# FROM CONFLICTS TO CONVERGENCE: A ZEROTH-ORDER METHOD FOR MULTI-OBJECTIVE LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Multi-objective learning (MOL) is a popular paradigm for learning problems under multiple criteria, where various dynamic weighting algorithms (e.g., MGDA and MODO) have been formulated to find an updated direction for avoiding conflicts among objectives. Recently, increasing endeavors have struggled to tackle the black-box MOL when the gradient information of objectives is unavailable or difficult to attain. Albeit the impressive success of zeroth-order method for single-objective black-box learning, the corresponding MOL algorithm and theoretical understanding are largely absent. Unlike single-objective problems, the errors of MOL introduced by zeroth-order gradients can simultaneously affect both the gradient estimation and the gradient coefficients $\lambda$, leading to further error amplification. To address this issue, we propose a Stochastic Zeroth-order Multiple Objective Descent algorithm (SZMOD), which leverages function evaluations to approximate gradients and develops a new decomposition strategy to handle the complicated black-box multi-objective optimization. Theoretically, we provide convergence and generalization guarantees for SZMOD in both general non-convex and strongly convex settings. Our results demonstrate that the proposed SZMOD enjoys a promising generalization bound of $\mathcal{O}(n^{-\frac{1}{2}})$, which is comparable to the existing results of first-order methods requiring additional gradient information. Experimental results validate our theoretical analysis.

## 1 INTRODUCTION

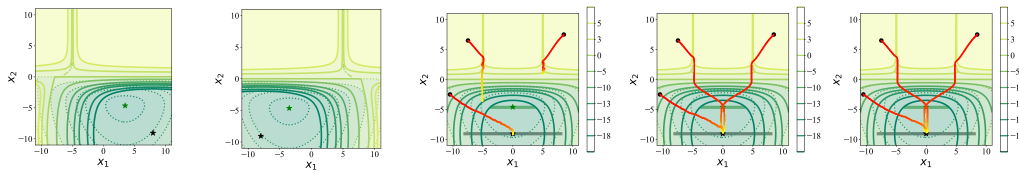
Multi-objective learning (MOL) aims to learn a single model that can optimize multiple potentially conflicting objectives simultaneously. An unconstrained multi-objective optimization problem can be defined as

$$\min_{x \in \mathbb{R}^d} F_S(x) := [f_{S,1}(x), \ldots, f_{S,M}(x)], \tag{1}$$

where $S = \{z_i\}_{i=1}^n$ is the training dataset, $f_{S,m}(x)$ is the $m$-th empirical objective for $m \in [M] =: \{1, 2, ...M\}$. Usually, we can set $f_{S,m}(x) = \sum_{i=1}^n f_{z_i,m}(x)$ as the empirical risk on the entire training dataset $S$, where $f_{z,m} : \mathbb{R}^d \mapsto \mathbb{R}$ measures the performance of a model $x \in \mathbb{R}^d$ on a datum $z$ for the $m$-th objective.

Multi-objective learning has gained increasing attention, due to the complex decision-making processes involved in many challenging tasks, e.g., managing traffic systems (Felten et al., 2024), electricity grids (Lu et al., 2022), and taxation policy design (Zheng et al., 2022). These burgeoning fields in practice, which require trading off multiple conflict objectives, underscore the significance of research in MOL. Specifically, balancing bias and variance (Neal et al., 2018), or accuracy and calibration (Guo et al., 2017), are well-known common objectives in machine learning that need to be optimized. To tackle these problems, this paper pays particular attention to multi-objective gradient methods that aim to find a common descent direction for all objectives. Désidéri (2012) initially introduced the concept of a Pareto stationary and the multi-gradient descent (MGDA) algorithm. Since then, stochastic variants such as MOCO (Fernando et al., 2023) and MODO (Chen et al., 2024) have been proposed. Those first-order multi-objective alpgrithms have have great performed in the white-box problem.

However, when we consider the black box problem, where obtaining explicit gradients is either unattainable or too expensive, these algorithms are no longer applicable. For instance, in the field



(a) Objective 1 (b) Objective 2 (c) Static (d) MODO (e) SZMOD

Figure 1: An example from (Liu et al., 2021) involves two objectives in Figure 1(a) and 1(b) to demonstrate the conflict between objectives. Figures 1(c)-1(e) show the optimization trajectories, where the black dots indicate the initialization points of the trajectories, with the colors transitioning from red (start) to yellow (end). The background solid/dotted contours represent the landscape of the average empirical and population objectives, respectively. The gray/green bars mark the empirical/population Pareto fronts, while the black ⋆ green ⋆ marks the solution to the average objectives.

of multiple-objective reinforcement learning (Hu et al., 2023; Felten et al., 2024; Terry et al., 2021; Gupta et al., 2017), agents often can only learn strategies through interaction and external reward signals, without access to the internal state or dynamics of the environment. Similarly, in most attack scenarios (Akhtar & Mian, 2018; Liu et al., 2022; Papernot et al., 2017; 2016), the attacker's knowledge of the classifier is very limited, which causes the attacker only to execute a black-box attack. Liang et al. (2022) state that the black-box attacks can manipulate model outputs by adjusting the trade-offs between true and false positives without direct access to the model's internals. Williams & Li (2023) consider a novel multi-objective sparse attack that can simultaneously reduce the number and the individual size of modified pixels during the attack process.

Most of the black-box MOL scenarios discussed above are traditionally optimized using the hypervolume indicator (Felten et al., 2024) as the standard performance metric and are typically solved using methods such as evolutionary algorithms (Zhou et al., 2024; Mathai et al., 2020; Liu et al., 2024). Unfortunately, these methods impose strict constraints on problem dimensionality. In contrast, zeroth-order (ZO) optimization algorithms demonstrate greater versatility in handling higher-dimensional problems and can achieve excellent performance, often comparable to or even surpassing that of white-box models where gradients are explicitly available. (Sun et al., 2022; Papernot et al., 2017). Unfortunately, there has been no endeavor to apply the zeroth-order optimization to multi-objective optimization.

To fill this gap, we present the Stochastic Zeroth-order Multiple Objective Descent algorithm (SZ-MOD), which integrates coordinate-based zeroth-order gradient estimations and employs a consistent directional selection strategy during the $\lambda$ iteration process. Specifically, by using the same direction for gradient approximation throughout the iterations, SZMOD ensures that the update direction of the dynamic weigh $\lambda_t$ is updated in alignment with the chosen direction, thereby maintaining stability and reducing variance in the optimization process. Combining coordinate zeroth-order techniques and unified directional updates enhances the algorithm's ability to effectively address black-box multi-objective learning problems.

- **Gradient Direction Conflict:** In first-order multi-objective optimization algorithms, the gradients of multiple objective functions are computed to determine a suitable direction for optimization. However, in zeroth-order multi-objective problems, we rely on zeroth-order gradient estimates, where the direction estimation depends entirely on a random vector $u$ (determined by the zeroth-order estimation process). This dependence makes it challenging to identify an appropriate CA direction (the proper direction to update $\lambda$, will defined in section 2.4), complicating the optimization process.

- **Excessive Error Risk:** Zeroth-order gradient estimation inherently introduces errors, which also propagate into the iterative updates of $\lambda$. These compounded errors affect the term of the CA direction, increasing the risk of divergence during the iteration of $x$. Therefore, it is crucial to control these errors effectively to ensure convergence and maintain the stability of the optimization process.

## 2 PRELIMINARIES

In this section, we first introduce MOL's problem formulation, the analysis target, and the metric to measure its optimization, generalization, and CA direction.

### 2.1 NOTATION

Denote the vector-valued objective function on datum $z$ as $F_z(x) = [f_{z,1}(x), \ldots, f_{z,M}(x)]$. The training and testing performance of $x$ can then be measured by the empirical objective $F_S(x)$ and the population objective $F(x)$ which are, respectively, defined as $F_S(x) := \frac{1}{n}\sum_{i=1}^{n} F_{z_i}(x)$ and $F(x) := \mathbb{E}_{z\sim\mathcal{D}}[F_z(x)]$. Their corresponding gradients are denoted as $\nabla F_S(x)$ and $\nabla F(x) \in \mathbb{R}^{d\times M}$.

### 2.2 METHOD OF MOL

Analogous to the stationary solution and optimal solution in single-objective learning, we define the Pareto stationary point and Pareto optimal solution for MOL problem $\min_{x\in\mathbb{R}^d} F(x)$ as follows.

**Definition 1** (Pareto stationary and Pareto optimal). *If there exists a convex combination of the gradient vectors that equals to zero, i.e., there exists $\lambda \in \Delta^M$ such that $\nabla F(x)\lambda = 0$, then $x \in \mathbb{R}^d$ is Pareto stationary. If there is no $x \in \mathbb{R}^d$ and $x \neq x^*$ such that, for all $m \in [M] f_m(x) \leq f_m(x^*)$, with $f_{m'}(x) < f_{m'}(x^*)$ for at least one $m' \in [M]$, then $x^*$ is Pareto optimal. If there is no $x \in \mathbb{R}^d$ such that for all $m \in [M]$, $f_m(x) < f_m(x^*)$, then $x^*$ is weakly Pareto optimal.*

By definition, at a Pareto stationary solution, there is no common descent direction for all objectives. A necessary and sufficient condition for $x$ being Pareto stationary for smooth objectives is that $\min_{\lambda\in\Delta^M} \|\nabla F(x)\lambda\| = 0$. Therefore, $\min_{\lambda\in\Delta^M} \|\nabla F(x)\lambda\|$ can be used as a measure of Pareto stationarity (PS). We will refer to the aforementioned quantity as the PS population risk henceforth and its empirical version as PS empirical risk or PS optimization error. We next introduce the target of our analysis based on the above definitions.

### 2.3 ZEROTH-ORDER GRADIENT ESTIMZATION

**Coordinate-wise Gradient Estimation.** When only function evaluations are available, here, we employ the deterministic coordinate-wise direction to derive the decent direction. Specifically, for the smoothing constant $v$ and vector $u_i$($u_i$ represents the unit vector where the i-th element is 1 and the remaining elements are 0), the directional derivative of $f_{z,m}$ in the direction $u$ for the smooth function $f_i, i \in [n]$, can be estimated as:

$$\hat{\nabla} f_{z,m}(x, u, v) = \sum_{j=1}^{d} \frac{f_{z,m}(x + vu_j) - f_{z,m}(x)}{v} u_j. \tag{2}$$

as the approximation of the full directional gradient. Since the smoothing constant $v$ is fixed, for simplicity, we leave out $v$ in these gradient estimations and set

$$\hat{\nabla} f_{z,m}(x, u) := \hat{\nabla} f_{z,m}(x, u, v). \tag{3}$$

Denote the vector-valued objective function on datum $z$ as $F_z(x) = [f_{z,1}(x), \ldots, f_{z,M}(x)]$. The training and testing performance of $x$ can then be measured by the empirical objective $F_S(x)$ and the population objective $F(x)$ which are, respectively, defined as $F_S(x) := \frac{1}{n}\sum_{i=1}^{n} F_{z_i}(x)$ and $F(x) := \mathbb{E}_{z\sim\mathcal{D}}[F_z(x)]$. Their corresponding estimate gradients are denoted as $\hat{\nabla} F_S(x)$ and $\hat{\nabla} F(x) \in \mathbb{R}^{d\times M}$. Thus the zeroth-order estimate for all objectives on datum $z$ should be written as $\hat{\nabla} F_z(x) = \left[\hat{\nabla} f_{z,1}(x), \ldots, \hat{\nabla} f_{z,M}(x)\right]$.

### 2.4 PROBLEM SETUP

**Proposition 1** ((Tanabe et al., 2019) Lemma 2.2). . *If $f_m(x)$ are convex or strongly convex for all $m \in [M]$, and $x \in \mathbb{R}^d$ is a Pareto stationary point of $F(x)$, then $x$ is weakly Pareto optimal or Pareto optimal.*

Next, we proceed to decompose the PS population risk.

**Error Decomposition.** Given a model $x$, the PS population risk can be decomposed into

$$\underbrace{\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\|}_{\text{PS population risk } R_{\text{pop}}(x)} = \underbrace{\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\| - \min_{\lambda\in\Delta^M}\|\nabla F_S(x)\lambda\|}_{\text{PS generalization error } R_{\text{gen}}(x)} + \underbrace{\min_{\lambda\in\Delta^M}\|\nabla F_S(x)\lambda\|}_{\text{PS optimization error } R_{\text{opt}}(x)}, \quad (4)$$

where the optimization error quantifies the training performance, i.e., how well does model $x$ perform on the training data; and the generalization error (gap) quantifies the difference between the testing performance on new data sampled from $\mathcal{D}$ and the training performance, i.e., how well the model $x$ performs on unseen testing data compared to the training data.

The zeroth-order optimization is a gradient-based black-box optimization that utilizes the difference information of function values to approximate the true gradient. Furthermore, this method does not alter the optimization objective, only the optimization process differs from the first-order one. As for MOL black-box problems, the optimization objective of the SZMOD remains $\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\| = 0$.

Let $A : \mathcal{Z}^n \mapsto \mathbb{R}^d$ denote a randomized MOL algorithm. Given training data $S$, we are interested in the expected performance of the output model $x = A(S)$, which is measured by $\mathbb{E}_{A,S}\left[R_{\text{pop}}(A(S))\right]$. From equation 4 and linearity of expectation, it holds that

$$\mathbb{E}_{A,S}\left[R_{\text{pop}}(A(S))\right] = \mathbb{E}_{A,S}\left[R_{\text{gen}}(A(S))\right] + \mathbb{E}_{A,S}\left[R_{\text{opt}}(A(S))\right]. \quad (5)$$

**Distance to CA direction.** Consider an update direction $d = -\nabla F_S(x)\lambda$, where $\lambda$ is the dynamic weights from a simplex $\lambda \in \Delta^M := \left\{\lambda \in \mathbb{R}^M \mid \mathbf{1}^\top\lambda = 1, \lambda \geq 0\right\}$. To obtain such a steepest CA direction in unconstrained learning that maximizes the minimum descent of all objectives, we can solve the following problem (Fliege et al., 2019)

$$\text{CA direction} \quad d(x) = \arg\min_{d\in\mathbb{R}^d}\max_{m\in[M]}\left\{\langle\nabla f_{S,m}(x), d\rangle + \frac{1}{2}\|d\|^2\right\} \quad (6)$$

$$\overset{\text{equivalent to}}{\Longleftrightarrow} \; d(x) = -\nabla F_S(x)\lambda^*(x) \text{ s.t. } \lambda^*(x) \in \arg\min_{\lambda\in\Delta^M}\|\nabla F_S(x)\lambda\|^2. \quad (7)$$

Defining $d_\lambda(x) = -\nabla F_S(x)\lambda$ given $x \in \mathbb{R}^d$ and $\lambda \in \Delta^M$, we measure the distance to $d(x)$ via (Fernando et al., 2023)

$$\text{CA direction error} \quad \mathcal{E}_{\text{ca}}(x,\lambda) := \|d_\lambda(x) - d(x)\|^2. \quad (8)$$

With the above definitions of measures that quantify the performance of algorithms in different aspects, we then introduce a stochastic gradient algorithm for MOL that is analyzed in this work.

## 3 A STOCHASTIC ALGORITHM FOR BLACK-BOX MOL

In this section, we first introduce our main algorithm, Stochastic Zeroth-order Multiple Objective Descent (SZMOD).

At each iteration $t$, $\alpha_t, \gamma_t$ are step sizes, and $\Pi_{\Delta^M}(\cdot)$ denotes Euclidean projection to the simplex $\Delta^M$. Denoting $z_{t,s}$ as an independent sample from $S$ with $s \in [3]$, and $\hat{\nabla} F_{z_{t,s}}(x_t)$ as the gradient estimate of $\nabla F_{z_{t,s}}(x_t)$.

**Remark 1.** *In the iteration process of $\lambda_t$, gradient direction conflicts prevent us from achieving convergence. To ensure the algorithm converges, SZMOD requires that $\hat{\nabla} f_{z,1}(x)$ and $\hat{\nabla} f_{z,2}(x)$ use the same stochastic direction. By this method, we have*

$$\mathbb{E}_{z_{t,1},z_{t,2}}\left[\hat{\nabla} F_{z_{t,1}}(x_t)^\top \hat{\nabla} F_{z_{t,2}}(x_t)\lambda_t\right] = \nabla F_S(x_t)^\top \nabla F_S(x_t)\lambda_t + \mathcal{O}(v),$$

*which means that we can stabilize the updates and control the error through $v$.*

**Algorithm 1** Stochastic Zeroth-order Multiple Objective Descent (SZMOD)

**Input:** Training data $S$, initial model $x_0$, weighting co- efficient $\lambda_0$, and their learning rates $\{\alpha_t\}_{t=0}^{T}, \{\gamma_t\}_{t=0}^{T}$.

**Output:** $x_T$

1: **for** $t = 0, \dots, T-1$ **do**
2: **for** $m = 1, \dots, M$ **do**
3: Compute zeroth-order gradients $\hat{\nabla} f_{m,z_{t,s}}(x_t)$ using same $\boldsymbol{u}$, $s \in [2]$
4: Compute zeroth-order gradients $\hat{\nabla} f_{m,z_{t,3}}(x_t)$ with coordinate
5: **end for**
6: Compute dynamic weight $\lambda_{t+1}$ following
7: Compute $\lambda_{t+1} = \Pi_{\Delta^M}\left(\lambda_t - \gamma_t \hat{\nabla} F_{z_{t,1}}(x_t)^\top \hat{\nabla} F_{z_{t,2}}(x_t)\lambda_t\right)$
8: Compute $x_{t+1} = x_t - \alpha_t \hat{\nabla} F_{z_{t,3}}(x_t)\lambda_{t+1}$
9: **end for**

In the iteration process of $x_t$, the zeroth-order method will also lead to excessive error risk, which is caused by the error of $\lambda_{t+1}$ and $\hat{\nabla} F_{z,3}$. The error of $\lambda_{t+1}$ can be control by remark 1. Here, we choose to use the coordinate zeroth-order estimate to minimize the error of $\hat{\nabla} F_{z,3}$.

# 4 Optimization of SZMOD

In this section, we bound the multi-objective PS optimization error $\min_{\lambda \in \Delta^M} \|\nabla F_S(x)\lambda\|$ (Fernando et al., 2023; Fliege et al., 2019; Désidéri, 2012). As discussed in Section 2.2, this measure being zero implies the model $x$ achieves a Pareto stationarity for the empirical problem.

Below, we list the standard assumptions used to derive the optimization error, which has been widely used for theoretical analysis for (Chen et al., 2024; Lei, 2023; Fliege et al., 2019).

**Assumption 1** (Lipschitz continuity of $F_z(x)$ ). *For all $m \in [M]$, $f_{z,m}(x)$ are $\ell_f$-Lipschitz continuous for all $z$. Then $F_z(x)$ are $\ell_F$-Lipschitz continuous in Frobenius norm for all $z$ with $\ell_F = \sqrt{M}\ell_f$.*

**Assumption 2** (Lipschitz continuity of $\nabla F_z(x)$). *For all $m \in [M], \nabla f_{z,m}(x)$ is $\ell_{f,1}$-Lipschitz continuous for all $z$. And $\nabla F_z(x)$ is $\ell_{F,1}$-Lipschitz continuous in Frobenius norm for all $z$.*

**Assumption 3.** *For all $m \in [M], z \in \mathcal{Z}, f_{z,m}(x)$ is $\mu$-strongly convex w.r.t. $x$ with $\mu > 0$.*

Note that in the strongly convex case, the gradient norm $\|\nabla F_z(x)\|_{\mathrm{F}}$ can be unbounded in $\mathbb{R}^d$. Therefore, one cannot assume Lipschitz continuity of $f_{z,m}(x)$ w.r.t. $x \in \mathbb{R}^d$. We address this challenge by showing that $\{x_t\}$ generated by the SZMOD algorithm is bounded as stated in Lemma 1. Notably, combined with Assumption 1, we can derive that the gradient norm $\|\nabla F_z(x_t)\|_{\mathrm{F}}$ is also bounded.

**Lemma 1** (Boundedness of $x_t$ for strongly convex and smooth objectives). *Suppose Assumptions 2, 3 hold. For $\{x_t\}, t \in [T]$ generated by SZMOD algorithm or other dynamic weighting algorithm with weight $\lambda \in \Delta^M$, step size $\alpha_t = \alpha$, and $0 \le \alpha \le \ell_{f,1}^{-1}$, there exists a finite positive constant $c_x$ such that $\|x_t\| \le c_x$. And there exists finite positive constants $\ell_f, \ell_F = \sqrt{M}\ell_f$, such that for all $\lambda \in \Delta^M$, we have $\|\nabla F(x_t)\lambda\| \le \ell_f$, $\|\nabla F(x_t)\|_F \le \ell_F$.*

## 4.1 Distance to CA direction

**Theorem 1** (Distance to CA direction). *Suppose either: 1) Assumptions 1, 3 hold; or 2) Assumptions 1, 2 hold, with $\ell_f$ and $\ell_F$ defined in Lemma 1. Consider $\{x_t\}, \{\lambda_t\}$ generated by the SZMOD algorithm. For all $\lambda \in \Delta^M$, it holds that:*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_A\left[\|d_{\lambda_t}(x_t) - d(x_t)\|^2\right] \le \frac{4}{\gamma T} + 6\sqrt{M\ell_{f,1}\ell_f^2\frac{\alpha}{\gamma}} + \gamma M \ell_f^4 + e \tag{9}$$

*Here $e = \frac{l_{f,1}^2 v^2 d}{4}\mathbb{E}_A\|\lambda_t - \lambda\|_1 + \frac{l_{f,1}v}{2}\mathbb{E}_A(\|\lambda_t - \lambda\|_1\|\nabla F_S\lambda\|_1 + d\|\nabla F_S(\lambda_t - \lambda)\|_1)$ caused by zeroth-order error. We should mention that $e$ can be seen as $\mathcal{O}(v)$. Analyzing convergence to* *the CA direction using the measure introduced in Section 2.4. By, e.g., choosing* $\alpha = \Theta\left(T^{-\frac{3}{4}}\right)$, $\gamma = \Theta\left(T^{-\frac{1}{4}}\right)$ *and* $v = \gamma/10$*, the RHS of equation 9 converges in a rate of* $\mathcal{O}\left(T^{-\frac{1}{4}}\right)$.

### 4.2 PS OPTIMIZATION ERROR

**Theorem 2.** *(PS optimization error of SZMOD). Suppose either 1) Assumptions 1, 3 hold or 2) Assumptions 1, 2 hold, with* $\ell_f$ *defined in Lemma 1. Define* $c_F$ *such that* $\mathbb{E}_A\left[F_S\left(x_0\right)\lambda_0\right] - \min_{x\in\mathbb{R}^d}\mathbb{E}_A\left[F_S(x)\lambda_0\right] \leq c_F$*. Considering* $\{x_t\}$ *generated by SZMOD (Algorithm 1), with* $\alpha_t = \alpha \leq 1/\left(2\ell_{f,1}\right), \gamma_t = \gamma$*, then under either condition 1) or 2), it holds that*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}_A\left[\min_{\lambda\in\Delta^M}\left\|\nabla F_S\left(x_t\right)\lambda\right\|\right] \leq \sqrt{\frac{c_F}{\alpha T}} + \sqrt{\frac{3}{2}\gamma M\ell_f^4} + \sqrt{\frac{1}{2}\alpha\ell_{f,1}\ell_{f,d}^2} + e. \tag{10}$$

The choice of step sizes $\alpha = \Theta(T^{-\frac{3}{4}})$, $\gamma = \Theta(T^{-\frac{1}{4}})$, and smoothing constant $v = \gamma/10$ to ensure convergence to CA direction is suboptimal for the convergence to Pareto stationarity. Then the RHS of equation 10 converges in a rate of $\mathcal{O}\left(T^{-\frac{1}{8}}\right)$.

## 5 GENERALIZATION OF SZMOD

In the following, we provide uniform stability for the black-box MOL algorithm, whose expected PS generalization error can be further bounded under several convexity scenarios.

**Proposition 2** ((Chen et al., 2024), Proposition 2)**.** *With* $\|\cdot\|_{\mathrm{F}}$ *denoting the Frobenius norm,* $R_{\mathrm{gen}}(A(S))$ *in (2.2) can be bounded by*

$$\mathbb{E}_{A,S}\left[R_{gen}\left(A(S)\right)\right] \leq \mathbb{E}_{A,S}\left[\|\nabla F(A(S)) - \nabla F_S(A(S))\|_{\mathrm{F}}\right]. \tag{11}$$

With Proposition 2, we introduce the concept of MOL uniform stability tailored for MOL problems. Then, we analyze their bounds in the general nonconvex and strongly convex cases, respectively.

**Definition 2** (MOL uniform stability)**.** *A randomized algorithm* $A : \mathcal{Z}^n \mapsto \mathbb{R}^d$*, is MOL-uniformly stable with* $\epsilon_{\mathrm{F}}$ *iffor all neighboring datasets* $S, S'$ *that differ in at most one sample, we have*

$$\sup_z \mathbb{E}_A\left[\|\nabla F_z(A(S)) - \nabla F_z\left(A\left(S'\right)\right)\|_{\mathrm{F}}^2\right] \leq \epsilon_{\mathrm{F}}^2.$$

Next, we show the relation between the upper bound of PS generalization error in 4 and MOL uniform stability in Proposition 3.

**Proposition 3** ((Chen et al., 2024), proposition 3)**.** *Assume for any z, the function* $F_z(x)$ *is differentiable. If a randomized algorithm* $A : \mathcal{Z}^n \mapsto \mathbb{R}^d$ *is MOL-uniformly stable with* $\epsilon_{\mathrm{F}}$*, then*

$$\mathbb{E}_{A,S}\left[\|\nabla F(A(S)) - \nabla F_S(A(S))\|_{\mathrm{F}}\right] \leq 4\epsilon_{\mathrm{F}} + \sqrt{n^{-1}\mathbb{E}_S\left[\mathbb{V}_{z\sim\mathcal{D}}\left(\nabla F_z(A(S))\right)\right]}. \tag{12}$$

where $\mathbb{V}_{z\sim\mathcal{D}}\left(\nabla F_z(A(S))\right) = \mathbb{E}_{z\sim\mathcal{D}}\left[\|\nabla F_z(A(S)) - \mathbb{E}_{z\sim\mathcal{D}}\left[\nabla F_z(A(S))\right]\|_{\mathrm{F}}^2\right]$ is the variacne.

Proposition 3 establishes a connection between the upper bound of the PS generalization error and the MOL uniform stability.

**Theorem 3** (PS generalization error of SZMOD in nonconvex case)**.** *If* $\sup_z \mathbb{E}_A\left[\|\nabla F_z(A(S))\|_{\mathrm{F}}^2\right] \leq G^2$ *for any S, then the MOL uniform stability, i.e.,* $\epsilon_{\mathrm{F}}^2$ *in Definition 2 is bounded by* $\epsilon_{\mathrm{F}}^2 \leq 4G^2T/n$*. And the PS generalization error* $\mathbb{E}_{A,S}\left[R_{gen}\left(A(S)\right)\right] = \mathcal{O}\left(T^{\frac{1}{2}}n^{-\frac{1}{2}}\right)$.

**Remark 2.** *The proof process of non-convex generalization does not involve parameter updates. Therefore, zeroth-order gradient approximation does not affect the generalization results. At this point, the generalization results of the first-order and zeroth-order methods are naturally the same.*

With Lemma 1 and Lemma **??**, the stability bound and PS generalization is provided below.

**Theorem 4** (PS generalization error of in strongly convex case)**.** *Suppose Assumptions 2 and 3 hold. Let $A$ be the SZMOD algorithm (Algorithm 1). For the MOL uniform stability $\epsilon_F$ of algorithm $A$ in Definition 2, if the step sizes satisfy $0 < \alpha_t \leq \alpha \leq 1/\left(2\ell_{f,1}\right)$, $0 < \gamma_t \leq \gamma \leq \min\left\{\frac{\mu^2}{484\ell_{f,d}^2\ell_{g,1}}, \frac{1}{8\left(3\ell_{f,d}^2+2\ell_{g,1}\right)}\right\}/T$, and smooth constant $v \leq \min\left\{\frac{1}{nd}, \frac{1}{nd\left(2\ell_{g,1}+\ell_{g,1}^2\right)}\right\}$ then it holds that*

$$\epsilon_{\mathrm{F}}^2 \leq \frac{48}{\mu n}\ell_{f,d}^2\ell_{F,1}^2\left(\alpha+\frac{12+4M\ell_{f,d}^2}{\mu n}+\frac{10M\ell_f^4\gamma}{\mu}\right)+\frac{4}{\mu n}\ell_{F,1}^2\left(\frac{10\alpha M\ell_{f,d}^2\gamma+\mu\gamma}{\mu\alpha}+\alpha\ell_{f,1}+\frac{2\alpha\ell_{f,1}^2}{n}\right). \tag{13}$$

*and* $\quad \mathbb{E}_{A,S}\left[R_{\mathrm{gen}}(A(S))\right] = \mathcal{O}\left(n^{-\frac{1}{2}}\right).$

**Remark 3.** *Theorem 3, 4 implies setting proper step sizes for different convexity helps to improve the generalization. Under strong convexity conditions, the proof process involving parameter updates will inevitably introduce the cumulative error brought by zeroth-order estimation. We must constrain the smoothness parameter $v$ to achieve the same generalization convergence rate as the first-order method.*

## 6 Connection between Optimization, conflict avoidance and generalization

In this section, we combine the proof process and theoretical results on optimization error, generalization bounds, and the distance to the CA direction to discuss the impact of introducing zeroth-order gradient approximations on multi-objective algorithms. Summarizing the findings from Sections 4 and 5, we derive the PS population risk. With $A_t(S) = x_t$ denoting the output of algorithm $A$ at the $t$-th iteration, we can decompose the PS population risk $R_{\text{pop}}\left(A_t(S)\right)$ as (cf. equation 4,equation 11)

$$\mathbb{E}_{A,S}\left[R_{\text{pop}}\left(A_t(S)\right)\right] \leq \mathbb{E}_{A,S}\left[\min_{\lambda\in\Delta^M}\left\|\nabla F_S\left(A_t(S)\right)\lambda\right\|\right]+\mathbb{E}_{A,S}\left[\left\|\nabla F\left(A_t(S)\right)-\nabla F_S\left(A_t(S)\right)\right\|_{\mathrm{F}}\right]$$

**Theorem 5** (The general nonconvex case)**.** *Suppose Assumptions 1, 2 hold. By the optimization error in Theorem 2 and the generalization error bound in Theorem 3, the PS population risk of the output of SZMOD can be bounded by*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}_{A,S}\left[R_{\text{pop}}\left(A_t(S)\right)\right] = \mathcal{O}\left(\alpha^{-\frac{1}{2}}T^{-\frac{1}{2}}+\alpha^{\frac{1}{2}}+\gamma^{\frac{1}{2}}+T^{\frac{1}{2}}n^{-\frac{1}{2}}\right)+\mathcal{O}\left(v\right).$$

**Remark 4.** *By selecting step sizes of $\alpha = \Theta\left(T^{-\frac{1}{2}}\right)$ and $\gamma = \Theta\left(T^{-\frac{1}{2}}\right)$, with the number of steps $T = \Theta\left(n^{\frac{2}{3}}\right)$, we can choose a smoothing parameter of $v = \Theta\left(n^{-\frac{1}{6}}\right)$, which effectively limits the impact of the zeroth-order approximation on optimization convergence. Under these conditions, the expected PS population risk is $\mathcal{O}\left(n^{-\frac{1}{6}}\right)$.*

**Theorem 6** (The strongly convex case)**.** *Suppose Assumptions 2, 3 hold. By the optimization error and the generalization error given in Theorems 2 and 4, SZMOD's PS population risk can be bounded by*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}_{A,S}\left[R_{\text{pop}}\left(A_t(S)\right)\right] = \mathcal{O}\left(\alpha^{-\frac{1}{2}}T^{-\frac{1}{2}}+\alpha^{\frac{1}{2}}+\gamma^{\frac{1}{2}}+n^{-\frac{1}{2}}\right)+\mathcal{O}\left(v\right).$$

**Remark 5.** *Choosing step sizes $\alpha = \Theta\left(T^{-\frac{1}{2}}\right), \gamma = o\left(T^{-1}\right)$. Under strongly convex and smooth conditions, generalization analysis requires smoothing parameter size of $v = \Theta\left((nd)^{-1}\right)$. And* *number of steps $T = \Theta\left(n^2\right)$. We have the expected PS population risk in gradients is $\mathcal{O}\left(n^{-\frac{1}{2}}\right)$, aligning with the upper bound for the PS population risk in general nonconvex first-order methods as shown in Chen et al. (2024).*

**Zeroth-order method demonstrates the connection between optimization, conflict avoidance, and generalization.**

The core of the SZMOD algorithm lies in its dynamic weighting mechanism, which uses approximate gradient information to update $\lambda$. A high-quality $\lambda$ is essential for balancing conflicts among multiple objectives. The distance to the CA direction is a critical metric for assessing the quality of these updates and plays a pivotal role in ensuring algorithmic convergence. In SZMOD, the deviation from the CA direction arises from the data and limited iterations and the cumulative error $e$ introduced by the zeroth-order method. This CA direction error transfers the cumulative error $e$ into an optimization error. Theoretical results indicate that in corresponding first-order algorithms, the relationship between CA direction error and optimization error is not as inherently inheritable and may exhibit a degree of antagonism (Chen et al., 2024). Thus, zeroth-order optimization opens a window into understanding the interaction between CA direction and optimization. Due to the propagation of cumulative error, optimization error imposes constraints on the smooth parameter $v$ to ensure convergence. Furthermore, under strongly convex and smooth conditions, achieving generalization depends on controlling the size of $v$. Therefore, determining the appropriate value of $v$ requires balancing the demands of both generalization and optimization.

# 7 Empirical Validation

In this section, we systematically evaluate the performance of our proposed SZMOD algorithm on toy examples and CIFAR-10 datasets. The experiments are designed to mimic a variety of multi-objective landscapes with adjustable complexity levels. We employ synthetic datasets and realistic image data that encapsulate the essential characteristics of multi-objective problems for evaluating the optimization accuracy, generalization capability, conflict avoidance, and convergence performance of our proposal SZMOD algorithm.

## 7.1 Synthetic Experiment

In the following content, we explore the subtleties of the SZMOD algorithm's efficacy across a spectrum of hyperparameters, particularly emphasizing the trade-offs between optimization, generalization capabilities, and the mitigation of conflicting objectives. The synthetic experiments have been meticulously crafted to emulate a multi-objective optimization context, which successfully evaluates the influence exerted by diverse hyperparameters.

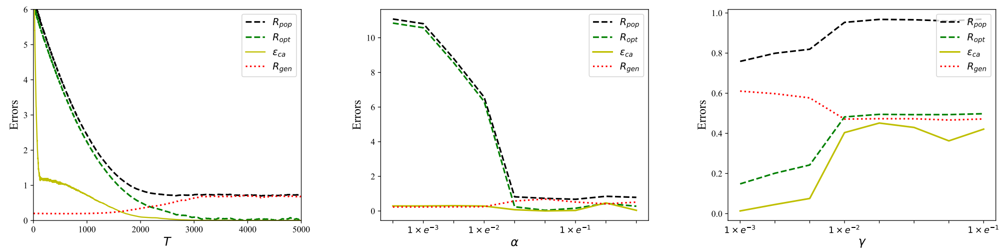
**Strongly Convex Scenario:** Inspired by (Chen et al., 2024), the following formulation is exploited to generate the MOL examples, whose $m$-th objective function is

$$f_{z,m}(x) = \frac{1}{2} b_{1,m} x^\top A x - b_{2,m} z^\top x,$$

where $b_{1,m} > 0$ for all $m \in [M]$, and $b_{2,m}$ is another scalar. We set $M = 3, b_1 = [b_{1,1}; b_{1,2}; b_{1,3}] = [1; 2; 1]$, and $b_2 = [b_{2,1}; b_{2,2}; b_{2,3}] = [1; 3; 2]$. Each experimental setting has been repeated ten times, where the average results with standard deviation information are recorded in Figure 7.1. The detailed experimental settings for nonconvex cases are left in Appendix A.

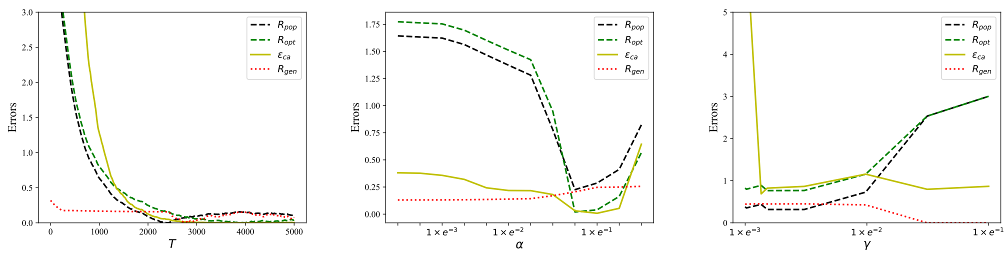
The number of iterations, $T$, plays a pivotal role in the convergence properties of the SZMOD algorithm. As depicted in Figure 2a, we maintain $\alpha = 0.05$ and $\gamma = 0.001$ while varying $T$. The results indicate that an increase in $T$ brings a decrease in both the optimization error and the distance to the conflict-avoidant (CA) direction, aligning with our theoretical predictions in Theorem 1, 2. This observation underscores the importance of sufficient training duration to achieve optimal solutions in multi-objective landscapes.

The step size for model parameters, $\alpha$, is another critical hyperparameter that influences the algorithm's ability to navigate the multi-objective space. In Figure 2b, we fix $T = 500$ and $\gamma = 0.001$ while adjusting $\alpha$. The findings reveal an initial decrease in the optimization error as $\alpha$ increases, while further enlarging $\alpha$ does not yield significant improvements. This non-linear relationship




(a) Number of iterations T. (b) Different step size (c) Different step size

Figure 2: Optimization, generalization, and CA direction errors of SZMOD in the strongly convex case under different $T, \alpha, \gamma$. The default parameters are $T = 500, \alpha = 0.05, \gamma = 0.001, v = 0.0001$.




(a) Number of iterations T. (b) Different step size (c) Different step size

Figure 3: Optimization, generalization, and CA direction errors of SZMOD in the nonconvex case for MNIST image classification under different $T, \alpha, \gamma$. The default parameters are $T = 500, \alpha = 0.05, \gamma = 0.001, v = 0.0001$.

between $\alpha$ and the optimization error highlights the need to carefully tune this hyperparameter to balance rapid convergence and potential overshooting of optimal solutions.

The weight step size, $\gamma$, is a unique aspect of SZMOD, controlling the update pace of the weighting parameters. In Figure 2c, with $T = 500$ and $\alpha = 0.05$, one can observe that the increasing $\gamma$ leads to a decrease in the distance to the CA direction, suggesting that a more aggressive update of weights can be beneficial for navigating conflicting objectives. However, too large $\gamma$ might lead to instability in convergence, indicating a delicate balance is required to harness the full potential of dynamic weighting.

The synthetic experiments provide valuable insights into the role of hyperparameters in shaping the trade-offs between optimization, generalization, and conflict avoidance in multi-objective learning. By systematically varying $T$, $\alpha$, and $\gamma$, we have demonstrated the nuanced interplay between these parameters and their impact on the algorithm's performance. These findings serve as a foundation for developing more sophisticated hyperparameter tuning strategies and provide empirical evidence to support theoretical analyses presented in prior sections. It is worth noting that, unlike the first-order MODO algorithm, the trends of $R_{opt}(\gamma)$ and are not always opposite. This is due to the error caused by $\varepsilon_c a(\gamma)$, which is related to $\gamma$. When the trends are aligned, the graph of $R_{opt}(\gamma)$ always shows similar changes after changes occur in the graph of $\varepsilon_c a(\gamma)$. This is precisely due to error propagation, which nicely validates our theory.

### 7.2 Attack Emperiment on CIFAR-10

Adversarial attacks trick machine learning models by adding carefully designed subtle perturbations to inputs, leading to mispredictions. Black-box adversarial attacks occur when attackers can't access a model's internals and must deduce its behavior from inputs and outputs. The Black-box attack method is closer to real-world attack scenarios. Therefore, we consider a multi-objection adversarial attack.

Table 1: Results for muti-objection black-box adverbial attacks

| model | Pixel ratio | ASR | L0_avg | L2_avg | AST_avg | SSIM_avg |
|---|---|---|---|---|---|---|
| CNN | 2% | 0.99 | 0.019 | 357.87 | 13.98 | 0.9 |
| CNN | 5% | 0.98 | 0.049 | 572.78 | 8.47 | 0.78 |
| CNN | 10% | 0.98 | 0.097 | 746.87 | 7.18 | 0.65 |
| VGG16 | 2% | 0.99 | 0.02 | 25.92 | 2.46 | 0.92 |
| VGG16 | 5% | 0.98 | 0.049 | 40.23 | 3.52 | 0.82 |
| VGG16 | 10% | 1 | 0.097 | 477.15 | 2.3 | 0.64 |
| Alexnet | 2% | 0.99 | 0.019 | 250.94 | 7.09 | 0.85 |
| Alexnet | 5% | 1 | 0.049 | 394.19 | 7.75 | 0.71 |
| Alexnet | 10% | 1 | 0.097 | 342.58 | 4.8 | 0.62 |
| Densenet | 2% | 0.91 | 0.019 | 22.71 | 10.7 | 0.88 |
| Densenet | 5% | 0.92 | 0.049 | 18.26 | 13.98 | 0.83 |
| Densenet | 10% | 0.86 | 0.097 | 12.22 | 13.18 | 0.87 |
| Res-net18 | 2% | 0.99 | 0.019 | 6.81 | 11.69 | 0.95 |
| Res-net18 | 5% | 0.98 | 0.049 | 3.85 | 11.04 | 0.97 |
| Res-net28 | 10% | 0.98 | 0.097 | 4.96 | 18.86 | 0.95 |

Define the loss function $\mathcal{L}(\mathrm{x}+\delta)$. We aim to generate a $\delta$ that solves the following optimization problem:

$$\min_{\vec{\delta}} F(\mathbf{x}+\vec{\delta}) \quad \text{s.t. } \|\vec{\delta}\|_0 \leq \epsilon, \quad 0 \leq \mathbf{x}+\vec{\delta} \leq 1,$$

where $F(\mathbf{x}+\vec{\delta}) = \Big(\mathcal{L}(\mathbf{x}+\vec{\delta}), \|\vec{\delta}\|_2, \|\vec{\delta}\|_0\Big)^\top$ is the objective vector. $\vec{\delta}$ is the universal perturbation that we seek to optimize e use the pre-trained model on the CIFAR-10 dataset, we attacked five classifiers: CNN, VGG16, AlexNet, DenseNet, and ResNet. Two types of attacks were implemented: targeted and non-targeted attacks. In the targeted attack, the cross-entropy loss function was used to misclassify the model into a specific target class, while the non-targeted attack employed margin loss to force the model's output to differ from the actual class. Additionally, the algorithm restricted perturbations to the discrete value set $\{-1, 1, 0\}$, which helped reduce the l2 norm and ensured sparsity, enhancing both the effectiveness and stealth of the attack. **Metrics to evaluate the performance of attack methods include:** Average Attack Success Rate (ASR_avg), which measures the average success rate of misclassification due to adversarial attacks; Attack Success Rate (ASR), indicating the proportion of successful misclassifications; $l_0$ and $l_2$ norms, where $l_0$ counts the modified pixels and $l_2$ assesses perturbation magnitude; and Structural Similarity Index (SSIM), evaluating the similarity between the adversarial example and the original image, with values closer to 1 indicating less perceptible modifications.

We set $M = 2$, $\alpha = 0.1$, $\gamma = 0.001$, $v = 0.0001$, the maximum number of attack attempts 1000, and maximum modification per pixel 0.5. The corresponding results in Table 1 imply that the higher accuracy of the model could bring better effectiveness of the attack, which aligns with the principles of the zeroth-order multi-objective algorithm (*the more accurate the loss, the more accurate the gradient based on the loss*). Moreover, our attack success rate is generally above 90 percent, further demonstrating the advantages of our algorithm.

## 8 CONCLUSION

In this paper, we introduce the SZMOD algorithm, designed explicitly for black-box multi-objective learning. Theoretically, we establish the statistical guarantees for optimization error, generalization bound, and distance to conflict avoidance directions comparable to the relevant first-order method. Furthermore, we discover that zeroth-order methods could bridge the above three evaluation criteria of SZMOD. Experimentally, we validate SZMOD's performance in terms of optimization accuracy, generalization capability, and conflict avoidance. Additionally, we demonstrate the effectiveness of our algorithm in practical black-box attack scenarios, as evidenced by high attack success rates and low modification rates.

## REFERENCES

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

Lisha Chen, Heshan Fernando, Yiming Ying, and Tianyi Chen. Three-way trade-off in multi-objective learning: Optimization, generalization and conflict-avoidance. *Advances in Neural Information Processing Systems*, 36, 2024.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012.

Florian Felten, Umut Ucak, Hicham Azmani, Gao Peng, Willem Röpke, Hendrik Baier, Patrick Mannion, Diederik M Roijers, Jordan K Terry, El-Ghazali Talbi, et al. Momaland: A set of benchmarks for multi-objective multi-agent reinforcement learning. *arXiv preprint arXiv:2407.16312*, 2024.

Heshan Fernando, Han Shen, Miao Liu, Subhajit Chaudhury, Keerthiram Murugesan, and Tianyi Chen. Mitigating gradient bias in multi-objective learning: A provably convergent approach. International Conference on Learning Representations, 2023.

Jörg Fliege, A Ismael F Vaz, and Luís Nunes Vicente. Complexity of gradient descent for multiobjective optimization. *Optimization Methods and Software*, 34(5):949–959, 2019.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Jayesh K Gupta, Maxim Egorov, and Mykel Kochenderfer. Cooperative multi-agent control using deep reinforcement learning. In *Autonomous Agents and Multiagent Systems: AAMAS 2017 Workshops, Best Papers, São Paulo, Brazil, May 8-12, 2017, Revised Selected Papers 16*, pp. 66–83. Springer, 2017.

Tianmeng Hu, Biao Luo, Chunhua Yang, and Tingwen Huang. Mo-mix: Multi-objective multi-agent cooperative decision-making with deep reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12098–12112, 2023.

Yunwen Lei. Stability and generalization of stochastic optimization with nonconvex and nonsmooth problems. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 191–227. PMLR, 2023.

Siyuan Liang, Longkang Li, Yanbo Fan, Xiaojun Jia, Jingzhi Li, Baoyuan Wu, and Xiaochun Cao. A large-scale multiple-objective method for black-box attack against object detection. In *European Conference on Computer Vision*, pp. 619–636. Springer, 2022.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021.

Shengcai Liu, Ning Lu, Wenjing Hong, Chao Qian, and Ke Tang. Effective and imperceptible adversarial textual attack via multi-objectivization. *ACM Transactions on Evolutionary Learning and Optimization*, 4(3):1–23, 2024.

Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *International Conference on Learning Representations*, 2022.

Junlin Lu, Patrick Mannion, and Karl Mason. A multi-objective multi-agent deep reinforcement learning approach to residential appliance scheduling. *IET Smart Grid*, 5(4):260–280, 2022.

Alex Mathai, Shreya Khare, Srikanth Tamilselvam, and Senthil Mani. Adversarial black-box attacks on text classifiers using multi-objective genetic optimization guided by deep networks. *arXiv preprint arXiv:2011.03901*, 2020.

Brady Neal, Sarthak Mittal, Aristide Baratin, Vinayak Tantia, Matthew Scicluna, Simon Lacoste-Julien, and Ioannis Mitliagkas. A modern take on the bias-variance tradeoff in neural networks. *arXiv preprint arXiv:1810.08591*, 2018.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 506–519, 2017.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. Black-box tuning for language-model-as-a-service. In *International Conference on Machine Learning*, pp. 20841–20855. PMLR, 2022.

Hiroki Tanabe, Ellen H Fukuda, and Nobuo Yamashita. Proximal gradient methods for multiobjective optimization and their applications. *Computational Optimization and Applications*, 72:339–361, 2019.

Jordan Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sullivan, Luis S Santos, Clemens Dieffendahl, Caroline Horsch, Rodrigo Perez-Vicente, et al. Pettingzoo: Gym for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34:15032–15043, 2021.

Phoenix Neale Williams and Ke Li. Black-box sparse adversarial attack via multi-objective optimisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12291–12301, 2023.

Stephan Zheng, Alexander Trott, Sunil Srinivasa, David C Parkes, and Richard Socher. The ai economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science advances*, 8(18):eabk2607, 2022.

Shasha Zhou, Mingyu Huang, Yanan Sun, and Ke Li. Evolutionary multi-objective optimization for contextual adversarial example generation. *Proceedings of the ACM on Software Engineering*, 1 (FSE):2285–2308, 2024.