ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240817

A Meta-Learning Approach for Multi-Objective Reinforcement Learning in Sustainable Home Energy Management

Junlin Lu [a,*], **Patrick Mannion** [a] **and Karl Mason**[a]

[a]University of Galway
ORCID (Junlin Lu): https://orcid.org/0000-0002-6014-9419, ORCID (Patrick Mannion): https://orcid.org/0000-0002-7951-878X, ORCID (Karl Mason): https://orcid.org/0000-0002-8966-9100

Abstract. Effective residential appliance scheduling is crucial for sustainable living. While multi-objective reinforcement learning (MORL) has proven effective in balancing user preferences in appliance scheduling, traditional MORL struggles with limited data in non-stationary residential settings characterized by renewable generation variations. Significant context shifts in the environment can invalidate previously learned policies. To address this, we extend state-of-the-art MORL algorithms with the meta-learning paradigm, enabling rapid, few-shot adaptation to shifting contexts. Additionally, we employ an auto-encoder (AE)-based unsupervised method to detect shifts in environmental context. We have also developed a residential energy environment to evaluate our method using real-world data from London residential settings. This study not only assesses the application of MORL in residential appliance scheduling but also underscores the effectiveness of meta-learning in energy management. Our top-performing method significantly surpasses the best baseline, while the trained model saves 3.28% on electricity bills, a 2.74% increase in user comfort, and a 5.9% improvement in expected utility. Additionally, it reduces the sparsity of solutions by 62.44%. Remarkably, these gains were accomplished using 96.71% less training data and 61.1% fewer training steps.

1 Introduction

Reducing greenhouse gas emissions has become an essential concern. Electricity production is responsible for more than a quarter of these emissions [12] in 2018, where the residential sector plays a considerable role. It accounted for approximately 26.8% of global electricity consumption in 2022 [13]. Although renewable energy sources can mitigate this challenge, their intermittent and variable nature limits their utilization [21]. Effective energy management techniques have the potential to overcome this. One of the candidate techniques is reinforcement learning [7, 32]. While the single-objective reinforcement learning (SORL) method is widely used in residential energy management, practical scenarios often requires users to make trade-offs between multiple objectives, e.g. saving costs and increasing comfort [21]. It is therefore more reasonable to render the problem as multi-objective reinforcement learning (MORL) [18, 14, 3]. Given that MORL effectively addresses practical scenarios in residential energy management, it is crucial to continually validate and refine the latest MORL methods in this field. We look into the residential appliance scheduling in this work specifically which is one of the most important parts of energy management.

Alegre et al. recently proposed two MORL algorithms, i.e. *genearalized policy improvement linear support (GPI-LS)*, i.e. the model-free version, and *genearalized policy improvement prioritized dyna (GPI-PD)*, i.e. the model-based version [3]. The agent guarantees rapid training by identifying the corner weights [34] along which the entire policy set can achieve the largest improvement. However, they were only evaluated in simulated environments in the original work. It is challenging for them to work well with the non-stationary environment of appliance scheduling, which arises from intermittent renewable energy production predominantly. In fact, our experiments revealed that *GPI-LS/PD* policies failed to surpass even a basic rule-based policy unless a whole year dataset is provided.

One intuitive approach to make the policy work well in a non-stationary environment is finetuning the policy in response to significant context shifts. However, identifying context shifts is challenging due to the lack of explicit labels for these qualitative changes. Moreover, our experimental results show that simply finetuning a trained policy with new data fails to effectively enhance performance. Another approach is to train the policy using more comprehensive datasets, such as data spanning an entire year; however, this method incurs significant computational overhead. The non-stationary nature of the environment presents two critical challenges: **(i)** How can the policy be both effective and easy to finetune with minimal data? **(ii)** How can environmental context shifts be identified in an unsupervised manner?

Meta-learning [9, 30] is especially effective in managing scenarios that involve varying environmental contexts and identifying an initial set of parameters that can be rapidly and effectively finetuned to adapt to new contexts. To address the first challenge, we have enhanced the *GPI* algorithms with the meta-learning method, i.e. *Reptile* [30], to facilitate few-shot finetuning. To tackle the second challenge, we employ an auto-encoder (AE) model as an unsupervised method to detect significant qualitative shifts in context [4, 45, 46]. This method allows for the identification of contextual shifts in the environment.

The contributions of this research are as follows:

1. We have demonstrated that residential energy management is inherently compatible with meta-learning by integrating *GPI* algo-

* Corresponding Author. Email: J.Lu5@universityofgalway.ie

rithms into this framework. Our top-performing method significantly surpasses the best baseline, while the trained model saves 3.28% on electricity bills, a 2.74% increase in user comfort, and a 5.9% improvement in expected utility. Additionally, it reduces the sparsity of solutions by 62.44%. Remarkably, these gains were accomplished using 96.71% less training data and 61.1% fewer training steps.

2. This study is the first application and evaluation of *GPI-LS/PD* algorithms in residential appliance scheduling and also the first use of a model-based MORL in this field. We have conducted a detailed discussion about model-based MORL in a non-stationary environment.

3. We have developed and released an open-source model for residential energy environments tailored to MORL, which conforms to the standards set by the OpenAI Gym.

2 Related Work

2.1 Multi-objective Reinforcement Learning

MORL methods are designed for multi-objective sequential decision-making problems. The MORL agent is trained to make trade-offs among multiple objectives. The return from the interaction between the agent and the environment in MORL is a vector rather than a scalar as in SORL. Nevertheless, as the return vector is usually scalarized as a utility by calculating the inner product with the preference weight vector [11], one intuitive way to sort a MORL problem is to separate a MORL policy training from the training of multiple SORL policies based on different preference weight vectors [26, 37, 38, 35, 28]. This method, however, is computationally expensive when there are many candidate preference weights, or even intractable when the preference is not given.

This limitation is mitigated by the MORL paradigm conditioned on the preference weight. Usually, a conditioned MORL model generates actions based on the given preference weight [1, 43, 15]. They do not need to know the user's specific preference but a sample from the preference weight space during training. Yang et al. proposed Envelop Q learning[43] that adopts vectorized update. At each training step, it samples preference weights randomly. However, rather than randomly sampling preference for training, enhancements were conducted on this algorithm where the weight that can achieve the largest improvement is sampled. Basaklar et al. [5] proposed a method to efficiently update the model based on the angle between the Q-values and the weight vectors. Alegre et al. [3] leveraged *GPI* to find an update direction with the largest potential to propel the improvement of the whole policy set.

2.2 Multi-objective Reinforcement Learning for Energy Management

MORL has been widely used in energy management, e.g. micro-grid control [42, 20], water heating system oversight [33], energy control in vehicles [47, 41], and the management of residential energy systems [21, 22, 23]. In assessing multi-objective energy management tasks, linear scalarization is usually employed to determine the aggregated return from various objectives [11]. The weights reflecting the potential preferences of users are on a simplex. As the number of objectives increases, the complexity of the problem escalates exponentially. A noteworthy consideration is that a finite number of weight factor combinations are available [34]. This has prompted the application of fuzzy logic techniques to streamline the solution set in MORL for energy management [8, 31, 36, 44]. Liu et al. [20] introduced a policy-based, model-free MORL algorithm that employs the Borg MOEA approach [10] for policy improvement. Despite these advancements, a value-based evolutionary MORL algorithm remains a gap. Although an actor-critic model of evolutionary reinforcement learning is proposed [17], it is tailored for single-objective scenarios and has not been adapted for MORL contexts. Moreover, the model-based MORL algorithm [3] is not evaluated within residential energy management yet.

Wu et al. have used a prioritized dueling double DQN to formulate a multi-objective energy management system, integrating multiple objectives within a singular value function to optimize cumulative rewards [41]. This strategy, similar to the approaches adopted by Riebel et al. [33] and Xu et al. [42], involves incorporating various objectives into one reward function. This potentially results in significant computational demands when the preference for the objectives changes. Cutting-edge MORL algorithms can dynamically adjust to changes in the weight combination of objectives [3], yet their practical evaluation in the context of residential energy management remains unexplored.

2.3 Meta Learning

Humans can get satisfactory performance on a new task with a few attempts if they have some related knowledge. Although AI players can reach the human levels in many cases, they always need more samples. Meta-learning was proposed to train a model to quickly adapt to new scenarios by using a small amount of data [9]. Vettoruzzo et al. conducted a comprehensive review of the meta-learning technologies. They provide a taxonomy of meta-learning methods, i.e. black-box meta-learning methods, optimization-based meta-learning methods, meta-learning via distance metric learning, and hybrid approaches [39].

In this work, we concentrate on optimization-based meta-learning methods. Consequently, while hybrid approaches, such as those discussed in [40], offer valuable insights and advancements by integrating various meta-learning strategies, they extend beyond the scope of our current analysis.

A foundational work in optimization-based meta-learning is *Model-Agnostic Meta-Learning (MAML)* [9], renowned for its broad applicability and impact. *MAML* learns initial model parameters which can fast adapt to a new task with only few-shot training. There is an inner loop and an outer loop in *MAML*. In the inner loop (fast adaptation), for each sampled task, the current model parameters are used to conduct one or several gradient descent steps. This is to obtain the task-optimized parameters. In the outer loop (meta-update), the performance across all tasks is evaluated. This is used to back-propagate through the model's initial parameters for update. With these two loops, a set of initial parameters that allows the model to quickly adapt to new tasks is found.

Nevertheless, due to the computational burden associated with the bi-level gradient optimization in *MAML*, algorithms like *Reptile* and *FOMAML* [30] have been developed to alleviate these issues. *Reptile* seeks to identify a set of initial parameters that are near the optimal parameters for individual tasks. In our research, we employ *Reptile* as the meta-learning methodology due to its computational efficiency and its compatibility with neural network architectures.

3 Preliminary

We model the multi-objective residential energy management problem as a *multi-objective Markov decision process* (MOMDP), i.e. $\mathcal{M} := (\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma, \mu, \boldsymbol{R})$ [11]. The spaces of states and actions are

denoted as $\mathcal{S}$ and $\mathcal{A}$; $\mathcal{T} : S \times A \times S \rightarrow [0, 1]$ is the probabilistic transition function; $\gamma \in [0, 1)$ is a discount factor; $\mu : S_0 \rightarrow [0, 1]$ is the initial state probability distribution; The function $\boldsymbol{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}^d$ is a vectorized reward function, focusing on two main objectives in this work: maximizing comfort and minimizing energy costs. We start with introducing some necessary concepts of MORL setting.

In MORL setting, the *value vector*, starting from an initial state distribution μ then following policy π is $\boldsymbol{v}^\pi := \mathbb{E}_{s_0 \sim \mu}[\boldsymbol{q}^\pi(s_0, \pi(s_0))]$. The i-th component of $\boldsymbol{v}^\pi$ represents the value returned for the i-th objective. With the value vector of policy, we can define the Pareto dominance relation ($\succ_p$): $\boldsymbol{v}^\pi \succ_p \boldsymbol{v}^{\pi'} \iff (\forall i : v_i^\pi \geq v_i^{\pi'}) \cap (\exists i : v_i^\pi > v_i^{\pi'})$. We say that $\boldsymbol{v}^\pi$ is non-dominated when at least one element of $\boldsymbol{v}^\pi$ is greater than all other $\boldsymbol{v}^{\pi'}$. The Pareto front (PF) is therefore defined as $\mathcal{F} := \{\boldsymbol{v}^\pi | \nexists \pi' \; s.t. \; \boldsymbol{v}^{\pi'} \succ_p \boldsymbol{v}^\pi\}$

In MORL, policy evaluation is dependent on the preference vector given. To involve different user-defined preferences over the objectives, a *utility function* is used [11] to scalarize the reward vector. Linear utility function is the most frequently used utility function. A linear weight vector $\boldsymbol{w}$ denoting the user preference of importance over each objective is given to scalarize $\boldsymbol{v}^\pi$: $u(\boldsymbol{v}^\pi, \boldsymbol{w}) = v_{\boldsymbol{w}}^\pi = \boldsymbol{v}^\pi \cdot \boldsymbol{w}$ where $\boldsymbol{w}$ from the simplex $\mathcal{W} : \sum_i^d w_i = 1, w_i \geq 0$. The convex coverage set, *CCS* represents a finite convex selection of the *Pareto front (PF)*[1]. Each point in the *CCS* is at least one optimal policy corresponding to a certain linear preference. *CCS* is formally defined as: $CCS := \{\boldsymbol{v}^\pi \in \mathcal{F} | \exists \boldsymbol{w} \; s.t. \; \forall \boldsymbol{v}^{\pi'} \in \mathcal{F}, \boldsymbol{v}^\pi \cdot \boldsymbol{w} \geq \boldsymbol{v}^{\pi'} \cdot \boldsymbol{w}\}$.

When using MORL in residential energy management, as the renewable energy source is introduced, different weather conditions over the year can cause very different generation backgrounds. Traditional approaches often struggle to cope with such environmental variability[2]. Meta-learning aims at fast adaptation to new tasks through training on a variety of tasks, which aligns seamlessly with our scenario. The next challenge is to recognize when the underlying environment changes. We use the unsupervised anomaly detection technique to distinguish different renewable generation contexts as distinct tasks and utilize a meta-learning approach to identify initial model parameters that are suitable for rapid finetuning during evaluation.

Our concern is to enhance the *GPI*-based approach with the meta-learning method to initialize the model with a set of parameters that performs generally well in all contexts and can be effectively fine-tuned through a few-shot manner.

4 R-GPI-LS/PD Algorithm

We first introduce the method we used to do context shift detection. Then we outline the meta-learning *GPI* algorithms.

4.1 Context Detection

In this work, the context-shifting points in residential energy management are defined as the points in time when changes in weather cause significant variations in the power output of renewable energy sources (substantial changes in the MDP). Those shifting points are not easily detectable, and there is also a lack of labeled data to support the supervised learning model. We leverage the unsupervised learning paradigm based on AE architecture [4, 45, 46].

The AE is designed to try to reconstruct the input data. Such a model, when trained on data from a specific context can gain good reconstruction ability. We reversely use its design mechanism to detect the context-shifting points. When data from a different context comes, the reconstruction loss will see a significant rise.

We first train the AE on an initial context data. Once the training converges, we proceed with the reconstruction of the subsequent data, while monitoring the AE's reconstruction loss. We average the loss values over a moving window. When the reconstruction loss for a specific context's data points exceeds the rolling average, it indicates that a significant shift in context has occurred[3]. At this point, we designate the current context as a new context and continue training the AE on this segment of context. This process is repeated until the detection is complete for the entire year's data. See Algorithm 1 for more detail.

Algorithm 1 Context Detection

1: **Input** Windowed annual dataset $\{win_1, ..., win_n\}$; Initial AE model; Threshold value $\delta = -\infty$; Reconstruction loss function $\mathcal{L}$; List of reconstruction loss $[\ ]_\mathcal{L} = [\ ]$; Last window of this context $win_c = win_1$; Context list $[\ ]_{context} = [\ ]$
2: **for** win_i in $\{win_1, ..., win_n\}$ **do**
3: **if** $\mathcal{L}(win_i, \text{AE}(win_i)) > \delta$ **then**
4: Add i to $[\ ]_{context}$
5: Empty the reconstruction loss list $[\ ]_\mathcal{L} = [\ ]$
6: Train AE with the data $\{win_c, ..., win_i\}$
7: **end if**
8: Add $\mathcal{L}(win_i, \text{AE}(win_i))$ to $[\ ]_\mathcal{L}$
9: Update the threshold $\delta \leftarrow mean([\ ]_\mathcal{L})$
10: **end for**
11: **Output** $[\ ]_{context}$

4.2 Reptile and GPI Algorithm

To make the model able to adjust to non-stationary environments while saving computational overhead, we use the meta-learning paradigm to improve the *GPI*-based algorithm. We use *Reptile* [30] to do the meta-learning task. *Reptile* is a first-order, gradient-based meta-learning algorithm that we favor due to its ability to avoid second-order gradient computations [9] and save computational resources.

4.2.1 R-GPI-LS/PD Meta Training

The details of the *Reptile*-based *GPI-LS/PD* (*R-GPI-LS/PD*) meta-training process are shown in Algorithm 2. After detecting the contexts with Algorithm 1, at each epoch of meta-training, one context is sampled from the context list. A new set of parameters ϕ' is calculated after *GPI* update. Then the parameters of the model are updated as done in line 6 in Algorithm 2.

After the meta-training phase, the model parameters are adjusted to ensure generally good performance and facilitate easy fine-tuning across various tasks.

[1] Under linear utility function the *CCS* is equivalent to the *PF*.

[2] In this work the weather data is from a London household [19] and the resultant renewable generation is from the simulation.

[3] We define a variation as "significant" when the reconstruction loss of the auto-encoder for the current day is greater than the average reconstruction loss calculated over the previous 7 days. For instance, transitioning from the continuous rainy weather of winter and spring to the clear skies of summer inevitably leads to changes in solar irradiance, resulting in a significant increase in the power output of renewable energy sources. This will result in a higher reconstruction loss and therefore create a context-shifting point. Such variations can also occur within the same season, especially in climates with variable weather.

Algorithm 2 R-GPI-LS/PD Meta Training

Input Learning rate ϵ; Initial model ϕ; Epochs number n_{epochs};
Run Algorithm 1 to find the set of different $[\ ]_{context}$
for $iteration = 1$ to n_{epochs} **do**
  Sample a $context$ from $[\ ]_{context}$
  Get updated parameter: $\phi' \leftarrow$GPI$(\pi_{\phi,context})$
  Do update the original parameter: $\phi \leftarrow \phi + \epsilon(\phi' - \phi)$
end for
Output Few-shot finetune MORL policy model

4.2.2 Finetune R-GPI-LS/PD

After the meta-training phase, the model is capable of being finetuned with few-shot learning. The finetuning procedure for the *R-GPI-LS/PD* model is detailed in Algorithm 3. Initially, the meta-trained model, denoted by ϕ, undergoes finetuning with data from the first day. The policy continues to operate normally until a contextual shift is detected. Upon detection of such a shift, ϕ is promptly finetuned again using data from the current day. This continuously progresses until all data from the entire year have been processed. During the finetuning process, all rewards obtained are recorded and later summed up at the end of the at the end of the year to compute the expected utility for the entire period.

Algorithm 3 R-GPI-LS/PD Finetuning

Input Learning rate α; Pre-trained (with Algorithm 2) model ϕ; Windowed annual dataset $\{win_1, ..., win_n\}$;
for win_i in $\{win_1, ..., win_n\}$ **do**
  if win_i is recognized as a new context by Algorithm 1 **then**
    Sample $win_i^{sub} \subsetneq win_i$
    $\phi_i \leftarrow$GPI$(\pi_{\phi,win_i^{sub}})$ ▷ Few-shot finetuning
  end if
  Conduct π_{ϕ_i} in the environment
end for

5 Experiments

In this section, we describe the simulation environment, which is constructed using real-world data. We detail the baselines, specifying the data volume and the number of training steps (interchangeably referred to as the training budget) allocated to each. We then introduce the metrics used, i.e. expected utility, PF approximation visualization, hypervolume, sparsity, and the return vector for two specific cases that either care more about the cost or maximize the comfort. We provide the code of our algorithm in [25]. We used the implementation of GPI-LS/PD provided in the original work [3].

5.1 Experiment Settings

5.1.1 Benchmark Environments

The weather data and background power demand[4] data used in this study are derived from residential settings in London (Latitude 51.331, Longitude 0.033, Elevation 182.3m, Hourly-based). Specifically, the weather data is obtained from [19] and processed using SAM [29] to simulate renewable energy generation (Using Solar Panel of SunPower Performance 17 SPR-P17-335-COM). Residential energy consumption data comes from [16], and electricity pricing is sourced from [6] with British Gas electricity rates at 36.62p/kWh during 08:00 - 23:00 and 15.18p/kWh from 23:00 - 08:00.

[4] The power demand of the other appliances manually turned on/off by the user.

We have removed the solar heat pump and other related boilers from the house. Instead, we have integrated the *Ariston VELIS EVO 80 L Electric Storage Water Heater* with a capacity of 1.5 KW, scheduling its operation for 4 hours between 0:00 - 8:00 every day. The agent manages the heater operation on an hourly basis.

The two objectives for MORL are:

1. to save the cost (£);

2. to maximize comfort (make the appliance work in assigned time slot).

The state space of our model is defined by a tuple consisting of the following elements: background power demand (kW), time (hrs), remaining task[5] (hrs), and renewable generation (kW).

The action space is binary, with only two options: 0 or 1. An action of "0" indicates turning the appliance off, whereas an action of "1" means turning the appliance on.

The reward space is constructed from two primary components: the hourly bill and comfort. The comfort reward is set to 1 when the appliance operates between 0:00 and 8:00 and there are remaining tasks, otherwise 0. The hourly bill reward is calculated as the negative value of the energy cost (£), penalizing higher bills and encouraging energy-efficient behaviors.

5.1.2 R-GPI-LS/PD & Finetune R-GPI-LS/PD

We summarize our methods, i.e. *R-GPI-LS/PD* and *Finetune R-GPI-LS/PD*.

i. *R-GPI-LS/PD*: It centers on meta-training the *GPI-LS/PD* policy with daily data at contextual shift points, emphasizing the ability to swiftly adapt to significant context shifts. Through this process, the method establishes a set of initial parameters that are easy to finetune.

ii. *Finetune R-GPI-LS/PD*: With the initial parameters from R-*GPI-LS/PD*. During the interaction with the environment, when new contextual shifts are detected, the policy is finetuned with the current day's data. This continuous finetuning can further improve the model's performance throughout changing contexts over the year.

5.1.3 Baselines

We detail the baselines we used in this Section. The primary baselines are variants of GPI-LS/PD as it is the current state-of-the-art. Notably, other cutting-edge MORL algorithms such as SFOLS [2] and Envelop Q Learning [43] were compared with GPI-LS/PD [3] and were found to be less effective.

5.1.3.1 Baselines in Main Evaluation

To evaluate our methods, several baseline models are used:

i. *GPI-LS/PD (month)*: It involves training a plain *GPI-LS/PD* model on data collected from January 2014 for 40,000 steps.

ii. *Finetuning GPI-LS/PD*: Starting with *GPI-LS/PD (month)*, it is retrained with the current day's data of detected contextual shifts, i.e. 12 shifts in total, for an additional 5,000 steps each.

iii. Rule-based policies:

Rule 1: Operates the appliance between 0:00 and 4:00.

Rule 2: Operates the appliance between 4:00 and 8:00.

5.1.3.2 Baselines in Ablation Study

To understand the contributions of various components of our method, we conduct an ablation study from two distinct perspectives:

i. Without the meta-learning method and context detection:

[5] "Remaining task" represents the number of hours left for an appliance to operate. For example, if an appliance needs to run for 4 hours every day and has been running for 1 hour, the remaining task is 3.

GPI-LS/PD (year): A plain *GPI-LS/PD* model is trained with the entire year data (2014-2015). This evaluates the performance of the *GPI-LS/PD* model simply with much more training data and without any meta-learning or context-detection approach.

ii. Without meta-learning method but with context detection:

Joint Training GPI-LS/PD: It incorporates solely context detection to the plain *GPI-LS/PD* to assess the impact of context detection on the model's adaptation to context shifts. We use a strong baseline i.e. joint training, which was also used in [30]. The plain GPI-LS/PD model is trained with concatenated day-based data from the 12 shifting points.

iii. The same rule-based policies as Section 5.1.3.1.

5.1.4 Candidate Methods Setting

We detail the training data volume and the training budget (the number of training steps) in Table 1 to compare the data and training efficiency.

The data volume is determined by multiplying the number of days by the 24 hours. Specifically, the *GPI-LS/PD (month)* and *GPI-LS/PD (year)* models utilize 720 and 8,650 data samples, respectively. *Joint Training GPI-LS/PD*, *R-GPI-LS/PD* and *Finetune R-GPI-LS/PD*, only use 288 data samples. *Finetune GPI-LS/PD* uses 720+288=1008 data samples[6].

GPI-LS/PD (month), *Finetune GPI-LS/PD*, *GPI-LS/PD (year)*, and *Joint Training GPI-LS/PD* use substantially more training budget. *Finetune GPI-LS/PD* are trained for 5000 steps at each shifting point. *R-GPI-LS/PD* are trained through 3 iterations on 10 shifting points sampled out of the 12 shifts at each iteration (each point for 480 steps). *Finetune R-GPI-LS/PD* are trained for 96 steps on each shifting point. However, the 14400 timesteps should also be counted as it is trained based on *R-GPI-LS/PD*[7].

Table 1. Data Volume and Training Budget

Methods	Data Volume	Training Budget
GPI-LS/PD (month)	30*24=720	40000
Finetune GPI-LS/PD	15*24+720=1008	12*5000+40000=100000
GPI-LS/PD (year)	365*24=8760	40000
Joint Training GPI-LS/PD	12*24=288	40000
R-GPI-LS/PD	12*24=288	10*3*480=14400
Finetune R-GPI-LS/PD	12*24=288	14400+12*96=15552

We have adjusted the number of gradient updates from 20 [3] to just 1 because the original frequency of gradient updates does not yield satisfactory results in our scenario. As each of the candidate algorithms uses the same gradient update frequency, this does not impact the fairness.

Each candidate method is executed multiple times using different random seeds to ensure the robustness of our results. The hyperparameters are illustrated in Table 3 in supplementary material [24].

5.1.5 Evaluation Metric

All baselines are evaluated on the performance over a full year (2014-2015). We use multiple MORL metrics to evaluate our methods and baselines, i.e. expected utility (EU)[8], PF approximation visualization, hypervolume (HV)[9], and sparsity (Sp)[10] of the PF solutions. For the specific evaluation for the energy community, we also evaluate the candidate methods by comparing the energy bill on preference [0.9,0.1] and comfort on preference [0.1,0.9][11].

As emphasized in the work of Hayes et al. [11], EU is a preferable metric as it is more suitable to assess the solution's practical value for the user, while the HV is a bit problematic for comparing solutions in real-scenario especially when utility function is known to be linear, and Sp is the metric to measure the density of coverage of the whole PF as a alternates of HV. The solution set with a higher HV and lower Sp is preferable. Given the high level of user involvement in our scenario, we prioritize the evaluation metrics accordingly, placing the greatest emphasis on the EU, followed by HV and Sp together to reflect the user-centric assessment. Simply speaking, the higher the EU and HV and the ratio of HV/Sp are, the lower the Sp is, the better the solutions are. In the field of energy applications, we assess the solution set using two metrics: The lower the bill with weights [0.9,0.1], and the higher the comfort with weights [0.1,0.9], the better the solution set is.

6 Experiment Results and Discussion

This section consists of the results of context detection, the main evaluation, the ablation study, and the discussion and summary.

6.1 Results of Context Shifting Detection

We present the outcomes of detecting context-shifting points (12 shifting points on days 1, 28, 42, 56, 70, 84, 112, 161, 203, 231, 266, and 357.) in Fig. 1.

One interesting observation is that the shifting points are more heavily concentrated towards the start of the year. This probably stems from the "spring cold snap", where spring temperatures suddenly drop to near-winter levels. This usually happens because cold air from the north can still affect warmer regions during spring and may cause sudden clouds, rain, or hail and influence renewable generation. Conversely, "autumn cold snap" is not common. This may be because the gradual temperature drop typically progresses steadily into winter without the sudden cooling seen in spring. It is therefore more stable than the spring season.

6.2 Main Evaluation

Evaluation results on EU, HV, Sp, and HV/Sp are shown in Fig. 2. The *GPI-LS (month)* and *Finetune GPI-LS* models display similar median values but the latter has a smaller variance. *GPI-PD (month)* has the highest median and max values among the first four baselines. It is counter-intuition that *Finetune GPI-PD* is the worst among the four baselines as the environment model was supposed to help with the policy improvement. This may be because the environment is

[6] *Finetune R-GPI-LS/PD* uses the same data as *R-GPI-LS/PD*.

[7] The same for *Finetune GPI-LS/PD*, it needs to add the steps used to train GPI-LS/PD (month).

[8] Calculated as $EU(\Pi) = E_{\mathbf{w}\sim\mathcal{W}}[max_{\pi\in\Pi} v^{\pi}_{\mathbf{w}}]$. This expectation is approximated by averaging the utility obtained from 100 evenly distributed weights from the simplex $\mathcal{W}$

[9] The sum of the hypercube spanned by the reference point and the solutions on PF. We use [-1300, 0] as the reference point.

[10] The sparsity $Sp(S) = \frac{1}{|S|-1}\sum_{j=1}^{m}\sum_{i=1}^{|S|}(\widetilde{S}_j(i) - \widetilde{S}_j(i+1))^2$, where S is the PF approximation, m is the number of objectives, $\widetilde{S}_j(i)$ is the i-th element of the sorted solutions on j-th objectives in S.

[11] The weight vector is the preference on [save cost, maximize comfort].

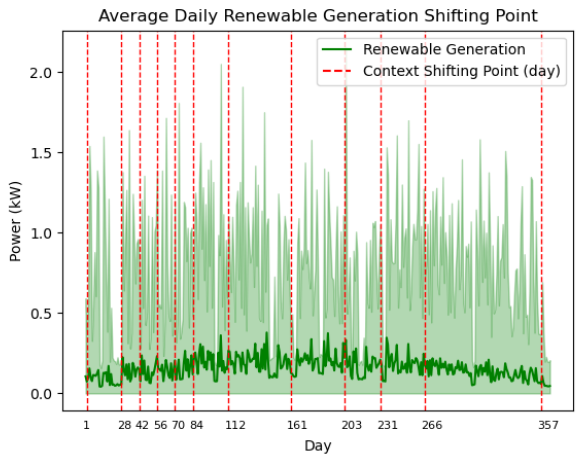

Figure 1. Context Shifting Detection

hard to accurately model during finetuning and therefore hinders policy improvement. Unfortunately, the medians of none of these four baselines exceed the rule-based policy. This shows that the pure GPI-LS/PD struggle in such changing environments.

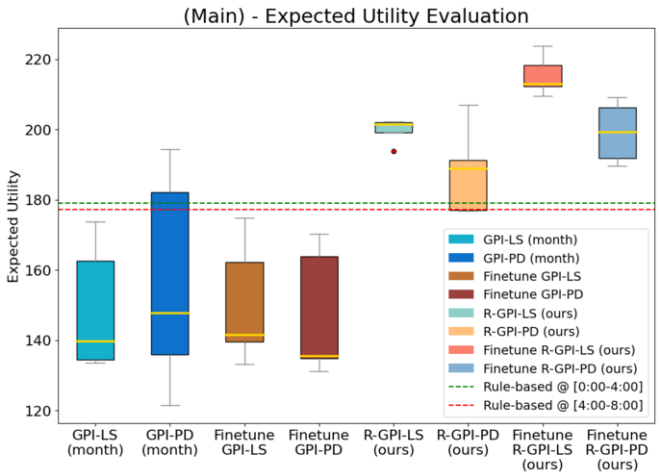

Figure 2. Main Evaluation Result - Expected Utility Evaluation

Our candidates, *R-GPI-LS/PD* and *Finetune R-GPI-LS/PD* surpass all baselines except a small overlap between *R-GPI-/PD* and *GPI-PD (month)*. Although *R-GPI-PD*'s performance sometimes falls below the rule-based policy, its median value is higher. After finetuning, *R-GPI-LS/PD* surpass other candidates. This demonstrates our methods' superior performance even with significantly less data (60.0% - 71.43% less) and a limited training budget (61.12% - 85.6% less). See Table 3 in the supplementary material [24] for more details of comparison. One interesting observation is that although *R-GPI-LS/PD* receive performance improvement after finetuning, the model-based variants fail to surpass their model-free counterparts. This aligns with the observation between *Finetune GPI-LS/PD* and implies that the environment model struggles to grab accurate information when the context is swiftly changing in finetuning.

In Fig. 3, we visualize the PF approximation and evaluate the HV, Sp, and their ratio. The first four baselines' evaluation shows that directly finetuning the learned policy does not have a significant influence on the performance. Though they can sometimes find better solutions than our methods, they still fail to achieve a better HV (except GPI-PD (month) that achieves a comparable HV as our methods) and lower sparsity. Our methods outperform all the baselines in the Sp, and HV/Sp. *R-GPI-LS/PD* has achieved the highest HV and it has achieved the lowest Sp after fintuning. We find that *R-GPI-LS/PD* have seen a drop after finetuning, however, their Sp have also decreased, which demonstrates an improvement of the solution density/quality [11]. During finetuning, the ratio of HV/Sp of our methods has either increased or maintained a comparable performance.

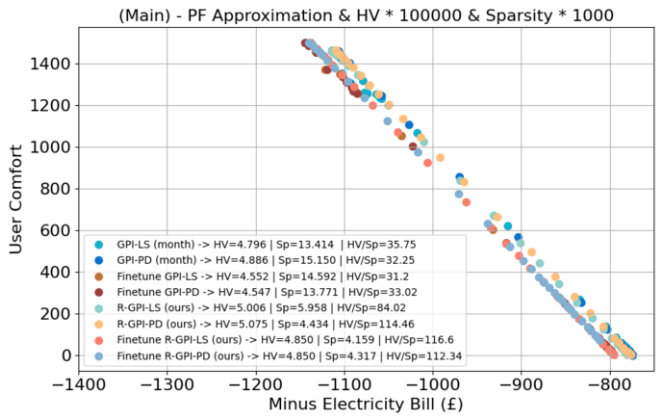

Figure 3. Main Evaluation Result- PF Approx. & HV & Sp

6.3 *Ablation Study*

We conduct the ablation study and show the results in Fig. 4 and Fig. 5. Recall that the ablation study focuses on two aspects, i.e. without both context detection and meta-learning, i.e. *GPI-LS/PD (year)* and only without meta-learning, i.e. *Joint Training GPI-LS/PD*.

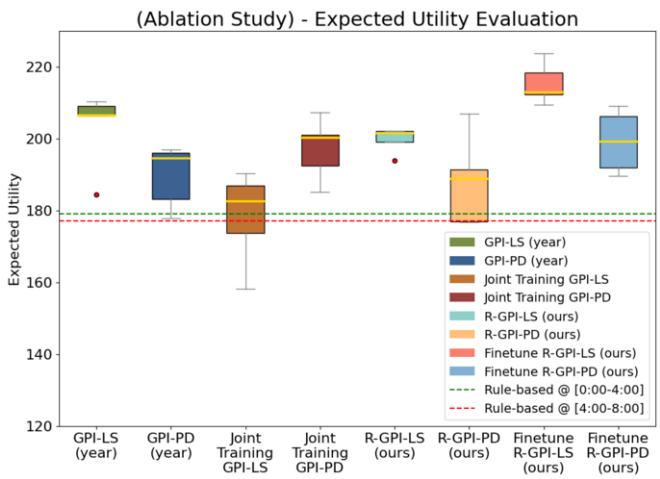

Figure 4. Ablation Study - Expected Utility Evaluation

Each candidate's EU medians surpass rule-based policies. In most cases, *GPI-PD* variants see larger variance than their *GPI-LS* counterparts, except the joint training cases where the *GPI-PD* variant is marginally lower in variance. This is consistent with Fig. 2 that *GPI-PD* variants can bring more variance. This may stem from the fact that some state elements are out of the agent's control, e.g. renewable generation, which makes the environment model's prediction of the next state incorrect and hinders the training. Notably, only the joint training setting and the month-based training setting have seen better performance from the model-based variant. The internal similarity of *GPI-PD (month)* and *Joint Training GPI-PD* is the training environment. *GPI-PD (month)* is trained with an almost stable environment (1-30 days) where the next shifting point is just detected on day 28.

Joint Training GPI-PD is trained with the day-based concatenated 12 shifting points. It is a synthetic environment and shifting points are evenly distributed. *GPI-PD (year)* can be deemed as a data-dense version of joint training where the different contexts are also concatenated but not evenly distributed. However, the difference of context distribution between year-based train and joint training has seen contrary results. These observations mean that the model-based MORL algorithm needs either a stable environment or an environment with evenly distributed shifting contexts. We leave this for future work that the model-based RL algorithm needs improvement to handle these problems, i.e. unevenly distributed contexts and performance drop in finetuning.

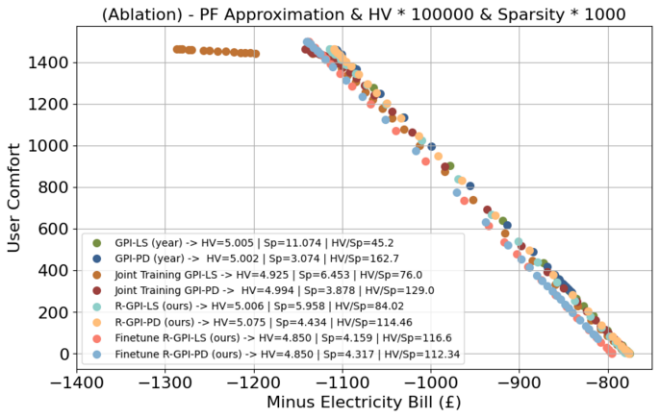

Figure 5. Ablation Study - PF Approx. & Hypervolume & Sparsity

In Fig. 5, we show the PF approximation, the HV, SP, and HV/SP. One interesting result is that the best performance on HV, SP, and HV/SP is *GPI-PD (year)* which is not very well-performing according to the EU evaluation. This is because it finds some good solutions by following incorrect preferences. This can be proved by the result shown in Table 2, where neither its bill nor comfort is the best. It is consistent with the statement from [11] that simply using HV or Sp cannot fully show the value of a solution set in a practical scenario. The second best method on HV/SP is *Joint Training GPI-PD*, while our methods still are among the best-performing ones.

The joint training methods have demonstrated better performance than *GPI-LS/PD* or *Finetune GPI-LS/PD* models, with 60% and 71.43% less data volume respectively. This indicates that context detection reduces the need for extensive data samples. The superior performance of our methods over the joint training further underscores the effectiveness of meta-learning, which achieves even better results with 64% less training budget.

Although the year-based *GPI-LS/PD* has comparable performance to other methods, it has a higher computational cost—using 96.71% more data and over 60% more training steps than our methods.

This demonstrates the advantages of integrating meta-learning and context detection and validates our highly efficient approach.

6.4 Result Summary

We summarize the evaluation in Table 2[12]. See percentage comparison in Appendix A and full solution visualization in Appendix B in the supplementary material [24].

It does not surprise us that simply finetuning the plain *GPI* models cannot achieve performance improvement (even causing a decrease). This is because the internal representation of the trained model may only be suitable to specific contexts but not to others. When doing finetuning, these "stubborn" representations can hinder the parameter update. With meta-learning, the model can find the most context-sensitive parameters that are capable of later fintuning.

The ablation study validates all components of our algorithms. Notably, *GPI-LS/PD (year)* accesses diverse and more extensive data. Common sense suggests that training with more data should enhance model performance, however, this is not always true according to our experiment, i.e. the *Joint Training GPI-PD* is better than *GPI-PD (year)*. As we mentioned in Section 6.3, this should be due to the unevenly distributed context shifting.

Joint Training GPI-LS/PD are still surpassed by *Finetune R-GPI-LS/PD*, and merely match *R-GPI-LS/PD*. As stated in [30], that the joint-training method seeks to optimize several tasks, while *Reptile* considers second-and-higher order of derivatives when multiple gradient updates are conducted. This can explain the superior outcomes of our method.

Our *Finetune R-GPI-LS* is the best performing algorithm when compared with the second best method *GPI-LS (year)*. It uses 96.71% less data and 61.1% less training steps to achieve 5.9% improvement on EU, 62.44% improvement on Sp, 3.28% on bill and 2.74% improvement on comfort but only has 3.1% drop on HV. We conduct a detailed comparison in supplementary material [24].

Table 2. Summary of Evaluation

	Metric				
Baselines	**EU↑**	**HV↑**	**Sp↓**	**Bill↓**	**Comfort↑**
GPI-LS (month)	148.85	479597.66	13414.23	£781.96	1296.2
GPI-PD (month)	156.36	488625.97	15149.91	**£775.03**	1347
Finetune GPI-LS	150.31	455233.65	14592.21	£802.41	1325.6
Finetune GPI-PD	147.10	454665.54	13770.9	£802.32	1324.8
GPI-LS (year)	203.37	500506.12	11073.96	£840.93	1460
GPI-PD (year)	189.72	500212.43	6453.05	£853.97	1460
Joint Training GPI-LS	178.37	492483.88	**3074.39**	£834.31	1449
Joint Training GPI-PD	197.29	499427.29	3878.36	£807.7	1457.8
R-GPI-LS	199.75	500631.35	5958.15	£791.39	1464
R-GPI-PD	188.2	**507535.27**	4434.33	£781.08	1464
Finetune R-GPI-LS	**215.37**	485013.61	4159.47	£813.35	**1500**
Finetune R-GPI-PD	199.23	484968.16	4316.89	£857.89	**1500**

7 Conclusion

In this paper, we highlighted the suitability of meta-learning for energy management and demonstrated superior performance by extending state-of-the-art MORL algorithms with *Reptile*. We applied and evaluated the *GPI-LS/PD* in residential appliance scheduling.

There are several interesting questions identified in our research that merit further investigation:

1. We notice that the finetuning technique cannot seamlessly fit model-based MORL. This was once mentioned in the work of Mendonca et al. [27]. However, it is not yet fully explored in MORL settings. It remains a open question that how to finetune the environment model to positively help with the training of MORL policy when the task contexts are not evenly distributed.

2. The environment used is only 2-objective, evaluations in the environment with more objectives (e.g. peak shaving) are still an open question. We also plan to do further investigation on our algorithm about the performance in other environments.

3. Our work only talks about single-agent cases, however, the real-life scenario usually involves multiple appliances for scheduling. This renders the problem as a multi-agent problem which needs further exploration.

[12] The up arrow ↑ means the higher the value is, the better the policy is, while the down arrow ↓ means the lower the value is, the better the policy is. The bold value is the best performance on that metric.

Acknowledgements

This research is funded by Irish Research Council, the Government of Ireland Postgraduate Scholarship (GOIPG/2022/2140).

References

[1] A. Abels, D. Roijers, T. Lenaerts, A. Nowé, and D. Steckelmacher. Dynamic weights in multi-objective deep reinforcement learning. In *International conference on machine learning*, pages 11–20. PMLR, 2019.

[2] L. N. Alegre, A. Bazzan, and B. C. Da Silva. Optimistic linear support and successor features as a basis for optimal policy transfer. In *International conference on machine learning*. PMLR, 2022.

[3] L. N. Alegre, A. L. Bazzan, D. M. Roijers, A. Nowé, and B. C. da Silva. Sample-efficient multi-objective learning via generalized policy improvement prioritization. *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, 2023.

[4] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga. Usad: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3395–3404, 2020.

[5] T. Basaklar, S. Gumussoy, and U. Y. Ogras. Pd-morl: Preference-driven multi-objective reinforcement learning algorithm. *International Conference on Learning Representations*, 2023.

[6] C. Casalis and A. Capstick. Is economy 7 worth it?, 2023. URL https://www.moneysavingexpert.com/utilities/economy-7/.

[7] N. Chauhan, N. Choudhary, and K. George. A comparison of reinforcement learning based approaches to appliance scheduling. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 253–258. IEEE, 2016.

[8] S. Conti, R. Nicolosi, S. Rizzo, and H. Zeineldin. Optimal dispatching of distributed generators and storage systems for mv islanded microgrids. *IEEE transactions on power delivery*, 27(3):1243–1251, 2012.

[9] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[10] D. Hadka and P. Reed. Borg: An auto-adaptive many-objective evolutionary computing framework. *Evolutionary computation*, 21(2):231–259, 2013.

[11] C. F. Hayes, R. Rădulescu, E. Bargiacchi, J. Källström, M. Macfarlane, M. Reymond, T. Verstraeten, L. M. Zintgraf, R. Dazeley, F. Heintz, et al. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022.

[12] IEA. Fuel share of co2 emissions from fuel combustion, iea, paris, 2018. URL https://www.iea.org/data-and-statistics/charts/fuel-share-of-co2-emissions-from-fuel-combustion-2018.

[13] IEA. Key world energy statistics, iea, paris, 2021. URL https://www.iea.org/reports/key-world-energy-statistics-2021.

[14] A. Jalalimanesh, H. S. Haghighi, A. Ahmadi, H. Hejazian, and M. Soltani. Multi-objective optimization of radiotherapy: distributed q-learning and agent-based simulation. *Journal of Experimental & Theoretical artificial intelligence*, 29(5):1071–1086, 2017.

[15] J. Källström and F. Heintz. Tunable dynamics in agent-based simulation using multi-objective reinforcement learning. In *Adaptive and Learning Agents Workshop (ALA-19) at AAMAS, Montreal, Canada*, 2019.

[16] J. Kelly and W. Knottenbelt. The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes. *Scientific data*, 2(1):1–14, 2015.

[17] S. Khadka and K. Tumer. Evolutionary reinforcement learning. *arXiv preprint arXiv:1805.07917*, 223, 2018.

[18] E. B. Laber, D. J. Lizotte, and B. Ferguson. Set-valued dynamic treatment regimes for competing outcomes. *Biometrics*, 70(1):53–61, 2014.

[19] L. K. Lawrie and D. B. Crawley. Development of global typical meteorological years (tmyx), 2022. URL http://climate.onebuilding.org.

[20] M. V. Liu, P. M. Reed, D. Gold, G. Quist, and C. L. Anderson. A multiobjective reinforcement learning framework for microgrid energy management. *arXiv preprint arXiv:2307.08692*, 2023.

[21] J. Lu, P. Mannion, and K. Mason. A multi-objective multi-agent deep reinforcement learning approach to residential appliance scheduling. *IET Smart Grid*, 5(4):260–280, 2022.

[22] J. Lu, P. Mannion, and K. Mason. Go-explore for residential energy management. In *European Conference on Artificial Intelligence*, pages 133–139. Springer, 2023.

[23] J. Lu, P. Mannion, and K. Mason. Inferring preferences from demonstrations in multi-objective residential energy management. *arXiv e-prints*, pages arXiv–2401, 2024.

[24] J. Lu, P. Mannion, and K. Mason. Full version of "a meta-learning approach for multi-objective reinforcement learning in sustainable home environments". *arXiv preprint arXiv:2407.11489*, 2024.

[25] J. Lu, P. Mannion, and K. Mason. Code and data for "a meta-learning approach for multi-objective reinforcement learning in sustainable home environments", 2024. URL https://github.com/JLu2022/Meta-Learning-MORL-Residential-Energy-Management.git.

[26] S. Mannor and N. Shimkin. The steering approach for multi-criteria reinforcement learning. *Advances in Neural Information Processing Systems*, 14, 2001.

[27] R. Mendonca, X. Geng, C. Finn, and S. Levine. Meta-reinforcement learning robust to distributional shift via model identification and experience relabeling. *arXiv preprint arXiv:2006.07178*, 2020.

[28] H. Mossalam, Y. M. Assael, D. M. Roijers, and S. Whiteson. Multi-objective deep reinforcement learning. *arXiv preprint arXiv:1610.02707*, 2016.

[29] National Renewable Energy Laboratory. System advisor model version 2022.11.29 (sam 2022.11.21), 2022. URL https://sam.nrel.gov. Accessed July 26, 2023.

[30] A. Nichol, J. Achiam, and J. Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

[31] P. Pourghasem, F. Sohrabi, M. Abapour, and B. Mohammadi-Ivatloo. Stochastic multi-objective dynamic dispatch of renewable and chp-based islanded microgrids. *Electric Power Systems Research*, 2019.

[32] A. C. Real, G. P. Luz, J. Sousa, M. Brito, and S. Vieira. Optimization of a photovoltaic-battery system using deep reinforcement learning and load forecasting. *Energy and AI*, 16:100347, 2024.

[33] A. Riebel, J. M. Cardemil, and E. López. Multi-objective deep reinforcement learning for a water heating system with solar energy and heat recovery. *Energy*, 291:130296, 2024.

[34] D. M. Roijers. Multi-objective decision-theoretic planning. *AI Matters*, 2(4):11–12, 2016.

[35] D. M. Roijers, S. Whiteson, and F. A. Oliehoek. Linear support for multi-objective coordination graphs. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multiagent Systems*, volume 2, pages 1297–1304. IFAAMAS/ACM, 2014.

[36] K. Saberi, H. Pashaei-Didani, R. Nourollahi, K. Zare, and S. Nojavan. Optimal performance of cchp based microgrid considering environmental issue in the presence of real time demand response. *Sustainable cities and society*, 45:596–606, 2019.

[37] G. Tesauro, R. Das, H. Chan, J. Kephart, D. Levine, F. Rawson, and C. Lefurgy. Managing power consumption and performance of computing systems using reinforcement learning. *Advances in neural information processing systems*, 20, 2007.

[38] K. Van Moffaert, M. M. Drugan, and A. Nowé. Scalarized multi-objective reinforcement learning: Novel design techniques. In *2013 IEEE symposium on adaptive dynamic programming and reinforcement learning (ADPRL)*, pages 191–199. IEEE, 2013.

[39] A. Vettoruzzo, M.-R. Bouguelia, J. Vanschoren, T. Rognvaldsson, and K. Santosh. Advances and challenges in meta-learning: A technical review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[40] D. Wang, Y. Cheng, M. Yu, X. Guo, and T. Zhang. A hybrid approach with optimization-based and metric-based meta-learner for few-shot learning. *Neurocomputing*, 349:202–211, 2019.

[41] J. Wu, D. Song, X. Zhang, C. Duan, and D. Yang. Multi-objective reinforcement learning-based energy management for fuel cell vehicles considering lifecycle costs. *International Journal of Hydrogen Energy*, 48(95):37385–37401, 2023.

[42] J. Xu, K. Li, and M. Abusara. Preference based multi-objective reinforcement learning for multi-microgrid system optimization problem in smart grid. *Memetic Computing*, 14(2):225–235, 2022.

[43] R. Yang, X. Sun, and K. Narasimhan. A generalized algorithm for multi-objective reinforcement learning and policy adaptation. *Advances in neural information processing systems*, 32, 2019.

[44] X. Yang, Z. Leng, S. Xu, C. Yang, L. Yang, K. Liu, Y. Song, and L. Zhang. Multi-objective optimal scheduling for cchp microgrids considering peak-load reduction by augmented ε-constraint method. *Renewable Energy*, 172:408–423, 2021.

[45] X. Yang, E. Howley, and M. Schukat. Adt: Agent-based dynamic thresholding for anomaly detection. *arXiv:2312.01488*, 2023.

[46] X. Yang, E. Howley, and M. Schukat. Adt: Time series anomaly detection for cyber-physical systems via deep reinforcement learning. *Computers & Security*, page 103825, 2024.

[47] C. Zheng, D. Zhang, Y. Xiao, and W. Li. Reinforcement learning-based energy management strategies of fuel cell hybrid vehicles with multi-objective control. *Journal of Power Sources*, 543:231841, 2022.