Sustainable Data Center Modeling: A Multi-Agent Reinforcement Learning Benchmark

Soumyendu Sarkar[†*], **Avisek Naug**[†], **Ricardo Luna Gutierrez**[†], **Antonio Guillen**[†],
Vineet Gundecha, Sahand Ghorbanpour, Sajad Mousavi, Ashwin Ramesh Babu

Hewlett Packard Enterprise (Hewlett Packard Labs)

```
soumyendu.sarkar, avisek.naug, rluna, antonio.guillen, vineet.gundecha,
    sahand.ghorbanpour, sajad.mousavi, ashwin.ramesh-babu @hpe.com
```

Abstract

The rapid growth of machine learning (ML) has led to an increased demand for computational power, resulting in larger data centers (DCs) and higher energy consumption. To address this issue and reduce carbon emissions, intelligent control of DC components such as cooling, load shifting, and energy storage is essential. However, the complexity of managing these controls in tandem with external factors like weather and green energy availability presents a significant challenge. While some individual components like HVAC control have seen research in Reinforcement Learning (RL), there's a gap in holistic optimization covering all elements simultaneously. To tackle this, we've developed $\mathcal{DCRL}$-$Green$ a multi-agent RL environment that empowers the ML community to research, develop, and refine RL controllers for carbon footprint reduction in DCs. $\mathcal{DCRL}$-$Green$ is a flexible, modular, scalable, and configurable platform that can handle large High Performance Computing (HPC) clusters. In its default setup, $\mathcal{DCRL}$-$Green$ also provides a benchmark for evaluating multi-agent RL algorithms, facilitating collaboration and progress in green computing research. Link: Data Center Green Dashboard

1 Introduction

With the increasing demand for computational power from artificial intelligence (AI) and other high-performance computing applications, the energy consumption required to run and cool Data Centers (DCs) is increasing exponentially. This high energy consumption contributes to carbon emissions and exacerbates climate change. Embracing sustainability practices helps minimize environmental impact by reducing energy consumption and leveraging renewable energy sources available in the power grid. Also, governments and regulatory bodies are increasingly focusing on environmental sustainability and imposing stricter regulations to reduce carbon emissions. Hence, there is an urgent necessity for sustainable High-Performance Computing (HPC) DCs.

The main contribution of this paper is a highly customizable data center model for an OpenAI Gym environment for sustainability, offering flexibility in controlling cooling, flexible load shifting, and energy storage in UPS batteries. This model supports multiple objectives like reduction in energy consumption, carbon footprint reduction, and energy cost. It also discusses multi-agent Reinforcement Learning (MARL) controllers and provides a benchmark for collaborative MARL with heterogeneous agents.

---

[*]Corresponding author
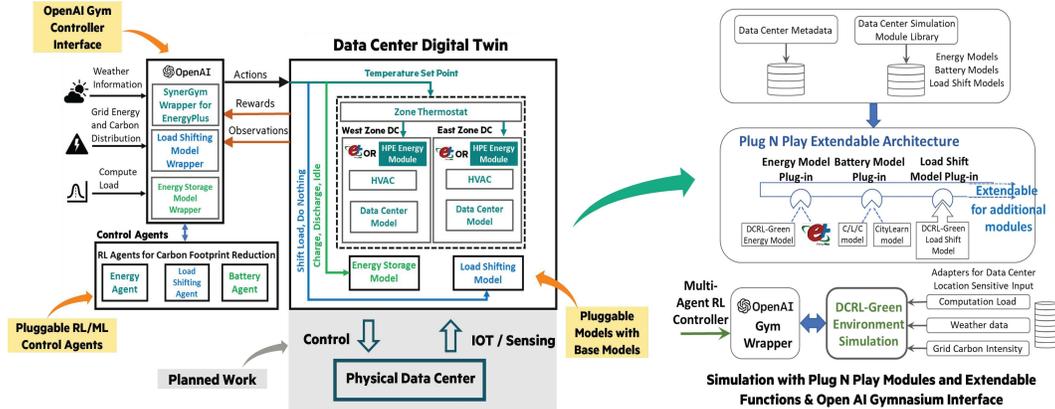[†]These authors contributed equally

Figure 1: DCRL-Green Data Center Digital Twin Configurable Modeling.

.

2   Background and Related Work

Recent advances in Reinforcement Learning (RL) have led to the creation of key RL environments for optimizing energy and resource allocation in building and data center management. Notable examples include CityLearn Vázquez-Canteli et al. (2019) for urban energy coordination, Sinergym for RL evaluation in building models, Energym Scharnhorst et al. (2021) for climate and energy control assessments, and RL-Testbed Moriyama et al. (2018) for dynamic data center cooling management. All of these rely on EnergyPlus Crawley et al. (2000), a widely-used energy simulation software.

Facebook's "Carbon Explorer" Acun et al. (2023) and Google's "Carbon-Aware Computing for Datacenters" Radovanović et al. (2023) both aim to reduce data center carbon emissions. Facebook shifts data center load to low-carbon hours along with energy storage, achieving a 4% reduction, while Google optimizes workload distribution for a 2% reduction. Both rely on static optimization and subsequently on accurate long-term forecasts, which are vulnerable to changing weather patterns and other factors, making their effectiveness less reliable.

Even though there are implementations on HVAC cooling, there is no framework that optimizes the cooling, load shifting, and energy simultaneously in real-time. The Energyplus implementations for data centers are inflexible and challenging to integrate with Machine Learning software due to their non-Pythonic nature. Our proposed approach, DCRL is developed wholly in Python and facilitates integration with Machine Learning software, overcoming the challenges posed by EnergyPlus' inflexible thermal modeling.

3   DCRL-Green Environment

DCRL is an OpenAI Gym-based framework designed for developing and evaluating collaborative, heterogeneous Multi-Agent Reinforcement Learning (MARL) algorithms in data centers (DCs), with a focus on sustainability. It supports three problem types: Carbon-Aware Flexible Load Shifting, DC HVAC Cooling Energy Optimization, and Carbon-Aware Auxiliary Battery Supply. These components are shown in figure 1 and described next:

1. **Data Center Digital Twin**: At the center of DCRL-Green is the Digital Twin of a Data Center. It can integrate external Data Center models implemented in EnergyPlus, Open Modelica, and Python. It hierarchically models different components of a Physical Data Center like servers, IT Cabinets, and entire IT Rooms. It also models the HVAC components like a chiller, evaporator, refrigerator sections, and a cooling tower to manage heat removal. It calculates their power consumption, accounting for factors like weather conditions and performance characteristics. Users can connect custom energy models to the HVAC system. The control and IOT/Sensing interface enables real-time "fully living" Digital Twin enablement in the data center landscape. Furthermore, users can extensively customize DC designs, specifying individual servers within racks, via a JSON object file. This customization covers DC geometry, server characteristics, HVAC specifics, and more. External settings like workload profiles, weather data, and grid carbon intensity are also adjustable Naug et al. (2023).

2

2. **Load Shifting**: DCRL features a load scheduler for optimizing flexible workload assignments, allowing users to customize load budgets and daily workloads. It employs a load-shifting reward mechanism to minimize the DC's carbon footprint by allocating workloads to time slots with lower carbon intensity.

3. **Energy Storage**: The energy storage component reduces grid dependency during high carbon intensity periods, following a model based on previous work with added modifications for simulating real-world charging rates. It considers charge levels, charge/discharge functions, rate limitations, and manufacturer specifications.

4. **Open AI Gym Controller Interface**: In order to enable real-time control using Deep Reinforcement Learning, DCRL-Green has a MultiAgent Interface. It supports RL agents for carbon-aware workload shifting, cooling energy optimization, and battery use. The individual wrappers are customized so that we can also test single agent reinforcement learning applications.

5. **Control Agents**: Based on the Controller Interface, we deploy three control agents: Energy Agent which optimizes HVAC cooling energy, Load Shifting Agent which performs carbon aware load shifting and the Battery Agent which reduces grid dependency during carbon intensive hours by using stored charge. These agents employ a Multi-Agent Reinforcement Learning (MARL) approach by utilizing information like real-time CPU workload, spatial temperature gradientsSun et al. (2021), power consumption estimates, and grid and weather data to chose the best set of actions for the load-shifting model, HVAC cooling and Energy storage model located with the Data Center Digital Twin. Key Performance metrics like total IT power, CRAC Fan power, and CRAC Evaporator power and overall Carbon footprint is calculated for measuring DCRL Green performance.

6. **Plug N Play Extendable Architecture**: On the right-hand side of figure 1, we showcase the most important aspect of the paper which allows plugging in different models, different spatio-temporal data sources and applying any choice of control algorithm. This opens up the DCRL Green architecture to any other dynamic system control problem and is not limited to data centers.

4   Multi Agent Control Problem

We now look at the theoretical formulation of the problem for solution using multi-agent reinforcement Learning. We outline 1) the system models, 2) the MDP formulation for each model, 3) the corresponding objectives and rewards, and 4) example solution architectures.

Figure 2: Internal and External System Dependencies

4.1   System Models

Flexible Load Shifter: Let B_t be the instantaneous DC server workload requirement at time t and $x\%$ of the load is flexible and can be shifted to the future up to a time horizon of N hours in the future. The goal of the Flexible Load Shifter is to use information like historical grid CI data (CI_t), UPS Battery State of Charge (Bat_SoC) and DC load ($P_{HVAC,cooling,t}$) to choose the carbon optimal actions (i.e.: $A_t = reschedule$ or $idle$) for rescheduling the flexible load components over the time horizon N.

If the episode is of length T, and the Net Carbon Footprint due to these actions at time t be, CFP_t then the goal of the agent would be:

$$(A_{bat,0}, A_{bat,1}, \ldots, A_{bat}) = argmin \sum_{t=0}^{t=T} CFP_t \qquad (1)$$

Here CFP_t would be evaluated based on the net DC load resulting from the rescheduling actions, the optimized HVAC cooling and charging and discharging of the battery at every time step. This
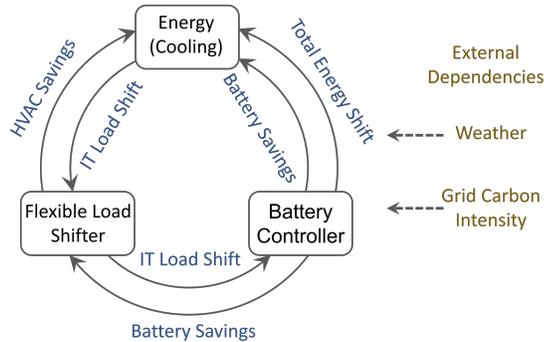
dependency is highlighted in Figure 2 where the outputs of Flexible Load Shifting affects the Battery Model and HVAC Cooling blocks.

Data Center IT & HVAC Cooling Model: For the DC IT and HVAC Cooling models please refer to Naug et al. (2023).

Battery Storage: The battery storage module implements the charging $f_{charging}(BatSoc, \delta\tau)$ and discharging $f_{charging}(BatSoc, \delta\tau)$ models from Acun et al. (2023). The charging and discharging processes are influenced by the battery state of charge $BatSoc$ and the time over which the process happens $\delta\tau$. The goal of the battery agent is to supplement the DC load $P_{HVAC,cooling,t}$ by discharging ($A_{bat,t} = discharge$) during higher grid CI periods, while charging ($A_{bat,t} = charge$) during lower grid CI (CI_t) or maintain the current battery state of charge ($A_{bat,t} = idle$).

If the episode is of length T, and the Net Carbon Footprint due to these actions at time t be, CFP_t then the goal of the agent would be:

$$(A_{bat,0}, A_{bat,1}, \ldots, A_{bat}) = argmin \sum_{t=0}^{t=T} CFP_t \tag{2}$$

In summary, we realize that three environments and their corresponding supervisory control agents need to exchange information due to the causal relationship between the components in a manner outlined below and schematically shown in Figure 2.

$$(Agent_{LS}, Env_{LS}) : (B_t \times Bat\_SoC \times CI_t \times P_{HVAC,cooling,t} \times \theta_{LS}) \rightarrow \tilde{B}_t \tag{3}$$

$$(Agent_{DC}, Env_{DC}) : (\tilde{B}_t \times Bat\_SoC \times CI_t \times \theta_{DC}) \rightarrow P_{HVAC,cooling,t} \tag{4}$$

$$(Agent_{BAT}, Env_{BAT}) : (P_{HVAC,cooling,t} \times Bat\_SoC \times CI_t \times \theta_{BAT}) \rightarrow Bat\_SoC \tag{5}$$

Here θ_{LS}, θ_{DC} and θ_{BAT} include variables that are not directly shared between environments; e.g. weather data and temporal information like *sine and cosine* of hour of day and day of year to capture non-stationary trends. Hence, we implement a MARL setup where the individual agents can exchange the relevant information that is needed for the decision-making as well as simulation.

4.2 MDPs

The observation space in each environment enables RL agents to collaboratively make informed decisions by sharing relevant variables. For instance, the Energy agent's observation space includes outdoor temperature, cooling setpoint, energy consumption, and shared variables like server CPU utilization and battery charge state. Additionally, the environments have shared global variables such as time, weather, and CI.

Table 1: Overview of the observation space and action space of each environment.

Environment	Observation Space	Action Space
$Agent_{LS}$	CPU utilization, Unassigned Flexible Load, CI, IT Energy consumption, Internal temp., BatSoC.	Allocate workload, Idle
$Agent_{DC}$	External/Internal temp., Cooling setpoint, HVAC/IT Energy consumption, CPU utilization, BatSoC.	Adjust temp. setpoint, Idle
$Agent_{BAT}$	BatSoC, CI, HVAC/IT Energy consumption, CPU utilization.	Charge, Discharge, Idle

Table 1 illustrates examples of observation and action spaces in the $\mathcal{DCRL}\text{-}Green$ framework for the load shifting agent $Agent_{LS}$, energy agent $Agent_E$, and battery agent $Agent_{BAT}$. These spaces can be tailored to the unique attributes and environmental conditions of the specific DC being modeled.

4.3 Objectives and Rewards

Our framework provides an interface where the end-user can choose to train three agents independently of each other's reward feedback, or consider a collaborative reward approach. The individual rewards are derived from objectives that are relevant to each agent:

$$r_{LS} = -(CO_2\ Footprint + LS_{Penalty})$$
$$r_{DC} = -(Total\ Energy\ Consumption)$$
$$r_{BAT} = -(CO_2\ Footprint)$$

Here $LS_{Penalty}$ is a penalty attributed to the Load Shifting Agent if it fails to schedule all the required load within a specified time horizon. Based on the individual rewards, we formulate a collaborative reward structure where each agent gets partial feedback in the form of rewards from the other agent-environment pair.

Hence, the collaborative feedback reward formulation for each agent is formulated as:

$$R_{LS} = 0.8 * r_{LS} + 0.1 * r_E + 0.1 * r_{BAT}$$
$$R_{DC} = 0.1 * r_{LS} + 0.8 * r_E + 0.1 * r_{BAT}$$
$$R_{BAT} = 0.1 * r_{LS} + 0.1 * r_E + 0.8 * r_{BAT}$$

The reward-sharing mechanism allows the agents to estimate the feedback from their actions in other environments. For example, the CPU Load affects the DC power demand, which in turn affects the battery optimizer's decision to charge or discharge, resulting in a particular net $CO_2\ Footprint$. Hence, the need to have a collaborative reward structure.

4.4 MARL agent architectures

We use two multi-agent training approaches: independent learning treats agents as individuals as in IPPO (de Witt et al., 2020) , while centralized learning with decentralized execution uses a centralized critic as in MADDPG (Lowe et al., 2017)

5 Results

5.1 Carbon Footprint Reduction

To assess the carbon reduction capabilities of the agents in our framework, we conduct evaluations on three distinct locations. For these experiments, we utilize EnergyPlus for the energy simulations. The results of this evaluation are shown in Table 2. These tables shows how MADDPG algorithm performed when compared to ASHRAE RBC when using $\mathcal{DCRL}$-$Green$.

DC Max Load 1.2MWh - Experiment period 1 year							
% Reduction of Carbon Footprint of RL (MADPPG) using $\mathcal{DCRL}$-$Green$ compared to Industry Standard ASHRAE Guideline 36							
Algorithms	Load Shifting(LS)	RL Cooling(DC)	Battery(BAT)	LS+DC	LS+BAT	DC+BAT	**LS+DC+BAT**
Arizona	8.76 ± 0.50	5.81 ± 2.09	0.24 ± 0.44	11.87 ± 1.36	8.96 ± 0.50	7.21 ± 1.98	**13.4 ± 0.48**
New York	8.02 ± 0.13	5.09 ± 0.09	0.17 ± 0.04	11.32 ± 0.05	8.27 ± 0.11	6.64 ± 0.13	**13.01 ± 0.12**
Washington	8.21 ± 0.05	7.19 ± 0.03	0.32 ± 0.05	12.21 ± 0.12	8.54 ± 0.07	7.68 ± 0.07	**13.27 ± 0.06**
% Reduction of Energy Consumption of RL (MADPPG) using $\mathcal{DCRL}$-$Green$ compared to ASHRAE							
Algorithms	LS	DC	BAT	LS+DC	LS+BAT	DC+BAT	**LS+DC+BAT**
Arizona	8.25 ± 0.43	5.71 ± 2.02	0.00 ± 0.00	11.76 ± 1.16	8.49 ± 0.45	7.02 ± 1.25	**13.38 ± 0.62**
New York	8.09 ± 0.12	4.89 ± 0.09	0.00 ± 0.00	11.02 ± 0.06	8.36 ± 0.12	6.46 ± 0.13	**12.77 ± 0.11**
Washington	8.11 ± 0.05	7.47 ± 0.03	0.00 ± 0.00	12.55 ± 0.11	8.42 ± 0.07	7.87 ± 0.06	**13.51 ± 0.05**

Table 2: **% Reduction of Carbon Footprint and Energy Consumption of RL** (MADPPG) compared to industry standard ASHRAE. We are ignoring the embodied footprint for server and battery and only considering operational footprint.

6 Conclusion and Future Work

The paper introduces $\mathcal{DCRL}$-$Green$ an OpenAI Gym environment designed for implementing reinforcement learning in data centers (DCs) to improve sustainability. It is the first environment that combines real-time RL control for optimizing cooling, load shifting, and energy storage in DCs. It allows users to customize their DC installations, particularly focusing on cooling and rack arrangements. Users can select from multiple reward functions to target specific sustainability goals such as reducing carbon footprint, energy consumption, and cost. It provides an opportunity for ML researchers to contribute to addressing climate change concerns related to the increasing ML workloads of DC.

The current data center environment implementation uses parameterization based on data center configuration and CFD studies to model heat flows. As part of future work, CFD neural surrogates can automate parameter generation for a custom data center configuration.

References

J. R. Vázquez-Canteli, J. Kämpf, G. Henze, Z. Nagy, Citylearn v1.0: An openai gym environment for demand response with deep reinforcement learning, in: Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 356–357. URL: `https://doi.org/10.1145/3360322.3360998`. doi:10.1145/3360322.3360998.

P. Scharnhorst, B. Schubnel, C. Fernández Bandera, J. Salom, P. Taddeo, M. Boegli, T. Gorecki, Y. Stauffer, A. Peppas, C. Politi, Energym: A building model library for controller benchmarking, Applied Sciences 11 (2021). URL: `https://www.mdpi.com/2076-3417/11/8/3518`. doi:10.3390/app11083518.

T. Moriyama, G. D. Magistris, M. Tatsubori, T. Pham, A. Munawar, R. Tachibana, Reinforcement learning testbed for power-consumption optimization, CoRR abs/1808.10427 (2018). URL: `http://arxiv.org/abs/1808.10427`. arXiv:1808.10427.

D. B. Crawley, L. K. Lawrie, C. O. Pedersen, F. C. Winkelmann, Energy plus: energy simulation program, ASHRAE journal 42 (2000) 49–56.

B. Acun, B. Lee, F. Kazhamiaka, K. Maeng, U. Gupta, M. Chakkaravarthy, D. Brooks, C.-J. Wu, Carbon explorer: A holistic framework for designing carbon aware datacenters, in: Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, ACM, 2023. URL: `https://doi.org/10.1145/3575693.3575754`. doi:10.1145/3575693.3575754.

A. Radovanović, R. Koningstein, I. Schneider, B. Chen, A. Duarte, B. Roy, D. Xiao, M. Haridasan, P. Hung, N. Care, S. Talukdar, E. Mullen, K. Smith, M. Cottman, W. Cirne, Carbon-aware computing for datacenters, IEEE Transactions on Power Systems 38 (2023) 1270–1280. URL: `https://doi.org/10.1109/tpwrs.2022.3173250`. doi:10.1109/tpwrs.2022.3173250.

A. Naug, A. Guillen, R. Luna Gutiérrez, V. Gundecha, S. Ghorbanpour, L. Dheeraj Kashyap, D. Markovikj, L. Krause, S. Mousavi, A. R. Babu, S. Sarkar, Pydcm: Custom data center models with reinforcement learning for sustainability, in: Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 232–235. URL: `https://doi.org/10.1145/3600100.3623732`. doi:10.1145/3600100.3623732.

K. Sun, N. Luo, X. Luo, T. Hong, Prototype energy models for data centers, Energy and Buildings 231 (2021) 110603.

A. Naug, A. Guillen, R. L. Gutiérrez, V. Gundecha, D. Markovikj, L. D. Kashyap, L. Krause, S. Ghorbanpour, S. Mousavi, A. R. Babu, et al., Pydcm: Custom data center models with reinforcement learning for sustainability, arXiv preprint arXiv:2310.03906 (2023).

C. S. de Witt, T. Gupta, D. Makoviichuk, V. Makoviychuk, P. H. S. Torr, M. Sun, S. Whiteson, Is independent learning all you need in the starcraft multi-agent challenge?, 2020. arXiv:2011.09533.

R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, I. Mordatch, Multi-agent actor-critic for mixed cooperative-competitive environments, 2017. URL: `https://arxiv.org/abs/1706.02275`. doi:10.48550/ARXIV.1706.02275.