Imperceptible Attacks on Fault Detection and Diagnosis Systems in Smart Buildings

Ismail R. Alkhouri, *Student Member, IEEE*, Akram S. Awad, *Student Member, IEEE*, Qun Z. Sun, *Member, IEEE*, and George K. Atia, *Senior Member, IEEE*

***Abstract*—Automated fault detection and diagnosis systems are critical to safe and efficient operation of smart buildings. A significant amount of building data can be collected and analyzed to detect building component failures. Attacks against such data that are contaminated with small additive disturbances (i.e., adversarial perturbation attacks) could dreadfully impact the performance of such systems while maintaining a high level of imperceptibility. The vulnerability studies of such data attacks is lacking. Specifically, most existing detection and classification models have flat structures, regarded as single-stage classifiers (SSCs), are prone to adversarial data perturbation attacks. In this article, we present a coarse-to-fine hierarchical fault detection and multilevel diagnosis (HFDD) model, and formulate a mathematical program to derive targeted attacks on the model with respect to a prespecified target diagnosis level. Two algorithms are developed based on convex relaxations of the formulated program for nontargeted attacks. An alternating direction method of multipliers-based solver is developed for the convex programs. Extensive experiments are conducted using two real-world datasets of measurements from air handling units and chillers, demonstrating the feasibility of the proposed attacks with regard to misclassification rate and imperceptibility of the attack. We also show that the HFDD is more robust to disturbances than SSC-based fault detection and multilevel diagnosis systems.**

***Index Terms*—Alternating direction method of multipliers (ADMM), adversarial additive disturbances, hierarchical fault detection and diagnosis (HFDD).**

I. Introduction

ACCORDING to the U.S. Department of Energy Buildings Energy Data Book, the buildings sector was responsible for about 41% of primary energy consumption in 2010, which exceeds the consumption of the transportation and industrial sectors by 44% and 36%, respectively [1]. Modern buildings make use of chillers, air handling units (AHUs), and advanced sensing and control platforms. With the increased number of interconnected components, faults tend to occur frequently in buildings. It is well known that a portion of the energy is wasted in the years after building commission due to the existence of various types of faults [2]. It is reported that 70% of commercial buildings need repairs [3]. As a result, automated fault detection and diagnosis (AFDD) studies have emerged over the last few decades. A study conducted by Lawrence Berkeley National Laboratory investigated 225 buildings with AFDD tools and found that 8% energy savings could be brought about by AFDD implementation [4]. The value of building AFDDs have been gradually recognized by building operators and the adoption of such platforms have become commonplace.

AFDD software continuously monitors building conditions and communicates with building automation systems. However, malicious actors could exploit existing vulnerabilities to attack critical building systems. In particular, many existing AFDD tools rely on black-box models that use data-driven approaches, such as neural networks, decision trees, Bayesian networks, and support vector machines (SVM) [5]. This is partly because these approaches require less modeling effort compared to physics-based methods, especially given the increased building sensing capability.

Although data-driven approaches are easy to implement, the accuracy of the model inputs is critical to avoid the "garbage in, garbage out" situation. To address some of the data challenges, recent studies have focused on selecting optimal data features [6], handling incomplete sensor measurements [7], and incorporating expert knowledge to detect unseen faults [8]. Semantic models with expert knowledge have also been used in AFDD [9], but they require large amounts of data and could be vulnerable to data breaches, creating a potential gateway for hackers to disrupt building operations. However, there is a scarcity of studies on data attacks against building AFDD algorithms.

Smart buildings have become a prime target for attackers seeking to weaken the security and operations of many organizations and companies, and the number of attacks has increased significantly in recent years. For instance, in 2013, a Google building was breached, resulting in access to sensitive information [10]. Therefore, protecting these buildings has become a top priority for companies. By studying malicious attacks, we can evaluate the robustness of the systems used in smart buildings. One possible approach is imperceptible attacks on data, which

Manuscript received 16 June 2022; revised 19 March 2023; accepted 9 June 2023. Date of publication 21 June 2023; date of current version 19 January 2024. This work was supported in part by the NSF CAREER Award under Grant CCF-1552497, in part by the NSF under Grant CCF-2106339, and in part by the DOE Award under Grant DE-EE0009152. Paper no. TII-22-2572. *(Corresponding author: Ismail R. Alkhouri.)*

The authors are with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32825 USA (e-mail: ismail.alkhouri@ucf.edu; akramawad@knights.ucf.edu; qz.sun@ucf.edu; george.atia@ucf.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2023.3288221.

Digital Object Identifier 10.1109/TII.2023.3288221

can manipulate sensor readings and trigger mistaken control actions, potentially leading to dire consequences.

One type of data attack is known as adversarial additive disturbance attacks. These attacks on observation and measurement vectors have been shown to successfully induce imperceptible misclassifications in many safety-critical systems, including image classifiers based on neural network systems [11]. Adversarial attacks are classified as nontargeted or targeted, depending on the attacker's goal. In a fault detection and diagnosis (FDD) system, if the predicted fault based on measurements $\mathbf{x}$ is denoted by $w(\mathbf{x})$, nontargeted attacks aim to add disturbances $\boldsymbol{\eta}$ that cause any misclassification, i.e., $w(\mathbf{x}+\boldsymbol{\eta}) \neq w(\mathbf{x})$. In contrast, targeted attacks aim to design $\boldsymbol{\eta}$ such that $w(\mathbf{x}+\boldsymbol{\eta}) = t$, where t is a prespecified target label [11].

Existing research on adversarial attacks has primarily focused on single-stage classifiers (SSCs) [11], which are commonly used in many AFDD algorithms. However, in this work, we investigate targeted and nontargeted attacks on hierarchical coarse-to-fine fault detection and multilevel diagnosis (HFDD) models that use a multistage approach to classify sensor measurements as either "faulty" or "nonfaulty." If a fault is detected, the system performs multilevel diagnosis to determine the root cause of the fault. We investigate these attacks in the white-box setting, where the attacker has access to the classification model. To the best of our knowledge, our study is the first to explore adversarial attacks on building AFDD systems using this type of multistage approach.

Contributions: The contributions of this work are summarized as follows. First, we propose an HFDD model that incorporates more than two fault intensity diagnosis levels, using SVMs. Second, we develop a convex program for generating targeted attacks, which induce disturbances in the diagnosis level to change the prediction to a specified target label. Third, we introduce two algorithms for nontargeted attacks: the first builds upon our targeted attack formulation, while the second is based on a convex formulation for SSCs. We develop an efficient iterative algorithm based on the alternating direction method of multipliers (ADMM), which outperforms the standard CVX solver for disciplined convex programming [12]. Finally, we present extensive experimental results using two real-world building dataset benchmarks, demonstrating the effectiveness of our proposed attacks in terms of attack success rate and imperceptibility.

Notation: We use bold uppercase letters to represent matrices and bold lowercase letters for vectors, unless otherwise specified. For any positive integer L, $[L] := \{1, 2, \ldots, L\}$. For sets A and B, the set difference $A \setminus B$ denotes the elements in A that are not in B. The cardinality of set A is denoted by $|A|$.

A. Related Work

There exist numerous learning and statistical based implementations of FDD systems, including ones that make use of convolutional neural networks, autoencoders, SVMs, sparse filtering, and deep belief networks (DBN) [5], [13]. Our proposed system considers a hierarchical version of conventional FDD systems [14]. Our HFDD is shown to be more robust against additive measurement disturbances in comparison to FDDs that do not use a hierarchical structure as demonstrated in the experimental results.

In order to ensure the security of smart buildings, it is important to develop attacks that can evaluate their vulnerabilities. These attacks can be categorized based on layers: i) field layer attacks, which aim to disrupt the operation of sensors, actuators, and controllers, and ii) management layer attacks, such as denial of service attacks [10]. Our proposed method falls under the field layer attacks category, as it targets the measurements of sensors. However, most attack approaches under this category focus on wireless protocols in order to establish remote control, as demonstrated in previous work such as [15]. In this article, we consider the actual sensor readings as the target of our attack, as the goal is to generate imperceptible perturbations that can deceive FDD systems.

The use of hierarchical structures for fault detection and recognition are studied in [16] and [17]. The authors in [16] considered a hierarchical feature enhancement DBN-based model, and train the hierarchical structure as one entity. In our HFDD formulation, the local classifiers are trained disjointly, which makes attacking the system more challenging since the attacker needs to fool many local classifiers and classification levels. The work in [17] proposed a two-stage hierarchical fault recognition network based on DBNs and wavelet packet transform. The authors show the effectiveness of their model against noise and other disturbances. Our HFDD formulation, however, considers a multistage fault diagnosis system. More importantly, we investigate robustness against malicious, carefully designed, additive perturbations.

It is important to distinguish our work from poisoning attacks [18], where the adversary targets the benign data during the training stage. Here, we consider additive perturbations for a deployed system at inference time and not in the training phase of the local classifiers. Our attacks also differ from false data injection attacks, such as the work in [19], as we neither require to determine the optimal attack region nor the timing of the attack. Our approach considers generating imperceptible disturbances to actual sensor reading measurements in HFDDs.

The authors in [20] proposed nontargeted attacks on two-stage coarse-to-fine classifiers, whereas our work extends the formulation to hierarchical models with an arbitrary number of stages and considers both targeted and nontargeted attacks. Moreover, our model employs SVMs instead of neural networks. In a separate work [21], nontargeted attacks on HFDDs with two diagnosis levels were presented. In contrast, we expand the formulation to accommodate an arbitrary number of diagnosis levels, introduce targeted and nontargeted attacks with respect to the target attack level, and evaluate our approach on additional building benchmarks.

II. HFDD Model Formulation

We consider a trained coarse-to-fine HFDD system. The first coarse detection stage determines whether the measurement vector is "faulty" or "nonfaulty." If the signal is detected as faulty, multilevel fine classifiers are used to predict the root cause

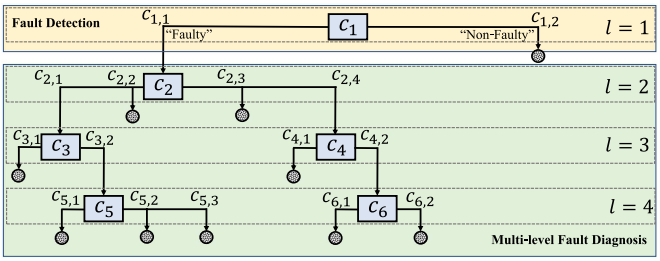

Fig. 1. Example of an HFDD structure with $L = 4$ levels and $Q = 6$ classifiers. The first level uses the binary classifier c_1 to determine whether a measurement vector is labeled "faulty" (label $c_{1,1}$) or "non-faulty" (label $c_{1,2}$). The subsequent levels (levels 2, 3, and 4) use local predictors c_2 to c_6 for the multilevel fault diagnosis task. The considered labels for each local classifier c_p are shown on the sides of the respective classifiers. The dotted circles represent end nodes.

and/or intensity of the fault. The classification level is indexed by $l \in [L]$, where $l = 1$ indicates the first binary classification fault detection level.

We assume that the HFDD comprises Q local classifiers, including the first binary detector. We use the notation $c_p : \mathbb{R}^N \to M_p$ to denote the predictor $p \in [Q]$, which assigns a label from the set of possible labels M_p to a measurement vector $\mathbf{x} \in \mathbb{R}^N$. Given a data point $\mathbf{x}$, the predicted label by the classifier c_p is obtained by maximizing over $|M_p|$ discriminant functionals $f_j(\mathbf{x})$, as follows:

$$c_p(\mathbf{x}) = \underset{j \in M_p}{\operatorname{argmax}}\, f_j(\mathbf{x}). \tag{1}$$

The labels in the HFDD structure are denoted as $c_{p,d}$, where p represents the local classifier used, and d denotes the classification decision of that particular classifier.

We define the set $S(l)$ consisting of the entries $c_{p,d}$ of all classifiers indexed by p in level l along with their decisions $d \in M_p$. As an example, in Fig. 1, the set $S(3) = \{c_{3,1}, c_{3,2}, c_{4,1}, c_{4,2}\}$. Furthermore, we define the route set $R(c_{p,d})$, which consists of all labels from the root to label $c_{p,d}$ given the structure of the HFDD. For example, for the HFDD in Fig. 1, $R(c_{4,2}) = \{c_{1,1}, c_{2,4}, c_{4,2}\}$. We also define $T(c_{p,d})$ as the set of indices of the local classifiers used to reach label $c_{p,d}$. Hence, in Fig. 1, $T(c_{6,2}) = \{1, 2, 4, 6\}$.

The classification at level l is expressed by $w(\mathbf{x}, l)$ and is obtained as $w(\mathbf{x}, l) = \operatorname{argmax}_{m \in S(l)} q_m(\mathbf{x})$, where $q_m(\mathbf{x}) \in \{0, 1\}$ is the discriminant functional obtained as

$$q_m(\mathbf{x}) = \mathbf{1}\left\{f_{c_{p,d}}(\mathbf{x}) > f_k(\mathbf{x}), \forall p \in T(m), \forall k \in M_p \setminus \{c_{p,d}\}\right\} \tag{2}$$

where, $\mathbf{1}\{.\}$ is the indicator function that returns 1 if the condition in its argument is true, and 0 otherwise. Equation (2) outlines the conditions that must be met by the local classifiers indexed by $T(m)$ to determine the classification as m.

We note that, if the measurement vector $\mathbf{x}$ is classified as end node $c_{p,d} \in S(l)$ for $l < L$, i.e., there is no subsequent classification stage, then we fix the same classification for all subsequent levels from l up to L. For example, in Fig. 1, if the predicted label of the observation $\mathbf{x}$ is $w(\mathbf{x}, 2) = c_{2,3}$, then we also have $w(\mathbf{x}, 3) = w(\mathbf{x}, 4) = c_{2,3}$. This is used to simplify the description of the upcoming proposed algorithms.

We note that the HFDD model is trained to detect and classify faulty measurements by learning from similar instances in the training dataset. These instances include measurements of subsystem malfunctions that belong to the same labels as the test measurements. However, the HFDD model is not designed to detect cyber-attacks. In this article, we investigate the impact of adversarial measurements on HFDD systems as a first step toward enhancing their robustness against this specific type of attack.

The HFDD system in this article is composed of local classifiers based on SVMs, including both binary and nonbinary classifiers. For the binary case, we assume that we have a training dataset $X_{\text{tr}} = \{\mathbf{x}_j, y_j\}_{j=1}^m$, where $\mathbf{x}_j \in \mathbb{R}^N$ is an observation vector and $y_j \in \{+1, -1\}$ is its corresponding label. Let $\Phi : \mathbb{R}^N \to \mathbb{R}^F$ denote a mapping to a high-dimensional feature space of dimension F. For any measurement $\mathbf{x} \in \mathbb{R}^N$, the SVM discriminant functional is given by $J(\mathbf{x}) = \mathbf{w}^T\Phi(\mathbf{x}) + b$, where $\mathbf{w} \in \mathbb{R}^{\mathbf{F}}$ and $b \in \mathbb{R}$ are the normal vector and the bias of the separating hyperplane, respectively. The normal vector is obtained as $\mathbf{w} = \sum_{j\in[m]} \alpha_j y_j \Phi(\mathbf{x}_j)$, which results in $J(\mathbf{x}) = \sum_{j\in[m]} \alpha_j y_j k(\mathbf{x}, \mathbf{x}_j) + b$, where $\alpha_j \geq 0$ is a Lagrangian multiplier representing observation vector $\mathbf{x}_j$, and $k(\mathbf{x}, \mathbf{x}_j) = \Phi(\mathbf{x})^T\Phi(\mathbf{x}_j)$ is the kernel function obtained as the dot product of the feature maps. Without loss of generality, we utilize the polynomial kernel to represent the discriminant functional. Hence, $J(\mathbf{x})$ is obtained as

$$J(\mathbf{x}) = \sum_{j\in[m]} \alpha_j y_j \left[\gamma \mathbf{x}^T\mathbf{x}_j + \beta\right]^d + b \tag{3}$$

where, γ and β are hyperparameters used to adjust the kernel function, and d is the degree of the polynomial kernel.

To handle the case where the local HFDD predictor p performs nonbinary classification, i.e., with $|M_p| > 2$ labels, we employ the traditional one versus all (OVA) method. Specifically, we train $|M_p|$ binary SVM classifiers, with each classifier assigned to distinguish between one label and the rest of the labels. Thus, for each binary classifier, we obtain a discriminant functional, which corresponds to the SVM model's output. The final prediction for a given data point is obtained by selecting the index of the largest discriminant functional. It is worth noting that each binary classifier in the OVA method has its own set of α_j's and b's.

III. Attacks on HFDDs

In this section, we propose methods for generating additive and imperceptible perturbations, $\boldsymbol{\eta} \in \mathbb{R}^N$, aimed at inducing misclassifications in the HFDD model presented in Section II. Given the multilevel structure of the HFDD, attacks are developed w.r.t. a certain level $l \in [L]$. We consider both targeted and nontargeted attack scenarios.

Targeted attacks: Here, the goal of the attack is to alter the classification of level l to target label $t \neq w^*$, where $w^* := w(\mathbf{x}, l)$. In other words, the perturbation $\boldsymbol{\eta}$ is crafted such that

Algorithm 1: nTAH-Algorithm for Nontargeted Attacks.

Input: $\mathbf{x}$, w, l.
Output: $\boldsymbol{\eta}^*$
for $m \in S(l)\backslash\{w\}$
  obtain $\boldsymbol{\eta}_m$ from (5) with $t = m$.
  if $w(\mathbf{x}+\boldsymbol{\eta}_m, l) \neq w(\mathbf{x}, l)$
    obtain $D(\boldsymbol{\eta}_m)$
$\boldsymbol{\eta}^* = \underset{m\in S(l)\backslash\{w\}}{\operatorname{argmin}} D(\boldsymbol{\eta}_m)$

Algorithm 2: nPath Algorithm for Nontargeted Attacks.

Input: $\mathbf{x}$, w, l.
Output: $\boldsymbol{\eta}^*$
for $p \in T(w)$
  obtain $\boldsymbol{\eta}_p$ w.r.t. c_p from (7).
  if $w(\mathbf{x}+\boldsymbol{\eta}_p, l) \neq w(\mathbf{x}, l)$
    obtain $D(\boldsymbol{\eta}_p)$
$\boldsymbol{\eta}^* = \underset{p\in T(w)}{\operatorname{argmin}} D(\boldsymbol{\eta}_p)$

$w(\mathbf{x}+\boldsymbol{\eta}, l) = t$. Hence, it is required that

$$f_{c_{p,d}}(\mathbf{x}+\boldsymbol{\eta}) > f_k(\mathbf{x}+\boldsymbol{\eta}), \forall p \in T(t), \forall k \in M_p \setminus \{c_{p,d}\} \tag{4}$$

which ensures that each local predictor along the path of the target makes a decision in favor of ultimately classifying $\mathbf{x}+\boldsymbol{\eta}$ as label t. To ensure the efficiency of the attack, it should also remain undetectable. To this end, we propose a mathematical program to minimize the distance function $D(\boldsymbol{\eta}) := \|\boldsymbol{\eta}\|_2^2$ subject to the constraints in (4). In order to maintain convexity, we use the first order Taylor series expansion to decompose the functionals in (4) and yield linear constraints in $\boldsymbol{\eta}$. Further, we introduce a small constant $\epsilon_r > 0$ to transform the strict inequalities to bounded ones. Thus, we formulate the convex program

$$\begin{aligned} &\min_{\boldsymbol{\eta}} D(\boldsymbol{\eta}) \quad \text{subject to} \\ &\boldsymbol{\eta}^T \left(\nabla_{\mathbf{x}} f_{c_{p,d}}(\mathbf{x}) - \nabla_{\mathbf{x}} f_k(\mathbf{x})\right) \geq f_k(\mathbf{x}) - f_{c_{p,d}}(\mathbf{x}) + \epsilon_r \\ &\forall p \in T(t), \forall k \in M_p \backslash \{c_{p,d}\} \end{aligned} \tag{5}$$

to generate additive disturbances. We term this method targeted attack on HFDD (TAH). Hence, the attacker increases the imperceptibility of the attack by minimizing the objective function D, and a number $|T(t)||M_p - 1|$ of constraints is used to induce false prediction to target label t.

Nontargeted attacks: The goal of nontargeted attacks is to generate small additive perturbations to fool the classification at level l in the HFDD, that is, enforcing that $w(\mathbf{x}+\boldsymbol{\eta}, l) \neq w(\mathbf{x}, l)$, while being imperceptible. To this end, we propose the following two algorithms.

1) nTAH-Algorithm: In this method, which we call nontaregeted attacks using the TAH formulation, we leverage the program in (5) to generate perturbations where in every iteration, the target class is selected as $m \in S(l) \setminus \{w\}$. If the generated perturbation satisfies the goal of the attack, the corresponding index is stored. Then, in the last step, the smallest disturbance (w.r.t. distance function D) is chosen. This procedure is described in Algorithm 1.

2) Nontargeted Path-Based (nPath)-Algorithm: In this method, which we dub as the nPath method, we enforce misclassification independently for the predictions of the classifiers along the path of the predicted label w. In other words, the attacker seeks to fool c_p, $\forall p \in T(w)$ separately.

To fool the classifier c_p independently of all possible upper coarser levels, the disturbances vector must satisfy that $c_p(\mathbf{x}+\boldsymbol{\eta}) \neq c_p(\mathbf{x})$. From (1), this amounts to the requirement that

$$\exists k \in M_p \backslash \{c_p(\mathbf{x})\} : f_k(\mathbf{x}+\boldsymbol{\eta}) > f_{c_p(\mathbf{x})}(\mathbf{x}+\boldsymbol{\eta}) \,. \tag{6}$$

To encode the existence condition in (6), we choose the second maximizing label before adding the perturbation, i.e., $k^* = \operatorname{argmax}_{k\in M_p\backslash\{c_p(\mathbf{x})\}} f_k(\mathbf{x})$. Similar to (5), we make use of the first order Taylor series expansion to formulate the convex program

$$\begin{aligned} &\min_{\boldsymbol{\eta}} D(\boldsymbol{\eta}) \quad \text{subject to} \\ &\boldsymbol{\eta}^T (\nabla_{\mathbf{x}} f_{k^*}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{c_p(\mathbf{x})}(\mathbf{x})) \geq f_{c_p(\mathbf{x})}(\mathbf{x}) - f_{k^*}(\mathbf{x}) + \epsilon_n \end{aligned} \tag{7}$$

to generate a disturbance to fool c_p independently. The constant $\epsilon_n > 0$ is used for the same purpose as ϵ_n in (5). We call this method fooling classifiers independently (FIN) attack.

The nPath procedure, presented in Algorithm 2, generates perturbations for every classifier along the path of the predicted label and tests whether it satisfies the requirement. If successful, it selects the minimum w.r.t. distance function D.

We remark that the FIN convex formulation in (7) may not be suitable for generating *targeted* attacks on some level $l > 1$ (diagnosis level) for two reasons. First, the constraint does not enforce a prespecified target label from a set $T(t)$. Second, the generated disturbances that alter the classification to a certain class w.r.t. one local classifier in the set $T(t)$ may not necessarily lead to the next class in the route set $R(t)$.

In addition to using FIN in Algorithm 2, the state-of-the-art approach proposed for SSCs in [22], which utilizes Taylor series approximation to result in linear constraints, will be utilized as a benchmark attack on nonhierarchical FDD systems.

ADMM-based solver: Here, we develop an ADMM-based solver for the convex programs (5) and (7). First, we introduce the matrix $\mathbf{G} \in \mathbb{R}^{N\times V}$ whose columns are obtained as

$$\mathbf{G} = \begin{cases} \left[\nabla_{\mathbf{x}} f_{c_{p,d}}(\mathbf{x}) - \nabla_{\mathbf{x}} f_f(\mathbf{x})\right] \\ \forall p \in T(t), \forall k \in M_p \backslash \{c_{p,d}\}, \text{ for (5)} \\ \left[\nabla_{\mathbf{x}} f_{k^*}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{c_p(\mathbf{x})}(\mathbf{x})\right], \text{ for (7)} \end{cases} \tag{8}$$

and vector $\mathbf{b} \in \mathbb{R}^V$ with entries

$$\mathbf{b} = \begin{cases} \left[f_k(\mathbf{x}) - f_{c_{p,d}}(\mathbf{x}) + \epsilon_r\right]^T \\ \forall p \in T(t), \forall k \in M_p \backslash \{c_{p,d}\}, \text{ for (5)} \\ \left[f_{c_p(\mathbf{x})}(\mathbf{x}) - f_k(\mathbf{x}) + \epsilon_n\right]^T, \text{ for (7)} \,. \end{cases} \tag{9}$$

The value of V is $|T(t)||M_p - 1|$ for program (5) and 1 for program (7). We introduce a slack variable $\mathbf{z} \in \mathbb{R}^V$, and write the minimization in the standard form of ADMM as [23]

$$\min_{\boldsymbol{\eta},\mathbf{z}} D(\boldsymbol{\eta}) + E(\mathbf{z}) \quad \text{subject to} \quad \mathbf{G}^T\boldsymbol{\eta} - \mathbf{b} - \mathbf{z} = \mathbf{0} \tag{10}$$

where, $E(\mathbf{z})$ is the penalty function that is equal to 0 if $\mathbf{z} \geq \mathbf{0}$ and $+\infty$ otherwise, and is used to write the inequality constraints as equalities, which is necessary for the standard ADMM form.

The augmented Lagrangian can be written as $\mathcal{L}_\lambda(\boldsymbol{\eta}, \mathbf{z}, \boldsymbol{\mu}) = \|\boldsymbol{\eta}\|_2^2 + E(\mathbf{z}) + \frac{\lambda}{2}(\|\mathbf{G}^T\boldsymbol{\eta} - \mathbf{b} - \mathbf{z} - \boldsymbol{\mu}\|_2^2 - \|\boldsymbol{\mu}\|_2^2)$, where λ is a penalty factor, and $\mu \in \mathbb{R}^V$ is the Lagrangian multiplier. We can readily formulate the steps of the ADMM for each iteration τ [23] as follows.

1) Given that $\nabla_{\boldsymbol{\eta}} D(\boldsymbol{\eta}) = 2\boldsymbol{\eta}$, we obtain $\boldsymbol{\eta}^{(t)}$ by minimizing the Lagrangian function w.r.t $\boldsymbol{\eta}$ while variables $\mathbf{z}$ and $\boldsymbol{\mu}$ are held constant. The closed-form solution is found as $\boldsymbol{\eta}^{(\tau+1)} = -\lambda(2\mathbf{I}_N + \lambda\mathbf{G}\mathbf{G}^T)^{-1}\mathbf{G}(\mathbf{b} + \mathbf{z}^{(\tau)} + \boldsymbol{\mu}^{(\tau)})$, where $\mathbf{I}_N$ is the identity matrix of size $N \times N$.
2) Similarly, update the slack variable $\mathbf{z}$ as $\mathbf{z}^{(\tau+1)} = \max(\mathbf{0}, \mathbf{G}^T\boldsymbol{\eta}^{(\tau)} - \mathbf{b} + \boldsymbol{\mu}^{(\tau)})$.
3) Update the Lagrangian as $\boldsymbol{\mu}^{(\tau+1)} = \boldsymbol{\mu}^{(\tau)} + \mathbf{G}^T\boldsymbol{\eta}^{(\tau+1)} - \mathbf{b} - \mathbf{z}^{(\tau+1)}$.

The steps are repeated for a prespecified number of iterations $\mathcal{T}$.

Given the SVM classification described in the previous section, we obtain $\nabla_{\mathbf{x}} f_i(\mathbf{x}) = \nabla_{\mathbf{x}} J_i(\mathbf{x}) = d\gamma \sum_{j\in[m]} \alpha_j y_j \mathbf{x}_j [\gamma \mathbf{x}^T\mathbf{x}_j + \beta]^{d-1}$, where i is some index from the pool of labels of the classifier of interest.

Computational complexity: The computational complexity of the presented methods corresponds to solving a convex program with N variables and only V linear constraints. The complexity of an iterative convex program solver is decided based on the initialization procedure, the worst case complexity of an iteration for a given target precision, and its rate of convergence [23]. Given our ADMM, the initialization process consists of calculating matrix $\mathbf{G}$ and vector $\mathbf{b}$. The computational complexity per iteration is $\mathcal{O}(NV)$, i.e., linear in the length of the primal N and the length of the slack variable V. The algorithm can obtain an ϵ_1-approximate solution in $\mathcal{O}(1/\epsilon_1)$ iterations [24].

IV. Experimental Results

In this section, we evaluate our proposed attacks on two HFDD examples using two building benchmark datasets that are provided by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) [25].

Datasets description: The first dataset is ASHRAE project 1312-RP. In this dataset, a diverse number of fault types have been modeled and generated for three different seasons (summer, winter, and spring). Sensor data (including fan speed, humidity, temperature, etc.) are generated using two identical AHUs (A and B) serving two similar zones. All fault experiments are conducted using AHU-A, while AHU-B is always running under normal conditions. The fault experiments are implemented for two to three weeks, with each fault test lasting for one day, and the data are sampled every one minute. Each fault day (from AHU-A) is compared with its counterpart normal days (from AHU-B) to identify each fault type. For our experimental analysis, we select 32 days with seven normal days and 25 faulty days (from the spring and summer seasons) during the occupancy mode (6:00 am–6:00 pm). The second dataset is ASHRAE project 1312-RP. Here, the experimental data is generated from a 90-ton centrifugal water-cooled chiller [26]. The chiller system consists of a shell-and-tube evaporator, a shell-and-tube condenser, a pilot-driven expansion valve, and a centrifugal compressor. Seven types of faults are investigated, according to the survey [27], each with four intensities. The data were collected with one second resolution.

Performance metrics: We evaluate the performance of the proposed attacks based on two metrics. First, the attack success ratio, denoted ζ, which is defined for targeted attacks as the ratio of the number of observations $\mathbf{Z}$ classified according to the prespecified target labels to the total number of observations $|X|$, and for nontargeted attacks as the fraction of misclassified instances. Second, the perceptibility factor ρ_p which is the ratio of the ℓ_p-norms of the perturbation $\boldsymbol{\eta}$ and the measurement vector $\mathbf{x}$ [11]. Formally, the two metrics are given as $\zeta = \mathbf{Z}/|X|$ and $\rho_p = \|\boldsymbol{\eta}\|_p/\|\mathbf{x}\|_p$. We use σ_p to denote the average perceptibility factor over the set of interest X. Furthermore, we examine the performance of the HFDD under nontargeted attacks using the multiclass confusion matrix or contingency table.

Experimental setup: For the ASHRAE 1312-RP dataset, we train our proposed HFDD local classifiers with 70% of the standardized AHU-A normal and faulty data, and the remaining 30% are used for testing. Table I presents the six main fault types considered with each fault category having two or three intensities/severities as further refinements. This HFDD model consists of four levels, one level for fault detection and three levels for fault diagnosis and refinement, as shown in Fig. 1. We make use of the SVM polynomial kernel function of degree 3 with five-fold cross validation to train our HFDD local classifiers. For the ASHRAE 1043-RP dataset, we utilize the method in [28] to select the steady-state data for each fault type. For each fault intensity, we randomly select 600 steady-state data samples, which yields a total of 16 800 faulty samples. Similar ASHRAE 1312-RP, we use 70% (30%) of the normal and faulty standardized data for training (testing). Here, the HFDD is a three-level system that uses SVM polynomial kernel function of degree 2 for training the fault detection classifier, and of degree 3 for the other local classifiers. Seven main fault categories are considered, each refined into four intensities, as shown in Table II. The HFDD model is illustrated in Fig. 3. All experiments are implemented using MATLAB2021 with AMD Ryzen 7-4800H CPU @2.9 GHz machine.

Parameters selection: Here, we show examples of the selecting parameters ϵ_r, ϵ_n, $\mathcal{T}$, and λ. We consider the nontargeted nTAH attack on level 2 of the ASHRAE 1312-RP.

We use $\epsilon_r(l)$ to denote the value of ϵ_r of the constraints of level l in the convex program (5), while $\sigma_2(l)$ and $\zeta(l)$ are used to represent the average perceptibility factor and success ratio for level l, respectively. From Fig. 4, we see that when $\epsilon_r(1)$ ($\epsilon_r(2)$) increases, the success ratio $\zeta(1)$ ($\zeta(2)$) and the average perceptibility factor $\sigma_2(1)$ ($\sigma_2(2)$) increase proportionally. Moreover,

TABLE I
FAULT CATEGORIES AND THEIR CORRESPONDING INTENSITIES FOR THE ASHRAE PROJECT 1312-RP

Fault category	Intensity level		
	1	2	3
Cooling coil valve stuck (CCVS)	fully closed	fully open	partially open (15%, 50%, 65% open)
Exhausted air damper stuck (EADS)	fully open	fully closed	40% open
Outdoor air damper stuck (OADS)	fully closed	40% open	–
Return fan (RF)	fixed speed (20%, 30%, 80% speed)	complete failure	–
Heating coil valve leaking (HCVL)	stage1 (0.4 GPM)	stage2 (1.0 GPM)	stage3 (2.0GPM)
AHU duct leaking (AHUDL)	after supply fan	before supply fan	–

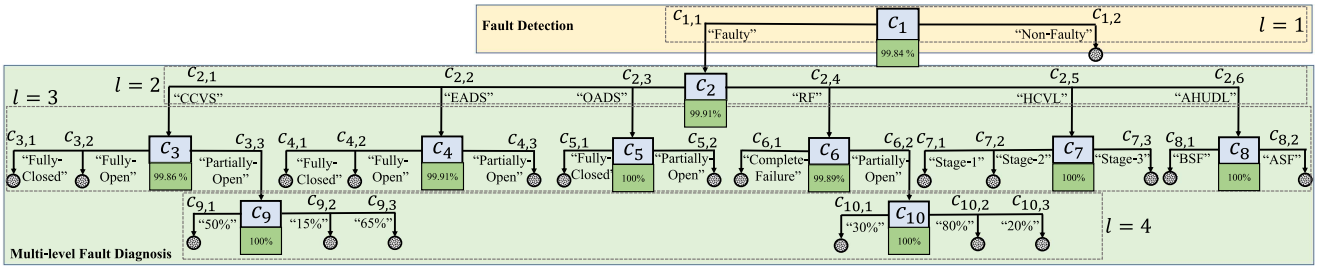

Fig. 2. HFDD model of the ASHRAE 1312-RP. The accuracy of each local classifier is shown in green squares.

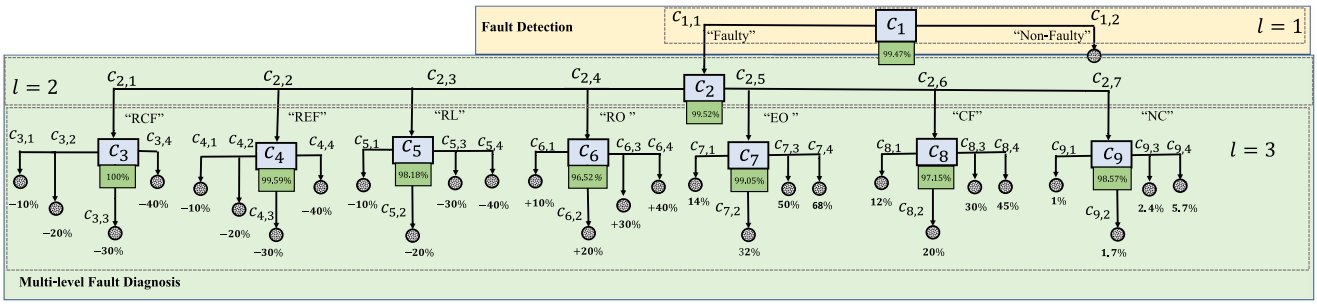

Fig. 3. HFDD model of the ASHRAE 1043-RP. The accuracy of each local classifier is shown in green squares.

TABLE II
FAULT CATEGORIES AND THEIR CORRESPONDING INTENSITIES FOR THE ASHRAE PROJECT 1043-RP

Fault category	Intensity level			
	1	2	3	4
Reduced condenser water flow (RCF)	−10%	−20%	−30%	−40%
Reduced evaporator water flow (REF)	−10%	−20%	−30%	−40%
Refrigerant leak (RL)	−10%	−20%	−30%	−40%
Refrigerant overcharge (RO)	+10%	+20%	+30%	+40%
Excess oil (EO)	14%	32%	50%	68%
Condenser fouling (CF)	12%	20%	30%	45%
Noncondensable gas in refrigerant (NC)	1%	1.7%	2.4%	5.7%

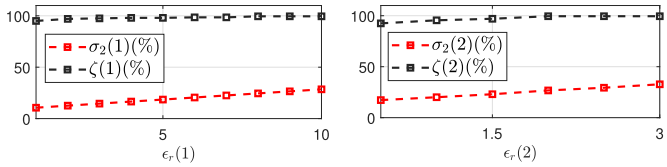

Fig. 4. Constant $\epsilon_r(l)$ for level 1 (*left*) and level 2 (*right*) as a function of the average perceptibility factor $\sigma_2(l)$ and success ratio $\zeta(l)$ averaged over 200 trials.

as $\epsilon_r(1)$ ($\epsilon_r(2)$) exceeds 5 (1.5), $\zeta(1)$ ($\zeta(2)$) saturates but $\sigma_2(1)$ ($\sigma_2(2)$) continues to increase. Hence, we choose $\epsilon_r(1) = 5$ and $\epsilon_r(2) = 1.5$. This selection is based on achieving a high success ratio and low perceptibility factor. For both datasets, a similar procedure is used to select the values for ϵ_r and ϵ_n. Results are presented in Table III. Note that the values of ϵ_r and ϵ_n are the same for the level 1 attack of the same dataset, due to the use of a binary classifier at the initial detection level.

The results presented in Table IV illustrate an example of selecting the hyperparameters λ and $\mathcal{T}$ of the proposed ADMM solver. The results are given in terms of the average perceptibility factor σ_2 and the success ratio ζ for different values of λ at

TABLE III
THE SELECTED VALUES OF ϵ_r AND ϵ_n FOR EACH ATTACK LEVEL FOR THE ASHRAE 1312-RP AND 1043 DATASETS

HFDD Level l	1312-RP		1043-RP	
	ϵ_r	ϵ_n	ϵ_r	ϵ_n
1	{5}	5	{4}	4
2	{5, 1.5}	2	{4, 1.5}	1.5
3	{5, 1.5, 0.1}	2	{4, 1.5, 0.5}	1.5
4	{5, 1.5, 0.1, 0.01}	2	–	–

$\mathcal{T} = 10$, 30, and 50. As λ and $\mathcal{T}$ increase, both ζ and σ_2 increase proportionally. It is observed that for a constant value of $\mathcal{T}$, such as $\mathcal{T} = 50$, ζ stops increasing as λ reaches a certain threshold ($\lambda = 0.2$ for $\mathcal{T} = 50$), while σ_2 continues to increase.

TABLE IV
ADMM PARAMETERS, λ AND $\mathcal{T}$, AS A FUNCTION OF THE PAIR $\{\zeta(\%), \sigma_2(\%)\}$ AVERAGED OVER 200 TRIALS

ADMM parameter λ	$\mathcal{T}=10$ $\{\zeta(\%), \sigma_2(\%)\}$	$\mathcal{T}=30$ $\{\zeta(\%), \sigma_2(\%)\}$	$\mathcal{T}=50$ $\{\zeta(\%), \sigma_2(\%)\}$
0.01	$\{25.00, 2.16\}$	$\{46.50, 6.75\}$	$\{73.0012.17\}$
0.05	$\{52.50, 8.78\}$	$\{88.00, 18.45\}$	$\{94.50, 21.69\}$
0.10	$\{76.00, 15.11\}$	$\{92.50, 22.20\}$	$\{96.50, 24.56\}$
0.15	$\{82.50, 18.37\}$	$\{93.50, 23.47\}$	$\{96.50, 24.56\}$
0.20	$\{87.00, 20.75\}$	$\{93.50, 24.11\}$	$\{\mathbf{97.00\ 25.19}\}$
0.25	$\{89.00, 22.54\}$	$\{93.50, 24.57\}$	$\{97.00, 25.43\}$

TABLE V
PENALTY FACTOR (λ) FOR EACH LEVEL ATTACK FOR ASHRAE 1312-RP AND 1043-RP DATASETS

HFDD level l	1312-RP			1043-RP		
	nPath	nTAH	Targeted	nPath	nTAH	Targeted
1	0.015	0.015	0.015	0.015	0.015	0.015
2	0.025	0.2	0.07	0.09	0.04	0.09
3	0.09	0.09	0.1	0.015	0.02	0.09
4	0.07	0.35	0.35	–	–	–

Therefore, to satisfy the twin objective of high success ratio and low perceptibility factor, we select $\mathcal{T}=50$, and $\lambda=0.2$. The same approach is followed to select the optimal ADMM hyperparameters in other experiments. Table V shows the best penalty factor λ for each dataset for every attack level. The number of iterations $\mathcal{T}$ is selected as 15, 50, and 50 for nPath, nTAH, and the average case targeted attacks, respectively.

Targeted attacks results: Here, we present results to show the efficiency of our proposed method for targeted attacks w.r.t. the intensity levels of the proposed HFDD models. Perturbations are generated by solving (5) where we use two criteria to select the target labels [11]. First, the "average" case where the target label t is chosen uniformly at random. Second, the "worst" case where the target label is the one yielding the maximum distance function $D(\boldsymbol{\eta})$ among all labels.

The results are shown in Table VI for ASHARE 1312-RP and 1043-RP. We compare our ADMM solver with the commercial CVX solver. In terms of the average imperceptibility, our proposed ADMM outperforms the CVX solver for all the scenarios except level 2 of ASHRAE 1043-RP experiments. For example, the level 3 ASHRAE 1312-RP average case requires an average imperceptibility of 46.41% for CVX in comparison to 34.07% with the ADMM solver. Furthermore, the required average run-time (t_{avg}) our attacks are significantly smaller than those needed for the CVX solver. For example, the average execution time for targeted attacks (average case) is 0.065 s for the ADMM solver versus 0.235 s for the CVX solver for ASHRAE 1312-RP dataset.

Figs. 6 and 7 show original and perturbed samples using the proposed targeted attacks of level 3 for ASHARE 1312-RP and 1043-RP, respectively, when the target label is selected based on the two aforementioned scenarios. For "average" case, we observe that the perturbed sample (red sold line) is very similar to the original sample (black) with $\rho_2(\%)=33.79\%$ (for 1312-RP) and 33.18% (for 1043-RP). For the worst case attack sample, we observe that the perturbed sample (red dashed line) is different from the original (as also reflected by the high ρ_2 values). The

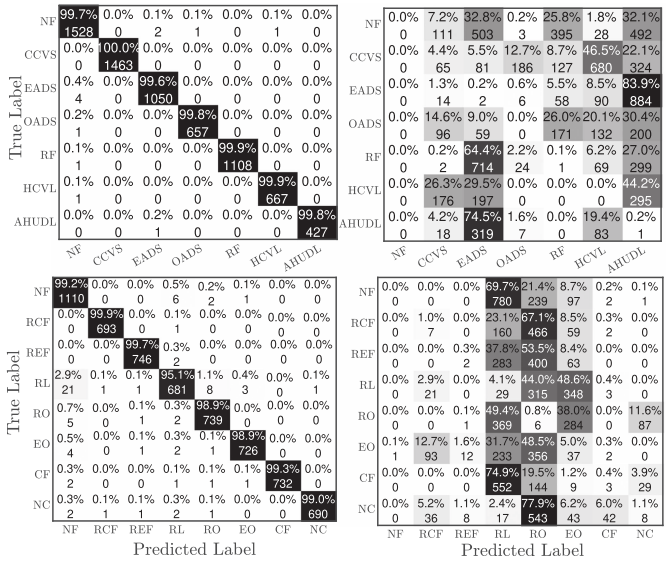

Fig. 5. Confusion matrices before (*right*) and after (*left*) applying the nTAH attack at $l=2$ for AHSRAE: 1312-RP (*top*), and 1043-RP (*bottom*).

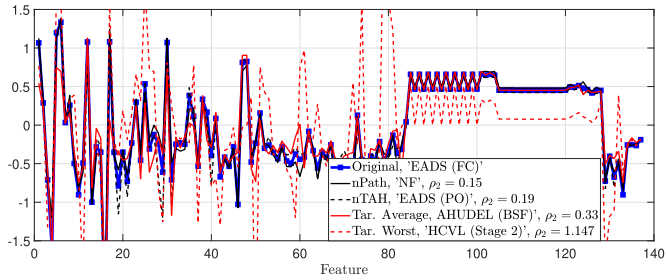

Fig. 6. Sample from the ASHRAE 1312 dataset classified as "EADS (FC)" (black), and its perturbed versions using the nPath (black), nTAH (dotted black), average case targeted (red), and worst case targeted (dotted red) attacks. The predicted labels for the perturbed vectors are different from the original prediction.

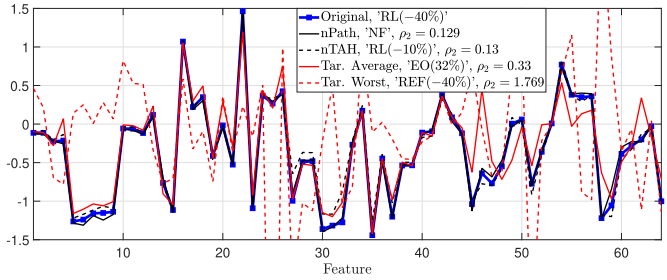

Fig. 7. Sample from the ASHRAE 1043 dataset classified as "RL(–40%)" (black), and its perturbed versions using the nPath (black), nTAH (dotted black), average case targeted (red), and worst case targeted (dotted red) attacks. The predicted labels for the perturbed vectors are different from the original prediction.

high values of imperceptibility is due to selecting the label with the largest distance function ($D(\boldsymbol{\eta})$).

Nontargeted attacks results: For the nontargeted attacks, we present results for the nTAH and nPath methods and compre them in terms of $\zeta(\%)$ and $\sigma_2(\%)$. Table VII shows results for every HFDD level l using our proposed ADMM and the CVX

TABLE VI
RESULTS OF THE TARGETED ATTACKS WITH WORST AND AVERAGE CASE SCENARIOS IN TERMS OF FOOLING RATIO, IMPECEPTIBILITY, AND RUN-TIME USING THE COMMERCIAL CVX SOLVER AND OUR PROPOSED ADMM-BASED SOLVER

Case	Targeted (average)		Targeted (worst)	
	CVX $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	ADMM $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	CVX $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	ADMM $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$
ASHRAE 1312-RP HFDD Level 1	{97.96, 18.24, 0.23}	{97.96, 18.19, 0.090}	{97.96, 18.24, 0.23}	{97.96, 18.19, 0.09}
ASHRAE 1312-RP HFDD Level 2	{93.53, 35.60, 0.26}	{88.44, 32.61, 0.130}	{95.10, 53.64, 1.72}	{86.98, 44.99, 0.08}
ASHRAE 1312-RP HFDD Level 3	{86.99, 46.41, 0.19}	{81.45, 34.07, 0.007}	{80.63, 98.76, 6.01}	{87.04, 50.84, 4.05}
ASHRAE 1312-RP HFDD Level 4	{84.98, 56.97, 0.1297}	{83.09, 45.57, 0.015}	{78.70, 91.86, 2.71}	{79.38, 57.88, 1.98}
ASHRAE 1043-RP HFDD Level 1	{98.53, 11.40, 0.190}	{98.53, 11.40, 0.012}	{98.53, 11.40, 0.19}	{98.53, 11.40, 0.012}
ASHRAE 1043-RP HFDD Level 2	{91.05, 26.22, 0.19}	{84.45,31.91, 0.08}	{89.29, 43.50 1.73}	{94.09, 46.52, 0.830}
ASHRAE 1043-RP HFDD Level 3	{91.93, 43.81, 0.107}	{89.68, 41.08, 0.0085 }	{95.97, 227.00, 7.21}	{94.49, 81.95, 4.43}

The results are presented w.r.t. the level of the considered HFDD structure.

TABLE VII
RESULTS OF THE NONTARGETED ATTACKS IN TERMS OF FOOLING RATIO, IMPECEPTIBILITY, AND RUN-TIME USING THE COMMERCIAL CVX SOLVER AND OUR PROPOSED ADMM-BASED SOLVER FOR THE TWO PROPOSED ALGORITHMS

Case	nPath		nTAH	
	CVX $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	ADMM $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	CVX $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$	ADMM $\{\zeta(\%), \sigma_2(\%), t_{\text{avg}}(\text{sec})\}$
ASHRAE 1312-RP HFDD Level 2	{99.90, 12.35, 0.32071}	{95.47, 10.62, 0.1256}	{99.00, 23.64, 1.72}	{98.08, 25.49, 0.09}
ASHRAE 1312-RP HFDD Level 3	{98.84,12.17, 0.73}	{97.60,11.56, 0.33}	{99.28, 21.85, 6.6}	{98.73, 21.86, 4.20}
ASHRAE 1312-RP HFDD Level 4	{95.01, 12.17, 0.76}	{93.56, 11.44, 0.35}	{99.88, 36.81, 3.21}	{96.61, 34.61, 2.01}
ASHRAE 1043-RP HFDD Level 2	{97.53, 8.53, 0.53}	{96.37, 8.12, 0.23}	{98.51, 11.77, 1.31}	{98.61,19.04, 0.52}
ASHRAE 1043-RP HFDD Level 3	{97.88, 6.88, 0.60}	{97.87, 6.65, 0.24}	{98.8, 7.13, 5.53}	{99.06,16.60, 2.14}

The results are presented w.r.t. the level of the considered HFDD structure.

solvers for the ASHRAE 1312-RP and 1043-RP datasets. We make the following observations.

First, for all the considered scenarios, the nPath method outperforms the nTAH approach in terms of impeceptibility as observed in the smaller $\sigma_2(\%)$ values. For instance, the nPath level 2 attack for 1312-RP requires $\sigma_2(\%)$ of 12.35% versus 23.6% for nTAH. Second, the ADMM and CVX solvers achieve similar values of $\zeta(\%)$ and $\sigma_2(\%)$ for each level, e.g., the pair $\{\zeta(\%), \sigma_2(\%)\}$ of the nTAH level 2 attack is $\{99.00, 23.64\}$ for CVX, and $\{98.08, 25.49\}$ for the ADMM solver (for ASHRAE 1312-RP). When compared to CVX, our ADMM solver requires less average run-time (t_{avg}) to generate perturbations.

Fig. 5 shows the confusion matrices of the true labels (*rows*) and the predicted ones (*columns*) before (*left*) and after (*right*) the nTAH attack on the second HFDD level of the 1312-RP (*top*) and 1043-RP (*bottom*) datasets. We observe that our nTAH attack is successful at altering the prediction of 6843 (6100) out of 6912 (6192) feature vectors for 1312-RP (1043-RP) dataset.

Figs. 6 and 7 show samples of the original and perturbed examples generated from the nTAH and nPath attacks. Both attacks are successful at altering the prediction. We observe that the nTAH attack is successful at changing the prediction from "EADS (FC)" and "RL (–40%)" to "EADS (FO)" and "RL (–10%)," while reporting $\rho_2(\%)$ of 19.5% and 13.3% for the 1312-RP and 1043-RP, respectively.

Comparison to SSC and overall performance: Here, we empirically show that the proposed HFDD model is more robust than the SSC. For the SSC, we train a 16-class (one nonfaulty class and 15 fault intensity classes) and 29-class (one nonfaulty class and 28 fault intensity classes) classifiers with classification accuracy of 99.83% and 98.13% for the ASHRAE 1312-RP and 1043-RP datasets, respectively. The parameter ϵ_s is selected to be 1.2 and 0.5 (0.65 and 0.5) for the targeted and nontargeted attacks, respectively, for ASHARE 1312-RP (1043-RP). To this

TABLE VIII
COMPARISON RESULTS BETWEEN THE PROPOSED HFDD MODEL AND THE SSC IN TERMS OF THE FOOLING RATIO AND IMPERCEPTIBILIY

Dataset	Model	Nontargeted $\{\zeta(\%), \sigma_2(\%)\}$	Targeted $\{\zeta(\%), \sigma_2(\%)\}$
1312-RP	SSC (attack formulation [29])	{99.91,**11.2**}	{84.19,**34.78**}
	HFDD	{99.28,21.85}	{86.99,46.41}
1043-RP	SSC (attack formulation [29])	{98.97,**4.78**}	{88.00,**40.37**}
	HFDD	{98.8, 7.13}	{91.93,43.81}

end, (11) is used to generate targeted and nontargeted disturbances against the SSC. For the targeted case, the target label t is chosen randomly. For the nontargeted case, we replace the target label t with the label $c \in [M] \setminus \{C_p(x)\}$ that achieves minimum perceptibility

$$\begin{aligned} &\min_{\boldsymbol{\eta}} D(\boldsymbol{\eta}) \quad \text{subject to} \\ &\boldsymbol{\eta}^T \left(\nabla_{\mathbf{x}} f_t(\mathbf{x}) - \nabla_{\mathbf{x}} f_{c_p(\mathbf{x})}(\mathbf{x})\right) \geq \\ &f_{c_p(\mathbf{x})}(\mathbf{x}) - f_t(\mathbf{x}) + \epsilon_s \,, \forall p \in [M] \setminus \{t\}. \end{aligned} \tag{11}$$

Table VIII presents the results for the SSC and the HFDD model attacks at level $l = 3$. We observe that at similar success ratio, $\zeta(\%)$, a larger perturbation is required to fool the HFDD model than the SSC (higher σ_2), indicating that it is easier to fool the SSC system when compared to the HFDD model. The reason is that for the HFDD model, in order induce any misclassification; the attacker must change the prediction of multiple local classifiers along the corresponding route of the hierarchical structure. For example, let the target label to be "RL (–10%)" $t = c_{5,1}$ (from Fig. 3). In this case, the perturbed features must be classified as "faulty" ($c_{1,1}$), then "RL" ($c_{2,3}$), and finally "–10%" ($c_{5,1}$). For the SSC, it is only required that the perturbed sample be classified as $c_{5,1}$.

V. Conclusion and Future Work

Existing FDD models have largely focused on SSCs. By contrast, in this article we studied attacks exposing the vulnerability of coarse-to-fine fault detection and multilevel diagnosis systems. To our knowledge, this work is the first to explore adversarial attacks on buildings AFDD models. We formulated convex optimization problems and developed two algorithms to obtain nontargeted and targeted attacks on the HFDD model. Based on experimental results using two real-world datasets of measurements from AHUs and chillers, we illustrated the efficiency of the proposed methods in terms of both the misclassification rate and the imperceptibility of the attack. Our ADMM-based solver was shown to outperform the state-of-the-art commercial convex solver. Our results have shown that the HFDD is more robust to additive disturbances than SSC-based FDD models since inducing misclassifications requires fooling multiple levels in the hierarchy.

As future directions, we plan to investigate attacks on HFDD in black-box settings where access to the classification model parameters is unavailable. In addition, we plan to extend the proposed attack formulations to complex dynamical networks settings with missing measurements and different environments, as done in the work presented in [30]. Furthermore, we aim to investigate the integration of the proposed methods with various defense approaches, such as the minimax formulation of adversarial training.

References

[1] E. DoE, "Building energy data book. department of energy," *Energy Efficiency Renewable Energy*, 2011.

[2] Y. Li and Z. O'Neill, "A critical review of fault modeling of HVAC systems in buildings," in *Building Simulation*, vol. 11, no. 5. New York City, NY USA: Springer, 2018, pp. 953–975.

[3] A. Behfar, D. Yuill, and Y. Yu, "Automated fault detection and diagnosis methods for supermarket equipment (RP-1615)," *Sci. Technol. Built Environ.*, vol. 23, no. 8, pp. 1253–1266, 2017.

[4] G. Lin, H. Kramer, and J. Granderson, "Building fault detection and diagnostics: Achieved savings, and methods to evaluate algorithm performance," *Building Environ.*, vol. 168, 2020, Art. no. 106505.

[5] W. Kim and S. Katipamula, "A review of fault detection and diagnostics methods for building systems," *Sci. Technol. Built Environ.*, vol. 24, no. 1, pp. 3–21, 2018.

[6] D. Li, Y. Zhou, G. Hu, and C. J. Spanos, "Optimal sensor configuration and feature selection for AHU fault detection and diagnosis," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1369–1380, Jun. 2017.

[7] D. Li, Y. Zhou, G. Hu, and C. J. Spanos, "Handling incomplete sensor measurements in fault detection and diagnosis for building HVAC systems," *IEEE Trans. Automat. Sci. Eng.*, vol. 17, no. 2, pp. 833–846, Apr. 2020.

[8] D. Li, Y. Zhou, G. Hu, and C. J. Spanos, "Identifying unseen faults for smart buildings by incorporating expert knowledge with data," *IEEE Trans. Automat. Sci. Eng.*, vol. 16, no. 3, pp. 1412–1425, Jul. 2019.

[9] J. Ploennigs, M. Maghella, A. Schumann, and B. Chen, "Semantic diagnosis approach for buildings," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3399–3410, Dec. 2017.

[10] P. Ciholas, A. Lennie, P. Sadigova, and J. M. Such, "The security of smart buildings: A systematic literature review," 2019, *arXiv:1901.05837*.

[11] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346–360, 2020.

[12] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *J. Mach. Learn. Res.*, vol. 17, no. 83, pp. 1–5, 2016.

[13] H. Shao, H. Jiang, X. Zhang, and M. Niu, "Rolling bearing fault diagnosis using an optimization deep belief network," *Meas. Sci. Technol.*, vol. 26, no. 11, 2015, Art. no. 115002.

[14] X. Dai and Z. Gao, "From model, signal to knowledge: A data-driven perspective of fault detection and diagnosis," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2226–2238, Nov. 2013.

[15] P. Morgner, S. Mattejat, and Z. Benenson, "All your bulbs are belong to us: Investigating the current state of security in connected lighting systems," 2016, *arXiv:1608.03732*.

[16] Z. Chen et al., "Construction of a hierarchical feature enhancement network and its application in fault recognition," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4827–4836, Jul. 2021.

[17] M. Gan et al., "Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings," *Mech. Syst. Signal Process.*, vol. 72, pp. 92–104, 2016.

[18] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, "Manipulating machine learning: Poisoning attacks and countermeasures for regression learning," in *Proc. IEEE Symp. Secur. Privacy*, 2018, pp. 19–35.

[19] X. Liu, Z. Bao, D. Lu, and Z. Li, "Modeling of local false data injection attacks with reduced network information," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1686–1696, Jul. 2015.

[20] I. R. Alkhouri and G. K. Atia, "Adversarial attacks on coarse-to-fine classifiers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 2855–2859.

[21] A. I. Awad, I. R. Alkhouri, and G. Atia, "Adversarial attacks on multi-level fault detection and diagnosis systems," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process., IEEE Signal Process. Soc.*, 2021, pp. 1–6.

[22] E. R. Balda, A. Behboodi, and R. Mathar, "On generation of adversarial examples using convex programming," in *Proc. IEEE 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 60–65.

[23] S. Boyd et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[24] B. He and X. Yuan, "On the $O(1/n)$ convergence rate of the Douglas–Rachford alternating direction method," *SIAM J. Numer. Anal.*, vol. 50, no. 2, pp. 700–709, 2012.

[25] J. Wen and S. Li, "Tools for evaluating fault detection and diagnostic methods for air-handling units," *ASHRAE 1312-RP*, 2011.

[26] M. C. Comstock, J. E. Braun, and R. Bernhard, *Development of Analysis Tools for the Evaluation of Fault Detection and Diagnostics in Chillers*. West Lafayette, IN, USA: Purdue Univ., 1999.

[27] M. C. Comstock, J. E. Braun, and E. A. Groll, "A survey of common faults for chillers/discussion," *Ashrae Trans.*, vol. 108, p. 819–825, 2002.

[28] A. Glass, P. Gruber, M. Roos, and J. Todtli, "Qualitative model-based fault detection in air-handling units," *IEEE Control Syst. Mag.*, vol. 15, no. 4, pp. 11–22, Aug. 1995.

[29] E. R. Balda, A. Behboodi, and R. Mathar, "Perturbation analysis of learning algorithms: Generation of adversarial examples from classification to regression," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 6078–6091, Dec. 2019.

[30] L. Wang, E. Tian, C. Wang, and S. Liu, "Secure estimation against malicious attacks for lithium-ion batteries under cloud environments," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 69, no. 10, pp. 4237–4247, Oct. 2022.

Ismail R. Alkhouri (Student Member, IEEE) received the B.S. degree in electronics and communications engineering from the University of Baghdad, Baghdad, Iraq, in 2009. He received the Ph.D. degree in electrical and computer engineering from the Data Science and Machine Learning Laboratory, ECE Department, University of Central Florida, Orlando, FL, USA, in May 2023.

He is an incoming Postdoctoral Research Associate with Michigan State University, East Lansing, MI, USA, and a Visiting Scholar with the University of Michigan, Ann Arbor, MI, USA, from July 2023. From 2019 to 2022, he was a research intern with the Information Directorate, AFRL. His research interests include adversarial machine learning, digital signal processing, reinforcement learning, and combinatorial optimization.

Akram S. Awad (Student Member, IEEE) received the B.S. degree in electrical engineering from the Jordan University of Science and Technology, Ar Ramtha, Jordan, in 2016. He received the M.S. degree in electrical and computer engineering from Oakland University, Rochester, MI, USA, in 2020. He is currently working toward the Ph.D. degree in electrical and computer engineering from the University of Central Florida, Orlando, FL, USA.

His research interests include robust learning, domain adaptation, and adversarial attacks.

Qun Z. Sun (Member, IEEE) received the Ph.D. degree in electrical engineering from Iowa State University, Ames, IA, USA, in 2011.

She is an Assistant Professor with the University of Central Florida (UCF), Orlando, FL, USA. She is the Director of UCF Smart Infrastructure Data Analytics Lab. Before joining UCF, she worked for Genscape and GE Grid Solutions as a Power System Engineer. Her research interests include grid-edge resources, including smart buildings, rooftop PVs, and batteries, and their interactions with the grid. She is dedicated to the research that improves energy sustainability, resiliency, and security. Her research leverages advanced data analytics and probabilistic algorithms to enhance energy efficiency and security.

George K. Atia (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Boston University, Boston, MA, USA, in 2009.

He is an Associate Professor with the Department of Electrical and Computer Engineering with a joint appointment in the Department of Computer Science, University of Central Florida, Orlando, FL, USA, where he directs the Data Science and Machine Learning Laboratory. From 2019 to 2020, he was a Visiting Faculty with AFRL. From 2009 to 2012, he was a Postdoctoral Research Associate with the University of Illinois at Urbana-Champaign, Champaign, IL, USA. His research interests include robust and scalable machine learning, statistical inference, and verifiable and explainable AI.

Dr. Atia was a recipient of many awards, including the Distinguished Paper Award at AAAI 2023, UCF Reach for the Stars Award in 2018, Inaugural UCF Luminary Award in 2017, NSF CAREER Award in 2016, and Charles Millican Faculty Fellowship Award (2015–2017). He is an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING.