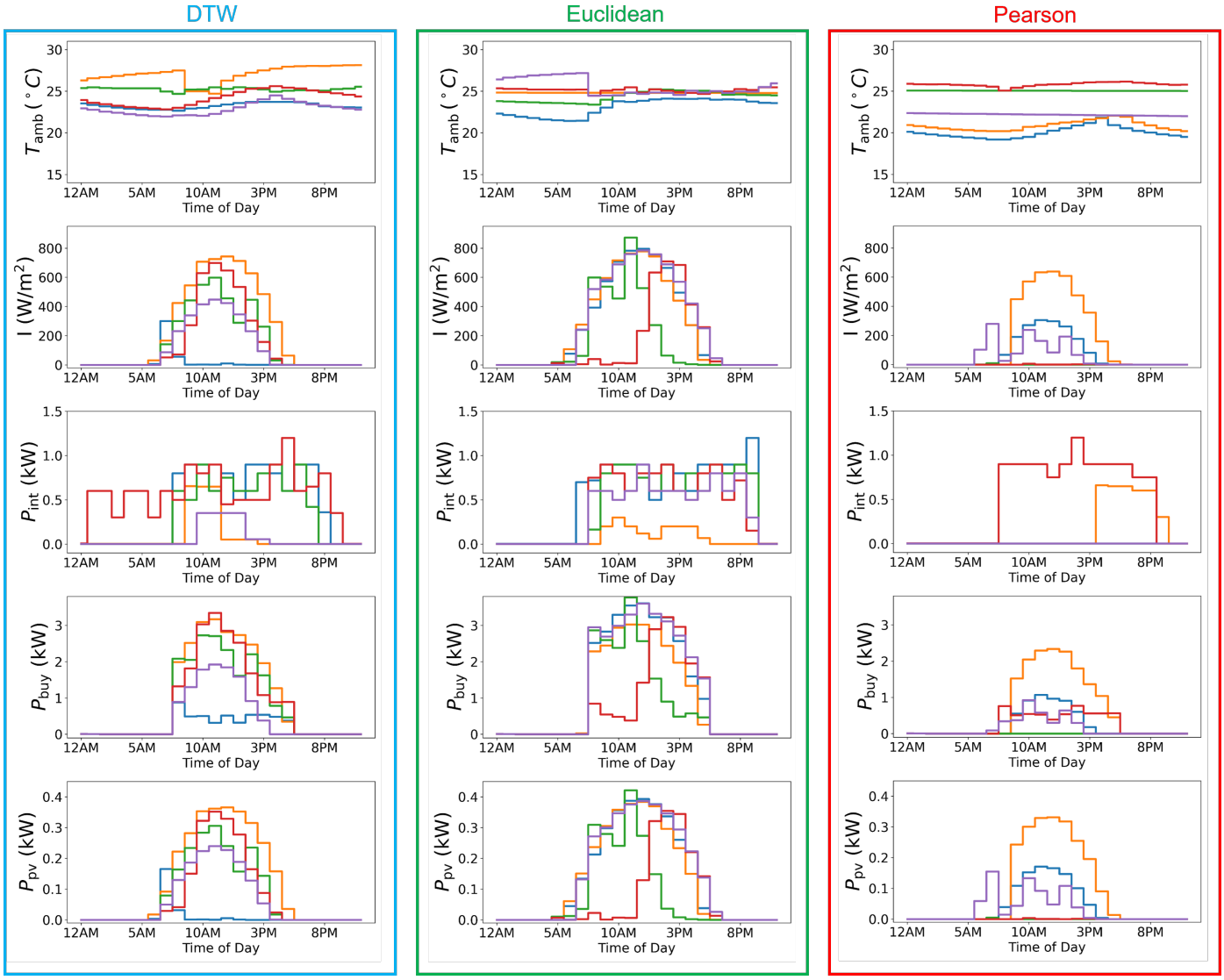

Figure 8: Illustrating the effect of diversity measures on CUCB performance. (Left) DTW distance, (Mid) Euclidean distance, and (Right) Pearson correlation. The top 3 rows are the disturbance scenarios and the bottom 2 rows are the simulation output trajectories.

iterations set to 500. As shown in Table 4, the deterministic approach yields consistently higher rewards for varying K. However, the relatively small difference in performance suggests that the random method is a robust and effective choice. Although the closest-to-center selection approach can achieve slightly higher rewards than the random approach, it has the drawback of repeatedly allocating simulation budget to the same disturbance scenarios because the same arm would be repeatedly selected during the CUCB algorithm with high chances. This limits the number of distinct scenarios explored within a fixed simulation budget. We also visually compare the final set of scenarios obtained by each approach in Figure 9. Although the difference in reward is relatively small, the resulting scenario sets differ significantly in composition, as expected. Both methods are capable of discovering diverse scenarios, but the random approach tends to generate output scenarios that are more distributed across the signal range, while the closest-to-center approach is more likely to identify extreme cases—such as the flat P_{buy} profiles highlighted in green and red in the right column. An advantage of the closest-to-center approach is that is repeatable.

4.5.3. Effect of varying number of representative scenarios

To investigate the effect of the number of representative scenarios K on the performance of the CUCB algorithm, we conduct additional experiments on the real disturbance data set. Following the same experimental setup as in Table 3, we cluster the disturbances into 50 groups and repeat each experiment 10 times, running the CUCB algorithm for 500 iterations per trial. As illustrated in Figure 10, the achieved reward initially increases with K, reaching a peak at $K = 7$, after which it begins to decline. This trend is explained as follows: when K is too small, CUCB may fail to capture the diversity of representative disturbance

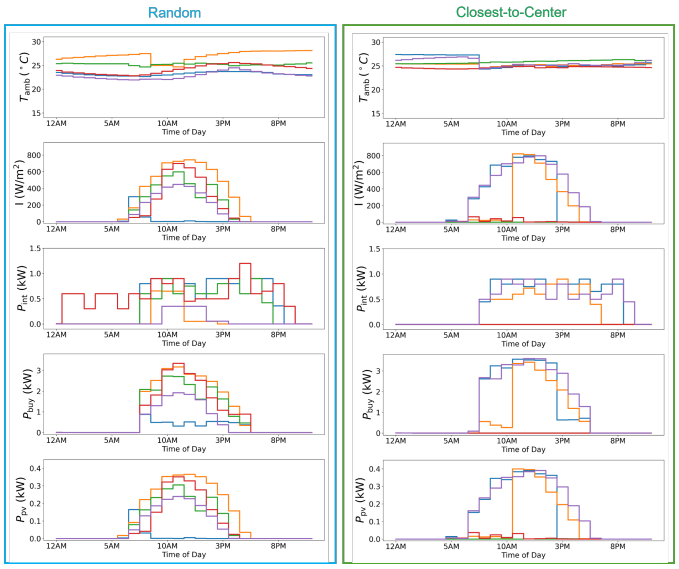

Figure 9: Scenarios obtained when selecting randomly within a cluster vs. the closest-to-center scenario. The top 3 rows are the disturbance scenarios and the bottom 2 rows are the simulation output trajectories.

scenarios, leading to suboptimal selections. Conversely, when K is too large, the algorithm tends to select scenarios that are similar to those already in the representative set. This redundancy reduces the achieved reward, as it is defined by the *average* pairwise DTW distance between all selected outcomes. In other words, an excessively large K may lead CUCB to choose overlapping scenarios, thereby diminishing the overall diversity and reward.

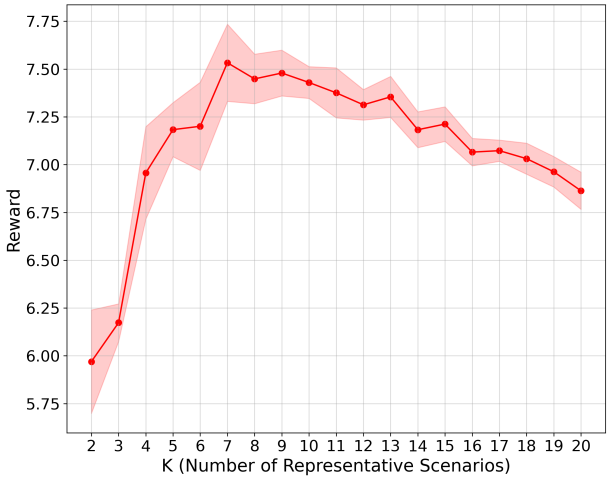

Figure 10: Impact of different K values on the achieved reward for the CUCB algorithm. Result shows the mean and one standard deviation across 10 replicates.

4.5.4. Simulation with additive Gaussian noise

In the experiments presented above, we considered only the noiseless setting for the building simulation model. To evaluate the robustness of the CUCB algorithm and greedy approaches under noisy conditions, we introduce i.i.d. Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma)$ to the performance outputs from the simulation model and report the effect of varying the noise level σ. As shown in Table 5, the performance of the CUCB algorithm remains comparable to the noiseless case for $\sigma = 0.1$ kW and 0.5 kW. Even when the noise level increases to $\sigma = 1$ kW, CUCB exhibits only a slight performance degradation and remains competitive. These results suggest that the CUCB algorithm is robust to moderate levels of noise in the simulation data. In contrast, the performance of the greedy approach deteriorates significantly as noise increases, indicating greater sensitivity to perturbations in the output.

σ (kW)	CUCB	Greedy+Cluster
0 (noiseless)	7.43 (0.08)	6.13 (1.49)
0.1	7.48 (0.09)	4.66 (1.55)
0.5	7.25 (0.14)	3.23 (0.92)
1	6.99 (0.27)	3.18 (0.76)

Table 5: Effect of varying noise levels on performance outputs. Format: mean (standard deviation) across 10 replicates.

4.6. Time complexity for baseline algorithms

Beyond reward optimization, computational efficiency is a crucial factor when evaluating different algorithms due to the computationally expensive nature of general building simulation models. Our time complexity analysis assumes a constant simulation time per bandit round, independent of the disturbance values. The time complexity of CUCB depends on the number of arms m, the number of rounds N_{CUCB}, and the number of representative clusters K. The initialization step requires $m \cdot K$ simulations, as indicated in lines 8–12 of Algorithm 1. The iterative step, beginning at line 13, executes K simulations per round, resulting in a total of $N_{\text{CUCB}} \cdot K$ simulations. Consequently, the time complexity of CUCB is $O((m + N_{\text{CUCB}}) \cdot K)$. Notably, N_{CUCB} is a hyperparameter that can be pre-specified by the user.

For the greedy approach without clustering, time complexity is determined by the total number of disturbance scenarios $\mathbb{N}$. Each iteration performs $\mathbb{N} - K$ simulations, leading to a time complexity of $O(\mathbb{N} \cdot K)$. In contrast, the greedy approach with clustering executes a total of $\sum_{k=0}^{K-1}(m-k)$ simulations, resulting in a time complexity of $O(m \cdot K)$. In the general case where $\mathbb{N} \gg m$, $\mathbb{N} \gg N_{\text{CUCB}}$, and $\mathbb{N} \gg K$, the greedy approach without clustering has the highest time complexity. By selecting $N_{\text{CUCB}} \ll m$, CUCB achieves the same time complexity as the greedy approach with clustering information. Given CUCB's superior empirical performance in terms of achieved rewards and its competitive time complexity compared to greedy approaches, we conclude that CUCB is the most suitable algorithm for solving the optimization problem in Eq. 3.

4.7. Controller redesign with representative scenarios

We conducted experiments to demonstrate that our CUCB-based scenario selection method identifies critical operating conditions that conventional approaches miss. These experiments show how the selected scenarios can guide practical improvements to HVAC controller design.

We compared three scenario selection methods using the MPC controller described in (22): random search (RS), greedy selection, and our proposed CUCB algorithm. Each method selected $K = 5$ representative disturbance scenarios from real building data. We tested whether the controller could maintain room temperatures between 22°C and 24°C during work hours, while managing energy consumption. The experiment had two phases. First, we tested the existing or 'legacy' MPC controller with scenarios from each selection method. Second, we redesigned the MPC (i.e. heat pump and battery storage) based on constraint violations revealed by the CUCB scenarios and retested performance.

Figure 11 shows room temperature trajectories and HVAC control inputs for each scenario selection method. Subplots [A] and [B] show that random search and greedy selection produce scenarios where the controller maintains comfortable temperatures throughout the day. All temperature trajectories stay within the 22-24°C bounds with stable performance. Subplot [C] presents a different result: the CUCB algorithm successfully identified scenarios that caused the controller to violate temperature constraints. Multiple scenarios push room temperatures above 24°C during afternoon hours (noon-3 PM). These are not caused by unrealistic scenarios: they represent legitimate combinations of occupancy, solar gain, and ambient conditions that the baseline RS and even Greedy failed to find. In fact, the green lines are historical data points that have been recorded in the building. The red dashed lines are induced by scenarios that were synthetically generated, although we confirm they were realistic also, due to post-processing to ensure physically meaningful scenarios were retained. Out of $K = 5$ scenarios, only one real scenario violates constraints, whereas two synthetic scenarios do. This highlights the CUCB algorithm's ability to discover challenging conditions that other methods miss and the utility of synthesizing realistic scenarios with generative networks. Subplot [E] shows the MPC cannot handle these representative scenarios. The heat pump inputs saturate during constraint violations, indicating insufficient system capacity.

We used insights from the CUCB scenarios to redesign the HVAC system. We increased heat pump capacity by 15% and doubled the battery capacity to enforce the MPC specifications. Subplots (D) and (F) confirm the effectiveness of the redesign. The new closed-loop system maintains all temperatures within comfort bounds, even for previously problematic scenarios. Temperature regulation improves during critical afternoon periods. As can be seen in subplot [F] one of the synthetic scenarios requires the entire heat pump capacity to maintain temperature within specified ranges.

5. Conclusions

This work presented a combinatorial MAB framework that selects an informative set of disturbance scenarios for assessing the closed-loop behaviour of GEBs. By treating scenario selection as a diversity-maximisation problem on measured outputs—specifically the photovoltaic power P_{pv} and the net grid exchange P_{buy}—and by using dynamic-time-warping distances as the bandit reward, the proposed CUCB algorithm condensed six hundred real disturbance days into only five representative cases while retaining the extreme values and temporal patterns needed to assess the performance of the controller. The approach enjoys a sub-linear regret bound and a computational cost that scales linearly with the number of arms and bandit rounds, making it practical for large data sets and expensive simulations.

Applied to a stochastic MPC-controlled single-zone building, the CUCB-selected scenarios reproduced the full range of grid import, export and bidirectional flow observed in the original data, whereas conventional greedy heuristics captured at most two-thirds of that range. When historical data were scarce, a diffusion-based time-series generator produced synthetic disturbances that enlarged the latent disturbance manifold and increased the diversity reward by up to a quarter. These artificial trajectories introduced operating conditions that were absent from the measurements yet crucial for uncovering vulnerabilities in the closed-loop.

This paper takes a step to indicate that bandit-driven scenario selection offers a rigorous, data-centred alternative to manual or worst-case testing and can dramatically reduce the number of simulations required for performance certification. Because the method is agnostic to both the choice of outputs and the underlying control strategy, it can be transferred to multiple zones, and even smart city settings. Synthetic augmentation further extends its applicability to newly commissioned buildings where available data is small.

Future work. The paper considered a single-zone model and deterministic forecasts; this can be extended to multi-zone or district-scale simulations and probabilistic forecasting in the flavor of [48]. In fact, the proposed scenario selection method scales naturally to multi-zone systems, albeit with added computational cost. The algorithm remains unchanged, with multivariate outputs handled via a weighted DTW-based reward, and zone coupling captured implicitly through the black-box simulator. Additional work is also needed to benchmark alternative generative models, to explore adaptive bandit policies for online scenario discovery. Together, these steps will help establish bandit-driven scenario generation as a standard tool for designing and validating GEB energy management systems.

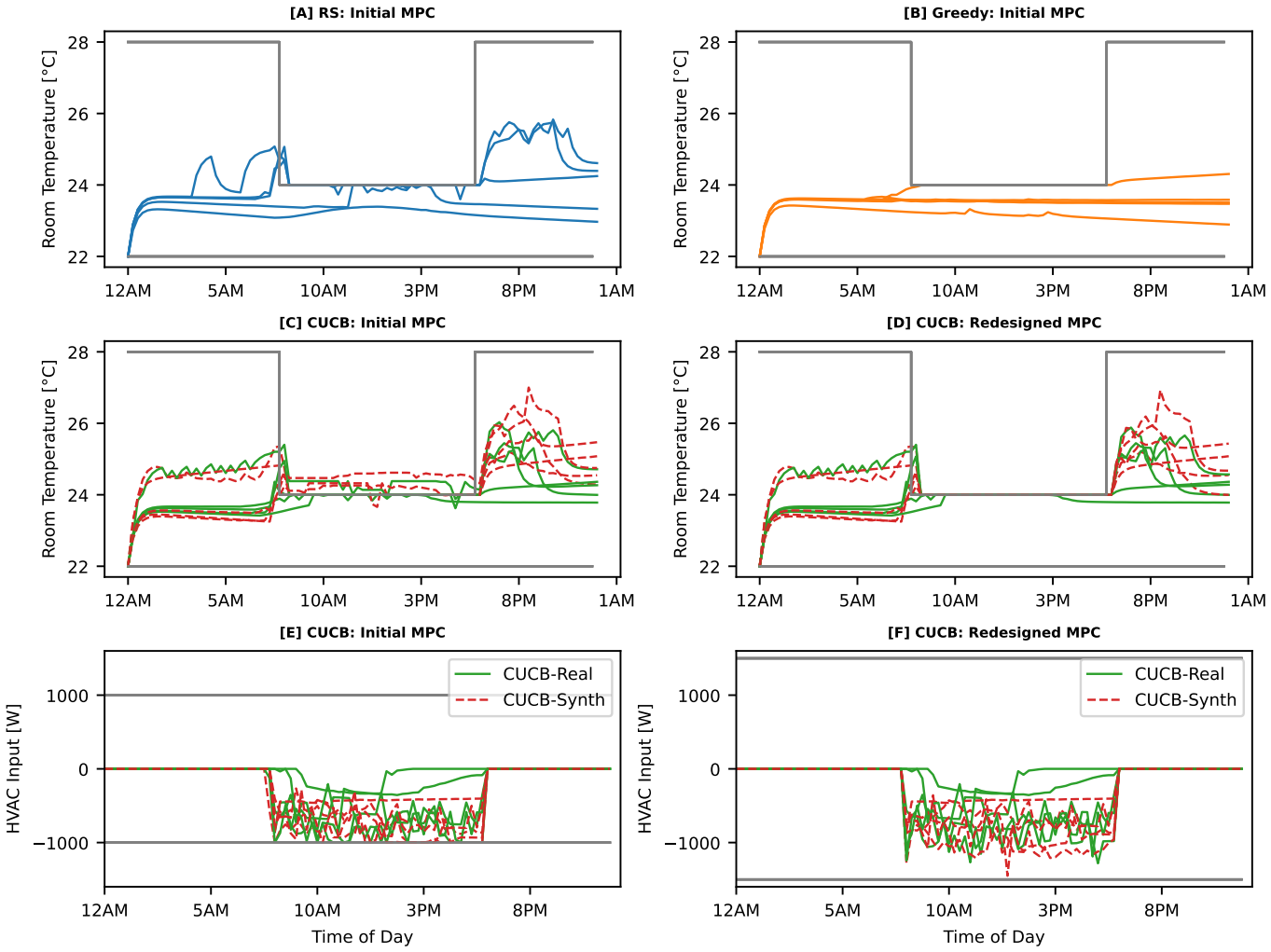

Figure 11: Room temperature profiles from the initial closed-loop MPC are shown for the randomly selected scenarios (top left), the greedy-selected scenarios (top right), and the CUCB-selected scenarios (middle left). The middle right figure shows the room temperature profiles for the CUCB-selected scenarios under the redesigned closed-loop MPC. The bottom left and bottom right figures illustrate the initial and redesigned controller actions, respectively, and the resized capacity of the HVAC is shown from 1kW to 1.5kW.

References

[1] S. woo Ham, D. Kim, T. Barham, K. Ramseyer, The first field application of a low-cost mpc for grid-interactive k-12 schools: Lessons-learned and savings assessment, Energy and Buildings 296 (2023) 113351.

[2] Z. Wei, J. K. Calautit, Field experiment testing of a low-cost model predictive controller (mpc) for building heating systems and analysis of phase change material (pcm) integration, Applied Energy 360 (2024) 122750.

[3] D. Wang, Y. Chen, W. Wang, C. Gao, Z. Wang, Field test of model predictive control in residential buildings for utility cost savings, Energy and Buildings 288 (2023) 113026.

[4] J. Rawlings, D. Mayne, Postface to model predictive control: Theory and design, Nob Hill Pub 5 (2012) 155–158.

[5] T. Pippia, J. Lago, R. De Coninck, B. De Schutter, Scenario-based nonlinear model predictive control for building heating systems, Energy and Buildings 247 (2021) 111108.

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

[7] P. Mohebi, S. Li, Z. Wang, Chance-constrained stochastic framework for building thermal control under forecast uncertainties, Energy and Buildings (2025) 115385.

[8] Z. Hu, Y. Gao, L. Sun, M. Mae, T. Imaizumi, Improved robust model predictive control for residential building air conditioning and photovoltaic power generation with battery energy storage system under weather forecast uncertainty, Applied Energy 371 (2024) 123652.

[9] Y. Gao, S. Miyata, Y. Akashi, Energy saving and indoor temperature control for an office building using tube-based robust model predictive control, Applied Energy 341 (2023) 121106.

[10] J. Hou, H. Li, N. Nord, G. Huang, Model predictive control under weather forecast uncertainty for hvac systems in university buildings, Energy and Buildings 257 (2022) 111793.

[11] A. Doma, M. M. Ouf, F. Amara, N. Morovat, A. K. Athienitis, Occupancy-informed predictive control strategies for enhancing the energy flexibility of grid-interactive buildings, Energy and Buildings 332 (2025) 115388.

[12] F. Langner, W. Wang, M. Frahm, V. Hagenmeyer, Model predictive control of distributed energy resources in residential buildings considering forecast uncertainties, Energy and Buildings 303 (2024) 113753.

[13] A. Mesbah, Stochastic model predictive control: An overview and perspectives for future research, IEEE Control Systems

Magazine 36 (6) (2016) 30–44.
[14] A. Chakrabarty, L. Vanfretti, W.-T. Tang, J. A. Paulson, S. Zhan, S. A. Bortoff, V. Deshpande, Y. Wang, C. R. Laughman, Assessing building control performance using physics-based simulation models and deep generative networks, in: 2024 IEEE Conference on Control Technology and Applications (CCTA), IEEE, 2024, pp. 547–554.
[15] W. Neiswanger, K. A. Wang, S. Ermon, Bayesian algorithm execution: Estimating computable properties of black-box functions using mutual information, in: International Conference on Machine Learning, PMLR, 2021, pp. 8005–8015.
[16] A. Salatiello, Y. Wang, G. Wichern, T. Koike-Akino, Y. Ohta, Y. Kaneko, C. Laughman, A. Chakrabarty, Synthesizing building operation data with generative models: Vaes, gans, or something in between?, in: Companion Proceedings of the 14th ACM International Conference on Future Energy Systems, 2023, pp. 125–133.
[17] T. Lattimore, C. Szepesvári, Bandit Algorithms, Cambridge University Press, 2020.
[18] A. Slivkins, et al., Introduction to multi-armed bandits, Foundations and Trends® in Machine Learning 12 (1-2) (2019) 1–286.
[19] W. Chen, Y. Wang, Y. Yuan, Combinatorial multi-armed bandit: General framework and applications, in: International conference on machine learning, PMLR, 2013, pp. 151–159.
[20] J. A. Paulson, F. Sorouifar, C. R. Laughman, A. Chakrabarty, Lsr-bo: Local search region constrained bayesian optimization for performance optimization of vapor compression systems, in: 2023 American Control Conference (ACC), IEEE, 2023, pp. 576–582.
[21] S. Chen, S. A. Billings, P. Grant, Non-linear system identification using neural networks, International journal of control 51 (6) (1990) 1191–1214.
[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Advances in neural information processing systems 27 (2014).
[23] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, L. Carin, Variational autoencoder for deep learning of images, labels and captions, Advances in neural information processing systems 29 (2016).
[24] M. Razghandi, H. Zhou, M. Erol-Kantarci, D. Turgut, Variational autoencoder generative adversarial network for synthetic data generation in smart home, in: ICC 2022-IEEE International Conference on Communications, IEEE, 2022, pp. 4781–4786.
[25] F. Khayatian, Z. Nagy, A. Bollinger, Using generative adversarial networks to evaluate robustness of reinforcement learning agents against uncertainties, Energy and Buildings 251 (2021) 111334.
[26] Y. Gu, Q. Chen, K. Liu, L. Xie, C. Kang, Gan-based model for residential load generation considering typical consumption patterns, in: 2019 IEEE power & energy society innovative smart grid technologies conference (ISGT), IEEE, 2019, pp. 1–5.
[27] C. Fan, M. Chen, R. Tang, J. Wang, A novel deep generative modeling-based data augmentation strategy for improving short-term building energy predictions, in: Building Simulation, Vol. 15, Springer, 2022, pp. 197–211.
[28] Y. Li, B. Dong, Y. Qiu, Conditional generative adversarial network (cgan) for generating building load profiles with photovoltaics and electric vehicles, Energy and Buildings (2025) 115584.
[29] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, Improved techniques for training gans, Advances in neural information processing systems 29 (2016).
[30] Y. Yacoby, W. Pan, F. Doshi-Velez, Failure modes of variational autoencoders and their effects on downstream tasks, arXiv preprint arXiv:2007.07124 (2020).
[31] X. Yuan, Y. Qiao, Diffusion-ts: Interpretable diffusion for general time series generation, arXiv preprint arXiv:2403.01742 (2024).
[32] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, B. Poole, Score-based generative modeling through stochastic differential equations, arXiv preprint arXiv:2011.13456 (2020).
[33] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, Advances in neural information processing systems 33 (2020) 6840–6851.
[34] J. M. L. Alcaraz, N. Strodthoff, Diffusion-based time series imputation and forecasting with structured state space models, arXiv preprint arXiv:2208.09399 (2022).
[35] M. Biloš, K. Rasul, A. Schneider, Y. Nevmyvaka, S. Günnemann, Modeling temporal data as continuous functions with stochastic process diffusion, in: International Conference on Machine Learning, PMLR, 2023, pp. 2452–2470.
[36] L. Lin, Z. Li, R. Li, X. Li, J. Gao, Diffusion models for time-series applications: a survey, Frontiers of Information Technology & Electronic Engineering 25 (1) (2024) 19–41.
[37] J. Li, R. Ma, M. Deng, X. Cao, X. Wang, X. Wang, A comparative study of clustering algorithms for intermittent heating demand considering time series, Applied Energy 353 (2024) 122046.
[38] W. Chen, Y. Wang, Y. Yuan, Q. Wang, Combinatorial multi-armed bandit and its extension to probabilistically triggered arms, Journal of Machine Learning Research 17 (50) (2016) 1–33.
[39] P. Auer, Finite-time analysis of the multiarmed bandit problem (2002).
[40] C. F. Higham, D. J. Higham, P. Grindrod, Diffusion models for generative artificial intelligence: An introduction for applied mathematicians, arXiv preprint arXiv:2312.14977 (2023).
[41] K. R. Rao, D. N. Kim, J. J. Hwang, Fast Fourier transform-algorithms and applications, Springer Science & Business Media, 2011.
[42] R. Tavenard, J. Faouzi, G. Vandewiele, F. Divo, G. Androz, C. Holtz, M. Payne, R. Yurchak, M. Rußwurm, K. Kolar, E. Woods, Tslearn, a machine learning toolkit for time series data, Journal of Machine Learning Research 21 (118) (2020) 1–6. URL http://jmlr.org/papers/v21/20-091.html
[43] B. Cui, C. Fan, J. Munk, N. Mao, F. Xiao, J. Dong, T. Kuruganti, A hybrid building thermal modeling approach for predicting temperatures in typical, detached, two-story houses, Applied Energy 236 (2019) 101–116.
[44] F. Langner, M. Frahm, W. Wang, J. Matthes, V. Hagenmeyer, Hierarchical-stochastic model predictive control for a grid-interactive multi-zone residential building with distributed energy resources, Journal of Building Engineering 89 (2024) 109401.
[45] S. Diamond, S. Boyd, CVXPY: A Python-embedded modeling language for convex optimization, Journal of Machine Learning Research 17 (83) (2016) 1–5.
[46] A. Domahidi, E. Chu, S. Boyd, ECOS: An SOCP solver for embedded systems, in: 2013 European control conference (ECC), IEEE, 2013, pp. 3071–3076.
[47] M. R. Berthold, F. Höppner, On clustering time series using euclidean distance and pearson correlation, arXiv preprint arXiv:1601.02213 (2016).
[48] Y.-J. Park, F. Germain, J. Liu, Y. Wang, G. Wichern, T. Koike-Akino, N. Azizan, C. R. Laughman, A. Chakrabarty, Probabilistic forecasting for building energy systems: Are time-series foundation models the answer?, in: NeurIPS Workshop on Time Series in the Age of Large Models, 2024.

Nomenclature

$\mathbb{W}$	Disturbance space
$\mathcal{I}$	Scoring function
$\mathcal{M}$	Black-box model
$\mathcal{S}$	Super arm set
r	Reward
S	Super arm
T	Length of time for time-series data
t	Time index
T_i	Number of times arm i been played
W	Disturbance
BES	Building energy system
BESS	Battery energy storage system
CPU	Central processing unit
CUCB	Combinatorial upper confidence bound
DDPM	Denoising diffusion probabilistic model
FFT	Fast Fourier transform
GEB	Grid-interactive efficient building
GPU	Graphics processing unit
HP	Heat pump
MAB	Multi-arm bandit
MAE	Mean absolute error
MASE	Mean absolute scaled error
MBRL	Model-based reinforcement learning
MPC	Model predictive control
PCA	Principal component analysis
PV	Photovoltaic
TSGN	Time-series generative network
UCB	Upper confidence bound
UMAP	Uniform manifold approximation & projection