The Adaptive Arms Race: Redefining Robustness in AI Security

Ilias Tsingenopoulos
KU Leuven
Leuven, Belgium
ilias.tsingenopoulos@kuleuven.be

Vera Rimmer
KU Leuven
Leuven, Belgium
vera.rimmer@kuleuven.be

Davy Preuveneers
KU Leuven
Leuven, Belgium
davy.preuveneers@kuleuven.be

Fabio Pierazzi
University College London
London, United Kingdom
f.pierazzi@ucl.ac.uk

Lorenzo Cavallaro
University College London
London, United Kingdom
l.cavallaro@ucl.ac.uk

Wouter Joosen
KU Leuven
Leuven, Belgium
wouter.joosen@kuleuven.be

***Abstract*—Despite considerable efforts on making them robust, real-world AI-based systems remain vulnerable to decision based attacks, as definitive proofs of their operational robustness have so far proven intractable. Canonical robustness evaluation relies on adaptive attacks, which leverage complete knowledge of the defense and are tailored to bypass it. This work broadens the notion of adaptivity, which we employ to enhance both attacks and defenses, showing how they can benefit from mutual learning through interaction. We introduce a framework for adaptively optimizing black-box attacks and defenses under the competitive game they form. To assess robustness reliably, it is essential to evaluate against realistic and worst-case attacks. We thus enhance attacks and their evasive arsenal *together* using reinforcement learning (RL), apply the same principle to defenses, and evaluate them first independently and then jointly under a multi-agent perspective.**

We find that active defenses, those that dynamically control system responses, are an essential complement to model hardening against decision-based attacks; that these defenses can be circumvented by adaptive attacks, something that elicits defenses being adaptive too. Our findings, supported by an extensive theoretical and empirical investigation, confirm that adaptive adversaries pose a serious threat to black-box AI-based systems, rekindling the proverbial arms race. Notably, our approach outperforms the state-of-the-art black-box attacks *and* defenses, while bringing them together to render effective insights into the robustness of real-world deployed ML-based systems.

I. Introduction

AI models are predominantly trained, validated, and deployed with little regard to their correct functioning under adversarial activity, often leaving safety and security considerations as an afterthought. Adversarial contexts further aggravate the typical generalization challenges that these models face with threats beyond model evasion (misclassification), like model extraction, model inversion, and model poisoning [1]. At the same time, the systems these models are components of often expose interfaces that can be queried and used as adversarial "instructors", like in constructing adversarial malware against existing AI-based malware detection [2], [3]. Focusing on adversarial examples for model evasion, the most reliable mitigation to date is adversarial training (AT) [4], [5], an approach not without limitations as these models often remain irreducibly vulnerable at deployment, particularly against black-box, decision-based attacks [6], [7], [8]. Nevertheless, all such attacks exhibit a behavior at-the-interface that can be described as adversarial itself, a generalization that subsumes adversarial examples and opens a path towards novel defenses and mitigations.

Adversarial behavior is a temporal extension of adversarial examples, perhaps not malicious or harmful in isolation, yet part of an attack as it unfolds over time; it is also the canonical description of adversarial examples in domains like dynamic malware analysis and adversarial RL [9], [10]. Aside from making the underlying models more robust, this behavior can be countered as such, rather than relying on hardened models exclusively. As AI models cannot update their decision boundary in an online manner and in response to adversarial activity on their interface, there *has* to be a complement to model hardening: for instance *active* defenses such as rejection or misdirection [11], [12], [13].

In this study we identify and address a crucial gap: evaluating the robustness of defenses against oblivious, non-adaptive, and therefore suboptimal attackers, renders any results unreliable [14], [15]. The key observation we make is that robustness *must* account for the ability of the adversary to adapt while interacting with the model. To that end, we expand the conventional notion of adaptive, from *adapted* attacks that have an empirical configuration to bypass the defense, to include the capability to *self-adapt*, where attacks adapt their parameters and evasive actions *together*, based on how the target model and its defenses respond [16]. We demonstrate theoretically and empirically how self-adaptive attacks can use RL to modify their policies to become both optimal *and* evade active detection. Notably, this can be performed in a gradient-based manner even in fully black-box contexts [III.3], and is a capability that *properly reflects* the level of adversarial threat and in that way does not overestimate the empirical robustness; real attackers will compute gradients after all [17].

Through proper threat modeling and self-adaptation, attacks can reach their full potential, enabling the development of effective defensive policies. To frame the need for adaptive evaluations in adversarial machine learning (AML) differently: a defense can be considered trustworthy only if it is evaluated against an optimal adversary. This mutual interdependence underscores the necessity for *both* attacks and defenses to be self-adaptive, thereby establishing the competitive, zero-sum dynamic inherent in their interaction. In this work, we examine robustness from both perspectives: first, how to fully optimize decision-based attacks, and second, how to devise reliable countermeasures. We explore both offensive and defensive strategies in depth, and make the following key contributions:

1. We demonstrate that active defenses against decision-based attacks are a *necessary* but *insufficient* complement to model hardening. Active defenses are inevitably bypassed by self-adaptive attackers however, and necessitate *self-adaptive* defenses too.

2. To facilitate reasoning on adaptive attacks and defenses, we introduce a unified framework called "Adversarial Markov Games" (**AMG**). We demonstrate how adversaries can optimize their attack policy and evade active detection *at the same time*; as a counter, we develop a novel active defense and employ RL agents to *adapt* and optimize both.

3. In an extensive empirical evaluation and across a wide set of adversarial scenarios, we validate our theoretical analysis and show that self-adaptation with RL *outperforms* vanilla black-box attacks, model hardening defenses like AT, and notably **both** the state-of-the-art adaptive attack (OARS [18]) and stateful defense (Blacklight [19]). This supports self-adaptation as an *essential* component when evaluating robustness to black-box attacks.

4. For reproducibility, and to facilitate further research, we open-source our code[1].

Our work highlights that in the domain of black-box AML, robust evaluations *should* go a step further than adapting attacks: both attacks and defenses should have the capability to optimize their strategies through interaction and in direct response to other agency in their environment. The remainder of the paper is structured as follows: Section II provides the necessary background on the domain and reviews the related work. Section III introduces and motivates our theoretical analysis of robustness under decision-based attacks. Section IV explains the threat model and the concrete design choices. In Section V we elaborate on our experimentation and analyze our results. We conclude with Section VI where we discuss key insights, limitations and challenges.

II. Preliminaries

In this work, we focus on the category of adversarial attacks known as **decision-based**, a subset of query-based attacks that operate solely on the **hard-label** outputs of the model and are a highly realistic and pervasive threat in AI-based cybersecurity environments. Despite the lack of the closed-form expression of the model under attack, given enough queries their effectiveness can match the one of white-box techniques [20], [15].

A. Attacks & Mitigations

While adversarial attacks have been extensively researched in both white and black-box contexts, defenses have predominantly focused on white-box [4], [5]. As the black-box setting discloses considerably less information, a seemingly intuitive conclusion is that white-box defenses should suffice for the black-box case too. Yet black-box attacks like [6], [7] have shown to be highly effective against a wide range of defenses like *gradient masking* [21], *preprocessing* [22], [23], and *AT* [4]. The vast majority of adversarial defenses provide either limited robustness or are eventually evaded by adapted attacks [14]. Characteristically, preprocessing defenses are identified and bypassed by expending queries for reconnaissance [24].

The partial exception to this rule is AT. Given dataset $D = (x_i, y_i)_{i=1}^{n}$ with classes C where $x_i \in \mathbb{R}^d$ is a clean example and $y_i \in 1, ..., C$ is the associated label, the objective of AT is to solve the following *min-max* optimization problem:

$$\min_{\phi} \mathbb{E}_{i\sim D} \max_{\|\delta_i\|_{L_p} \leq \epsilon} \mathcal{L}(h_\phi(x_i + \delta_i), y_i) \tag{1}$$

where $x_i + \delta_i$ is an adversarial example of x_i, $h_\phi : \mathbb{R} \to \mathbb{R}^{\mathbb{C}}$ is a hypothesis function and $\mathcal{L}(h_\phi(x_i + \delta_i), y_i)$ is the loss function for the adversarial example $x_i + \delta_i$. The inner maximization loop finds an adversarial example of x_i with label y_i for a given L_p-norm (with $L_p \in \{0, 1, 2, \inf\}$), such that $\|\delta_i\|_l \leq \epsilon$ and $h_\phi(x_i + \delta_i) \neq y_i$. The outer loop is the ordinary minimization task, typically solved with stochastic gradient descent. While the convergence and robustness properties of AT have been investigated through the computation of the saddle point and by interleaving normal and AT [5], the min-max principle is conspicuous: minimize the possible loss for a worst-case (max) scenario.

B. Stateful Defenses

Decision-based attacks possess properties that can be valuable for devising defenses against them, *in addition* to AT. One such property is their intrinsic sequentiality: by following a policy toward the optimal adversarial example, the generated candidates are correlated. Note that this might not hold for the queries *themselves*, as the adversary may apply transformations that the model is invariant to, such as the query blinding strategy in Chen et al. [13]. This work is the first to employ a *stateful* defense against query-based attacks. Another stateful defense is PRADA [25], devised against model extraction but effective against evasion too. These approaches assume however that queries can be consistently linked (via metadata like IP or account, cf. Table I) to uniquely identifiable actors – who also exhibit limited to no collaboration – so that a query buffer can be built for each.

This limitation, together with the scalability issues, was recently addressed in the Blacklight defense, by employing

[1] https://anonymous.4open.science/r/AMG-AD16

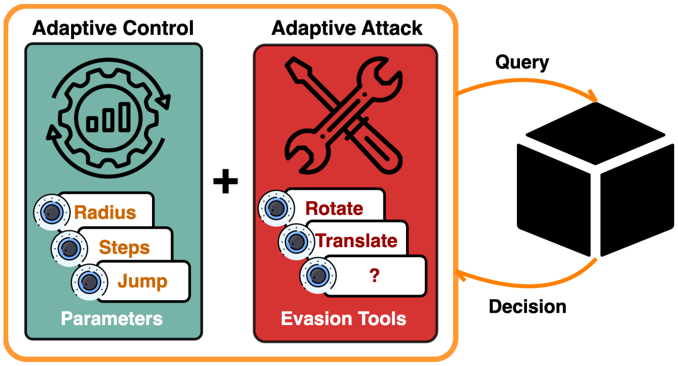

Fig. 1. In AML, adaptive attacks are those with the capabilities (knobs) to bypass a defense; adaptive control is rather the precise tuning of all the known knobs. Against black-box systems, we can reformulate adaptive so that it signifies **both**. For instance in HSJA [7], radius, steps, and jump are parameters of the attack, while rotate and translate are transformations that can evade a similarity-based defense.

hashing and quantization [19]. Blacklight remains a similarity-based defense, thus vulnerable to circumvention if an adversary can find a query generation policy that preserves the attack functionality while evading detection. OARS achieved this by adapting existing attacks through the rejection signal Blacklight returns [18]. Ultimately, any (stateful) defense has to balance the trade-off between robust and clean accuracy; as we demonstrate in this work, this trade-off can be measured reliably only if the attacker is properly adaptive.

C. On Being Adaptive

The correct way to evaluate any proposed defense is against *adaptive* attacks, that is with explicit knowledge of the concrete mechanisms of a defense [14]. In computer security this is known as the stipulation that security through obscurity does not work, as the robustness of defenses should not rely on keeping their mechanism secret. If model hardening – for instance by AT – is the defensive counterpart to white-box attacks, active defenses like stateful detection are the counterpart to decision-based attacks, and as we will further demonstrate, also the *necessary* complement to hardening a model against them.

At the same time, the level of threat that attacks pose is often unclear or not thoroughly evaluated. Previous work has demonstrated that the loss functions and parameters of attacks are often suboptimal, leading to *underestimating* their performance and thus *overestimating* the claimed degree of robustness [15], [26]. This underestimation is further aggravated in decision-based contexts, where the attacker is largely oblivious of any preprocessing or active defenses the black-box system might have. The true performance of attacks therefore rests on the ability to adapt their operation policy and their evasive capabilities *in tandem*.

In AML, "adaptive" by convention refers to attacks with full knowledge of how a defense works and the tools to bypass it; we denote such attacks as **adapted**. In this work we expand the term to include *adaptive control*, defined as the ability of a system to **self-adapt**: *automatically* reconfigure itself in response to changes in the dynamics of the environment in order to achieve optimal behavior [16]. We use adaptive control in the sense "attack optimization" is used by Pintor et al. [26], but here for black-box systems. What is to be controlled is typically known in advance and well-defined. However, the moment we consider adaptive evaluations, *new* controls are directly implied: in a similarity-based defense for instance, such controls would be transformations to the input that the model is invariant to. To flesh out the twofold meaning of adaptive, one has to *both* invent new knobs [27] – the conventional understanding of adaptive, intractable to automate yet – *and* dynamically control their correct configuration that would lead to the optimal result (self-adaptive). We conceptualize this more general definition of adaptive, essential for having accurate evaluations against decision-based attacks, in Figure 1.

D. Research Gap

Prior work has focused on *adapted* attacks, which incorporate general knowledge of any defenses, then empirically configured to evade it [20], [7], [6]. Defenses also follow the same adapted paradigm of empirically defined and fixed operation [13], [19]. Our observation is that neither of them are formalized or performed in a fully adaptive manner, that is in response to how they influence their environment and with respect to other adaptive agents in it, with clear limitations when the latter is a given, e.g. in cybersecurity. To bridge this gap, we provide a theoretical analysis and an empirical study of existing and novel methodologies adapting through direct interaction with their environment, denoting them as **self-adaptive**.

Our work builds on a long line of prior research that focuses on both sides of the competition between adversaries and defenses. **Carlini and Wagner** [20] show that evaluating existing attacks out-of-the-box is insufficient and that adapted white-box attackers can break defensive distillation. **Bose et al.** [29] propose Adversarial Examples Games, a zero-sum game between a white-box attacker and a local surrogate of the target model family. At the equilibrium the attacker can generate adversarial examples that have a high success rate against models from the same family, constituting a zero-query, non-interactive approach for generating transferable adversarial examples. **Pal et al.** [30] propose a game-theoretic framework for studying white-box attacks and defenses that occur in equilibrium. **Feng et al.** [18] introduce **OARS**: adaptive versions of existing attacks that bypass **Blacklight** [19], the state-of-the-art stateful defense. To function, OARS presupposes the rejection signal that a defense like Blacklight returns; a strong assumption that as we show in this work does not have to hold for stateful defenses. As we demonstrate in section V and Table III, Blacklight can be bypassed without assuming rejection, while the novel stateful defense we introduce can fully withstand the OARS adaptive attack.

TABLE I

POSITIONING OF OUR WORK RELATIVE TO PROMINENT DECISION-BASED ATTACKS AND DEFENSES AND THEIR INDIVIDUAL PROPERTIES. WHILE PRIOR WORKS FOCUS EXCLUSIVELY ON OFFENSE OR DEFENSE, OURS UNIFIES AND REASONS FROM BOTH PERSPECTIVES.

Work	**Offensive**				**Defensive**			
	Optimized	**Evasive**	**Adaptive**	**¬Rejection**	**Active**	**Adaptive**	**¬Metadata**	**Misdirection**
Boundary (2018) [6]	○	○	○	●	○	○	—	—
BAGS (2018) [28]	○	○	○	●	○	○	—	—
HSJA (2020) [7]	●	○	○	●	○	○	—	—
OARS (2023) [18]	●	●	◐	○	●	○	—	—
Adv. Training (2017) [4]	●	○	○	—	○	○	—	—
Stateful (2020) [13]	●	●	○	○	●	○	○	○
Blacklight (2022) [19]	●	●	○	○	●	○	●	○
Our work	●	●	●	●	●	●	●	●

As the most relevant and representative threat against real-world AI systems, in this work we scope on decision-based, interactive attacks and defenses. We contribute a theoretical *and* practical framework for self-adaptation, under which the full extent of the offensive potential, and thus also the defensive, are properly assessed. Our work can be viewed as the synthesis of adaptive black-box attacks and defenses in a unified framework, and to facilitate comparison to related work, in Table I we situate it according to offensive properties:

- **Optimized** indicates offensive processes that undergo some form of optimization.
- **Evasive** indicates the use of evasive tactics, like input transformations, during querying.
- **Adaptive** denotes attacks that adapt their policy based on how the defense responds.
- **¬Rejection** denotes attacks that function *without* relying on a rejection signal – and defenses that do not return one.

And defensive properties:

- **Active** indicates defenses that actively monitor and respond to queries, and attacks that assume such defenses.
- **Adaptive** denotes defenses that adapt their policy based on how the attack behaves.
- **¬Metadata** denotes defenses that function *without* access to query metadata, e.g. IP address or account.
- **Misdirection** indicates an active defense that instead of rejecting queries, it misdirects attackers (cf. Figure 3).

For the remainder of the paper, the term **"adaptive"** is used interchangeably with **"self-adaptive"**, and we use the term **"adapted"** for what is commonly known as adaptive evaluations in AML.

III. THEORETICAL FRAMEWORK

In this section, we abstract through the individual properties of decision-based attacks and defenses to extract more general insights than a purely empirical study would render. To investigate how robust real-world systems are to evasion, two related perspectives are crucial: a) resisting decision-based attacks, and b) adapting attacks and evasive capabilities *together*. When attacks (and defenses) are evaluated in a non-adaptive manner, in the expansive sense we outlined in subsection II-C, results are unreliable [14], [26]. Note that with offensive or defensive methodologies adapting, their environments become non-stationary [31], putting further pressure on the IID foundations that ML builds on. To understand the implications of this adaptation, we perform an analysis of the possible interactions on the interface of an ML-based system, interactions that can be more generally considered as sequential zero-sum games [32], [33], [29]. In the following sections, introduced terms and notation are highlighted in orange.

A. Attacks

The most compelling threat that deployed ML-based systems face are decision-based black-box attacks, where no access is assumed to the model or its parameters, only the capacity to submit queries and receive hard-label responses. One of the first decision-based attacks was Boundary Attack [6], followed by others that improve the overall performance, typically measured as the lowest perturbation achieved for the minimal amount of queries submitted. Prominent examples are HSJA [7], Guessing Smart (BAGS) [28], Sign-Opt [34], Policy-driven (PDA) [8], QEBA [35], and SurFree [36].

White-box attacks like C&W [20] do not function in black-box environments, as there is no access to the inference pipeline. To facilitate optimization, decision-based attacks commonly initialize from a sample belonging to the target class, as it can be considered an adversarial example with an unacceptably large perturbation. This switch allows the task to be solved continuously, by minimizing the perturbation while always staying on the adversarial side of the boundary. Decision-based attacks share further common aspects in their function that we can abstract through: given **starting** and **original** samples x_g and x_c respectively, the goal is to iteratively propose adversarial **candidates** x_t, until the **distance** $\delta = d(x_t, x_c)$ is minimized. This process follows different algorithmic approaches that represent different geometrical intuitions; we can describe it more generally by means of a candidate generation policy:

$$\pi_\theta^{\mathcal{A}} = P\left(x_t | x_g, x_c, p^{\mathcal{A}}, s^{\mathcal{A}}\right), \tag{2}$$

that given x_g and x_c, with $p^{\mathcal{A}}$ the **parameters** and $s^{\mathcal{A}}$ the **state** of the attack, generates a candidate x_t. As attacks execute over discrete time steps, if we assume that the model always answers the attack procedure can be construed as a Markov Decision Process (MDP) to be solved, by finding the parameters θ that minimize δ for a given number of queries.

Consider now a multinomial classification model $\mathcal{M}$ under attack, with a discriminant function $F : \mathbb{R}^d \rightarrow \mathbb{R}^m$, that for each input $x \in [0,1]^d$ generates an output $y := \{y \in [0,1]^m | \sum_{c=1}^{m} y_c = 1\}$ – a probability distribution over the m classes. As black-box environments prevent access to these probabilities, one can only observe the decision of the classifier C that returns the highest probability class:

$$C(x) := \arg\max_{c\in[m]} F_c(x) = D(F_c(x)) \tag{3}$$

with D being the decision function, here $D = \arg\max$. The goal in targeted attacks is to change the **decision** $c_g \in [m]$ for a correctly classified example x, to a predefined **target** class $c_o \neq c_g$. This process can be facilitated through a function ψ which given a perturbed example x_t at step t, it returns a binary indicator of success:

$$\psi(x_t) = \begin{cases} +1 & \text{if} \quad C(x_t) = c_o \\ -1 & \text{if} \quad C(x_t) \neq c_o \end{cases} \tag{4}$$

As long as the model responds, ψ can always be evaluated, and constitutes the fundamental mechanism upon which decision-based attacks build. The adversarial goal can then be described as the following constrained optimization problem:

$$\min_{x_t} \; d(x_t, x_c) \quad \text{s.t.} \quad \psi(x_t) = 1, \tag{5}$$

where the distance metric d is an ℓ_p-norm, with $p \in \{0, 1, 2, \inf\}$. As the threshold between adversarial and non-adversarial relies on the subjectivity of human perception, this highlights the indefinite nature of adversarial examples, further exemplified in domains where visual proximity is of little importance. Successful or unsuccessful adversarial examples are therefore delimited by an **threshold** ϵ on perturbation, where $d(x, x_t) \leq \epsilon$.

Real-world attacks being black-box does not make them less effective. For instance, HSJA is guaranteed to converge to a stationary point of Eq. (5). Given typical ϵ values for imperceptibility, this results in high attack success rates, even against *adversarially trained* models. The limitations of AT against decision-based attacks can be attributed to the out-of-distribution (OOD) nature of adversarial examples, and the saddle point optimization problem of Eq. (1) that make it difficult for algorithms to converge to a global solution. Furthermore, incorporate decision-based attacks in training is not scalable as it can take orders of magnitude more steps (queries) to produce an adversarial example, than white-box attacks which take a few steps (1-50 in e.g. PGD [4]).

Decision-based attacks search for the **optimal parameters** θ^* of the generation policy (2), those that given x_c^i, with i denoting the i-th adversarial episode, minimize Eq. (5) in expectation:

$$\arg\min_{\theta} \mathbb{E}[\sum_{i=1}^{N} d(x_b^i, x_c^i)], \quad \text{s.t.} \quad \psi(x_b^i) = 1, \tag{6}$$

where x_b^i is the **best** adversarial example generated by policy $\pi_\theta^{\mathcal{A}}$ during episode i. Given the dimensionality of the input, it can be intractable to learn a policy that modifies the feature space directly [37]; CIFAR-10, for instance, has more than 3K features to perturb.

In AI-based systems, the best practice is to freeze the model after validation so that no novel issues are introduced by re-training: for all queries x_t submitted during an attack session, we can therefore assume that $F_0 = F_1 = ... = F_t, \forall t$. While this is representative of real-world settings, it also enables adversaries to discover adversarial examples that were not identified beforehand. Consequently, while model-hardening through AT is *necessary*, it can also be *insufficient* against decision-based attacks like HSJA.

Proposition III.1. *Let F_c denote the discriminant function of an adversarially trained model $\mathcal{M}$, and let $C(x) = D(F_c(x))$ denote its classifier. Then in HSJA, to satisfy $\mathbb{E}[\sum_{i=1}^{N} d(x_b^i, x_c^i)] \geq \epsilon$ it is necessary that: (a) $D \neq \arg\max$, and (b) context τ exists s.t. for some query x_t, $D(F_c(x_t)) \neq D'(\tau, F_c(x_t))$, where D' is a stateful extension of D.*

Intuitively, HSJA operates in 3 stages which repeat: a binary search that puts x_t on the decision boundary, a gradient estimation step, and projection step along the estimated gradient. If the model *always* responds truthfully, the adversary will be able to accurately perform all these steps and converge to the optimal adversarial; without loss of generality, we can extend this intuition to other decision-based attacks which navigate the boundary. Secondly, the model should be able to distinguish between two, otherwise identical, queries, when one is part of an attack and the other is not, a capability achievable through statefulness; see Appendix A for the proof.

B. Defenses

Proposition III.1 suggests that alternative classification policies are necessary in the presence of decision-based attacks, e.g. classification with rejection or intentional misdirection. Rejection has been realized in the form of conformal prediction, where model predictions are sets of classes including the empty one, or learning with rejection [11], [38]; while misdirection has emerged as a technique in adversarial RL and cybersecurity domains [10], [12]. While adversarially training the discriminant function F empirically shows some degree of robustness to decision-based attacks, the manner in which the model responds has a complementary potential. The gap between the empirical and theoretically achievable robustness is the source for an active defense *distinct* from model hardening. Active defenses have direct implications on attacks themselves however. Let us now assume an agent carrying out an **active defense policy**:

$$\pi_\phi^{\mathcal{D}} = P(\alpha_t | x_t, s_t^{\mathcal{D}}),\ \alpha \in \{0, 1\} \tag{7}$$

with x_t the query, $s_t^{\mathcal{D}}$ the **state** for the defense as created by past queries, and α the **binary decision**: for queries deemed adversarial, $\boldsymbol{\alpha} = 1$, otherwise $\boldsymbol{\alpha} = 0$. When this policy is stationary, the environment dynamics become stationary in turn, thus besides the adversarial task itself, bypassing the defense can *also* be formulated as an MDP to be solved (Figure 2). In two-player, zero-sum games, the moment an agent follows a stationary policy, it becomes *exploitable* through the reward obtained by an adversary [39]. Active defenses, as consequence of decision-based attacks, entail therefore *adaptive* adversaries.

Proposition III.2. *Against an active defense $\pi_\phi^{\mathcal{D}}$ and for time horizon T, a decision-based attack following a non-adaptive candidate generation policy $\pi_t = \pi_\theta^{\mathcal{A}}, \forall\, t \in [0, T]$ will perform worse in expectation (6), that is $\mathbb{E}[\sum_{i=1}^{N} d(x_b^i, x_c^i)]^{\mathcal{D}} > \mathbb{E}[\sum_{i=1}^{N} d(x_b^i, x_c^i)]^{\not\mathcal{D}}$.*

A proof for BAGS and HSJA is included in Appendix A. An adversary can reason, as a corollary to Proposition III.1, that such defenses *have to* be in place as it is suboptimal not too. However, there is a second reason to consider adaptive attacks even in the absence of active defenses, as attack policies are often suboptimal with their default, empirically defined parameters. Adapting attack policies is essentially the optimization of these parameters, and as an approach has proven very effective in other black-box or expensive-to-evaluate domains, like Neural Architecture Search and Data Augmentation [40], [41], [42]. Our results in section V further indicate the correspondence between adaptive and self-optimizing, showing that adaptive consistently outperform non-adaptive attacks, particularly against active defenses.

Consider now an active defense that is based on a similarity or conformal metric. In the twofold meaning we introduced in subsection II-C, adaptive attack implies the *capability* to bypass a similarity based defense; adaptive control implies optimization instead, the active tuning of all the available tools to evade the defense *and* minimize the perturbation (cf. Figure 1). The updated adversarial objective then is to find the optimal policy that *also* evades detection, and the way to achieve this is by adapting the candidate generation policy (2) itself. Notably, and despite the black-box and discontinuous nature of the task, this optimization can be *fully* gradient-based. Decision-based attacks can recover **gradient-based** solutions to their objective, despite *neither* the active defense *nor* the model itself being accessible in closed-form. For model $\mathcal{M}$, adversarial queries x_t, and active defense $\pi_\phi^{\mathcal{D}}$ making decisions α_t, we can thus formulate the following:

Proposition III.3 (Adversarial Policy Gradient)**.** *Given adversarial policy $\pi_\theta^{\mathcal{A}}$ (2) that generates episodes τ_i of queries x_t, and reward function $r(\tau_i) = \sum_{x_t \in \tau_i} (1 - \alpha_t)$, the optimal evasive policy $\pi_{\theta^*}^{\mathcal{A}}$ is obtained by gradient ascent on the policy's expected reward, $\nabla_\theta \mathbb{E}_{\pi_\theta^{\mathcal{A}}}[r(\tau_i)]$.*

The proof is included in Appendix A. We thus have established that, **a)** in the presence of decision-based attacks, active defenses are necessary, yet conditional on adversarial agency they are insufficient and, **b)** adaptive attacks can become optimal in terms of both evasion and efficiency by observing and adapting to the discrete model decisions. To complete the puzzle, the last piece is turning active defenses also adaptive.

Corollary III.4. *The active defense achieves its optimal $\pi_{\phi^*}^{\mathcal{D}}$ (7), i.e. maximizing expectation $\mathbb{E}[\sum_{x_t \in \tau_i} P(\alpha_t | x_t, s_t^{\mathcal{D}})]$, by adapting its policy against the optimal evasive policy $\pi_{\theta^*}^{\mathcal{A}}$.*

Proof. Since the game is zero-sum, we may define the defensive policy reward ρ on any trajectory $\tau_i = (x_1, \ldots, x_T)$ as $\rho(\tau_i) = \sum_{x_t \in \tau_i} \alpha_t$. Treating the adversary's policy $\pi_{\theta^*}^{\mathcal{A}}$ as fixed, we perform gradient ascent on the expected reward $J(\phi) = \mathbb{E}_{\tau_i \sim (\pi_{\theta^*}^{\mathcal{A}}, \pi_\phi^{\mathcal{D}})}\big[\rho(\tau_i)\big]$. Under standard smoothness assumptions, this converges to $\phi^* = \arg\max_\phi J(\phi)$, which is precisely the defender's best response to $\pi_{\theta^*}^{\mathcal{A}}$. □

C. Adversarial Markov Games

By reasoning on both offensive and defensive capabilities, we highlight why one cannot consider them independently. As adaptive attacks and defenses are logical consequences of each other, their composition forms a turn-taking competitive game. A precise game-theoretic formulation requires full knowledge of the environment: its models, players and their utility functions, as well as the permitted interactions and the transition dynamics, something typically intractable in this and other cybersecurity settings. Model-free methods however can learn optimal (offensive *and* defensive) responses directly through interaction with their environment [12], [43], avoiding the need for explicit modeling or solving the NP-hard bi-level optimization problem of Eq. (1) [44].

To that end, Turn-Taking Partially-Observable Markov Games (TT-POMGs), introduced by Greenwald et al. [45], generalize Extensive-Form Games (EFGs) that model non-cooperative, sequential decision-making games of imperfect and/or incomplete information. TT-POMG is a suitable formalism for decision-based attacks and defenses, with the added benefit that it can be transformed into an equivalent belief state MDP, significantly simplifying its solution.

Prior work has explored the competition underlying adversarial example generation in no-box and white-box settings [29], [46]. We instead focus on decision-based, interactive environments, with unknown but stationary dynamics: all other agents present are considered part of the environment and therefore fixed in their behavior. By folding the strategies of other agents into the transition probabilities and the initial probability distribution of the game, an optimal policy computed in the resulting MDP will correspond to the best-response strategy in the original TT-POMG. The congruence

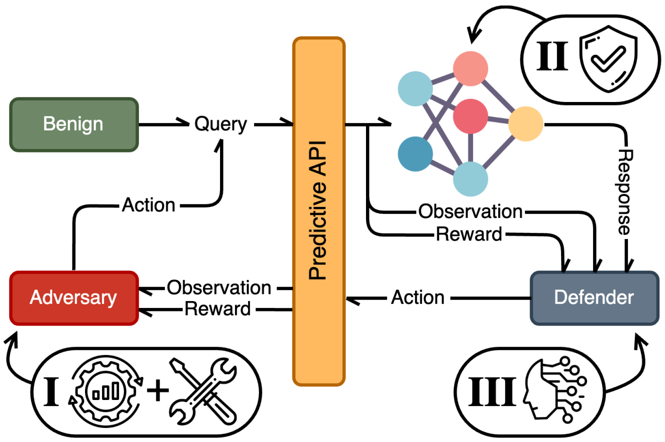

Fig. 2. Schematic model of an AMG environment. Due to the inherent uncertainty of behavior at either side of the interface, it is a partially observable MDP, mirrored for each agent where one's decisions become the other's observations. (I) denotes an adaptive attacker (cf. Fig. 1), (II) model hardening (passive defense), and (III) an active defense.

between TT-POMGs and MDPs has both theoretical *and* practical value for securing AI-based systems: once adversarial agents and their capabilities are identified through rigorous threat modeling, the best-response strategy in the simulated environment yields the optimal defense.

We describe the environment that encompasses adversarial attacks, adversarial defenses, and benign queries, as an Adversarial Markov Game (AMG) – a special case of TT-POMG – depicted in Figure 2. Formally, we represent AMG as a tuple $\langle i, S, O, A, \tau, r, \gamma \rangle$

- $i = \{\mathcal{D}, \mathcal{A}\}$ are the players, where $\mathcal{D}$ denotes the defender and $\mathcal{A}$ denotes the adversary. In our model, benign queries are modeled as moves by nature.
- S is the full state space of the game, while $O = \{O^{\mathcal{D}}, O^{\mathcal{A}}\}$ are partial observations of the full state for each player.
- $A = \{A^{\mathcal{D}}, A^{\mathcal{A}}\}$ denotes the action set of each player.
- $\tau(s, a^i, s')$ represents the transition probability to state $s' \in S$ after player i chooses action a^i.
- $r = \{r^{\mathcal{D}}, r^{\mathcal{A}}\} : O^i \times A^i \rightarrow \mathbb{R}$ is the reward function where $r^i(s, a^i)$ is the reward of player i if in state s action a^i is chosen.
- $\gamma^i \in [0, 1)$ is the discount factor for player i.

The goal of each player i is to determine a policy $\pi^i(A^i|O^i)$ that, given the policy of the other(s), maximizes their expected reward. When a player employs a stationary policy, the AMG reduces to a belief-state MDP where the other interacts with a fixed environment. The game is sequential and turn-taking, so each player i chooses an action a from their set of actions A^i which subsequently influences the observations of others.

We have shown that an adaptive defense policy $\pi_\phi^{\mathcal{D}}$ is necessary to deter decision-based attacks, and that consequently the candidate generation policy $\pi_\theta^{\mathcal{A}}$ has to be also be adaptive. As with plausible assumptions we cannot assume access to the exact state of the other agent, the states $O^{\mathcal{D}}$, $O^{\mathcal{A}}$ are partial observations of the complete state $\mathcal{S}$ of the full game. For instance, when the competing agents (holding beliefs about each other) are human, they engage in recursive reasoning expressed as [I believe that [my opponent believes [that I believe...]]]. In the study of opponent modeling, considering other agent policies as a stationary part of the environment is equivalent to *0th* level recursive reasoning: the agent models how the opponent behaves based on the observed history, but *not* how the opponent *would* act based on how the agent behaves [47], [48]. In this work we consider more involved recursive reasoning out of scope, as AMGs can be solved by single-agent RL algorithms, and perform the empirical evaluation without building explicit models of opponent behavior.

IV. Threat Model

The empirical study we conduct in Section V reflects diverse instantiations of the general theoretical framework introduced in Section III. When working forward from the theoretical to the practical, concrete design choices have to be made when specifying the latter, choices that can have considerable influence on the results. To elucidate our proposed robustness evaluation methodology, in this section we provide the concrete details on the threat model and the environment.

[Threat Model]. Our AMG framework describes a two-player competitive game; while extensible to more players, in this work we assume that at a given moment only one attack takes place. From the defensive perspective, incoming queries can be either benign or part of an attack. An assumption that influences the effectiveness of stateful detection is that queries can be attributed to UIDs, e.g., an IP address or a user account. However, adversaries can collude, create multiple accounts, use VPNs, or in fact accounts and IP addresses might not even be necessary to query the model. To address this, we treat queries irrespective to their source. This is a strictly more challenging setting for stateful defenses, where we operate solely on the content of queries and not on any other metadata, similar to [19]. Unlike Blacklight however, instead of rejecting queries, something that in itself provides *more* information to the adversary and thus facilitates evasion (cf. OARS [18], Table I), we misdirect by returning the second highest probability class. Furthermore, Gaussian noise is added to the benign queries to simulate a noisy channel and a shift in distribution, so that is not trivial for a defense to tell adversarial noise apart. In summary, the black-box threat model we consider is delineated as follows:

- **Assets:** Trained and deployed model $\mathcal{M}$ with corresponding weights w.
- **Agents:** Adversary / Defender / Benign user.
- **Adversary Goal:** Generate minimal perturbation adversarial examples in as few queries as possible, while evading the defense.
- **Defender Goal:** Stop the adversary from generating adversarial examples, while preserving the correct functionality of the model $\mathcal{M}$ on benign users.
- **Adversary Knowledge:** The model $\mathcal{M}$ is known as the black-box function that transforms inputs $x \in [0, 1]^d$ to outputs

$c \in [m]$, m being the number of classes. The weights w and the closed-form expression of $\mathcal{M}$ are unknown, as unknown is if an active defense $\pi_\phi^{\mathcal{D}}$ is present or not.

- **Defender Knowledge:** The defender observes only the content of incoming queries, without knowing if they come from a benign user or the adversary.
- **Adversary Capabilities:** Adapt the parameters of the attack and of any evasive transformations; in essence, optimize the candidate generation policy $\pi_\theta^{\mathcal{A}}$.
- **Defender Capabilities:** For each query x, decide between answering truthfully with the actual prediction $C(x) = c_T$, or misdirect with the second highest probability class c_S.

[Similarity]. Decision-based attacks typically follow a policy that generates successive queries: these exhibit degrees of similarity which can be quantified by an appropriate L_p norm. If that norm is computed on the original inputs however, an adversary can adapt by employing evasive transformations the model is invariant to and bypass the similarity detection (cf. Figure 1). To account for this capability, we train a Siamese network with contrastive loss in order to learn a latent space $\mathcal{L}(\cdot)$ where similar inputs are mapped close together, unaffected by added noise or transformations on the inputs. For the stateful characterization of queries, we use two queues: one for the detected adversarial queries as determined by the defensive agent, and one for the benign and undetected ones.

[Active defense]. Recall that decision-based attacks evaluate a Boolean-valued function to determine if the query is adversarial or not; a straightforward counter to this behavior is to misdirect by returning a decision different from the actual through a system of confinement. When new query x_t is received, a state is constructed based on x_t and the queue $k_{-n}, k_{-n-1}, ..., k_0$ of known adversarial queries. Based on this state, the defensive agent takes a single continuous action $\{\sigma \in \mathbb{R} \,|\, 0 \leq \sigma \leq 1\}$, with σ being the radius of a hypersphere centered on the last known adversarial query k_0 in the latent space $\mathcal{L}$. If $\|\mathcal{L}(x_t) - \mathcal{L}(k_0)\|_2 < \sigma$ the query is considered adversarial and is appended to the adversarial queue as the latest k_0. This system of confinement is depicted in Figure 3.

[Adaptivity]. No evaluation in AML is complete without considering adaptive adversaries; a notion we expand in this work, that is with the instruments to bypass the defense *and* their optimal configuration. As stateful defenses are so far similarity based, to bypass them intuition points towards input transformations the model is invariant to. For a given query x_t we want to compute a transformation $x'_t = T(x_t)$ so that $||x'_t - x_t||_2 \gg ||x_t - x_{t-1}||_2$ while $F(T(x_t)) \approx F(x_t)$. Depending on magnitude and composition of transformations T, the identity $F(T(x_t)) = F(x_t)$ might not always hold. As we also demonstrate in Section V, T interferes with the perturbations of the adversarial policy: the performance and evasiveness of an attack are thus in a natural trade-off.

At this point one should inquire what is the correct composition of transformations T to apply. When shall T be applied, and how does it affect the attack fundamentals? The transformations T can be considered as a set of additional controls, and like attack parameters they themselves can be

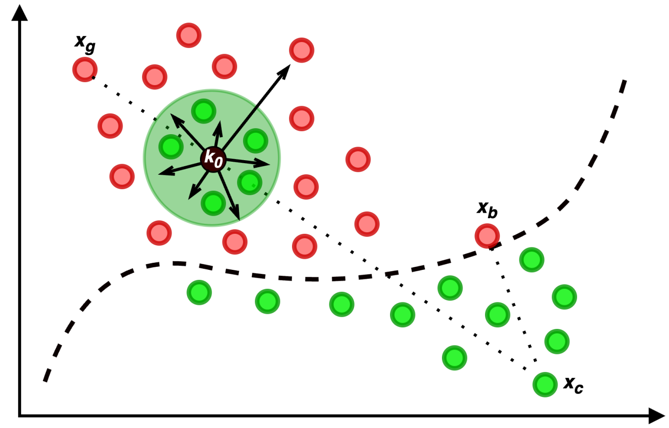

Fig. 3. Misdirection in a hypothetical 2D decision boundary. The adaptive defense controls a single parameter, the hypersphere radius around k_0 (the last known adversarial); for queries x_t that fall within this hypersphere the model responds with a non-adversarial decision. x_g is the starting sample, x_c the original, and x_b the best possible adversarial.

suboptimal out-of-the-box [15]. Thus the combined control of attack and evasion parameters is a *prerequisite* to properly assess the strength of a defense. Their trade-off illustrates why the twofold definition of adaptive is necessary in AML evaluations: first to impart the tools to accomplish to the task through the definition of *what* can be controlled, and then to find the precise optimal configuration and strategy of the attack.

[Agents & Environments]. Unlike common competitive games, in AMGs the two players have different action and state sets. AMGs are also asymmetric in the playing cadence: while the defender plays every round, the adversary might wait one to several rounds; HSJA for example is controlled on the iteration rather on the query level. Training is complicated further given that the experience upon which each agent learns arrives only *after* the opponent moves. We address these challenges by developing custom learning environments (with the OpenAI Gym and Stable-Baselines3 libraries) for asymmetrical agents, with delayed experience collection, and asynchronous training.

[States & Actions]. For the definition and the rationale behind the states we use, we point the reader to Appendix B. For actions, we control BAGS through 4 parameters: orthogonal step size, source step size, mask bias, and Perlin bias. HSJA is controlled by 3: the gradient estimation radius, the number of estimation queries, and the jump step size. All evaluations start from controlling these attack parameters *only*; if the active defense proves impossible to defeat, we introduce additional knobs that control the magnitude and probability of transformations on the input, with the goal to evade detection *while* preserving semantic content and hence the correct classification. The range of transformations we experimented with as well as their magnitude and probability are listed in Table V. Finally, in both BAGS and HSJA the active defense consists of an 1-dimensional continuous action that controls the radius of confinement a, as depicted in Fig 3.

[Rewards]. Success in an RL task relies heavily on *how* it is rewarded. Engineering an effective reward function is non-trivial and hides intricacies, as reward hacking and specification gaming are common phenomena and the learned behavior can vary [49]. For adversaries, the rewards we experimented with are variations on minimizing the distance to the original example – with extra reward shaping based on the fundamental operation of each attack – while defenders are rewarded or penalized for intercepting adversarial or benign queries respectively. The rewards are described in Appendix B.

V. Evaluation

For evaluation, we define a range of scenarios intended to reflect all possible and realistic combinations between adversarial attacks and defenses, and their adaptive versions. Concretely, the research questions we want to evaluate are: 1) Are active defenses a necessary complement to model hardening and to what extent? 2) Are attacks more threatening when adaptive, i.e., do they outperform their vanilla versions *and* evade active detection? 3) If yes, to what extend active defenses recoup their performance by also turning adaptive?

[Metrics]. We employ **ASR** (Attack Success Rate) and $\mathbf{L_2}$ **norm** of the perturbation. For the former we set a fixed threshold of 3 for consistency between experiments, while the latter is a more fine-grained metric well suited for comparing baseline attacks, defenses, and their adaptive versions, as it is not based on an (arbitrary) perceptual threshold that can yield widely varying results when moved. The budgets we evaluate over are 1K, 2K and 5K queries. As robustness and classification accuracy are typically in trade-off, the third metric of interest is the benign sample accuracy (**Clean Acc.**) that the original model and the active defense achieve together.

A. Evaluation Setup

Our goal is to learn offensive and defensive policies that are *general* and transfer to *any* other evasion task, with the same attack and against the same model. Thus after training and validating the agents, the final performance is reported on a fixed hold-out set of 100 adversarial episodes where the starting and original samples are selected at random. As is best practice in AML, candidate samples are only those that are correctly classified by the model. For each scenario we perform a limited hyperparameter and reward function exploration (max 30 trials), with the intention to root out poor combinations rather than exhaust the search space, described in more detail in Appendix D. In our empirical evaluation we explore *4 types of attacks and 3 defenses* against each other, including the (at the time of writing) SotA attack and defense. We list them here together with the rationale behind their selection.

[Attacks]. The two black-box attacks we render self-adaptive are 1: **BAGS** and 2: **HSJA**. They represent two fundamentally different approaches, are highly effective, *and* have the highest evasion potential compared to others, as demonstrated in Blacklight [19]. BAGS is a stochastic, search-based method where every query submitted is a new and potentially better adversarial example. Contrastively, HSJA is deterministic and composed of 3 different stages where the queries are generated in an aggregated manner: the majority of them are not candidate adversarial examples but means of approximating the gradient at the decision boundary. Our 3: **Adaptive Attack** can be understood as an optimization wrapper around an existing black-box attack, so to accurately measure its potential we additionally compare it to 4: **OARS** [18], as this is the SotA approach.

[**Defenses**]. We explore both static and active defenses. For static, we evaluate 1: **AT**, as the preeminent method for empirical robustness. As AT is a static, non-interactive defense, we duplicate all our experimental scenarios over models with and without AT. For active, we evaluate 2: **Blacklight**, the SotA defense against black-box querying attacks, and our 3: **Adaptive Defense**, as defined in section IV.

In training and evaluation, the adversarial game is played as follows: the adversary starts by submitting a query, then the defender responds either *truthfully* (the actual model prediction) or by *misdirecting* (the second highest probability class). Then the environment decides with chance p if the adversary moves next, otherwise a benign query is drawn. In either case, it is the defender's turn; during testing they are also oblivious to the nature of the query and know only the content. All experiments are performed with $p = 0.5$; we also evaluate our trained defense when no attack is present ($p = 0$) in Appendix C.

The scenarios for all possible combinations of (non-) adaptive attacks and defenses are repeated over two datasets – CIFAR10 and MNIST – and over two models with the same architecture but different training regimes: with and without AT. As the transition from single to multi-agent RL introduces non-stationarity, we approach the AMG as a belief-state MDP (relaxing the requirement of knowing the exact opponent policies), and use PPO [43] agents to learn optimal policies that will also constitute best responses for the full game [48]. Note that learning independently of other agency breaks the theoretical guarantees of convergence [50], eg. in scenarios 7 & 8 where both agents learn simultaneously. *Vanilla* indicates a baseline approach, *Trained* indicates an adapted approach (fixed and stationary), while *Adaptive* indicates an approach that is learning and adapting to the Trained/Vanilla counterpart. Additionally, coloring denotes the agent that is being evaluated in each scenario. The complete list of scenarios is as follows:

0) **VA-ND** – Vanilla Attack / No Defense: Baseline performance of attacks (BAGS & HSJA) out-of-the-box, without any active defense.
1) **AA-ND** – Adaptive Attack / No Defense: How much more optimal is the adaptive version compared to the baseline attack.
2) **VA-VD** – Vanilla Attack / Vanilla Defense: The performance of our active defense, the non-adaptive version that has an empirically defined detection threshold.

TABLE II

ASR AND MEAN L_2 PERTURBATION FOR 1K, 2K, AND 5K QUERIES FOR CIFAR-10, AGAINST NORMALLY AND ADVERSARIALLY TRAINED MODELS. CLEAN ACC. REPORTS THE ACCURACY ON BENIGN QUERIES OF THE BASE MODEL PLUS ANY DEFENSES PRESENT; IN THE FIRST TWO SCENARIOS (NO ACTIVE DEFENSE) THE BASELINE CLEAN ACCURACY IS REPORTED. YELLOW SCENARIOS DENOTE THE BASELINE ATTACK PERFORMANCE, WHILE GREEN AND RED DENOTE DEFENSIVE AND OFFENSIVE SCENARIOS RESPECTIVELY. THE ASTERISK DENOTES WHERE INPUT TRANSFORMATIONS WERE USED FOR EVASION.

Adv. Trained	Scenario	**CIFAR-10 Gap: 20.01**									
		BAGS				HSJA				Clean Acc.	
		1K	2K	5K	ASR	1K	2K	5K	ASR	BAGS	HSJA
✘	**0: VA-ND**	8.27	7.86	7.26	**5%**	3.42	1.43	0.41	**100%**	91.69	91.69
	1: AA-ND	1.26	0.71	0.49	**100%**	3.14	1.31	0.39	**100%**	91.69	91.69
	2: VA-VD	15.27	15.26	15.20	**0%**	11.14	10.81	10.33	**7%**	91.68	91.68
	3: AA-VD	2.63	2.03	1.77	**93%**	5.68	3.61	2.12	**85%**	91.69	91.69
	4: VA-AD	20.01	20.01	20.00	**0%**	17.17	16.35	15.56	**0%**	91.60	91.50
	***5: AA-TD**	6.28	5.45	4.52	**30%**	13.19	11.82	10.69	**2%**	91.52	91.46
	***6: TA-AD**	19.52	19.40	18.95	**0%**	16.48	16.13	15.69	**0%**	91.38	91.62
	***7: AA-AD**	9.95	9.80	9.80	**5%**	10.30	9.04	7.55	**23%**	91.66	91.55
	***8: AA-AD**	19.85	19.85	19.85	**0%**	14.46	13.93	13.08	**1%**	91.69	91.37
✔	**0: VA-ND**	8.72	8.42	7.94	**4%**	3.73	1.74	0.75	**100%**	87.76	87.76
	1: AA-ND	1.74	1.13	0.79	**100%**	3.64	1.77	0.73	**100%**	87.76	87.76
	2: VA-VD	15.42	15.35	15.20	**0%**	11.10	10.73	10.38	**4%**	87.72	87.73
	3: AA-VD	2.82	2.26	2.06	**81%**	5.66	3.36	1.94	**86%**	87.74	87.74
	4: VA-AD	20.01	20.01	20.00	**0%**	17.06	16.40	15.81	**0%**	87.66	87.66
	***5: AA-TD**	8.48	7.68	6.82	**9%**	13.59	12.65	11.39	**1%**	87.58	87.52
	***6: TA-AD**	19.58	19.40	18.95	**0%**	16.60	16.26	15.99	**0%**	87.50	87.68
	***7: AA-AD**	10.43	10.24	10.17	**1%**	10.21	9.22	7.82	**12%**	87.73	87.61
	***8: AA-AD**	19.86	19.86	19.86	**0%**	15.71	15.35	14.30	**1%**	87.67	87.40

3) **AA-VD** – Adaptive Attack / Vanilla Defense: Similar to scenario (2), but now the attack is adaptive.
4) **VA-AD** – Vanilla Attack / Adaptive Defense: The first scenario where the active defense is also adaptive, against the baseline adversary.
5) **AA-TD** – Adaptive Attack / Trained Defense: After the adaptive defense is optimized, its policy is fixed and an adaptive attack is trained against it.
6) **TA-AD** – Trained Attack / Adaptive Defense: The best policy found in the previous scenario is fixed and an adaptive defense is trained against it.
7) **AA-AD** – Adaptive Attack / Adaptive Defense: The first scenario where both agents learn simultaneously, making the environment non-stationary. In practice, the convergence will vary and depend on the chosen hyperparameters and rewards. Here we report the best-case for the attack.
8) **AA-AD** – Adaptive Attack / Adaptive Defense: The exact setup as scenario 7, but the best-case for the defense is reported instead.

In each successive scenario, we evaluate using the most successful past policy, following standard practice in Markov Games: the worst-case opponent policy is fixed, and a best response to it is learned [32], [39]. Fixing other policies when computing a best response stabilizes learning in multi-agent environments, as it simplifies the problem to a single-agent setting – one that, as discussed in Section III-C, can be solved with standard RL.

Comparison to SotA. In Scenarios 0-8 we evaluate all possible combinations between (adaptive) attack and defenses. As a baseline to compare to, we additionally evaluate our approach to the state-of-the-art stateful defenses and adaptive attacks, that is Blacklight [19] and OARS [18] respectively. We implement both Blacklight and OARS in our interactive environments by using their publicly available code and parameters. As our environments do not return a rejection signal and to make a fair comparison, for OARS rejection coincides with a non-adversarial decision. We thus define 5 further scenarios:

9) **VA-BD** – Vanilla Attack / Blacklight Defense: Baseline performance of the attacks against Blacklight.
10) **OA-BD** – OARS Attack / Blacklight Defense: OARS against Blacklight.
11) **AA-BD** – Adaptive Attack / Blacklight Defense: Our adaptive attack against Blacklight.
12) **OA-TD** – OARS Attack / Trained Defense: OARS against our trained defense from Scenario 6.
13) **OA-AD** – OARS Attack / Adaptive Defense: Our adaptive defense retrained against OARS.

Our experiments were run on multiple machines, yet to give an idea for the time complexity of our defense, on an Intel i7-7700 CPU one forward pass in CIFAR – that is one response to one query – takes 8 ± 1.4 ms for $\sim$700 MFLOPs.

B. Results

For consistency and comparability between evaluations, all results are from the *same* 100 test episodes. The **gap** value denotes the L_2 perturbation that initially separates the starting and the original samples, averaged over the 100 episodes.

TABLE III
ASR AND MEAN L_2 PERTURBATION FOR CIFAR-10, COMPARING OUR ADAPTIVE ATTACK (AA) AND ADAPTIVE DEFENSE (AD) TO BLACKLIGHT (BD) AND OARS (OA).

Adv. Trained	Scenario	CIFAR-10 Gap: 20.01									
		BAGS				HSJA				Clean Acc.	
		1K	2K	5K	ASR	1K	2K	5K	ASR	BAGS	HSJA
✘	**9: VA-BD**	9.55	9.32	9.17	**0%**	8.41	8.19	7.80	**15%**	91.71	91.71
	10: OA-BD	9.46	9.46	9.46	**1%**	6.54	5.83	4.67	**50%**	91.71	91.71
	11: AA-BD	2.26	1.39	1.32	**98%**	4.55	3.08	2.44	**78%**	91.71	91.71
	12: OA-TD	20.01	20.01	20.01	**0%**	7.07	6.38	5.53	**50%**	91.61	91.59
	13: OA-AD	20.01	20.01	20.01	**0%**	11.03	11.00	10.95	**5%**	91.61	91.69
✔	**9: VA-BD**	9.75	9.56	9.46	**0%**	8.67	8.50	8.28	**7%**	87.76	87.76
	10: OA-BD	9.79	9.79	9.79	**1%**	5.77	4.53	3.26	**72%**	87.76	87.76
	11: AA-BD	5.59	4.04	2.55	**79%**	5.59	4.04	2.55	**79%**	87.76	87.76
	12: OA-TD	20.01	20.01	20.01	**0%**	6.44	5.49	4.38	**65%**	87.66	87.64
	13: OA-AD	20.01	20.01	20.01	**0%**	11.31	11.12	10.97	**7%**	87.66	87.74

By testing the trained agents on budgets higher than 5K we discovered that the trend in reducing L_2 holds; to make the agent training tractable and the evaluation wider however, we limit the maximum query budget per adversarial episode to 5K. Table II & Table III report the results for CIFAR10, while Table IV reports MNIST. The closer examination of the empirical results help us extract and highlight several important insights, practical observations, and general implications for the broader AML field:

- The initial performance of an attack can be misleading: out-of-the-box HSJA appears to be the better attack, but it is often outperformed by adaptive BAGS, especially in CIFAR and against active defenses.
- The performance of both attacks deteriorates considerably against active defenses (VD), however the defenses reach their full potential only when *also adaptive* (AD).
- Our adaptive defense (AD) outperforms both Blacklight (BD) and non-adaptive (VD) defenses, also when transferred (cf. Sc.12). Compared to Blacklight, it reduces ASR by $\sim$90% (in HSJA) when trained against OARS, while it offers similar protection when transferred from another attack.
- Overall, against the strongest attacks and in the worst case (Sc.7) for it, our defense contains the ASR in the range of $1-36\%$.
- Our adaptive attack (AA) outperforms OARS (Sc.10 & 11) and vanilla attacks (VA) by a wide margin, *without* access to rejection sampling, and irrespective of the defense it faces; the one exception is our adaptive defense (AD), against which it has very limited success.
- The advantage of adaptive attacks is more pronounced against active defenses, where they significantly outperform non-adaptive versions, cf. Sc.0→1, Sc.2→3.
- When comparing the upper and lower halves of each table, we can observe that AT adds a limited amount of robustness; otherwise, *the practical effect of AT is a tax on the attacker*, forcing them to expend more queries for the same perturbation or having higher perturbation for the same query budget.
- Evasive transformations interfere with the attack policy, as illustrated by the difference between BAGS and HSJA in Sc.5. For an attack to reach its full potential, these two should be adaptively controlled together.
- In Sc.1 to 8, agents train against the best opponent policy as previously discovered, and ASR oscillates since following a fixed policy enables the learning of an optimal counter to it. Over successive adaptations, ASR eventually plateaus, indicating an equilibrium for each specific dataset and attack combination. This is illustrated in Figure 4.
- The first time active defenses resisted adaptive attacks was in Sc.5 of CIFAR; we employed evasive transformations from then onward.
- Different attack fundamentals respond differently to active defenses; the gradient estimation stage of HSJA has a disadvantage against similarity detection, while the jump and binary stages have an advantage.
- For HSJA, engineering a state the adaptive defense could learn on, merely by leveraging our knowledge of the attack and its geometric functioning, proved impossible. What did prove effective however, was pure computation[2]: we used Contrastive Learning [52] to learn an embedding from raw queries, then used as state that transfers exceedingly well to other attacks like BAGS.

The main takeaway here is that no matter the underlying approach, to achieve optimal results adaptive strategies are necessary. Let us take HSJA as an example, which is an optimized attack (cf. Table I) and comes with convergence guarantees, and examine how its effectiveness changes when it switches from a vanilla or trained strategy to an adaptive strategy. In Table II, when no active defense is present (Scenario 0), HSJA achieves a perfect ASR in CIFAR-10. Against active defenses (Scenario 2) the ASR drops to $4-7\%$. However, when HSJA is turned adaptive (Scenario 3), ASR is recouped back

[2]This is reminiscent of Sutton's Bitter Lesson [51], the observation that progress in AI is often driven by gains in computation rather than problem-specific expert knowledge.

TABLE IV
ASR AND MEAN L_2 PERTURBATION FOR 1K, 2K, AND 5K QUERIES AND ACCURACY ON CLEAN DATA FOR MNIST. THE EVALUATION SCENARIOS ARE IDENTICAL TO TABLES II AND III.

Adv. Trained	Scenario	MNIST Gap = 10.62									
		BAGS				HSJA				Clean Acc.	
		1K	2K	5K	ASR	1K	2K	5K	ASR	BAGS	HSJA
✘	**0: VA-ND**	5.30	5.28	5.26	**3%**	3.59	3.07	2.61	**73%**	99.37	99.37
	1: AA-ND	2.74	2.57	2.47	**78%**	3.61	3.09	2.60	**74%**	99.37	99.37
	2: VA-VD	7.44	6.66	5.63	**22%**	5.82	5.78	5.73	**2%**	99.34	99.20
	3: AA-VD	3.79	3.66	3.44	**29%**	3.54	3.09	2.77	**61%**	99.37	99.31
	4: VA-AD	10.57	10.57	10.57	**0%**	10.05	10.05	10.05	**0%**	99.31	99.30
	5: AA-TD	3.57	3.29	3.14	**39%**	5.00	3.97	3.38	**36%**	99.32	98.84
	6: TA-AD	10.62	10.62	10.62	**0%**	10.23	10.23	10.18	**0%**	99.28	99.34
	7: AA-AD	4.89	4.89	4.86	**8%**	5.06	4.76	4.38	**36%**	99.31	99.35
	8: AA-AD	10.62	10.62	10.62	**0%**	10.21	10.21	10.21	**0%**	99.32	99.23
	9: VA-BD	10.62	10.62	10.62	**0%**	5.65	5.65	5.65	**2%**	99.37	99.37
	10: OA-BD	10.62	10.62	10.62	**0%**	4.53	4.00	3.15	**46%**	99.37	99.37
	11: AA-BD	3.83	3.69	3.60	**17%**	4.18	3.66	3.19	**52%**	99.37	99.37
	12: OA-TD	10.62	10.62	10.62	**0%**	10.21	10.20	10.20	**0%**	99.22	99.28
	13: OA-AD	10.62	10.62	10.62	**0%**	10.21	10.20	10.20	**0%**	99.32	99.28
✔	**0: VA-ND**	5.26	5.25	5.24	**2%**	4.61	4.04	3.41	**30%**	99.15	99.15
	1: AA-ND	3.28	3.08	2.96	**51%**	4.59	3.97	3.35	**34%**	99.15	99.15
	2: VA-VD	7.70	6.86	5.86	**17%**	5.81	5.78	5.76	**2%**	99.14	99.12
	3 AA-VD	4.18	4.08	3.86	**22%**	4.63	4.27	3.86	**25%**	99.13	99.15
	4: VA-AD	10.55	10.55	10.55	**0%**	10.02	10.02	10.02	**0%**	99.09	99.08
	5: AA-TD	4.04	3.74	3.54	**27%**	5.82	5.09	4.26	**16%**	99.11	98.78
	6: TA-AD	10.62	10.62	10.62	**0%**	10.20	10.20	10.20	**0%**	99.06	99.06
	7: AA-AD	5.59	5.56	5.56	**5%**	5.47	5.16	4.99	**14%**	99.09	99.13
	8: AA-AD	10.62	10.62	10.62	**0%**	10.12	10.12	10.12	**0%**	99.10	99.01
	9: VA-BD	10.62	10.62	10.62	**0%**	5.65	5.64	5.64	**1%**	99.15	99.15
	10: OA-BD	10.62	10.62	10.62	**0%**	5.18	4.80	4.11	**17%**	99.15	99.15
	11: AA-BD	4.31	4.07	3.96	**13%**	5.04	4.65	4.20	**19%**	99.15	99.15
	12: OA-TD	10.62	10.62	10.62	**0%**	10.26	10.26	10.26	**0%**	99.00	99.06
	13: OA-AD	10.62	10.62	10.62	**0%**	10.26	10.26	10.26	**0%**	99.10	99.06

to $85 - 86\%$.

How do the above scenarios translate to a real-world attack, and what would "turning adaptive" entail in practice? Let us assume here a black-box deployed model $\mathcal{M}$ (e.g. Google Cloud Vision API) with an active defense in place that aims to stop querying attacks, as introduced in section IV. If we employed HSJA with default parameters, it would fail to create a successful adversarial example. Instead, under our AMG formulation (III-C) we can assign the controllable HSJA parameters as the action set of the attacker $A^{\mathcal{A}}$, then Proposition III.3 describes how the attack will recoup its performance based on the model's responses. Naturally, this requires several attack episodes and multiple queries to collect state–action–reward trajectories with which the policy parameters are gradually adjusted, till convergence.

VI. Discussion

Our work has several implications for performing robust inference in the real-world. While AT remains the most reliable defense, the empirical robustness it imparts will vary and even be insufficient. We note that the robustness AT confers is against *all* adversarial examples under the same L_p-norm; active defenses protect only against querying attacks, but as they do transfer between attacks (Scenario 12) they can be used jointly as complementary approaches. We demonstrated how AI-enabled systems are susceptible to adaptive adversaries that *devise* new evasive techniques and *control them jointly* with other attack parameters. This has been achieved in the *fully black-box* case and *against active defenses*.

We note here that the level of threat posed by adaptive adversaries remains considerable, as Proposition III.3 does not rely on the specifics of the underlying decision process and can therefore be extended to other domains or modalities under similar conditions. This is further supported by our empirical investigation, where we evaluated with diverse methodologies, including the state-of-the-art attacks and defenses at the time of writing. Our results rekindle the proverbial arms race, where as a consequence to adaptive attacks, defenses should also have adaptive capabilities.

Limitations. To keep the amount of evaluations practical, we narrowed the scope to targeted attacks and to L_2 as the more suitable norm for visual similarity. Targeted attacks are strictly more difficult to perform than untargeted, while for binary classification targeted and untargeted coincide; our framework, however, can accommodate any adversarial goal or metric. Another simplifying assumption we make is that only one attack can take place at a time; however, the queuing technique we use for incoming queries is readily extensible to handle concurrent attacks. While we demonstrate that our active defense does transfer between attacks, another possibility to explore is training the defense on queries from different kinds of attacks. It should be emphasized here that active defenses – as any other defense – should always be evaluated w.r.t. the cost they incur. The counterpart to the protection provided against querying attacks, is the reduction of performance on clean data. If we however compare the upper and lower halves of Tables II & IV, we observe that AT incurs a greater cost on clean accuracy than our active defense. Finally, in our evaluation we focus on a wide range of adaptive and non-adaptive scenarios where agents learn and adapt interactively, which limits the number of datasets we experiment with; nevertheless, our empirical investigation is complemented by a theoretical analysis that lends support to the generality of our findings independent of context.

Future Work. The AMG framework we introduce is general by design and can accommodate the learning of optimal offensive and defensive policies in any domain of interest beyond image classification. A promising path for future research is the extension of our adaptive attacks and defenses to other domains and modalities, for instance malware, bot, and network intrusion detection. This is specifically because our approach circumvents the main obstacle of mapping gradient-based perturbations to feasible objects (eg. binaries) and instead can function directly in the problem space [37]. Another compelling and formidable challenge is automating the adaptive evaluations in AML, that is adapting beyond a specification by inventing tools to bypass defenses and thus imparting controllability to adversarial tasks. Finally, in our work we considered opponent agency as part of the environment; other domains, like malware detection, might benefit from explicit opponent modeling.

VII. Conclusion

With adaptive, decision-based attacks becoming more pervasive in multiple domains, every AI-based system that exposes a queryable interface is inherently vulnerable. To aggravate matters, this vulnerability cannot be mitigated by employing model hardening approaches like adversarial training alone. To fully defend in the presence of such attacks, active *and* adaptive defenses are necessary, and we demonstrate how optimal defensive policies can be learned. However, the existence of such defenses elicits in turn adaptive attacks which are able to recover part of their original performance.

We perform a theoretical and empirical investigation of decision-based attacks and stateful defenses under a unified framework we name "Adversarial Markov Games" (AMG). In self-adaptive, we introduce a novel twofold definition of adaptive: both inventing new techniques to outmaneuver opponents *and* adapting one's operating policy with respect to other agency in the environment. Furthermore, through our theoretical analysis we demonstrate how any combination of adversarial goals, be it performance, stealthiness [53], or disruption, can be optimized in a gradient based manner, even in the *complete* black-box case and in *any* domain. As new attacks and defenses constantly emerge and are surpassed, our proposed methodology is generally applicable as it turns any such approaches in the current arms-race self-adaptive, thus ensuring accurate and robust assessment of their performance.

The AMG framework we introduce helps us reason on and properly assess the vulnerabilities of AI-based systems, disentangling the inherently complex and non-stationary task of learning in the presence of competing agency. By modeling the latter as part of the environment, we can simplify this task by computing a best response to the observed behavior. This has a significant consequence for the security of AI-based systems independent of modality or application: as long as proper threat modeling is carried out, one can readily employ RL agents in order to devise optimal defenses, but only after they devised optimal attacks too.

Acknowledgment

This research is partially funded and supported by: the Research Fund KU Leuven; the Cybersecurity Research Program Flanders; the EU funded project KINAITICS (Grant Agreement Number 101070176); Google ASPIRE Research Awards; EPSRC Grant EP/X015971/2.

References

[1] Y. He, G. Meng, K. Chen, X. Hu, and J. He, "Towards security threats of deep learning systems: A survey," *IEEE Transactions on Software Engineering*, vol. 48, no. 5, pp. 1743–1770, 2020.

[2] H. S. Anderson, A. Kharkar, B. Filar, D. Evans, and P. Roth, "Learning to evade static pe machine learning malware models via reinforcement learning," *arXiv preprint arXiv:1801.08917*, 2018.

[3] L. Demetrio, B. Biggio, G. Lagorio, F. Roli, and A. Armando, "Functionality-preserving black-box optimization of adversarial windows malware," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3469–3478, 2021.

[4] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[5] Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu, "On the convergence and robustness of adversarial training," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6586–6595.

[6] W. Brendel, J. Rauber, and M. Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models," in *International Conference on Learning Representations*, 2018.

[7] J. Chen, M. I. Jordan, and M. J. Wainwright, "Hopskipjumpattack: A query-efficient decision-based attack," in *2020 ieee symposium on security and privacy (sp)*. IEEE, 2020, pp. 1277–1294.

[8] Z. Yan, Y. Guo, J. Liang, and C. Zhang, "Policy-driven attack: learning to query for hard-label black-box adversarial examples," in *International Conference on Learning Representations*, 2020.

[9] I. Tsingenopoulos, A. M. Shafiei, L. Desmet, D. Preuveneers, and W. Joosen, "Adaptive malware control: Decision-based attacks in the problem space of dynamic analysis," in *Proceedings of the 1st Workshop on Robust Malware Analysis*, 2022, pp. 3–14.

[10] A. Gleave, M. Dennis, N. Kant, C. Wild, S. Levine, and S. Russsell, "Adversarial policies: Attacking deep reinforcement learning," *In Proc. ICLR-20*, 2020.

[11] F. Barbero, F. Pendlebury, F. Pierazzi, and L. Cavallaro, "Transcending transcend: Revisiting malware classification in the presence of concept drift," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 805–823.

[12] S. Sengupta and S. Kambhampati, "Multi-agent reinforcement learning in bayesian stackelberg markov games for adaptive moving target defense," *arXiv e-prints*, pp. arXiv–2007, 2020.

[13] S. Chen, N. Carlini, and D. Wagner, "Stateful detection of black-box adversarial attacks," in *Proceedings of the 1st ACM Workshop on Security and Privacy on Artificial Intelligence*, 2020, pp. 30–39.

[14] F. Tramer, N. Carlini, W. Brendel, and A. Madry, "On adaptive attacks to adversarial example defenses," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1633–1645, 2020.

[15] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[16] K. J. Åström and B. Wittenmark, *Adaptive control*, 1995.

[17] G. Apruzzese, H. S. Anderson, S. Dambra, D. Freeman, F. Pierazzi, and K. Roundy, ""real attackers don't compute gradients": bridging the gap between adversarial ml research and practice," in *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2023, pp. 339–364.

[18] R. Feng, A. Hooda, N. Mangaokar, K. Fawaz, S. Jha, and A. Prakash, "Stateful defenses for machine learning models are not yet secure against black-box attacks," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 786–800.

[19] H. Li, S. Shan, E. Wenger, J. Zhang, H. Zheng, and B. Y. Zhao, "Blacklight: Scalable defense for neural networks against {Query-Based}{Black-Box} attacks," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2117–2134.

[20] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[21] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International conference on machine learning*, 2018, pp. 274–283.

[22] Z. Qin, Y. Fan, H. Zha, and B. Wu, "Random noise defense against query-based black-box attacks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 7650–7663, 2021.

[23] J. Byun, H. Go, and C. Kim, "On the effectiveness of small input noise for defending against query-based black-box attacks," in *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022.

[24] C. Sitawarin, F. Tramèr, and N. Carlini, "Preprocessors matter! realistic decision-based attacks on machine learning systems," *arXiv preprint arXiv:2210.03297*, 2022.

[25] M. Juuti, S. Szyller, S. Marchal, and N. Asokan, "Prada: protecting against dnn model stealing attacks," in *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2019, pp. 512–527.

[26] M. Pintor, L. Demetrio, A. Sotgiu, A. Demontis, N. Carlini, B. Biggio, and F. Roli, "Indicators of attack failure: Debugging and improving optimization of adversarial examples," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 063–23 076, 2022.

[27] D. R. Hofstadter, *Metamagical themas: Questing for the essence of mind and pattern*. Hachette UK, 2008.

[28] T. Brunner, F. Diehl, M. T. Le, and A. Knoll, "Guessing smart: Biased sampling for efficient black-box adversarial attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[29] J. Bose, G. Gidel, H. Berard, A. Cianflone, P. Vincent, S. Lacoste-Julien, and W. Hamilton, "Adversarial example games," *Advances in neural information processing systems*, vol. 33, pp. 8921–8934, 2020.

[30] A. Pal and R. Vidal, "A game theoretic analysis of additive adversarial attacks and defenses," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1345–1355, 2020.

[31] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. de Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," *arXiv preprint arXiv:1707.09183*, 2017.

[32] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine learning proceedings*. Elsevier, 1994.

[33] M. Hardt, N. Megiddo, C. Papadimitriou, and M. Wootters, "Strategic classification," in *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, 2016, pp. 111–122.

[34] M. Cheng, S. Singh, P. H. Chen, P.-Y. Chen, S. Liu, and C.-J. Hsieh, "Sign-opt: A query-efficient hard-label adversarial attack," in *International Conference on Learning Representations*, 2019.

[35] H. Li, X. Xu, X. Zhang, S. Yang, and B. Li, "Qeba: Query-efficient boundary-based blackbox attack," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[36] T. Maho, T. Furon, and E. Le Merrer, "Surfree: a fast surrogate-free black-box attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 430–10 439.

[37] F. Pierazzi, F. Pendlebury, J. Cortellazzi, and L. Cavallaro, "Intriguing properties of adversarial ml attacks in the problem space," in *2020 IEEE symposium on security and privacy (SP)*. IEEE, 2020, pp. 1332–1349.

[38] C. Cortes, G. DeSalvo, and M. Mohri, "Learning with rejection," in *International Conference on Algorithmic Learning Theory*. Springer, 2016, pp. 67–82.

[39] F. Timbers, N. Bard, E. Lockhart, M. Lanctot, M. Schmid, N. Burch, J. Schrittweiser, T. Hubert, and M. Bowling, "Approximate exploitability: Learning a best response," *IJCAI, Jul*, 2022.

[40] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," *arXiv preprint arXiv:1611.01578*, 2016.

[41] H. Pham and Q. Le, "Autodropout: Learning dropout patterns to regularize deep networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 9351–9359.

[42] I. Tsingenopoulos, J. Cortellazzi, B. Bošanskỳ, S. Aonzo, D. Preuveneers, W. Joosen, F. Pierazzi, and L. Cavallaro, "How to train your antivirus: Rl-based hardening through the problem space," in *Proceedings of the 27th International Symposium on Research in Attacks, Intrusions and Defenses*, 2024, pp. 130–146.

[43] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv:1707.06347*, 2017.

[44] M. Brückner and T. Scheffer, "Stackelberg games for adversarial prediction problems," in *Proceedings of the 17th ACM SIGKDD conference on Knowledge discovery and data mining*, 2011, pp. 547–555.

[45] A. Greenwald, J. Li, and E. Sodomka, "Solving for best responses and equilibria in extensive-form games with reinforcement learning methods," pp. 185–226, 2017.

[46] X.-s. Gao, S. Liu, and L. Yu, "Achieving optimal adversarial accuracy for adversarial deep learning using stackelberg games," *Acta Mathematica Scientia*, vol. 42, no. 6, pp. 2399–2418, 2022.

[47] S. V. Albrecht and P. Stone, "Autonomous agents modelling other agents: A comprehensive survey and open problems," *Artificial Intelligence*, vol. 258, pp. 66–95, 2018.

[48] Y. Wen, Y. Yang, R. Luo, J. Wang, and W. Pan, "Probabilistic recursive reasoning for multi-agent reinforcement learning," in *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[49] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in ai safety," *arXiv preprint arXiv:1606.06565*, 2016.

[50] K. Tuyls and G. Weiss, "Multiagent learning: Basics, challenges, and prospects," *Ai Magazine*, vol. 33, no. 3, pp. 41–41, 2012.

[51] R. Sutton, "The bitter lesson," *Incomplete Ideas (blog)*, 2019.

[52] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.

[53] E. Debenedetti, N. Carlini, and F. Tramèr, "Evading black-box classifiers without breaking eggs," in *2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 2024, pp. 408–424.

[54] R. S. Sutton, D. A. McAllester, S. P. Singh, Y. Mansour *et al.*, "Policy gradient methods for reinforcement learning with function approximation." in *NIPs*, vol. 99. Citeseer, 1999, pp. 1057–1063.

Appendix A
Proofs

For a more intuitive understanding of the proofs, we provide a high-level description of the attack fundamentals. **BAGS** [28] performs a random walk along the boundary between the adversarial and the non-adversarial regions, by first taking a random step orthogonal to the original image direction, then a source step towards it. The randomness in the directions searched is reduced by utilizing Perlin noise and masks computed on the difference between starting and original samples. **HSJA** [7] operates in 3 stages: a binary search that places the current best adversarial on the decision boundary, an estimation step that computes the gradient at that point of the boundary, and projection step along the estimated gradient. These steps repeat until convergence.

[Proposition III.1]

Proof. We proceed in two parts, corresponding to conditions (a) and (b).

[Part 1] Let us denote by x_c, x_g, x_t the original (unperturbed), the starting, and the current sample at step t respectively. Given a target class $c_0 \in m$ we define a function:

$$S_{x_c}(x_t) = F_{c_0}(x_t) - \max_{c \neq c_0}(F_c(x_t)) \tag{8}$$

HSJA operates in 3 stages that alternate until convergence: (1) binary search between x_g and x_c that places x_t on the decision boundary, (2) gradient estimation approximating $\nabla S(x_t)$, (3) a step along the direction of the gradient $\nabla S(x_t)$. We repeat Eq. 9 of [7], denoting the gradient direction as the Monte Carlo estimate:

$$\widetilde{\nabla S_{x_c}}(x_t, \delta) = \frac{1}{B} \sum_{b=1}^{B} \phi_{x_c}(x_t + \delta u_b) u_b \tag{9}$$

where $\{u_b\}_{b=1}^{B}$ are i.i.d. draws from the uniform distribution over the d-dimensional sphere, δ is a small positive parameter, and ϕ_{x_c} is the Boolean-valued function that all stages rely on:

$$\phi_{x_c}(x_t) = \text{sign}(S_{x_c}(x_t)) = \begin{cases} +1 & \text{if} \quad S(x_t) > 0, \\ -1 & \text{if} \quad S(x_t) \leq 0. \end{cases} \tag{10}$$

Given x_c, in search of adversarial examples HSJA iteratively applies the following update function:

$$x_{t+1} = a_t x_c + (1 - a_t) \left\{ x_t + \xi_t \frac{\nabla S_{x_c}(x_t)}{\|\nabla S_{x_c}(x_t)\|_2} \right\} \tag{11}$$

where ξ_t is a positive step size and a_t is a line search parameter in $[0, 1]$ s.t. $S(x_{t+1}) = 0$, i.e. the next query lies on the boundary. Now let us assume that the decision function D is $\arg\max$, i.e. $D : \mathbb{R}^m \mapsto \mathbb{N}^m$, $C(x) = D(F_c(x)) = \arg\max F_c(x)$, then from Eq. 3 and Eq. 8 we have:

$$\begin{aligned} S_{x_c}(x_t) > 0 &\iff C(x_t) = c_0 \\ S_{x_c}(x_t) < 0 &\iff C(x_t) \neq c_0 \\ S_{x_c}(x_t) = 0 &\iff C(x_t) = \{c_0, a\},\ a \neq c_0 \\ \implies S_{x_c}(x_t) \leq 0 &\iff C(x_t) \neq c_0 \end{aligned} \tag{12}$$

Let us define the function $\mathcal{I}$ of two variables:

$$\mathcal{I}(a, b) := \begin{cases} +1 & \text{if} \quad a = b, \\ -1 & \text{if} \quad a \neq b. \end{cases} \tag{13}$$

From 12 and 13 we can rewrite Eq. 10 as follows:

$$\phi_{x_c}(x_t) = \mathcal{I}(C(x_t), c_0) \tag{14}$$

Provided that the gradient estimation happens at the decision boundary where $S(x_t) = 0$, Theorem 2 of [7] guarantees that the gradient estimation is an asymptotically unbiased direction of the true gradient:

$$\widetilde{\nabla S_{x_c}}(x_t, \delta) \approx \nabla S_{x_c}(x_t), \delta \to 0 \tag{15}$$

For $b_t = 1 - a_t$ and by plugging 14 & 15 in Eq. 9, and the result in 11, we get:

$$\begin{aligned} x_{t+1} &= a_t x_c + b_t \left\{ x_t + \xi_t \frac{\nabla S_{x_c}(x_t)}{\|\nabla S_{x_c}(x_t)\|_2} \right\} \\ &= a_t x_c + b_t \left\{ x_t + \xi_t \frac{\frac{1}{B}\sum_{b=1}^{B} \phi_{x_c}(x_t + \delta u_b) u_b}{\|\frac{1}{B}\sum_{b=1}^{B} \phi_{x_c}(x_t + \delta u_b) u_b\|_2} \right\} \\ &= a_t x_c + b_t \left\{ x_t + \xi_t \frac{\frac{1}{B}\sum_{b=1}^{B} \mathcal{I}(C(x_t + \delta u_b), c_0) u_b}{\|\frac{1}{B}\sum_{b=1}^{B} \mathcal{I}(C(x_t + \delta u_b), c_0) u_b\|_2} \right\} \end{aligned} \tag{16}$$

In Eq. (16), the iterates x_t are *guaranteed* by Theorem 1 of HSJA [7] to converge to a stationary point x_b of Eq. (5), that is $\mathbb{E}[\sum_{i=1}^{N} d(x_b^i, x_c^i)] < \epsilon$, for ϵ a standard imperceptibility threshold and N adversarial episodes, which contradicts the requirement $\mathbb{E}[\sum_i d(x_b^i, x_c^i)] \geq \epsilon$. Since the only model-dependent term in Eq. (16) is the classifier $C(\cdot)$, the contradiction can be avoided only with an alternative classifier C'. With $C(x) = D(F_c(x))$, and the discriminant function F_c unable to change without retraining, it follows that $D' \neq \arg\max$, so for adv. example x_t misclassified as c_0, Eq. 4 can return -1:

$$\begin{gathered} C(x_t) = c_0 \Rightarrow \psi(x_t) = -1 \\ \therefore C(x_t) = \hat{c},\ \hat{c} = \{c_0, m \setminus c_0, \varnothing\} \end{gathered} \tag{17}$$

where $\varnothing$ denotes rejection and $\{m \setminus c_0\}$ denotes misdirection, i.e. intentional misclassification.

[Part 2] Let us assume that at timestep t, x_t is not yet adversarial, it is however still *part of* an ongoing adversarial attack. To deter the attack, a perfect defense would have to misclassify/reject this example; yet if an identical but benign example x_n was submitted, classifier C should preserve its capacity to classify it correctly. Since any memoryless classifier must assign the same label whenever $x_n = x_t$, we require a richer decision rule $C'(x, \tau) = D(\tau,\ F_c(x))$ that takes as auxiliary input context τ, e.g. the history of queries $\{x_0, \ldots, x_t\} \cup x_n$. By construction, even if $x_n = x_t$, differing contexts $\tau_n \neq \tau_t$ can force $C'(x_n, \tau_n) \neq C'(x_t, \tau_t)$, thereby separating benign from adversarial queries. □

[Proposition III.2] We annotate terms with $\mathcal{D}$ when an active defense $\pi_\phi^{\mathcal{D}}$ is present, and with $\not{\mathcal{D}}$ otherwise.

Proof. **BAGS**. This attack is in effect a gradual interpolation from x_g towards x_c, by first taking orthogonal steps x_s on the hypersphere around x_c and then source steps towards x_c in order to minimize $d(x_c - x_b)$, where x_b is the best adversarial example found so far. The source step parameter $\epsilon = (1.3 - \min(\lambda_n, 1)) \cdot c$ – with λ_n the ratio of the n last queries x_t that are adversarial and c a positive constant – controls the projection towards x_c:

$$x_t = x_s + \epsilon \cdot (x_c - x_s) \tag{18}$$

Then if we again assume that a non-zero amount of the adversarial queries x_t is flagged as such by the defense, it follows that $\lambda_n^{\not{\mathcal{D}}} > \lambda_n^{\mathcal{D}}$ and from the definition of ϵ we get $\epsilon^{\mathcal{D}} < \epsilon^{\not{\mathcal{D}}}$. At given t, from Eq. (18) we get that $d(x_c, x_t^{\mathcal{D}}) > d(x_c, x_t^{\not{\mathcal{D}}})$, and ceteris paribus the expectation (6) will be larger with $\pi_\phi^{\mathcal{D}}$ present than without.

HSJA. We denote the queries during gradient estimation as $x_n = x_t + \delta u$, $u \sim Uniform_{Sphere}(d)$, the ratio of those x_n detected as adversarial by the active defense as $\eta \in [0, 1]$, and the estimate $\widetilde{\nabla S_{x_c}}(x_t, \delta)$ as u_t. We investigate the behavior of active defenses as the ratio of detections η goes to 1.

For $\eta = 1 \implies \mathbb{E}[\phi_{x_c}(x_n)] = -1$, and as u_b are uniformly distributed, from Eq. (9) we get:

$$\lim_{\eta \to 1} u_t = \lim_{\eta \to 1} \frac{1}{B} \sum_{b=1}^{B} \phi_{x_c}(x_t + \delta u_b) u_b = \frac{1}{B} \sum_{b=1}^{B} -u_b \tag{19}$$

At the limit of detection we observe that the gradient estimate u_t behaves like a uniformly drawn vector around x_t of shrinking size. By the Law of Large Numbers, as B increases the average direction of u_t will align with the expected value: that is a random direction on the unit hypersphere. However, due to the $\frac{1}{B}$ term, the size of u_t goes to 0. From Eq. (19) then we get: $\lim_{\eta \to 1} u_t = 0$. The gradient estimation step is followed by the "jump" step that computes x_{t+1} as follows:

$$x_{t+1} = x_t + \xi u_t \tag{20}$$

As the ratio of detections η approaches 1, we observe that the adversarial iterates x_{t+1} converge prematurely: then all else being equal and for given t, $d(x_c, x_t^{\mathcal{D}}) > d(x_c, x_t^{\not{\mathcal{D}}})$.

□

[Proposition III.3]

Proof. Let $\pi_\theta^{\mathcal{A}}$ be the adversarial policy generating queries x_t, in N episodes τ_i of length L. The defense $\pi_\phi^{\mathcal{D}}$ (7), upon receiving a query x_t outputs a decision $\alpha_t = \pi_\phi^{\mathcal{D}}(x_t, s_t^{\mathcal{D}}) \in \{0, 1\}$, with 1 and 0 indicating rejection and acceptance respectively. The goal of the adversary is to find the parameters θ^* that maximize the expected reward:

$$\mathcal{J}(\theta) = \mathbb{E}_{\pi_\theta^{\mathcal{A}}}[r(\tau_i)] = \frac{1}{N} \sum_{i=1}^{N} r(\tau_i) = \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{L} (1 - \alpha_t) \tag{21}$$

The gradient of $\mathcal{J}(\theta)$ with respect to the policy parameters θ is the direction of steepest ascent for Eq. (21). Through the Policy Gradient Theorem [54] we can express the gradient of the expected reward in terms of the gradient of the log-likelihood of the policy, weighted by the reward:

$$\begin{aligned} \nabla_\theta \mathcal{J}(\theta) &= \nabla_\theta \mathbb{E}_{\pi_\theta^{\mathcal{A}}}[r(\tau_i)] \\ &= \mathbb{E}_{\pi_\theta^{\mathcal{A}}}\left[r(\tau_i) \nabla_\theta \log \pi_\theta(\tau_i)\right] \\ &= \mathbb{E}_{\pi_\theta^{\mathcal{A}}}\left[\sum_{t=1}^{L} (1 - a_t) \nabla_\theta \log \pi_\theta^{\mathcal{A}}(\tau_i)\right] \\ &\approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{L} (1 - a_t) \nabla_\theta \log \pi_\theta^{\mathcal{A}}(\tau_i) \end{aligned} \tag{22}$$

The gradient is thus estimated by sampling N episodes from the policy $\pi_\theta^{\mathcal{A}}$ to compute Eq. (22). To maximize the expectation, we iteratively update the policy parameters θ using gradient ascent: $\theta \leftarrow \theta + \eta \nabla_\theta \mathcal{J}(\theta)$. With $\eta > 0$ the learning rate, this process converges to θ^*. Recall that Eq. (21) attains its maximum for $\sum_{i=1}^{N} \sum_{t=1}^{L} a_t = 0$, therefore the converged policy $\pi_{\theta^*}^{\mathcal{A}}$ will correspond to the optimal evasive policy.

□

Appendix B
On States & Rewards

[States]. To handle the partial observability, we engineer states that incorporate past information. For **BAGS**, the adversary uses an 8-dimensional state representation with the following information normalized in the range $[0, 1]$: current amount of queries i, average queries that are adversarial a, the initial gap g, the current gap d, the location $l = \frac{d}{g}$, the slope $s = m - l$ where m is a moving average of the location, the frequency of improvement f, and r which is a moving average of the perturbation reduction n, In **HSJA** the state representation is slightly different: $r = \frac{n}{g}$, and $f = \frac{1}{j}$ with j number of jump steps in last iteration.

For the defense, in **HSJA** (and to a lesser extent **BAGS**) it has been difficult to engineer a state for policies to effectively learn on. The knowledge of the attack internals and fundamentals, geometric properties and distances, model activations and logits, and any combination thereof, did not suffice. Ultimately we decided to learn a representation instead. This representation is a 64-dimensional embedding of a CNN trained with triplet loss, on data generated by HSJA and benign queries, with the input being a tensor of the last query subtracted from the 25 most recent adversarial queries and then stacked.

[Rewards]. The concrete definitions of the rewards for each type of agent are:

- BAGS adversary: with $x \in [1, 50]$ the number of queries to a better adversarial example and t the maximum queries: **R1** $= \frac{n \cdot x}{g}$ if $n > 0$ else 0 | **R2** $= \frac{n}{g \cdot (x+1)}$ if $n > 0$ else 0 | **R3** $= (1 - \sqrt{\frac{d}{g}})^2 - (1 - \sqrt{\frac{d+n}{g}})^2$ | **R4** $= \sqrt{i} \cdot R2$ | **R5** $= |\log(d/g)|$ if $i \geq t$ else 0 | **R6** $= \sqrt[4]{i} \cdot a$ | **R7** $= R4 + R6$.
- HSJA adversary: with e the gradient estimation steps: **R1** $= 2 \cdot n$ | **R2** $= \frac{-e}{1000} + R1$ | **R3** $= \frac{10 \cdot n}{d}$ | **R4** $= \frac{1}{d}$ | **R5** $= \frac{2 \cdot (g-d)}{g}$ if $i \geq t$ else 0 | **R6** $= 2 \cdot (0.5 - |\frac{a+1}{2} - 0.5|) + b$, where $b = \frac{j}{20}$ if $j < 3$ else 0 | **R7** $= R3 + R6$ | **R8** $= R5 + R6$.

- BAGS defender: where x_g is the starting sample, x_t the last query, x_b the best adversarial so far, s_t the average step size between queries, $h \in [0,1]$ the last action of the defender, $z \in [0,1]$ the ℓ_2 distance of x_t and the last known adversarial query in embedding space, x: **R1** $= |\log(0.1g + \|x_g, x_b\|_{\ell_2})| \cdot 0.1$ | **R2** $= |\log_{10} s_t|$ | **R3** $= \frac{g}{\|x_g, x_t\|}$ | **R4** $= -\psi(x_t)$, where ψ is Eq. 4 | **R5** $= h - z$.
- HSJA defender: where x_{BS} are queries during the binary search: **R1** $= 1 - 2(\frac{\|x_g, x_b\|}{g})$ | **R2** $= h - z$ | **R3** $= R2 - 2\psi(x_{BS})$ | **R4** $= -\|\psi(x_{BS})\|$ | **R5** $= R2$ if $\psi(x_t)$ else $2 \cdot R2$.
- For both BAGS and HSJA defenders, the aforementioned are the rewards when x_t is adversarial; when it is benign, the reward is $R = 1 - h$ if the model responded correctly, otherwise $R = -1$.

TABLE V
INPUT TRANSFORMATIONS.

Input Transformations	**Magnitude**	**Probability**
Brightness & Contrast	0 – 0.5	0 – 1
Random Horizontal Flip	–	0 – 1
Random Vertical Flip	–	0 – 1
Sharpness	0.8 – 1.8	0 – 1
Perspective	0.25 – 0.5	0 – 1
Rotation	°0 – °180	0 – 1
Uniform Pixel Scale	0.8 – 1.2	0 – 1
Crop & Resize	0.6 – 1	0 – 1
Translation	-0.2 – 0.2	0 – 1

APPENDIX C
BASE RATE OF ATTACKS

In all the evaluations so far we use a fixed probability $P(adv) = 0.5$ that an incoming query is adversarial. To assess how our adaptive stateful defenses (Scenarios 4 & 6) perform in the complete absence of attacks, we evaluate them with $P(adv) = 0$ *without* retraining; the results are shown in Table VI. We observe a small reduction in the accuracy on clean samples that can be attributed to the considerably different base rate of adversarial and benign queries. Note however that as the probability of adversarial queries is an intrinsic property of each environment, if the base rate of attacks changes the defensive agents can be retrained to adjust to it.

TABLE VI
CLEAN ACCURACY ON CIFAR-10 FOR SCENARIOS 4 & 6, WHERE $P(adv)$ DENOTES THE PROBABILITY THAT A QUERY IS PART OF AN ATTACK.

Adv. Trained	$P(adv)$	BAGS4	BAGS6	HSJA4	HSJA6
✘	0.5	91.55	91.38	91.59	91.62
	0.0	90.91	90.95	90.48	90.86
✔	0.5	87.61	87.50	87.58	87.68
	0.0	87.02	87.11	86.70	86.98

APPENDIX D
MODELS & HYPERPARAMETERS

The image classification models we use are ResNet-20 for CIFAR-10 and a standard 2 convolutional / 2 fully-connected layer NN for MNIST. For adversarially training models, we follow the canonical approach as described in [5]: the model is trained for 20 epochs, where the first 10 are trained normally and the last 10 on batches containing additional adversarial examples generated with 40 steps of PGD. For learning the similarity space, that is the metric space where defensive agents control the radius of interception around which a query is adversarial or not, we use a Siamese CNN. This network is trained with contrastive loss, where dissimilar examples are generated by adding Gaussian noise and performing evasive transformations on the input from the list in Table V. For the PPO agents trained for each scenario, we use the open source library Stable-Baselines3 [3]. Policies are parameterized by a two fully-connected layer NN; the hyperparameter search space is shown in Table VII.

Blacklight & OARS. For evaluating Blacklight [19] and OARS [18] we use their default hyperparameters as those are provided in the publicly available implementations. In particular, as OARS spends 200 extra queries per episode to adapt the proposal distribution, we add those on top of b the evaluation budget. Additionally, as our defense is not rejection based, we replace the rejection decision with a non-adversarial one.

TABLE VII
HYPERPARAMETER RANGES DURING PPO TRAINING.

Hyperparameter	**BAGS**	**HSJA**
learning rate	3e-3 – 1e-4	3e-3 – 1e-4
episode steps	600 – 3000	1000 – 5000
total steps	1e5 – 4e5	2e4 – 2e5
total queries	25K – 100K	5K – 50K
batch size	32 – 128	32 – 64
buffer size	2048 – 2048	64 – 1024
epochs	20 – 20	20 – 20
gamma	0.85 – 0.99	0.9 – 0.99

[3]https://github.com/DLR-RM/stable-baselines3

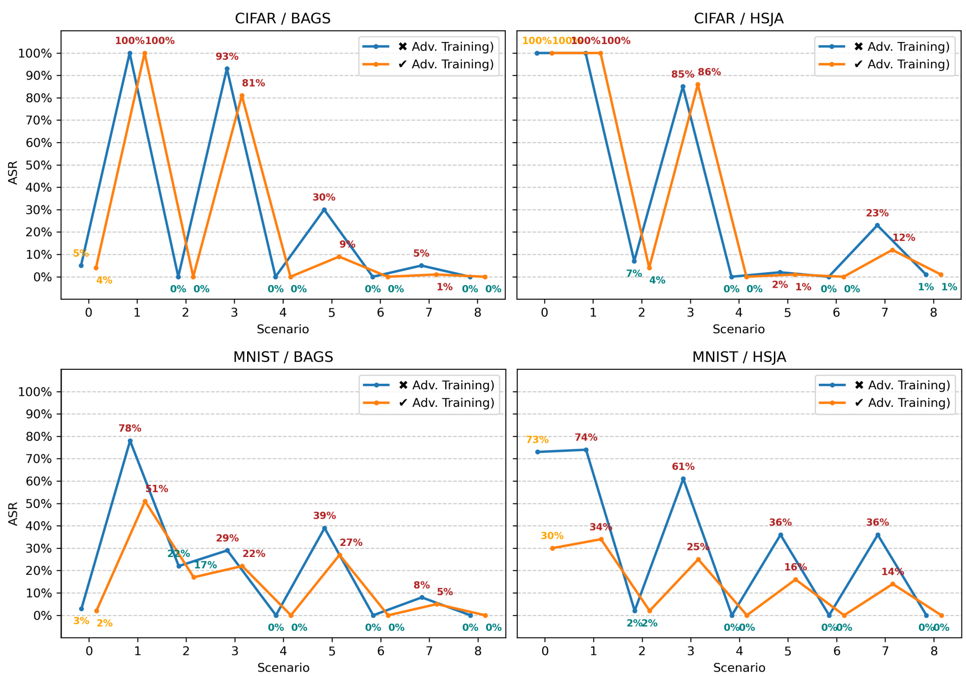

Fig. 4. Progression of ASR over successive adaptations. Red and green values in ASR denote offensive and defensive scenarios respectively.