# Towards Off-policy Evaluation as a Prerequisite for Real-world Reinforcement Learning in Building Control

Bingqing Chen
Carnegie Mellon University
Pittsburgh, PA, USA
bingqinc@andrew.cmu.edu

Ming Jin
Virginia Tech
Blacksburg, VA, USA
jinming@vt.edu

Zhe Wang
Tianzhen Hong
zwang5@lbl.gov
thong@lbl.gov
Lawrence Berkeley National Laboratory
Berkeley, CA, USA

Mario Berges
Carnegie Mellon University
Pittsburgh, PA, USA
marioberges@cmu.edu

## ABSTRACT

We present an initial study of off-policy evaluation (OPE), a problem prerequisite to real-world reinforcement learning (RL), in the context of building control. OPE is the problem of estimating a policy's performance without running it on the actual system, using historical data generated by the existing controller. It enables the control engineers to ensure a new, pretrained policy satisfies the minimal performance requirements and safety constraints of a real-world system, prior to interacting with it. While many methods have been developed for OPE, no study has evaluated which ones are suitable for natural building operational data, which are generated by deterministic policies and have limited coverage of the state-action space. After reviewing existing works and their assumptions, we adopted the approximate model (AM) method. Furthermore, we used bootstrapping to quantify uncertainty and correct for bias. In a simulation study, we evaluated the proposed approach on 10 policies pretrained with imitation learning. On average, the AM method estimated the energy and comfort costs with 1.84% and 14.1% error, respectively.

## CCS CONCEPTS

• **Computing methodologies** → **Reinforcement learning**; *Learning from demonstrations.*

## KEYWORDS

Building Control; Off-policy Evaluation; Real-world Reinforcement Learning;


*RLEM '20, Nov. 17, 2020, Yokohama, Japan*

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

**ACM Reference Format:**
Bingqing Chen, Ming Jin, Zhe Wang, Tianzhen Hong, and Mario Berges. 2020. Towards Off-policy Evaluation as a Prerequisite for Real-world Reinforcement Learning in Building Control. In *RELM '20: ACM 1st Int'l Workshop on Reinforcement Learning for Energy Management in Buildings and Cities, Nov. 17, 2020, Yokohama, Japan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

## 1 INTRODUCTION

Imitation learning is a promising approach to warm-start a policy with historical data from existing building controllers [1, 6]. In fact, it was demonstrated in [1] that a policy pretrained on historical data could match the performance of the existing controller, prior to any interaction with the environment. Such approach enables real-world deployment of reinforcement learning (RL) agents, with minimal disruption to normal building operations. However, minimizing the imitation loss does not directly translate to improved control performance. Furthermore, before deploying a RL agent in a real building, one should be able to address confidently concerns from building stakeholders, such as how well the comfort would be maintained, or whether equipment damage could occur.

This motivates us to study the problem of off-policy evaluation (OPE), i.e. estimating a policy's performance without running it on the actual system, an open challenge for real-world RL [4]. In addition to enabling a control engineer to evaluate if a policy satisfy minimal performance requirements and safety constraints, OPE allows one to select the best-performing policy, under different combinations of network architectures and hyperparameters. Finally, OPE is closely related to the topic of safe RL [5]. For instance, OPE is used as a subroutine for safe policy improvement in [7].

In this paper, we examine the potential applications and challenges for OPE in the context of building control. While many methods have been developed for OPE [13], no study has evaluated which of these methods are appropriate given the characteristics of building operational data. The majority of buildings are operated by highly-deterministic polices, i.e. rule-based ones, such as hysteresis and proportional–integral–derivative control. As a result of that, building operational data typically cover a limited portion of the state-action space. Such characteristics violate the minimal requirement of some of the OPE methods, as elaborated in Section

2. Given that some state-action pairs may never be observed in the historical data, we explored the use of the approximate model (AM) method, which could extrapolate to unseen state-action space. We also used the bootstrap method to quantify uncertainty and correct for bias (Section 3). We evaluated this approach on 10 different policies, and found that it estimated the the energy and comfort cost with 1.8% and 14.1% error on average (Section 4).

## 2 RELATED WORK

OPE, proposed in [9], is the problem of evaluating a policy's performance without running it on the actual system, using historical data generated by the existing control. Using the terminology in the OPE literature, we also call the existing building control the *behaviour policy*, $\pi_b$, and the policy to be evaluated the *target policy*, $\pi_e$. OPE methods can be classified as importance sampling (IS) methods, direct methods (DM), and hybrid methods [13]. IS methods use IS to account for the distribution mismatch between the target and behaviour policies [9]. The minimal requirement of the IS methods [9] is that the behaviour policy has a non-zero probability over the state-action space. This requirement is easy to understand: the probability of the behaviour policy is on the denominator of the importance weights, and thus has to be non-zero. This presents an obstacle for applying IS methods to building control, i.e. some state-action pairs may never be observed in operational data. DM directly estimate the value function of the target policy using regression, with or without a model of the environment [13]. AM is a model-based DM, which learns a model of the environment on historical data, and uses it as a proxy for the actual environment. Alternatively, one may fit the value function directly with data. While IS estimators are unbiased, they suffer from large variance. On the other hand, model-based methods have small variance, but produce biased estimates. Hybrid methods, such as doubly robust estimator [7] and MAGIC [12], combine the strength of both IS methods and model-based methods to trade off bias and variance. Note that hybrid estimators contain importance weights, and thus the non-zero probability requirement of IS methods carries over, albeit being phrased differently (e.g. stochastic behaviour policy in [7] or bounded importance weights in [12]). For a comprehensive review and comparison, we refer interested readers to [13].

## 3 APPROACH

We formulate OPE in the context of building control in Section 3.1, and introduce the AM method in Section 3.2. We also use bootstrapping to quantify uncertainty and correct for bias (Section 3.3).

### 3.1 Problem Formulation

A RL problem is commonly formulated as a Markov Decision Process (MDP). At each time step $t$, the agent selects an action $u_t$ based on its policy $\pi$ given the current state $x_t$, i.e. $u_t \sim \pi(\cdot|x_t)$. When the agent takes the action $u_t$, the state changes based on the system dynamics, i.e. $x_{t+1} \sim P(\cdot|x_t, u_t)$, and the agent receives a reward $r_{t+1} \sim R(x_{t+1}, u_t)$.

In an OPE problem, one wants to evaluate the performance of a target policy, $\pi_e$, based on historical data generated by a behaviour policy, $\pi_b$. We denote the historical data set as $D = \{\tau^i\}_{i=i}^N$, which is composed of N trajectories, each denoted as $\tau^i$. Each trajectory consists of state-action pairs over an episode, i.e. $\tau^i = \{x_t, u_t\}_{1:T}$. The objective of OPE is to estimate the value function of the target policy as given in Eq. 1, where $\gamma$ is a discount factor and $d_0$ is the initial state distribution [13].

$$V(\pi_e) = \mathbb{E}_{x\sim d_0}\left[\sum_{t=1}^{T} \gamma^{t-1} r_t | x_0 = x\right] \tag{1}$$

In the context of building controls, the behaviour policy would be the existing control, and the target policy would be the new control strategy to be evaluated. We assume historical operational data are readily available in existing buildings. We also assume the reward function is specified by the control engineers, and thus is known to us. In this work, the reward is simply the negative of the cost, and so we use these two terms interchangeably. We define each episode as one natural day, starting at midnight.

### 3.2 Approximate Model Method

As mentioned in Section 2, AM uses a model learned from historical data as a proxy for the actual environment, and evaluates the target policy through simulation in it [7]. More concretely, we summarize the AM method in Algorithm 1, where $\hat{P}$ denotes the AM learned from historical data, $D$. As there are numerous works on building modeling [10], one may refer to those to make informed decisions on modeling procedures and model forms. The initial state distribution, $d_0$, could be implemented as the empirical distribution from data.

**Algorithm 1:** Approximate Model Method

**Input:** The target policy, $\pi_e$; An AM of the environment $\hat{P}$ ;
**for** $i = 1, \ldots, N$ **do**
  $x_0 \sim d_0$;
  **for** $t = 0, \ldots, T\text{-}1$ **do**
    $u_t = \pi_e(x_t)$;
    $x_{t+1} = \hat{P}(x_t, u_t)$;
    $r_{t+1} = R(x_{t+1}, u_t)$;
  **end**
**end**
**Output:** $\hat{V}(\pi_e) = \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{t=1}^{T}\gamma^{t-1} r_t | x_0 = x\right]$;

### 3.3 Bootstrap Bias Estimation and Confidence Interval

Aside from the point estimate, i.e. $\hat{V}(\pi_e)$, it is also important to quantify uncertainty. Practically, the building stakeholders may be more interested in an agent that performs well in the worst case scenario, rather than one that performs well on average. While there are theoretical ways to derive confidence intervals, they generally require an impractical amount of data before they are tight enough to be useful [12]. Thus, we adopt the bootstrap method instead. We are also inspired by [3], which used bootstrapping to quantify the parameter uncertainty in identifying linear systems.

Bootstrapping uses random sampling with replacement to estimate properties of an estimator, such as bias, variance, or confidence intervals. We refer readers unfamiliar with this classical technique

to [14] for details. Specifically, we take $B$ bootstrap samples from $D$, and fit a new AM for each bootstrap sample. Then, we calculate bootstrap estimates of performance, denoted by $\hat{V}_i^*(\pi_e)$, following the same procedures in Algorithm 1. We use the bootstrap percentile interval following [12]. Furthermore, we use the bootstrap method to estimate and correct for bias [11]. The bias could be estimated as $\frac{1}{B}\sum_{i=1}^{B}\hat{V}_i^*(\pi_e) - \hat{V}(\pi_e)$Thus, the bias-corrected estimator could be calculated as $\hat{V}^{BC}(\pi_e) = \hat{V}(\pi_e) - \text{bias} = 2\hat{V}(\pi_e) - \frac{1}{B}\sum_{i=1}^{B}\hat{V}_i^*(\pi_e)$.

## 4 EXPERIMENT AND RESULTS

We generated a "historical" dataset from a simulation testbed, as described in Section 4.1. We pretrained multiple target policies through imitation learning on the dataset (Section 4.2). We developed an AM (Section 4.3), with which we evaluated the performance of the target policies. We summarized the results with comparison to ground truth performance in Section 4.4.

### 4.1 Simulation Testbed

The simulation testbed, as shown in Figure 1a, was modeled after the Intelligent Workspace (IW) on Carnegie Mellon University (CMU) campus. The IW is an EnergyPlus (E+) [2] model of a 600m$^2$ multi-functional space, including a classroom, a common area, and offices. We control the water-based radiant heating system, illustrated in Figure 1b. Specifically, we control the *supply water temperature*, so as to maintain the state variable, i.e. the *zone temperature*, close to its setpoint. In the existing control, the supply water (SW) is maintained at a constant flow rate, and its temperature is managed by a proportional (P) controller. For more information on the simulation test, refer to [15].

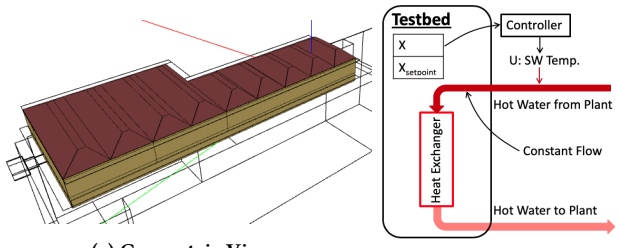
We generated a "historical" dataset from the simulation testbed using a baseline P-controller that was calibrated against data traces from the real system. The dataset was based on typical meteorological year 3 (TMY3) weather sequence from Jan. 1$^{st}$ to Mar. 31$^{st}$.

The cost at each time-step is a weighted sum of two terms that are proxies for energy and comfort (Eq. 2), where $x_{sp,t}$ is the *zone temperature setpoint*, and $\eta_t$ is a hyperparameter balancing comfort and energy, at time t. This cost function is justified because the heating demand of the system is linear to the *supply water temperature*, and the predicted percentage dissatisfied (PPD) is approximately quadratic to the deviation of the *zone temperature* from its setpoint, when the deviation is small. We used $\eta = 3$ when the space is occupied and $\eta = 0.1$ when it is not. Later analyses on energy and comfort performance is based on the cost defined here. We report averaged daily energy and comfort cost using $\gamma = 1$. The cost function here may be replaced by any performance metrics of interest to the building stakeholders.

$$C(x_{t+1}, a_t) = \underbrace{\eta_t (x_{t+1} - x_{sp,t+1})^2}_{\text{comfort}} + \underbrace{u_t}_{\text{energy}} \tag{2}$$

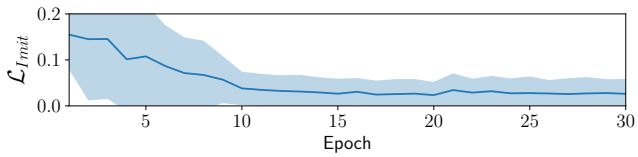
### 4.2 Target Policies

We pretrained the agents with imitating learning, a supervised approach for an agent to learn a policy. The premise is that it is easier for the expert to demonstrate the desired behaviour, compared to asking the expert to encode or fine-tune a policy. The imitation loss is given in Eq. 3, where $\hat{u}_t$ is the action by the learner. That

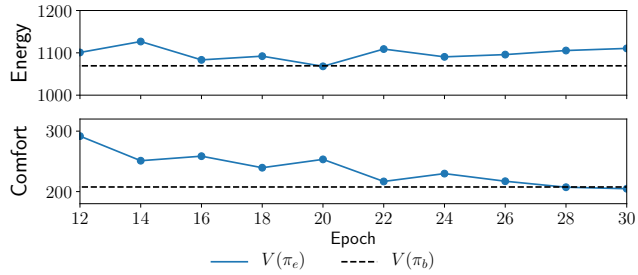



(a) Geometric View (b) System Schematic

**Figure 1: Simulation Testbed**




**(a) We pretrained target policies with imitation learning.**




**(b) Each data point represents the ground truth energy / cost cost of one target policy (out of ten) used for evaluation.**

**Figure 2: Pretraining Target Policies**

is to say, the learner tries to minimize the difference between its predicted actions with those of the expert.

$$\mathcal{L}_{\text{Imit}} = \sum_t (u_t - \hat{u}_t)^2 \tag{3}$$

Specifically, the policy we used is a 2-layer long short-term memory (LSTM) block with 8 hidden units, along with fully-connected layers as encoder for states and decoder for actions[1]. Denoting the planning horizon as $l$, the policy takes as input the current state $x_t$ and the future disturbances $d_{t:t+l-1}$, and outputs the predicted actions $u_{t:t+l-1}$. The disturbances include information on weather and occupancy. To avoid compounding errors, we let the policy predict actions multi-step ahead and minimize the imitation loss over the planning horizon. The number of steps was randomly sampled from 4 to 12. That is to say the planning horizon ranged from 1 to 3 hours, given a 15-min control time-step. We used ADAM [8] as the optimizer with a learning rate of $1 \times 10^{-3}$. The training loss decreases over epochs, as shown in Figure 2a.

To make the results reliable, we want to evaluate the proposed approach on multiple target policies. An easy way to implement

[1]We will make the code available

that is to use the policies at different number of training epochs. Specifically, we picked the policy at every other epoch starting from epoch 12, after which the learner is performing reasonably close to the expert. We evaluated the ground truth performance of the 10 selected target policies by running them in the simulation testbed, the result of which is summarized Figure 2b. While the imitation loss decreases over epochs, the cost and energy performance is more nuanced. The energy cost is lowest at the $20^{\text{th}}$ epoch, while the comfort cost is lowest at the $30^{\text{th}}$ epoch. Such information is not available from training loss. This reaffirms the need for OPE to make informed decision in policy selection. Finally, we would like to draw readers' attention to Figure 3a, where we included a comparison of a target policy with the behaviour policy, i.e. the baseline P-controller. The target policy, $\pi_e$ has similar actions as the P-controller, $\pi_b$, and does not suffer from compounding error due to our training procedure.

## 4.3 Approximate Modeling

We modeled the simulation testbed using an autoregressive model with exogenous variable (ARX) [10] given in Eq. 4, where $\vec{d}_t$ is a vector of disturbance terms, and $a_i$, $b_u$, and $\vec{b}_d$ are model parameters. The model order, i.e. $p = 12$, was determined by visually examining the partial autocorrection function. The model parameters are identified on the historical data using prediction error minimization [10]. The root mean squared error (RMSE) on a unseen test set based on the weather sequence in 2017 is 0.14°C.

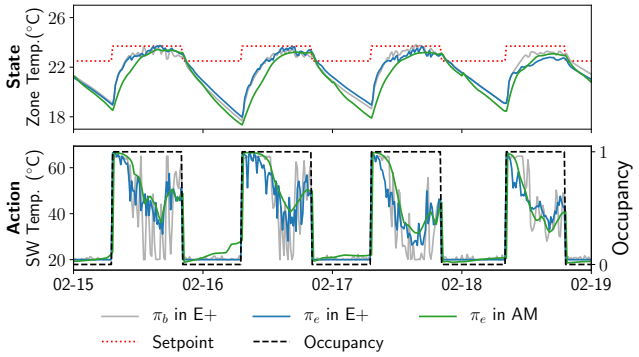
$$x_{t+1} = \sum_{i=0}^{p-1} a_i x_{t-i} + b_{u,t} u_t + \vec{b}_d \cdot \vec{d}_t \tag{4}$$

## 4.4 Results

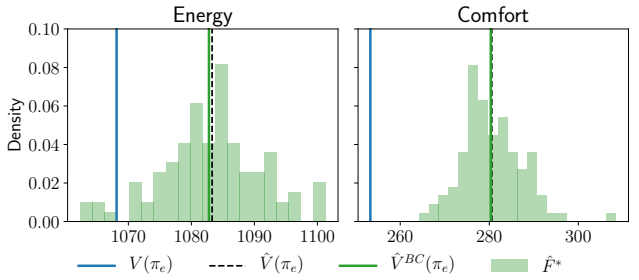
We demonstrate the use of the proposed approach on a specific target policy in Figure 3. Figure 3a compares the agent's behaviour in the simulation testbed vs. the AM, over a four-day period. The actions generated by the agent's policy, i.e. $\pi_e$, are similar in the EnergyPlus model and in the AM. At the same time, the state trajectory is qualitatively similar, despite some discrepancy. This means that an AM may serve as a reasonable proxy of the actual environment for performance evaluation.

Figure 3b compares the ground truth performance $V(\pi_e)$ vs. the estimate by the AM method $\hat{V}(\pi_e)$. In this case, the AM estimates the energy and comfort with 1.37% and 10.7% error respectively. We also used bootstrapping to quantify uncertainty and correct for bias. Specifically, from the state-action pairs in the historical dataset, we resampled with replacement 100 bootstrap samples of the same data size as the original dataset. Figure 3b also shows the empirical distribution of the bootstrap estimates, i.e. $\hat{F}^*$. The 95% confidence interval (CI) is bounded by value of $2.5^{\text{th}}$ and $97.5^{\text{th}}$ percentile of the empirical distribution. While the energy estimate falls within the CI, the comfort estimate does not. Furthermore, bias-correction barely changed the estimates, despite the clear bias.

We elaborate on the limitation of bootstrapping here. While it accounted for the parameter uncertainty [3], there are two sources of bias it did not account for. Firstly, we assumed a model form to explain the data, which allows us generalize to unseen state-action space. While this makes the OPE problem tractable, it also introduce




**(a) The AM serves as a proxy of the actual environment.**




**(b) We use bootstrapping to correct for bias and quantify uncertainty.**

**Figure 3: AM method and Bootstrapping**

a bias that is hard to quantify from data [7]. Secondly, the historical dataset covers a limited portion of the state-action space. Thus, the empirical distribution that we obtained the bootstrap samples from may not be representative of the true distribution, and introduces a bias stemming from this distribution mismatch.

Finally, Figure 4 compares the estimates by the AM method vs. the ground truth performance of the 10 target policies. Were the estimation perfect, all the data points would have fallen on the dashed black line. On average, the AM method estimated the energy and comfort cost with 1.84% and 14.1% error respectively. Note that the AM method systematically overestimates the cost and has little impact on the ordering of the performance. That is to say, one would have picked the correct policy if one were using the proposed method to select the best-performing policy.

## 5 CONCLUSIONS

To summarize, we introduce OPE to the building control community, a problem that should be addressed to enable real-world RL for building control. While there exists a rich literature on this topic, the characteristics of building operational data, i.e. generated by deterministic policy and limited coverage of the state-action space, present challenges for applying some of these methods to building control. By comparing existing methods, we determined that the AM method is a simple, yet feasible approach. We evaluated it on 10 different target policies in a simulation testbed. On average, the AM method estimated the energy and comfort cost with 1.84% and 14.1% error respectively.

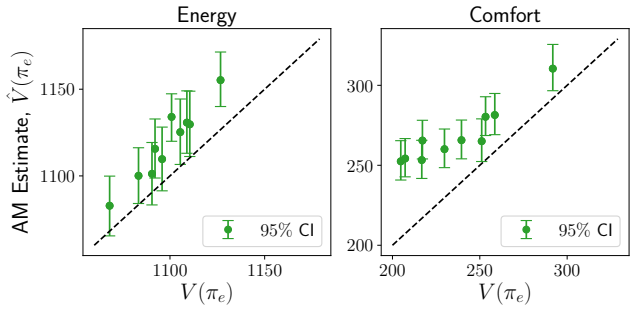



**Figure 4: Summary. AM Estimates vs. Ground Truth**

While AM shows promising results in this preliminary work and allows one to tap into existing knowledge on building modeling, it by no means precludes the possibility of other methods being more suitable. We attempted to use bootstrapping to correct for bias and quantify uncertainty, but some of the ground truth values are outside of the corresponding confidence interval. As discussed, there are two sources of biases that are not accounted for, and further work is required to address these issues.

## ACKNOWLEDGMENTS

To Robert, for the bagels and explaining CMYK and color spaces.

## REFERENCES

[1] Bingqing Chen, Zicheng Cai, and Mario Bergés. 2019. Gnu-rl: A precocial reinforcement learning solution for building hvac control using a differentiable mpc policy. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. 316–325.

[2] Drury B Crawley, Linda K Lawrie, Curtis O Pedersen, and Frederick C Winkelmann. 2000. Energy plus: energy simulation program. *ASHRAE journal* 42, 4 (2000), 49–56.

[3] Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. 2019. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics* (2019), 1–47.

[4] Gabriel Dulac-Arnold, Daniel Mankowitz, and Todd Hester. 2019. Challenges of real-world reinforcement learning. *arXiv preprint arXiv:1904.12901* (2019).

[5] Javier Garcıa and Fernando Fernández. 2015. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research* 16, 1 (2015), 1437–1480.

[6] Ruoxi Jia, Ming Jin, Kaiyu Sun, Tianzhen Hong, and Costas Spanos. 2019. Advanced building control via deep reinforcement learning. *Energy Procedia* 158 (2019), 6158–6163.

[7] Nan Jiang and Lihong Li. 2016. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*. 652–661.

[8] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[9] Doina Precup. 2000. Eligibility traces for off-policy policy evaluation. *Computer Science Department Faculty Publication Series* (2000), 80.

[10] Samuel Privara, Jiří Cigler, Zdeněk Váňa, Frauke Oldewurtel, Carina Sagerschnig, and Eva Žáčeková. 2013. Building modeling as a crucial part for building predictive control. *Energy and Buildings* 56 (2013), 8–22.

[11] Harald Steck and Tommi S Jaakkola. 2004. Bias-corrected bootstrap and model uncertainty. In *Advances in Neural Information Processing Systems*. 521–528.

[12] Philip Thomas and Emma Brunskill. 2016. Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*. 2139–2148.

[13] Cameron Voloshin, Hoang M Le, Nan Jiang, and Yisong Yue. 2019. Empirical Study of Off-Policy Policy Evaluation for Reinforcement Learning. *arXiv preprint arXiv:1911.06854* (2019).

[14] Larry Wasserman. 2013. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media.

[15] Zhiang Zhang and Khee Poh Lam. 2018. Practical Implementation and Evaluation of Deep Reinforcement Learning Control for a Radiant Heating System. In *Proceedings of the 5th Conference on Systems for Built Environments* (Shenzen, China) *(BuildSys '18)*. ACM, New York, NY, USA, 148–157.