

2024학년도 동계 이화글로벌프론티어 프로그램 탐사보고서

HVAC 시스템에서 맞춤형 환경의 중요성: 강화학습을 중심으로

The Importance of Customized Environment
Simulation in HVAC Systems: Focusing on
Reinforcement Learning



Bring Green

목 차

I. 서론	3
1. 탐사 및 연구 주제	
2. 연구 배경	
3. 연구 필요성	
4. 탐사 방법	
II. 본론	8
1. 선행 연구 분석	
2. 국내 탐사	
3. 해외 탐사	
(1) 현지 탐사의 필요성	
(2) 현지 탐사의 기대효과	
(3) 인터뷰 내용 정리	
4. 탐사의 의의	
5. 연구 과정	
6. 연구 결과	
III. 결론	30
1. 탐사 및 연구 요약	
2. 연구의 의의	
3. 연구의 한계 및 제언	
IV. 참고문헌	33
[부록 1] 인터뷰 전문	35
[부록 2] 연구 과정	65

I. 서론

1. 탐사 및 연구 주제

HVAC 시스템에서 맞춤형 환경의 중요성: 강화학습을 중심으로

The Importance of Customized Environment Simulation in HVAC Systems:
Focusing on Reinforcement Learning

2. 연구 배경

건물은 전 세계 이산화탄소 배출의 40%를 차지하며 전체 온실가스의 25%를 차지한다. 이를 통해 건물 부문의 에너지 소비와 탄소 배출이 지구 온난화 및 환경 문제에 직접적인 영향을 미친다는 것을 알 수 있다. 또한, 2022년 기준 서울특별시 에너지 정보에 따르면 이화여자대학교는 연간 온실가스 배출량이 31,704.4tCO₂eq으로, 서울시 소재 대학 캠퍼스 중 5위를 기록하고 있었다. 이는 캠퍼스의 에너지 소비가 상당히 높은 수준이며, 이화여자대학교가 탄소 중립 목표를 달성을 위해 적극적인 대책을 마련해야 함을 시사하고 있다.

이에 따라, 본 팀은 이화여자대학교 캠퍼스의 건물 운영 및 에너지 절감 방안을 모색하며, HVAC(Heating, Ventilation, and Air Conditioning) 시스템이 현대 건물의 에너지 소비와 탄소 배출에 가장 큰 영향을 미치는 핵심 요소라는 점에 주목하였다. 특히, HVAC 시스템이 실내 환경의 쾌적성을 결정하는 주요 요인이며, 동시에 에너지 절감과 효율적인 운영을 위해 필수적인 기술이라는 점을 고려할 때, 본 연구에서 심층적으로 탐구할 가치가 충분하다고 판단하였다.

HVAC 시스템의 효율적인 운영은 건물의 에너지 소비 절감과 온실가스 배출 저감에 중요한 역할을 한다. 그러나 기존의 HVAC 시스템 제어 방식에는 여러 가지 한계가 존재한다. 먼저, 현재 대부분의 시스템은 규칙 기반 제어 또는 모델 예측 제어 방식을 활용하여 설정값을 조정하는 방식이다. 규칙 기반 제어 방식은 특정한 온도 설정 값과 일정한 작동 규칙을 기반으로 HVAC 시스템을 운영하는 방식으로, 환경 변화에 대한 유연성이 부족하여 에너지 낭비가 발생할 가능성이 높다. 예를 들어, 건물 내 사용자가 적은 시간에도 동일한 난방 및 냉방이 유지될 수 있으며, 이는 불필요한 에너지 소비로 이어진다. 모델 예측 제어 방식은 데이터 기반으로 보다 정교한 HVAC 제어를 가능하게 하지만, 초기 모델 구축 및 운영 과정에서 막대한 계산 비용과 시간이 소요된다는 단점이 있다. 또한, 실시간으로 변하는 건물 내부 환경을 즉각적으로 반영하기 어려운 한계가 존재한다.

이러한 문제를 해결하기 위해 최근 강화학습(Reinforcement Learning, RL)기반의 HVAC 제어 방식이 주목받고 있다. 강화학습 기반의 HVAC 제어 시스템은 실시

간 데이터를 학습하면서, 다양한 환경 조건에 적응할 수 있는 최적의 제어 정책을 스스로 학습하는 능력을 갖추고 있다. 이를 통해 에너지 소비를 줄이면서도 실내 환경의 쾌적성을 유지할 수 있는 HVAC 시스템을 구축하는 것이 가능하다.

한편, 강화학습을 HVAC 시스템에 효과적으로 적용하기 위해서는 맞춤형 환경 구축이 필수적이다. 건물 내부의 온도 및 공기 질이 창외 방향, 온도 조절 가능 기계의 개수 등 다양한 요소에 영향을 받기때문에 실제 건물과 유사한 환경에서 학습된 모델을 사용해야 해당 건물에 적절한 HVAC 제어가 가능해진다. 따라서, 일반적인 제어 모델을 넘어 개별 건물의 특성에 맞춰 구축된 환경 기반으로 모델을 학습시킬 수 있어야 한다.

이번 탐사에서 본 팀은 가장 대표적인 건물 에너지 시뮬레이터인 EnergyPlus를 이용해 커스텀 환경을 구축하는 방식과 구축된 맞춤형 환경에서 강화학습으로 학습된 모델이 HVAC 최적화에 어떤 영향을 미칠지 분석하는 것을 목표로 한다. 특히, 에너지 소비 패턴, 사용자 선호 온도 등을 반영한 동적 제어 모델이 실제 운영 환경에서 어떻게 적용될 수 있는지에 집중해 탐사를 진행할 예정이다. 이를 통해 강화학습 기반 HVAC 시스템을 구축할 방법과 해당 방법이 미래의 캠퍼스 건물에서 에너지 효율성과 사용자 편의를 동시에 달성할 수 있는 핵심 기술이 될 가능성을 평가한 최종 보고서를 작성할 예정이다.

3. 연구 필요성

이화여자대학교는 2021년 창립 135주년 기념식에서 발표한 ‘Ewha Vision 2030+’ 발전 계획을 통해 ‘지속가능 사회를 선도하는 창의·혁신 플랫폼’이라는 비전을 제시하였다. 이는 단순한 연구를 넘어 실질적인 지속 가능성 정책과 교육을 대학 운영에 적극 반영하겠다는 의지를 의미한다.

본 팀은 이를 실현하기 위해 2029년 준공 예정인 이화 웨스트 캠퍼스를 포함한 미래 캠퍼스 설계 과정에서 지속 가능한 캠퍼스 구축과 효율적인 에너지 관리 방안을 도입할 필요가 있다고 판단하였다. 지속가능한 캠퍼스 구축 논의는 단순한 기술적 해결책을 넘어, 대학이 사회에 미치는 영향과 역할을 재조명하는 계기가 되어야 한다.

그러나 현재 국내 대학들은 지속가능한 캠퍼스를 운영하고자 하는 의지는 있으나, 실질적인 시스템 구축과 체계적 운영에는 한계를 보이고 있다. 이대학보 미디어 부에서 제작한 다큐멘터리 “내가 그린, 그린 캠퍼스 | 이대학보 해외취재 특집”에 따르면, 한국의 그린캠퍼스 평가 체계는 여전히 미흡하며, 해외 대학과 비교했을 때 지속가능성 시스템이 체계적으로 운영되지 못하는 실정이다. 하버드 대학교는 지속

가능성 개념을 대학 운영에 처음 도입한 사례로, 전담 부서를 세분화하고 독자적인 홈페이지를 운영하며 체계적인 지속가능성 시스템을 구축하고 있다. 반면, 국내 대학들은 이러한 시스템을 효과적으로 운영할 수 있는 평가 기준이 부족하며, 구성원들의 체감도 또한 낮아 실질적인 효과가 제한적이다.

하지만 지속 가능성 시스템이 체감되지 않는다고 해서 논의가 멈춰서는 안 된다. 이화여대 에코과학부 석좌교수 최재천 교수는 “대학은 연구와 교육을 수행하는 기관으로서, ESG 경영을 기업에 맡길 것이 아니라 연구와 기준을 마련하고 교육해야 한다”고 강조했다. 대학이 지속가능한 캠퍼스를 연구·운영하며 이를 교육한다면, 장기적으로 사회와 기업의 변화를 주도할 수 있다. HVAC(Heating, Ventilation, and Air Conditioning) 시스템 최적화는 이러한 지속가능성 실현을 위한 핵심적인 기술적 요소이며, 강화학습을 활용한 효율적 제어 방안을 통해 보다 혁신적인 해결책을 마련할 수 있다. 이에 따라, 다음과 같은 측면에서 논의가 필요하다.

첫째, 기존 모델 예측 제어(MPC)의 한계를 극복할 필요성이 있다. 현재 HVAC 최적화 연구는 주로 물리 모델을 기반으로 하는 모델 예측 제어(Model Predictive Control, MPC)에 의존하고 있다. 그러나 MPC는 환경 변화에 대한 적응력이 부족하고, 초기 모델링 과정에서 높은 연산 비용이 소요되는 문제가 있다. 반면, 강화학습 기반 최적화 모델은 실시간 학습과 적응을 통해 변화하는 환경에서도 효과적으로 작동할 수 있어, 보다 정교하고 효율적인 제어가 가능하다.

둘째, 맞춤형 환경 제어를 통해 사용자 경험을 개선할 필요성이 있다. 기존 HVAC 시스템은 일반적으로 사전 설정된 기준에 따라 작동하며, 개별 공간의 특성과 사용자의 선호도를 세밀하게 반영하지 못하는 한계가 있다. 하지만 강화학습을 적용하면 특정 공간의 사용 패턴, 실내 환경 데이터, 사용자 피드백 등을 실시간으로 반영하여 최적화된 제어 정책을 생성할 수 있다. 이를 통해 에너지 효율성과 실내 쾌적성을 동시에 확보하는 맞춤형 환경 조성이 가능해진다.

셋째, 강화학습 기반 HVAC 최적화를 위한 다양한 접근법을 탐색할 필요성이 있다. 실질적인 활용을 위해서는 새로운 환경에서도 빠르게 적응할 수 있는 능력과 안정적인 성능 확보가 중요하다. 이를 위해 다양한 방법론을 탐색하고 실험하는 과정이 필요하다. 예를 들어, 메타 학습(Meta-Learning)과 전이 학습(Transfer Learning)을 결합하면, 학습된 정책을 새로운 건물이나 다양한 기후 조건에 빠르게 적용할 수 있어 실용성을 극대화할 수 있다. 이처럼 다양한 시도를 통해 특정 도메인에 최적화된 정책을 도출하고, 실질적인 에너지 절감 및 쾌적한 실내 환경 조성에 기여할 수 있는 방향을 모색할 수 있을 것이다.

이러한 논의를 바탕으로, 강화학습을 활용한 HVAC 최적화 연구는 지속가능한

미래를 선도하는 이화여대의 비전에 부합하며, 에너지 효율성과 실내 환경 개선을 통해 지속가능한 캠퍼스 구축에 기여할 수 있을 것이다.

4. 탐사 방법

본 팀은 연구 방법을 5단계로 나누어 진행할 계획이다.

첫째, 국내 문헌 조사를 통해 HVAC 시스템 제어와 관련하여 대한민국 내에서 이루어지고 있는 기존 연구들을 분석할 것이다. 이를 통해 국내에서 진행된 연구 동향과 현재 적용되고 있는 기술들을 파악하여 본 연구의 기초 이론을 다질 것이다. 특히, 본 팀의 연구는 HVAC 시스템 제어에 인공지능 강화학습 알고리즘을 사용하여 건물의 에너지 효율화를 목표로 하기 때문에, 컴퓨터공학과 건축학 두 분야에서 여러 논문을 읽고 전문가를 찾는 것이 중요하다. 따라서 인공지능학과와 건축학과 교수님들의 연구를 바탕으로 HVAC 시스템에 관한 이론적 토대를 마련하고, 국내 전문가들과의 인터뷰를 위한 심도 깊은 조사를 진행할 예정이다.

둘째, 국내 전문가 인터뷰를 진행할 것이다. 이 단계에서는 이화여자대학교 인공지능학과 김수경 교수님과 이화여자대학교 건축학과 박지현 교수님을 지도 교수님으로 삼아 이들과의 논의를 통해 본 연구의 목표와 방향성을 구체적으로 설정할 것이다. 이를 통해 HVAC 시스템 제어 및 인공지능 기술 적용에 대한 구체적인 접근과 현실적인 연구 가능성에 대해 도움을 받고 교수님들의 지도 하에 연구 설계를 발전시킬 것이다.

셋째, 해외 논문 리뷰를 통해 HVAC 시스템 제어와 관련된 최신 연구와 기술들을 분석할 예정이다. 해외에서 활용되고 있는 다양한 알고리즘과 설계된 HVAC 시스템들을 연구하고, 국내에서의 적용 가능성에 대해서 비교하며 이러한 해외에서의 적용 사례들이 한국에 어떻게 도입될 수 있을지에 대해 논의할 것이다.

넷째, 해외 전문가 인터뷰를 통해 HVAC 시스템 최적화와 관련된 실무적 경험을 가진 전문가들에게 실질적인 조언을 얻을 것이다. 이를 통해 연구의 구체적인 방향을 설정하고, 보다 효율적인 접근법을 모색할 수 있다. 또한, 국내외 연구 환경의 차이를 이해함으로써 연구의 폭을 넓히고 실질적인 적용 가능성을 높일 수 있을 것이다. 전문가 인터뷰는 본 연구의 이론적 기반과 실무적 조언을 결합하여 연구의 신뢰성을 높이는 중요한 과정이 될 것이다.

다섯째, 국내외 전문가들의 조언과 기존 연구 분석을 바탕으로 본격적인 개발 프로젝트를 진행할 것이다. 해외 탐사에 앞서 EnergyPlus와 Sinergym을 활용해 에너지 소비 시뮬레이션 환경을 구축하고, 탐사 동안 전문가들의 조언을 반영하여 이를 지속적으로 개선해 나갈 계획이다. 탐사 이후에는 구축된 시뮬레이션 환경에서

적절한 알고리즘을 적용하고, 하이퍼파라미터 튜닝을 통해 최적의 모델을 도출할 것이다. 또한, fixed control 대비 알고리즘의 성능을 평가하여 실무 적용 가능성을 검토하고, 연구의 실효성을 극대화할 방안을 모색할 것이다. 이를 통해 HVAC 시스템 최적화를 위한 실질적인 솔루션을 마련하는 초석을 다지고자 한다.

II. 본론

1. 선행 연구 분석

(1) 기존 논의 요약 및 정리

HVAC(Heating, Ventilation, and Air Conditioning) 시스템 최적화는 건물의 에너지 절감을 위해 필수적인 연구 분야이며, 최근 인공지능(AI) 기술이 발전하면서 머신러닝과 강화학습(Deep Reinforcement Learning, DRL)을 적용한 연구가 활발하게 이루어지고 있다.

최근 연구들은 AI가 HVAC 시스템의 운영 효율성을 높이고, 에너지 절감 및 탄소 배출 저감에 기여할 가능성을 분석하는 데 초점을 맞추고 있다. Ding et al. (2024)은 기존의 물리 모델 기반 최적화 기법과 비교했을 때, 머신러닝과 AI 모델이 더욱 유연하고 실시간 적응력이 뛰어나기 때문에 보다 정교한 에너지 관리가 가능하다고 주장하였다. 특히, AI 기반 최적화는 기존의 단순한 Rule-based Control보다 효과적으로 HVAC 시스템을 조정할 수 있으며, 운영 비용 절감과 에너지 효율성 향상 측면에서 실질적인 이점을 제공할 수 있음을 제시하였다.

이와 같은 연구들을 바탕으로 기존 HVAC 최적화 연구들은 크게 ① MPC(Model Predictive Control) 기반 최적화 연구, ② DRL(Deep Reinforcement Learning) 기반 HVAC 최적화 연구, 그리고 ③ HVAC 시스템 모델링 및 제어 접근성 향상 연구로 구분할 수 있다.

MPC는 물리 모델을 기반으로 HVAC 시스템의 최적 제어를 수행하는 방식으로, 높은 신뢰도를 가지지만 계산량이 많고 새로운 환경에서 모델링 비용이 높다는 한계가 있다. 이러한 한계를 극복하기 위해 RL(Reinforcement Learning)이 제안되었다. RL은 MPC에 비해 낮은 컴퓨팅 성능을 요구하는 반면, 히터 하나와 같은 작은 요소의 변화에도 민감해 재학습을 요구한다는 단점을 가지고 있었다. 이후 DRL을 활용한 연구가 등장했으며, DRL은 데이터 기반 학습을 통해 환경 적응성을 높일 수 있는 가능성을 보였지만, 초기 학습 과정에서 비효율성이 발생하고 일반화가 어렵다는 한계가 존재한다.

또한, HVAC 최적화를 위해서는 정확한 모델링과 데이터 연계가 필수적이거나, 기존 연구에서 사용된 모델링 기법들은 특정 환경에서만 유효하여 확장성이 제한적이었다. 이에 따라, HVAC 제어의 접근성을 높이기 위한 표준화된 방법론과 데이터 활용 연구들이 병행되고 있다.

① MPC(Model Predictive Control) 기반 최적화 연구

MPC는 미래의 에너지 소비를 예측하고 최적의 HVAC 제어 전략을 수립하는

방식으로, 물리 모델을 기반으로 하기 때문에 높은 신뢰도를 가지며, 안정적인 HVAC 제어를 가능하게 한다는 장점이 있다.

- *Systems Approach to Energy Efficient Building Operation: Case Studies and Lessons Learned in a University Campus*

본 연구에서는 대학 캠퍼스의 중앙 냉각 시스템(chilled water plant system)에 MPC를 적용하여 에너지 절감 효과를 분석하였다. 연구 결과, 기존 휴리스틱(Heuristic) 제어 방식 대비 MPC 적용 시 약 5%의 에너지 절감 효과가 확인되었으며, 냉각수 온도, 냉각 타워 온도, 칠러 가동 스케줄을 최적화함으로써 HVAC 운영의 효율성을 개선할 수 있음을 입증하였다. 더불어, 건물의 실제 점유 수준을 측정하고 이를 기반으로 환기 수요(Demand Controlled Ventilation, DCV)를 동적으로 조정한다. 이는 에너지 절감 효과를 극대화하고 다양한 외부 조건에 가변적인 환기 전략을 적용할 수 있도록 이끌었다. 그러나, MPC의 계산량이 많고, 새로운 환경에서는 추가적인 모델링 비용이 발생한다는 한계가 존재한다.

② DRL(Deep Reinforcement Learning) 기반 HVAC 최적화 연구

MPC가 고정된 물리 모델을 기반으로 최적화를 수행하는 방식이라면, DRL은 강화학습을 통해 환경과 상호작용하며 최적의 HVAC 제어 정책을 학습하는 방식이다. 특히, 복잡한 시스템을 모델링할 필요 없이, 다양한 환경에서 직접 학습할 수 있다는 점에서 유연성이 높다.

- *An experimental evaluation of Deep Reinforcement Learning algorithms for HVAC control*

해당 연구에서는 PPO, TD3, DDPG 등 다양한 DRL 알고리즘을 EnergyPlus 시뮬레이터를 활용해 비교하여 HVAC 제어 성능을 분석하였다. 연구 결과, DRL 기반 HVAC 최적화는 Rule-based Control보다 에너지 절감 효과가 뛰어나며, 자동화된 HVAC 제어를 가능하게 한다는 점에서 효율적으로 HVAC 시스템을 관리할 수 있다는 높은 가능성을 보였다. 특히, SAC과 TD3 알고리즘이 가장 높은 성능을 보였으며, 최대 15%의 에너지 소비 절감 효과를 나타냈다.

- *Deep Reinforcement Learning for Building HVAC Control*

해당 연구에서는 DRL을 활용하여 실시간 HVAC 운영을 최적화하고자 하였으며, 기존 방식보다 뛰어난 제어 성능을 보였다. 연구 결과, DRL 기반 제어 방식은 기존 Rule-based Control 방식과 비교했을 때 최대 70%의 에너지 비용 절감 효과를 나타냈다. 특히, DRL 알고리즘이 충분히 학습된 후에는 온도 범위를 유지하면서도 에너지를 효율적으로 사용하는 것이 가능했다. 그러나, 초기 학습

과정에서는 비효율적인 제어가 발생할 가능성이 높았으며, 새로운 환경에서 적응하려면 재학습이 필요하다는 한계가 확인되었다.

- *Multi-source transfer learning method for enhancing the deployment of deep reinforcement learning in multi-zone building HVAC control*

해당 연구에서는 다중 소스 전이 학습(Multi-source Transfer Learning, MTL)을 적용하여 DRL 모델이 새로운 환경에서도 효율적으로 학습할 수 있도록 설계하였다. MTL은 여러 개의 소스 도메인에서 학습한 지식을 활용해 새로운 타겟 도메인에서 모델의 성능을 향상시키는 기법이다. 전이 학습 기법을 적용한 DRL 모델은 기존 모델보다 빠르게 적응할 수 있었으며, 기본 DRL 모델 대비 최대 20%의 학습 시간 단축, 1.43%~3.12%의 에너지 소비 감소, 최대 14.32%의 실내 온도 편차 감소를 기록하였다. 이는 더 적은 비용으로 모델의 성능을 높일 수 있는 방법을 제시했다는 데에 의의가 있으며, 이후 메타 학습 또한 적용해볼 수 있다는 인사이트를 제안했다.

③ HVAC 시스템 모델링 및 제어 접근성 향상 연구

HVAC 시스템 최적화 연구는 DRL 및 MPC 기반의 제어 기술뿐만 아니라, 보다 많은 연구자와 산업 관계자가 쉽게 접근할 수 있도록 모델링 및 제어 접근성을 향상하는 방향으로도 발전하고 있다. 이러한 연구들은 AI 기반 최적화 가능성을 분석하는 연구, 표준화된 벤치마크 모델을 구축하여 확장성을 높이는 연구, 그리고 HVAC 시스템에 대한 이해도를 높이고 모델링을 시각화하는 연구로 구분할 수 있다.

- *Development of whole building energy models for detailed energy insights of a large office building with green certification rating in Singapore*

해당 연구에서는 싱가포르의 대형 오피스 건물을 대상으로 전체 에너지 모델을 구축하고, 이를 활용하여 건물 에너지 소비 패턴을 분석하였다. 특히, 미국 DOE(Department of Energy)에서 제공하는 표준 빌딩 벤치마크 모델을 기반으로, 싱가포르의 기후 조건과 에너지 정책을 반영하여 건물 모델을 맞춤형으로 조정하는 과정이 포함되었다. 이를 통해, 특정 환경 및 에너지 사용 기준을 충족시키기 위한 섬세한 건물 모델링이 가능함을 보여주었으며, 향후 다른 국가에서도 유사한 방식으로 건물 에너지 성능을 평가하고 개선하는 데 활용할 수 있는 가능성을 제시하였다. 연구에서는 또한, 건물 내 주요 에너지 소비 패턴을 정량적으로 분석하여, HVAC 시스템 및 기타 전력 사용(plug loads)이 차지하는 비율을 평가

하였다. 이를 바탕으로, 특정 운영 전략이 건물의 에너지 효율성 및 지속 가능성에 미치는 영향을 보다 정확하게 예측할 수 있도록 하였다.

- *Field Demonstration of the Brick Ontology to Scale up the Deployment of ASHRAE Guideline 36 Control Sequences*

이 연구에서는 HVAC 제어 시퀀스를 표준화하고, 이를 보다 직관적으로 모델링할 수 있도록 Brick Ontology 기반의 프레임워크를 도입하였다. 기존 건물 자동화 시스템(BAS)의 불규칙한 명명 규칙으로 인해 HVAC 제어 전략 적용 시 발생하는 매핑 작업의 비효율성을 해결하고자, 센서 및 액추에이터 등의 제어 요소를 일관된 데이터 구조로 표현하는 방법을 제안하였다. 또한, ASHRAE Guideline 36(G36) 기반의 고급 HVAC 제어 시퀀스를 실증 평가하여 표준화된 제어 전략이 실제 에너지 절감 효과를 가질 수 있음을 확인하였다. 연구에서는 BACnet 및 CDL(Control Description Language)과 같은 표준 프로토콜을 활용하여 HVAC 시스템의 데이터 매핑 및 제어 전략 적용을 최적화하는 방안을 제시하였다.

결과적으로, Brick Ontology를 활용한 표준화된 데이터 구조가 HVAC 시스템 운영을 보다 쉽게 이해하고 유연하게 적용하는 데 기여할 수 있음을 보여주었으며, 향후 건물 에너지 최적화 연구에서 확장성이 높은 제어 시스템 구축에 중요한 역할을 할 것으로 기대된다.

(2) 기존 연구의 한계점

기존 연구들은 HVAC 최적화를 위한 중요한 방향성을 제시하며, 다양한 접근법을 통해 에너지 절감과 운영 효율성을 높이는 데 기여해왔다. MPC 기반 연구는 신뢰성이 높은 제어 기법을 제공하며, DRL 기반 연구는 보다 유연하고 환경 적응력이 뛰어난 접근법을 제시하였다. 또한, 일부 연구에서는 건물 모델링과 데이터 활용의 표준화를 시도하며 HVAC 최적화의 실무 적용 가능성을 높이려는 노력이 이루어졌다.

그러나 이러한 연구들이 실질적인 HVAC 최적화 솔루션으로 자리 잡기 위해서는 여전히 해결해야 할 과제들이 존재한다. 기존 연구들은 특정 환경에서 높은 성능을 보일 수 있도록 설계되었으나, 다양한 건물 유형과 기후 조건에 적용하는데 어려움이 있으며, 특히 대학 캠퍼스와 같은 복합적 환경을 충분히 고려하지 못했다. 이를 바탕으로, 기존 연구들이 가진 주요 한계점은 다음과 같이 정리할 수 있다.

① 환경 일반화의 어려움

HVAC 최적화를 위한 기존 연구들은 특정 환경에 최적화된 모델을 중심으로 진행되어 왔다. MPC(Model Predictive Control) 기반 접근법은 물리 모델을 바탕으로 높은 신뢰도의 제어 전략을 제공하지만, 특정 건물과 기후 조건에 맞춘 정교한 모델링이 필요하다. 새로운 건물이나 기후 지역에 적용하려면, 기존의 물리 모델을 다시 조정해야 하며, 이는 높은 계산 비용과 추가적인 데이터 수집을 필요로 한다. 따라서 MPC는 환경이 변경될 때마다 상당한 모델링 비용이 발생하며, 다양한 건물 유형에 적용하는 것이 어렵다는 한계를 지닌다.

한편, DRL(Deep Reinforcement Learning) 기반 연구는 데이터를 학습하여 HVAC 시스템을 최적화하는 접근법을 사용하므로, MPC보다 유연한 적용이 가능하다. 그러나, DRL 역시 특정 환경에서 학습된 모델을 새로운 환경에 그대로 적용하기 어려운 문제가 있다. 건물의 구조, 점유율 패턴, 외부 기후 조건 등이 달라지면, 기존에 학습한 모델이 예상보다 낮은 성능을 보일 수 있으며, 이를 해결하기 위해 새로운 환경에서 추가적인 학습이 필요하다.

때문에 DRL 모델이 학습된 환경을 벗어나면 재학습 과정이 필요하며, 초기 데이터가 부족한 경우에는 학습 속도가 저하되거나 최적화 성능이 저하될 가능성이 있다. 기존 연구에서는 DRL의 일반화 문제를 해결하기 위해 전이 학습(Transfer Learning) 기법을 적용한 연구가 이루어지고 있지만, 건물 유형이나 기후 조건이 크게 다를 경우 여전히 성능 저하가 발생할 수 있다.

이러한 한계로 인해 기존 연구들은 특정 유형의 건물과 특정 기후 조건을 중심으로 실험이 진행되었으며, 다양한 환경에서도 효과적으로 적용할 수 있는 일반화된 프레임워크가 부족한 상황이다. 실무 적용을 고려한다면, DRL 기반 HVAC 최적화 모델이 실제 다양한 건물 환경에서도 효과적으로 작동할 수 있도록 일반화하는 연구가 필수적이다.

② 대학 캠퍼스를 고려한 연구 부족

기존 연구는 대부분 상업용 건물이나 산업용 건물을 대상으로 진행되었으며, 대학 캠퍼스와 같은 복합적인 환경을 충분히 반영하지 못했다. 일반적인 상업용 건물은 사무실과 회의실 등의 공간으로 구성되어 비교적 일정한 점유율과 에너지 소비 패턴을 보이지만, 캠퍼스 건물은 강의실, 연구실, 도서관, 행정 건물 등 다양한 용도로 활용된다. 이로 인해, 점유율과 에너지 소비 패턴이 일정하지 않고, 시간대별 변동성이 크며, HVAC 최적화가 더욱 복잡해지는 특성이 있다.

대학 캠퍼스에서의 에너지 소비 규모는 상당히 크다. 국토교통부에서 발간한 건물 에너지 사용량 통계 자료에 따르면, 2021년 기준으로 교육시설 전체의 에너

지 소비 원단위는 평균 $7.88 \text{ kgoE/m}^2\cdot\text{yr}$ 인 반면, 대학시설의 에너지 소비 원단위는 $10.76 \text{ kgoE/m}^2\cdot\text{yr}$ 로, 단위면적당 에너지 사용 강도가 약 37% 더 높게 나타났다. 이는 연구실, 데이터 서버룸, 대형 강의실 등의 전력 및 냉난방 수요가 많기 때문이다. 또한, 대학 캠퍼스에서 HVAC 시스템이 차지하는 에너지 소비 비중은 전체 소비량의 약 40~60%에 달하며, 연구 및 실험 시설이 포함된 캠퍼스의 경우가 비율이 더 높아질 수 있다.

특히, 대학 캠퍼스는 공간마다 사용 되는 시간이 다르다는 특이성을 가지고 있다. 이를테면, 강의실은 오전 8시부터 오후 6시까지 주로 이용되지만, 연구실과 기숙사 건물은 심야까지 24시간 운영되어 난방 및 냉방이 지속적으로 가동될 가능성이 높다. 이로 인해, 단순한 시간 기반 제어 방식(Fixed Control)으로는 효율적인 에너지 관리가 어려우며, 점유율 변화와 실시간 환경 조건을 반영할 수 있는 고도화된 제어 기법이 필수적이다. 그러나 기존 연구에서는 이러한 복합적인 점유 패턴과 다양한 건물 유형을 고려한 최적화 연구가 부족하며, 현재의 DRL 및 MPC 기반 HVAC 최적화 연구들이 상업용 빌딩을 중심으로 진행되었다는 한계가 있다.

(3) 본 연구의 기여

기존 연구들은 특정 환경에 최적화된 HVAC 제어 모델을 중심으로 진행되어 일반화 가능성이 부족하고, 대학 캠퍼스와 같은 복합적인 건물 유형을 충분히 반영하지 못했다. 이러한 한계를 극복하기 위해, 본 연구에서는 메타 전이 학습(Meta-Transfer Learning)을 활용한 DRL 기반 HVAC 최적화 프레임워크를 제안한다. 이를 통해 다양한 환경에서도 신속하게 적응할 수 있는 강화학습 기반 HVAC 제어 시스템을 구축하고, 실제 환경에서의 적용 가능성을 높이는 데 기여하고자 한다.

우선, 다양한 건물과 기후 조건에서도 학습할 수 있도록 메타 전이 학습을 적용하여 특정 건물이나 기후에만 최적화되지 않고, 새로운 환경에서도 빠르게 적응할 수 있는 DRL 모델을 개발하고자 한다. 기존의 DRL 기반 연구들은 개별적인 환경에서 최적화된 모델을 학습하는 방식이었기 때문에, 새로운 건물이나 기후 조건에서 다시 학습해야 하는 비효율성이 존재했다. 반면, 메타 전이 학습을 적용하면, 여러 환경에서 공통적인 특성을 학습하여 새로운 환경에서도 신속하게 최적화가 가능하다. 이를 통해, 기존 DRL 모델의 일반화 가능성을 높이고 재학습에 필요한 시간과 비용을 절감할 수 있다.

또한, 대학 캠퍼스를 대상으로 한 HVAC 최적화 연구를 진행하여 기존 연구의 한계를 보완하고자 한다. 일반적인 상업용 건물과 달리, 대학 캠퍼스는 강의실, 연구실, 도서관, 기숙사 등 다양한 공간으로 구성되어 있으며, 점유율 패턴이 복잡하게 변동하는 특성을 갖는다. 이러한 특성을 반영하지 않은 기존 연구들은 캠퍼스 건물에서의 적용 가능성이 제한적일 수밖에 없었다. 본 연구에서는 실제 대학 캠퍼스 데이터를 기반으로 실험을 수행하고, 점유율 및 에너지 소비 패턴을 고려한 맞춤형 HVAC 제어 모델을 구축함으로써, 대학 캠퍼스 환경에서도 효과적으로 적용할 수 있는 최적화 방안을 제시하고자 한다.

결과적으로, 본 연구는 메타 전이 학습을 활용하여 DRL 기반 HVAC 최적화 모델의 일반화 가능성을 향상시키고, 대학 캠퍼스와 같은 복잡한 환경에서도 효과적으로 적용할 수 있도록 연구를 확장하는 데 기여할 것이다. 이를 통해, 여러 환경에 대한 적응성을 높이고 다양한 건물 유형과 기후 조건에서 에너지 절감 효과를 극대화할 수 있는 방향을 제시하고자 한다.

2. 국내 탐사

본 연구팀은 국내 탐사를 통해 HVAC 시스템 최적화 및 AI 기반 자동화 기술의 실무 적용 가능성을 탐색하고자 하였다. 이를 위해 인공지능, 건축학, 산업공학, 에너지 관리 등 다양한 분야의 전문가들과 논의를 거치며, 강화학습 기반 HVAC 최적화의 연구 방향과 실무 적용 시 고려해야 할 사항들을 심층적으로 검토하였다. 특히, AI 기반 HVAC 제어의 필요성, 캠퍼스 환경에서의 에너지 절감 전략, 실무 적용 시 발생하는 한계와 해결 방안을 중심으로 탐사를 수행하였다.

먼저, 강화학습을 활용한 HVAC 최적화의 필요성과 연구 방향을 분석하기 위해 이화여자대학교 인공지능학과 김수경 교수 및 연세대학교 산업공학과 윤현수 교수와 면담을 진행하였다. 여기서 기존의 Model Predictive Control(MPC) 방식은 높은 연산량과 추가적인 모델링 비용이 요구되는 한계가 있으며, 이를 보완하기 위해 강화학습이 적절한 대안이 될 수 있다는 점이 강조되었다. 강화학습을 적용할 경우, 기존 MPC 기반 HVAC 제어보다 유연한 방식으로 작동할 수 있으며, 연산 비용 절감 효과를 기대할 수 있다. 또한, 실험을 위한 시뮬레이션 환경으로 BOPTTEST를 활용하는 방안이 제안되었고, 강화학습 모델 설계를 위해 RL Baselines Zoo를 참고하는 방법이 논의되었다. HVAC 시스템이 실시간으로 변화하는 외부 환경과 점유율 변화를 반영해야 하므로, 지도학습보다 보상 기반 학습이 가능한 강화학습이 더욱 적절한 방식이라는 점도 확인하였다. 추가적으로, 연구팀이 참고할 만한 연구자로 Jin Wen이 소개되었으며, 그의 연구 논문을 분석하

여 강화학습 기반 HVAC 최적화 모델의 최신 연구 동향을 파악하는 기회를 얻을 수 있었다.

한편, 건축적 측면에서 HVAC 최적화의 방향성을 모색하기 위해 이화여자대학교 건축학과 박지현 교수와 인터뷰를 하게되었다. HVAC 시스템의 최적화를 위해서는 건물의 물리적 특성과 자연환경을 반영하는 것이 필수적이며, 창문 개방, 바람의 흐름, 햇빛의 방향과 같은 요소가 실내 온도 조절에 미치는 영향을 종합적으로 고려해야 한다는 점이 강조되었다. 최근 건축 설계의 흐름은 특정 용도의 공간을 설계하는 방식에서 벗어나, 다목적 활용이 가능한 공간을 조성하는 방향으로 변화하고 있으며, 이에 따라 HVAC 시스템 역시 보다 유연한 제어 방식이 요구됨을 확인할 수 있었다. 또한, AI 기반 HVAC 최적화 기술이 해외에서는 활발히 도입되고 있으나, 국내에서는 아직 실질적인 적용 사례가 부족하므로, 탐사를 통해 해외 사례를 연구하고 로컬라이징(Localization)하는 과정이 중요하다는 점이 논의되었다.

또한, 강화학습 기반 HVAC 최적화의 실무 적용 가능성과 한계를 평가하기 위해 한국산업기술시험원에 자문을 구하였다. AI 기반 HVAC 최적화 모델이 연구 단계에서는 높은 성과를 보일 수 있지만, 실무 환경에서는 여러 현실적인 제약이 존재한다는 점이 지적되었다. 기존의 강화학습 방식은 사계절을 고려한 학습이 필요하며, 개별 HVAC 시스템마다 반복적인 훈련을 요구해 높은 연산량이 필요하다는 문제가 확인되었다. 이는 특정 환경에서만 최적화될 가능성이 높고, 실시간 제어 시 과도한 연산 자원이 요구될 수 있음을 시사한다. 또한, 기후 변화에 대한 반영이 충분하지 않을 경우 예상치 못한 성능 저하가 발생할 우려가 있으며, 이를 해결하기 위해 학습 데이터의 다양성을 확보하고 연산 부담을 줄이는 방안이 필요하다는 점이 논의되었다.

마지막으로, 본 연구팀은 해외 탐사와의 연계성을 고려하여 Lawrence Berkeley National Laboratory(LBNL)의 Mary Ann Piette 박사와 사전 협의를 거치며, 해외 연구 사례를 국내 환경에 적용할 때 고려해야 할 요소들을 분석하였다. 건물의 점유율 모델링(Occupancy Modeling)이 HVAC 최적화에서 중요한 변수로 작용하며, 단순한 기계적 제어가 아닌 건물 사용자들의 행동 패턴을 반영한 최적화가 필수적이라는 점이 확인되었다. 또한, 미국과 한국의 업무 환경이나 실내 온도 설정 방식이 다를 수 있기 때문에, 해외 연구 결과를 그대로 적용하는 것이 아니라 지역별 문화적 차이를 반영한 맞춤형 최적화가 필요하다는 점이 강조되었다.

정리하면, 국내 탐사를 통해 HVAC 최적화 연구가 다양한 분야에서 활발히 이

루어지고 있음을 확인하였으며, 실무 적용을 위해 해결해야 할 여러 과제들이 도출되었다. 기존의 HVAC 최적화 연구는 특정 환경에서 최적화된 모델을 개발하는데 초점을 맞추었으나, 실제 운영 환경에서의 적용 가능성을 고려할 필요가 있으며, 점유율 변화, 건축 설계, 컴퓨팅 자원 문제 등을 종합적으로 반영해야 함을 확인하였다. 이에 따라 본 연구팀은 강화학습 기반 HVAC 최적화 모델의 맞춤형 환경 구축이 중요하다는 점을 인식하였으며, 이를 실무 환경에 효과적으로 적용하기 위한 시뮬레이션과 검증 과정이 필요하다는 결론을 도출하였다.

3. 해외 탐사

(1) 현지 탐사의 필요성

본 팀은 국내 사전 탐사를 통해 이화여자대학교 건축학과 박지현 교수님과 2차례 자문을 진행하였다. 박지현 교수님과의 면담에서, 인공지능을 활용한 HVAC 시스템 제어가 해외에서는 활발히 연구·적용되고 있지만, 국내에서는 활용 사례가 적고 관련 기관을 찾기 어려워 자문에 한계가 있음을 확인하였다. 이에 해외 현지 조사를 통해 HVAC 시스템을 국가, 지역, 용도별로 로컬라이징하여 쾌적 범위(comfort range)를 충족시키는 방식을 심도 있게 이해할 필요가 있다고 판단하였다.

또한, 한국산업기술시험원(KTL)과의 사전 탐사에서는 사계절을 모두 고려한 단순 강화학습 방식이 막대한 컴퓨팅 파워를 요구하여 현실적으로 구축이 어려울 수 있다는 조언을 받았다. 이를 바탕으로 연구 방향을 딥러닝과 강화학습을 결합한 DRL(심층 강화학습) 방식으로 수정하였다. 해외 전문가와의 인터뷰를 통해 DRL 프레임워크 선택에 대한 조언과 함께, 기존 연구에서 MPC, RL, DRL 등 다양한 알고리즘을 선택한 이유를 물어 구체적인 연구 방향을 설정할 기회를 얻을 수 있을 것이다.

한편, 본 팀은 사전 탐사 과정에서 이화여자대학교 시설관리팀에게 캠퍼스 전체의 전력 소비량과 학교가 지불하고 있는 전기세 등 건물 관리와 관련된 구체적인 정보 중 공개 가능한 것들이 있는지 요청하였으나, 해당 정보들 모두 비밀 사항으로 분류되어 제공받을 수 없었다. 이는 국내에서 건물 관리와 관련된 정보의 접근성이 매우 제한적이며 연구를 위해 필요한 데이터를 확보하기가 매우 어렵다는 현실적인 한계를 느끼게 했다. 반면, 해외에서는 건물 관리팀이 에너지 소비이터를 연구 목적으로 적극적으로 제공하거나 연구자들과 협의하여 제한적으로 활용할 수 있게 하는 사례가 많다는 점을 사전조사를 통해 확인하였다. 특히 미국의 경우 건물마다 ASHRAE(미국냉난방공조학회) 기준을 기반으로 건물의 용도,

크기, 지역적 조건에 맞게 HVAC 시스템을 설계하고 효율적으로 운영하는 사례가 많다는 점도 흥미로웠다. 해외 탐사를 통해 이러한 데이터의 제공 방식, 협의 절차 등에 대해서 구체적으로 조사할 수 있을 것이다. 이러한 정보는 국내에서도 HVAC 연구 및 에너지 효율화 방안을 추진할 때 참고할 만한 주요 사례가 될 수 있다고 생각하였다.

본 팀은 모두 소프트웨어 전공자로 구성되어 있어 하드웨어 지식이 부족하지만, HVAC 시스템의 구성 및 구현에는 냉각기(Chiller)와 같은 하드웨어에 대한 깊은 이해가 필수적임을 논문 리뷰를 통해 알게 되었다. 이에 사전 학습 내용을 바탕으로, 실제 현장에서 하드웨어를 어떻게 활용하는지 전문가들에게 질문하고, 부족한 하드웨어 지식을 보완함으로써 소프트웨어와 하드웨어를 통합적으로 고려한 연구를 수행할 계획이다.

또한, 본 팀은 해외 탐사에 앞서 EnergyPlus와 Sinergym을 활용한 물리적 HVAC 모델 및 에너지 소비 시뮬레이션 환경을 구축하고, 다양한 캠퍼스 건물과 날씨 데이터를 반영한 시뮬레이션을 통해 PPO 알고리즘 코드를 실행하는 사전 프로젝트를 진행하였다. 그러나 EnergyPlus에 커스텀 빌딩을 추가하는 과정에서 어려움을 겪었고, 현재는 기본 제공되는 "office medium" 모델을 이용해 시뮬레이션을 실행 중이다. 해외 전문가 인터뷰를 통해 커스텀 빌딩 적용 과정에서 발생한 문제 해결 방안을 조언받고, 기본 모델을 활용한 연구가 현실성을 가질 수 있는지도 확인할 예정이다.

이처럼, 현지조사와 전문가 인터뷰는 연구 과정에서의 실질적 문제를 해결하고, 보다 완성도 높은 연구를 수행하기 위한 필수적인 과정으로 작용할 것이다.

(2) 현지 탐사의 기대효과

현지조사 및 전문가 인터뷰는 본 연구를 위해 필수적인 과정으로, 다양한 기대효과를 가져올 수 있다.

첫째, HVAC 시스템 제어가 활발히 연구되고 적용되고 있는 미국 샌프란시스코로의 탐사는 국내에서 얻기 어려운 구체적인 정보를 얻을 기회를 제공하며, 이를 통해 보다 심도 깊은 이해를 할 수 있을 것이다. 또한, 국내에서는 HVAC 시스템 제어가 적용되는 사례가 적기 때문에 해외 사례를 통해 그 적용 가능성과 장점을 실질적으로 파악할 수 있을 것이다.

둘째, 연구의 방향성을 명확히 설정하는 기회가 될 것이다. 사전 탐사에서 받은 조언을 바탕으로 연구 방향을 DRL 방식으로 수정하였다. 이에 맞게 사전 국내 탐사에서는 AI로 HVAC 시스템 제어를 연구하는 전문가들을 위주로 만나 조언을

구했었다. 반면, 해외 탐사에서는 주로 MPC를 사용하는 전문가들을 만날 예정이다. 전문가들이 MPC를 주로 사용한 이유를 묻고, 본 팀이 선택한 DRL 프레임워크에 대한 조언을 얻는 과정은 연구 방향을 더욱 구체화하고, 효율적이며 현실적인 HVAC 최적화 방법을 제시할 수 있는 기회를 제공할 것이다. 또한, DRL 방식의 장점과 실현 가능성에 대해서도 전문가들의 직접적인 피드백을 받을 수 있을 것이다.

셋째, 국내와 해외의 HVAC 연구 및 건물 효율화 방식의 차이를 명확히 이해하고, 발전 가능성을 탐색하는 기회가 될 것이다. 사전 조사를 통해 국내에서는 연구 목적으로 건물 데이터를 제공받기 어려운 현실을 알게 되었으나, UC Berkeley나 Purdue 대학교와 같은 해외 대학에서는 연구 목적에 맞게 제한적으로 건물 데이터를 제공하는 시스템이 운영되고 있다는 점을 확인하였다. 현지 조사를 통해 국내와 해외의 건물 관리 데이터 제공 방식을 비교하면서, HVAC 연구 환경의 차이점을 실질적으로 파악할 수 있을 것이다.

넷째, HVAC 제어에 사용되는 하드웨어에 대해서도 깊은 이해를 얻을 기회가 될 것이다. 본 팀은 소프트웨어 전공자로 이루어져 있기 때문에 하드웨어에 대한 지식이 부족하지만, 현지 전문가와의 인터뷰를 통해 냉각기(Chiller)와 같은 하드웨어의 실제 활용 사례를 배우고, 이를 연구에 어떻게 적용할 수 있을지 구체적으로 논의할 수 있을 것이다.

다섯째, EnergyPlus 시뮬레이션에서 발생한 커스텀 빌딩 적용 문제를 해외 전문가들과의 인터뷰를 통해 해결할 수 있을 것이다. 커스텀 빌딩 적용 과정에서의 기술적 난제를 해결하고, 시뮬레이션 개발 중 발생한 문제들을 전문가들과 논의하면서, 향후 연구에 필요한 데이터와 환경 설정에 대한 실질적인 조언을 받을 수 있을 것이다.

(3) 인터뷰 내용 정리

① HVAC 시스템 최적화 및 연구 방향 설정

건물의 에너지 소비 양상은 기후에 따라 달라지기 때문에 외부 환경에 따라 냉방 혹은 난방 기기의 효율을 다르게 접근하는 것이 필요하다. 이외에도 건물 부하도 기후의 영향을 받기 때문에 연간 에너지 소비량의 요소로 함께 고려해야 한다. 또한, 지역별 에너지 소비 패턴과 정책적 요인, 문화적 차이를 고려해 기존 연구들 분석 및 실험 설계를 진행해야 한다. 예를 들어, 한국과 일본에서는 점심시간 동안 전력 소비가 급격히 감소하는 경향이 있지만, 미국에서는 이러한 현상이 나타나지 않는다. 따라서 미국에서 개발된 AI 모델을 한국 건물

환경에 적용할 때 단순한 기술적 이식이 아닌, 시간대별 소비 패턴과 같은 문화적 요소까지 반영해야 한다.

이렇게 분석한 정보들을 활용한 에너지 절약 방법에는 크게 두가지 양상이 있는데 기타 전력 사용을 줄이는 방법과 HVAC 제어 최적화를 하는 방법이다. 대부분에 연구에서는 HVAC 제어를 최적화하는 것을 채택하고 있지만 기타 전력이 전체 에너지 소비의 40%를 차지하기에 HVAC 보다 더 큰 영향을 미칠 수 있다. 컴퓨터, 디스플레이, 엘리베이터, Wi-Fi 라우터 등은 점유율과 관계없이 24시간 작동하는 경우가 많으며, 이로 인해 불필요한 에너지 낭비가 발생한다. 특히, 대기 전력 소비 문제가 심각한데, 건물이 사용되지 않는 시간에도 각종 기기가 대기 상태에서 지속적으로 전력을 소비하는 현상이 빈번하게 발생하고 있지만 현재 에너지 관리 시스템은 주로 HVAC과 조명 제어에 집중하고 있어 플러그 필드까지 포함하는 포괄적 에너지 최적화 전략이 필요하다. 이 방법을 활용하면 건물 전체의 포괄적인 에너지 최적화가 가능하다. 그런데, 이 연구에서 중요한 것은 공실 상태를 직접 감지하는 센서 혹은 점유율 데이터를 기반으로 간접적으로 예측하는 방법을 이용해 공실 상태를 파악하는 것이다. 그러나 본 팀은 해당 연구를 위한 데이터를 구할 수 없으며, 캠퍼스의 특성상 센서 없는 사람의 존재 유무를 예측을 특정하기엔 예외 상황이 많기에 HVAC 최적화가 더 실현 가능성이 높다고 판단했다.

HVAC 제어를 통한 에너지 절약은 크게 MPC 모델을 이용하는 방법과 강화학습을 이용하는 방법이 있다. MPC 연구에서 크게 거론되는 문제는 컴퓨팅 파워의 한계이다. 강화학습을 이용한 방법론은 이런 문제를 극복하기 위한 방법으로 제시되었으나, 경우에 따라 강화학습 역시 높은 연산량을 요구할 수 있어 연구의 최적화 문제는 단순하고 효율적으로 표현할 필요가 있다. 또한, 강화학습은 데이터와 시뮬레이션 환경이 필수적이며, 학습 속도가 느리거나 안정성이 부족할 가능성이 있기에 적절한 알고리즘 선택과 하이퍼파라미터 튜닝이 중요하다. 실제로도 건물 운영에서는 신뢰성과 확장 가능성이 중요한 요소이기에 강화학습과 같이 실험적으로 검증된 방법보다 검증된 방법인 MPC가 우선적으로 고려된다. 이처럼 두 연구 방법 모두 높은 컴퓨팅 자원을 요구하는 만큼 연구 효율성을 높이기 위해서는 분산 학습이나 클라우드 컴퓨팅 자원을 활용하는 방안 고려가 필요하며 강화학습이 실제 환경에 적용될 수 있도록 연구의 신뢰성을 높이고 실제 건물 기반의 연구가 진행되어야 한다.

강화학습을 바로 실제 건물에 적용하기 위해서는 건물 설비 단계부터 학습 데이터 선별까지 많은 과정이 필요하기에 본 팀은 HVAC 시스템 연구에서 커

스텀 환경을 조성하고자 하였다. 커스텀 환경의 조건은 강의실, 실험실, 사무실 등 다양한 공간이 존재하는 캠퍼스의 특성을 반영해야 한다. 또한 실험을 진행하는 과정에서 냉난방 장치를 활용하여 쾌적한 환경을 유지하며 에너지를 절약할 수 있어야 한다. 이를 위해 에너지 플러스의 “office medium” 모델을 활용해 연구를 진행했으나, 이 역시 제공된 모델이라는 문제로 인해 5분 단위의 샘플링을 활용한 학습만 가능하다. 이는 HVAC 시스템의 짧은 시간 단위 동역학을 정확하게 반영하지 못한다는 한계가 존재한다.

또한 성능을 높이기 위한 방법으로 실시간 데이터 수집을 활용하는 것이 있다. 이는 시간 단위 데이터 아카이브 시스템을 설정하고 데이터 활용도를 극대화 하기 위한 표준화된 데이터 형식 구축 및 데이터 라벨링을 명명해 활용해 데이터를 수집하는 방법이다. 이러한 실시간 데이터 수집 및 처리는 신뢰성이 높은 데이터기에 학습 과정에서 핵심적인 역할을 한다. 그러나 한국에서는 건물 관리 시스템이 설치되어 있음에도 데이터를 충분히 활용하지 않는 경우가 많아 현장 전문가와의 협력이 필요한 경우가 많다는 상황적 어려움이 존재한다. 따라서 실제 환경에 적용한다면 실시간 데이터를 수집하고 처리하는 논리적 설계 과정이 추가적으로 필요할 것이다.

② 건물 설계 및 HVAC 적용을 위한 고려 사항

HVAC 최적화 연구를 진행하기 위해서는 건물 설계와 설계된 건물에 HVAC 적용을 하기 위한 여러 변수를 고려해야 한다. 건물 설계는 연구 목적에 따라 그 설계가 달라진다. 특정 건물만을 위한 프로젝트를 진행한다면 설계 도면, 기계 장비 도면, 건물 내 비치 정보 등 구체적인 데이터를 최대한 많이 반영하는 것이 중요하지만, 더 일반화 된 건물을 목표로 프로젝트를 진행한다면 설계 방식은 달라진다. 이때 건물 설계는 더 일반화된 모델을 제작해야 한다. 예를 들어 특정 지역에 건설되는 건물을 목표로 프로젝트를 진행한다면 해당 지역에서 일반적으로 건설되는 건물 유형을 통계적으로 분석할 필요성이 있다. 본 팀의 프로젝트 역시 이화의 캠퍼스 건물에 적합한 일반적 건물을 HVAC를 통해 제어하는 것이므로 이화여자대학교의 캠퍼스 건물의 특징을 분석했다. 프로젝트를 진행하며 고려한 건물의 설계 요소는 건물의 크기, 면적, 내부 공간의 개수, 공간의 용도 다양성을 고려하여 건물 설계를 진행했다.

다음으로 캠퍼스 건물을 분석했다면 해당 건물을 에너지 플러스의 시뮬레이션 모델로 구체화하는 작업이 필요하다. 많은 논문에서 ASHARE의 일반화된 건물 모델을 이용해 진행하였기에 이와 유사한 대형 건물로 초점을 맞춰 AI 모

델에 접목시켜 실험을 진행했다. 그러나, 효율성을 더 고려한다면 공간 점유율의 측면에서 장비 밀도나 조명등의 세부 조건을 확인하고 이를 바탕으로 에너지 시뮬레이션을 통해 얼마나 많은 에너지를 절약할 수 있는지 추가적인 연구가 필요하다 판단하였다.

본 팀의 실험의 궁극적 목표는 아직 지어지지 않은 미완공 건물에서의 HVAC 시스템 제어이기에 이를 위해서는 에너지 플러스와 같은 시뮬레이션 도구가 반드시 필요하다. 이를 통해 실제 기상 데이터를 반영한 가상 시나리오로 건물의 에너지 효율성을 분석하는 것이 가능하다. 그러나 이는 운영도구로 사용하기에는 한계가 있기에 건물이 완공된 이후 단열, 창문, 냉각 장치 등의 주요 요소를 조정하면서 추가적인 시뮬레이션을 진행해야 한다. 이를 통해 실질적인 HVAC 운영 최적화를 이루고 연구의 실용성을 높일 수 있다.

또한, HVAC 제어 효율성을 더 높일 수 있는 방안으로 Brick 스키마의 이용을 고려해볼 수 있다. Brick 스키마는 건물 내 요소들을 특정한 범주로 구분하여 계층적으로 정리하는 방식이다. 이는 모든 연구자가 동일한 수준의 정보를 가지고 있지 않기에 일반적인 정보를 입력한 후에도 더 구체적인 정보를 추가할 수 있도록 한다. 해당 스키마를 이용해 다양한 건물을 하나의 방법으로 제어할 수 있어 특성이 다른 건물의 제어를 모델의 적응력을 향상시키는 방법이 아닌 제어 방식의 일반화를 통해 해결 가능하다. 이는 기존 연구에서 집중한 단순히 모델의 적응력을 높이는 것에서 벗어나 제어 환경의 일반화를 진행할 수 있는 새로운 해결책이다. 해당 방법을 이용한다면 이화의 캠퍼스의 기존 건물과 새로운 건물 역시 일반화된 브릭 온톨로지 설정을 통해 모델 개발에서의 비용을 절약할 수 있을 것이다.

③ AI 및 강화학습 기반 HVAC 제어 기술 연구

강화학습(Deep Reinforcement Learning, DRL)은 HVAC 최적화뿐만 아니라 다양한 산업 분야에서 자동화와 최적화 문제를 해결하는 데 활용되고 있다. 이번 Osaro 인터뷰를 통해 물류 및 로봇 비전 시스템에서 강화학습이 어떻게 적용되고 있는지를 살펴볼 수 있었으며, 이를 통해 강화학습 기반 시스템이 실무에서 어떤 방식으로 운영되고, 어떤 한계를 가지는지에 대한 중요한 시사점을 얻을 수 있었다.

Osaro는 물류 자동화 솔루션을 개발하는 기업으로, 대표적인 시스템으로는 Bagging System, Kitting System, Piece Picking System, Depalletizing System이 있다. 이들은 비정형적인 환경에서도 안정적인 작업 수행이 가능하도

록 설계되었으며, 머신러닝과 강화학습을 활용하여 로봇이 환경 변화에 적응할 수 있도록 개발되었다. 특히, Osaro AutoModel™ 기술을 통해 데이터 수집, 학습, 평가, 배포 과정을 자동화하여 새로운 데이터를 빠르게 반영할 수 있도록 하였으며, Osaro SightWorks™는 로봇 비전 시스템의 신뢰성을 높이기 위해 센서와 머신러닝을 결합한 기술을 적용하고 있다. 이러한 기술들은 물류 환경에서 충분한 학습 시간을 확보하기 어려운 현실을 고려하여, 실시간 데이터 학습을 통해 빠르게 적용될 수 있도록 설계되었다.

강화학습 기반 자동화 시스템의 중요한 특징 중 하나는 새로운 환경이나 아 이템이 추가되더라도 운영을 멈추지 않고 지속적으로 학습할 수 있다는 점이다. 일반적으로 전이 학습(Transfer Learning)을 활용하는 방식이 아닌, 로봇이 환경을 관찰하며 데이터를 수집하고 최적의 동작을 학습하는 접근 방식을 적용하고 있다. 만약 새로운 데이터가 기존 학습 범위를 벗어날 경우, 성능 저하가 발생할 수 있지만, 시스템이 완전히 멈추는 것이 아니라 일정 수준의 성공률을 유지하면서 점진적으로 학습을 진행한다. 또한, Osaro는 최근 Zero-shot Learning 기법을 도입하여, 추가 학습 데이터 없이도 기존 데이터를 활용해 성능을 개선하는 연구를 진행하고 있다. 이러한 접근 방식은 HVAC 시스템에도 적용될 수 있는 가능성이 있으며, 지속적인 운영과 학습을 동시에 진행하는 방안에 대한 연구가 필요함을 시사한다.

Osaro의 시뮬레이션 및 학습 환경은 Unity 기반 가상 환경에서 운영되며, 이를 통해 실제 운영 환경에서 발생할 수 있는 문제들을 사전에 탐색하고 해결할 수 있다. 예를 들어, Kitting 시스템에서는 부품이 과다하게 쌓이는 오버플로우(Overflow) 문제를 시뮬레이션을 통해 검출하고, 이를 해결하기 위한 최적의 배치 전략을 도출할 수 있다. 이는 HVAC 최적화 연구에서도 단순히 EnergyPlus와 같은 시뮬레이션을 활용하는 것뿐만 아니라, 실제 운영 데이터를 반영하여 실무 적용성을 높이는 것이 중요함을 다시 한번 확인할 수 있는 부분이었다.

이번 인터뷰를 통해 강화학습 기반 시스템이 실무에서 효과적으로 적용되기 위해서는 단순한 알고리즘 개발뿐만 아니라, 물리적 환경의 제약과 실제 데이터를 반영한 검증 과정이 필수적이라는 점을 알 수 있었다. Osaro의 사례에서도 머신러닝만으로는 100% 완벽한 동작을 보장할 수 없으며, 센서와 같은 물리적 요소를 함께 활용하여 시스템의 신뢰성을 높이고 인간의 개입을 최소화하는 것이 핵심 요소임이 강조되었다. 예를 들어, Pick & Place Solution에서는 로봇이 물건을 놓쳤을 경우 레이저 센서와 무게 측정을 통해 정확한 포장이 이루어

졌는지 확인하는 기능을 추가하여 신뢰성을 높이고 있다. 이는 강화학습을 HVAC 시스템에 적용할 때도 유사한 방식으로 실제 환경에서의 변수를 반영하고, 센서 데이터를 활용하여 신뢰성을 확보하는 전략이 필요하다는 점을 시사한다.

비록 Osaro에서 다른 강화학습 연구가 HVAC 최적화와 직접적인 연관이 있는 것은 아니었지만, 이를 통해 강화학습 기반 시스템이 실무에서 적용되고 운영되는 방식, 그리고 학습 환경을 어떻게 구성해야 하는지에 대한 중요한 교훈을 얻을 수 있었다. 특히, HVAC 제어에 강화학습을 적용할 경우, 단순한 시뮬레이션 기반 연구만으로는 실질적인 성능 개선을 보장하기 어렵고, 실시간 운영과 검증을 동시에 고려한 시스템 설계가 필요하다는 점을 인식하게 되었다. 또한, 단순히 성능 향상을 목표로 하기보다는, 기존 제어 시스템(Fixed Control)과의 차별점을 명확하게 제시하고, 비용 절감 효과와 실무 적용 가능성을 증명하는 것이 필수적이라는 점을 다시 한번 확인할 수 있었다.

④ HVAC 연구 및 시뮬레이션의 한계와 개선 방안

강화학습을 HVAC 최적화에 적용하기 위해서는 실무 환경과의 괴리를 최소화하는 것이 중요하다. 그러나 현재 HVAC 최적화 연구에서는 알고리즘 성능 향상에만 초점을 맞추는 경우가 많아, 실질적인 제어 효과를 거두지 못하는 한계가 존재한다. 이에 따라, 단순한 시뮬레이션 기반 접근이 아니라 물리적 시스템과 실제 운영 데이터를 반영한 연구가 필수적이라는 점이 강조된다.

HVAC 최적화 연구에서는 주로 강화학습(Deep Reinforcement Learning, DRL)이나 최적화 알고리즘의 성능 향상에 초점을 맞추는 경우가 많지만, UC Berkeley의 박지원 박사님과의 인터뷰를 통해 단순한 알고리즘 개선만으로는 실제 환경에서 효과적인 제어가 어렵다는 점을 깨닫게 되었다. 특히, HVAC 시스템의 하드웨어적 요소, 예를 들어 칠러(Chiller)와 같은 물리적 기기의 특성을 고려하지 않으면, 이론적으로 최적화된 알고리즘이라도 실제 환경에서는 비효율적인 제어가 발생할 가능성이 크다는 점이 강조되었다. 예를 들어, 칠러 시스템의 부하 분배(Load Distribution)와 성능 계수(COP: Coefficient of Performance)를 충분히 반영하지 않은 최적화 모델은 실질적인 에너지 절감 효과를 제공하지 못할 수 있다. 따라서, 효과적인 HVAC 최적화를 위해서는 알고리즘 개선뿐만 아니라 물리적 시스템의 제약까지 함께 고려해야 함이 중요하다.

또한, HVAC 최적화 연구는 주로 EnergyPlus와 같은 시뮬레이션 환경에서

진행되지만, 이러한 시뮬레이션 결과를 그대로 신뢰하기 어렵다는 문제가 존재한다. 박지원 박사님은 실제 건물 환경과의 차이로 인해 시뮬레이션에서 도출된 최적화 결과가 실제 운영 환경과 다르게 나타날 가능성이 크다는 점을 지적하였다. 시뮬레이션에서는 현실적인 변수를 단순화하는 경우가 많아, 예를 들어 사람들의 발열량을 일정하다고 가정하거나 창문 개방 여부를 고려하지 않는 등의 문제로 인해 실제 환경과의 괴리가 발생할 수 있다. 또한, 실제 점유율 변화와 날씨 데이터를 충분히 반영하지 않을 경우, 시뮬레이션에서는 효과적인 것으로 보이는 제어 전략이 실무 적용 시 기대만큼의 성능을 발휘하지 못할 가능성이 높다. 따라서, 시뮬레이션 결과만을 기반으로 연구를 진행하는 것이 아니라, 실제 데이터를 활용한 검증(Validation) 과정을 포함하는 것이 필수적이며, 이를 보완하기 위한 연구가 필요함이 강조되었다. HVAC 최적화를 위해서는 단순한 시뮬레이션 기반 접근에서 벗어나, 실제 건물 운영 데이터를 반영하여 모델을 보정하고 최적화하는 과정이 반드시 포함되어야 한다.

한편, 현재 실무에서 가장 널리 사용되는 HVAC 제어 방식은 Fixed Control 방식으로, 특히 한국에서는 약 95%의 건물이 이를 채택하고 있다. Fixed Control은 특정 시간에 HVAC를 단순히 켜고 끄는 방식으로, 실내 환경의 변화를 반영하지 못하는 한계가 있음에도 불구하고 유지보수가 용이하고 안정성이 높다는 점에서 여전히 선호되고 있다. DRL이나 MPC 기반의 최적화 방식이 활발히 연구되고 있음에도 불구하고, 실제 현장에서는 기존 방식에서 쉽게 전환되지 않는 이유도 여기에 있다. 박지원 박사님은 새로운 제어 알고리즘이 실무에서 적용되기 위해서는 Fixed Control 대비 명확한 성능 개선과 비용 절감 효과를 입증해야 하며, 동시에 유지보수의 용이성도 확보되어야 한다고 강조하였다. 따라서, AI 기반 HVAC 최적화 모델을 개발할 때 단순히 성능을 향상시키는 것만이 아니라, 실무 적용 가능성을 고려한 현실적인 접근 방식이 필요하다는 점이 논의되었다.

4. 탐사의 의의

이번 탐사를 통해, HVAC(Heating, Ventilation, and Air Conditioning) 시스템에서의 에너지 최적화는 지속 가능한 스마트 빌딩 구축의 핵심 과제로써 우리나라 뿐만 아니라 미국에서도 많은 주목을 받고 있는 분야임을 실감하였다. 또한, LBNL에서 HVAC 시스템 및 에너지 효율적인 건물 기술을 위해 구축한 실험용 건물인 FLEXLAB과 UC Berkeley 건축학과 학생들에게 제공되는 실제 공간 실험 환경을 직접 확인하며, 미국의 연구소와 대학교가 시뮬레이션 중심의 접근법을 넘어 실험

적 환경을 적극적으로 활용하고 있음을 목격하였다. 이는 HVAC를 포함한 건물 에너지 효율화 연구에서 실제 데이터를 기반으로 한 실험 환경 구축이 가장 중요하다는 것을 체감하게 하였다. 또한, 이러한 측면에서 미국 우리나라보다 앞서 있다는 점을 확인할 수 있었다.

특히 Mary씨와 Donghun씨와의 인터뷰에서 HVAC 시스템의 최적화 과정에서 모델링과 데이터 준비와 같은 물리적 제어 요소가 전체의 90%를 차지한다는 답변을 들었다. 이러한 통찰은 기존 주제인 “캠퍼스 건물의 HVAC 제어 최적화 속도를 향상시키기 위한 적응형 메타-전이 DRL 프레임워크”가 지나치게 이상적이고 실현 가능성이 떨어진다는 문제를 인식하게 하였다. 메타 학습과 전이 학습을 적용한 강화학습 알고리즘 연구에 집중하는 것 보다, EnergyPlus에서 대한민국 서울 지역과 같은 구체적인 물리적 조건을 충족하는 커스마이징에 집중하는 것이 필요하다는 결론에 도달하였다.

이에 따라 우리는 연구 주제를 “HVAC 시스템에서 맞춤형 환경의 중요성: 강화학습을 중심으로”로 변경하였다. 이번 주제는 강화학습 알고리즘 자체의 설계보다는 HVAC 시스템 최적화를 위한 데이터 준비와 환경 시뮬레이션의 중요성을 조명하는 데에 초점을 맞추고 있다. LBNL과 UC Berkely 등 여러 기관 탐사를 통해 알게 되었던 정교한 환경 모델링의 중요성을 반영하여 수정한 것이다. 인터뷰를 통해 강화학습이 성공적으로 작동하려면, 학습 환경이 실제 시스템의 동적 특성을 충실히 반영하는 것이 필요하며 이를 위해서는 정교한 환경 모델링에 따른 시뮬레이션 구축이 선행되어야 함을 확실하게 느꼈다. 또한, 캠퍼스와 같은 대규모 복합 건물에서는 다양한 공간이 혼합되어 있다는 특성과 사용 패턴에 따라 최적화 조건이 상이하기 때문에, 환경 커스터마이징이 더 중요하다는 점도 알게 되었다.

이번 탐사를 통해 HVAC 시스템 최적화 연구에서 데이터 준비와 맞춤형 환경 구축의 본질적 중요성을 깊이 깨달았으며, 기존 연구 방향성을 재고하는 계기가 되었다. 이러한 통찰은 본 팀이 연구 방향을 올바르게 설정하는 데 크게 기여하였으며, 탐사를 통해 만난 전문가들과의 지속적인 소통은 앞으로의 연구 진행에 있어 중요한 발판이 될 것이다.

5. 연구 과정

본 연구에서는 HVAC(Heating, Ventilation, and Air Conditioning) 시스템의 최적화를 위해 강화학습(Deep Reinforcement Learning, DRL)을 적용하고, 기존의 Fixed Control 방식과 비교하여 그 효과를 분석하고자 한다. 실험은 EnergyPlus 기반 시뮬레이션 환경에서 진행되며, Sinergym 프레임워크를 활용하

여 HVAC 시스템의 제어 성능을 평가한다.

우선, 실험을 위한 맞춤형 환경을 구축하였다. EnergyPlus의 표준 모델 중 ASHRAE901_OfficeLarge_STD2019_Denver 빌딩 모델을 기반으로 실험을 진행하였으며, 한국의 기후 데이터를 반영하기 위해 KOR_Inchon.471120_IWEC 날씨 데이터를 적용하였다. 이를 통해, 실제 캠퍼스 건물과 유사한 환경에서의 시뮬레이션을 가능하게 하였다.

실험의 주요 비교 대상은 기존의 Fixed Control 방식과 강화학습 기반의 DDPG(Deep Deterministic Policy Gradient) 알고리즘이다. Fixed Control 방식은 정해진 온도 설정값을 기반으로 일정한 스케줄에 따라 HVAC 시스템을 제어하는 전통적인 방식이며, DDPG는 실시간 학습을 통해 최적의 HVAC 운용 정책을 탐색하는 방식이다.

DDPG 모델은 환경과 상호작용하면서 에이전트가 최적의 행동을 학습할 수 있도록 구성되었다. 이를 위해 Actor-Critic 구조를 기반으로, Actor 네트워크는 현재 상태에서 최적의 액션을 결정하고, Critic 네트워크는 해당 액션의 가치를 평가하는 역할을 수행한다. 학습 과정에서는 에너지 소비를 최소화하면서 실내 온도 쾌적성을 유지하는 방향으로 보상 함수를 설계하여, DRL이 학습 목표를 효과적으로 달성할 수 있도록 하였다.

또한, 모델의 성능을 향상시키기 위해 Optuna를 활용한 하이퍼파라미터 최적화를 수행하였다. 이를 통해, 학습 속도(learning rate), 배치 크기(batch size), 액션 노이즈(action noise) 등의 최적 값을 탐색하고 적용하였다.

각 알고리즘의 성능 비교는 15개의 에피소드를 실행한 후, 에너지 소비량, 온도 안정성, 보상 함수의 변화를 분석하는 방식으로 이루어졌다. 실험 결과는 전력 소비량 및 에너지 절감율 비교를 통해 정량적으로 평가되며, 이를 기반으로 강화학습을 적용한 HVAC 최적화의 실효성을 검증하고자 한다.

6. 연구 결과

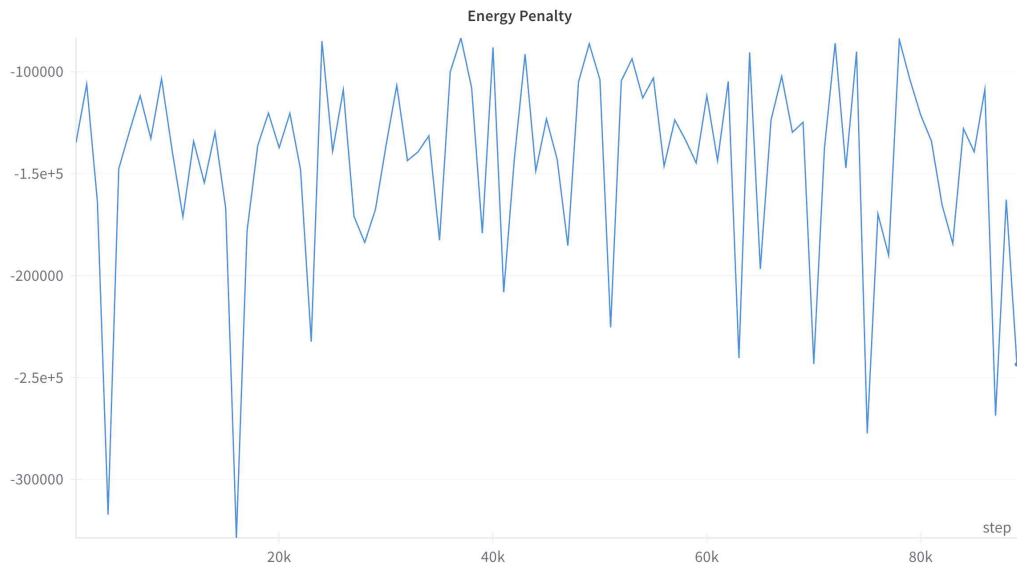
수치적 비교를 통해 DDPG 알고리즘과 Fixed-Control 알고리즘을 각각 사용했을 때의 에너지 소비량에 대해 알아보하고자 한다.

(1) DDPG에서의 Energy Penalty와 Comport Penalty 값 확인

Energy penalty와 Comport penalty 값을 통해 DDPG 에이전트가 HVAC 시스템을 어떻게 학습했는지 해석할 수 있다.

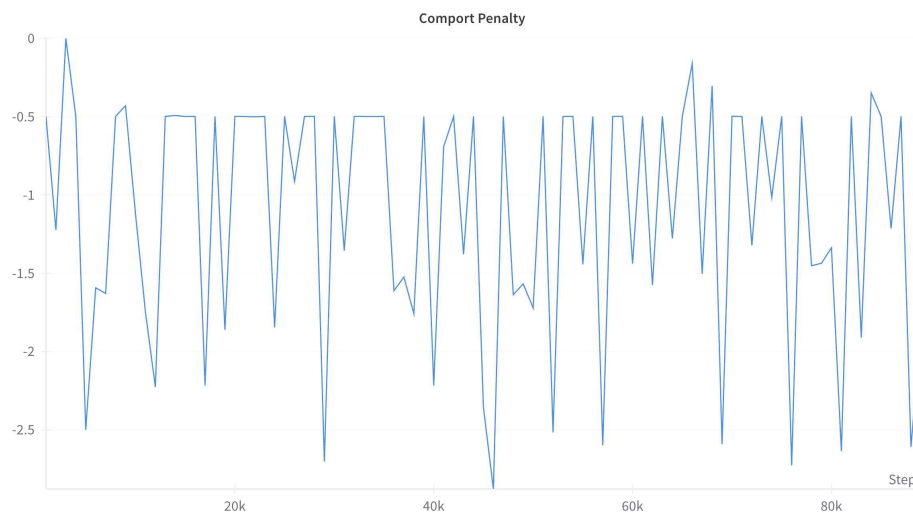
Energy penalty는 HVAC 시스템이 에너지를 소비하는 정도를 패널티로 변환한 값이다. Energy Penalty의 경우 값이 전체적으로 -100,000에서 -300,000 사

이에 분포되어 있고, 학습이 진행됨에 따라 변화가 많지만 일정한 범위 내에서 움직임이 보인다.



[그림 11] Energy Penalty 그래프

Comfort penalty는 실내 온도가 설정된 범위를 벗어날 때 부과되는 패널티이다. 값이 -2.5에서 0 사이에 분포함을 확인할 수 있다. 현재는 패널티가 지속적으로 발생하고 있음을 의미하며, 에이전트가 실내 온도 조절을 위해 더욱 적절한 조절이 필요한을 의미한다.



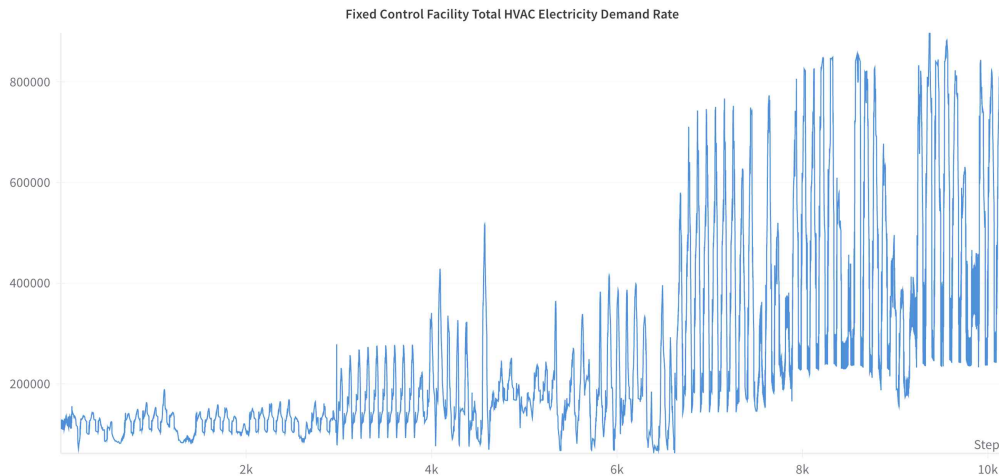
[그림 12] Comfort Penalty 그래프

(2) Fixed-Control과 DDPG 코드를 통해 에피소드 별 Facility Total HVAC

Electricity Demand Rate 평균값 비교

Facility Total HVAC Electricity Demand Rate는 건물 전체에서 HVAC 시스템이 소비하는 전력 수요율을 의미한다. 이는 HVAC 시스템이 운용되는 동안 시간에 따라 얼마나 많은 전력을 소비하는지를 나타내는 중요한 지표이다.

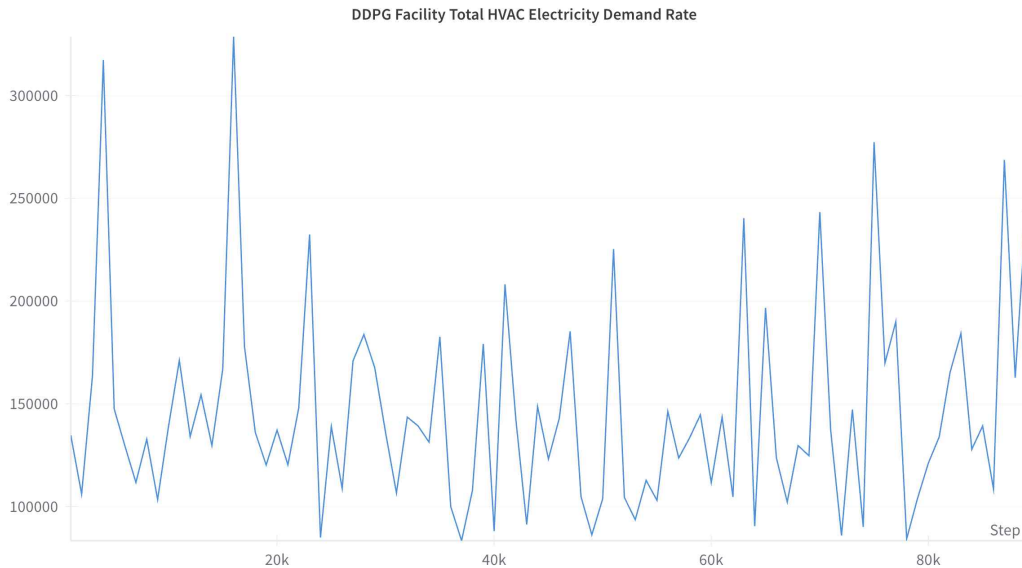
현실 세계에서는 HVAC 시스템이 사전에 정의된 알고리즘을 기반으로 동작하는 Fixed-Control 방식을 주로 사용한다. Fixed Control의 Facility Total HVAC Electricity Demand Rate 그래프는 다음과 같다. 그래프 후반부에서 에너지 소비량이 급격히 증가하면서 HVAC 시스템이 필요 이상의 전력을 소모하고 있다. 특히 중반부 이후부터는 에너지 소비 패턴이 불안정하여 전력 수요 예측이 어려워진다. 이는 HVAC 시스템이 비효율적으로 작동하고 있을 가능성을 시사한다. DDPG 모델과 비교했을 때, 전력 소비량을 줄이려는 노력 없이 지속적으로 증가하는 모습을 보인다. 즉, 학습 기반 제어가 아닌 Fixed Control 방식이 에너지 절약에 비효율적이라는 것을 보여준다.



[그림 13] Fixed Control의 Facility Total HVAC Electricity Demand Rate 그래프

DDPG의 Facility Total HVAC Electricity Demand Rate 그래프는 다음과 같다. 현재 그래프에서 전력 소비량이 넓은 범위에서 변동하는 것은 DDPG 모델이 다양한 액션을 시도하며 최적의 HVAC 운영 방안을 학습하는 과정을 보여준다. 특히, 특정 구간에서 급격한 소비량 변화가 발생하는 것은 모델이 새로운 전략을 실험하며 학습을 지속적으로 진행하고 있음을 의미한다. 뿐만 아니라 에너지 소비 패턴이 유동적으로 변화하면서 적응하는 모습을 보인다. 이는 모델이 주어진 환경 조건에 맞춰 HVAC 시스템을 조정하는 능력을 학습하는 과정으로 장기적으로 볼 때, 학습이 진행되면서 이러한 변동이 점차 줄어들고 안정적인 제어로 이

어질 가능성이 크다는 것을 알 수 있다.



[그림 14] DDPG의 Facility Total HVAC Electricity Demand Rate 그래프

(3) 실험 결과 분석

① 전력 소비량

Fixed Control 방식에서는 일정한 패턴으로 전력 소비가 발생하며, 특정 조건에서 급격한 소비 증가가 발생하였다. 반면, DDPG 모델을 활용한 강화학습 기반 제어는 에너지 소비 패턴을 최적화하며 불필요한 전력 낭비를 줄이는 데 성공하였다.

② 사용자 쾌적성

Fixed Control 방식에서는 환경 변화에 대한 즉각적인 대응이 어려워, 특정 시간대에 실내 온도가 쾌적 범위를 벗어나는 문제가 발생했다. DDPG 모델은 실내 온도와 외부 환경 변화를 실시간으로 분석하며 적절한 온도 조정을 수행하여 사용자 쾌적성을 유지하는 데 더욱 효과적이었다.

③ 에너지 절감율 비교

강화학습 기반 DDPG 모델을 활용할 경우, 고정 제어 방식 대비 전력 소비량을 더 낮추면서도 실내 환경을 일정한 수준으로 유지할 수 있음이 확인되었다. 이는 맞춤형 환경 시뮬레이션을 활용하여 HVAC 시스템을 최적화할 경우, 에너지 효율을 극대화하면서도 실내 환경을 보다 안정적으로 관리할 수 있음을 시사한다.

III. 결론

1. 탐사 및 연구 요약

지속 가능한 사회 선도라는 이화여자대학교의 목표에서 비롯한, 본 팀의 탐사 주제는 '강화학습을 활용한 HVAC 시스템의 맞춤형 환경 구축의 중요성'이다. 기존 연구들이 주로 상업용 건물의 에너지 절감과 효율성 개선에 초점을 맞춘 반면, 본 탐사를 통해 캠퍼스 환경에 최적화된 HVAC 시스템 연구의 필요성을 강조하고 이를 강화학습에 적용할 방법을 제시하고자 했다. 따라서, 주어진 기간동안 . 또한, 탐사를 통해 캠퍼스 맞춤형 환경 구축을 위한 사전 고려 사항을 분석하고, HVAC 시스템에 효과적으로 접목할 수 있는 강화학습 기술을 탐색하기로 했다.

탐사를 통해 본 팀의 주제에 대해 3가지 측면에서 인사이트를 얻을 수 있었다. 'HVAC 제어'에서는 매우 다양한 기계가 상호작용하며 작동하고 이를 제어하기 위해서는 다양한 기기가 존재하는 건물을 잘 설계해야 컴퓨터를 이용한 제어가 가능해진다는 것이다. 실제 연구에서도 가정하는 변수들이 실제 환경과 완전 동일한 상황은 만들기 어려우며 이를 위한 검증 과정이 필수적이다. 그러나 이렇게 특정 환경에 맞춰 학습된 모델은 확장성이 부족하기에 이를 극복할 방안 역시 필요하다는 것이다. '강화학습'에서는 데이터 부족으로 인한 성능 저하 문제를 HVAC의 물리적 조건들을 활용해 해결할 수 있지만 이 역시도 학습된 환경에서 모델을 사용해야 성능이 보장된다는 문제를 해결해야 한다. 마지막으로 '에너지 효율의 방법'으로 HVAC 제어 모델을 구축하는 것 이외에도 다양한 방법을 활용해볼 수 있다. HVAC 일반화와 맞춤화 사이 균형을 찾는 노력 이외에 제어 방식을 일반화하는 것을 통해 같은 모델을 최대한 많은 곳에 활용하는 방법과 HVAC 이외에 공실의 에너지를 절약하는 방법에 대한 고안도 해볼 수 있었다.

위와 같이 탐사를 통해 얻은 지식을 기반으로 실험을 통해 EnergyPlus에서 제공하는 건물 모델과 기후 데이터를 통해 최대한 일반적 캠퍼스와 유사한 환경을 구축한 뒤 DDPG과 Fixed control 결과를 비교하는 실험을 통해 강화학습 기반 HVAC 시스템이 캠퍼스 환경에서 최적의 에너지 효율성과 실내 쾌적성을 동시에 달성할 수 있는 가능성을 평가하고, 실질적인 적용 방안을 도출하는 것을 목표로 했다.

2. 연구의 의의

현대 건물에서 난방, 환기 및 공조 시스템은 전체 에너지 소비의 상당 부분을 차지하며, 이에 따라 에너지 절감과 실내 환경의 쾌적성을 동시에 고려한 최적화 전략이 필수적이다. 기존의 HVAC 제어 방식은 일반적으로 고정된 온도 설정 값(Fixed Control)을 기반으로 운영되며, 이는 건물의 실내 환경 및 외부 기후 변화

에 따라 효율적인 에너지 소비를 보장하지 못할 가능성이 높다.

특히, 최근 인공지능과 머신러닝을 활용한 HVAC 제어 기술이 해외에서는 활발하게 연구 및 적용되고 있음에도 불구하고, 국내에서는 여전히 전통적인 제어 방식에 의존하는 경우가 많아 에너지 절감 및 사용자 쾌적성을 극대화하는 데 한계가 존재한다. 이에 본 연구는 강화학습을 활용한 HVAC 제어 시스템의 가능성을 검증하고, 이를 기존의 Fixed Control 방식과 비교함으로써 맞춤형 환경 시뮬레이션이 HVAC 시스템 최적화에 미치는 영향을 분석하는 것을 목표로 했다.

본 연구에서는 EnergyPlus 기반의 Sinergym 환경을 활용하여 맞춤형 HVAC 시뮬레이션 환경을 구축하고, DDPG 강화학습 모델과 Fixed Control 방식을 비교하였다. 실험 결과, DDPG 모델을 활용한 강화학습 기반 HVAC 제어는 기존 Fixed Control 방식 대비 에너지 소비를 효과적으로 절감하면서도 사용자 쾌적성을 더 잘 유지하는 것으로 확인되었다.

이는 맞춤형 환경 시뮬레이션이 HVAC 시스템 최적화에서 가지는 중요성을 실험적으로 검증하였다는 점에서 의미가 크다. 기존의 Fixed Control 방식이 일정한 전력 소비 패턴을 가지지만 비효율적인 측면이 존재함을 확인하였으며, 강화학습을 적용한 HVAC 제어가 더 적은 전력으로 실내 환경을 효과적으로 관리할 수 있음을 입증하였다. 또한 DDPG 모델이 실내 환경의 변화를 실시간으로 반영하며 최적의 제어 정책을 학습하는 과정을 통해, 맞춤형 환경 시뮬레이션이 강화학습 기반 HVAC 최적화에서 필수적인 요소임을 강조하였다.

현재까지 한국에서는 AI 기반 HVAC 최적화 연구가 상대적으로 활발하지 않은 상황이지만, 본 연구는 강화학습을 통한 HVAC 최적화의 가능성을 실험적으로 검증하고, 이를 바탕으로 국내 스마트 빌딩 시스템의 에너지 절감 전략을 수립하는 데 기여할 수 있는 첫 걸음을 내딛었다는 점에서 의미가 크다.

3. 연구의 한계 및 제언

이번 탐사를 통해 HVAC 시스템 최적화에 대한 다양한 접근법을 연구하고 시도했지만, 캠퍼스와 같이 특정 목적을 가진 건물에 대한 연구는 충분히 다루지 못한 부분으로 남았다. 이를 보완하기 위해, 특정 목적의 건물을 설정한 후 해당 건물의 일반적인 특성을 기반으로 모델링하고, 유사한 환경을 가진 건물에서도 적용할 수 있는 범용적인 HVAC 제어 모델을 개발하는 연구가 필요하다. 이를 위해 메타러닝과 전이학습을 활용하여 다양한 건물 환경에 적응할 수 있는 모델을 구축하고, 기존의 고정된 제어 방식보다 효율적인 최적화 방안을 모색하는 것이 중요하다.

또한, 중앙 제어식 HVAC 시스템을 운영하는 한국의 건물 환경에 효율적 자동화

시스템을 도입하기 위한 연구가 필요하다. 현재 대부분의 건물에서는 사전에 설정된 규칙에 따라 HVAC 시스템이 작동하며, 실시간 최적화 기능이 부족하다. 이를 개선하기 위해서는 먼저 실제 건물 운영 데이터를 수집하고 유사한 건물 사이 데이터를 공유하는 과정이 필요하다. 이에 더해, 고해상도 시뮬레이션 환경을 구축하여 최적화된 HVAC 제어 모델을 구축하는 연구를 진행해야 한다. 이와 같은 연구를 통해 한국의 기존 HVAC 시스템을 발전시키고, 에너지 절감을 통한 지속 가능성을 높일 수 있을 것으로 기대된다.

IV. 참고문헌

- Ding, C., Ke, J., Levine, M., & others. (2024). *Potential of artificial intelligence in reducing energy and carbon emissions of commercial buildings at scale*. *Nature Communications*, 15, 5916. <https://doi.org/10.1038/s41467-024-50088-4>
- Duarte, C., Raftery, P., & Schiavon, S. (2017). *Development of whole-building energy models for detailed energy insights of a large office building with green certification rating in Singapore*. *Energy Technology*. <https://doi.org/10.1002/ente.201700564>
- Hou, F., Cheng, J. C. P., Kwok, H. H. L., & Ma, J. (2024). *Multi-source transfer learning method for enhancing the deployment of deep reinforcement learning in multi-zone building HVAC control*. *Energy and Buildings*, 114696. <https://doi.org/10.1016/j.enbuild.2024.114696>
- Kim, S. K., Song, C.-O., Mostafavi, S., & Aswani, A. (2020). Discovering symbolic policy for building control using reinforcement learning. *IFAC-PapersOnLine*, 53(2), 12713–12718.
- Manjavacas, A., Campoy-Nieves, A., Jiménez-Raboso, J., Molina-Solana, M., & Gómez-Romero, J. (2024). *An experimental evaluation of deep reinforcement learning algorithms for HVAC control*. arXiv. <https://doi.org/10.48550/arXiv.2401.05737>
- Narayanan, S., Apte, M. G., Haves, P., Piette, M. A., & Elliott, J. (2010). *Systems approach to energy-efficient building operation: Case studies and lessons learned in a university campus*. 2010 ACEEE Summer Study on Energy Efficiency in Buildings. Building Technologies Department, Building Technology and Urban Systems Division, Simulation Research.
- Roa, C. D., Raftery, P., Prakash, A., & et al. (2023). *Field demonstration of the Brick ontology to scale up the deployment of ASHRAE Guideline 36 control sequences*. UC Berkeley, Controls and Information Technology. <https://escholarship.org/uc/item/5zt2d66r>
- Wei, T., Wang, Y., & Zhu, Q. (2017). *Deep reinforcement learning for building HVAC control*. In *Proceedings of the 54th ACM/EDAC/IEEE Design Automation Conference (DAC)* (pp. 1–6). IEEE. <https://doi.org/10.1145/3061639.3062224>

정혜진, 이성동, 이장호. (2023). 대학 건물의 공간 단위 에너지 소비량 분석. 기후변
화연구, 14(1), 41-52. <https://doi.org/10.15531/KSCCR.2023.14.1.041>

홍준기. (2024.05.18). 탄소 배출 주범은 공장·車? 40%는 빌딩서 나온다.... 해법은?
조 선 일 보 .
<https://www.chosun.com/economy/weeklybiz/2024/05/16/VSVLSQEXMRASBJJOZKL335OVCI/>

이대학보 미디어부. (2023.12.10). 내가 그린, 그린 캠퍼스 / 이대학보 해외취재 특집.
YouTube. <https://www.youtube.com/watch?v=JSbWclMUZ1Y>

[부록 1] 인터뷰 전문

[인터뷰 1. Chao Ding]

1. 인터뷰 개요

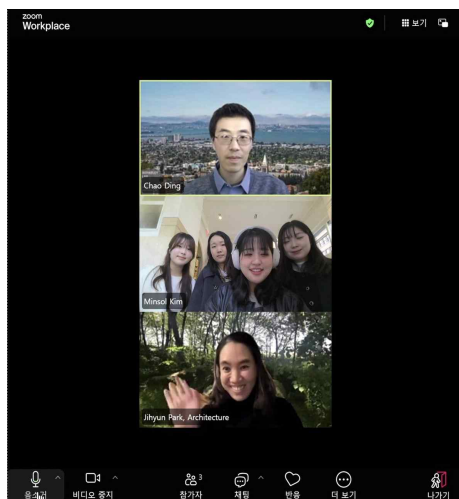
인터뷰 1. Chao Ding	
인터뷰 장소	비대면 Zoom
인터뷰 일시	2025년 1월 15일 오후 12시 (GMT-8)

<표 1> LBNL Chao Ding 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

Chao Ding은 LBNL의 Energy Technologies 부양에서 연구하고 있는 연구진이다. 특히 국제 에너지 분석 부서에서 기술 연구원으로 일하고 있다. 주로 Heting, Ventilation, Air conditioning and Refrigeration 시스템의 효율성을 높이고 있어 HVAC 시스템의 전반적으로 적용되고 있는 현황에 대해 배우고 이를 기존에 강화학습이 적용되지 않았던 HVAC 시스템만 사용했던 곳에서 현실적으로 어떻게 적용시킬 수 있을지 인사이트를 얻을 것으로 기대한다. 또한 Chao Ding의 ‘Potential of artificial intelligence in reducing energy and carbon emissions of commercial buildings at scale’이라는 article에서는 사무실에 초점을 맞춰 인공지능을 효과적으로 사용할 수 있는 방법에 대해 연구를 하고 있기에 건물에 인공지능을 접목 시킬 방법을 배우고 주제로 선정한 방식의 성능을 높일 수 있는 아이디어를 함께 모색하고자 하였다.

3. 인터뷰 사진



[그림 1] Chao Ding 인터뷰 사진

4. 인터뷰 전문

(Potential of artificial intelligence in reducing energy and carbon emissions of commercial buildings at scale (2024)) 해당 논문에서 특정 건물 유형으로의 적용을 하기 위해서는 추가적으로 고려해야할 부분이 있다고 언급 하셨는데 구체적으로 어떤 점을 고려해야하는지 여쭙보고 싶습니다. 특히 캠퍼스의 경우 어떤 것을 주의해야 하는 지 말해주세요.

이 논문을 보면 상업용 건물, 특히 중간 규모의 상업용 건물, 사무실 건물을 예시로 사용하여 방법론과 프레임워크를 설명하고 있습니다. 하지만 이 방법론은 사실 일반적이기 때문에 같은 아이디어를 다양한 유형의 건물에도 적용할 수 있습니다. 다른 유형의 상업용 건물이 있을 것입니다. 그렇게 하려면 논문을 읽고 방법을 따를 수 있다고 생각합니다. 하지만 에너지 모델을 재구성해야 합니다. 예를 들어, 우리가 했던 에너지 플러스 모델 모델링 작업을 얼마나 잘 알고 있는지 잘 모르겠습니다. 따라서 기본적으로 상업용 오피스 빌딩 유형은 나침반 건물과는 다릅니다. 효율성 수준의 점유율 측면에서 볼 때, 점유율이나 장비 밀도와 마찬가지로 조명 밀도는 모두 다를 것입니다. 따라서 여러분에게 가장 먼저 해야 할 일은 이 모델링 에너지 시뮬레이션을 통해 최대 절약 가능성이 무엇인지 이해하는 것이라고 생각합니다. 다시 말해, 예를 들어 개발된 PNL 모델과 같은 몇 가지 모델을 참조할 수 있습니다. 캠퍼스 건물 유형을 찾아보세요. 예를 들어, 대형 오피스 빌딩이나 중형 오피스 빌딩, 소형 오피스 빌딩 등 가장 가까운 유형을 찾은 다음 RANA 모델을 시도해 볼 수 있습니다. 그러면 그 건물 유형의 에너지 성능이 얼마나 되는지 이해합니다. 그리고 이 논문에서 특히 제어에 관심이 많으신 것 같아서 국립 연구소 중 한 곳의 보고서에 대해서도 이야기합니다. 다양한 제어 기술과 각 제어 기술의 저장 잠재력에 대해 이야기하는 매우 포괄적인 기술 보고서가 있으며, 이를 참고하여 이러한 저장 잠재력에 대한 명확한 아이디어를 얻으면 AI 역할을 수행할 수 있습니다. 하지만 기본적으로 이 논문의 가정에 따르면 AI가 사람들이 이러한 에너지 절약을 가속화하는 데 도움이 될 수 있다고 가정합니다.

각 기후대별 에너지 절감율(reduction rate)을 나타내주셨는데 비교적 난방의 수요가 높은 곳에서 에너지 절감률이 더 높게 나타난 원인이 무엇이라고 생각하시는 지 설명 부탁드립니다. 예를 들어, 캠퍼스처럼 연구, 학습 등 복합적 유형의 공간이 모인 건물에 적용하려면 어떤 점을 추가로 고려해야 하나요?

건물과 실외 기후가 건물 에너지 소비에 영향을 미친다는 것을 알고 있죠? 기후가

다르기 때문이죠. 예를 들어 건물이 매우 추운 지역에 위치해 있다면 난방이 냉방에 비해 지배적인 에너지 소비일 수도 있고 냉방이 전혀 필요하지 않을 수도 있습니다. 예를 들어 건물이 매우 더운 기후대에 위치해 있다면 냉방이 지배적이고 난방도 많지 않을 것입니다. 따라서 실제로 장비 관점에서 보면 냉방 효율이 난방 효율과 다르기 때문입니다. 예를 들어 시장의 장비를 보면 실제로 히트펌프 시스템이라면 난방 성능과 냉방 성능을 모두 가질 수 있습니다. 냉방과 난방의 효율도 찾을 수 있습니다. 따라서 실제로 이러한 수치는 기계적 관점에서 볼 때 기계가 좋은 냉방만큼 좋은 난방을 제공할 수 없다는 것을 의미합니다. 이는 다를 것입니다. 즉, 첫 번째입니다. 두 번째는 기후 때문에 건물 부하도 다를 것입니다. 따라서 기후에 따라 건물 부하도 달라질 것입니다. 따라서 기후에 따라 달라질 것입니다. 예를 들어 에너지 플롯을 사용하여 이전에 에너지 시뮬레이션을 수행하면 실제로 실외 조건에 따라 건물 부하와 연간 에너지 소비량이 달라집니다. 이것이 바로 기후대에 따라 에너지 소비가 달라지는 또 다른 이유입니다. 또한 이러한 실외 조건, 효율성 문제로 인해 시스템도 기후대에 따라 달라질 것입니다. 그렇기 때문에 기후대에 따라 이러한 수치는 다를 것입니다. 기후대에 따라 이러한 그림을 보면 기후대, 실외 조건에 크게 달라집니다.

동일한 시뮬레이션을 다른 기후 지역 또는 건물 유형에 적용할 경우, 어떤 추가적인 변수나 요소를 고려해야 한다고 보시나요?

첫 번째 단계에서는 에너지 플러스 모델을 만들어야 한다는 뜻입니다. 왜냐하면 에너지 플러스 모델에서는 실제로 구역별로 구역을 정의할 수 있기 때문입니다. 따라서 기능에 따라 각 구역마다 에너지 사용 패턴이 다를 수 있습니다. 예를 들어, 실험실은 일반 오픈 스페이스 오피스에 비해 훨씬 더 높은 에너지 사용량을 가질 수 있습니다. 예를 들어, 실험실처럼 초기에는 팬이나 펌프 같은 고밀도 에너지 소비 장비나 기타 유형의 기계 장치가 많이 있을 것입니다. 이는 일반 사무실에 비해 많은 에너지를 사용할 것입니다. 예를 들어, 에어컨만 사용하거나 테이블 램프 같은 것만 사용합니다. 이것이 첫 번째 쿼맨이라면 매우 다를 것입니다. 두 번째, 설정 지점도 다를 수 있습니다. 예를 들어, 설정 지점은 에어컨, HWAC 설정 지점, 예를 들어 온도와 습도 조절을 의미합니다. 하지만 예를 들어 특정 실험실의 경우, 실험실에서 기계적 작동이 필요하거나 특정 화학적 작동이 필요하거나 매우 위험한 화학 물질이 필요한 경우입니다. 사실 그런 종류의 장비의 경우 온도 뿐만 아니라 습도도 제어해야 하죠? 하지만 온도 때문에 기계적 설정 지점은 교실 설정이나 사무실 설정과 같이 일반에 비해 매우 다를 것입니다. 세 번째가 점유율이 될 것이라고 생각합니다. 따라서 사무실 건물이나 교실 건물의 경우 교사가 수

업을 할 때나 학생이 교실에 앉을 때 수업과 사용 패턴이 밀접하게 관련이 있다고 가정하기 때문에 일정도 달라질 것입니다. 하지만 실험실의 경우 특정 실험실의 경우 24시간 또는 7일 동안 사용해야 할 수도 있습니다. 이는 실제로 실험실의 기능에 따라 달라질 수 있습니다. 따라서 이러한 모든 차이를 합치면 에너지 플러스 모델에 완전히 다른 에너지 프로필을 제공합니다.
저희 역시 에너지 플러스를 통해 시뮬레이션을 하려고 합니다. 처음엔 캠퍼스 건물과 기후 환경이 가장 비슷하도록 커스텀을 하려고 했으나 실패해 교수님께서 사용하셨던 office medium을 이용해 실험을 진행하고자 합니다. 이렇게 진행해도 저희 캠퍼스 환경으로 일반화 시킬 수 있을지 아니면 추가적으로 어떤 노력을 더 해야할지 여쭙보고 싶습니다.
우선, 버전은 매우 중요합니다. 버전마다 버전을 놓치면 심각한 오류가 발생한다는 것을 알고 있습니다. 예를 들어, 소프트웨어가 하나의 버전을 지원하지만 소프트웨어가 지원하지 않는다면, 모델은 다른 에너지 플러스 모델 버전을 사용하여 개발되었으며, 이는 가장 먼저 해결해야 할 문제입니다.

[인터뷰 2. Alen Meier]

1. 인터뷰 개요

인터뷰 3. Alen Meier	
인터뷰 장소	LBNL Building 90
인터뷰 일시	2025년 1월 16일 오후 1시 (GMT-8)

<표 2> LBNL Alen Meier 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

Alan Meier는 LBNL의 수석 과학자이며, UC 데이비스(UC Davis) 환경과학 및 정책학과(Department of Environmental Science and Policy) 겸임 교수이자, UC Davis Energy and Efficiency Institute의 연구원이다. 그의 연구는 에너지 소비 패턴을 분석하고, 이를 줄일 수 있는 방안을 탐구하는 것을 중심으로 진행된다. 특히, 대기 전력(standby power) 사용에 대한 연구를 수행하며, 전 세계적으로 대기 전력 손실을 1W 이하로 줄이자는 국제 정책을 제안한 바 있다. 이 정책은 G8 국가들에 의해 공식적으로 채택되었으며, 현재까지도 가전제품 및 전자기기의 에너지 효율 개선에 영향을 미치고 있다. 이번 인터뷰는 LBNL 방문 당시, Mary Ann Piette의 소개로 진행되었으며, 대학 캠퍼스의 에너지 효율 향상과 HVAC 외 전력 사용이 미치는 영향을 논의하는 데 목적을 두었다.

3. 인터뷰 사진



[그림 2] Alen Meier 인터뷰 사진

4. 인터뷰 전문

HVAC 시스템에서 점유율(Occupancy)과 공실(Vacancy) 관리의 중요성에 대해 궁금합니다.
현재 대부분의 HVAC 최적화 연구는 점유율(occupancy), 즉 특정 공간에 사람이 얼마나 머무르는지를 기반으로 에너지 소비를 조절하는 방식으로 진행되고 있습니다. 건물 내 점유율이 높을수록 실내 공조 시스템이 적극적으로 가동되며, 사람이 적거나 없는 경우에는 HVAC 시스템의 출력을 줄이는 방식이 일반적입니다. 하지만 공실(vacancy), 즉 사람이 없는 상태를 활용한 최적화 연구는 상대적으로 부족한 상황입니다. 미국의 상업용 건물, 특히 소규모 오피스 빌딩이나 공공시설의 경우, 야간이나 주말 동안 아무도 없지만 HVAC 시스템이 여전히 작동하는 경우가 많아 최대 30%까지 불필요한 에너지가 낭비될 수 있습니다. 이는 점유율 기반 제어의 한계로 인해 발생하는 문제 중 하나입니다. 현재 많은 연구들이 점유율 데이터에 기반하여 HVAC를 최적화하는 방법을 모색하고 있지만, 실제 운영 환경에서는 사람이 없을 때 발생하는 불필요한 에너지 사용을 적극적으로 줄이는 방법이 더욱 중요한 연구 주제가 될 수 있습니다. 공실 상태를 보다 정확하게 감지하고 HVAC 최적화에 반영하려면, 다양한 센서 데이터를 종합적으로 분석하여 보다 정밀한 예측 모델을 구축하는 것이 필요합니다. 예를 들어, 실내 온도 변화 패턴, 공기 흐름 데이터, 스마트 IoT 디바이스와 연계된 사용 패턴 등을 조합하여 사람이 없는 상태를 보다 정밀하게 감지하는 방식이 연구될 필요가 있습니다.
건물 내 기타 전력 사용(Miscellaneous Energy Use)이 전체 에너지 사용량에 어느 정도의 영향을 끼치고 있는지 궁금합니다.
건물 내 에너지 소비에서 HVAC 시스템이 차지하는 비중이 크지만, 기타 전력 사용(Miscellaneous Energy Use)도 전체 에너지 소비량의 약 40%를 차지하며,

HVAC보다 더 많은 영향을 미칠 수도 있습니다. 일반적으로 건물 내 전력 사용량 분석은 HVAC, 조명, 기타 전력 사용(Plug Loads)으로 구분되며, 많은 연구들이 HVAC 최적화에 집중하고 있지만, 플러그 로드(Plug Loads)와 대기 전력 소비(Standby Power Consumption)에 대한 연구는 상대적으로 부족한 편입니다. 특히, 컴퓨터, 디스플레이, 엘리베이터, Wi-Fi 라우터, 음료 냉각기(drinking fountain) 등은 점유율과 관계없이 24시간 작동하는 경우가 많아 에너지 낭비의 주요 원인이 됩니다. 예를 들어, 사무실이나 연구실에서 야간이나 주말 동안 사용되지 않는 기기들이 대기 상태에서 전력을 지속적으로 소비하는 Standby Power Consumption 현상이 심각하게 발생할 수 있습니다. 현재의 건물 에너지 관리 시스템(EMS)은 주로 HVAC와 조명 제어에 집중되어 있으며, 플러그 로드(Plug Loads)까지 포함하여 최적화하는 경우는 많지 않습니다. 하지만 전체 에너지 소비량에서 기타 전력 사용이 차지하는 비율이 40% 이상이라는 점을 고려할 때, AI 기반 최적화가 HVAC뿐만 아니라 플러그 로드까지 확장될 필요가 있습니다. 결론적으로, 건물 에너지 절감을 위해서는 HVAC 시스템뿐만 아니라 기타 전력 소비까지 고려하는 통합적인 최적화 전략이 필요하며, AI를 활용한 자동 제어 시스템이 도입된다면 더욱 효과적인 에너지 절감이 가능할 것입니다.
미국에서 연구 중인 데이터셋을 한국에서 활용할 경우 고려해야 할 점이 있을까요?
미국에서 연구된 데이터셋을 한국에서 활용하려면 국가별 건물 운영 방식, 사용자 행동 패턴, 정책적 차이 등을 고려하는 것이 필수적입니다. HVAC 최적화 모델은 단순히 기술적인 문제뿐만 아니라 건물 사용자의 생활 패턴, 문화적 요소, 에너지 정책 등 다양한 요인에 의해 영향을 받기 때문입니다. 예를 들어, 한국과 일본에서는 점심시간 동안 전력 소비가 급격히 감소하는 현상이 관찰됩니다. 이는 한국과 일본의 직장 문화에서 점심시간이 명확하게 보장되며, 많은 기업이 이 시간 동안 조명, 컴퓨터, HVAC 시스템을 끄는 방식으로 운영되기 때문입니다. 반면, 미국에서는 점심시간 동안에도 건물 내 활동이 지속되기 때문에 전력 소비 패턴이 크게 변하지 않습니다. 이처럼, 같은 알고리즘이라도 국가별 데이터 특성이 다르면 HVAC 시스템의 최적화 방식도 달라져야 합니다. 만약 미국 데이터를 기반으로 한국에서 모델을 적용한다면, 점심시간과 같은 특정 시간대별 에너지 사용 패턴을 추가적으로 반영해야 합니다. 또한, 미국에서는 개별 난방(Individual Heating)이 일반적이지만, 한국은 중앙 난방(Central Heating)이 보편적이기 때문에 시스템 차이도 고려해야 합니다.

다.

정책적 요소도 중요한 차이점입니다. 미국에서는 에너지 절감 정책이 주(state) 단위로 운영되는 반면, 한국에서는 국가 차원에서 통합적인 에너지 절감 기준이 적용되는 경우가 많습니다. 이러한 차이를 반영하지 않으면, HVAC 최적화 모델이 현실적인 적용 가능성을 가지기 어려울 수 있습니다.

결론적으로, 미국의 HVAC 최적화 데이터셋을 한국에서 활용하려면 건물 운영 방식, 시간대별 소비 패턴, 냉난방 시스템의 차이, 정책적 요소 등을 반영하는 맞춤형 접근이 필요합니다.

[인터뷰 3. Mary Ann Piette, Donghun Kim]

1. 인터뷰 개요

인터뷰 2. Mary Ann Piette & Donghun Kim	
인터뷰 장소	LBNL Building 90
인터뷰 일시	2025년 1월 16일 오전 11시 (GMT-8)

<표 3> LBNL Mary Ann Piette & Donghun Kim 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

LBNL의 ETA 연구소는 에너지 기술 및 지속 가능한 환경 연구를 선도하는 미국 국립 연구소로, 건물 에너지 효율, 재생 에너지 통합, 스마트 그리드, 산업 에너지 관리 등 다양한 분야에서 혁신적인 연구를 수행하고 있다. ETA는 건물 에너지 최적화와 HVAC(Heating, Ventilation, and Air Conditioning) 시스템 개선을 위한 연구를 집중적으로 진행하고 있으며, 인공지능(AI)과 데이터 분석 기술을 활용한 차세대 에너지 관리 기법을 개발하는 데 주력하고 있다.

Mary Ann Piette는 LBNL의 ETA 연구실 부소장로, 건물 기술, 에너지 분석, 에너지 효율화 등 여러 부서를 총괄하며 건물의 에너지 시스템 효율화와 에너지 저장 시스템에 대해 연구한다. Donghun Kim은 LBNL ETA 소속 연구자이고 Purdue 대학교의 조교수로, 건물 HVAC&R 시스템, 에너지 모델링, 머신러닝, 제어 및 최적화 연구를 통해 에너지 효율성과 시스템 운영을 개선하는 것을 연구한다. 컨택 메일을 통해 Mary Ann Piette씨와 두 차례의 온라인 면담을 거쳐 인터뷰를 진행하게 되었다. Mary씨의 추천으로, HVAC을 주 연구 분야로 삼고 있는 Donghun Kim 박사님도 함께 인터뷰에 참여하게 되었다.

3. 인터뷰 사진



[그림 3] Mary Ann Piette & Donghun Kim 인터뷰 사진

4. 인터뷰 전문

저희는 Mary씨가 쓰신 논문에서 MPC 모델을 사용하여 연구를 진행한 것을 보았습니다. 하지만 저희는 computing power의 한계로 MPC 모델을 사용하는 대신에, 강화학습을 기반으로 연구를 진행하려고 합니다. 혹시 그렇게 진행할 때에 주의해야할 점이 있을까요?

MPC 모델 대신 강화학습을 기반으로 연구를 진행하려는 이유가 컴퓨팅 파워의 한계 때문이라면, 몇 가지 주의할 점이 있습니다. 우선, MPC 모델은 수학적 최적화와 제어 이론에 기반하여 복잡한 계산을 요구하는데, 강화학습에서도 유사한 수준의 복잡성을 다루서 비슷한 컴퓨팅 파워가 필요할 수 있습니다. 따라서 수학적 최적화 문제를 보다 공식 등을 이용해 간단하고 효율적으로 표현하는 방법을 고민해야 합니다. 또한, 강화학습은 충분한 데이터와 시뮬레이션 환경이 필요하며, 학습 속도가 느리거나 안정성이 떨어질 수 있으므로 적합한 알고리즘 선택과 하이퍼파라미터 튜닝이 중요합니다. 특히, 강화학습에서는 환경과의 상호작용을 통해 학습이 이루어지므로, 현실을 잘 반영한 정교한 시뮬레이션 환경을 설계하고 컴퓨팅 자원이 허용하는 한 시뮬레이션 속도를 최대화해야 합니다.

MPC와 강화학습 모두 높은 컴퓨팅 성능을 요구하기 때문에 분산 학습이나 클라우드 컴퓨팅 자원을 활용하는 방안을 고려해 자원을 효율적으로 활용하는 것이 필요합니다.

교수님께서서는 주로 MPC 모델을 연구에 사용하셨는데 후에 강화학습을 적용하실 의향이 있으신지 궁금하고, 없으시다면 이 이유를 여쭙보고 싶습니다.

우리는 특정 기술에 국한되지 않고 어떤 기술에도 열려 있습니다. 다만, 시설 운영자, 에너지 관리자, 건물 소유자 등 많은 이해관계자들이 건물과 관련된 결정에 매우 민감하기 때문에 선택한 기술이 신뢰할 수 있고 확장 가능하며 증명된 방법론이어야 합니다. 실제로 다른 연구실에서는 관련 방법론을 실험한 적이 있고, 제 동

료 중 한 명도 강화학습을 활용해 작업한 경험이 있습니다. 결론적으로 신뢰성, 확장 가능성을 갖춘 기술을 적용할 계획입니다.
저희 팀은 HVAC에 사용되는 Chiller(냉각기)와 같은 하드웨어 기기에 익숙하지 않은데, 혹시 이러한 하드웨어 기기를 활용하는 구조에 대해서 설명해 주실 수 있나요? 저희 연구의 키워드인 ‘캠퍼스’를 충족시키려면, 어떤 하드웨어 기기를 중점적으로 살펴봐야할까요?
HVAC 시스템은 냉난방 시스템으로 간단히 요약할 수 있습니다. 에어컨은 전력을 사용해 압축기를 작동시키고 냉매를 순환시켜 열을 한 환경에서 다른 환경으로 이동시키는 방식으로 작동합니다. 예를 들어, 여름철 에어컨 실외기에서 나오는 뜨거운 바람은 실내의 열이 실외로 이동했기 때문입니다. 히터는 반대로 열을 공간으로 가져오는 방식입니다. 이외에도 HVAC 시스템의 장치들은 전력을 사용해 열을 추출하거나 방출하는 방식으로 작동합니다. HVAC 시스템은 냉난방뿐만 아니라 환기나 공간 조절 기능도 포함할 수 있습니다. 연구 키워드인 ‘캠퍼스’를 충족시키기 위해서는 캠퍼스가 다양한 목적의 공간들을 복합적으로 포함하는 공간임을 기억해야 합니다. 캠퍼스는 실험실, 강의실, 사무실 등 공간마다 다른 HVAC 설계가 필요합니다. 강화학습이나 MPC를 HVAC 시스템에 적용할 때 가장 중요한 목표는 온도 조절 장치를 통해 쾌적한 환경을 제공하면서 에너지를 절약하는 것입니다. 이를 위해 냉난방 하드웨어 기기에 중점을 두고 살펴봐야 합니다. 하지만, 하드웨어 기기를 살펴보기 전에 현재 시스템이 제대로 작동하는지에 대한 데이터 수집과 분석이 선행되어야 합니다. 기본적인 결함 감지와 시스템 점검이 이루어져야만 MPC와 같은 정교한 제어 방법이 효과적으로 적용될 수 있습니다. 이는 강화학습 등 다른 접근법에서도 동일한 원칙입니다.
저희는 Energyplus의 “official medium”을 채택하여 연구 중인데 하드웨어 기기와 관련해 더 추가하거나 수정해야할 부분이 있을까요?
건물마다 고유의 특성과 HVAC 시스템이 다 다르기 때문에, 기존 모델을 활용하는 것보다는 시스템에 맞는 모델을 개발하는 것을 추천합니다. EnergyPlus는 일반적인 도구로는 유용하지만, 특정 제어에는 한계가 있습니다. 예를 들어, EnergyPlus는 제어해야 할 대상의 짧은 시간 동역학을 포착하지는 못합니다. 초기 설정에 따라 보통 5분 단위로 샘플링 시간이 제한되기 때문입니다. 또한, EnergyPlus는 초기 설계나 시뮬레이션에는 유용하지만 운영 단계에서는 한계를 갖기 때문에 사용자 정의를 통해 시스템을 실제 환경과 일치시키는 커스텀 작업의 도입을 추천합니다.
Mary씨의 논문 ‘Systems Approach to Energy Efficient Building Operation:

Case Studies and Lessons Learned in a University Campus’을 읽어보았을 때 기상 변수를 입력값으로 사용하신 것이 인상적이었습니다. 이 기상 변수가 실시간 기상 정보를 입력한 것인지 궁금합니다.
네, 실시간 기상 정보를 기상 변수로 사용했습니다. 날씨 데이터는 모델과 훈련 과정 모두에 큰 영향을 미치기 때문에 실시간으로 온도, 습도, 태양광 등의 정보를 모두 고려해 실험을 진행해야 합니다.
위 질문과 동일한 논문에서, DCV를 최적화하기 위해 센서를 통하여 실시간 데이터를 수집하고 처리했다는 것을 알게 되었습니다. 해당 실시간 데이터 수집 및 처리 과정을 어떤 식으로 DCV 최적화에 이용할 수 있는지 여쭙보고 싶습니다.
논문에서는 시간 단위 데이터 아카이브 시스템을 설정하고, 데이터 형식을 표준화 하였습니다. AI와 데이터의 상호운용성을 확보하기 위해서는 데이터 형식 표준화, 데이터 라벨링 규칙을 정리하는 과정이 매우 중요합니다. 이 과정을 거쳐야 DCV 최적화에 활용 가능한 데이터 상태를 만들 수 있습니다. 추가적으로 한국에서는 빌딩 관리 시스템(BMS)이 설치되어 있어도 데이터를 제대로 활용하지 않는 경우가 많습니다. 이러한 데이터를 활용하기 위해서는 현장 건물 전문가와 지속적으로 협력하는 것이 가장 중요할 것입니다.
교수님께서 정확성과 실용성을 높일 필요가 있다는 논문에서 언급하신 한계점에 대해 큰 공감을 하고 있습니다. 실용성을 더 높이기 위한 방법은 어떤 것들이 있을 지 생각해보신 방법이 있을까요?
DDPG(Deep Deterministic Policy Gradient)를 많은 연구원들이 사용하고 있습니다. 하지만 강화학습을 실제 빌딩에 적용하는 데는 모델링과 데이터 준비가 90%의 작업을 차지할 정도로 중요하기 때문에 DDPG를 사용하기 이전에 물리적 요소의 제어 연구가 먼저 선행되어야 합니다.
만약 지어지지 않은 건물의 HVAC 시스템을 제어하려면 어떻게 해야 할까요?
EnergyPlus를 설계 도구로 사용하여 다양한 가정과 가상 시나리오를 통해 분석할 수 있습니다. 예를 들어 실제로 비슷한 기상 정보를 가져와서 시뮬레이션을 진행할 수 있습니다. 다만 EnergyPlus는 운영 도구로 사용하기에는 한계가 많기 때문에 설계 도구로 사용한 이후 건물이 지어진 이후에는 단열, 창문, 냉각 장치 등을 변경하며 다시 시뮬레이션을 진행해야 합니다.

[인터뷰 4. Carlos Duarte]

1. 인터뷰 개요

인터뷰 2. Carlos Duarte	
인터뷰 장소	UC Berkeley CBE Lab
인터뷰 일시	2025년 1월 22일 오후 12시 30분 (GMT-8)

<표 4> UC Berkeley Carlos Duarte 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

UC Berkeley의 Center for the Built Environment(CBE Lab)는 건물 환경 및 에너지 효율 연구를 전문으로 하는 연구 기관이다. CBE Lab은 건물의 에너지 소비를 줄이고, 실내 환경의 품질을 향상시키는 데 초점을 맞춘 다양한 연구를 수행하며, 특히 HVAC 최적화, 스마트 빌딩 운영, 사용자 중심의 에너지 관리 시스템 등을 연구하고 있다. 이번 방문을 통해, CBE Lab이 수행하는 연구 방향과 HVAC 최적화 관련 최신 연구 내용을 탐색하고자 하였다.

Carlos Duarte는 UC Berkeley의 교수로 재직중이며 Building Technology 분야를 연구 중이다. 주로 건물 에너지와 Brick을 연결짓는 연구를 주로 진행하였다. 그의 연구를 기반으로 Brick과 같은 온톨로지 방법을 통한 맞춤형 환경 구축의 편의성 증대 방안과 HVAC 연구의 방향성 예측을 중심으로 논의하였다. 또한, 국가 혹은 지역에 맞는 건물을 모델링함으로써 HVAC 시스템 조절을 시뮬레이션하는 것에 대한 자세한 설명을 듣고자 했다.

3. 인터뷰 사진



[그림 4] Carlos Duarte 인터뷰 사진

4. 인터뷰 전문

저희의 원래 주제는 건물 별 HVAC 제어 시스템을 통합하여 어떤 건물이든지 빠르게 적응할 수 있는 강화학습 알고리즘을 구축하는 것이었습니다. 그러나 해당

주제가 현실 가능성이 매우 낮을 뿐더러, HVAC은 알고리즘 외의 다른 part가 오히려 더 중요하다는 피드백을 받아 주제를 특정 건물에 대한 강화학습 알고리즘 적용 가능성으로 주제를 축소하게 되었습니다. 바뀐 주제에 관련한 조언을 받을 수 있을까요?

빌딩에 스마트 컨트롤을 도입하려고 하는군요. 하지만 강화 학습과 모델 예측 제어를 통해 배운 것 중 하나는 이러한 종류의 방법이 건물 자체에 맞게 맞춤화되었다는 점입니다. 따라서 이 절차를 개발하여 강화 학습이나 기타 고급 제어 알고리즘을 설치하면 해당 건물에만 적용할 수 있고, 다음 건물에 적용하기 위해 많은 훈련을 하거나 직접 많은 정보를 수집하지 않으면 다른 건물에 확장성이 떨어지거나 휴대할 수 없습니다. 따라서 이러한 모든 데이터 기반 제어 알고리즘의 주요 한계 중 하나는 훈련에 많은 데이터가 필요하고 건물이 있는 특정 환경과 특정 건물 자체에서 데이터를 수집할 수 있기 때문입니다. 따라서 실제로는 실현 가능하지 않습니다.

그리고 또 다른 점은 이러한 스마트 제어 알고리즘을 작동시키기 위해 필요한 기술 세트가 시설 관리자로부터 자주 제공되지 않는다는 것입니다. 따라서 이 알고리즘을 개발하여 건물에 설치하려고 하면, 효과적으로 작동하기 위해 유지보수가 필요한 사람들이 있고, 그런 사람들은 문제가 발생했을 때 시설 관리자가 유지보수, 디버깅 또는 무엇이 문제인지 파악할 수 있는 기술 세트가 없을 수도 있습니다. 그래서 이것이 또 다른 주요 한계 중 하나입니다. 건물 관리자는 종종 난방이 켜지지 않는 것처럼 보이고 프로그램을 추적하여 이 임계값처럼 말할 수 있기 때문에 건물 업계가 규칙 기반 제어에 집중해 온 이유는 이 온도 임계값에 도달하지 못했기 때문이며, 작동하지 않은 센서나 액추에이터가 좋지 않았기 때문입니다. 하지만 일반적으로 기술 세트 내에서 현재 알고리즘의 문제를 찾을 수 있고 해결할 수 있는 무언가가 있습니다. 하지만 블랙박스 모델을 넣기 시작하면 무엇을 해야 할지 알 수 없습니다.

(Development of whole building energy models for detailed energy insights of a large office building with green certification rating in Singapore) 건물을 모델링할 때 싱가포르의 BCA에서 제시한 Green Mark Scheme 4.0 인증 수준 요건을 기준으로 커스텀을 했다고 이해했습니다. 그 과정에서 가장 신경 쓴 부분이 있나요?

에너지 시뮬레이션 모델의 경우 미국을 기준으로 한 경우가 많습니다. 국가마다 정의하는 비용이 다르기 때문에 그 외의 국가의 건물을 모델링 할 때에는 이를 유의해야 합니다. 해당 연구를 진행할 때에는 우선, 싱가포르의 상황에 맞춰야 했습니

다. 이를 테면, 건물 점유율은 미국과 다를 것입니다. 어느 건물이나 100%의 점유율을 가질 수는 없고, 얼마만큼의 점유율을 가지고 있는 지는 상황에 따라 다르죠. 때문에 저희가 Green Mark Scheme를 활용한 것입니다. 뿐만 아니라 저희는 디자이너들을 대상으로 어떤 건물을 일반적으로 디자인하고, 어떻게 해당 벤치마크를 적요하는 것이 좋은 지에 대해 설문조사를 진행했습니다. 이는 저희의 최종 목표가 최소한의 에너지를 소비하는 건물이 어떤 형태인지 이를테면 어떤 창문을 가지고 있고, 벽은 어떤 재료로 만들어야 하는 지 알아내는 것이기 때문입니다.

만약 정부 혹은 관련 기관에서 제시한 기준이 없다면 어떤 것을 기준으로 건물을 커스텀 해야하나요? 예를 들어, 우리는 ASH에서 제공하는 캠퍼스 빌딩을 모델링하고자 했는데 특히 고려할만 한 기준이 있나요?

건축회사나 기계 엔지니어 같은 건축 디자이너들과 가장 큰 부분을 이야기할 것 같아요. 건축 회사나 기계 설계 회사마다 각기 다른 종류의 건축 기준이 있을 거예요. 하지만 그 중 여러 곳에서 데이터를 수집하면 최소 건축물에 대한 그들의 기준이 어떤 것인지 알 수 있을 거예요.

미국에서는 대부분의 캠퍼스가 주로 주정부 소유이기 때문에 일반 대중과 동일한 요구 사항이 없기 때문에 한국에서는 어떻게 운영되는지 잘 모르지만 상황에 따라 다릅니다. 미국에서는 많은 캠퍼스가 일반 대중과 동일한 요구 사항을 가지고 있지 않습니다. 따라서 캠퍼스 관리자에게 자체 표준이 있는지 또는 일반적으로 구축하는 애쉬레 가이드라인이나 애쉬레 표준과 같은 문서를 참조할 수 있는지 확인할 수도 있습니다. 애쉬레는 미국 전역이나 다른 국가에서 표준이 아니기 때문에 애쉬레 표준을 채택할 것이라고 가정할 수 없기 때문입니다. 참고 사항일 뿐입니다. 하지만 많은 조직이 자체적으로 작업을 수행합니다.

논문을 통해 모델링한 과정을 확인했습니다. BCA에서 제시한 기준을 바탕으로 애쉬레에서 제공한 기준을 조절하고, 이를 오픈 소스 코드를 활용해 파일로 구현하고 결과를 확인한 것 같은데 내가 제대로 이해한 것이 맞나요? 맞다면 해당 내용을 더 자세하게 알려줄 수 있나요?

Energy Plus는 IDF라는 파일 시스템을 사용합니다. 이러한 GUI의 배경에는 모델, 모든 장비, 모든 매개변수 및 제어하려는 모든 종류의 값이 포함된 파일이 생성됩니다. 그리고 이 파일을 IDF라고 하며, 텍스트 편집기에서 열고 원할 경우 직접 편집할 수 있습니다. 좋아요, 저는 처음에 그래픽 사용자 인터페이스를 사용하여 Energy Plus와 상호 작용하기 시작했습니다. 하지만 지금은 이 아이디어 파일을 직접 편집하고 파일 자체 내에서 필요에 따라 모델을 변경할 수 있습니다. 그래서

오픈 소스는 독점 파일 시스템이 아니기 때문에 오픈 소스라는 뜻입니다. 텍스트 편집기에서 직접 열고 직접 편집할 수 있는 기능입니다. 또한 라이선스가 없기 때문에 오픈 소스이기도 합니다. 따라서 싱가포르에서 제공한 벤치마크 건물을 사용하여 수수료나 기타 비용 없이 자체 프로젝트에 사용할 수 있습니다.
오픈 소스 코드를 이용해 파일로 구현한 과정을 조금 더 자세하게 설명해줄 수 있나요? 어떤 부분을 어떻게 바꾼 것인가요?
저는 Python을 사용하여 이 IDF를 직접 모델링하고, 이 IDF 텍스트 파일을 직접 모델링합니다. 따라서 Python을 사용하면 Python이 이미 포함하고 있는 표준 텍스트 파일 편집 프로그램 패키지를 사용할 수 있는 유연성이 뛰어납니다. 하지만 EPI 파이썬 패키지라는 것이 있는데, 이 패키지는 객체와 참조 필드를 보다 직접적으로 참조할 수 있는 더 다재다능한 기능을 제공합니다. 네, 바로 Eppy입니다. 이 패키지는 Python 패키지로, 설치한 다음 이를 사용하여 IDF, 즉 에너지 플러스 파일을 로드하고 필요에 따라 수정할 수 있습니다. 그래서 Python을 사용하여 이 아이디어 파일을 상호작용하거나 개발한 다음 Energy Plus 프로그램을 호출하여 시뮬레이션을 실행합니다. 그런 다음 Python을 사용하여 결과를 수집하고 필요에 따라 분석합니다.
(Field Demonstration of the Brick Ontology to Scale up the Deployment of ASHRAE Guideline 36 Control Sequences) 고급 제어 시퀀스에서 논문에서 시연하신 과정들이 가지는 효율성을 쿼리 시간 이외에 정량적으로 표현할 수 있는지와 그 방법이 궁금합니다.
이 논문에 나오는 연구는 주로 파일럿 연구였습니다. 그래서 성능을 정량적으로 평가할 수 있는 많은 지표가 없었습니다. 이는 단순히 개방형 소스 프로젝트들을 연결할 수 있는지를 보여주는 연구였습니다. 저희가 연결한 개방형 프로젝트는 총 4개의 기술인 Macnet, Brick, CDL, Guideline 36 Control Sequences를 연결하는 것이었습니다. Macnet은 건물 시스템을 위한 통신 프로토콜입니다. Brick은 센서, 액추에이터, 건물 내 자산들을 표준화된 방식으로 설명하는 시스템입니다. CDL은 소프트웨어와 독립적으로 제어 알고리즘을 프로그래밍할 수 있도록하는 표준 방식입니다. Guideline 36 Control Sequences는 어떤 건물에도 적용할 수 있는 표준화된 제어 시퀀스 목록으로, 높은 성능을 보장합니다. 저희 연구의 목표는 이들을 연결하여 건물을 제어할 수 있다는 것을 입증하는 것이었습니다. 연구 당시, 에너지 절약량을 측정하고 검증하는 것보다는 이러한 시스템들을 효과

적으로 연결하는 것에 집중했습니다. 현재 이 프로젝트의 후속 연구를 진행중입니다. 이를 위한 현장 실증 연구가 두 개 있는데 하나는 캠퍼스 내 건물, 또 다른 하나는 버클리 시내 건물에서 이루어지고 있습니다.

이 후속 연구에서는 에너지 절감량을 실제로 정량화하는 데 초점을 맞추고 있습니다. 에너지 절감량을 정량적으로 평가하는 방식은 기존 제어 방식으로 운영된 건물의 데이터를 수집한 후 새로운 고급 제어 방식으로 운영된 데이터를 수집하여 두 값의 차이를 통해 에너지 절감 효과를 분석하는 것입니다. 이때 HVAC 운영과 실내 온도 같은 지표를 기반으로 에너지 소비 차이를 정량적으로 측정하려고 합니다.

환경이 실제 건물과 비슷할수록 건물에 적용할 시스템의 성능이 좋아지지만 그만큼 시간 및 비용이 많이 소모될 것이라 생각합니다. 앞으로 연구 방향이 구체화된 환경을 효율적으로 구축할 수 있도록하는 연구가 많아질지 혹은 일반화할 수 있는 통합적 모델에 대한 연구가 진행이 될지 궁금합니다.

저는 건물의 모든 요소를 일반화해 설계하는 방향으로 연구가 진행될 것이라고 생각합니다. 현재 모든 건물은 설계 방식, 제어 방식, 프로그래밍 방식 등이 각기 다릅니다. 저는 이번 연구에서 중요한 역할을 한 Brick 온톨로지를 활용하여 제어 시스템을 표준화하는 방안을 연구 중입니다. 예를 들어, Brick 스키마는 특정 용어를 사용하여 각 건물 요소를 표준화된 방식으로 설명하는 구조를 가지고 있습니다. 전등을 특정 용어로 정의하고 그 용어에 특정 의미와 속성을 부여해 컴퓨터가 이를 항상 같은 방식으로 인식하는 것입니다.

이는 프로그래밍 언어인 Python과 비슷합니다. Python에서는 특정한 방식으로 코드를 작성해야 컴퓨터가 이해할 수 있습니다. 이를 우리는 건물 시스템에도 적용하려고 합니다.

즉, 건물의 각 요소를 특정 용어와 속성으로 정의하며, 다른 건물에서도 같은 용어를 사용해 동일한 데이터 구조를 유추할 수 있게 됩니다. 이럴 경우, 제어 알고리즘이 어떤 건물에서도 적용할 수 있는 범용적인 모델이 될 수 있습니다.

현재 저희가 연구 중인 제어 알고리즘은 캠퍼스 내 건물에서 개발된 것입니다. 하지만 같은 Brick 데이터 모델을 사용하여 버클리 시내 건물에도 동일하게 적용해 볼 예정입니다. 만약 이 방법이 성공하다면 별도의 프로그래밍 없이 다른 건물에서도 제어 시스템이 정상적으로 작동해 필요한 센서, 액추에이터, 통신 프로토콜 등을 자동으로 인식하고 즉시 실행될 것입니다.

새로운 건물에 적용할 때마다 처음부터 다시 프로그래밍해야 하기 때문에 특정 건물에서만 작동하는 모델은 확장성이 없습니다. 따라서 저는 제어 시스템이 일반화

하는 방향으로 가야한다고 생각합니다.
건물을 설계할 때 특별히 고려해야 할 사항이 있을까요?
이는 연구 목적에 따라 달라집니다. 만약 특정한 프로젝트를 진행한다면, 설계 도면, 기계 장비 도면, 건물 내 배치 정보 등 가능한 많은 구체적인 데이터를 반영하는 것이 중요합니다. 하지만, 더 넓은 범위의 건물을 대표하는 모델을 만들고 싶다면 그 지역에서 일반적으로 건설되는 건물 유형을 통계적으로 분석해야 합니다. 예를 들어, 뉴욕의 평균적인 사무실 건물 유형을 모델링하려면 해당 지역의 건축 스타일, 평균 건축 연도, 건물의 일반적인 크기 등을 조사해야 합니다.
왜 건물과 건물 내부의 여러 요소들의 관계들이 계층적인지 설명해주실 수 있나요? 예를 들어 1층, 2층 등 분리되는 공간들이 왜 계층적으로 표현되어야 하는 것인지 설명해주실 수 있나요?
이는 Brick 스키마의 기본 구조 때문입니다. 제가 이 스키마를 직접 개발한 것은 아니지만, 이 구조를 개선하는 과정에 참여했습니다. Brick 스키마는 건물 내 요소들을 특정한 범주로 구분하여 계층적으로 정리하는 방식을 따릅니다. 예를 들어 이 컴퓨터를 건물 데이터 모델에 추가한다고 가정해보겠습니다. 우리는 이 컴퓨터를 장비라고 정의할 수 있습니다. 그런데, 이 장비가 어떤 유형인지 더 구체적으로 정의할 수도 있습니다. 즉, 더 세부적인 정보가 필요하면 계층 구조에서 더 깊이 들어가게 됩니다. 처음에는 장비로 정의하고 그 다음 단계에서는 컴퓨터, 더 깊게 내려가면 노트북처럼 점점 더 구체적인 속성을 추가할 수 있습니다. 이런 방식이 중요한 이유는 모든 연구자가 동일한 수준의 정보를 가지고 있는 것이 아니기 때문입니다. 어떤 연구자는 건물의 모든 세부 정보를 알고 있을 수도 있고, 어떤 연구자는 대략적인 정보만 가지고 있을 수도 있습니다. 그래서, 일반적인 정보를 입력한 후에도, 나중에 더 구체적인 정보를 추가할 수 있도록 설계한 것입니다. 즉, 연구자가 알고 있는 정보의 수준에 따라 Brick 스키마의 계층 구조를 활용하여 필요한 만큼의 정보를 입력할 수 있도록 설계되었기에 계층 구조로 표현되는 것입니다.

[인터뷰 5. 박지원]

1. 인터뷰 개요

인터뷰 1. 박지원	
인터뷰 장소	UC Berkeley Ong&Ong Plaza

인터뷰 일시	2025년 1월 16일 오후 3시 (GMT-8)
--------	----------------------------

<표 5> UC Berkely 박지원 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

박지원 박사는 UC Berkeley CBE Lab에서 건물 에너지 및 실내 환경 최적화를 연구하고 있으며, 특히 HVAC 시스템의 효율적인 제어와 에너지 절감 기술을 중점적으로 다루고 있다. 이번 인터뷰는 박사님이 수행하고 있는 연구 내용을 바탕으로 강화 학습 기반 HVAC 최적화의 가능성과 실무 적용을 위한 현실적인 한계점을 논의하는 것을 주요 목표로 삼았다. 특히, 칠러(Chiller) 등 HW 시스템의 영향, 시뮬레이션 기반 연구의 신뢰성 문제, Fixed Control의 실무 적용성 등을 중점적으로 논의하였다.

3. 인터뷰 사진



[그림 5] 박지원 인터뷰 사진

4. 인터뷰 전문

박사님 연구에서 MATLAB을 ANN 모델 구현 도구로 선택한 이유가 궁금합니다. Python(TensorFlow, PyTorch 등) 같은 다른 도구와 비교했을 때, MATLAB이 연구에서 더 적합했던 이유가 있었는지 궁금합니다. 특히, MATLAB의 어떤 기능이 ANN 설계, 시뮬레이션, 또는 EnergyPlus와의 통합 과정에서 이점을 제공했나요?
당시 연구실에서는 AI 기반 제어 연구가 막 시작되던 시기였으며, MATLAB이 연구 환경에서 더 일반적으로 사용되었기 때문에 선택하게 되었습니다. 현재는 연구실에서도 Python으로 전환하는 추세입니다. MATLAB을 선택한 주요 이유는 EnergyPlus와의 연동 문제 때문입니다. EnergyPlus는 기본적으로 물리 기반 시뮬레이션 도구이며, 당시에는 Python을

활용한 연동이 원활하지 않았습니다. MATLAB은 EnergyPlus와의 연동을 위해 사용된 BCVTB(Building Controls Virtual Test Bed) 같은 중간 인터페이스와의 호환성이 좋았기 때문에 선택되었습니다. 현재는 Python 기반의 pyenergyplus 같은 라이브러리를 통해 EnergyPlus 연동이 가능해졌기 때문에, MATLAB을 사용할 필요가 없어졌습니다. MATLAB의 단점은 코딩 유연성이 낮고, Python에 비해 오픈소스 라이브러리 활용이 제한적이라는 점입니다.
연구에서 비교군으로 Fixed Control을 설정하신 이유는 무엇인가요? Fixed Control은 고정된 설정 값으로 작동하는 비교적 단순한 방식인데, 이를 기준으로 ANN 모델의 성능을 비교한 이유가 무엇인지 궁금합니다.
현재 가장 일반적인 HVAC 제어 방식은 여전히 Fixed Control입니다. 한국에서는 약 95%의 건물이 Fixed Control을 사용하고 있으며, 이는 특정 시간에 HVAC를 단순히 켜고 끄는 방식으로 운영됩니다. 한국뿐만 아니라 미국에서도 Fixed Control이 가장 널리 사용되는 방식이며, 시스템이 단순하고 유지보수가 용이하다는 장점이 있습니다. 이러한 점을 고려하여 연구에서는 가장 보편적으로 사용되는 HVAC 운영의 표준 방식인 Fixed Control을 비교군으로 설정하였습니다.
저희 팀은 컴퓨터공학 전공자로서 HVAC 시스템의 물리적인 구성 요소에 대한 이해가 부족한 부분이 많습니다. 특히, 칠러 시스템이 HVAC 제어에서 어떤 역할을 하는지에 대해 자세히 알고 싶습니다.
건물의 에너지 소비는 칠러 시스템의 크기(사이징, Sizing)와 매우 밀접한 관련이 있습니다. 일반적으로 HVAC 시스템에서 칠러의 용량을 결정하는 방식은 특정 기준일(예: 여름철 가장 더운 날, 겨울철 가장 추운 날)의 부하를 만족할 수 있도록 설정하는 것입니다. 그러나 이러한 전통적인 사이징 방식에는 몇 가지 문제점이 존재합니다. 그중 가장 큰 문제는 과대 사이징(Over-Sizing)으로 인한 비효율성입니다. 예를 들어 가장 더운 날을 기준으로 칠러의 용량을 결정할 경우, 비교적 온화한 봄과 가을에는 냉방 부하가 현저히 줄어들게 됩니다. 이때 칠러가 너무 큰 용량을 가지고 있으면 부하 대비 낮은 효율로 운영되며, 이는 에너지 낭비로 이어질 수 있습니다. 특히 칠러의 성능은 COP(Coefficient of Performance, 성능 계수)에 의해 결정되는데, COP는 칠러의 용량 대비 현재 부하의 크기에 매우 의존적인 함수입니다.

예를 들어, 100의 용량을 가진 칠러가 있다고 가정했을 때, 실제 부하가 10이라면 칠러는 매우 낮은 효율로 작동할 가능성이 높습니다. 이는 에너지 절약과 비용 절감 측면에서 비효율적인 결과를 초래할 수 있습니다.

이러한 문제를 해결하기 위해서는 멀티 칠러(Multiple Chiller) 시스템을 고려하는 것이 중요합니다. 한 개의 대형 칠러를 운영하는 대신, 여러 개의 중소형 칠러를 운영하면 가변 부하 상황에서 더 높은 효율을 유지할 수 있습니다. 즉 두 개의 칠러를 운영하는 경우, 각 칠러의 용량을 절반으로 줄일 수 있으며, 부하가 작은 경우 일부 칠러만 가동함으로써 전체 시스템의 효율을 극대화할 수 있습니다.

즉, 칠러의 크기와 개수는 HVAC 최적화에서 매우 중요한 요소이며, 단순히 제어 알고리즘을 최적화하는 것만으로는 한계가 있을 수 있습니다. 아무리 좋은 강화학습 알고리즘을 사용하더라도, 이미 칠러의 용량이 너무 크거나 작게 설계된 경우, 최적의 제어가 이루어지더라도 물리적인 한계를 극복할 수 없을 것입니다.

HVAC 제어에서 강화학습이 효과적인 접근 방식인지 궁금합니다. 저희가 인터뷰 요청을 하게 된 까닭은 최근 이화여자대학교에 ‘이화 웨스트 캠퍼스’가 신설됨에 따라 건축학과의 ‘박지현’ 교수님의 추천으로 해당 건물을 배경으로 인공지능을 이용한 HVAC 시스템을 제안하게 되었습니다. 이에 따라, 건축학 박사 입장에서 바라본 적용 가능성이 얼마나 될지 궁금합니다.

현재 건물에서 고도화된 제어 방식을 도입하는 가장 큰 장애물 중 하나가 데이터 수집의 어려움이라고 생각합니다. 예를 들어 기존 AI 기반 최적화 방법, 특히 머신러닝이나 MPC(Model Predictive Control) 기반 접근법은 대량의 고품질 데이터가 필요하지만, 대부분의 건물에서는 이러한 데이터를 충분히 확보하기 어렵다는 점이 문제로 작용하고 있습니다.

이러한 상황에서 강화학습(Deep Reinforcement Learning, DRL)이 데이터 수집의 어려움을 극복하는 데 있어 오히려 유리한 접근법이 될 수 있다고 예상합니다. 기존의 데이터 기반 최적화 방식은 충분한 학습 데이터가 없는 경우 성능이 저하될 수 있지만, 강화학습은 환경과의 상호작용을 통해 데이터를 직접 생성하면서 학습할 수 있기 때문에, 초기 데이터가 부족한 상황에서도 효과적으로 적용될 가능성이 높을 것입니다.

EnergyPlus에서 “Office Medium” 모델을 사용하여 실험을 진행할 경우, 이를 저희 캠퍼스 환경에 일반화할 수 있을까요? 추가적으로 고려해야 할 요소가 있을까요? “Office Medium” 모델을 사용해도 실제 캠퍼스 건물 환경과 유사한 결과를 얻을 수 있을까요? 만약 그렇지 않다면, 캠퍼스 환경에 맞게 일반화하기 위해 어

떤 추가적인 조정이나 수정이 필요할까요?

시뮬레이션은 매우 유용한 방법이며, 특히 한국에서는 많이 활용되는 접근법이라고 설명하셨습니다. 그러나 동시에 시뮬레이션에는 많은 불확실성이 내포되어 있으며, 이를 완전히 신뢰하기 어렵다는 점을 강조하고 싶습니다.

예를 들어, 시뮬레이션에서는 사람들이 계절에 따라 항상 동일한 옷을 입는다고 가정하거나, 모든 사람이 동일한 신체 발열량을 가진다고 가정하는 등의 인위적인 설정이 포함될 수 밖에 없습니다. 또한, 건물 환경과 관련해서도 창문을 열지 않는다고 가정하거나, 환기가 일정한 비율로 유지된다고 가정하는 등 현실과 다른 요소들이 존재할 수 있습니다.

이러한 가정들은 실제 건물 운영 환경과 차이가 크기 때문에, 시뮬레이션 결과가 현실에서 그대로 적용될 수 있는지에 대한 확신을 가지기 어렵다는 점이 문제로 지적되었습니다. 실제 저의 석사 과정 동안 지속적으로 시뮬레이션 연구를 진행하면서도, 이러한 불확실성으로 인해 결과의 신뢰성을 스스로 확신하기 어려웠습니다.

결론적으로 시뮬레이션 결과를 맹목적으로 신뢰하기보다는, 현실적인 밸리데이션(Validation) 과정을 거쳐 신뢰할 수 있는 방향으로 연구를 진행하는 것이 중요하다는 점을 강조하고 싶습니다.

[인터뷰 6. Kazu Komoto, Michael Kahane]

1. 인터뷰 개요

다른 기관과 달리, Osaro는 본사 방문 시 본 팀의 세부 일정을 미리 기획해주었고, 이에 따라 진행된 일정은 <표 6>과 같다.

인터뷰 장소	Osaro
인터뷰 일시	2025년 1월 15일 오후 1시 (GMT-8)
세부 일정	1부: Kazu Komoto(VP of APAC Sales & Solutions Engineering)의 Osaro 회사 소개 2부: Fahim(Solutions Support Engineer)의 로봇 시연 3부: Michael(Osaro Co-Founder and CTO)와의 ML/RL Q&A - 인터뷰 진행

<표 6> Osaro Kazu Komoto, Michael Kahane 인터뷰 일정

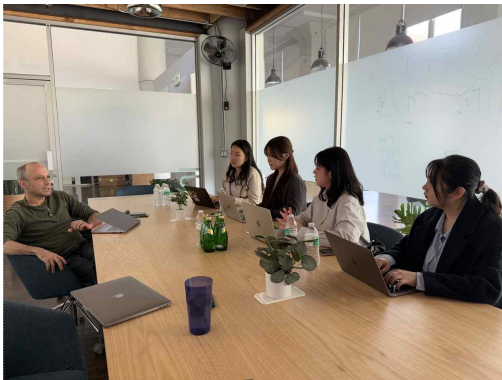
2. 인터뷰 기관 및 인물 소개

Osaro는 머신러닝과 강화학습을 활용하여 물류, 창고 관리, 포장 등 다양한 산업 분야에서 로봇 자동화 시스템을 개발하는 미국의 첨단 기술 기업이다. 특히, 컴퓨터

비전과 인공지능을 결합한 로봇 컨트롤 솔루션을 통해 기존의 물류 및 제조 자동화 시스템이 해결하기 어려운 복잡한 작업을 수행할 수 있도록 하는 솔루션을 구축하는 데에 기여하고 있다.

Kazu Komoto는 Osaro의 아시아·태평양 지역(APAC) 영업 부문 임원이자 Solution Engineer로, 컨택 메일을 통해 인터뷰를 진행하게 되었다. Q&A에 응해준 Michael Kahane은 OSARO의 공동 창업자이자 CTO로, AI와 알고리즘 개발, 로봇 통합, 클라우드 및 On-premise 인프라 구축을 담당하며, 고객과 협력하여 로봇 솔루션 성능 최적화에 주력하고 있다. Osaro Vision과 Osaro Pick에 적용된 Osaro의 독자적인 머신러닝 및 강화학습 기술은 팀의 탐사 주제와 직접적인 연관은 없지만, 강화학습 알고리즘에 대한 심층적인 이해를 돕는 데 유용할 것으로 판단하였다. 학교 수업이나 논문 리뷰를 통해 배운 강화학습 원리가 실무에서 어떻게 적용되는지를 탐구하기 위해 인터뷰를 진행했으며, 이를 통해 팀의 기술적 지식을 확장하는 데 도움이 되었다.

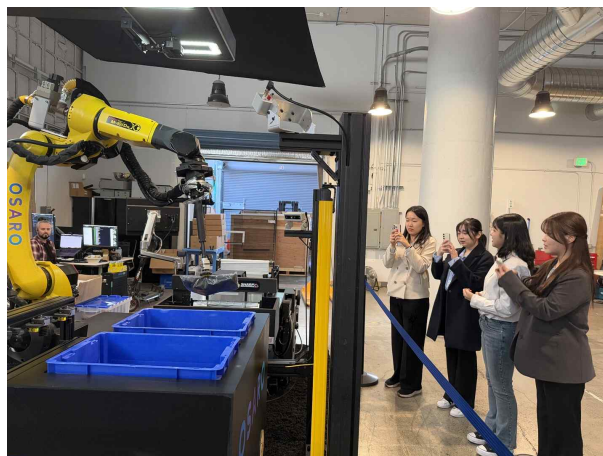
3. 인터뷰 사진



[그림 6] Osaro 인터뷰 사진1



[그림 7] Osaro 인터뷰 사진2



[그림 8] Osaro 로봇 시연 사진

4. 인터뷰 전문

Osaro 회사에 대해 설명 부탁드립니다.

Osaro는 강화학습의 핵심 요소인 Observation, State, Action, Reward, Observation의 약어로, 회사의 목표와 비전을 잘 상징합니다. Osaro는 인공지능 기술을 통해 복잡한 문제를 해결하고, 스스로 학습하여 발전하며, 이전에 경험하지 않은 문제를 로봇이 해결할 수 있도록 돕는 기술 개발에 주력하고 있습니다.

현재 Osaro의 로봇은 한국의 쿠팡, 현대, CJ, 아모레퍼시픽과 같은 대형 e-commerce 및 물류 회사에서 사용되고 있습니다. 미국에서는 높은 노동 비용으로 인해 상용화가 빠르게 이루어지고 있는 반면, 한국과 일본에서는 로봇이 인건비보다 비싸기 때문에 아직 대형 이커머스 회사 중심으로 사용되고 있습니다.

과거에는 AI뿐만 아니라 로봇, 컨베이어 벨트 등 전체 솔루션을 제공하며 로봇을 판매하는 방식으로 운영했지만, 현재는 RaaS(Robot-as-a-Service) 모델을 도입하여 Osaro의 로봇 소프트웨어를 대여하는 서비스를 제공합니다.

Osaro의 대표적인 4가지 솔루션은 Bagging System, Depalletizing System, Piece Picking System, Kitting System입니다.

Bagging System은 로봇이 물건을 빠르고 정확하게 가방이나 포장용 봉투에 담는 자동화 시스템입니다. 크기와 형태가 다양한 물체를 인식하며, 비정형적이고 불규칙한 모양의 물건도 처리할 수 있는 강력한 물체 인식 기술을 포함하고 있습니다.

Kitting System은 로봇이 서로 다른 6가지 물건을 조합하여 하나의 키트를 완성하는 시스템입니다. 이 때 로봇이 가지는 주요 과제는 3가지로, 첫 째 키트에 넣는 물건이 매번 달라져야하고 둘 째 매번 달라지는 물건 구성을 로봇이 인지하고 패턴 매칭을 통해 처리해야하며 셋째 3D 탐지 기술을 활용하여 물건을 상자에 정돈되게 배치하며 키트를 완성해야합니다. 또한, 새로운 물건에 대한 이미지를 즉시 인식하고 스스로 학습할 수 있는 기술도 포함되어 있습니다.

Piece Picking System은 로봇이 물체를 인식하여 집어 올린 후 바코드를 스캔하고, 이어서 썰링 작업까지 처리하는 시스템입니다. 탐지가 어려운 컵라면 포장 상자 같은 물체도 효과적으로 처리할 수 있는 기술이 특징입니다.

Depalletizing System은 혼합된 상자를 인식하고 팔레트에 쌓인 외부 물체(박스 커터, 테이프 디스펜서 등)를 식별하여 경고를 발합니다.

Osaro 기술의 차별점 모든 머신러닝 모델을 물건별로 새로 커스터마이징을 할 필요가 없다는 점입니다. 일반적으로 머신러닝에서는 데이터를 수집하고 라벨링하는

과정이 필수적이지만, 물류 회사에서는 매 시즌 상품이 바뀔 때마다 이를 학습시키는 것이 시간적으로나 비용적으로 비효율적입니다. 그렇기 때문에 Osaro의 기술은 완전한 최적화가 이루어지지 않은 자동화된 창고에서 G2R(goods-to-robot) 기술을 지원하여 더 효율적인 작업이 가능합니다. 여기에 사용된 Osaro의 기술이 AutoModel™입니다.(여기에 사진) AutoModel™은 데이터 수집, 학습, 평가, 배포까지의 과정을 자동화한 인프라로, 기존 시스템에서 인간의 개입이 필수적인 Annotate(데이터 라벨링) 단계를 없애 인간의 개입이 없고 시스템이 자율적으로 데이터를 수집하고 학습하며 배포합니다. 따라서 작업을 중지시키는 다운 타임이 없는 기술입니다. 또한, Osaro의 비전 소프트웨어인 OSARO SightWorks™는 키팅, 배킹, 인덱션, 피스 피킹 시스템에 활용되며, 물류 업체와 통합하여 효율적인 주문 처리 운영을 지원합니다. 물류 회사의 경우 학습 시간이 부족한 환경이 많기 때문에 Osaro의 기술은 보고 스스로 학습하여 빠르게 사용할 수 있다는 점에서 큰 장점을 가집니다.
Osaro의 기술은 새로운 아이템이나 작업이 추가되더라도 로봇이 멈추지 않고 자동으로 학습한다고 들었습니다. 이 부분이 전이 학습(transfer learning)과 같은 기술과 관련이 있나요?
Osaro의 기술이 새로운 아이템이 들어와도 로봇이 멈추지 않고 자동으로 학습할 수 있는 것은 전이 학습을 썼기 때문이 아닙니다. 로봇은 아이템을 관찰하며 데이터를 수집하고, 이를 바탕으로 어디를 집어야 하는지 어느 정도 인식할 수 있습니다. 그러나 샘플 범위를 벗어난 데이터가 있을 경우 기술의 성공률이 떨어질 수 있습니다. 하지만 95% 이상인 성공률이 약 70%까지 떨어진다고 해서 시스템이 쓸모없게 되는 수준이 아니며, 여전히 작동을 이어나갈 수 있습니다. 또한, 대부분의 고객들은 작업 흐름에 방해가 되기 때문에 시스템을 멈추고 데이터를 재수집하고 재학습하기를 선호하지 않습니다. Osaro의 기술은 이러한 점들을 반영하여 70%의 성공률로 계속 멈추지 않고 운영되면서 동시에 데이터를 수집합니다. 이후 수집한 데이터를 활용해 모델을 재학습하고 배포하면 시간이 지나면서 성공률이 다시 상승하는 패턴을 만들게 되는 것입니다. 추가적으로, Osaro는 최근 zero-shot learning을 도입하였습니다. zero-shot learning은 작업 데이터 중 마지막 20-30개의 데이터를 버퍼로 사용하여 새로운 모델을 완전히 학습시키기 전에도 성능을 약간 개선할 수 있는 기술입니다. 이러한 방식은 질문하신 전이학습의 사용 없이도 성공률 하락을 최소화하고 점진적으

로 성능을 회복할 수 있게 합니다.
Osaro CTO께서 ML과 RL 관련 연구에서 주목하고 있는 새로운 트렌드가 있는지 궁금합니다.
최근 휴머노이드 기술에 대한 관심이 높아지고 있습니다. 비록 이를 실제 환경에 적용하고 배포하는 단계(deploy)에 이르지 못했지만, 많은 연구자들이 이 분야에 집중하고 있습니다.
Osaro 기술에서 사용되는 시뮬레이터나 학습 환경은 커스텀 기술인가요?
Osaro는 주로 Unity 기반의 환경에서 시뮬레이션을 진행합니다. Unity 환경에서는 로봇이 실제로 작업을 수행할 수 있는 가상 환경을 구축할 수 있으며, 이는 특별하게 조정된 방식으로 시뮬레이션을 돕습니다. 예를 들어, Kitting 시스템에서는 상자에 너무 많은 부품이 들어가거나 공간을 초과하는 오버플로우가 발생하기 쉬운데, 이를 해결하기 위해 Unity에서 특별히 조정된 방식으로 시뮬레이션을 진행하고 이후에 최적화합니다.
Osaro CTO께서 생각하시는 Osaro AutoModel™만의 차별점이 무엇이라고 생각하시나요?
기존 방식과 비교했을 때, Osaro 기술의 가장 큰 개선점은 객체 등록이 필요 없다는 점입니다. 기존 머신러닝 시스템은 모든 객체를 개별적으로 등록하고 데이터 라벨링(Annotate) 단계를 거쳐야 했기 때문에 많은 시간이 소모되었습니다. 그러나 Osaro는 이러한 단계를 제거하고 자동화하여 높은 시간 효율성을 제공합니다. 두 번째로, Osaro 기술은 견고성이 크게 개선되었습니다. 기존 방식에서는 등록된 객체라도 아이템이 약간 구부러지거나 불완전한 상태이거나 조명 조건이 달라지는 등 환경 변화가 있을 경우 객체 인식에 문제가 발생하는 경우가 많았습니다. 하지만 Osaro의 머신러닝 기반 비전 시스템은 이러한 한계를 극복하여 훨씬 더 안정적이고 신뢰할 수 있는 성능을 제공합니다.
로봇 시연
Mixed-case Depalletization System Mixed-case Depalletization System은 기존의 로봇공학 기술이 어려워했던 혼합

팔레트에서의 재고 프로세스 자동화를 가능하게 하는 로봇 시스템입니다.

작동 방식: ① 박스 안에 여러 아이템이 담겨 있으며, 위에 장착된 3D 스캐닝 카메라와 센서가 물건의 위치와 크기를 인식합니다. ② 로봇은 센서로부터 얻은 정보를 바탕으로 아이템을 들어 올려 목적지에 맞는 박스에 분리하여 넣습니다.

Pick & Place Solution

픽 앤 플레이스 솔루션은 선도적인 자동 포장 장비와 통합하여 품목 포장을 자동화하는 로봇 시스템입니다.

작동 방식: ① 로봇은 물건의 들 수 있는 부분을 포인트로 파악하고, 해당 부분을 잡았을 때 들어올릴 수 있는 확률을 계산합니다. ② 이 확률에 따라 여러 번 들어보면서 가장 성공률이 높았던 부분을 선택하여 집습니다.

만약 물건을 놓쳤을 경우, 센서를 활용하여 보완합니다. 예를 들어, 레이저로 포장지에 물건이 제대로 들어갔는지 확인하거나 무게 측정을 통해 물건이 정확히 집혔는지 확인하는 과정을 센서를 통해 거칩니다.

머신러닝 AI 기술을 사용하고 있음에도, 기술에 센서를 굉장히 많이 사용하고 있는 것 같은데 특별한 이유가 있나요?

머신러닝과 강화학습은 100% 완전할 수 없기 때문에 센서와 같은 물리적인 기계로 보완이 필수적입니다. 물리적 기계를 사용함으로써 인간의 개입이 필요한 부분을 최소화하고, 시스템이 보다 효율적으로 작동할 수 있습니다.

사용되는 예: Bagging(OSARO and Zenni Deploy Robotic Bagging System에서 안경 케이스를 집고 바코드를 스캔할 수 있도록 회전시키는 기술), Kitting

[인터뷰 7. Ashesh Chattopadhyay]

1. 인터뷰 개요

인터뷰 1. Ashesh Chattopadhyay	
인터뷰 장소	비대면 Zoom
인터뷰 일시	2025년 1월 21일 오후 5시 (GMT-8)

<표 7> UCSC Ashesh Chattopadhyay 인터뷰 일정

2. 인터뷰 기관 및 인물 소개

University of California, Santa Cruz(UCSC)는 미국 캘리포니아주의 대표적인 연구 중심 대학으로, 지속 가능한 캠퍼스 운영과 친환경 에너지 시스템 구축에 앞장서고 있다. UCSC는 캘리포니아대학교 시스템의 일원으로, 2045년까지 탄소 중립을 달성하기 위한 UC 시스템의 목표에 맞춰 캠퍼스 내 빌딩 에너지 관리 시스템(BEMS,

Building Energy Management System)을 도입하고, HVAC 최적화, 재생 가능 에너지 활용, 데이터 기반 에너지 효율성 개선 등 다양한 노력을 기울이고 있다.

UCSC의 Ashesh Chattopadhyay 교수는 과거 연구인 “Data-driven surrogate models for climate modeling: application of echo state networks to the multi-scale Lorenz system as a test case”에서 기후 모델의 복잡성을 단순화하기 위한 데이터 기반 서로게이트 모델링을 연구했다. 이 연구는 Echo State Network(ESN)와 같은 머신러닝 기법을 활용해 전통적인 기후 모델의 연산 비용 문제를 해결하면서도 정확도를 유지할 수 있는 방법론을 탐구한다. 이를 통해 복잡한 물리 기반 모델의 한계를 극복하고, 데이터 기반 접근법이 가진 잠재력을 입증하였다. 이에 기후 모델과 같이 복잡한 물리 데이터를 사용하는 HVAC 제어 시스템에 적용한 컴퓨터공학적 아이디어를 목적으로 인터뷰를 진행하게 되었다.

3. 인터뷰 사진



[그림 9] Ashesh Chattopadhyay 인터뷰 사진

4. 인터뷰 전문

교수님께서 작성하신 ‘FourCastNet: A Global Data-Driven High-Resolution Weather Model Using Adaptive Fourier Neural Operators’에서 FourCastNet은 날씨를 예측하는 AI 모델로 연산이 적어 속도가 빠르며 이에 따라 에너지 소모가 적어 환경에 이롭지만 기존 수치적 기상 예보 Numerical Weather Prediction(NWP) 모델의 경우 연산량이 많아 계산 속도가 느리다고 언급 해주셨는데요. 저희가 연구하고자 하는 HVAC 제어에서도 MPC와 AI 모델이 유사한 관계를 가지고 있습니다. 이에 관련하여 AI 모델이 물리 기반 모델 대비 에너지 효율적이라는 정확한 평가 기준이 있을까요?

이것은 시뮬레이션하고자하는 시스템의 종류에 따라 다릅니다.

MPC 제어는 예측을 제공하는 순방향 모델로 시작합니다. 이 예측을 바탕으로 관찰을 하고, 관찰된 차이를 바탕으로 학습하여 제어를 수행합니다. 이와 비슷한 맥락으로 기상 예측이나 다른 과학적 시스템의 제어를 떠올릴 수 있습니다. 날씨의 경우 제어를 고려할 필요는 없지만 매우 비싼 순방향 모델을 실행하는 것을 포함한다고 볼 수 있습니다. HVAC 시스템 역시 비싼 순방향 모델을 실행해야 합니다. 열 교환기가 있는 전통적 HVAC 시스템을 가정할 때 매우 비싼 방정식을 풀어야 합니다. 왜냐하면 열 교환기 내부에 있는 유체의 운동량을 기술하는 방정식과 에너지 방정식이 결합되어 있기에 유체의 흐름을 이해하고 계산하는 과정은 매우 많은 계산을 필요로 합니다. 따라서, 이 과정에는 슈퍼 컴퓨터가 수반되며, 많은 비용이 들게 됩니다.

만약 전체 시뮬레이션을 대체할 모델을 개발한다면 이 모델을 훈련하는데 많은 에너지가 소모됩니다. 그러나 일반적으로 모델을 훈련하는데 드는 에너지를 계산하지 않습니다. 사람들은 모델이 예측 혹은 추론하는 상황에서 소비되는 에너지만 계산하게 됩니다. 한 번 훈련된 모델은 다양한 상황이나 다양한 제어 실험에서 계속 실행될 수 있습니다. 그래서 학습의 전반적 비용은 훈련된 모델을 수백, 수천 번 사용하는 경우, 그 값을 소거할 수 있기 때문입니다. 이처럼 계산했을 때, 전통적인 수치 해석적 해결책에 비해 AI 기반 모델이 에너지 소비 측면에서 수십배는 저렴합니다.

물론 이에 몇가지 제한 사항이 있습니다. 특정 환경에서 모델을 훈련시킨 경우, 그 환경에서 반복적으로 추론을 실행해야 합니다. 만약 환경이 변한다면, 모델은 그 환경에 잘 일반화되지 않을 가능성이 높습니다. 그래서 훈련 비용을 무시한 경우 다양한 상황을 고려해야한다면 계산이 훨씬 복잡해집니다. NWPS의 맥락에서는 그 이후에도 반복적으로 같은 환경 혹은 유사한 시나리오에서 사용되기에 훈련 비용을 무시할 수 있습니다.

물리 기반 모델보다 AI 모델을 더 효율적으로 만들 수 있는 전략이 있나요?

기본적으로 AI 모델들은 물리 기반 모델보다 훨씬 효율적이기에 별도의 전략이 필요치 않습니다. 예를 들어, 모델이 일반적으로 수행하는 부동 소수점 연산의 수나 실행 시간의 측면을 생각했을 때 훨씬 효율적입니다.

이를 비교할 수 있는 과정을 설명드리겠습니다. 사람들은 이를 비교할 때 일반적으로 단일 GPU에서의 실행 시간과 물리 기반 모델을 실행하는데 필요한 총 CPU의 시간을 비교하는 방식을 사용합니다.

물리 기반 모델은 단일 CPU에서 실행될 수 없습니다. 이를 여러 CPU 노드에 걸

쳐 슈퍼 컴퓨터에서 병렬 처리할 수 있어야 합니다. 우리가 고려하는 총 실행시간은 일반 시간이 아닌 총 CPU 시간입니다. 그리고 우리는 AI 모델을 실행하는데 필요한 총 GPU 시간과 비교합니다. 이 비용은 보통 수십 배 더 저렴합니다. 이를 통해서 사람들은 에너지 소비를 대리 측정할 수 있습니다. 실제로 에너지를 계산하지는 않지만 GPU 시간 수를 대리 지표로 사용합니다. 이처럼 GPU와 CPU 시간 수를 에너지 소비의 대체제로 사용해 계산할 수 있고 이를 통해 대부분의 AI 모델이 효율적임을 비교할 수 있었습니다.

HVAC은 에너지 보존 법칙, 열역학 법칙 등과 같이 물리적 제약이 뚜렷한 시스템으로, 교수님께서 연구하신 Physics-Informed Machine Learning(PIML)을 적용하기에 적합한 분야라고 생각합니다. HVAC에서 PIML을 적용한다면, 기존 데이터 기반 모델과 비교해 어떤 이점이 있을까요?

우선 답변을 드리기 전, 여러 실험에서 데이터는 많으면 많을수록 좋다는 것을 인지해야 합니다. 그래서 실험이나 시뮬레이션을 통해 많은 양의 데이터를 가지고 있을 수 있다면 물리적 제약은 자동으로 충족됩니다. 따라서 물리적 제약은 과학 분야에 적용된 머신러닝에 큰 역할을 하게 됩니다. 과학에서는 일반적으로 데이터가 적은 경우가 많기 때문에 수백만 건의 실험을 할 수 없습니다. 이는 HVAC 실험에서도 마찬가지입니다.

데이터가 부족한 경우, 물리적 제약은 큰 역할을 합니다. 충분한 데이터가 없기 때문에 데이터로부터 이 제약을 학습할 수 없습니다. 하지만 에너지 보존 혹은 질량 보존과 같은 특정 조건 하에서는 이런 제약들을 적용할 수 있습니다.

특히 HVAC 시스템과 같이 좋은 방정식을 사용할 수 있는 경우는 물리적 제약을 더욱 이용하기 좋습니다. 데이터가 많을 때는 데이터로부터 이런 제약을 학습할 수 있지만, 데이터가 적을 경우 물리적 제약을 통해 머신러닝 모델의 정확도를 크게 향상시킬 수 있습니다.

예를 들어, 실험을 통해 데이터를 수집하거나 고해상도의 HVAC 시뮬레이션을 실행해 많은 데이터를 사용할 수 있게되었다면 물리적 제약이 모델에 포함되는지는 그다지 중요하지 않습니다. 데이터는 물리적 시뮬레이션 혹은 실제 실험을 기반으로 얻은 것이며, 이 데이터는 이미 이러한 제약들을 만족하기 있기 때문에 모델은 데이터 자체에서 제약을 학습할 수 있습니다.

하지만 대부분의 과학 문제에서는 실험실에서 데이터를 수집하기 위해 실제로 수행할 수 있는 실험의 수가 제한되어 있고, 제한된 자원으로 수행할 수 있는 시뮬레이션의 수도 많지 않습니다. 따라서 과학 문제에 적용된 머신러닝에 사용할 데이터를 적은 것이 일반적이고 이때 물리적 제약을 적용하는 것은 머신러닝 모델의

정확도를 크게 향상시킬 수 있습니다.
교수님이 하신 연구 중 HVAC 제어 시스템에 적용할 수 있는 기술과 구체적인 연구 사례와 함께 설명해주실 수 있나요?
제가 하는 많은 작업은 물리 시스템에 대한 대체 모델을 구축하는 것입니다. 물리 시스템은 비용이 많이 드는 경우가 많습니다. 그래서 비싼 모델을 저렴한 머신러닝 기반 모델로 대체하는 일들을 진행하고 있습니다. 이것은 MPC 기반 제어에 매우 유용합니다. MPC 알고리즘은 다양한 것들이 필요하지만 가장 우선적으로 필요한 것은 모델이며 이 모델은 저렴해야 합니다. 그래서 이 모델을 사용해 매우 빠르게 순방향 시뮬레이션을 수행할 수 있습니다. 자동 회귀 시간 단계 모델링과 같이 제가 하는 작업들은 미래를 예측하는데 매우 유용합니다. 이 모델들은 시스템에 쉽게 통합할 수 있어서 기존의 HVAC 시뮬레이션을 자동 회귀 모델로 대체할 수 있습니다. 또한 이는 시간 예측을 할 때 사용할 수 있습니다. 또한 몇 번의 단계를 거치면 관측치를 얻을 수 있고 이 결과를 바탕으로 제어를 수행할 수 있습니다. 그래서 이런 모델을 영구적으로 실행할 필요는 없습니다. 자동 회귀 모델링의 아이디어와 제가 작성한 논문을 참고하여 여러분들이 직접 HVAC 데이터에 적용할 수 있습니다. 여기에 주로 사용할 수 있는 기술이 데이터 동화입니다. 데이터 동화는 예측 모델과 주변 환경에서 얻은 희소하고 노이즈가 많은 관측치를 결합하는 방법입니다. 이를 통해 시스템은 더 나은 예측을 할 수 있습니다. 이 동화의 방법에는 여러가지가 있는데 저는 주로 앙상블 칼만 필터 방식을 사용합니다. 앙상블 칼만 필터가 무엇인지 모를 수 있지만 검색해보면 이해가 수월할 것입니다. 이는 시스템의 더 나은 추정치를 도출하기 위해 예측 모델과 희소한 관측치를 사용하는 간단한 알고리즘입니다.
HVAC 시스템에는 외부 기후 조건, 내부 구조적 특성 및 사용자 행동 패턴과 같은 요소들이 복잡하게 상호작용하는데 이에 강화학습이 어떤 과정으로 적용될 수 있을지 교수님이 연구하신 방법과 비교하여 설명해주실 있을까요?
저희가 알아야 할 사실은 HVAC에서의 여러 상호 작용들은 복잡하게 보이지만 사실 그렇게 어렵지는 않다는 것입니다. 외부 기후 조건은 기본적으로 외부 온도의 영향이 크며, 내부 조건은 시스템의 동적인 상황을 기본적으로 말합니다. 사용자 행동 패턴의 경우, HVAC 시스템은 실제로 사용자 행동에 따라 달라지지 않습니다. 오히려 사용자가 노출되는 기후 조건에 따라 달라집니다. 예를 들어 고급 에어컨은 HVAC 시스템입니다. 작동 방식은 대략 다음과 같습니다. 우리는 에어컨의 온도를 설정합니다. 그것이 사용자 행동 패턴입니다. 우리가

에어컨 온도를 설정한다면 이것은 사용자의 입력입니다. 그래서 HVAC 시스템은 외부 온도가 사용자 입력과 동일하도록 유지하는 것을 목표로 작동해야 합니다. 이러면 이제 상황은 온도를 제어 매개변수로 하는 간단한 제어 문제가 됩니다. 시스템은 온도를 사용자가 설정한 온도를 맞추기 위해 행동을 취할 것입니다. 이때 모델링 과정을 말씀드리자면, 먼저 차가운 유체를 불어넣기 위해 HVAC 시스템의 컴프레서 내부에서 유체의 속도를 알려주는 일련의 방정식을 풀어야 합니다. 이 시스템은 외부 기후를 확인할 것입니다. 그리고 시스템은 사용자의 요구 온도를 맞추기 위해 이 과정을 계속 반복할 것입니다. 이때 시스템의 속도의 흐름 때문에 온도가 변하기 때문에 순방향 모델인 시스템의 속도 변화 과정과 온도 측정 값이 중요하게 됩니다.

만약 원하는 온도보다 높다면 계속해서 순방향 모델을 실행할 것입니다. 모델의 외부 온도를 통합하는 방법은 데이터 동화입니다. 아까 질문에 답변드린 것과 같이 앙상블 칼만 필터 기법이 그 예입니다. 이는 시스템의 관찰을 모델에 통합하는 쉬운 알고리즘입니다. 순방향 모델은 머신러닝으로 대체될 수 있습니다. 여기서 우리는 강화 학습 기반 제어를 만드는 것과 같이 다양한 시도를 할 수 있습니다. 생각해보면 이는 앞서 설명한 것과 비슷한 일을 한다고 볼 수 있습니다. 보상 함수가 외부 온도와 원하는 온도 사이의 차이가 얼마나 작은지에 따라 결정됩니다. 모델이 계속 실행되고 있을 때, 손실 함수가 높으면 보상을 받지 못하고, 손실 함수가 낮으면 보상을 받습니다. 이것이 강화 학습의 아이디어입니다.

그래서 이때 사용자 패턴 등을 고려할 수 있습니다. 이는 복잡하지 않습니다. 설정하고자 하는 온도를 고려하는 것이 전부이며 이것이 기본적인 사용자와의 상호작용의 전제입니다. 복잡한 다른 계층은 없습니다. 사용자가 불편함을 느끼는지 아닌지를 알아내고 변경하는 것이 아니며, 실제로 온도를 설정하고 모델은 온도를 유지하기 위해 공기 흐름을 제어하는 것에 집중하면 됩니다.

[부록 2] 연구 과정

(1) 환경 구축 및 초기 설정

① 필수 요구 사항 - MacOS 기준

- Python 3.12
- VSCode (Visual Studio Code)
- Git
- Homebrew
- EnergyPlus (관련해서는 ③에 추가 설명)

② 프로젝트 클론 및 가상환경 생성

해당 과정은 프로젝트의 종속성을 효율적으로 관리하고, 실험 환경을 통일하기 위해 가상환경을 설정하는 과정이다.

프로젝트에서 사용하는 라이브러리는 지속적으로 업데이트되며, 각 프로젝트마다 요구하는 라이브러리 버전이 다를 수 있다. 가상환경을 활용하면 특정 프로젝트에 필요한 라이브러리와 버전을 독립적으로 관리할 수 있어, 다른 프로젝트와의 종속성 충돌을 방지할 수 있다.

이를 위해 requirements.txt 파일을 활용하여, 연구자가 동일한 환경을 쉽게 재현할 수 있도록 하고, 협업 과정에서도 일관된 실행 환경을 유지할 수 있도록 한다. requirements.txt에는 프로젝트 실행에 필요한 모든 라이브러리와 해당 버전이 명시되어 있으며, 이를 설치함으로써 연구 환경을 통일하고 실험의 재현성을 확보할 수 있다.

본 프로젝트에서는 가상환경을 통해 EnergyPlus, Sinergym, 강화학습 알고리즘 관련 라이브러리 등을 독립적으로 관리하며, 이를 통해 안정적인 연구 환경을 구축하고자 한다.

```
# 프로젝트 저장소를 로컬에 복제
git clone https://github.com/seoahS01/global-frontier-24.git
cd global-frontier-24

# 가상환경 생성 및 활성화
python3 -m venv sinergym_env
source sinergym_env/bin/activate

# 필요한 패키지 설치
pip install -r requirements.txt
```

③ EnergyPlus 설치 및 환경변수 등록

HVAC 시뮬레이션을 진행하기 위해 EnergyPlus를 설치하고, 이를 Python과 연동할 수 있도록 환경 변수를 설정해야 한다. (공식 다운로드 페이지: <https://energyplus.net/downloads>)

EnergyPlus는 건물 에너지 시뮬레이션을 수행하는 소프트웨어로, 공식 웹사이트에서 최신 버전을 다운로드하여 설치할 수 있다. EnergyPlus를 설치한 후, 시스템에서 올바르게 인식되는지 확인하기 위해 설치 경로를 확인하는 과정이 필요하다. 일반적으로 macOS에서는 /Applications/EnergyPlus-24-2-0 경로에 기본적으로 설치되며, 해당 경로를 환경 변수에 추가하여 Python과 연동할 수 있도록 설정해야 한다.

환경 변수를 설정하는 과정은 일시적인 설정과 영구적인 설정 두 가지 방식으로 나뉜다. 터미널에서 export 명령어를 사용하여 환경 변수를 설정하면, 해당 터미널 세션에서만 적용되며, 터미널을 닫으면 설정이 사라진다. 따라서 매번 환경 변수를 다시 입력해야 하는 불편함이 발생할 수 있다. 이를 방지하기 위해, 영구적으로 환경 변수를 유지하려면 설정 파일(~/.bashrc 또는 ~/.zshrc)을 수정해야 하며 명령어는 다음과 같다.

```
echo 'export EPLUS_PATH="/Applications/EnergyPlus-24-2-0"' >> ~/.zshrc
echo 'export PYTHONPATH="$PYTHONPATH:/Applications/EnergyPlus-24-2-0"' >>
~/.zshrc
source ~/.zshrc # 변경 사항 적용

# 혹은

echo 'export EPLUS_PATH="/Applications/EnergyPlus-24-2-0"' >> ~/.bashrc
echo 'export PYTHONPATH="$PYTHONPATH:/Applications/EnergyPlus-24-2-0"' >>
~/.bashrc
source ~/.bashrc # 변경 사항 적용
```

④ Target Building 및 Weather 설정

Sinergym은 EnergyPlus 기반의 시뮬레이션 환경을 제공하지만, 기본적으로 포함된 빌딩 및 날씨 파일이 제한적이다. 따라서 설치된 Sinergym 라이브러리에 타겟 파일이 기본값으로 존재하지 않기 때문에, 해당 설정을 생략하고 실행할 경우 FileNotFoundError가 발생할 수 있다.

이를 방지하기 위해, 각자의 EnergyPlus 설치 폴더 내에 포함된 파일을 활용

하여 타겟 빌딩 모델과 날씨 데이터를 설정하는 것이 보다 효율적인 방법이다. 따라서 본 프로젝트에서는 EnergyPlus의 ExampleFiles에 존재하는 빌딩 모델을 변환하고, 별도로 제공되지 않는 날씨 파일은 외부에서 다운로드하여 적용하는 방식을 사용하고자 한다.

타겟 빌딩 모델은 **ASHRAE901_OfficeLarge_STD2019_Denver**이다. 해당 파일은 EnergyPlus 내부의 ExampleFiles 디렉토리에 포함되어 있으므로, 이를 그대로 사용할 수 있다. 하지만 Sinergym과의 호환성을 위해 .idf 파일을 .epJSON 형식으로 변환하는 과정이 필요하다.

```
# EnergyPlus 설치 경로 내 ExampleFiles 디렉토리에서 'Office'라는 단어가
# 포함된 파일을 검색
ls /Applications/EnergyPlus-24-2-0/ExampleFiles | grep "Office"
# EnergyPlus의 .idf 파일을 .epJSON 형식으로 변환
/Applications/EnergyPlus-24-2-0/energyplus --convert-only
/Applications/EnergyPlus-24-2-0/ExampleFiles/ASHRAE901_OfficeLarge_STD2019_Denver.idf

# 변환된 .epJSON 파일을 Sinergym 빌딩 데이터 디렉토리로 복사
cp
/Applications/EnergyPlus-24-2-0/ExampleFiles/ASHRAE901_OfficeLarge_STD2019_Denver.epJSON \
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/data/buildings/
```

타겟 날씨 데이터는 **KOR_Inchon.471120_IWEC**이다. 해당 파일은 EnergyPlus 기본 ExampleFiles 디렉토리에 포함되어 있지 않으므로, 별도로 다운로드하여 적용해야 한다. 따라서, 이화여자대학교 캠퍼스와 기후가 유사한 지역인 '인천'의 기후 데이터를 적용하기 위해 EnergyPlus 공식 웹사이트에서 .epw 및 .ddy 파일을 다운로드하는 과정이 필요하다. (공식 다운로드 페이지: https://energyplus.net/weather-location/asia_wmo_region_2/KOR/KOR_Inchon.471120_IWEC)

```
# 다운로드한 기후 데이터 파일을 프로젝트 디렉토리로 이동
cp {설치경로}/KOR_Inchon.471120_IWEC.ddy
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/data/weather/
cp {설치경로}/KOR_Inchon.471120_IWEC.epw
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages
```

```
s/sinergym/data/weather/
```

⑤ 실행을 위한 코드 변경

본 섹션에서는 HVAC 시뮬레이션 실행을 위해 필수적인 코드 수정 및 설정 변경 사항을 정리하고자 한다. 실제 강화학습 모델이 원활하게 작동하도록 하기 위해 빌딩 모델 설정(RunPeriod 수정), 보상 함수(reward.py) 수정, 환경 래퍼(wrapper.py) 수정 등의 작업을 수행해야 한다.

- RunPeriod 수정

EnergyPlus의 기본 설정에서는 시뮬레이션을 실행할 특정 연도를 명확히 지정하지 않기 때문에, 이를 2024년을 기준으로 변경하는 과정이 필요하다. 먼저, .epJSON 파일을 편집하기 위해 nano 편집기 혹은 VSCode를 사용하여 파일을 연다. 이후, 코드 내 RunPeriod 항목을 다음과 같이 수정하여 2024년을 기준으로 설정한다.

```
# nano 에디터 실행
nano
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/data/buildings/ASHRAE901_OfficeLarge_STD2019_Denver.epJSON
# 혹은
# VSCode 실행
code
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/data/buildings/ASHRAE901_OfficeLarge_STD2019_Denver.epJSON
```

```
"RUNPERIOD 1": {
    "apply_weekend_holiday_rule": "No",
    "begin_day_of_month": 1,
    "begin_month": 1,
    "begin_year": 2024,
    "day_of_week_for_start_day": "Monday",
    "end_day_of_month": 31,
    "end_month": 12,
    "end_year": 2024,
    "use_weather_file_daylight_saving_period": "No",
    "use_weather_file_holidays_and_special_days": "No",
    "use_weather_file_rain_indicators": "Yes",
    "use_weather_file_snow_indicators": "Yes"
}
```

- reward.py 수정

EnergyPlus의 기본 설정에서는 시뮬레이션을 실행할 특정 연도를 명확히 지정하지 않기 때문에, 이를 2024년을 기준으로 변경하는 과정이 필요하다. 먼저, .epJSON 파일을 편집하기 위해 nano 편집기 혹은 VSCode를 사용하여 파일을 연다. 이후, 코드 내 RunPeriod 항목을 다음과 같이 수정하여 2024년을 기준으로 설정한다.

```
# nano 에디터 실행
nano
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/utils/rewards.py
# 혹은
# VSCode 실행
code
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/utils/rewards.py
```

```
def _get_temperature_violation(
    self, obs_dict: Dict[str, Any]) -> Tuple[float, List[float]]:
    """Calculate the total temperature violation (°C) in the current
    observation.

    Returns:
        Tuple[float, List[float]]: Total temperature violation (°C) and
        list with temperature violation in each zone.
    """

    # 기존 코드 주석처리
    # month = obs_dict['month']
    # day = obs_dict['day_of_month']
    # year = YEAR
    # current_dt = datetime(int(year), int(month), int(day))

    # 추가 start

    # 기본값 설정
    year = 2024 # 연도를 고정하거나 필요시 obs_dict에서 동적으로
    가져오도록 수정
    month = obs_dict.get('month', 1) # 기본값: 1월
    day = obs_dict.get('day_of_month', 1) # 기본값: 1일
```

```

        # 추가 end

    try:
        current_dt = datetime(int(year), int(month), int(day))
    except ValueError as e:
        print(f"Error creating datetime object: {e}")
        return 0.0, [] # 오류 발생 시 기본값 반환

    # Periods
    summer_start_date = datetime(
        int(year),
        self.summer_start[0],
        self.summer_start[1])
    summer_final_date = datetime(
        int(year),
        self.summer_final[0],
        self.summer_final[1])

    if current_dt >= summer_start_date and current_dt <=
summer_final_date:
        temp_range = self.range_comfort_summer
    else:
        temp_range = self.range_comfort_winter

    temp_values = [
        v for k, v in obs_dict.items() if k in self.temp_names]
    total_temp_violation = 0.0
    temp_violations = []
    for T in temp_values:
        if T < temp_range[0] or T > temp_range[1]:
            temp_violation = min(
                abs(temp_range[0] - T), abs(T - temp_range[1]))
            temp_violations.append(temp_violation)
            total_temp_violation += temp_violation

    return total_temp_violation, temp_violations

```

- wrapper.py 수정

강화학습 모델이 EnergyPlus와 상호작용하는 과정에서, 시뮬레이션 로그를 저장하는 경로(episode_path)가 설정되지 않은 경우 오류가 발생할 가능성이

있다 따라서, 로그 저장 경로가 None인 경우 경고 메시지를 출력하고 불필요한 오류를 방지하도록 코드 수정이 필요하다.

```
# nano 에디터 실행
nano
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/utils/wrapper.py
# 혹은
# VSCode 실행
code
/Users/{사용자명}/global-frontier-24/sinergym_env/lib/python3.12/site-packages/sinergym/utils/wrapper.py
```

```
def dump_log_files(self) -> None:
    """Dump all logger data in CSV files.
    """
    # 추가 start
    episode_path = self.get_wrapper_attr('episode_path')
    # episode_path가 None인 경우 처리
    if episode_path is None:
        self.logger.warning("Episode path is not set. Skipping log file dumping.")
        return # 아무 작업도 하지 않고 종료
    # 추가 end

    monitor_path = self.get_wrapper_attr('episode_path') + '/monitor'
    os.makedirs(monitor_path, exist_ok=True)
    episode_data = self.get_wrapper_attr('data_logger')

    if len(episode_data.rewards) > 0:

        # Observations
        with open(monitor_path + '/observations.csv', 'w') as f:
            writer = csv.writer(f)

        writer.writerow(self.get_wrapper_attr('observation_variables'))
        writer.writerows(episode_data.observations)

        # Normalized Observations
        if len(episode_data.normalized_observations) > 0:
```

```

        with open(monitor_path + '/normalized_observations.csv', 'w')
as f:
            writer = csv.writer(f)
            writer.writerow(
                self.get_wrapper_attr('observation_variables'))
            writer.writerows(episode_data.normalized_observations)

# Rewards
with open(monitor_path + '/rewards.csv', 'w') as f:
    writer = csv.writer(f)
    writer.writerow(['reward'])
    for reward in episode_data.rewards:
        writer.writerow([reward])

# Infos (except excluded keys)
with open(monitor_path + '/infos.csv', 'w') as f:
    writer = csv.writer(f)
    column_names = [key for key in episode_data.infos[-1].keys(
        ) if key not in self.get_wrapper_attr('info_excluded_keys')]
    # Skip reset row
    rows = [[value for key, value in info.items() if key not in
self.get_wrapper_attr(
        'info_excluded_keys')] for info in
episode_data.infos[1:]]
    writer.writerow(column_names)
    # write null row for reset
    writer.writerow([None for _ in range(len(column_names))])
    writer.writerows(rows)

# Agent Actions
with open(monitor_path + '/agent_actions.csv', 'w') as f:
    writer = csv.writer(f)
    writer.writerow(self.get_wrapper_attr('action_variables'))
    if isinstance(episode_data.actions[0], list):
        writer.writerows(episode_data.actions)
    else:
        for action in episode_data.actions:
            writer.writerow([action])

# Simulated actions

```

```

with open(monitor_path + '/simulated_actions.csv', 'w') as f:
    writer = csv.writer(f)
    writer.writerow(self.get_wrapper_attr('action_variables'))
    # reset_action = [None for _ in range(
    #     len(self.get_wrapper_attr('action_variables')))]
    simulated_actions = [list(info['action'])
                          for info in episode_data.infos[1:]]
    if isinstance(simulated_actions[0], list):
        writer.writerows(simulated_actions)
    else:
        for action in simulated_actions:
            writer.writerow([action])

# Custom metrics
if len(episode_data.custom_metrics) > 0:
    with open(monitor_path + '/custom_metrics.csv', 'w') as f:
        writer = csv.writer(f)

writer.writerow(self.get_wrapper_attr('custom_variables'))
writer.writerows(episode_data.custom_metrics)

# Update progress.csv
episode_summary = self.get_wrapper_attr('get_episode_summary')()

with open(self.get_wrapper_attr('progress_file_path'), 'a+') as
f:
    writer = csv.writer(f)
    # If first episode, write header
    if self.get_wrapper_attr('episode') == 1:
        writer.writerow(list(episode_summary.keys()))
        writer.writerow(list(episode_summary.values()))

```

⑥ 강화학습 환경 변수 및 액추에이터 설정

본 프로젝트에서는 강화학습 모델이 올바르게 학습할 수 있도록 EnergyPlus 내에서 사용할 변수(variables, 즉 observations)와 액추에이터(actuators)를 직접 설정하는 과정이 필요하다.

EnergyPlus는 여러 개의 시뮬레이션 변수를 제공하지만, 모든 변수를 사용할 필요는 없으며, HVAC 최적화에 필요한 핵심 변수만을 선별하여 적용해야 한다. 이를 위해, EnergyPlus를 실행하여 생성된 결과 파일(.rdd, .mdd,

data_available.txt)을 분석하여 필요한 변수를 직접 찾아내는 과정을 진행했다.

- Observations

외부 환경(Outdoor Temperature), 실내 온도(Zone Temperature), 전력 소비량(HVAC Electricity Demand) 등을 핵심 변수로 포함하여 최적의 제어가 가능하도록 하였다.

```
variables = {
  "Outdoor Air Temperature": ("Site Outdoor Air Drybulb Temperature",
    "ENVIRONMENT"),
  "Zone Mean Air Temperature": ("Zone Mean Air Temperature",
    "CORE_BOTTOM"),
  "Facility Total HVAC Electricity Demand Rate": ("Facility Total HVAC
    Electricity Demand Rate", "WHOLE BUILDING")
}
```

변수명	EnergyPlus Key	카테고리	설명
Outdoor Air Temperature	Site Outdoor Air Drybulb Temperature	ENVIRONMENT	외부 공기 온도 (외부 환경 반영)
Zone Mean Air Temperature	Zone Mean Air Temperature	CORE_BOTTOM	실내 평균 공기 온도 (목표 온도 유지)
Facility Total HVAC Electricity Demand Rate	Facility Total HVAC Electricity Demand Rate	WHOLE BUILDING	HVAC 시스템 전체 전력 소비량 (에너지 최적화)

- Actuators

난방(Heating Setpoint)과 냉방(Cooling Setpoint)의 목표 온도를 조정할 수 있도록 설정하여, 최적의 온도 조절을 가능하게 하였다.

```
actuators = {
  "Heating Setpoint": ("Schedule:Compact", "Schedule Value",
    "HTGSETP_SCH_NO_OPTIMUM"),
  "Cooling Setpoint": ("Schedule:Compact", "Schedule Value",
    "CLGSETP_SCH_NO_SETBACK")
}
```

액추에이터명	EnergyPlus Key	카테고리	설명
Heating	Schedule:Compa	Schedule Value	난방 온도 설정

Setpoint	ct		값 (난방 에너지 절감)
Cooling Setpoint	Schedule:Compa ct	Schedule Value	냉방 온도 설정 값 (냉방 에너지 절감)

(2) 구축한 환경 내 알고리즘 적용

① DDPG 알고리즘 적용

DDPG(Deep Deterministic Policy Gradient)는 기존의 강화학습 기법인 Deep Q-Learning을 연속적인 행동을 다룰 수 있도록 확장한 것이다. 일반적인 Q-Learning 방식은 정해진 개수의 행동 중에서 하나를 선택하는 구조이지만, DDPG는 실수 값을 가지는 연속적인 행동을 직접 결정할 수 있도록 한다. 예를 들어, 로봇이 움직일 때 단순히 "왼쪽으로 이동"과 같은 정해진 명령이 아니라 "0.75만큼 왼쪽으로 이동"처럼 세밀한 조절이 가능해진다. 또한, DDPG는 정책을 직접 학습하는 방식을 사용하며, Actor-Critic이라는 두 개의 신경망을 활용하는 구조로 이루어져 있다. Actor 신경망은 현재 상태에서 어떤 행동을 해야 할지 결정하는 역할을 하며, Critic 신경망은 Actor가 선택한 행동이 얼마나 좋은지 평가하는 역할을 한다. Critic의 피드백을 바탕으로 Actor는 더 나은 행동을 학습하게 된다. 이를 통해 DDPG는 연속적인 값으로 이루어진 행동 공간에서도 효과적으로 학습할 수 있으며, 로봇 제어나 자율 주행과 같은 다양한 실제 환경에서 활용될 수 있다.

DDPG를 HVAC 제어에 적용하면 연속적인 액션을 직접 다룰 수 있어 온도, 습도, 공기 흐름 등을 세밀하게 조정할 수 있다. 기존의 모델 예측 제어 방식은 높은 계산 비용과 실시간 적용의 어려움이 있지만, DDPG는 학습된 정책을 활용해 빠르게 최적의 제어를 수행하며, 환경 변화에도 적응할 수 있다. 또한, 보상 함수에 에너지 효율과 사용자 만족도를 반영하여 균형 잡힌 최적화를 수행하고, 센서 데이터를 활용한 IoT 기반 스마트 HVAC 시스템과 통합이 용이하다. 이를 통해 불필요한 에너지 소비를 줄이면서도 쾌적한 실내 환경을 유지할 수 있어, HVAC 제어의 효율성과 사용자 만족도를 동시에 향상시키는 효과적인 방법이 될 수 있다.

DDPG 구현을 위해 사용되는 라이브러리는 'gymnasium', 'numpy', 'stable baseline3', 'sinergym'이다.

```
import numpy as np
import gymnasium as gym
from stable_baselines3 import DDPG
```

```
from stable_baselines3.common.noise import NormalActionNoise
from sinergym.envs import EplusEnv
from sinergym.utils.rewards import LinearReward
```

- 모델 구축

stable baseline3에서 제공하는 DDPG 모델을 활용하여 모델을 구축한다. 모델 설정을 위한 매개변수로 MlpPolicy를 사용하여 다중 퍼셉트론 정책 네트워크를 사용하도록 설정하고, 환경을 앞서 만들었던 맞춤형 환경을 매개변수로 전달한다.

```
model = DDPG("MlpPolicy", env, verbose=0)
```

- 모델 학습 및 평가

DDPG(Deep Deterministic Policy Gradient) 강화학습 에이전트를 학습시키고, 학습된 모델을 사용하여 환경에서 성능을 평가한 후, 최종 보상을 계산하는 과정을 실행한다. learn 메서드를 통해 모델을 학습시키며, 이때 total_timesteps 설정을 통해 몇 번의 타임스텝 동안 학습을 진행할지 설정한다.

이후 평가를 위해 평가 환경, 보상, 완료 여부를 초기화한다. 그다음, 완료 여부가 True가 될 때까지 반복문을 실행하며 학습된 모델의 성능을 평가한다. 반복문에서는 현재 관측값을 매개변수로 하는 predict 메서드를 사용하여 에이전트의 다음 행동을 예측하고, 이 행동을 다음 스텝으로 설정하여 관측값, 보상, 종료 여부 등을 변수에 저장하는 과정을 수행한다.

반복문이 종료되면 최종 보상을 합산하여 반환한다.

```
model.learn(total_timesteps=5000, log_interval=100)

# 평가
obs, info = eval_env.reset()
rewards = []
done = False
while not done:
    action, _ = model.predict(obs, deterministic=True)
    obs, reward, terminated, truncated, info = eval_env.step(action)
    rewards.append(reward)
    done = terminated or truncated
```

```
total_reward = sum(rewards)

# 모델과 환경 종료
model.env.close()
return total_reward
```

② Optuna를 활용한 하이퍼 파라미터 튜닝

Optuna는 하이퍼파라미터 최적화를 위한 자동화된 프레임워크로, 사용자가 지정한 목표 함수를 최적화하기 위해 효율적인 탐색을 수행하는 라이브러리이다. 기존의 그리드 검색이나 랜덤 검색보다 더 효율적으로 최적의 하이퍼파라미터를 찾을 수 있도록 설계되었으며, 특히 베이지안 최적화 기반의 TPE 알고리즘을 활용해 탐색 공간을 동적으로 조정한다.

주어진 문제에서 최적의 성능을 내는 하이퍼파라미터를 자동으로 탐색하여 실험 시간을 단축하고, 사용자의 수작업 부담을 줄이며, 보다 정밀한 모델 성능을 확보할 수 있기 때문에 Optuna를 통해 하이퍼파라미터 튜닝을 진행하고자 했다. 이 과정을 통해 머신러닝 및 강화학습 모델의 성능을 향상시키고, 복잡한 최적화 문제를 효과적으로 해결하는 것을 목표로 한다.

• optuna 초기 설정

Optuna를 사용하기 위해 하이퍼파라미터 최적화를 수행하는 objective 함수를 설정해준다. 이때, 최적화가 필요한 하이퍼파라미터를 샘플링하여 그 범위를 지정한다. 학습에 가장 많은 영향을 미칠 수 있는 파라미터인 'learning_rate', 'batch_size', 'action_noise_sigma'를 변수로 선언한 뒤, 각각의 범위를 지정해준다. batch_size는 학습 속도를 고려하여 128로 고정값을 지정하였다. 이후, 앞서 설명한 모델을 설정하고 학습 및 평가를 수행하는 코드를 작성한다.

```
import optuna

def objective(trial):
    # 하이퍼파라미터 샘플링
    learning_rate = trial.suggest_loguniform('learning_rate', 1e-5, 1e-3)
    batch_size = trial.suggest_categorical('batch_size', [128])
    action_noise_sigma = trial.suggest_uniform('action_noise_sigma', 0.01,
0.2)
```

```

# 액션 노이즈 설정
n_actions = env.action_space.shape[-1]
action_noise = NormalActionNoise(mean=np.zeros(n_actions),
sigma=action_noise_sigma * np.ones(n_actions))

# 모델 설정
model = DDPG(
    "MlpPolicy",
    env,
    action_noise=action_noise,
    verbose=0, # 로그 출력 최소화
    batch_size=batch_size,
    learning_rate=learning_rate,
    tensorboard_log=None
)

# 학습
model.learn(total_timesteps=5000, log_interval=100)

# 평가
obs, info = eval_env.reset()
rewards = []
done = False
while not done:
    action, _ = model.predict(obs, deterministic=True)
    obs, reward, terminated, truncated, info = eval_env.step(action)
    rewards.append(reward)
    done = terminated or truncated

total_reward = sum(rewards)

# 모델과 환경 종료
model.env.close()
return total_reward

```

- 하이퍼 파라미터 튜닝

기존 샘플 데이터를 기반으로 다음 샘플링할 파라미터를 결정하는 TPE 샘플러를 사용하여 하이퍼파라미터 탐색 기능을 수행한다. seed=42를 설정하여 난

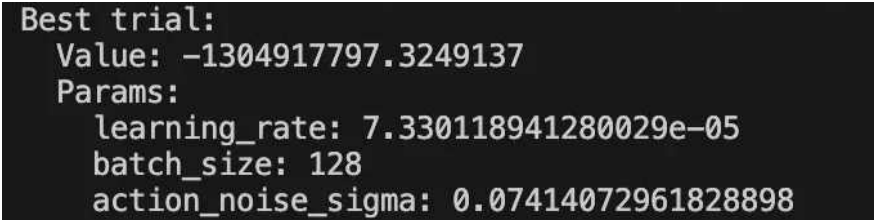
수 시드를 고정해준다. 또한, `optuna.create_study(direction='maximize', sampler=sampler)`를 사용하여 새로운 Optuna 스터디를 생성하며, 여기서 `direction='maximize'`는 보상을 최대화하는 방향으로 최적화를 수행하도록 설정한다.

이후, 앞서 정의한 objective 함수와 실행 횟수를 매개변수로 하는 optimize 메서드를 활용하여 최적화 과정을 실행하고, 찾은 최적의 하이퍼파라미터를 출력한다.

이때, 성능을 더 높이기 위해 seed 값을 다양하게 시도한 뒤 여러 최적화된 하이퍼파라미터를 구하고, 앞서 설정한 objective 함수의 범위를 해당 하이퍼파라미터들의 리스트로 설정하여 가장 최적화된 하이퍼파라미터를 계산해준다.

```
# Optuna 스터디 생성
sampler = optuna.samplers.TPESampler(seed=42)
study = optuna.create_study(direction='maximize', sampler=sampler) #
리워드를 최대화
study.optimize(objective, n_trials=20) # 20번의 실험 실행

# 최적의 하이퍼파라미터 출력
print("Best trial:")
trial = study.best_trial
print(f" Value: {trial.value}")
print(" Params: ")
for key, value in trial.params.items():
    print(f"    {key}: {value}")
```



```
Best trial:
Value: -1304917797.3249137
Params:
  learning_rate: 7.330118941280029e-05
  batch_size: 128
  action_noise_sigma: 0.07414072961828898
```

[그림 10] Optuna로 구한 최적의 하이퍼파라미터

③ 최적의 파라미터를 이용한 DDPG 모델 학습

앞선 과정을 실행해 구한 하이퍼 파라미터로 DDPG 모델을 구축하고 총 15 번의 에피소드를 실행하는 코드를 작성한다.

- 모델 구축

```
# 최적 하이퍼파라미터
best_params = {
    "learning_rate": 5.6115164153345e-05,
    "batch_size": 128,
    "action_noise_sigma": 0.19063571821788408
}

n_actions = env.action_space.shape[-1]
best_action_noise = NormalActionNoise(
    mean=np.zeros(n_actions),
    sigma=best_params['action_noise_sigma'] * np.ones(n_actions)
)

#
model = DDPG(
    "MlpPolicy",
    env,
    action_noise=best_action_noise,
    verbose=0, # 로그 출력 최소화
    batch_size=best_params['batch_size'],
    learning_rate=best_params['learning_rate'],
    tensorboard_log=None
)
```

- 15번의 에피소드 실행

15의 에피소드를 실행하면서 에피소드별 리워드를 계산해 총 리워드와 평균 HVAC 전력 요구량을 함께 계산해준다.

```
num_episodes = 15
all_rewards = []

logger.info("Starting episode runs...")
for episode in range(1, num_episodes + 1):
    cum_hvac = 0
    i = 0
    obs, info = env.reset()
    episode_rewards = []
    done = False
```

```

while not done:
    action, _ = model.predict(obs, deterministic=True)
    obs, reward, terminated, truncated, info = env.step(action)
    episode_rewards.append(reward)
    done = terminated or truncated
    obs_dict = dict(zip(env.get_wrapper_attr('variables'), obs))
    print(obs_dict)
    i+=1
    cum_hvac += obs_dict['Facility Total HVAC Electricity Demand Rate']

mean_cum_hvac = cum_hvac / i
total_reward = sum(episode_rewards)
all_rewards.append(total_reward)
logger.info(f"mean_cum_hvac: {mean_cum_hvac}")
logger.info(f"Episode {episode}/{num_episodes} - Total Reward:
{total_reward}")

# 결과 출력
average_reward = np.mean(all_rewards)
logger.info("Episode runs completed.")
logger.info(f"Average Total Reward over {num_episodes} episodes:
{average_reward}")

```

④ Fixed Control 알고리즘 적용

Fixed Control 알고리즘은 사전에 정의된 규칙과 임계값을 기반으로 시스템을 제어하는 방식이다. 이 알고리즘은 특정한 환경 조건이나 입력 값에 대해 미리 설정된 제어 로직을 따르며, 일반적으로 간단한 온도 조절, 일정한 주기로 작동하는 시스템, 혹은 사전 설정된 스케줄을 따르는 장치 등에 적용된다.

Fixed Control은 별도로 모델이 구현된 것이 아니기 때문에, sinergym에서 제공하는 문서를 참고하여 코드를 작성하였으며, typing 모듈을 사용하여 변수의 타입을 명시해 오류를 예방하였다.

- 환경 구축

앞서 구축했던 맞춤형 환경을 CustomZone 객체로 설정하고, 현재 날짜에 따라 계절 범위를 판단하는 act 함수를 정의하였다.

```

class CustomZone(object):
    building_file = "ASHRAE901_OfficeLarge_STD2019_Denver.epJSON"

```

```

weather_file = "KOR_Inchon.471120_IWEC.epw"
experiment_date = datetime.today().strftime('%Y-%m-%d_%H:%M')
experiment_name = f'DDPG-LargeOffice-{experiment_date}'

env = env

observation_variables = env.get_wrapper_attr(
    'variables')

action_variables = env.get_wrapper_attr(
    'actuators')

setpoints_summer = np.array((23.0, 26.0), dtype=np.float32)
setpoints_winter = np.array((20.0, 23.5), dtype=np.float32)

def act(self, observation: List[Any]) -> Sequence[Any]:
    """Select action based on indoor temperature.

    Args:
        observation (List[Any]): Perceived observation.

    Returns:
        Sequence[Any]: Action chosen.
    """
    obs_dict = dict(zip(self.observation_variables, observation))
    year = 2024
    month = obs_dict.get('month', 1) # 기본값: 1월
    day = obs_dict.get('day_of_month', 1) # 기본값: 1일

    summer_start_date = datetime(year, 6, 1)
    summer_final_date = datetime(year, 9, 30)

    current_dt = datetime(year, month, day)

    # Get season comfort range
    if current_dt >= summer_start_date and current_dt <=
summer_final_date: # pragma: no cover
        season_range = self.setpoints_summer
    else: # pragma: no cover
        season_range = self.setpoints_winter

```

```
return season_range
```

- 컨트롤러 구현

앞서 정의했던 커스텀 환경을 상속받아 규칙 기반으로 HVAC을 제어할 수 있는 코드를 작성하였다. 이 객체는 주어진 환경의 상태인 온도와 날짜 등을 기반으로 난방과 냉방 설정값을 조정하는 역할을 한다.

컨트롤러는 현재 환경에서 관측된 변수를 observation으로 받아 이를 obs_dict에 저장하여 실내 및 실외 온도를 확인한다. 또한, act_dict를 통해 현재 난방 및 냉방 설정값을 가져온다. 이후, 현재 날짜를 기준으로 여름철과 겨울철로 구분하여 각 계절에 맞는 쾌적 온도 범위를 설정하고, 실내 온도와 비교하여 난방 및 냉방 설정값을 조정한다.

만약 실내 온도가 설정된 범위보다 낮다면 난방과 냉방 온도를 높이고, 반대로 높다면 온도를 낮추는 방식으로 동작한다. 설정값이 환경에서 허용된 최소 및 최대 범위를 벗어나지 않도록 클리핑 과정을 거쳐 조정된 값을 유지한다. 또한, 주말의 경우 특정 온도 범위를 적용하여 에너지 절약을 고려한 설정을 수행하도록 한다.

최종적으로 조정된 난방 및 냉방 설정값을 환경에 적용할 액션으로 반환하여 HVAC 시스템이 실내 온도를 효율적으로 유지할 수 있도록 한다.

```
class MyRuleBasedController(CustomZone):
    def act(self, observation: List[Any]) -> Sequence[Any]:
        """Select action based on outdoor air drybulb temperature and
        daytime.

        Args:
            observation (List[Any]): Perceived observation.

        Returns:
            Sequence[Any]: Action chosen.
        """

        obs_dict = dict(zip(self.env.get_wrapper_attr(
            'variables'), observation))

        act_dict = dict(zip(self.env.get_wrapper_attr(
            'actuators'), observation))
```

```

print(obs_dict)

out_temp = obs_dict['Outdoor Air Temperature']
year = 2024 # 연도를 고정하거나 필요시 obs_dict에서 동적으로
가져오도록 수정
month = obs_dict.get('month', 1) # 기본값: 1월
day = obs_dict.get('day_of_month', 1) # 기본값: 1일

summer_start_date = datetime(year, 6, 1)
summer_final_date = datetime(year, 9, 30)

current_dt = datetime(year, month, day)

# Get season comfort range
if current_dt >= summer_start_date and current_dt <=
summer_final_date:
    season_comfort_range = self.setpoints_summer
else:
    season_comfort_range = self.setpoints_summer
season_comfort_range = self.setpoints_winter
# Update setpoints
in_temp = obs_dict['Zone Mean Air Temperature']

current_heat_setpoint = act_dict[
    'Heating Setpoint']
current_cool_setpoint = act_dict[
    'Cooling Setpoint']

new_heat_setpoint = current_heat_setpoint
new_cool_setpoint = current_cool_setpoint

if in_temp < season_comfort_range[0]:
    new_heat_setpoint = current_heat_setpoint + 1
    new_cool_setpoint = current_cool_setpoint + 1
elif in_temp > season_comfort_range[1]:
    new_cool_setpoint = current_cool_setpoint - 1
    new_heat_setpoint = current_heat_setpoint - 1

# Clip setpoints to the action space

```

```

        if new_heat_setpoint >
self.env.get_wrapper_attr('action_space').high[0]:
            new_heat_setpoint = self.env.get_wrapper_attr(
                'action_space').high[0]
        if new_heat_setpoint <
self.env.get_wrapper_attr('action_space').low[0]:
            new_heat_setpoint = self.env.get_wrapper_attr(
                'action_space').low[0]
        if new_cool_setpoint >
self.env.get_wrapper_attr('action_space').high[1]:
            new_cool_setpoint = self.env.get_wrapper_attr(
                'action_space').high[1]
        if new_cool_setpoint <
self.env.get_wrapper_attr('action_space').low[1]:
            new_cool_setpoint = self.env.get_wrapper_attr(
                'action_space').low[1]

        action = (new_heat_setpoint, new_cool_setpoint)
        if current_dt.weekday() > 5:
            action = (
                2 * (np.array([18.33, 23.33]) -
self.env.get_wrapper_attr('action_space').low) /
                (self.env.get_wrapper_attr('action_space').high -
self.env.get_wrapper_attr('action_space').low) - 1
            )

        return action

```

- 15번의 에피소드 실행

앞서 정의했던 컨트롤러의 객체를 생성하고, 에피소드를 15번으로 설정하여 반복 실행한다. 반복문 내부에서는 에피소드별 총 에너지 소비량을 계산하고, 평균 HVAC 전력 요구량을 산출한다.

```

agent = MyRuleBasedController()

# 에피소드 반복 설정
num_episodes = 15 # 에피소드 횟수 설정

ep_cum_hvac = 0

```

```

for i in range(num_episodes):
    obs, info = env.reset()
    rewards = []
    truncated = terminated = False
    current_month = 0
    cum_hvac = 0
    i = 0
    print(f"Starting Episode {i + 1}...")
    while not (terminated or truncated):
        action = agent.act(obs)
        obs, reward, terminated, truncated, info = env.step(action)
        rewards.append(reward)
        if info['month'] != current_month: # 매달 결과 출력
            current_month = info['month']
            print(f"Monthly Reward: {sum(rewards):.2f} | Info: {info}")
        obs_dict = dict(zip(env.get_wrapper_attr('variables'), obs))
        i+=1
        cum_hvac += obs_dict['Facility Total HVAC Electricity Demand Rate']

    mean_cum_hvac = cum_hvac / i
    print(f"mean_cum_hvac: {mean_cum_hvac}")

    ep_cum_hvac += cum_hvac

    print(
        f"Episode {i + 1} Completed: Mean Reward = {np.mean(rewards):.2f},
        Cumulative Reward = {sum(rewards):.2f}"
    )

print(
    'Episode ',
    i,
    'Mean reward: ',
    np.mean(rewards),
    'Cumulative reward: ',
    sum(rewards))

```

(3) 각 알고리즘별 성능 평가

① wandb 초기 세팅

WandB(Weights & Biases)는 머신러닝 및 강화학습 실험을 체계적으로 관리하고 시각화하는 MLOps 툴이다. WandB의 가장 큰 장점은 실험 로깅, 시각화, 하이퍼파라미터 튜닝, 협업 기능을 모두 제공한다는 점이다. 특히, 강화학습 환경에서 WandB는 에피소드별 보상, 에너지 소비량, 학습 성능 변화 등을 시각화하는 데 유용하다. 예를 들어, HVAC 시스템 최적화와 같은 문제에서 각 에피소드의 평균 전력 소비량을 기록하고 학습 진행에 따라 변화하는 양상을 분석할 수 있다.

- 터미널에 다음 명령어를 입력해 wandb를 설치하고, 연결한다.

```
# wandb 설치
pip install wandb

# wandb 연결
wandb login
```

- wandb: Paste an API key from your profile and hit enter, or press ctrl+c to quit: 메시지가 나오면 Wandb 사이트를 통해 API key를 확인해 입력한다.
- wandb를 초기화하고, 팀과 프로젝트를 선택한다.

```
# wandb 초기화
wandb init
```

- 기존 코드에 wandb에 로깅하고, 시각화하는 코드를 추가한다.

```
import wandb

wandb.init(project="global-frontier")
```

② 성능 평가

다음 2가지 값을 통해 성능 평가를 하고자 한다.

1. DDPG에서의 Energy Penalty와 Comport Penalty 값 확인
2. Fixed-Control과 DDPG 코드를 통해 Facility Total HVAC Electricity Demand Rate 평균값 비교

- DDPG에서의 Energy Penalty와 Comport Penalty 값 확인

Energy Penalty와 Comport Penalty 값을 별도로 설정하지 않아도, Sinergym의 내부 로깅 시스템과 WandB 통합 기능 덕분에 해당 값들이 자동으로 WandB 대시보드에 기록되고, 그래프로 시각화된다.

우선, WandB에 로깅을 하기 위해 Sinergym의 WandBLogger 래퍼를 추가한다. entity 에는 WandB 사용자 ID, save_code 는 코드 저장 여부, dump_frequency 에는 몇 스텝마다 저장을 할 것인지를 설정한다.

```
env = WandBLogger(env,
                    entity='global-frontier-bringgreen',
                    project_name='eplus-ddpg',
                    run_name=experiment_name,
                    group='Train_example',
                    tags=['DRL', 'PPO', '5zone', 'continuous', 'stochastic',
                        'v1'],
                    save_code = True,
                    dump_frequency = 1000,
                    artifact_save = False)
```

다음으로, WandB의 커스텀 콜백을 설정한다. sinergym-version 에서 Sinergym의 버전을 기록하고, python-version 에서 python 버전을 기록한다. 이후 해당 파라미터들을 WandB의 설정으로 업데이트 해준다.

```
if is_wrapped(env, WandBLogger):
    experiment_params = {
        'sinergym-version': sinergym.__version__,
        'python-version': sys.version
    }
    env.get_wrapper_attr('wandb_run').config.update(experiment_params)

    callbacks = []
```

SB3(Stable-Baselines3)에서 WandB를 활용하기 위해 WandBOutputFormat()을 로깅 포맷에 추가한다. 이를 통해 학습 중간중간 step별 결과가 자동으로 WandB 대시보드에 업로드되며, 시각적으로 학습이 어떻게 진행되고 있는지를 확인할 수 있다.

```

if is_wrapped(env, WandBLogger):
    logger = SB3Logger(
        folder=None,
        output_formats=[
            HumanOutputFormat(
                sys.stdout,
                max_length=120),
            WandBOutputFormat())
    model.set_logger(logger)

```

학습이 끝나면 `model.save()`를 통해 학습된 모델을 저장하고, 환경을 닫아 종료한다.

```

model.save(env.get_wrapper_attr('workspace_path') + '/model')

env.close()

```

- Fixed-Control과 DDPG 코드를 통해 Facility Total HVAC Electricity Demand Rate 평균값 비교

[Fixed-Control]

우선, WandB를 초기화한다.

```

wandb.init(
    project="global-frontier",
    entity="seoah-ewha-womans-university",
    name="Fixed-Control-HVAC",
    group="Fixed_Control",
    tags=["Fixed", "Control", "HVAC", "5zone"],
)

```

다음으로 WandB에 스텝별 평균 HVAC 소비량을 기록한다. 이는 각 스텝이 종료될 때 WandB에 데이터를 기록하는 코드이다.

```

wandb.log({
    "Step": step_count,
    "HVAC Demand": hvac_demand
}, step=step_count)

```

마지막으로 하나의 에피소드가 끝난 후 해당 에피소드에서의 평균 HVAC 전력 소비량을 기록한다. 이후 `wandb.finish()` 를 통해 세션을 종료시킨다.

```
wandb.log({
    "Step": step_count,
    "HVAC Demand": hvac_demand
}, step=step_count)
```

[DDPG]

모델이 학습을 진행하면서 한 스텝씩 환경을 업데이트할 때마다, 현재 시점의 Facility Total HVAC Electricity Demand Rate 값을 저장한다. 각 스텝에서 관찰된 HVAC 전력 소비량을 리스트(`self.episode_hvac_demand`)에 추가하는 방식으로 이루어진다.

```
def _on_step(self) -> bool:
    logger = self.training_env.get_attr('logger')[0] # LoggerWrapper 객체 가져오기

    if hasattr(logger, "get_log_data"):
        log_data = logger.get_log_data()

        # HVAC 전력 소비량 저장 (step 단위 & 에피소드 단위)
        if "Facility Total HVAC Electricity Demand Rate" in log_data:
            hvac_value = log_data["Facility Total HVAC Electricity Demand Rate"]

            self.episode_hvac_demand.append(hvac_value) # 에피소드별 리스트에 저장

    return True # 학습 계속 진행
```

에피소드가 끝나면 (`_on_rollout_end` 호출), 해당 에피소드 동안 기록된 모든 step의 HVAC 전력 소비량의 평균값을 계산하여 WandB에 기록한다. `np.mean(self.episode_hvac_demand)` 를 사용해 에피소드 동안의 HVAC 소비량 평균값을 계산하고, `wandb.log()`를 사용하여 WandB에 데이터를 기록한

다.

```
def _on_rollout_end(self) -> None:
    if len(self.episode_hvac_demand) > 0:
        avg_hvac_demand = np.mean(self.episode_hvac_demand) # 에피소드별
평균 계산

        print(f"[Episode {self.episode_count}] Avg HVAC Demand:
{avg_hvac_demand}")

        wandb.log({
            "Episode": self.episode_count, # 명확하게 Episode 값을
WandB에 로깅
            "Avg Facility HVAC Demand": avg_hvac_demand
        }, step=self.episode_count) # Step을 에피소드 번호로 설정

        # 데이터 초기화 (다음 에피소드 데이터 저장)
        self.episode_hvac_demand = []
        self.episode_count += 1 # 에피소드 증가
```

정의된 WandBLoggingCallback을 실제 학습 과정에서 사용할 수 있도록 callbacks 리스트에 추가한다.

```
hvac_logging_callback = WandBLoggingCallback()
callbacks.append(hvac_logging_callback)
```

SB3(Stable-Baselines3)에서 학습을 수행할 때, 콜백을 함께 전달하여 학습 중 WandB 로깅이 실행되도록 한다.

```
model.learn(
    total_timesteps=timesteps,
    callback=callback, # 여기서 WandB 로깅 콜백 포함
    log_interval=100
)
```