计算机科学
COMPUTER SCIENCE

坐标步长单调的动量对抗攻击方法

陈军, 陶蔚, 鲍蕾, 陶卿

引用本文

陈军, 陶蔚, 鲍蕾, 陶卿. 坐标步长单调的动量对抗攻击方法[J]. 计算机科学, 2026, 53(5): 426-434.

CHEN Jun, TAO Wei, BAO Lei, TAO Qing. Momentum Method with Monotonical Coordinate-wise Step-sizes for Adversarial Attacks [J]. Computer Science, 2026, 53(5): 426-434.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.250600185

坐标步长单调的动量对抗攻击方法

陈　军[1]　陶　蔚[2,3]　鲍　蕾[1]　陶　卿[1,4]
1 陆军兵种大学　合肥 230031
2 国防科技大学大数据与决策重点实验室　长沙 410073
3 军事科学院　北京 100091
4 合肥理工学院　合肥 238076
(chenjun342423@sina.com)

摘　要　对抗样本生成可以归结为最大化模型目标函数的优化问题,目前的求解策略主要采用符号梯度或者符号动量方法。然而这种方法牺牲了关键的梯度和动量方向信息,常常导致收敛性问题,从而造成了对抗攻击的不稳定性。受 AMSGrad 收敛性分析方法的启发,通过限定各坐标维度的步长单调递减,在 MI-FGSM 基础上提出了一种坐标步长单调的动量对抗攻击算法 MCS-MI。在一般凸条件下,证明了 MCS-MI 可以获得最优收敛速率 $O(1/\sqrt{T})$,其中 T 是迭代步数;并且,限定坐标步长单调作为一种通用且高效的策略,可以与现有的动量攻击算法相结合。在标准数据集上与近年来表现优异的 8 种对抗攻击算法进行了实验比较,其不仅具有很好的稳定性,还明显提升了攻击成功率,其中在 CNN 模型与 ViT 模型上的攻击成功率最高分别提升了 12.3%与 5.9%。

关键词:机器学习;对抗攻击;动量;符号梯度;收敛性
中图分类号　TP391

Momentum Method with Monotonical Coordinate-wise Step-sizes for Adversarial Attacks

CHEN Jun[1],TAO Wei[2,3],BAO Lei[1] and TAO Qing[1,4]
1 Army Arms University of PLA,Hefei 230031,China
2 Key Laboratory of Big Data and Decision-making,National University of Defense Technology(NUDT),Changsha 410073,China
3 Academy of Military Science,Beijing 100091,China
4 Hefei University of Technology,Hefei 238076,China

Abstract　The generation of adversarial samples can be due to an optimization problem aimed at maximizing the objective functions of models. Currently, the strategies to solve the induced problems primarily rely on sign-gradient or sign-momentum methods. However, these approaches sacrifice critical gradient and momentum direction information, often leading to convergence issues and then resulting in instability of adversarial attacks. Inspired by the convergence analysis of AMSGrad, this paper proposes a momentum method with monotonical coordinate-wise step-size(MCS-MI) based on MI-FGSM, which enforces monotonically decreasing coordinate-wise step-sizes. For general convex cases, MCS-MI is proved to attain an optimal convergence rate of $O(1/\sqrt{T})$, where T is the number of iterations. Furthermore, the strategy of enforcing monotonic coordinate-wise step-sizes is a general and efficient technique that can be integrated with existing momentum-based attack algorithms. Experimental comparisons with eight state-of-the-art adversarial attack methods on benchmark datasets demonstrate that the proposed approach not only exhibits superior stability but also significantly improves attack success rates, achieving maximum increases of 12.3% on CNN models and 5.9% on ViTs(Vision Transformers) respectively.

Keywords　Machine learning, Adversarial attacks, Momentum, Sign-gradient, Convergence

到稿日期:2025-06-26　返修日期:2025-09-22
基金项目:国家自然科学基金(60903098,62576351);中国博士后科学基金面上项目(2024M764294)
This work was supported by the National Natural Science Foundation of China(60903098,62576351) and China Postdoctoral Science Foundation (General Program) (2024M764294).
通信作者:陶卿(taoqing@gmail.com)

1　引言

深度神经网络(Deep Neural Network，DNN)因具有强大的表征能力，在图像分割[1]、目标检测[2]等多个领域展现了广泛的应用前景。然而，DNN 模型也暴露出了某种脆弱性，即对输入数据的微小变化高度敏感，这种特性可能导致模型易于过拟合[3]。通过向原始样本添加极小的、人类观察者几乎不可感知的噪声，可以轻易地误导 DNN 模型的判断。通常将这种向样本添加噪声的行为称作对抗攻击，而将添加噪声的样本称为对抗样本。研究表明，对抗样本具有迁移性，即在一个模型上生成的对抗样本有能力欺骗其他未知结构与参数的模型。对这一现象的直观理解是，不同模型训练得到的分类器具有相似性，这种相似性使得攻击者即使在无法直接访问目标模型的情况下，依然可以实施有效的攻击[4-6]。

对抗样本的生成过程通常归结为求解有约束条件下的最大值优化问题[6-7]。目前主流的求解策略为基于符号梯度或动量的优化方法，典型的方法包括：快速符号梯度方法(Fast Gradient Sign Method，FGSM)[4]、迭代快速符号梯度方法(Iterative Fast Gradient Sign Method，I-FGSM)[5]，以及基于动量的迭代快速符号梯度方法(Momentum Iterative Fast Gradient Sign Method，MI-FGSM)[6]等。它们与常规梯度方法最大的不同，是使用了符号梯度作为更新的方向。MI-FGSM 在 I-FGSM 的基础上，通过累积历史梯度信息，有效抑制了梯度更新过程中的振荡现象，更好地略过局部极值点，提高了对抗样本生成过程的稳定性及对抗攻击成功率。这种方法不仅在白盒模型上展现了出色的攻击效果，在黑盒模型上也实现了显著的攻击成效，曾在 2017 年 NIPS 的无目标攻击和有目标攻击竞赛中均获得了第一名，成为当前基于梯度优化攻击算法中一种至关重要的对比基准方法。为了进一步提升对抗样本的迁移性，后续研究中提出了许多基于 MI-FGSM 动量的方法，如 NI-FGSM[8]，GI-FGSM[9]，VMI-FGSM[10]，EMI-FGSM[11]，IE-FGSM[12]，PGN[13] 和 NCS[14] 等。其中，2024 年的 NCS 算法在多个模型上取得了最好的攻击成功率。

与标准的求解约束问题的优化算法相比，符号梯度算法具有一些明显的优点。首先，在计算过程中对传输的梯度值进行了有效的压缩，即仅传输 3 个值$\{-1,0,1\}$，这一策略显著降低了并行计算中梯度通信的成本；其次，得益于符号算法的简洁性，仅需适当地限制步长就可保证添加的扰动位于约束区域内，从而满足给定的约束条件。实验表明，符号梯度算法在图像攻击领域具有优异的性能[14]，特别是 MI-FGSM 的成功应用，使得符号动量方法成为研究者主要关注的优化求解策略。

然而，早在 2019 年 Karimireddy 等就通过具体的反例说明了，即使在简单凸条件下，符号梯度算法也可能无法收敛[15]。主要的原因是，符号梯度只是真实梯度非常粗糙的近似，丢失了确保收敛稳定性的关键梯度方向信息。在对抗攻击中，符号梯度方法常常会表现出不稳定的现象，如随着迭代步数的增加，MI-FGSM 的攻击成功率出现了大的波动，出现不增反降的结果(见第 4.5 节)。本文的主要目的是修正符号梯度方法，从算法收敛性角度来缓解对抗攻击的不稳定问题。

2018 年，Reddi 等[16]给出反例，指出 Adam 方法在一些情况下会出现不收敛的情形。为了得到一般凸情形下的收敛性，他们采取了限定步长单调的方法，在 Adam 基础上提出了 AMSGrad。本文将这种思想引入对抗攻击的符号动量方法中。与 AMSGrad 不同的是，本文没有要求整体步长单调递减，而是要求各坐标维度的步长单调递减，从而保持了符号方法的特点。

从形式上看，本文所提限定步长单调的方法只是对已有方法进行了简单修改，但无论从理论还是应用角度来看，该方法均取得了很好的效果。本文的主要贡献如下：

(1)在 MI-FGSM 基础上提出坐标步长单调的动量方法 MCS-MI(Momentum Method with Monotonically Coordinate-wise Step-Sizes)，在一般凸情形下得到最优的收敛速率，解决了对抗攻击领域优化方法缺乏收敛性的问题。

(2)所提方法作为一种通用且高效的策略，能够与现有的动量攻击算法结合，形成新的对抗攻击算法。在 ImageNet 验证集上，与近年来对抗攻击领域表现优异的 8 种对抗攻击算法在 6 类模型上(CNN/ViT)进行对比，所提方法不仅具有很好的稳定性，其平均对抗攻击成功率均有显著提升，其中在 CNN 模型与 ViT 模型上的攻击成功率最高分别提升了 12.3%与 5.9%。

2　相关工作

对抗攻击[10]可分为有目标攻击与无目标攻击。其中，有目标攻击是生成的对抗样本，使模型输出的类别为指定的标签，即 $f(\boldsymbol{x}^{\mathrm{adv}})=y^*$。这里，$\boldsymbol{x}^{\mathrm{adv}}$为原始样本 $\boldsymbol{x}$ 通过添加噪声生成的对抗样本；y^*为攻击者指定的目标标签，y 为原始样本的标签，且$y^*\neq y$。与此相对的是，无目标攻击只需满足生成的对抗样本使模型输出结果为错误的标签即可，即 $f(\boldsymbol{x}^{\mathrm{adv}})\neq y$。为了确保生成对抗样本的质量，通常对添加的扰动量进行约束 $\|\boldsymbol{x}^{\mathrm{adv}}-\boldsymbol{x}\|_p\leqslant\varepsilon$。这里，$\|\ \|_p$表示 p 范数距离，p 的取值可以设定为 0，1，2，∞。本文主要关注L_∞范数下的无目标攻击方法。

本章对 FGSM，I-FGSM，MI-FGSM，NI-FGSM 和 PGN 等符号梯度攻击算法进行简单介绍。

生成对抗样本的过程可归结为求解有约束条件下的模型最大化损失问题。

$$\max J(\boldsymbol{x}^{\mathrm{adv}},y)\quad \text{s.t.}\ \|\boldsymbol{x}^{\mathrm{adv}}-\boldsymbol{x}\|_\infty\leqslant\varepsilon \tag{1}$$

其中，$J(\boldsymbol{x}^{\mathrm{adv}},y)$表示一阶可微的损失函数。

(1)快速符号梯度算法

Goodfellow 等在 2015 年提出了符号梯度攻击算法 FGSM[4]。其更新规则为：

$$\boldsymbol{x}^{\mathrm{adv}}=\boldsymbol{x}+\varepsilon\,\mathrm{sign}(\nabla_x J(\boldsymbol{x},y)) \tag{2}$$

其中，sign(·)为符号函数，J 为损失函数，$\nabla_x J(\boldsymbol{x},y)$表示损失函数对输入样本 $\boldsymbol{x}$ 的梯度值。

(2)迭代快速梯度符号算法

Kurakin 等在 FGSM 的基础上，于 2017 年提出了迭代快速符号梯度方法 I-FGSM[5]。其通过迭代若干次来生成对抗样本，主要更新步骤为：

$$\begin{cases} \boldsymbol{x}_0^{\text{adv}} = \boldsymbol{x} \\ \boldsymbol{x}_{t+1}^{\text{adv}} = \boldsymbol{x}_t^{\text{adv}} + \alpha \operatorname{sign}(\nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y)) \end{cases} \tag{3}$$

为了满足 $\| \boldsymbol{x} - \boldsymbol{x}^{\text{adv}} \|_\infty \leqslant \varepsilon$ 的约束条件，设定每次更新迭代时的步长 α 为 ε/T，T 为迭代次数。研究表明，与单步攻击方法相比，I-FGSM 方法具有更高的白盒攻击成功率，但是迁移性稍弱。

(3)基于 Momentum 的迭代快速符号梯度算法

Dong 等将动量融入 I-FGSM 算法中，形成了 MI-FGSM 方法[6]。其主要更新步骤为：

$$\begin{cases} \boldsymbol{x}_0^{\text{adv}} = \boldsymbol{x}, \boldsymbol{g}_0 = 0, \alpha = \dfrac{\varepsilon}{T} \\ \boldsymbol{g}_{t+1} = \mu \boldsymbol{g}_t + \dfrac{\nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y)}{\| \nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y) \|_1} \\ \boldsymbol{x}_{t+1}^{\text{adv}} = \boldsymbol{x}_t^{\text{adv}} + \alpha \operatorname{sign}(\boldsymbol{g}_{t+1}) \end{cases} \tag{4}$$

其中，$\boldsymbol{g}_t$ 为前 t 次迭代中累加的梯度，μ 为动量系数。MI-FGSM 方法由于添加了动量项，累积了历史梯度信息，因此能够稳定更新方向并容易跳过局部极值点。与 FGSM 和 I-FGSM 方法相比，MI-FGSM 具有更高的攻击成功率与更好的过程稳定性。

(4)基于 Nesterov 动量的迭代快速符号梯度方法

Lin 等在 2020 年提出了基于 Nesterov 动量的迭代快速梯符号梯度算法(Nesterov Iterative Fast Gradient Sign Method，NI-FGSM)[8]。其在式(4)中求梯度时用 $\boldsymbol{x}_t^{\text{adv}} + \alpha * \mu * \boldsymbol{g}_t$ 取代了 $\boldsymbol{x}_t^{\text{adv}}$，这种向前一步的梯度攻击方法生成的对抗样本进一步远离了局部最优解，提高了在黑盒模型上的迁移性。

(5)PGN 算法

PGN(Penalizing Gradient Norm)是 Ge 等[13]于 2023 年提出的一种用于提升对抗样本迁移性的攻击方法。其核心思想是通过在损失函数中引入梯度范数的惩罚项，引导对抗样本生成在损失函数的“平坦局部最大值”区域，从而提高对抗样本的迁移性。其主要更新步骤为：

$$\nabla_g J(\boldsymbol{x}_{\text{adv}}, y) = (1-\delta) \nabla_{x'} J(\boldsymbol{x}', y) + \delta \nabla_{x'} J(\boldsymbol{x}' + a * v, y) \tag{5}$$

$$\boldsymbol{g}_{t+1} = \mu \boldsymbol{g}_t + \frac{\bar{\boldsymbol{g}}}{\| \bar{\boldsymbol{g}} \|} \tag{6}$$

$$\boldsymbol{x}_{t+1}^{\text{adv}} = \boldsymbol{x}_t^{\text{adv}} + \alpha \operatorname{sign}(\boldsymbol{g}_{t+1}) \tag{7}$$

其中，$\boldsymbol{x}'$ 是在 $\boldsymbol{B}_\xi(\boldsymbol{x}_{\text{adv}})$ 内随机采样的点；δ 是平均系数，用于控制处罚项的权重；$v = -\dfrac{\nabla_{x'} J(\boldsymbol{x}', y)}{\| \nabla_{x'} J(\boldsymbol{x}', y) \|}$ 是归一化的梯度方向；$\bar{\boldsymbol{g}}$ 是按照式(5)对多个随机采样点的梯度求平均得到的稳定梯度。

(6)NCS 算法

NCS(Neighborhood Conditional Sampling)是 Qiu 等[14]于 2024 年提出的对抗攻击算法，其核心思想是通过邻域条件采样和双层优化框架，寻找具有高期望对抗损失和低标准差的平坦对抗区域，从而生成迁移性更好的对抗样本。研究表明，NCS 可被看作当前基于梯度的对抗攻击算法中迁移性能最强的算法之一。然而，这些方法都采用符号梯度来确定更新方向，丢失了梯度的大小与方向信息，面临难以收敛的挑战。第 3 章中将以 MI-FGSM 动量方法为基础，重点分析算法的收敛性问题。

3 算法及收敛性分析

MI-FGSM[6]将基于 heavy ball 的动量优化方法与 I-FGSM 方法相结合，用以求解对抗攻击优化问题。实验结果表明，梯度除以自身 L_1 范数这种梯度规一化处理方式在图像对抗攻击方面取得了很好效果。因此，使用深度学习中常用的 EMA 动量方法(Exponential Moving Average)[18]，保留 MI-FGSM 使用 L_1 范数规一化处理方法的特点，易得出下列符号动量方法求解式(1)对抗攻击优化问题。

$$\begin{cases} \boldsymbol{m}_t = \beta_t \boldsymbol{m}_{t-1} + (1-\beta_t) \dfrac{\nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y)}{\| \nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y) \|_1} \\ \boldsymbol{x}_{t+1} = P_Q(\boldsymbol{x}_t + \alpha_t \operatorname{sign}(\boldsymbol{m}_t)) \end{cases} \tag{8}$$

其中，P_Q 为迭代后的值向定义域 Q 的投影操作。实际上，归一化的 EMA 符号动量方法等价于：

$$\begin{cases} \boldsymbol{m}_t = \beta_t \boldsymbol{m}_{t-1} + (1-\beta_t) \dfrac{\nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y)}{\| \nabla_x J(\boldsymbol{x}_t^{\text{adv}}, y) \|_1} \\ \boldsymbol{x}_{t+1} = P_Q(\boldsymbol{x}_t + \alpha_t \boldsymbol{D}_t \boldsymbol{m}_t) \end{cases} \tag{9}$$

其中，假设 $\boldsymbol{m}_t$ 和 $\boldsymbol{d}_t$ 为 d 维向量，$d_{t,i} = 1/|m_{t,i}|$ 表示动量 $\boldsymbol{m}_t$ 第 i 维绝对值的倒数，$\boldsymbol{D}_t = diag(\boldsymbol{d}_t)$ 表示正定的对角矩阵。

分析 2015 年 Kingma 等提出的 Adam 优化算法[19]：

$$\begin{cases} \boldsymbol{m}_t = \beta_{1t} \boldsymbol{m}_{t-1} + (1-\beta_{1t}) \nabla_x f(\boldsymbol{x}_t) \\ \boldsymbol{v}_t = \beta_{2t} v_{t-1} + (1-\beta_{2t}) diag(\nabla_x f(\boldsymbol{x}_t) \nabla f(\boldsymbol{x}_t)^{\mathrm{T}}) \\ \boldsymbol{x}_{t+1} = P_Q(\boldsymbol{x}_t - \alpha_t v_t^{-\frac{1}{2}} \boldsymbol{m}_t) \end{cases} \tag{10}$$

不难发现，上述两种算法具有相似性。实际上，EMA 形式的符号动量对抗攻击算法可被看作一种特殊的 Adam 算法，其中 $\boldsymbol{D}_t = -\boldsymbol{v}_t^{-\frac{1}{2}}$。

然而，Reddi 等指出 Adam 算法存在不收敛问题[16]，并构造了一个特殊的一般凸问题。考虑定义域为[-1,1]的损失函数：

$$f_t(w) = \begin{cases} cw, & \text{for } t \bmod 3 = 1 \\ -w, & \text{otherwise} \end{cases} \tag{11}$$

其中，$c=3$。在这个损失函数中可以很明显得出，$w=-1$ 时可得最小的后悔界。然而，Adam 算法错误地将方向指向 $w=1$ 进行更新，导致不收敛。

分析 Adam 算法不收敛的主要原因是其使用了 EMA 操作后，得到的 $\boldsymbol{v}_t$ 的单调性无法保证，导致最终无法获得理想的收敛性[16]。基于上述原因，Reddi 等提出了一种改进的 Adam 方法 AMSGrad。使用 $\hat{\boldsymbol{v}}_t = \max(\hat{\boldsymbol{v}}_{t-1}, \boldsymbol{v}_t)$ 取代 $\boldsymbol{v}_t$ 作为其更新时的自适应步长因子，进而确保迭代步长单调递减并，从理论上证明算法的收敛性。受上述方法启发，以 MI-FGSM 动量方法为基础，本文提出一种坐标单调的动量方法(MCS-MI)，确保更新步长在每一维度上单调，并从理论上证明算法的收敛性。具体过程如算法 1 所示。

算法 1 坐标步长单调的动量方法(MCS-MI)

输入：干净样本 $\mathbf{x}$；标签 y；损失函数 J；扰动 ε；步长 α_t；动量参数 $0<\beta<1$；$0<\lambda<1$；$\gamma>0$；迭代次数 T；$\alpha = \dfrac{\varepsilon}{\mathrm{T}}$

输出：对抗样本 $\mathbf{x}_{\mathrm{T}}$

1. $\mathbf{x}_0^{\text{adv}} = \mathbf{x}$；$\mathbf{g}_0 = 0$；$\mathbf{m}_0 = 0$；

$d_{0,i}$ 为初始值取 ∞

2. for t=1 to T−1 do

3. $\alpha_t=\alpha/\sqrt{t}$，$\beta_t=\beta\lambda^{t-1}$

4. $\mathbf{m}_t=\beta_t\mathbf{m}_{t-1}+(1-\beta_t)\frac{\nabla_x J(\mathbf{x}_t,y)}{\|\nabla_x J(\mathbf{x}_t,y)\|_1}$

5. $d_{t,i}=\min\left(\frac{\gamma}{|m_{t,i}|},d_{t-1,i}\right)$

6. $\mathbf{D}_t=\mathrm{diag}(\mathbf{d}_t)$

7. $\mathbf{x}_{t+1}=P_{Q,D_t^{-1}}(\mathbf{x}_t+\alpha_t\mathbf{D}_t\mathbf{g}_t)$

8. end for

为了便于计算并保持公式整洁，与 AMSGrad 算法一样，使用马式范数投影 $P_{Q,D_t^{-1}}(\boldsymbol{x}_t+\alpha\boldsymbol{D}_t\boldsymbol{m}_t)$。其定义为：

$$\arg\min_{x\in Q}\|\boldsymbol{D}_t^{-\frac{1}{2}}(\boldsymbol{x}-(\boldsymbol{x}_t+\alpha\boldsymbol{D}_t\boldsymbol{m}_t))\|$$

引理 1[18]　对于正定矩阵 $\boldsymbol{D}$，闭凸集 $Q\subseteq R^d$，假设：

$$\boldsymbol{u}_1=\min_{x\in Q}\|\boldsymbol{D}^{\frac{1}{2}}(\boldsymbol{x}-\boldsymbol{z}_1)\|,\boldsymbol{u}_2=\min_{x\in Q}\|\boldsymbol{D}^{\frac{1}{2}}(\boldsymbol{x}-\boldsymbol{z}_2)\|$$

则下列不等式成立：

$$\|\boldsymbol{D}^{\frac{1}{2}}(\boldsymbol{u}_1-\boldsymbol{u}_2)\|\leqslant\|\boldsymbol{D}^{\frac{1}{2}}(\boldsymbol{z}_1-\boldsymbol{z}_2)\| \tag{12}$$

引理 2　$\forall t,\gamma\leqslant d_{t,i}\leqslant d_{t,i-1}$ (13)

证明　由于 $\boldsymbol{m}_t$ 是由 $\frac{\nabla_x J(\boldsymbol{x}_t,y)}{\|\nabla_x J(\boldsymbol{x}_t,y)\|_1}$ 组成的凸组合，则 $m_{t,i}$ 是 $\frac{\nabla_x J(\boldsymbol{x}_{t,i},y)}{\|\nabla_x J(\boldsymbol{x}_t,y)\|_1}$ 组成的凸组合且 $\frac{\nabla_x J(\boldsymbol{x}_{t,i},y)}{\|\nabla_x J(\boldsymbol{x}_t,y)\|_1}\leqslant 1$，容易得出 $|m_{t,i}|\leqslant 1$，因为 $d_{t,i}=\min\left(\frac{\gamma}{|m_{t,i}|},d_{t-1,i}\right)$，则 $\gamma\leqslant d_{t,i}\leqslant d_{t-1,i}$。本文算法将 $\alpha_t d_{t,i}$ 作为更新时每一维度坐标的步长，确保了步长在每一维度上单调，故将所提算法命名为坐标步长单调的动量对抗攻击方法。

为了分析 MCS-MI 算法的收敛性，提出如下假设。

假设 1　假设存在常数 $G\geqslant 0$，$\forall\,\boldsymbol{x}_1,\boldsymbol{x}_2\in Q$：

$$\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_\infty\leqslant G \tag{14}$$

假设 2　假设存在常数 $M\geqslant 0$，$\forall\,\boldsymbol{x}_t\in Q$：

$$\|\nabla_x J(\boldsymbol{x}_t,y)\|_1\leqslant M \tag{15}$$

定理 1　假设式(14)和式(15)成立，$\boldsymbol{x}^*$ 为算法 1 的最优值，可以推出：

$$J(\boldsymbol{x}^*)-J(\bar{\boldsymbol{x}}_t)\leqslant\frac{MG^2d}{a\gamma(1-\beta)\sqrt{T}}+\frac{MG^2\beta}{2\gamma\alpha(1-\beta)(1-\lambda)\sqrt{T}}+\frac{Mad\gamma}{(1-\beta)} \tag{16}$$

其中，$\bar{\boldsymbol{x}}_t=\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{x}_t$，$d$ 为动量 $\boldsymbol{m}_t$ 的维数。

为了便于理解 MCS-MI 算法的收敛性定理，说明如下。

(1)定理 1 表明，对于一般凸函数，本文算法取得平均最优收敛速率 $O(1/\sqrt{T})$，克服了 MI-FGSM 动量方法由于使用梯度的符号导致的不收敛问题。

(2)本文算法基于 AMSGrad 改进：继承其步长单调递减策略，新增梯度归一化以加速收敛，并将优化方向调整为对抗样本生成的损失最大化(与模型参数训练的损失最小化形成对偶关系)。

(3)假设样本 $\boldsymbol{x}$ 能够被正确分类，则 $\boldsymbol{x}$ 通过迭代生成对抗样本 $\boldsymbol{x}_t$ 时，不难推出损失函数 $J(\boldsymbol{x}_t)>J(\boldsymbol{x})$。因此，直观上可以假设损失函数是局部凹的。在运算过程中，由于约束区域一般范围较小，认为上述假设在约束条件下成立。因此，MCS-MI 算法在对抗攻击中不仅对一般凸问题有效，也适用于非凸情况下的神经网络模型。

(4)MCS-MI 算法的核心创新在于去除动量符号并引入坐标步长单调性约束。该策略可泛化至 NI-FGSM，VMI-FGSM 和 GI-FGSM 等主流动量符号方法，形成 MCS-NI，MCS-VMI 和 MCS-GI 等优化变体。第 4.2 节实验结果表明：2023 年提出的 PGN 算法与 2024 年提出的 NCS 算法在梯度优化对抗攻击中表现优异，其对应的 MCS-PGN 和 MCS-NCS 算法通过步长单调优化仍能显著提升对抗样本的迁移性。算法 2 完整展示了 MCS-PGN 的实现步骤，其他变体算法均可通过算法 1 的基础框架，结合原算法进行推导。

算法 2　坐标步长单调的 PGN 方法(MCS-PGN)

输入：干净样本 $\mathbf{x}$；标签 y；损失函数 J；扰动 ε；步长参数 a；动量参数 $0<\beta<1$，$0<\lambda<1$，$\gamma>0$，迭代次数 T，随机取样数 N，插值因子 τ，随机取样界 ζ

输出：对抗样本 $\mathbf{x}_T$

1. $\mathbf{x}_0^{adv}=\mathbf{x}$；$\mathbf{g}_0=0$；$\alpha=\frac{\varepsilon}{T}$；

2. for t=1 to T−1 do

3. 　set $\bar{\mathbf{g}}=0$，$\alpha_t=\alpha/\sqrt{t}$

4. 　for i=0 to N−1 do

5. 　　随机取样 $\mathbf{x}'\in\mathcal{B}_\zeta(\mathbf{x}_t^{adv})$

6. 　　计算当前样本 $\mathbf{x}'$ 梯度：$\mathbf{g}'=\nabla J(\mathbf{x}',y)$

7. 　　计算预测点样本 $\mathbf{x}^*$：$\mathbf{x}^*=\mathbf{x}'-\alpha_t*\frac{\mathbf{g}'}{\|\mathbf{g}'\|_1}$

8. 　　计算预测点样本梯度：$\mathbf{g}^*=\nabla J(\mathbf{x}^*,y)$

9. 　　计算输出更新的梯度：

$$\bar{\mathbf{g}}=\bar{\mathbf{g}}+\frac{1}{N}*[(1-\tau)*\mathbf{g}'+\tau*\mathbf{g}^*]$$

10. 　　$\mathbf{m}_0=\mathbf{0}$；$d_{0,i}=\infty$；$\beta_t=\beta\lambda^{t-1}$

11. 　　$\mathbf{m}_t=\beta_t\mathbf{m}_{t-1}+\frac{\bar{\mathbf{g}}}{\|\bar{\mathbf{g}}\|_1}$

12. 　　$d_{t,i}=\min(\frac{\gamma}{|m_{t,i}|},d_{t-1,i})$，$\mathbf{D}_t=\mathrm{diag}(\mathbf{d}_t)$为正定对角矩阵

13. 　　$\mathbf{x}_{t+1}=P_{Q,D_t^{-1}}(\mathbf{x}_t+a_t\mathbf{D}_t\mathbf{m}_t)$

14. end for

4　实验及结果分析

本文使用对抗样本攻击成功率作为主要指标来衡量不同算法的有效性。攻击成功率是指生成的对抗样本被模型误分类的概率，其计算式为：

$$\frac{\sum_{i=1}^{N}[f(\boldsymbol{x}_i)=y_i\wedge f(\boldsymbol{x}_i^*)\neq y_i]}{\sum_{i=1}^{N}[f(\boldsymbol{x}_i)=y_i]} \tag{17}$$

其中，$\boldsymbol{x}_i$ 表示干净的样本，$\boldsymbol{x}_i^*$ 表示对抗样本，y_i 表示样本的真实标签，N 是数据集样本的总数。

本部分实验主要有以下目的。

(1)将本文提出的动量方法 MCS-MI，MCS-NI，MCS-VMI，MCS-GI，MCS-EMI，MCS-IE，MCS-PGN，MCS-NCS 分别与其对应的 MI-FGSM[6]，NI-FGSM[8]，VMI-FGSM[9]，GI-FGSM[10]，EMI[11]，IE[12]，PGN[13]，NCS[14] 等原符号动量方法在一般模型上进行比较，验证所提算法的有效性。后续为了进一步验证算法的有效性，同时便于比较，主要以 Res34 为白盒，以 MCS-MI 和 MCS-PGN 算法与对应的原

算法 MI-FGSM 和 PGN 进行比较。以其他模型为白盒，各类不同算法的比较均已实施并进行了实验验证，这里不再详述。

(2)以 Res34 为白盒，使用 MCS-MI 和 MCS-PGN 算法与对应的 MI-FGSM 和 PGN 算法在防御模型上比较对抗样本的攻击成功率，验证算法的有效性。

(3)以 Res34，Inc-v3 和 Vgg16 为白盒，比较 MCS-MI 和 MCS-PGN 算法与 MI-FGSM 和 PGN 算法在集成模式下生成的对抗样本的攻击成功率，验证算法的有效性。

(4)以 Res34 为白盒，比较 MCS-MI 与 MI-FGSM 算法在使用数据增强方法后生成对抗样本的攻击成功率，验证算法的有效性。

(5)通过对比 MCS-MI 算法与 MI-FGSM 算法的迭代次数与攻击成功率之间的变化情况，验证本文算法在迭代过程的稳定性。

4.1 实验设置

4.1.1 数据集

本文实验使用的数据集与 MI-FGSM，NI-FGSM 和 PGN 等算法使用的数据集相同，均来自 ILSVRC2012 验证集[20]随机抽取的 1000 张属于不同类别的图片。

4.1.2 模型选择

本文使用的模型包括 8 个常规模型与 4 个对抗训练模型。常规训练模型分别从 CNN 与 ViT 模型中选取，其中 CNN 模型共有 5 种，分别为 ResNet-34(Res34)[21]，Inception-v3(Inc-v3)[22]，Vgg16[23]，Densenet121(Dens-121)[24] 和 Mobilenet-V2(Mob-V2)[25]，它们均采用 Torchvision 库提供的模型预训练参数；ViT 模型有 Vit-Base-patch16(Vit-B)[26]，Visformer-Smal(Vis-S)和 Swin-Tiny-patch4(Swin-T)[27]这 3 种，对抗训练模型有 Inc-v3adv，IncRes-v2ens，Efficient-B0adv 和 Efficient-B1adv[28]这 4 种，这 7 种模型均采用 timm 库提供的模型预训练参数。

4.1.3 超参设置

实验中所有参与比较的对抗攻击算法，其超参数与 TransferAttack[1)] 中设置的参数保持一致。本文提出的坐标步长单调的动量对抗攻击方法涉及的动量系数 β 和 λ 通过网格搜索方法得到，均设置为 0.999，γ 设置为 3/2。

4.2 基于动量的对抗攻击

本节在 8 种常规模型上分别对比 MI-FGSM，NI-FGSM，VMI-FGSM，GI-FGSM，EMI-FGSM，IE，PGN，NCS 算法与对应的坐标步长单调动量算法 MCS-MI，MCS-NI，MCS-VMI，MCS-GI，MCS-EMI，MCS-IE，MCS-PGN，MCS-NCS 的黑盒攻击成功率。实验分别以 Res34，Inc-v3，Vgg16，Vit-B，Vis-S 和 Swin-T 为白盒，攻击成功率如表 1 所列。其中，“*”标注的是白盒攻击，本文所提坐标步长单调算法均已加粗表示。具体可视化情况如图 1 所示。表 1 表明，坐标步长单调算法在大部分模型中均能有效提升原算法的迁移性，尤其是 MCS-NCS 算法在各个模型上，其攻击率相较于原 NCS 算法仍有显著提高。

表 1 不同算法在一般模型上的攻击成功率

Table 1 Attack success rates of different algorithms on general models

(%)

模型	攻击方法	Res34	Inc-v3	Vgg16	Dens-121	Mob-V2	Vit-B	Vis-S	Swin-T	平均
Res34	MI-FGSM	100.0*	56.3	70.5	68.0	75.4	16.8	35.4	40.4	57.9
	MCS-MI	100.0*	58.7	74.2	70.1	76.5	19.5	36.7	42.7	59.8
	NI-FGSM	100.0*	59.7	73.9	71.9	77.4	18.5	35.8	41.3	59.8
	MCS-NI	100.0*	60.7	76.1	72.1	78.8	18.7	39.1	43.7	61.2
	VMI-FGSM	100.0*	74.3	83.9	80.2	88.8	32.1	56.2	58.7	71.8
	MCS-VMI	100.0*	76.1	86.0	82.9	90.0	33.0	57.9	58.6	73.1
	GI-FGSM	100.0*	63.5	79.5	77.3	80.3	21.1	40.2	43.6	63.2
	MCS-GI	100.0*	66.0	82.1	78.7	80.9	20.7	41.8	46.3	64.6
	EMI	100.0*	72.4	87.8	84.4	89.1	23.8	50.6	53.8	70.2
	MCS-EMI	100.0*	73.2	89.6	84.2	89.5	23.4	50.3	54.0	70.5
	IE	100.0*	62.8	76.8	72.5	80.8	19.7	41.0	44.9	62.3
	MCS-IE	100.0*	62.4	76.1	75.4	81.1	19.8	41.1	46.1	62.8
	PGN	100.0*	**86.1**	93.0	88.2	94.5	36.5	60.5	67.6	78.3
	MCS-PGN	100.0*	85.7	**93.4**	**89.4**	**94.9**	**39.4**	**63.0**	**68.4**	**79.3**
	NCS	100.0*	90.2	94.7	96.3	92.7	48.8	74.3	77.9	84.4
	MCS-NCS	100.0*	**91.3**	**96.2**	**98.5**	**94.3**	**50.5**	**77.9**	**80.1**	**86.1**
Inc-v3	MI-FGSM	42.8	98.0*	50.1	49.5	44.6	13.3	25.9	30.1	44.3
	MCS-MI	55.1	98.9*	59.9	59.3	55.2	25.2	35.1	40.0	53.6
	NI-FGSM	49.2	98.4*	54.3	56.4	51.6	14.7	28.1	32.8	48.2
	MCS-NI	61.1	98.7*	64.2	63.1	61.6	24.8	37.9	42.5	56.7
	VMI-FGSM	57.6	98.7*	59.0	60.1	59.4	21.7	34.6	39.7	53.9
	MCS-VMI	59.2	98.7*	61.7	62.2	60.7	22.1	37.8	40.5	55.4
	GI-FGSM	51.9	98.0*	57.0	57.6	51.1	16.3	28.6	33.1	49.2
	MCS-GI	53.9	98.8*	61.3	60.9	55.8	16.5	31.9	34.4	51.7
	EMI	63.4	99.8*	66.5	64.2	62.8	18.9	32.7	39.3	55.6
	MCS-EMI	65.4	100.0*	67.5	67.3	64.4	19.6	34.2	39.9	57.3
	IE	50.5	98.3*	55.5	55.8	53.4	15.6	29.0	33.0	48.9
	MCS-IE	52.8	99.1*	56.3	58.6	54.2	17.2	30.1	36.1	50.6
	PGN	72.3	100.0*	70.5	67.9	72.1	27.4	**43.8**	48.8	62.9
	MCS-PGN	**73.1**	100.0*	**71.0**	**68.2**	**72.9**	**27.9**	42.0	**49.6**	**63.1**
	NCS	81.6	99.1*	78.1	81.5	76.8	39.2	57.9	59.8	71.8
	MCS-NCS	**85.1**	99.4*	**82.4**	**85.2**	**81.7**	**42.5**	**62.9**	**63.2**	**75.3**

1) https://GitHub-Trustworthy-AI-Group/TransferAttack

(续表)

模型	攻击方法	Res34	Inc-v3	Vgg16	Dens-121	Mob-V2	Vit-B	Vis-S	Swin-T	平均
Vgg16	MI-FGSM	57.7	45.9	99.9*	67.0	60.6	13.6	31.1	36.5	51.5
	MCS-MI	60.9	49.0	100.0*	69.1	64.6	14.2	33.7	39.4	53.9
	NI-FGSM	60.4	45.8	99.9*	69.9	61.2	13.6	31.0	36.1	52.2
	MCS-NI	61.6	48.4	100.0*	72.9	63.0	15.0	34.4	37.8	54.1
	VMI-FGSM	75.2	60.8	100.0*	79.3	77.5	22.2	47.4	53.4	64.5
	MCS-VMI	77.3	61.7	100.0*	80.1	78.7	22.3	48.7	54.7	65.4
	GI-FGSM	65.1	50.2	99.9*	74.1	68.2	14.4	33.9	40.4	55.8
	MCS-GI	68.2	53.9	99.9*	77.3	69.5	15.2	36.0	42.9	57.9
	EMI	71.8	56.9	100.0*	78.8	74.4	16.9	40.6	47.5	60.9
	MCS-EMI	72.7	56.0	100.0*	81.1	74.2	16.1	40.7	47.8	61.1
	IE	66.7	50.4	99.9*	72.4	70.3	14.8	37.2	41.6	56.7
	MCS-IE	66.3	49.7	100.0*	75.0	70.4	15.7	37.2	42.2	57.1
	PGN	86.6	**74.3**	100.0*	86.4	85.6	27.0	53.3	**61.5**	71.8
	MCS-PGN	**87.3**	73.2	100.0*	**87.8**	**86.3**	**27.9**	**54.4**	60.5	**72.2**
	NCS	91.2	78.6	100.0*	92.2	91.0	36.8	65.4	72.4	78.5
	MCS-NCS	**94.2**	**82.0**	**100.0***	**93.7**	**94.6**	**38.2**	**70.6**	**74.2**	**80.9**
Vit-B	MI-FGSM	48.0	46.4	57.6	56.7	50.8	97.4*	42.8	55.5	56.9
	MCS-MI	51.6	49.6	63.2	60.1	54.6	98.5*	46.3	58.1	60.3
	NI-FGSM	48.6	49.6	58.8	56.4	53.8	97.3*	44.5	56.5	58.2
	MCS-NI	52.9	53.9	63.7	61.2	58.5	98.5*	47.3	60.5	62.1
	VMI-FGSM	56.9	55.0	64.5	62.2	59.9	98.7*	57.7	67.0	65.2
	MCS-VMI	60.6	60.4	67.5	67.3	65.5	99.4*	60.4	70.4	68.9
	GI-FGSM	59.3	58.5	70.5	69.7	65.9	99.4*	56.8	68.6	68.6
	MCS-GI	61.8	61.0	71.4	70.0	66.6	99.6*	58.1	69.8	69.8
	EMI	64.0	63.5	70.2	70.5	69.0	99.4*	64.6	75.9	72.1
	MCS-EMI	67.9	66.1	73.7	73.4	72.8	99.8*	67.4	78.7	75.0
	IE	53.4	52.7	62.9	62.2	58.3	99.0*	49.5	61.8	62.5
	MCS-IE	54.8	52.7	65.3	63.8	59.1	99.9*	48.9	62.2	63.3
	PGN	73.8	73.5	77.2	77.4	78.7	97.3*	76.1	81.4	79.4
	MCS-PGN	**76.8**	**75.8**	**78.4**	**80.3**	**80.4**	99.6*	**79.0**	**84.0**	**81.8**
	NCS	74.3	72.6	74.9	76.8	77.2	95.8*	75.5	81	74.3
	MCS-NCS	**79.9**	**79.5**	**82.3**	**83.1**	**83.0**	99.1*	**81.5**	**86.8**	**79.9**
Vis-S	MI-FGSM	52.0	50.7	65.1	63.8	59.6	29.6	97.2*	58.2	59.5
	MCS-MI	53.8	52.1	70.1	67.2	62.2	31.4	98.7*	60.0	61.9
	NI-FGSM	53.5	51.9	66.9	66.5	61.3	31.3	98.8*	59.1	61.2
	MCS-NI	56.3	53.3	69.9	69.4	63.6	32.2	99.1*	61.7	63.2
	VMI-FGSM	68.4	65.7	75.6	75.5	74.4	55.6	97.2*	76.2	73.6
	MCS-VMI	71.7	70.3	80.1	79.0	78.5	59.4	98.9*	80.2	77.3
	GI-FGSM	65.3	63.1	78.6	77.1	74.3	41.6	99.3*	73.8	71.6
	MCS-GI	65.3	64.2	78.1	76.9	74.6	41.0	99.5*	73.6	71.7
	EMI	72.2	69.0	81.2	81.8	79.8	51.8	99.6*	83.2	77.3
	MCS-EMI	74.4	70.2	83.3	82.2	81.2	52.3	99.7*	82.3	78.2
	IE	56.9	54.7	70.6	70.3	64.1	34.7	98.9*	65.7	64.5
	MCS-IE	56.8	54.5	71.0	70.9	64.8	33.5	99.6*	65.7	64.6
	PGN	84.4	82.2	83.8	84.6	86.0	76.4	95.7*	86.8	85.0
	MCS-PGN	**86.1**	**84.4**	**87.3**	**88.3**	**89.3**	**78.5**	97.7*	**89.4**	**87.6**
	NCS	84.2	81.7	86.6	87.3	86.1	76.6	96.8*	86.9	85.8
	MCS-NCS	**88.2**	**87.8**	**91.2**	**91.7**	**90.4**	**82.0**	98.1*	**90.9**	**90.0**
Swin-T	MI-FGSM	36.2	34.9	48.9	51.9	38.8	21.0	34.1	95.9*	45.2
	MCS-MI	37.5	34.8	51.7	55.1	42.0	20.8	35.4	97.4*	46.8
	NI-FGSM	37.2	37.1	48.8	53.5	37.6	20.3	34.7	96.5*	45.7
	MCS-NI	40.4	36.9	52.5	56.0	42.8	22.1	38.1	97.0*	48.2
	VMI-FGSM	54.5	53.5	62.8	66.7	59.2	46.4	60.3	97.5*	62.6
	MCS-VMI	58.7	56.7	66.7	70.8	62.9	49.3	66.2	98.8*	66.3
	GI-FGSM	47.5	43.9	59.9	63.6	51.8	28.6	46.2	99.6*	55.1
	MCS-GI	48.5	43.9	62.1	64.3	52.0	29.7	46.0	99.8*	55.8
	EMI	52.2	47.4	61.8	67.0	54.1	32.7	52.4	99.8*	58.4
	MCS-EMI	52.7	50.1	64.8	68.5	55.6	34.4	53.5	99.8*	59.9
	IE	37.1	35.8	51.4	55.7	39.5	19.5	35.1	98.2*	46.5
	MCS-IE	40.1	34.9	51.3	55.4	41.6	19.6	35.1	98.1*	47.0
	PGN	83.3	81.9	86.5	88.9	86.3	78.2	87.0	99.5*	86.5
	MCS-PGN	**86.3**	**84.1**	**89.0**	**90.6**	**89.0**	**80.3**	**89.5**	99.8*	**88.6**
	NCS	85.3	83.4	88.9	87.7	90.6	80.6	90.5	99.3*	88.3
	MCS-NCS	**90.6**	**87.0**	**92.9**	**93.4**	**94.8**	**86.0**	**94.8**	**100***	**92.4**

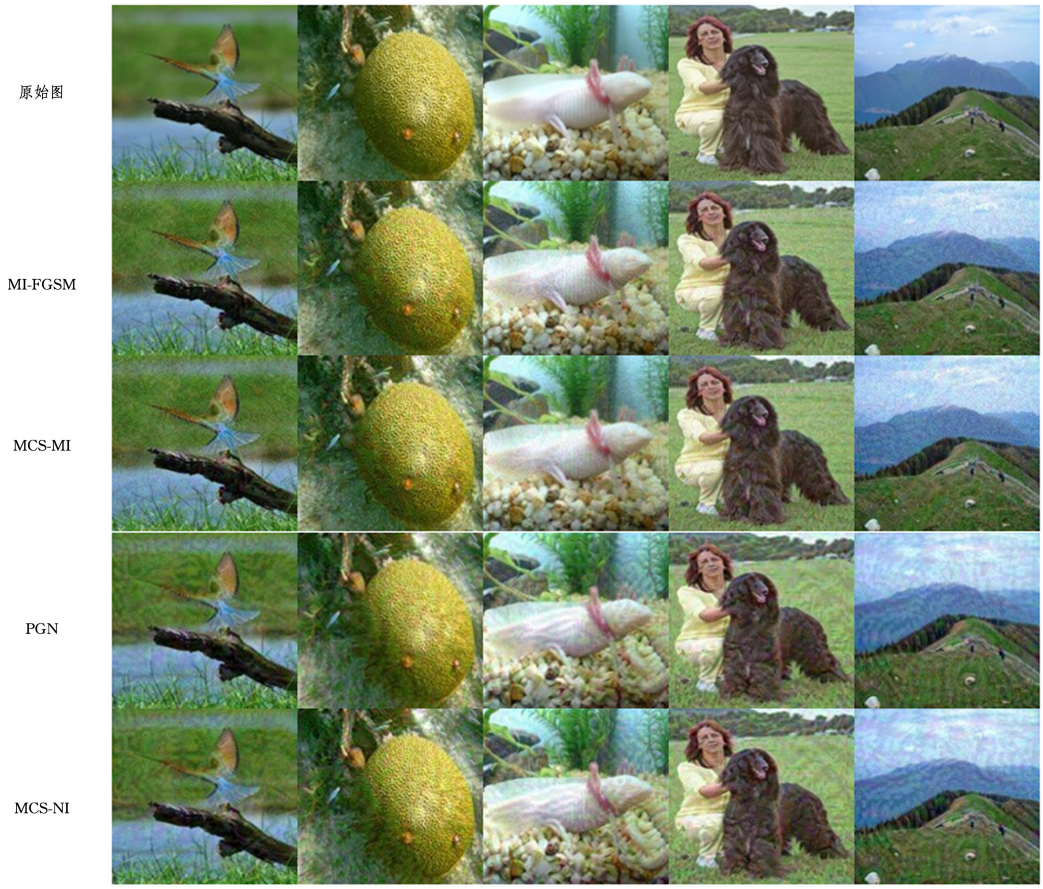


图 1 MIFGSM, PGN, MCS-MI 和 MCS-PGN 在扰动值为 16 的对抗样本图片

Fig. 1 Adversarial sample images with a perturbation value of 16 for MIFGSM, PGN, MCS-MI, and MCS-PGN

为了进一步验证算法的有效性，以 Res34 为白盒，以 4 个防御模型为黑盒，将 MI-FGSM 和 PGN 算法与 CMS-MI 和 MCS-PGN 算法分别进行比较，其对抗攻击成功率如表 2 所列。

表 2 不同算法在防御模型上的黑盒攻击成功率

Table 2 Black-box attack success rates of different algorithms on defensive models

(%)

模型	攻击方法	Inc-v3adv	IncRes-v2ens	Efficient-B0adv	Efficient-B1adv	平均
Res34	MI-FGSM	42.3	25.7	43.1	37.4	37.1
	MCS-MI	43.3	28.9	46.6	42.2	40.3
	PGN	73.2	**61.2**	76.8	73.1	71.1
	MCS-PGN	**74.6**	60.4	**78.5**	**74.4**	**72.0**

分析表 2 可以发现，无论在一般模型上，还是在防御模型上，坐标步长单调方法均能在原有算法基础上有效提高对抗样本的攻击成功率。

4.3 集成攻击方法

研究表明，同时使用多个模型集成作为白盒，能够有效提升对抗样本的迁移性[29-30]。为了进一步验证算法的有效性，将 MCS-MI, MCS-PGN, MI-FGSM 和 PGN 算法分别运用于集成攻击模式。需要说明的是，对模型的集成，采用 Dong 等[6]提出的 logits 集成方法，采用加权平均的方式获取最终输出的 logits 值。这里使用 Res34, Inc-v3 和 Vgg16 作为白盒生成对抗样本。攻击成功率如表 3 所列。分析表 3 可以得出，在集成攻击模式下，所提算法与原算法相比，仍能有效提升对抗样本的攻击成功率。

表 3 不同算法在集成模型下的攻击成功率

Table 3 Attack success rates of different algorithms under ensemble models

(%)

攻击方法	Res34*	Inc-v3*	Vgg16*	Dens-121	Mob-V2	Vit-B	Vis-S	Swin-T	Inc-v3adv	IncRes-v2ens	Efficient-B0adv	Efficient-B1adv	平均
MI-FGSM	100.0	99.6	99.6	74.3	81.4	28.1	51.5	52.4	60.2	41.2	58.4	55.5	66.9
MCS-MI	100.0	99.8	99.9	77.4	83.3	29.2	53.8	55.3	62.3	42.0	60.7	57.6	68.4
PGN	100.0	99.8	99.9	95.0	96.8	**60.0**	81.1	83.4	91.9	81.9	90.5	88.8	89.1
MCS-PGN	100.0	100.0	100.0	**95.6**	**98.0**	59.6	**83.1**	**85.5**	**93.0**	**84.3**	**91.5**	**91.6**	**90.2**

4.4 数据增强方法

与其他优化算法一样，本文算法可以与数据增强方法相结合，进一步提升对抗样本的迁移性。为了进一步验证算法的有效性，分别将 MCS-MI 和 MI-FGSM 两种算法与 3 种典型的数据增强方法 TI[31]，DI[32] 和 SI[8] 相结合，攻击成功率如表 4 所列。表 4 表明，与数据增强方法结合后，所提算法与原算法相比，仍能有效提升对抗样本的攻击成功率。

表 4　算法在不同数据增强方法下的攻击成功率

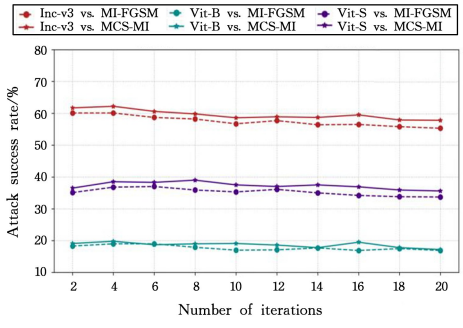
Table 4　Attack success rates of algorithms under different data augmentation methods

（%）

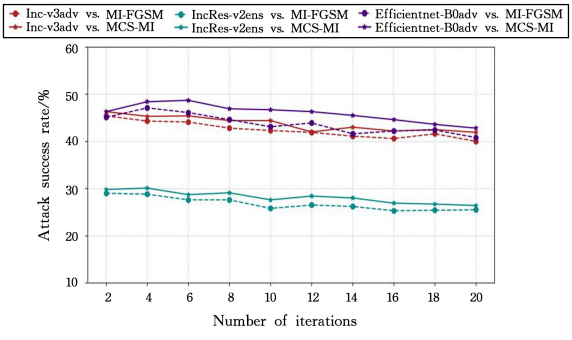
攻击方法	Res34*	Inc-v3	Vgg16	Dens-121	Mob-V2	Vit-B	Vis-S	Swin-T	Inc-v3adv	IncRes-v2ens	Efficient-B0adv	Efficient-B1adv	平均
TI＋MI-FGSM	100.0	52.6	71.5	59.2	70.9	15.9	29.2	37.7	43.4	34.4	41.2	41.0	49.8
TI＋MCS-MI	100.0	55.9	74.3	63.1	75.0	17.2	30.7	39.6	46.3	36.4	45.4	43.8	52.3
SI＋MI-FGSM	100.0	76.9	84.5	78.8	89.8	26.0	49.3	54.5	60.3	41.6	61.4	55.2	64.9
SI＋MCS-MI	100.0	78.3	**86.7**	82.0	91.0	27.9	53.1	54.4	62.9	44.2	63.8	57.7	66.8
DI＋MI-FGSM	100.0	77.6	84.8	79.4	89.8	30.6	53.4	55.3	63.5	47.2	64.5	62.2	67.4
DI＋MCS-MI	100.0	**79.6**	84.3	**82.2**	**91.7**	**32.2**	**55.9**	**56.3**	**66.1**	**48.4**	**68.5**	**65.6**	**69.2**

4.5　算法稳定性实验

MI-FGSM 等符号动量算法由于使用了动量项，因而能够稳定地更新样本数据[6]。为了验证本文算法的稳定性，以 Res34 为白盒，研究迭代次数 T 值在 2 到 20 内变化时，MI-FGSM 与 CMS-MI 算法攻击成功率的变化情况，具体情况如图 2 所示。图 2(a)表明，随着 T 值不断增大，两种算法在一般模型上均能保持良好的稳定性。图 2(b)表明，在对抗训练模型上，随着 T 值不断增大，两种算法均出现了一定程度的波动，但整体上所提算法保持了更好的稳定性。



(a)对一般模型的攻击成功率



(b)对防御模型的攻击成功率

图 2　不同迭代次数下算法的攻击成功率

Fig. 2　Attack success rate of algorithms at different iteration counts

结束语　本文分析了动量迭代快速梯度符号方法(MI-FGSM)不易收敛的原因，提出了一种坐标步长单调的动量对抗攻击算法(MCS-MI)，并在一般凸情形下得到最优的收敛速率，解决了对抗攻击领域优化方法缺乏收敛性的问题。实验结果验证了所提方法在提升对抗样本迁移性方面具有显著效果。未来的工作将集中在以下几个方面：首先，探索如何将算法应用于大模型(如 Transformer 架构的模型)微调训练中，以提高模型的鲁棒性和泛化能力；其次，进一步分析所提算法在非凸优化问题中的行为，并探索可能的改进方法，以提高算法的性能和稳定性；最后，利用坐标步长单调算法生成的对抗样本，设计更有效的对抗防御策略，推动对抗攻击和防御技术的发展，提高深度学习模型的鲁棒性和安全性。

参 考 文 献

[1] LANG C, CHENG G, TU B, et al. Learning What Not to Segment: A New Perspective on Few-Shot Segmentation[C]// 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2022: 8047-8057.

[2] TIAN Z, SHEN C, CHEN H, et al. FCOS: Fully Convolutional One-Stage Object Detection[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV). 2019: 9626-9635.

[3] GOODFELLOW I J, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial networks [J]. Communications of the ACM, 2014, 63: 139-144.

[4] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and Harnessing Adversarial Examples[J]. arXiv: 1412.6572, 2014.

[5] KURAKIN A, GOODFELLOW I J, BENGIO S. Adversarial examples in the physical world[J]. arXiv: 1607.02533, 2016.

[6] DONG Y, LIAO F, PANG T, et al. Boosting Adversarial Attacks with Momentum [C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 9185-9193.

[7] MADRY A, MAKELOV A, SCHMIDT L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks [J]. arXiv: 1706.06083, 2017.

[8] LIN J, SONG C, HE K, et al. Nesterov Accelerated Gradient and Scale Invariance for Adversarial Attacks[J]. arXiv: 1908.06281, 2019.

[9] WANG J, CHEN Z, JIANG K, et al. Boosting the Transferability of Adversarial Attacks with Global Momentum Initialization [J]. arXiv: 2211.11236, 2022.

[10] WANG X, HE K. Enhancing the Transferability of Adversarial Attacks through Variance Tuning[C]// 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021: 1924-1933.

[11] WANG X, LIN J, HU H, et al. Boosting Adversarial Transferability through Enhanced Momentum[C]// British Machine Vision Conference. 2021.

[12] PENG A, LIN Z, ZENG H, et al. Boosting Transferability of Adversarial Example via an Enhanced Euler's Method[C]// ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2023: 1-5.

[13] GE Z, SHANG F, LIU H, et al. Boosting Adversarial Transferability by Achieving Flat Local Maxima[J]. arXiv: 2306.05225, 2023.

[14] QIU C, DUAN Y, ZHAO L, et al. Enhancing Adversarial Transferability Through Neighborhood Conditional Sampling[J]. arXiv:2405.16181, 2024.
[15] KARIMIREDDY S P, REBJOCK Q, STICH S U, et al. Error Feedback Fixes SignSGD and other Gradient Compression Schemes[J]. arXiv:1901.09847, 2019.
[16] REDDI S J, KALE S, KUMAR S. On the Convergence of Adam and Beyond[J]. arXiv:1904.09237, 2019.
[17] ZINKEVICH M A. Online Convex Programming and Generalized Infinitesimal Gradient Ascent[C]// International Conference on Machine Learning. 2003.
[18] LONG S, TAO W, ZHANG Z, et al. Optimal Convergence Rate of Adam-Type Algorithms for Non-Smooth Strongly Convex Problems[J]. Journal of Electronics, 2022(9): 2049-2059.
[19] KINGMA D P, BA J. Adam: A Method for Stochastic Optimization[J]. arXiv:1412.6980, 2014.
[20] RUSSAKOVSKY O, DENG J, SU H, et al. ImageNet Large Scale Visual Recognition Challenge[J]. International Journal of Computer Vision, 2014, 115: 211-252.
[21] HE K, ZHANG X, REN S, et al. Deep Residual Learning for Image Recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2016: 770-778.
[22] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the Inception Architecture for Computer Vision[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 2818-2826.
[23] SIMONYAN K, ZISSERMAN A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. arXiv:1409.1556, 2014.
[24] HUANG G, LIU Z, WEINBERGER K Q. Densely Connected Convolutional Networks[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017: 2261-2269.
[25] SANDLER M, HOWARD A G, ZHU M, et al. MobileNetV2: Inverted Residuals and Linear Bottlenecks[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 4510-4520.
[26] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale[J]. arXiv:2010.11929, 2020.
[27] LIU Z, LIN Y, CAO Y, et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows[C]// 2021 IEEE/CVF International Conference on Computer Vision (ICCV). 2021: 9992-10002.
[28] TRAMÈR F, KURAKIN A, PAPERNOT N, et al. Ensemble Adversarial Training: Attacks and Defenses[J]. arXiv: 1705.07204, 2017.
[29] LIU Y, CHEN X, LIU C, et al. Delving into Transferable Adversarial Examples and Black-box Attacks[J]. arXiv: 1611.02770, 2016.
[30] BAO L, TAO W, TAO Q. Enhancing Transferability of Adversarial Attacks by Combining Adaptive Step Size Strategy and Data Augmentation Mechanism[J]. Journal of Electronics, 2024(1): 157-169.
[31] DONG Y, PANG T, SU H. Evading Defenses to Transferable Adversarial Examples by Translation-Invariant Attacks[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR). 2019: 4307-4316.
[32] XIE C, ZHANG Z, WANG J, et al. Improving Transferability of Adversarial Examples With Input Diversity[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019: 2725-2734.

CHEN Jun, born in 1989, postgraduate. His main research interests include machine learning and mathematical optimization.

TAO Qing, born in 1965, Ph.D, professor, doctoral supervisor, is a senior member of CCF(No. 09081S). His main research interests include machine learning, pattern recognition and applied mathematics.

(责任编辑:柯颖)