How to beat a Bayesian adversary

Ding, Zihan; Jin, Kexin; Latz, Jonas; Liu, Chenguang

DOI
10.1017/S0956792525000105

Publication date
2025

Document Version
Final published version

Published in
European Journal of Applied Mathematics

European Journal of Applied Mathematics (2025), 1–23
doi:10.1017/S0956792525000105

CAMBRIDGE UNIVERSITY PRESS

PAPER

How to beat a Bayesian adversary

Zihan Ding[1], Kexin Jin[2], Jonas Latz[3] and Chenguang Liu[4]

[1]Department of Electrical and Computer Engineering, Princeton University, Princeton, USA
[2]Department of Mathematics, Princeton University, Princeton, USA
[3]Department of Mathematics, University of Manchester, Manchester, UK
[4]Delft Institute of Applied Mathematics, Technische Universiteit Delft, Delft, The Netherlands
Corresponding author: Jonas Latz; Email: jonas.latz@manchester.ac.uk

Received: 11 July 2024; **Revised:** 13 January 2025; **Accepted:** 22 February 2025

Keywords: Machine learning; adversarial robustness; Stochastic differential equations; McKean–Vlasov process; particle system

2020 Mathematics Subject Classification: 90C15, 65C35, 68T07

Abstract

Deep neural networks and other modern machine learning models are often susceptible to adversarial attacks. Indeed, an adversary may often be able to change a model's prediction through a small, directed perturbation of the model's input – an issue in safety-critical applications. Adversarially robust machine learning is usually based on a minmax optimisation problem that minimises the machine learning loss under maximisation-based adversarial attacks. In this work, we study adversaries that determine their attack using a Bayesian statistical approach rather than maximisation. The resulting Bayesian adversarial robustness problem is a relaxation of the usual minmax problem. To solve this problem, we propose Abram – a continuous-time particle system that shall approximate the gradient flow corresponding to the underlying learning problem. We show that Abram approximates a McKean–Vlasov process and justify the use of Abram by giving assumptions under which the McKean–Vlasov process finds the minimiser of the Bayesian adversarial robustness problem. We discuss two ways to discretise Abram and show its suitability in benchmark adversarial deep learning experiments.

1 Introduction

Machine learning and artificial intelligence play a major role in today's society: self-driving cars (e.g. [3]), automated medical diagnoses (e.g. [41]) and security systems based on face recognition (e.g. [45]), for instance, are often based on certain machine learning models, such as *deep neural networks* (DNNs). DNNs often approximate functions that are discontinuous with respect to their input [48] making them susceptible to so-called *adversarial attacks*. In an adversarial attack, an adversary aims to change the prediction of a DNN through a directed, but small perturbation to the input. We refer to [14] for an example showing the weakness of DNNs towards adversarial attacks. Especially when employing DNNs in safety-critical applications, the training of machine learning models in a way that is robust to adversarial attacks has become a vital task.

Machine learning models are usually trained by minimising an associated loss function. In adversarially robust learning, this loss function is considered to be subject to adversarial attacks. The adversarial attack is usually given by a perturbation of the input data that is chosen to maximise the loss function. Thus, adversarial robust learning is formulated as a minmax optimisation problem. In practice, the inner maximisation problem needs to be approximated: [14] proposed the fast gradient sign method (FGSM), which perturbs the input data to maximise the loss function with a single step. Improvements of FGSM were proposed by, e.g. [25, 51, 57]. Another popular methodology is projected gradient descent

(PGD) [32] and its variants, see, for example, [9, 10, 33, 36, 50, 61]. Similar to FGSM, PGD considers the minmax optimisation problem but uses multi-step gradient ascent to approximate the inner maximisation problem. Notably, [57] showed that FGSM with random initialisation is as effective as PGD.

Other defense methods include preprocessing (e.g. [16, 47, 59, 63]) and detection (e.g. [5, 31, 35, 58]), as well as provable defenses (e.g. [15, 21, 46, 55]). Various attack methods have also been proposed, see, for instance, [6, 9, 13, 56]. More recently, there is an increased focus on using generative models to improve adversarial accuracy, see for example [38, 54, 60].

In the present work, we study the case of an adversary that finds their attack following a Bayesian statistical methodology. The Bayesian adversary does not find the attack through optimisation, but by sampling a probability distribution that can be derived using Bayes' Theorem. Importantly, we study the setting in which the adversary uses a Bayesian strategy, but the machine learner/defender trains the model using optimisation, which is in contrast to [62]. Thus, our ansatz is orthogonal to previous studies of adversarial robustness by assuming that the attacker uses a significantly different technique. On the other hand, the associated Bayesian adversarial robustness problem can be interpreted as a stochastic relaxation of the classical minmax problem that replaces the inner maximisation problem with an integral. Thus, our ansatz should also serve as an alternative way to approach the computationally challenging minmax problem with a sampling based strategy. After establishing these connections, we

- propose *Abram* (short for *Adversarial Bayesian Particle Sampler*), a particle-based continuous-time dynamical system that simultaneously approximates the behaviour of the Bayesian adversary and trains the model via gradient descent.

Particle systems of this form have been used previously to solve such optimisation problems in the context of maximum marginal likelihood estimation, see, e.g. [2] and [24]. In order to justify the use Abram in this situation, we

- show that Abram converges to a McKean–Vlasov stochastic differential equation (SDE) as the number of particles goes to infinity, and
- give assumptions under which the McKean–Vlasov SDE converges to the minimiser of the Bayesian adversarial robustness problem with an exponential rate.

Additional complexity arises here compared to earlier work as the dynamical system and its limiting McKean–Vlasov SDE have to be considered under reflecting boundary conditions. After the analysis of the continuous-time system, we briefly explain its discretisation. Then, we

- compare Abram to the state of the art in adversarially robust classification of the MNIST and the CIFAR-10 datasets under various kinds of attacks.

This work is organised as follows. We introduce the (Bayesian) adversarial robustness problem in Section 2 and the Abram method in Section 3. We analyse Abram in Sections 4 (large particle limit) and 5 (longtime behaviour). We discuss different ways of employing Abram in practice in Section 6 and compare it to the state of the art in adversarially robust learning in Section 7. We conclude in Section 8.

2 Adversarial robustness and its Bayesian relaxation

In the following, we consider a supervised machine learning problem of the following form. We are given a *training dataset* $\{(y_1, z_1), \dots, (y_K, z_K)\}$ of pairs of features $y_1, \dots, y_K \in Y := \mathbb{R}^{d_Y}$ and *labels* $z_1, \dots, z_K \in Z$. Moreover, we are given a parametric model of the form $g{:}X \times Y \to Z$, with $X := \mathbb{R}^d$ denoting the parameter space. The goal is now to find a parameter θ^*, for which

$$g(y_k|\theta^*) \approx z_k \qquad (k = 1, \dots, K).$$

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

In practice the function $g(\cdot\,|\theta^*)$ shall then be used to predict labels of features (especially such outside of training dataset).

The parameter θ^* is usually found through optimisation. Let $\mathcal{L}{:}Z \times Z \to \mathbb{R}$ denote a loss function – a function that gives a reasonable way of comparing the output of g with observed labels. Usual examples are the square loss for continuous labels and cross entropy loss for discrete labels. Then, we need to solve the following optimisation problem:

$$\min_{\theta \in X} \frac{1}{K} \sum_{k=1}^{K} \Phi(y_k, z_k|\theta), \tag{2.1}$$

where $\Phi(y, z|\theta) := \mathcal{L}(g(y|\theta), z)$.

Machine learning models g that are trained in this form are often susceptible to adversarial attacks. That means, for a given feature vector y, we can find a 'small' $\xi \in Y$ for which $g(y + \xi\,|\theta^*) \neq g(y|\theta^*)$. In this case, an adversary can change the model's predicted label by a very slight alteration of the input feature. Such a ξ can usually be found through optimisation on the input domain:

$$\max_{\xi \in B(\varepsilon)} \Phi(y + \xi, z|\theta),$$

where $B(\varepsilon) = \{\xi : \|\xi\| \leq \varepsilon\}$ denotes the ε-ball centred at 0 and $\varepsilon > 0$ denotes the size of the adversarial attack. Hence, the attacker tries to change the prediction of the model whilst altering the model input only by a small value $\leq \varepsilon$. Other kinds of attacks are possible, the attacker may, e.g. try to not only change the predicted label to any other label, but rather to a particular target label, see, e.g. [26].

In adversarially robust training, we replace the optimisation problem (2.1) by the minmax optimisation problem below:

$$\min_{\theta \in X} \frac{1}{K} \sum_{k=1}^{K} \max_{\xi_k \in B(\varepsilon)} \Phi(y_k + \xi_k, z_k|\theta). \tag{2.2}$$

Thus, we now train the network by minimising the loss also with respect to potential adversarial attacks. Finding the accurate solutions to such minmax optimisation problems is difficult: usually there is no underlying saddlepoint structure, e.g. $\Phi(y, z|\theta)$ is neither convex in θ nor concave in y, X and Y tend to be very high-dimensional spaces, and the number of data points K may prevent the accurate computation of gradients. However, good heuristics have been established throughout the last decade – we have mentioned some of them in Section 1.

In this work, we aim to study a relaxed version of the minmax problem, which we refer to as the *Bayesian adversarial robustness problem*. This problem is given by

$$\min_{\theta \in X} \frac{1}{K} \sum_{k=1}^{K} \int_{B(\varepsilon)} \Phi(y_k + \xi_k, z_k|\theta) \pi_k^{\gamma,\varepsilon}(\mathrm{d}\xi_k|\theta), \tag{2.3}$$

where the *Bayesian adversarial distribution* $\pi_k^{\gamma,\varepsilon}(\,\cdot\,|\theta)$ has (Lebesgue) density

$$\xi \mapsto \frac{\exp\left(\gamma\, \Phi(y_k + \xi, z_k|\theta)\right) \mathbf{1}[\xi \in B(\varepsilon)]}{\int_{B(\varepsilon)} \exp\left(\gamma\, \Phi(y_k + \xi', z_k|\theta)\right) \mathrm{d}\xi'},$$

where $\gamma > 0$ is an *inverse temperature*, $\varepsilon > 0$ still denotes the size of the adversarial attack, and $\mathbf{1}[\,\cdot\,]$ denotes the indicator: $\mathbf{1}[\text{true}] := 1$ and $\mathbf{1}[\text{false}] := 0$. The distribution $\pi_k^{\gamma,\varepsilon}(\,\cdot\,|\theta)$ is concentrated on the ε-ball, $\varepsilon > 0$ controls the range of the attack, $\gamma > 0$ controls its focus. We illustrate this behaviour in Figure 1. Next, we comment on the mentioned relaxation and the Bayesian derivation of this optimisation problem.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

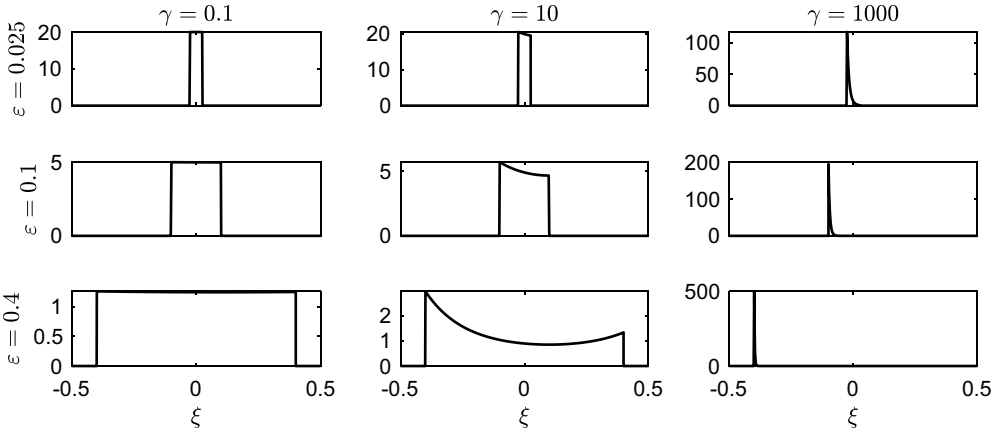

Figure 1. *Plots of the Lebesgue density of* $\pi_1^{\gamma,\varepsilon}(\cdot\,|\theta_0)$ *for energy* $\Phi(y_1+\xi, z_1|\theta_0) = (\xi - 0.1)^2/2$, *choosing parameters* $\varepsilon \in \{0.025, 0.1, 0.4\}$ *and* $\gamma \in \{0.1, 10, 1000\}$.

2.1 Relaxation

Under certain assumptions,[1] one can show that

$$\pi_k^{\gamma,\varepsilon}(\cdot\,|\theta) \to \mathrm{Unif}(\mathrm{argmax}_{\xi\in Y}\Phi(y_k+\xi, z_k|\theta))$$

weakly as $\gamma \to \infty$, see [20]. Indeed, the Bayesian adversarial distribution converges to the uniform distribution over the global maximisers computed with respect to the adversarial attack. This limiting behaviour, that we can also see in Figure 1, forms the basis of simulated annealing methods for global optimisation. Moreover, it implies that the optimisation problems (2.2) and (2.3) are identical in the limit $\gamma \to \infty$, since

$$\lim_{\gamma\to\infty} \frac{1}{K}\sum_{k=1}^{K}\int_{B(\varepsilon)} \Phi(y_k+\xi_k, z_k|\theta)\pi_k^{\gamma,\varepsilon}(\mathrm{d}\xi_i|\theta)$$
$$= \frac{1}{K}\sum_{k=1}^{K}\int_{B(\varepsilon)} \Phi(y_k+\xi_k, z_k|\theta)\mathrm{Unif}(\mathrm{argmax}_{\xi\in Y}\Phi(y_k+\xi, z_k|\theta))(\mathrm{d}\xi_i),$$

and since $\xi_k \sim \mathrm{Unif}(\mathrm{argmax}_{\xi\in Y}\Phi(y_k+\xi, z_k|\theta))$ implies $\Phi(y_k+\xi_k, z_k|\theta) = \max_{\xi\in B(\varepsilon)}\Phi(y_k+\xi, z_k|\theta)$ almost surely for $k = 1, \ldots, K$. A strictly positive γ on the other hand leads to a relaxed problem circumventing the minmax optimisation. [8] have also discussed this relaxation of an adversarial robustness problem in the context of a finite set of attacks, i.e. the ε-ball $B(\varepsilon)$ is replaced by a finite set. *Probabilistically robust learning* is another type of relaxation, see for example [4, 43]. Similar to our work, instead of doing the worst-case optimisation, i.e. finding the perturbation ξ that maximises the loss, they replace it with a probability measure on ξ. This probability measure, however, follows a different paradigm.

2.2 Bayesian attackers

We can understand the kind of attack that is implicitly employed in (2.3) as a Bayesian attack. We now briefly introduce the Bayesian learning problem to then explain its relation to this adversarial attack. In Bayesian learning, we model θ as a random variable with a so-called *prior (distribution)* π_{prior}. The prior incorporates information about θ. In Bayesian learning, we now inform the prior about data

[1] Assume, for instance, that Φ is three times differentiable and has only finitely many maximisers and note that $B(\varepsilon)$ is compact.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

$\{(y_1, z_1), \ldots, (y_K, z_K)\}$ by conditioning θ on that data. Indeed, we train the model by finding the conditional distribution of θ given that $g(y_k|\theta) \approx z_k$ $(k = 1, \ldots, K)$. In the Bayesian setting, we represent '$\approx$' by a noise assumption consistent with the loss function $\mathcal{L}$. This is achieved by defining the so-called *likelihood* as $\exp(-\Phi)$. The conditional distribution describing θ is called the *posterior (distribution)* π_{post} and can be obtained through Bayes' theorem, which states that

$$\pi_{\text{post}}(A) = \frac{\int_A \exp\left(-\frac{1}{K}\sum_{k=1}^{K} \Phi(y_k, z_k|\theta)\right) \pi_{\text{prior}}(\mathrm{d}\theta)}{\int_X \exp\left(-\frac{1}{K}\sum_{i=k}^{K} \Phi(y_k, z_k|\theta)\right) \pi_{\text{prior}}(\mathrm{d}\theta)},$$

for measurable $A \subseteq X$. A model prediction with respect to feature y can then be given by the posterior mean of the output g, which is

$$\int_{\mathbb{R}^n} g(y|\theta)\pi_{\text{post}}(\mathrm{d}\theta).$$

The Bayesian attacker treats the attack ξ_k in exactly such a Bayesian way. They define a prior distribution for the attack, which is the uniform distribution over the ε-ball:

$$\text{Unif}(B(\varepsilon)) = \int_{B(\varepsilon)} \mathbf{1}[\xi_k \in \cdot]\mathrm{d}\xi_k.$$

The adversarial likelihood is designed to essentially cancel out the likelihood in the Bayesian learning problem, by defining a function that gives small mass to the learnt prediction and large mass to anything that does not agree with the learnt prediction:

$$\exp(\gamma\,\Phi(y_k + \xi_k, z_k|\theta)).$$

Whilst this is not a usual likelihood corresponding to a particular noise model, we could see this as a special case of *Bayesian forgetting* [12]. In Bayesian forgetting, we would try to remove a single dataset from a posterior distribution by altering the distribution of the parameter θ. In this case, we try to alter the knowledge we could have gained about the feature vector by altering that feature vector to produce a different prediction.

3 Adversarial Bayesian particle sampler

We now derive a particle-based method that shall solve (2.3). To simplify the presentation in the following, we assume that $K = 1$, i.e. there is only a single data point. The derivation for multiple data points is equivalent – computational implications given by multiple data points will be discussed in Section 6. We also ignore the dependence of Φ on particular data points and note only the dependence on parameter and attack. Indeed, we write (2.3) now as

$$\min_{\theta \in X} F(\theta) := \int_{B(\varepsilon)} \Phi(\xi, \theta)\pi^{\gamma,\varepsilon}(\mathrm{d}\xi|\theta).$$

To solve this minimisation problem, we study the gradient flow corresponding to the energy F, that is: $\mathrm{d}\zeta_t = -\nabla_\zeta F(\zeta_t)\mathrm{d}t$. The gradient flow is a continuous-time variant of the gradient descent algorithm. The gradient flow can be shown to converge to a minimiser of F in the longterm limit if F satisfies certain

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

regularity assumptions. The gradient of F has a rather simple expression:

$$\begin{aligned}\nabla_\theta F(\theta) &= \nabla_\theta \frac{\int_{B(\varepsilon)} \Phi(\xi,\theta)\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi}{\int_{B(\varepsilon)}\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi}\\ &= \frac{\int_{B(\varepsilon)} \nabla_\theta\Phi(\xi,\theta)\cdot\exp(\gamma\Phi(\xi,\theta)) + \gamma\nabla_\theta\Phi(\xi,\theta)\cdot\Phi(\xi,\theta)\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi}{\int_{B(\varepsilon)}\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi}\\ &\quad - \frac{\left(\int_{B(\varepsilon)}\Phi(\xi,\theta)\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi\right)\left(\int_{B(\varepsilon)}\gamma\nabla_\theta\Phi(\xi,\theta)\cdot\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi\right)}{\left(\int_{B(\varepsilon)}\exp(\gamma\Phi(\xi,\theta))\mathrm{d}\xi\right)^2}\\ &= \int_{B(\varepsilon)}\nabla_\theta\Phi(\xi,\theta)\pi^{\gamma,\varepsilon}(\mathrm{d}\xi\,|\theta) + \gamma\mathrm{Cov}_{\pi^{\gamma,\varepsilon}(\cdot|\theta)}(\Phi(\cdot,\theta),\nabla_\theta\Phi(\cdot,\theta)),\end{aligned}$$

where we assume that Φ is continuously differentiable, bounded below and sufficiently regular to be allowed here to switch gradients and integrals. As usual, we define the covariance of appropriate functions f, g with respect to a probability distribution π, by

$$\mathrm{Cov}_\pi(f,g) := \int_X f(\theta)g(\theta)\pi(\mathrm{d}\theta) - \int_X f(\theta)\pi(\mathrm{d}\theta)\int_X g(\theta)\pi(\mathrm{d}\theta).$$

The structure of $\nabla_\theta F$ is surprisingly simple, requiring only integrals of the target function and its gradient with respect to $\pi^{\gamma,\varepsilon}$, but, e.g. not its normalising constant. In practice, it is usually not possible to compute these integrals analytically or to even sample independently from $\pi^{\gamma,\varepsilon}(\cdot\,|\theta)$, which would be necessary for a stochastic gradient descent approach. The latter approach first introduced by [42] allows the minimisation of expected values by replacing these expected values by sample means; see also [22] and [28] for continuous-time variants. Instead, we use a particle system approach that has been studied for a different problem by [2] and [24]. The underlying idea is to approximate $\pi^{\gamma,\varepsilon}(\cdot\,|\theta)$ by an overdamped Langevin dynamics, which is restricted to the ε-Ball $B(\varepsilon)$ with reflecting boundary conditions:

$$\mathrm{d}\xi_t = \gamma\nabla_\xi\Phi(\xi_t,\theta)\mathrm{d}t + \sqrt{2}\mathrm{d}W_t,$$

where $(W_t)_{t\geq 0}$ denotes a standard Brownian motion on Y. Alternatively, one may write the dynamics as $\mathrm{d}\xi_t = \nabla_\xi\Phi(\xi_t,\theta)\mathrm{d}t + \sqrt{2/\gamma}\mathrm{d}W_t$, which is equivalent to the current form after a time re-scaling. Under weak assumptions on Φ, this Langevin dynamics converges to the distribution $\pi^{\gamma,\varepsilon}(\cdot\,|\theta)$ as $t\to\infty$. However, due to the heavy computational costs, in practice, we are not able to simulate the longterm behaviour of this dynamics for all fixed θ to produce samples of $\pi^{\gamma,\varepsilon}(\cdot\,|\theta)$ as required for stochastic gradient descent. Instead, we run a number N of (seemingly independent) Langevin dynamics $(\xi_t^{1,N})_{t\geq0},\dots,(\xi_t^{N,N})_{t\geq0}$. We then obtain an approximate gradient flow $(\theta_t^N)_{t\geq0}$ that uses the ensemble of particles $(\xi_t^{1,N})_{t\geq0},\dots,(\xi_t^{N,N})_{t\geq0}$ to approximate the expected values in the gradient $\nabla_\theta F$ and then feed $(\theta_t^N)_{t\geq0}$ back into the drift of the $(\xi_t^{1,N})_{t\geq0},\dots,(\xi_t^{N,N})_{t\geq0}$. Hence, we simultaneously approximate the gradient flow $(\zeta_t)_{t\geq0}$ by $(\theta_t^N)_{t\geq0}$ and the Bayesian adversarial distribution $(\pi^{\gamma,\varepsilon}(\cdot\,|\theta_t^N))_{t\geq0}$ by $(\xi_t^{1,N})_{t\geq0},\dots,(\xi_t^{N,N})_{t\geq0}$. Overall, we obtain the dynamical system

$$\begin{aligned}\mathrm{d}\theta_t^N &= -\frac{1}{N}\sum_{n=1}^N\nabla_\theta\Phi(\xi_t^{n,N},\theta_t^N)\mathrm{d}t - \gamma\widehat{\mathrm{Cov}}(\xi_t^N)\mathrm{d}t,\\ \mathrm{d}\xi_t^{i,N} &= \gamma\nabla_\xi\Phi(\xi_t^{i,N},\theta_t^N)\mathrm{d}t + \sqrt{2}\mathrm{d}W_t^i \qquad (i=1,\dots,N).\end{aligned}$$

where $(W_t^i)_{t\geq0}$ are mutually independent Brownian motions on Y for $i=1,\dots,N$. Again, the Langevin dynamics $(\xi_t^{1,N})_{t\geq0},\dots,(\xi_t^{N,N})_{t\geq0}$ are defined on the ball $B(\varepsilon)$ with reflecting boundary conditions – we formalise this fact below. The empirical covariance is given by

$$\widehat{\mathrm{Cov}}(\xi_t^N) = \frac{1}{N}\sum_{i=1}^N\Phi(\xi_t^{i,N},\theta_t^N)\nabla_\theta\Phi(\xi_t^{i,N},\theta_t^N) - \frac{1}{N^2}\sum_{i=1}^K\Phi(\xi_t^{i,N},\theta_t^N)\sum_{j=1}^K\nabla_\theta\Phi(\xi_t^{j,N},\theta_t^N).$$

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

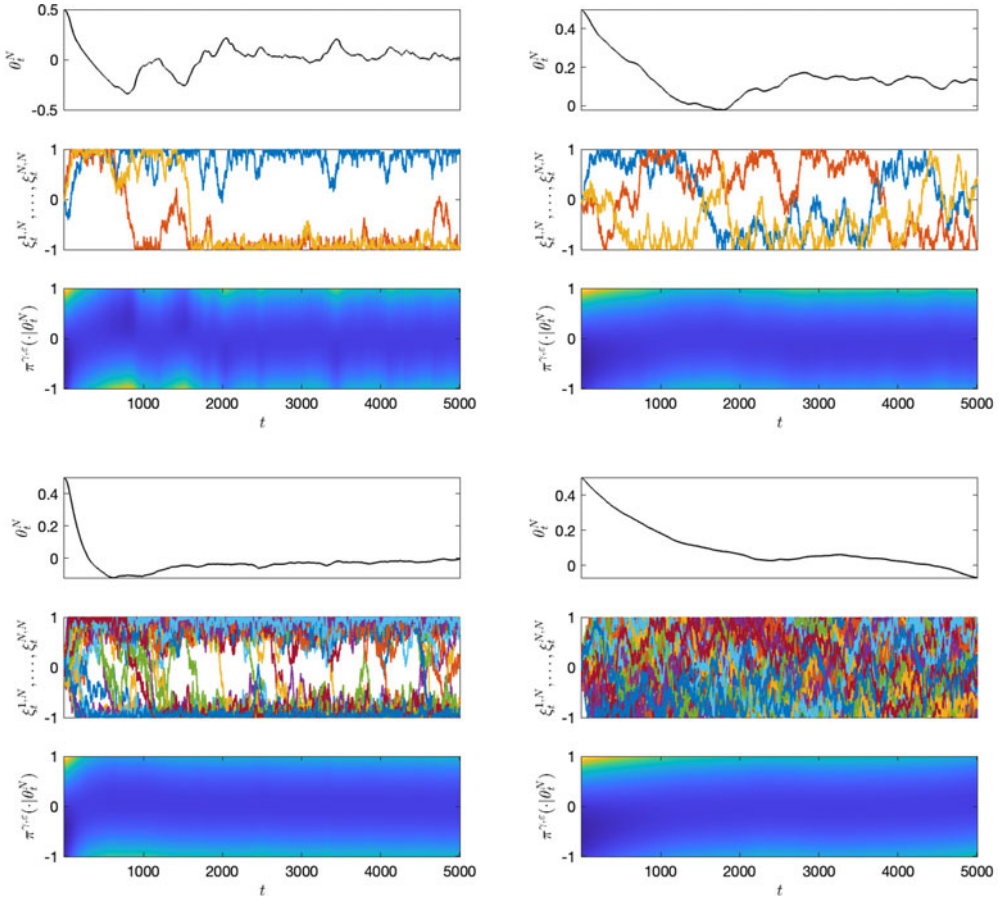

Figure 2. *Examples of the Abram method given* $\Phi(\xi,\theta)=\frac{1}{2}(\xi+\theta)^2$, $\varepsilon=1$, *and different combinations of* $(\gamma,N)=(10,3)$ *(top left),* $(0.1,3)$ *(top right),* $(10,50)$ *(bottom left),* $(0.1,50)$ *(bottom right). In each of the four quadrants, we show the simulated path* $(\theta_t^N)_{t\geq 0}$ *(top), the particle paths* $(\xi_t^{1,N},\dots,\xi_t^{N,N})_{t\geq 0}$ *(centre), and the path of probability distributions* $(\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_t^N))_{t\geq 0}$ *(bottom) that shall be approximated by the particles. The larger* γ *leads to a concentration of* $\pi^{\gamma,\varepsilon}$ *at the boundary, whilst it is closer to uniform if* γ *is small. More particles lead to a more stable path* $(\theta_t^N)_{t\geq 0}$. *A combination of large N and* γ *leads to convergence to the minimiser* $\theta_*=0$ *of F.*

We refer to the dynamical system $(\theta_t^N,\xi_t^{1,N},\dots,\xi_t^{N,N})_{t\geq 0}$ as *Abram*. We illustrate the dynamics of Abram in Figure 2, where we consider a simple example.

We have motivated this particle system as an approximation to the underlying gradient flow $(\zeta_t)_{t\geq 0}$. As $N\to\infty$, the dynamics $(\theta_t^N)_{t\geq 0}$ does not necessarily convergence to the gradient flow $(\zeta_t)_{t\geq 0}$, but to a certain McKean–Vlasov stochastic differential equation (SDE), see [34]. We study this convergence behaviour in the following, as well as the convergence of the McKean–Vlasov SDE to the minimiser of F and, thus, justify Abram as a method for Bayesian adversarial learning. First, we introduce the complete mathematical set-up and give required assumptions. To make it easier for the reader to keep track of the different stochastic processes that appear throughout this work, we summarise them in Table 1.

3.1 Mean-field limit

In the following, we are interested in the mean field limit of Abram, i.e. we analyse the limit of $(\theta_t^N)_{t\geq 0}$ as $N\to\infty$. Thus, we can certainly assume for now that $\gamma:=1$ and $\varepsilon\in(0,1)$ being fixed.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Table 1. *Definitions of stochastic processes throughout this work*

$(\theta_t^N, \xi_t^{1,N}, \dots, \xi_t^{1,N})_{t\geq 0}$	$(\theta_t, \xi_t)_{t\geq 0}$	$(\xi_t^1, \dots, \xi_t^N)_{t\geq 0}$	$(\widehat{\xi}_t)_{t\geq 0}$
(3.1)	(3.2)	(4.1)	(5.3)
Particle system / Abram	Limiting equation	Independent sampling	Coupling process

We write $B := B(\varepsilon)$. Then, Abram $(\theta_t^N, \xi_t^{1,N}, \dots, \xi_t^{N,N})_{t\geq 0}$ satisfies

$$\theta_t^N = \theta_0 - \int_0^t \mu_s^N(\nabla_\theta \Phi(\cdot, \theta_s^N))\mathrm{d}s - \int_0^t \mathrm{Cov}_{\mu_s^N}(\Phi(\cdot, \theta_s^N), \nabla_\theta \Phi(\cdot, \theta_s^N))\mathrm{d}s, \tag{3.1}$$
$$\xi_t^{i,N} = \xi_0^i + \int_0^t \nabla_x \Phi(\xi_s^{i,N}, \theta_s^N)\mathrm{d}s + \sqrt{2}W_t^i + \int_0^t n(\xi_s^{i,N})\mathrm{d}l_s^{i,N} \qquad (i = 1, \dots, N).$$

Here, $(W_t^1)_{t\geq 0}, \dots, (W_t^N)_{t\geq 0}$ are independent Brownian motions on Y and the initial particle values $\xi_0^1, \dots, \xi_0^N$ are independent and identically distributed. There and throughout the rest of this work, we denote the expectation of some appropriate function f with respect to a probability measure π by $\pi(f) := \int_X f(\theta)\pi(\mathrm{d}\theta)$. We use μ_t^N to denote the empirical distribution of the particles $(\xi_t^{1,N}, \dots, \xi_t^{N,N})$ at time $t \geq 0$. That is $\mu_t^N := \frac{1}{N}\sum_{i=1}^N \delta(\cdot - \xi_t^{i,N})$, where $\delta(\cdot - \xi)$ is the Dirac mass concentrated in $\xi \in B$. This implies especially that we can write

$$\mu_t^N(f) = \frac{1}{N}\sum_{i=1}^N f(\xi_t^{i,N}), \qquad \mathrm{Cov}_{\mu_t^N}(f, g) = \frac{1}{N}\sum_{i=1}^N f(\xi_t^{i,N})g(\xi_t^{i,N}) - \frac{1}{N^2}\sum_{i=1}^N\sum_{j=1}^N f(\xi_t^{i,N})g(\xi_t^{j,N}),$$

for appropriate functions f and g. The particles are constrained to stay within B by the last term in the equations of the $(\xi_t^{1,N}, \dots, \xi_t^{N,N})_{t\geq 0}$. Here, $n(x) = -x/\|x\|$ for $x \in \partial B$ is the inner normal vector field. Although we focus on Abram $(\theta_t^N, \xi_t^{1,N}, \dots, \xi_t^{N,N})_{t\geq 0}$, we remark that the solution of equations (3.1) is $(\theta_t^N, \xi_t^{1,N}, \dots, \xi_t^{N,N}, l_t^{1,N}, \dots, l_t^{N,N})_{t\geq 0}$. The functions $(l_t^{1,N}, \dots, l_t^{N,N})_{t\geq 0}$ are uniquely defined under the additional conditions:

(1) $l^{i,N}$'s are non-decreasing with $l^{i,N}(0) = 0$ and
(2) $\int_0^t \mathbf{1}[\xi_s^{i,N} \notin \partial B(\varepsilon)]dl^{i,N}(s) = 0$.

Condition (2) implies that $l^{i,N}$ can increase only when $\xi^{i,N}$ is in $\partial B(\varepsilon)$. Intuitively, $l^{i,N}$ cancels out part of $\xi^{i,N}$ so that it stays inside $B(\varepsilon)$. For more discussion on diffusion processes with reflecting boundary conditions, see e.g. [40]. Additionally, it is convenient to define

$$G(\theta, \nu) = \nabla_\theta\Big[\nu(\Phi(\cdot, \theta)) + \mathrm{Var}_\nu[\Phi(\cdot, \theta)]/2\Big] = \nu(\nabla_\theta \Phi(\cdot, \theta)) + \mathrm{Cov}_\nu(\Phi(\cdot, \theta), \nabla_\theta \Phi(\cdot, \theta)),$$

for any probability measure ν on $(B, \mathcal{B}B)$ and $\theta \in X$, where $\mathcal{B}B$ denotes the Borel-σ-algebra corresponding to B and, following the notation above, $\nu(\Phi(\cdot, \theta)) = \int_B \Phi(\xi, \theta)\nu(\mathrm{d}\xi)$. We finish this background section by defining the limiting McKean–Vlasov SDE with reflection

$$\theta_t = \theta_0 - \int_0^t \mu_s(\nabla_\theta \Phi(\cdot, \theta_s))\mathrm{d}s - \int_0^t \mathrm{Cov}_{\mu_s}(\Phi(\cdot, \theta_s), \nabla_\theta \Phi(\cdot, \theta_s))\mathrm{d}s, \tag{3.2}$$
$$\xi_t = \xi_0 + \int_0^t \nabla_x \Phi(\xi_s, \theta_s)\mathrm{d}s + \sqrt{2}W_t + \int_0^t n(\xi_s)\mathrm{d}l_s,$$

with μ_t denoting the law of ξ_t at time $t \geq 0$. The goal of this work is to show that the particle system (3.1) converges to this McKean–Vlasov SDEs as $N \to \infty$ and to then show that the McKean–Vlasov SDE can find the minimiser of F.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

3.2 Assumptions

We now list assumptions that we consider throughout this work. We start with the Lipschitz continuity of $\nabla\Phi$ and G.

Assumption 3.1 (Lipschitz). *The function $\nabla_\xi\Phi$ is Lipschitz continuous, i.e. there exists a Lipschitz constant $L > 1$ such that*

$$\left\|\nabla_\xi\Phi(\xi,\tilde\theta) - \nabla_\xi\Phi(\tilde\xi,\tilde\theta)\right\| \le L\Big(\left\|\xi-\tilde\xi\right\| + \left\|\theta-\tilde\theta\right\|\Big),$$

for any $\xi,\tilde\xi \in B$ and $\theta,\tilde\theta\in\mathbb{R}^n$. Similarly, we assume that $G(\theta,\mu)$ is Lipschitz in the following sense: there is an $L>1$ such that

$$\left\|G(\theta,\nu) - G(\tilde\theta,\tilde\nu)\right\| \le L\Big(\left\|\theta-\tilde\theta\right\| + \mathcal{W}_1(\nu,\tilde\nu)\Big),$$

for any probability measures ν, $\tilde\nu$ on $(B,\mathcal{B}B)$ and $\theta,\ \tilde\theta\in\mathbb{R}^n$. □

In Assumption 3.1 and throughout this work, $\mathcal{W}_p$ denotes the *Wasserstein-p distance* given by

$$\mathcal{W}_p^p(\nu,\nu') = \inf\left\{\int_{X\times X}\left\|y-y'\right\|^p\,\Gamma(\mathrm{d}y,\mathrm{d}y'):\Gamma \text{ is a coupling of } \nu,\nu'\right\},$$

for probability distributions ν,ν' on $(X,\mathcal{B}X)$ and $p\ge 1$. In addition to the Wasserstein distance, we sometimes measure the distance between probability distributions ν,ν' on $(X,\mathcal{B}X)$ using the *total variation distance* given by

$$\|\nu-\nu'\|_{\mathrm{TV}} = \sup_{A\in\mathcal{B}X}|\nu(A)-\nu'(A)|.$$

The Lipschitz continuity of G actually already implies the Lipschitz continuity of $\nabla_\theta\Phi$. By setting $\nu=\delta(\cdot-\xi)$ and $\tilde\nu=\delta(\cdot-\tilde\xi)$, we have

$$\begin{aligned}\left\|\nabla_\theta\Phi(\xi,\tilde\theta) - \nabla_\theta\Phi(\tilde\xi,\tilde\theta)\right\| &= \left\|G(\theta,\delta(\cdot-\xi)) - G(\tilde\theta,\delta(\cdot-\tilde\xi))\right\|\\ &\le L\Big(\left\|\theta-\tilde\theta\right\| + \mathcal{W}_1(\delta(\cdot-\xi),\delta(\cdot-\tilde\xi))\Big) = L\Big(\left\|\xi-\tilde\xi\right\| + \left\|\theta-\tilde\theta\right\|\Big).\end{aligned}$$

We assume throughout that the constant $L>1$ to simplify the constants in the Theorem 5.5. Finally, we note that Assumption 3.1 implies the well-posedness of both (3.1) and (3.2), see ([1], Theorems 3.1, 3.2).

Next, we assume the strong monotonicity of G, which, as we note below, also implies the strong convexity of $\Phi(x,\cdot)$ for any $x\in B$. This assumption is not realistic in the context of deep learning (e.g. [7]), but not unusual when analysing learning techniques.

Assumption 3.2 (Strong monotonicity). *For any probability measure ν on $(B,\mathcal{B}B)$, $G(\cdot,\nu)$ is 2λ-strongly monotone, i.e. for any $\theta,\tilde\theta\in\mathbb{R}^n$, we have*

$$\left\langle G(\theta,\nu) - G(\tilde\theta,\nu),\theta-\tilde\theta\right\rangle \ge 2\lambda\left\|\theta-\tilde\theta\right\|^2,$$

for some $\lambda>0$. □

By choosing $\nu=\delta(\cdot-\xi)$ in Assumption 3.2 for $\xi\in B$, we have $\mathrm{Cov}_\nu(\Phi(\cdot,\theta),\nabla_\theta\Phi(\cdot,\theta))=0$, which implies that $\langle\nabla_\theta\Phi(x,\theta)-\nabla_\theta\Phi(x,\theta'),\theta-\theta'\rangle \ge 2\lambda\|\theta-\theta'\|^2$. Thus, the 2λ-strong monotonicity of G in θ also implies the 2λ-strong convexity of Φ in θ.

The assumptions stated throughout this sections are fairly strong, they are satisfied in certain linear-quadratic problems on bounded domains. We illustrate this in an example below.

Example 3.3. *We consider a prototypical adversarial robustness problem based on the potential $\Phi(\xi,\theta) := \|\xi-\theta\|^2$ with θ in a bounded set $X'\subseteq X$ – problems of this form appear, e.g. in adversarially robust linear regression. Next, we are going to verify that this problem satisfies Assumptions 3.1 and 3.2.*

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

We have $\nabla_\xi \Phi(\xi,\theta) = 2(\xi-\theta)$, *which is Lipschitz in both* θ *and* ξ. *Since*

$$\nabla_\theta \Phi(\xi,\theta) = 2(\xi-\theta),$$
$$\Phi(\xi,\theta) - \int_B \Phi(\xi,\theta)\nu(\mathrm{d}\xi) = \left(\|\xi\|^2 - \int_B \|\xi\|^2\,\nu(\mathrm{d}\xi)\right) - 2\theta\cdot\left(\xi - \int_B \xi\,\nu(\mathrm{d}\xi)\right),$$
$$\nabla_\theta \Phi(\xi,\theta) - \int_B \nabla_\theta\Phi(\xi,\theta)\nu(\mathrm{d}\xi) = -2\left(\xi - \int_B \xi\,\nu(\mathrm{d}\xi)\right),$$

we have that

$$G(\theta,\nu) = 2\theta - 2\mathbb{E}_\nu[\xi] + 4\theta\cdot \mathrm{Var}_\nu(\xi) - 2\mathrm{Cov}_\nu(\|\xi\|^2,\xi),$$

where $\mathbb{E}_\nu[\xi] = \int_B \xi\,\nu(\mathrm{d}\xi)$ *and* $\mathrm{Cov}_\nu(\|\xi\|^2,\xi) = \int_B (\|\xi\|^2 - \mathbb{E}_\nu[\|\xi\|^2])(\xi - \mathbb{E}_\nu[\xi])\nu(\mathrm{d}\xi)$. *Since the* ε*-ball and* $\theta \in X'$ *are bounded, we have that* $G(\theta,\nu)$ *is Lipschitz in both* θ *and* ν. *Thus, it satisfies Assumption 3.1. In order to make* $G(\theta,\nu)$ *satisfy Assumption 3.2, we choose* ε *small enough such that the term* $4\theta\cdot\mathrm{Var}_\nu(\xi)$ *is* 1*-Lipschitz. In this case, we can verify that* $\langle G(\theta,\nu) - G(\theta',\nu), \theta-\theta'\rangle \geq \|\theta-\theta'\|^2$ *and, thus, Assumption 3.2.*

4 Propagation of chaos

We now study the large particle limit ($N\to\infty$) of the Abram dynamics (3.1). When considering a finite time interval $[0,T]$, we see that the particle system (3.1) approximates the McKean–Vlasov SDE (3.2) in this limit. We note that we assume in the following that $0<\varepsilon<1$. Moreover, we use the Wasserstein-2 distance instead of Wasserstein-1 distance in Assumption 3.1. We have $\mathcal{W}_1(\nu,\nu') \leq \mathcal{W}_2(\nu,\nu')$ for any probability measures ν,ν' for which the distances are finite, see [52]. Thus, convergence in $\mathcal{W}_2$ also implies convergence in $\mathcal{W}_1$. We now state the main convergence result.

Theorem 4.1. *Let Assumption 3.1 hold. Then, there is a constant* $C_{d,T}>0$ *such that for all* $T\geq 0$ *and* $N\geq 1$ *we have the following inequality*

$$\sup_{t\in[0,T]} \mathbb{E}\left[\left\|\theta_t^N - \theta_t\right\|^2 + \mathcal{W}_2^2(\mu_t^N,\mu_t)\right] \leq o_{d,T,N} := C_{d,T}\begin{cases} N^{-\alpha_d}, & \text{if } d\neq 4,\\ \log(1+N)N^{-\frac{1}{2}}, & \text{if } d=4,\end{cases}$$

where $\alpha_d = 2/d$ *for* $d>4$ *and* $\alpha_d = 1/2$ *for* $d<4$.

The dependence of d,T on $C_{d,T}$ is not explicit except in some special cases which we discuss in Section 5. The upper bound is essentially $N^{-2/d}+N^{-1/2}$ with the dominating term differing for $d>4$ and $d<4$. In fact, when $d<4$, the convergence rate can not be better than $N^{-1/2}$, see ([11], Page 2) for an example in which the lower bound is obtained.

Hence, we obtain convergence of both the gradient flow approximation $(\theta_t^N)_{t\geq 0}$ and the particle approximation (μ_t^N) to the respective components in the McKean–Vlasov SDE. We prove this result by a coupling method. To this end, we first collect a few auxiliary results: studying the large sample limit of an auxiliary particle system and the distance of the original particle system to the auxiliary system. To this end, we sample N trajectories of $(\xi_t)_{t\geq 0}$ from equations (3.2) as

$$\xi_t^i = \xi_0^i + \int_0^t \nabla_x\Phi(\xi_s^i,\theta_s)\mathrm{d}s + \sqrt{2}W_t^i + \int_0^t n(\xi_s^i)\mathrm{d}l_s^i \qquad (i=1,\dots,N), \tag{4.1}$$

where the Brownian motions $(W_t^1,\dots,W_t^N)_{t\geq 0}$ are the ones from (3.1). Of course these sample paths $(\xi_t^1,\dots,\xi_t^N)_{t\geq 0}$ are different from the $(\xi_t^{1,N},\dots,\xi_t^{N,N})_{t\geq 0}$ in equation (3.1): Here, $(\theta_t)_{t\geq 0}$ only depends on the law of $(\xi_t)_{t\geq 0}$, whereas $(\theta_t^N)_{t\geq 0}$ depends on position of the particles $(\xi_t^{i,N})_{t\geq 0}$. As the $(\xi_t^1)_{t\geq 0},\dots,(\xi_t^N)_{t\geq 0}$ are i.i.d., we can apply the empirical law of large numbers from [11] and get the following result.

Proposition 4.2. *Let Assumption 3.1 hold. Then,*

$$\sup_{t\geq 0}\mathbb{E}\left[\mathcal{W}_2^2\left(N^{-1}\sum_{i=1}^N \delta_{\xi_t^i},\mu_t\right)\right] \leq o_{d,T,N},$$

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

where $o_{d,T,N}$ is the constant given in Theorem 4.1.

For any $i = 1, \dots, N$, we are now computing bounds for the pairwise distances between ξ_t^i and $\xi_t^{i,N}$ for $t \geq 0$. We note again that these paths are pairwise coupled through the associated Brownian motions $(W_t^i)_{t\geq 0}$, respectively.

Lemma 4.3. *Let Assumption 3.1 hold and recall that $\xi_0^{i,N} = \xi_0^i$. Then,*

$$\left\|\xi_t^{i,N} - \xi_t^i\right\|^2 \leq 2L \int_0^t \left[\left\|\xi_s^{i,N} - \xi_s^i\right\|^2 + \left\|\theta_s^N - \theta_s\right\|^2 \right] \mathrm{d}s \qquad (i = 1, \dots, N),$$

for $t \in [0, T]$.

Proof. Recall that $(l_t^{1,N}, \dots, l_t^{N,N})_{t\geq 0}$ is non-decreasing in time and, hence, has finite total variation. We apply Itô's formula to $\left\|\xi_t^{i,N} - \xi_t^i\right\|^2$ and obtain

$$\begin{aligned}\left\|\xi_t^{i,N} - \xi_t^i\right\|^2 &= \underbrace{2\int_0^t \left\langle \xi_s^{i,N} - \xi_s^i, \nabla_x \Phi(\xi_s^{i,N}, \theta_s^N) - \nabla_x \Phi(\xi_s^i, \theta_s) \right\rangle \mathrm{d}s}_{(I1)} \\ &\quad + \underbrace{2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^{i,N} - \xi_s^i \right\rangle \mathrm{d}l_s^{i,N} - 2\int_0^t \left\langle n(\xi_s^i), \xi_s^{i,N} - \xi_s^i \right\rangle \mathrm{d}l_s^i}_{(I2)}.\end{aligned}$$

We first argue that $(I2) \leq 0$. Recall that $n(x) = -x/\|x\|$ and that the processes $(\xi_t^{i,N})_{t\geq 0}$ and $(\xi_t^i)_{t\geq 0}$ take values in the ε-ball B with $\varepsilon < 1$. Then, we have

$$\begin{aligned}2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^{i,N} - \xi_s^i \right\rangle \mathrm{d}l_s^{i,N} &= 2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^{i,N} \right\rangle \mathrm{d}l_s^{i,N} - 2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^i \right\rangle \mathrm{d}l_s^{i,N} \\ &= -2\varepsilon l_t^{i,N} - 2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^i \right\rangle \mathrm{d}l_s^{i,N} \leq -2\varepsilon l_t^{i,N} + 2\varepsilon \int_0^t \mathrm{d}l_s^{i,N} = 0,\end{aligned}$$

where the last inequality holds since $-2\int_0^t \left\langle n(\xi_s^{i,N}), \xi_s^i \right\rangle \mathrm{d}l_s^{i,N} \leq 2\int_0^t \left|\left\langle n(\xi_s^{i,N}), \xi_s^i \right\rangle\right| \mathrm{d}l_s^{i,N} \leq 2\varepsilon \int_0^t \mathrm{d}l_s^{i,N}$. Similarly, we have

$$-2\int_0^t \left\langle n(\xi_s^i), \xi_s^{i,N} - \xi_s^i \right\rangle \mathrm{d}l_s^i = 2\int_0^t \left\langle n(\xi_s^i), \xi_s^i - \xi_s^{i,N} \right\rangle \mathrm{d}l_s^i \leq 0.$$

Hence, we have $(I2) \leq 0$.

For $(I1)$, due to Assumption 3.1 and, again, due to the boundedness of B, we have

$$(I1) \leq L\int_0^t \left\|\xi_s^{i,N} - \xi_s^i\right\| \left[\left\|\xi_s^{i,N} - \xi_s^i\right\| + \left\|\theta_s^N - \theta_s\right\| \right] \mathrm{d}s \leq 2L\int_0^t \left[\left\|\xi_s^{i,N} - \xi_s^i\right\|^2 + \left\|\theta_s^N - \theta_s\right\|^2 \right] \mathrm{d}s.$$

□

Finally, we study the distance between θ_t^N and θ_t for $t \geq 0$.

Lemma 4.4. *Let Assumption 3.1 hold. Then, we have*

$$\left\|\theta_t^N - \theta_t\right\|^2 \leq 3L\int_0^t \left\|\theta_s^N - \theta_s\right\|^2 \mathrm{d}s + \frac{2L}{N}\sum_{i=1}^N \int_0^t \left\|\xi_s^{i,N} - \xi_s^i\right\|^2 \mathrm{d}s + 2L\int_0^t \mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s)\mathrm{d}s,$$

for $t \in [0, T]$.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Proof. Due to Assumption 3.1 and since $\mathcal{W}_1(\mu_s^N, \mu_s) \le \mathcal{W}_2(\mu_s^N, \mu_s)$, we have

$$\begin{aligned}\left\|\theta_t^N - \theta_t\right\|^2 &= -2\int_0^t \left\langle \theta_s^N - \theta_s, G(\theta_s^N, \mu_s^N) - G(\theta_s, \mu_s)\right\rangle \mathrm{d}s \\ &\le 2L\int_0^t \left\|\theta_s^N - \theta_s\right\| \Big(\left\|\theta_s^N - \theta_s\right\| + \mathcal{W}_2(\mu_s^N, \mu_s)\Big)\mathrm{d}s \\ &\le 2L\int_0^t \left\|\theta_s^N - \theta_s\right\|^2 \mathrm{d}s + 2L\int_0^t \left\|\theta_s^N - \theta_s\right\| \mathcal{W}_2(\mu_s^N, \mu_s)\mathrm{d}s \\ &\le 3L\int_0^t \left\|\theta_s^N - \theta_s\right\|^2 \mathrm{d}s + L\int_0^t \mathcal{W}_2^2(\mu_s^N, \mu_s)\mathrm{d}s. \end{aligned} \tag{4.2}$$

The triangle inequality implies that

$$\begin{aligned}\mathcal{W}_2^2(\mu_s^N, \mu_s) &\le 2\mathcal{W}_2^2(\mu_s^N, N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}) + 2\mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s) \\ &\le \frac{2}{N}\sum_{i=1}^N \left\|\xi_s^{i,N} - \xi_s^i\right\|^2 + 2\mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s). \end{aligned} \tag{4.3}$$

Combining (4.2) and (4.3), we obtain

$$\left\|\theta_t^N - \theta_t\right\|^2 \le 3L\int_0^t \left\|\theta_s^N - \theta_s\right\|^2 \mathrm{d}s + \frac{2L}{N}\sum_{i=1}^N \int_0^t \left\|\xi_s^{i,N} - \xi_s^i\right\|^2 \mathrm{d}s + 2L\int_0^t \mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s)\mathrm{d}s.$$

□

We now proceed to the proof of Theorem 4.1.

Proof of Theorem 4.1 We commence by constructing an upper bound for

$$u_t^N := N^{-1}\sum_{i=1}^N \left\|\xi_t^{i,N} - \xi_t^i\right\|^2 + \left\|\theta_t^N - \theta_t\right\|^2.$$

From Lemma 4.3 and Lemma 4.4, we have

$$u_t^N \le 5L\int_0^t u_s^N \mathrm{d}s + 2L\int_0^t \mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s)\mathrm{d}s.$$

Grönwall's inequality implies that

$$u_t^N \le 2Le^{5Lt}\int_0^t \mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s)\mathrm{d}s.$$

According to Proposition 4.2, we have

$$\mathbb{E}[u_t^N] \le 2Le^{5Lt}\int_0^t \mathbb{E}[\mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s)]\mathrm{d}s \le 2C_d Le^{(1+5L)t} o_{d,T,N},$$

whereas (4.3) implies

$$\left\|\theta_t^N - \theta_t\right\|^2 + \mathcal{W}_2^2(\mu_s^N, \mu_s) \le u_t^N + 2\mathcal{W}_2^2(N^{-1}\sum_{i=1}^N \delta_{\xi_s^i}, \mu_s).$$

Therefore,

$$\sup_{t\in[0,T]} \mathbb{E}\Big[\left\|\theta_t^N - \theta_t\right\|^2 + \mathcal{W}_2^2(\mu_t^N, \mu_t)\Big] \le \sup_{t\in[0,T]} \mathbb{E}[u_t^N] + \sup_{t\in[0,T]} \mathbb{E}\Big[\mathcal{W}_2^2(\mu_t^N, \mu_t)\Big] \le C_{d,T} o_{d,T,N},$$

where $C_{d,T} = 2C_d(1 + Le^{(1+5L)t})$.

□

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

5 Longtime behaviour of the McKean–Vlasov process

Theorem 4.1 implies that the gradient flow approximation in Abram $(\theta_t^N)_{t\geq 0}$ converges to the corresponding part of the McKean–Vlasov SDE $(\theta_t)_{t\geq 0}$ given in (3.2). In this section, we show that this McKean–Vlasov SDE is able to find the minimiser θ_* of $F = \int_{B(\varepsilon)} \Phi(\xi, \cdot)\pi^{\gamma,\varepsilon}(\mathrm{d}\xi|\cdot)$. This, thus, gives us a justification to use Abram to solve the Bayesian adversarial robustness problem. We start by showing that F admits a minimiser.

Proposition 5.1. *Let Assumptions 3.1 and 3.2 hold. Then, F admits at least one minimiser in X.*

Proof. We first argue that F is bounded below and obtains a minumum at some point θ_*. From Subsection 3.2, we already know that $\Phi(0,\theta)$ is 2λ-strongly convex in θ. Without loss of generality, we assume $\Phi(0,0) = 0$ and $\nabla_\theta \Phi(0,0) = 0$, that is $\Phi(0,\theta)$ reaches its minimum 0 at $\theta_* = 0$. Since $\Phi(\xi,\cdot)$ is 2λ strongly convex for any $\xi \in B$, we have that

$$\Phi(\xi,\theta) \geq \Phi(\xi,0) + \theta \cdot \nabla_\theta \Phi(\xi,0) + \lambda \|\theta\|^2 . \tag{5.1}$$

Assumption 3.1 implies that,

$$\|\nabla_\theta \Phi(\xi,0)\| = \|\nabla_\theta \Phi(\xi,0) - \nabla_\theta \Phi(0,0)\| \leq L\|\xi\| \leq L,$$

and

$$|\Phi(\xi,0)| = |\Phi(\xi,0) - \Phi(0,0)| \leq \sup_{\zeta\in B} \left\|\nabla_\xi \Phi(\zeta,0)\right\| \|\xi\| \leq (L + C_0)\|\xi\| \leq L + C_0,$$

where $C_0 = \left\|\nabla_\xi \Phi(0,0)\right\|$. Therefore, we have $\Phi(\xi,\theta) \geq -L - C_0 - L\|\theta\| + \lambda\|\theta\|^2$, which is bounded below by $-L - C_0 - \frac{L^2}{4\lambda}$. Thus, F is bounded below by the same value. We can always choose some $R_0 = R_0(L,\lambda,C_0)$, such that for $\|\theta\| \geq R_0$, $\Phi(\xi,\theta) \geq C_0 + L$. Moreover, we already have $\Phi(\xi,0) \leq L + C_0$. Thus, $F(\theta) \geq C_0 + L$ when $\|\theta\| \geq R_0$ and $F(0) \leq C_0 + L$. Hence, F attains its minimum on the R_0-ball $\{\theta \in X : \|\theta\| \leq R_0\}$.

□

Before stating the main theorem of this section – the convergence of the McKean–Vlasov SDE to the minimiser of F – we need to introduce additional assumptions.

Assumption 5.2 (Neumann Boundary Condition). *Let $\Phi(\cdot,\theta)$ satisfy a Neumann boundary condition on ∂B,*

$$\frac{\partial_\xi \Phi(\xi,\theta)}{\partial n} = \nabla_\xi \Phi(\xi,\theta)\cdot n(\xi) = 0,$$

for any $\theta \in X$. □

For a general function Φ defined on B, this assumption can be satisfied by smoothly extending Φ on B' with radius 2ε such that it vanishes near the boundary of B'. We shall see that this assumption guarantees the existence of the invariant measure of the auxiliary dynamical system (5.3) that we introduce below.

Assumption 5.3 (Small-Lipschitz). *For any probability measures ν, $\tilde{\nu}$ on $(B, \mathcal{B}B)$ and $\theta \in \mathbb{R}^n$,*

$$\|G(\theta,\nu) - G(\theta,\tilde{\nu})\| \leq \ell \|\nu - \tilde{\nu}\|_{TV},$$

where $\ell = (\frac{(\delta\wedge\lambda)\sqrt{\lambda}e^{-t_0}}{4\sqrt{2}CL}) \wedge (\frac{\sqrt{\lambda}}{\sqrt{2}L})$ and $t_0 = t_0(\delta,\lambda,C) = (\delta\wedge\lambda)^{-1}\log(4C)$. The constants δ and C appear in Proposition 5.6. □

Equivalently, we may say that this assumption requires G to have a small enough Lipschitz constant. If ε (the radius of B) is very small, this assumption is implied by Assumption 3.1, since $\mathcal{W}_1(\nu,\tilde{\nu}) \leq \varepsilon^d \int_B \int_B \mathbf{1}_{x\neq y}\pi(\mathrm{d}x,\mathrm{d}y) = \varepsilon^d \|\nu - \tilde{\nu}\|_{\mathrm{TV}}$.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

We illustrate these assumptions again in the linear-quadratic problem that we considered in Example 3.3 and show that Assumptions 5.2 and 5.3 can be satisfied in this case.

Example 5.4 (Example 3.3 continued). *We consider again* $\Phi(\xi,\theta)=\|\xi-\theta\|^2$ *with* θ *in a bounded* $X'\subseteq X$. *Unfortunately,* Φ *does not satisfy Assumption 5.2, since the term* $(\xi-\theta)\cdot\xi$ *is not necessary to be zero on the boundary of B. Instead, we study a slightly larger ball by considering* $\widehat{\varepsilon}=2\varepsilon$ *instead of* ε *and also replace* Φ *by* $\widehat{\Phi}(\xi,\theta)=\|m(\xi)-\theta\|^2$, *where* $m{:}\mathbb{R}^d\to\mathbb{R}^d$ *is smooth and equal to* ξ *on the* ε*-ball and vanishes near the boundary of the* 2ε*-ball. Since* $m(\xi)$ *varnishes near the boundary of* 2ε*-ball,* $\widehat{\Phi}$ *satisfies Assumption 5.2.*

We note that $\nabla_\xi\widehat{\Phi}(\xi,\theta)=2D_\xi m(\xi)(m(\xi)-\theta)$. *Hence,* $\nabla_\xi\widehat{\Phi}$ *is Lipschitz in both* θ *and* ξ *which directly follows from the boundedness and Lipschitz continuity of m,* $D_\xi m$. *Analogously to Example 3.3, we have*

$$G(\theta,\nu)=2\theta-2\mathbb{E}_\nu[m(\xi)]+4\theta\cdot\mathrm{Var}_\nu(m(\xi))-2\mathrm{Cov}_\nu(\|m(\xi)\|^2,m(\xi)),$$

and also see that it still satisfies Assumptions 3.1, 3.2 when θ *is bounded and* ε *is small. Finally, Assumption 5.3 is satisfied if* ε *is chosen to be sufficiently small.*

We are now able to state the main convergence theorem of this section. Therein, we still consider θ_* to be a minimiser of function of F.

Theorem 5.5. *Let Assumptions 3.1, 3.2, 5.2, and 5.3 hold and let* $(\theta_t,\mu_t)_{t\geq0}$ *be the solution to the McKean–Vlasov SDE (3.2). Then, there are constants* $\eta>0$ *and* $\tilde{C}>0$ *with which we have*

$$\|\theta_t-\theta_*\|^2+\|\mu_t-\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{TV}^2\leq\tilde{C}\Big(\|\theta_0-\theta_*\|^2+\|\mu_0-\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{TV}^2\Big)e^{-\eta t}. \tag{5.2}$$

We can see this result as both a statement about the convergence of $(\theta_t^N)_{t\geq0}$ to the minimiser, but also as an ergodicity statement about $(\theta_t^N,\xi_t)_{t\geq0}$. The ergodicity of a McKean–Vlasov SDE with reflection has also been subject of Theorem 3.1 in [53]. In their work, the process is required to have a non-degenerate diffusion term. Hence, their result does not apply immediately, since the marginal $(\theta_t)_{t\geq0}$ is deterministic (conditionally on $(\xi_t)_{t\geq0}$). Our proof ideas, however, are still influenced by [53].

We note additionally that Theorem 5.5 implies the uniqueness of the minimiser θ_* – we had only shown existence in Proposition 5.1: If there exists another minimiser θ_*', then the dynamics (3.2) is invariant at $(\theta_0,\xi_0)\sim\delta_{\theta_*'}\otimes\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*')$, which means $(\theta_t,\xi_t)\sim\delta_{\theta_*'}\otimes\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*')$ for all $t\geq0$. Hence, we have $\|\theta_*'-\theta_*\|\leq\tilde{C}\,\|\theta_*'-\theta_*\|\,e^{-\eta t}$. The right-hand side vanishes as $t\to\infty$, which implies $\theta_*'=\theta_*$.

In order to prove Theorem 5.5, we first consider the case where $\theta_t\equiv\theta_*$, i.e.

$$\widehat{\xi}_t=\xi_0+\int_0^t\nabla_x\Phi(\widehat{\xi}_s,\theta_*)\mathrm{d}s+\sqrt{2}W_t+\int_0^t\nu(\widehat{\xi}_s)\mathrm{d}\widehat{l}_s. \tag{5.3}$$

We denote the law of $\widehat{\xi}_t$ by $\widehat{\mu}_t$, $t\geq0$. Motivated by [53], we first show the exponential ergodicity for the process $(\widehat{\xi}_t)_{t\geq0}$.

Proposition 5.6. *Let Assumptions 3.1 and 5.2 hold. Then,* $(\widehat{\xi}_t)_{t\geq0}$ *defined in (5.3) is well-posed and admits an unique invariant measure* $\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)$. *Moreover,* $(\widehat{\xi}_t)_{t\geq0}$ *is exponentially ergodic. In particular, there exist* $C,\delta>0$, *such that*

$$\|\widehat{\mu}_t-\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{TV}^2\leq C\,\|\mu_0-\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{TV}^2\,e^{-\delta t}.$$

Proof. The well-posedness and exponential ergodicity is a direct corollary of ([53], Theorem 2.3). We only need to verify that $\pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)$ is invariant under the dynamics (5.3). We know that the probability distributions $(\widehat{\mu}_t)_{t\geq0}$ satisfies the following linear PDE with Neumann boundary condition

$$\partial_t\widehat{\mu}_t=\Delta\widehat{\mu}_t-\mathrm{div}(\widehat{\mu}_t\nabla_\xi\Phi(\xi,\theta_*)),\qquad\frac{\partial\widehat{\mu}_t}{\partial n}\Big|_{\partial B}=0.$$

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

So any invariant measure of the dynamics (5.3) is a probability distribution that solves the following stationary PDE

$$\Delta\widehat{\mu} - \operatorname{div}(\widehat{\mu}\nabla_\xi \Phi(\xi, \theta_*)) = 0, \qquad \frac{\partial \widehat{\mu}}{\partial n}\Big|_{\partial B} = 0.$$

Now, $\widehat{\mu} = \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)$ is a basic result in the theory of Langevin SDEs with reflection, see, e.g. [44]. □

Most of the time, we are not able to quantify the constants C and δ: the Harris-like theorem from [53] is not quantitative. A special case in which we can quantify C and δ is when the potential separates in the sense that $\nabla_\xi \Phi(\xi, \theta_*) = (f_1(\xi_1, \theta_*), \ldots, f_{d_Y}(\xi_{d_Y}, \theta_*))$. Then (5.3) can be viewed as d_Y independent reflection SDEs. If we denote their ergodicity constants as C_i and δ_i for $i = 1, \ldots, d_Y$, then ([22], Proof of Proposition 1) implies that we can choose $C := \sum_{i=1}^{d} C_i$ and $\delta := \min_{i=1,\ldots,d} \delta_i$. Thus, in this case, the constant C is linear in the dimension d.

Next, we bound the distance $\|\mu_t - \widehat{\mu}_t\|_{\mathrm{TV}}$ by Girsanov's theorem – a classical way to estimate the distance between two SDEs with different drift terms. This is again motivated by ([53], proof of Lemma 3.2). There, the method is used to bound the distance between two measure-dependent SDEs. In our case, it also involves the state θ_t, which depends on t. Hence, the right-hand side depends on the path of $(\theta_s)_{0\le s\le t}$.

Lemma 5.7. *Let Assumption 3.1 hold. Then, we have*

$$\|\mu_t - \widehat{\mu}_t\|_{TV}^2 \le L^2 \int_0^t \|\theta_s - \theta_*\|^2 \,\mathrm{d}s.$$

Proof. We follow the same idea as ([53], proof of Lemma 3.2). In our case, we need to choose

$$Z_t = \exp\Big(\int_0^t z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}W_s - \frac{1}{2}\int_0^t \|z(\theta_*, \theta_s, \xi_s)\|^2 \,\mathrm{d}s\Big),$$

where $z(\theta_*, \theta, x) = (\nabla_x \Phi(x, \theta) - \nabla_x \Phi(x, \theta_*))/\sqrt{2}$. ([29], Proposition 5.6) implies that the process $(Z_t)_{t\ge 0}$ is a martingale due to $z(\theta_*, \theta_s, \xi_s)$ being bounded and $\int_0^t \|z(\theta_*, \theta_s, \xi_s)\|^2 \,\mathrm{d}s$ being the quadratic variation process of $\int_0^t z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}W_s$.

We define the probability measure $\mathbb{Q}_t := Z_t\mathbb{P}$, i.e. $\mathbb{Q}_t(A) := \mathbb{E}[Z_t \mathbf{1}_A]$ for any $\mathcal{F}_t$-measurable set A. And we notice that the quadratic covariation between $\int_0^t z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}W_s$ and W_t is given by

$$\Big\langle \int_0^{\cdot} z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}W_s, W_{\cdot} \Big\rangle_t = \int_0^t z(\theta_*, \theta_s, \xi_s)\mathrm{d}s.$$

Hence by Girsanov's theorem (see ([29], Theorem 5.8, Conséquences (c))]), $\tilde{W}_t := W_t - \int_0^t z(\theta_*, \theta_s, \xi_s)\mathrm{d}s$ is a Brownian motion under $\mathbb{Q}_t$ with the same filtration $\mathcal{F}_t$.

We rewrite (3.2) as

$$\xi_t = \xi_0 + \int_0^t \nabla_x \Phi(\xi_s, \theta_*)\mathrm{d}s + \sqrt{2}\tilde{W}_t + \int_0^t n(\xi_s)\mathrm{d}l_s,$$

which has the same distribution as $\widehat{\xi}_t$ under $\mathbb{Q}_t$. Hence

$$\begin{aligned}
\|\mu_t - \widehat{\mu}_t\|_{\mathrm{TV}} &= \sup_{|f|\le 1} |\mathbb{E}[f(\xi_t)] - \mathbb{E}[f(\xi_t)Z_t]| \le \mathbb{E}[\,|Z_t - 1|\,] \\
&\le 2\mathbb{E}[R_t \log(R_t)]^{\frac{1}{2}} = 2\mathbb{E}_{Q_t}\Big[\int_0^t z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}W_s - \frac{1}{2}\int_0^t \|z(\theta_*, \theta_s, \xi_s)\|^2 \,\mathrm{d}s\Big]^{\frac{1}{2}} \\
&= 2\mathbb{E}_{Q_t}\Big[\int_0^t z(\theta_*, \theta_s, \xi_s) \cdot \mathrm{d}\tilde{W}_s + \frac{1}{2}\int_0^t \|z(\theta_*, \theta_s, \xi_s)\|^2 \,\mathrm{d}s\Big]^{\frac{1}{2}} \\
&= \sqrt{2}\mathbb{E}_{Q_t}\Big[\int_0^t \|z(\theta_*, \theta_s, \xi_s)\|^2 \,\mathrm{d}s\Big]^{\frac{1}{2}} \le L\Big(\int_0^t \|\theta_s - \theta_*\|^2 \,\mathrm{d}s\Big)^{\frac{1}{2}}.
\end{aligned}$$

where the first "≤" is implied by Pinsker's inequality. □

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Using these auxiliary results, we can now formulate the proof of Theorem 5.5.

Proof of Theorem 5.5 We take the time derivative of $\|\theta_t - \theta_*\|^2$,

$$\begin{aligned}\frac{\mathrm{d}\,\|\theta_t-\theta_*\|^2}{\mathrm{d}t} &= -\langle G(\theta_t,\mu_t) - G(\theta_*,\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)),\theta_t-\theta_*\rangle\\ &\le -2\lambda\,\|\theta_t-\theta_*\|^2 + \ell\,\|\theta_t-\theta_*\|\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}\\ &\le -\lambda\,\|\theta_t-\theta_*\|^2 + \frac{\ell^2}{\lambda}\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2,\end{aligned}$$

where the first "$\le$" is due to the ε-Young's inequality. This implies

$$\frac{\mathrm{d}(e^{\lambda t}\,\|\theta_t-\theta_*\|^2)}{\mathrm{d}t} \le \frac{\ell^2}{\lambda}e^{\lambda t}\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2.$$

Hence, we have

$$\|\theta_t-\theta_*\|^2 \le e^{-\lambda t}\,\|\theta_0-\theta_*\|^2 + \frac{\ell^2}{\lambda}\int_0^t \|\mu_s-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\,\mathrm{d}s. \tag{5.4}$$

Then, using the triangle inequality, we see that

$$\begin{aligned}\|\theta_t-\theta_*\|^2 + m\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 &\le \underbrace{\|\theta_t-\theta_*\|^2}_{(5.4)} + \underbrace{2m\,\|\mu_t-\widehat{\mu}_t\|_{\mathrm{TV}}^2}_{Lemma5.7} + \underbrace{2m\,\|\widehat{\mu}_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2}_{Proposition5.6}\\ &\le \int_0^t\Big(\frac{\ell^2}{\lambda}\,\|\mu_s-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 + 2mL^2\,\|\theta_s-\theta_*\|^2\Big)\mathrm{d}s\\ &\quad + 2C\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)e^{-(\delta\wedge\lambda)t}.\end{aligned}$$

Let $m = m(\ell, L, \lambda) = \frac{\ell}{L\sqrt{2\lambda}}$, we conclude from the above inequality that

$$\begin{aligned}\|\theta_t-\theta_*\|^2 + m\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 &\le 2mL^2\int_0^t\Big(m\,\|\mu_s-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 + \|\theta_s-\theta_*\|^2\Big)\mathrm{d}s\\ &\quad + 2C\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)e^{-(\delta\wedge\lambda)t}.\end{aligned}$$

Hence, by Grönwall's inequality, we have

$$\begin{aligned}&\|\theta_t-\theta_*\|^2 + m\,\|\mu_t-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\\ &\quad\le C\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)\Big(2mL^2\int_0^t e^{2mL^2(t-s)}e^{-(\delta\wedge\lambda)s}\mathrm{d}s + e^{-(\delta\wedge\lambda)t}\Big)\\ &\quad= C\Big(\frac{2mL^2}{2mL^2+\delta\wedge\lambda}(e^{2mL^2t}-e^{-(\delta\wedge\lambda)t}) + e^{-(\delta\wedge\lambda)t}\Big)\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)\\ &\quad\le C\Big(\frac{2mL^2}{\delta\wedge\lambda}e^{2mL^2t} + e^{-(\delta\wedge\lambda)t}\Big)\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)\\ &\quad\le C_{t,L,\ell,\lambda}\Big(\,\|\theta_0-\theta_*\|^2 + m\,\|\mu_0-\pi^{\gamma,\varepsilon}(\,\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big),\end{aligned} \tag{5.5}$$

where $C_{t,L,\ell,\lambda} = C\Big(\frac{2mL^2}{\delta\wedge\lambda}e^{2mL^2t} + e^{-(\delta\wedge\lambda)t}\Big)$. According to Assumption 5.3, we know that $2mL^2 = \frac{\ell L\sqrt{2}}{\sqrt{\lambda}} \le 1$. Next, we are going to show that $0 < C_{t_0,L,\ell,\lambda} \le 1/2$ for $t_0 = t_0(\delta,\lambda,C) = (\delta\wedge\lambda)^{-1}\log(4C)$. Again, from Assumption 5.3, we know $\frac{2mL^2}{\delta\wedge\lambda}e^{t_0} \le \frac{1}{4C}$. Hence we finally have,

$$C_{t_0,L,\ell,\lambda} \le \frac{2CmL^2}{\delta\wedge\lambda}e^{t_0} + Ce^{-(\delta\wedge\lambda)t_0} \le \frac{1}{2}.$$

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

For any $t \geq 0$, we always have $[\frac{t}{t_0}]t_0 \leq t < [\frac{t}{t_0}]t_0 + t_0$, where $[x]$ denotes the greatest integer $\leq x$. Hence,

$$\begin{aligned}\|\theta_t - \theta_*\|^2 + m\,\|\mu_t - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 &\leq 2^{-[\frac{t}{t_0}]}\Big(\Big\|\theta_{t-[\frac{t}{t_0}]t_0} - \theta_*\Big\|^2 + m\,\Big\|\mu_{t-[\frac{t}{t_0}]t_0} - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\Big\|_{\mathrm{TV}}^2\Big)\\ &\leq 2^{-\frac{t}{t_0}+1}\sup_{0\leq s\leq t_0}\Big(\|\theta_s - \theta_*\|^2 + m\,\|\mu_s - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)\\ &\leq 2^{-\frac{t}{t_0}+1}C\Big(\frac{e^{t_0}}{\delta\wedge\lambda}+1\Big)\Big(\|\theta_0 - \theta_*\|^2 + m\,\|\mu_0 - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big),\end{aligned}$$

where the last inequality is from (5.5) and $C_{s,L,\ell,\lambda}$ could be bounded by $C(\frac{e^{t_0}}{\delta\wedge\lambda}+1)$ for $0 \leq s \leq t_0$. And since $m \leq \frac{1}{2L^2} < 1$, we conclude that

$$\begin{aligned}\|\theta_t - \theta_*\|^2 + \|\mu_t - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2 &\leq m^{-1}2^{-\frac{t}{t_0}+1}C\Big(\frac{e^{t_0}}{\delta\wedge\lambda}+1\Big)\Big(\|\theta_0 - \theta_*\|^2 + \|\mu_0 - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big)\\ &\leq 2L^2 2^{-\frac{t}{t_0}+1}C\Big(\frac{e^{t_0}}{\delta\wedge\lambda}+1\Big)\Big(\|\theta_0 - \theta_*\|^2 + \|\mu_0 - \pi^{\gamma,\varepsilon}(\cdot\,|\theta_*)\|_{\mathrm{TV}}^2\Big).\end{aligned}$$

Finally, we choose the constants $\eta = \eta(\delta,\lambda,C) = \log(2)t_0^{-1} = (\delta\wedge\lambda)\frac{\log(2)}{\log(4C)}$ and $\tilde{C} = \tilde{C}(L,C,\delta,\lambda) = 4CL^2\Big(\frac{e^{t_0}}{\delta\wedge\lambda}+1\Big) = 4CL^2\Big(\frac{(4C)^{(\delta\wedge\lambda)^{-1}}}{\delta\wedge\lambda}+1\Big)$. □

6 Algorithmic considerations

Throughout this work, we have considered Abram as a continuous-time dynamical system. To employ it for practical adversarially robust machine learning, this system needs to be discretised, i.e. we need to employ a time stepping scheme to obtain a sequence $(\theta_k^N, \xi_k^{1,N}, \ldots, \xi_k^{N,N})_{k=1}^{\infty}$ that approximates Abram at discrete points in time. We now propose two discrete schemes for Abram, before then discussing the simulation of Bayesian adversarial attacks.

6.1 Discrete Abram

We initialise the particles by sampling them from the uniform distribution in the ε-ball. Then, we employ a projected Euler–Maruyama scheme to discretise the particles $(\xi_t^{1,N}, \ldots, \xi_t^{N,N})_{t\geq 0}$. The Euler–Maruyama scheme (see, e.g. [19]) is a standard technique for first order diffusion equation—we use a projected version to adhere to the reflecting boundary condition inside the ball B. Projected Euler–Maruyama schemes of this form have been studied in terms of almost sure convergence [49] and, importantly, also in terms of their longtime behaviour [27]. The gradient flow part $(\theta_t^N)_{t\geq 0}$ is discretised using a forward Euler method – turning the gradient flow into a gradient descent algorithm [39]. In applications, it is sometimes useful to allow multiple iterations of the particle dynamics $(\xi_t^{1,N}, \ldots, \xi_t^{N,N})_{t\geq 0}$ per iteration of the gradient flow $(\theta_t^N)_{t\geq 0}$. This corresponds to a linear time rescaling in the particle dynamics that should lead to a more accurate representation of the respective adversarial distribution.

If the number of data points $(y_k, z_k)_{k=1}^K$ is large, we may be required to use a data subsampling technique. Indeed, we approximate $\frac{1}{K}\sum_{k=1}^K \Phi(y_k, z_k|\theta) \approx \Phi(y_{k'}, z_{k'})$ with a $k' \sim \mathrm{Unif}(\{1,\ldots,K\})$ being sampled independently in every iteration of the algorithm. This gives us a stochastic gradient descent-type approximation of the gradients in the algorithm, see [42]. We note that we have not analysed data subsampling within Abram – we expect that techniques from [22, 28] may be useful to do so. We summarise the method in Algorithm 1.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Algorithm 1 Abram

initialise learning rate h, θ_0, γ, ε
for $j = 1, 2, \ldots, J$ **do**
  pick a data point (y_j, z_j) from training data
  for $i = 1, 2, \ldots, N$ **do**
    initialise $\xi_{0,j}^i = \xi_{T,j-1}^i$ if $j > 1$ else $\xi_{0,j}^i \sim \mathrm{Unif}[-\varepsilon, \varepsilon]$
    for $\tau = 1, 2, \ldots, T$ **do**
      $\xi_{\tau,j}^i \leftarrow \mathrm{Proj}_{\|\cdot\| \le \varepsilon}(\xi_{\tau-1,j}^i + h\nabla_\xi \Phi(y_j + \xi_{\tau-1,j}^i, z_j | \theta_{j-1}) + \gamma^{-1}\sqrt{2h} w_{\tau,j}^i)$ $(w_{\tau,j}^i \sim \mathrm{N}(0, \mathrm{Id})$ iid.)
    end for
  end for
  $\mu_j^N \leftarrow \frac{1}{N}\sum_{i=1}^N \delta(\cdot - \xi_{T,j}^i)$
  $\widehat{C}_j \leftarrow \mathrm{Cov}_{\mu_j^N}(\Phi(y_j + \cdot, z_j|\theta_{j-1}), \nabla_\theta \Phi(y_j + \cdot, z_j|\theta_{j-1}))$
  $\theta_j \leftarrow \theta_{j-1} - \frac{h}{N}\sum_{i=1}^N \nabla_\theta \Phi(y_j + \xi_{T,j}^i, z_j|\theta_{j-1}) - \gamma h \widehat{C}_j$
end for
return θ_J

Algorithm 2 Mini-batching Abram

initialise learning rate h, θ_0, γ, ε
for $j = 1, 2, \ldots, J$ **do**
  pick N data points $(y_j^i, z_j^i)_{i=1}^N$ from the training data $(y_k, z_k)_{k=1}^K$
  for $i = 1, 2, \ldots, N$ **do**
    initialise $\xi_{0,j}^i \leftarrow \xi_{T,j-1}^i$ if $j > 1$ else $\xi_{0,j}^i \sim \mathrm{Unif}[-\varepsilon, \varepsilon]$ iid.
    for $\tau = 1, 2, \ldots, T$ **do**
      $\xi_{\tau,j}^i \leftarrow \mathrm{Proj}_{\|\cdot\| \le \varepsilon}(\xi_{\tau-1,j}^i + h\nabla_\xi \Phi(y_j^i + \xi_{\tau-1,j}^i, z_j^i|\theta_{j-1}) + \gamma^{-1}\sqrt{2h} w_{\tau,j}^i)$ $(w_{\tau,j}^i \sim \mathrm{N}(0, \mathrm{Id})$ iid.)
    end for
  end for
  $\mu_j^N \leftarrow \frac{1}{N}\sum_{i=1}^N \delta(\cdot - (y_j^i + \xi_{T,j}^i))$
  $\widehat{C}_j \leftarrow \mathrm{Cov}_{\mu_j^N}(\Phi(\cdot, z_j|\theta_{j-1}), \nabla_\theta \Phi(\cdot, z_j|\theta_{j-1}))$
  $\theta_j \leftarrow \theta_{j-1} - \frac{h}{N}\sum_{i=1}^N \nabla_\theta \Phi(y_j^i + \xi_{T,j}^i, z_j|\theta_{j-1}) - \gamma h \widehat{C}_j$
end for
return θ_J

6.2 Discrete Abram with mini-batching

When subsampling in machine learning practice, it is usually advisable to choose mini-batches of data points rather than single data points. Here, we pick a mini-batch $\{y_{k'}, z_{k'}\}_{k' \in K'} \subseteq \{y_k, z_k\}_{k=1}^K$, with $\#K' \ll K$ and perform the gradient step with all elements with index in K' rather than a single element in the whole data set $\{y_k, z_k\}_{k=1}^K$. Abram would then require a set of N particles for each of the elements in the batch, i.e. NK' particles in total. In practice, N and K' are both likely to be large, leading to Abram becoming computationally infeasible. Based on an idea discussed in a different context in [17], we propose the following method: in every time step $j = 1, \ldots, J$ we choose an identical number of particles $(\xi_{T,j}^i)_{i=1}^N$ and data points $(y_j^i, z_j^i)_{i=1}^N$ in the mini-batch, i.e. $\#K' = N$. Then, we employ the Abram dynamics, but equip each particle $\xi_{T,j}^i$ with a different data point (y_j^i, z_j^i) $(i = 1, \ldots, N)$. As opposed to Abram with separate particles per data point, we here compute the sampling covariance throughout all subsampled

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Algorithm 3 Bayesian sample attack

Require: unperturbed input data set y

initialise h, γ, ε, $\xi_0 \sim \mathrm{Unif}[-\varepsilon, \varepsilon]$

for $j = 1, 2, \ldots, J$ **do**

  $\xi_j \leftarrow \mathrm{Proj}_{\|\cdot\| \le \varepsilon}(\xi_{j-1} + h\nabla_\xi \Phi(x + \xi_{j-1}, \theta) + \gamma^{-1}\sqrt{2h}w_j)$ $(w_j \sim \mathrm{N}(0, \mathrm{Id}))$

end for

return adversarially perturbed input data point $y + \xi_J$

Algorithm 4 Bayesian mean attack

Require: unperturbed input data point y

initialise h, γ, ε, $\xi_0 \sim \mathrm{Unif}[-\varepsilon, \varepsilon]$

for $j = 1, 2, \ldots, J$ **do**

  $\xi_j \leftarrow \mathrm{Proj}_{\|\cdot\| \le \varepsilon}(\xi_{j-1} + h\nabla_\xi \Phi(x + \xi_{j-1}, \theta) + \gamma^{-1}\sqrt{2h}w_j)$ $(w_j \sim \mathrm{N}(0, \mathrm{Id}))$

end for

return adversarially perturbed input data point $y + \frac{1}{J}\sum_{j=1}^{J} \xi_j$

data points rather than separately for every data point. The resulting dynamics are then only close to (3.1), if we assume that the adversarial attacks for each data point are not too dissimilar of each other. However, the dynamics may also be successful, if this is not the case. We summarise the resulting method in Algorithm 2.

6.3 Bayesian attacks

The mechanism used to approximate the Bayesian adversary in Algorithm 1 can naturally be used as a Bayesian attack. We propose two different attacks:

1. We use the projected Euler–Maruyama method to sample from the Bayesian adversarial distribution $\pi^{\gamma,\varepsilon}$ corresponding to an input dataset $y \in Y$ and model parameter θ^*. We summarise this attack in Algorithm 3.
2. Instead of attacking with a sample from $\pi^{\gamma,\varepsilon}$, we can attack with the mean of said distribution. From Proposition 5.6, we know that the particle system $(\widehat{\xi}_t)_{t\ge 0}$ that is based on a fixed parameter θ_*, is exponentially ergodic. Thus, we approximate the mean of $\pi^{\gamma,\varepsilon}$, by sampling $(\widehat{\xi}_t)_{t\ge 0}$ using projected Euler–Maruyama and approximate the mean by computing the sample mean throughout the sampling path. We summarise this method in Algorithm 4.

7 Deep learning experiments

We now study the application of Abram in deep learning. The model parameter θ is updated with batch size/number of particles N. For each particle in the ensemble, the perturbation parameter ξ is updated for T steps. Each experimental run is conducted on a single Nvidia A6000 GPU.

7.1 MNIST

We test Algorithm 1 and Algorithm 2 on the classification benchmark data set MNIST [30] against different adversarial attacks and compare the results with the results after an FGSM-based [57] adversarial training. We utilise the Adversarial Robustness Toolbox (ART) for the experiments,

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Table 2. *Comparison of test accuracy (%) on MNIST with different adversarial attack after Abram, mini-batching Abram, and FGSM [57] adversarial training*

Adversarial Attack ($\varepsilon = 0.1$)	Abram	Mini-batching Abram	FGSM
Benign Test	92.41±0.05	99.28±0.04	99.44±0.05
Auto-PGD	78.18±0.20	95.86±0.18	98.84±0.05
PGD	78.24±0.17	95.86±0.17	98.85±0.04
Wasserstein Attack	86.27±0.12	96.51±0.13	96.97±0.04
Carlini & Wagner Attack	8.76±0.015	5.14±0.1	62.60±0.02
Bayesian sample attack	92.43±0.10	99.29±0.03	99.44±0.06
Bayesian mean attack	92.42±0.08	99.28±0.04	99.44±0.05

see [37] for more details. ART is a Python library for adversarial robustness that provides various APIs for defence and attack. We use a neural network with two convolution layers each followed by a max pooling. In Algorithm 1, we set $\gamma = 1, h = \varepsilon, \varepsilon = 0.2$. In Algorithm 2, we set $\gamma = 1, h = 10\varepsilon, \varepsilon = 0.2$. We observe that setting larger noise scale for the attack during training helps Abram's final evaluation performance. We train the neural network for 30 epochs (i.e. 30 full iterations through the data set) for each method. The number of particles (and batch size) is $N = 128$, and the inner loop is trained for $T = 10$ times. To better understand how Abram responds to different attacks, we test against six attack methods: PGD [32], Auto-PGD [9], Carlini and Wagner [6], Wasserstein Attack [56], as well as the Bayesian attacks introduced in this paper – see Algorithms 3 and 4. We also test the method's accuracy in the case of benign (non-attacked) input data. For the Bayesian sample attack and Bayesian mean attack, we set $\gamma = 1000$. See Table 2 for the comparison. The results are averaged over three random seeds. We observe that Abram performs similarly to FGSM under Wasserstein, Bayesian sample and Bayesian mean attack. FGSM outperforms Abram under Auto-PGD, PGD and Carlini & Wagner attack. We conclude that Abram is as effective as FGSM under certain weaker attacks, but can usually not outperform the conventional FGSM.

Another observation is that mini-batching Abram outperforms Abram significantly. Recall that in Abram we have used 128 particles for each data point which can be viewed as SGD with batch size 1, whereas the mini-batching Abram is similar to the mini-batching SGD. Mini-batching Abram has the freedom to set the batch size which helps to reduce the variance in the stochastic optimisation and, thus, gives more stable results. In particular, with mini-batching Abram, gradients are approximated by multiple data points instead of one data point which is the case in Abram. Having a larger batch size also increases computation efficiency by doing matrix multiplication on GPUs, which is important in modern machine learning applications as the datasets can be expected to be large.

7.2 CIFAR10

Similarly, we test Algorithm 2 on the classification benchmark dataset CIFAR10 [23] by utilising ART. The dataset is pre-processed by random crop and random horizontal flip following [23] for data augmentation. The neural network uses the Pre-act ResNet-18 [18] architecture. For Abram, we set $\gamma = 1, h = \varepsilon, \varepsilon = 16/255$. Similar as in the MNIST experiments, practically we find that setting larger noise scale for attack in training Abram helps to obtain a better final evaluation performance. The batch size $N = 128$ and the inner loop is simulated for $T = 10$ times. We train both mini-batching Abram and FGSM for 30 epochs. Due to its worse performance for MNIST and the large size of CIFAR10, we have not used the non-mini-batching version of Abram in this second problem. For the Bayesian sample attack and the Bayesian mean attack, we set $\gamma = 1000$. We present the results in Table 3. There, we observe that mini-batching Abram outperforms FGSM under Wasserstein and the Bayesian attacks, but not in any of the other cases.

https://doi.org/10.1017/S0956792525000105 Published online by Cambridge University Press

Table 3. *Comparison of test accuracy (%) on CIFAR10 with different adversarial attack after mini-batching Abram and FGSM [57] adversarial training*

Adversarial Attack ($\varepsilon = 8/255$)	Mini-batching Abram	FGSM
Benign Test	65.35±0.05	55.61±0.03
Auto-PGD	11.15±0.12	43.70±0.06
PGD	11.22±0.09	43.65±0.04
Wasserstein Attack	58.04±0.15	55.30±0.03
Carlini & Wagner Attack	19.01±0.12	62.60±0.02
Bayesian sample attack	62.52±0.03	55.83±0.05
Bayesian mean attack	63.72±0.06	55.81±0.05

8 Conclusions

We have introduced the Bayesian adversarial robustness problem. This problem can be interpreted as either a relaxation of the usual minmax problem in adversarial learning or as learning methodology that is able to counter Bayesian adversarial attacks. To solve the Bayesian adversarial robustness problem, we introduce Abram – the Adversarially Bayesian Particle Sampler. Under restrictive assumptions, we prove that Abram approximates a McKean–Vlasov SDE and that this McKean–Vlasov SDE is able to find the minimiser of certain (simple) Bayesian adversarial robustness problems. Thus, at least for a certain class of problems, we give a mathematical justification for the use of Abram. We propose two ways to discretise Abram: a direct Euler–Maruyama discretisation of the Abram dynamics and an alternative method that is more suitable when training with respect to large data sets. We apply Abram in two deep learning problems. There we see that Abram can effectively prevent certain adversarial attacks (especially Bayesian attacks), but is overall not as strong as classical optimisation-based heuristics.

Competing interests. The authors declare none.

References

[1] Adams, D., dos Reis, G., Ravaille, R., Salkeld, W. & Tugaut, J. (2022) Large deviations and exit-times for reflected McKean–Vlasov equations with self-stabilising terms and superlinear drifts. *Stoch. Proc. Appl.* **146**, 264–310.

[2] Johnston, T., Crucinio, F. R., Akyildiz, Ö. D., Sabanis, S. & Girolami, M. (2023) Interacting particle Langevin algorithm for maximum marginal likelihood estimation. *ESAIM: PS* (forthcoming). DOI: 10.1051/ps/2025005

[3] Bachute, M. R. & Subhedar, J. M. (2021) Autonomous driving architectures: Insights of machine learning and deep learning algorithms. *Mach. Learn. Appl.* **6**, 100164. DOI: 10.1016/j.mlwa.2021.100164. https://www.sciencedirect.com/science/article/pii/S2666827021000827

[4] Bungert, L., Trillos, N. G., Jacobs, M., McKenzie, D., Nikolić, D. & Wang, Q. (2024) It begins with a boundary: A geometric view on probabilistically robust learning, *arXiv e-prints*, 2305.18779. URL https://arxiv.org/abs/2305.18779

[5] Carlini, N. & Wagner, D. (2017a) Adversarial examples are not easily detected: Bypassing ten detection methods, Association for Computing Machinery. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec'17, pp. 3–14, New York, NY, USA, 9781450352024.

[6] Carlini, N. & Wagner, D. (2017b) Towards evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy (SP), pp. 39–57, Los Alamitos, CA, USA: IEEE Computer Society.

[7] Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B. & LeCun, Y. (2015) The loss surfaces of multilayer networks. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, Vol. **38**, pp. 192–204, San Diego, CA, USA: PMLR.

[8] Cipriani, C., Scagliotti, A. & Wöhrer, T. (2024) A Minimax Optimal Control Approach for Robust Neural ODEs. In *2024 European Control Conference (ECC)*, Stockholm, Sweden, pp. 58–64. DOI: 10.23919/ECC64448.2024.10590973

[9] Croce, F. & Hein, M. (2020) Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In Proceedings of the 37th International Conference on Machine Learning, Vol. **119**, pp. 2206–2216, PMLR.

[10] Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X. & Li, J. (2018) Boosting adversarial attacks with momentum. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9185–9193, Los Alamitos, CA, USA: IEEE Computer Society.

[11] Fournier, N. & Guillin, A. (2015) On the rate of convergence in Wasserstein distance of the empirical measure. *Probab. Theory Relat. Fields* **162**(3–4), 707–738.
[12] Fu, S., He, F., Xu, Y. & Tao, D. (2021) Bayesian inference forgetting. *arXiv eprints*, 2101.06417. URL https://arxiv.org/abs/2101.06417
[13] Ghiasi, A., Shafahi, A. & Goldstein, T. (2020) Breaking certified defenses: Semantic adversarial examples with spoofed robustness certificates. In International Conference on Learning Representations.
[14] Goodfellow, I. J., Shlens, J. & Szegedy, C. (2015) Explaining and harnessing adversarial examples, In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, CA, USA, May 7-9, 2015. URL https://arxiv.org/abs/1412.6572
[15] Gowal, S., Dvijotham, K., Stanforth, R., Bunel, R.,Qin, C., Uesato, J., Arandjelovic, R., Mann, T. A. & Kohli, P. (2019) Scalable verified training for provably robust image classification. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 4841–4850.
[16] Guo, C., Rana, M., Cisse, M. & van der Maaten, L. (2018) Countering adversarial images using input transformations. In International Conference on Learning Representations.
[17] Hanu, M., Latz, J. & Schillings, C. (2023) Subsampling in ensemble Kalman inversion. *Inverse Probl.* **39**(9), 094002. DOI: 10.1088/1361-6420/ace64b. URL https://dx.doi.org/10.1088/1361-6420/ace64b
[18] He, K., Zhang, X., Ren, S. & Sun, J. (2016) Identity mappings in deep residual networks. In Computer Vision – ECCV 2016, pp. 630–645, Springer International Publishing.
[19] Higham, D. & Kloeden, P. (2021) *An introduction to the numerical simulation of stochastic differential equations*. SIAM.
[20] Hwang, C.-R. (1980) Laplace's method revisited: Weak convergence of probability measures. *Ann. Probab.* **8**(6), 1177–1182. URL http://www.jstor.org/stable/2243019
[21] Jia, J., Qu, W. & Gong, N. Z. (2022) Multiguard: Provably robust multi-label classification against adversarial examples. In Advances in Neural Information Processing Systems.
[22] Jin, K., Latz, J., Liu, C. & Schönlieb, C.-B. (2023) A continuous-time stochastic gradient descent method for continuous data. *J. Mach. Learn. Res.* **24**(274), 1–48.
[23] Krizhevsky, A. (2009) Learning multiple layers of features from tiny images, Technical Report. URL https://www.cs.toronto.edu/ kriz/learning-features-2009-TR.pdf
[24] Kuntz, J., Lim, J. N. & Johansen, A. M. (2023) Particle algorithms for maximum likelihood training of latent variable models. In Proceedings of The 26th International Conference on Artificial Intelligence and Statistics, Vol. **206** of Proceedings of Machine Learning Research, pp. 5134–5180, PMLR.
[25] Kurakin, A., Goodfellow, I. J. & Bengio, S. (2017a) Adversarial examples in the physical world. In Artificial Intelligence Safety and Security, pp. 99–112.
[26] Kurakin, A., Goodfellow, I. J. & Bengio, S. (2017b) Adversarial machine learning at scale. In International Conference on Learning Representations. URL https://openreview.net/forum?id=BJm4T4Kgx
[27] Lamperski, A. (2021) Projected stochastic gradient Langevin algorithms for constrained sampling and non-convex learning. In Belkin, M. & Kpotufe, S. (eds.), Proceedings of 34th Conference on Learning Theory, Vol. **134** of Proceedings of Machine Learning Research, pp. 2891–2937, PMLR. URL https://proceedings.mlr.press/v134/lamperski21a.html
[28] Latz, J. (2021) Analysis of stochastic gradient descent in continuous time. *Stat. Comput.* **31**(4), 39. DOI: 10.1007/s11222-021-10016-8
[29] Gall, J.-F. Le (2013) *Mouvement Brownien, Martingales et Calcul Stochastique*, Springer.
[30] LeCun, Y. & Cortes, C. (2005) The MNIST database of handwritten digits. URL http://yann.lecun.com/exdb/mnist
[31] Liu, J., Levine, A., Lau, C., Chellappa, R. & Feizi, S. (2022) Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 14953–14962.Los Alamitos, CA, USA: IEEE Computer Society.
[32] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. & Vladu, A. (2018) Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations.
[33] Maini, P., Wong, E. & Kolter, Z. (2020) Adversarial robustness against the union of multiple perturbation models. In H., D.III & Singh, A. (eds.), Proceedings of the 37th International Conference on Machine Learning, Vol. **119** of Proceedings of Machine Learning Research, pp. 6640–6650, PMLR.
[34] McKean, H. P. (1966) A class of Markov processes associated with nonlinear parabolic equations. *Proc. Natl. Acad. Sci.* **56**(6), 1907–1911. DOI: 10.1073/pnas.56.6.1907 URL https://www.pnas.org/doi/abs/10.1073/pnas.56.6.1907
[35] Metzen, J. H., Genewein, T., Fischer, V. & Bischoff, B. (2017) On detecting adversarial perturbations. In International Conference on Learning Representations.
[36] Mosbach, M., Andriushchenko, M., Trost, T., Hein, M. & Klakow, D. (2019) Logit pairing methods can fool gradient-based attacks. In *NeurIPS 2018 Workshop on Security in Machine Learning*. URL https://arxiv.org/abs/1810.12042
[37] Nicolae, M.-I., Sinn, M., Tran, M. N., Buesser, B., Rawat, A., Wistuba, M., Zantedeschi, V., Baracaldo, N., Chen, B., Ludwig, H., Molloy, I., & Edwards, B. (2018) Adversarial robustness toolbox v1.2.0, *arXiv eprints,* 1807.01069, 2018.
[38] Nie, W., Guo, B., Huang, Y., Xiao, C., Vahdat, A. & Anandkumar, A. (2022) Diffusion models for adversarial purification. In Proceedings of the 39th International Conference on Machine Learning, Vol. **162** of Proceedings of Machine Learning Research, pp. 16805–16827, PMLR.
[39] Nocedal, J. & Wright, S. J. (1999) *Numerical Optimization*, Springer.
[40] Pilipenko, A. (2014) *An introduction to stochastic differential equations with reflection*. Universität Potsdam, Lectures in Pure and Applied Mathematics

[41] Rajkomar, A., Dean, J. & Kohane, I. (2019) Machine learning in medicine. *New England J. Med.* **380**(14), 1347–1358. DOI: 10.1056/NEJMra1814259 URL https://www.nejm.org/doi/full/10.1056/NEJMra1814259
[42] Robbins, H. & Monro, S. (1951) A stochastic approximation method. *Ann. Math. Stat.* **22**(3), 400–407.
[43] Robey, A., Chamon, L., Pappas, G. J. & Hassani, H. (2022) Probabilistically robust learning: Balancing average and worst-case performance, Proceedings of the 39th International Conference on Machine Learning, Vol. **162** of Proceedings of Machine Learning Research, pp. 18667–18686, PMLR.
[44] Sato, K., Takeda, A., Kawai, R. & Suzuki, T. (2024) Convergence error analysis of reflected gradient Langevin dynamics for non-convex constrained optimization. *Japan J. Indust. Appl. Math.* **42**, 127–151. DOI: 10.1007/s13160-024-00667-1
[45] Sharma, S., Bhatt, M. & Sharma, P. (2020) Face recognition system using machine learning algorithm. In 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 1162–1168. DOI: 10.1109/ICCES48766.2020.9137850
[46] Sheikholeslami, F., Lotfi, A. & Kolter, J. Z. (2021) Provably robust classification of adversarial examples with detection. In International Conference on Learning Representations.
[47] Song, Y., Kim, T., Nowozin, S., Ermon, S. & Kushman, N. (2018) Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In International Conference on Learning Representations.
[48] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J. & Fergus, R. (2014) Intriguing properties of neural networks, In *International Conference on Learning Representations*. URL https://arxiv.org/abs/1312.6199
[49] Słomiński, L. (1994) On approximation of solutions of multidimensional SDE's with reflecting boundary conditions. *Stoch. Proc. Appl.* **50**(2), 197–219. DOI: 10.1016/0304-4149(94)90118-X, https://www.sciencedirect.com/science/article/pii/030441499490118X
[50] Tramer, F. & Boneh, D. (2019) Adversarial training and robustness for multiple perturbations, Advances in Neural Information Processing Systems, Vol. **32**.
[51] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D. & McDaniel, P. (2018) Ensemble adversarial training: Attacks and defenses. In International Conference on Learning Representations.
[52] Villani, C. (2009) *Optimal Transport: Old and New*. Springer.
[53] Wang, F.-Y. (2023) Exponential ergodicity for singular reflecting McKean–Vlasov SDEs. *Stoch. Proc. Appl.* **160**, 265–293.
[54] Wang, Z., Pang, T., Du, C., Lin, M., Liu, W. & Yan, S. (2023) Better diffusion models further improve adversarial training. In Proceedings of the 40th International Conference on Machine Learning, Vol. **202** of Proceedings of Machine Learning Research, pp. 36246–36263, PMLR.
[55] Wong, E. & Kolter, Z. (2018) Provable defenses against adversarial examples via the convex outer adversarial polytope. In Proceedings of the 35th International Conference on Machine Learning, Vol. **80** of Proceedings of Machine Learning Research, pp. 5286–5295, PMLR.
[56] Wong, E., Schmidt, F. & Kolter, Z. (2019) Wasserstein adversarial examples via projected Sinkhorn iterations. In Proceedings of the 36th International Conference on Machine Learning, Vol. **97** of Proceedings of Machine Learning Research, pp. 6808–6817, PMLR.
[57] Wong, E., Rice, L. & Kolter, J. Z. (2020) Fast is better than free: Revisiting adversarial training. In International Conference on Learning Representations.
[58] Xu, K., Xiao, Y., Zheng, Z., Cai, K. & Nevatia, R. (2023) Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 4621–4630. Los Alamitos, CA, USA: IEEE Computer Society.
[59] Xu, W., Evans, D. & Qi, Y. (2018) Feature squeezing: Detecting adversarial examples in deep neural networks. In Network and Distributed System Security Symposium.
[60] Xue, H., Araujo, A., Hu, B. & Chen, Y. (2023) Diffusion-based adversarial sample generation for improved stealthiness and controllability. In Advances in Neural Information Processing Systems, Vol. **36**, pp. 2894–2921.
[61] Yang, Y., Zhang, G., Katabi, D. & Xu, Z. (2019) ME-net: Towards effective adversarial robustness with matrix estimation. In Proceedings of the 36th International Conference on Machine Learning, Vol. **97** of Proceedings of Machine Learning Research, pp. 7025–7034, PMLR.
[62] Ye, N. & Zhu, Z. (2018) Bayesian adversarial learning. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N. & Garnett, R. (eds.), Advances in Neural Information Processing Systems, Vol. **31**.
[63] Yun, S., Han, D., Chun, S., Oh, S., Yoo, Y. & Choe, J. (2019) Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6022–6031. Los Alamitos, CA, USA: IEEE Computer Society.

Cite this article: Ding Z., Jin K., Latz J. and Liu C. How to beat a Bayesian adversary. *European Journal of Applied Mathematics*, https://doi.org/10.1017/S0956792525000105