# Iterated Vector Fields and Conservatism, with Applications to Federated Learning

**Zachary Charles** ZACHCHARLES@GOOGLE.COM
**Keith Rush** KRUSH@GOOGLE.COM
*Google Research*

**Editors:** Sanjoy Dasgupta and Nika Haghtalab

## Abstract

We study whether iterated vector fields (vector fields composed with themselves) are conservative. We give explicit examples of vector fields for which this self-composition preserves conservatism. Notably, this includes gradient vector fields of loss functions associated to some generalized linear models. In the context of federated learning, we show that when clients have loss functions whose gradient satisfies this condition, federated averaging is equivalent to gradient descent on a surrogate loss function. We leverage this to derive novel convergence results for federated learning. By contrast, we demonstrate that when the client losses violate this property, federated averaging can yield behavior which is fundamentally distinct from centralized optimization. Finally, we discuss theoretical and practical questions our analytical framework raises for federated learning.

**Keywords:** Optimization, federated learning, dynamical systems, parallel and distributed algorithms

## 1. Introduction

In this work, we consider vector fields of the form $V : \mathbb{R}^n \to \mathbb{R}^n$. Recall that $V$ is conservative if there is some function $f : \mathbb{R}^n \to \mathbb{R}$ such that $V = \nabla f$. We are interested in whether *iterated* vector fields (vector fields of the form $V \circ V \circ \cdots \circ V$) are conservative. This question has rich connections to a variety of areas, including differential geometry, dynamical systems, and optimization. As we will show, conservative iterated vector fields are particularly important for understanding optimization algorithms for federated learning.

**Notation.** Let $\mathcal{V}(\mathbb{R}^n, \mathbb{R}^m)$ denote the collection of functions from $\mathbb{R}^n$ to $\mathbb{R}^m$. We let $\mathcal{D}(\mathbb{R}^n, \mathbb{R}^m)$ denote the subset of differentiable functions, and $\mathcal{C}^k(\mathbb{R}^n, \mathbb{R}^m)$ denote the subset of $\mathcal{C}^k$ functions. If $m = n$, we abbreviate these by $\mathcal{V}(\mathbb{R}^n)$, $\mathcal{D}(\mathbb{R}^n)$ and $\mathcal{C}^k(\mathbb{R}^n)$. Throughout, $\|\cdot\|$ denotes the $\ell_2$ norm on $\mathbb{R}^n$ with corresponding inner product $\langle \cdot, \cdot \rangle$. We let $I \in \mathcal{V}(\mathbb{R}^n)$ denote the identity map.

Given $V \in \mathcal{V}(\mathbb{R}^n)$, we use exponents to denote repeated iterations of $V$. That is, for $k \geq 1$ we define:

$$V^k(x) := \underbrace{V \circ V \circ \cdots \circ V}_{\text{k times}}(x)$$

By convention, for any $V \in \mathcal{V}(\mathbb{R}^n)$ we define $V^0 := I$.

**Summary.** Let $V \in \mathcal{V}(\mathbb{R}^n)$, and let $k$ be a positive integer. We study the following question.

**Question 1** *If $V$ is conservative, is $V^k$ also conservative?*

This leads to the following definition.

**Definition 1** *$V$ is $k$-conservative if $V^k$ is conservative. $V$ is $\infty$-conservative if $V^k$ is conservative for all $k \geq 1$.*

For convenience, we use "conservative" and "1-conservative" interchangeably. In a slight abuse of notation, we say that $\mathcal{A} \subseteq \mathcal{V}(\mathbb{R}^n)$ is $k$-conservative if for all $V \in \mathcal{A}$, $V$ is $k$-conservative. In order to show that $\mathcal{A}$ is $\infty$-conservative, it suffices to show that $\mathcal{A}$ is conservative and closed under self-composition, as reflected in the following definition.

**Definition 2** *$\mathcal{A} \subseteq \mathcal{V}(\mathbb{R}^n)$ is closed under self-composition if for all $V \in \mathcal{A}$ and $k \geq 1$, $V^k \in \mathcal{A}$.*

This leads us to the following specialization of Question 1.

**Question 2** *Let $\mathcal{A} \subseteq \mathcal{V}(\mathbb{R}^n)$ be conservative. Is $\mathcal{A}$ closed under self-composition?*

**Vector Fields and Optimization.** Motivated by optimization, we will often consider vector fields of the form $V(x) = \nabla f(x)$, where $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable. Given $\mathcal{F} \subseteq \mathcal{D}(\mathbb{R}^n, \mathbb{R})$, we define $\nabla\mathcal{F} = \{V \in \mathcal{V}(\mathbb{R}^n) : V = \nabla f, f \in \mathcal{F}\}$. For $\gamma \in \mathbb{R}$, we define $I - \gamma\nabla\mathcal{F} := \{I - \gamma\nabla f : f \in \mathcal{F}\}$. A recurring theme in this work is whether a set $I - \gamma\nabla\mathcal{F}$ is $k$-conservative. Such vector fields arise naturally in optimization, as gradient descent on a function $f$ with learning rate $\gamma$ corresponds to the discrete-time dynamical system given by $x_{t+1} = (I - \gamma\nabla f)(x_t)$.

Given an initial point $x_0$, the iterates of gradient descent then satisfy $x_k = V^k(x_0)$ where $V = I - \gamma\nabla f$. Therefore, if $I - \gamma\nabla f$ is $\infty$-conservative, then the $k$-th iterate of gradient descent is actually $\nabla h_k(x_0)$ for some function $h_k : \mathbb{R}^n \to \mathbb{R}$. While this observation is essentially trivial for centralized optimization, it will prove much more useful when trying to understand the behavior of distributed and federated optimization algorithms, as we discuss below.

## 2. Connections to Federated Learning

Questions about whether a vector field is $k$-conservative have important implications for federated learning, one noteworthy approach to which is *federated averaging* (FEDAVG) (McMahan et al., 2017). A slightly simplified version of FEDAVG operates as follows. Suppose we have clients $c = 1, 2, \ldots, C$, each with loss function $f_c : \mathbb{R}^n \to \mathbb{R}$. At each round, the server broadcasts its model the clients. The clients perform $k$ steps of gradient descent (with learning rate $\gamma$) on their loss functions, and send the resulting models to the server. The server updates its model as the average of these client models. Since communication from clients to the server is often a bottleneck (McMahan et al., 2017; Bonawitz et al., 2019), this algorithm is often practical only when $k > 1$. When $k = 1$, this is equivalent to gradient descent with learning rate $\gamma$ on the average of the client loss functions.

More formally, let $V_c := I - \gamma\nabla f_c$. At each round $t$, each client computes $V_c^k(x_t)$, and the server updates its model via $x_{t+1} = C^{-1}\sum_{c=1}^{C} V_c^k(x_t)$. This "operator-theoretic" view of FEDAVG has been previously used to leverage techniques from operator theory to analyze and design federated learning algorithms (Malinovskiy et al., 2020; Pathak and Wainwright, 2020; Malekmohammadi et al., 2021). In order to allow the server to determine the magnitude of its update at each step, Reddi et al. (2021) propose a "model delta" version of FEDAVG. This corresponds to the server update

$$x_{t+1} = x_t - \frac{\eta}{C}\sum_{c=1}^{C}\Big(x_t - V_c^k(x_t)\Big) \tag{1}$$

where $\eta > 0$ is the server learning rate, which we may set to 1 to recover the average of the client models. In the sequel we let FEDAVG denote the update rule in (1). If we let $V_s$ be the "server" vector field given by

$$V_s = \frac{1}{C}\sum_{c=1}^{C}(I - V_c^k) \tag{2}$$

then (1) is equivalent to

$$x_{t+1} = x_t - \eta V_s(x_t). \tag{3}$$

This leads us to our guiding observation: If each $V_c$ is $k$-conservative, then $V_s$ is an average of conservative vector fields and is conservative as well. Therefore, there is some function $f_s$ such that $\nabla f_s = V_s$, and (3) is equivalent to $x_{t+1} = x_t - \eta\nabla f_s(x_t)$ and FEDAVG is equivalent to applying gradient descent to the function $f_s$ (see Theorem 11 for a formal statement of this). This allows us to reduce the behavior of FEDAVG to the behavior of gradient descent on this "surrogate loss" $f_s$. Such an approach was used by Charles and Konečný (2021) to understand the dynamics of FEDAVG and related methods on quadratic functions. In this work, we consider more general functions, including some non-convex functions.

### 2.1. Non-Conservative Dynamics in Federated Learning

As we sketched in the section above (and formalize in Section 6), when the vector fields $I - \gamma\nabla f_c$ are $k$-conservative, FEDAVG with $k$ local steps behaves identically to gradient descent on some surrogate loss function. On the other hand, in this section we show that without $k$-conservatism, FEDAVG can demonstrate fundamentally non-conservative behavior, making its dynamics distinct from those of gradient descent. Notably, this can occur even when $C = 2$ and there is no stochasticity whatsoever. For example, for $c \in \{1, 2\}$, consider the client loss functions

$$f_c(x, y) := f_c^{(1)}(x, y) + f_c^{(2)}(x, y) \tag{4}$$

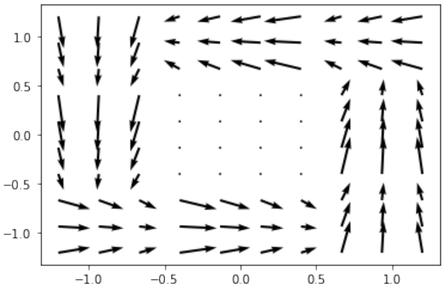
where

$$f_c^{(1)}(x, y) := \min\left(\frac{\alpha_c}{2}(y - y_c)^2 + \frac{\beta_c}{2}(x - x_c)^2, 1\right),$$
$$f_c^{(2)}(x, y) := \min\left(\frac{\alpha_c}{2}(y + y_c)^2 + \frac{\beta_c}{2}(x + x_c)^2, 1\right).$$

Notably, $I - \gamma\nabla f_c$ may not be $k$-conservative for $k > 1$. As we show in Appendix C, for some choice of $\alpha_c, \beta_c \in \mathbb{R}$, $x_c, y_c \in \mathbb{R}^2$ (for $c = 1, 2$), $\gamma > 0$ and $k$ sufficiently large, the resulting server vector field $V_s$ in (2) is non-conservative.

To help illustrate this, in Fig. 1, we plot this non-conservative server vector field $V_s$. Note there is a region of initial points $x_0$ under which the dynamics of FEDAVG are entirely circular and periodic, as long as $\eta$ is sufficiently small. In short, FEDAVG may behave badly in the absence of $k$-conservatism.

Figure 1: Non-conservative server vector field $V_s$ induced by $f_1, f_2$ in (4) for $k$ sufficiently large.



## 3. Examples of $k$-Conservative Vector Fields

We now give concrete examples of $k$-conservative vector fields. These include vector fields associated to linear and logistic regression. Let $\mathcal{P}_d(\mathbb{R}^n, \mathbb{R}^m)$ denote the subset of $\mathcal{V}(\mathbb{R}^n, \mathbb{R}^m)$ whose coordinate functions are homogeneous polynomials of degree $d$. We abbreviate this as $\mathcal{P}_d(\mathbb{R}^n)$ when $n = m$. For more in-depth examples, see Appendix A.

**Constant Vector Fields.** The space $\mathcal{P}_0(\mathbb{R}^n)$ of constant vector fields is clearly closed under self-composition. Constant vector fields are conservative, so $\mathcal{P}_0(\mathbb{R}^n)$ is $\infty$-conservative.

**Affine Vector Fields.** Let $\mathcal{A}(\mathbb{R}^n)$ be the set of affine vector fields in $\mathcal{V}(\mathbb{R}^n)$. This consists of all $V$ of the form $V(x) = Ax + b$ for $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$. Let $\mathcal{S}(\mathbb{R}^n)$ denote the set of such $V$ where $A$ is symmetric. If $V \in \mathcal{A}(\mathbb{R}^n)$ is conservative, it is the gradient of some quadratic function. Simple algebraic manipulation then implies that $V$ is conservative iff $A$ is symmetric. Since $\mathcal{S}(\mathbb{R}^n)$ is closed under self-composition, $\mathcal{S}(\mathbb{R}^n)$ is $\infty$-conservative while $\mathcal{A}(\mathbb{R}^n)$ is not conservative. In particular, if $f$ is a quadratic function, $\nabla f$ and $I - \gamma\nabla f$ are both $\infty$-conservative.

**Continous Univariate Functions.** Consider the set $\mathcal{C}^0(\mathbb{R})$ of continuous functions from $\mathbb{R}$ to $\mathbb{R}$. By elementary analysis, $\mathcal{C}^0(\mathbb{R})$ is closed under self-composition, and by the fundamental theorem of calculus, it is conservative. Thus, $\mathcal{C}^0(\mathbb{R})$ is $\infty$-conservative.

More generally, let $\mathcal{C}^0(\mathbb{R})^n$ denote the subset of $\mathcal{V}(R^n)$ containing vector fields of the form

$$V(x_1, \dots, x_n) = (f_1(x_1), f_2(x_2), \dots, f_n(x_n))$$

where $f_1, \dots, f_n \in \mathcal{C}^0(\mathbb{R})$. Then note that $V(x_1, \dots, x_n) = \nabla\left(\sum_{i=1}^n \int_0^{x_i} f_i(t)dt\right)$ so $\mathcal{C}^0(\mathbb{R})^n$ is conservative. Since $\mathcal{C}^0(\mathbb{R})^n$ is closed under self-composition, it is also $\infty$-conservative.

**Non-example: Cubic Polynomials.** Let $f(x, y) = x^2y$. By direct computation,

$$(\nabla f)^2(x, y) = \begin{pmatrix} 4x^3y \\ 4x^2y^2 \end{pmatrix} =: \begin{pmatrix} h_1(x, y) \\ h_2(x, y) \end{pmatrix}.$$

We then have $\frac{\partial}{\partial y}h_1(x, y) = 4x^3$, $\frac{\partial}{\partial x}h_2(x, y) = 8xy^2$. By Clairaut's theorem (see (Spivak, 2018, Chapter 4)), $(\nabla f)^2$ is not conservative. Thus, $\nabla\mathcal{P}_3(\mathbb{R}^2, \mathbb{R})$ is conservative but not 2-conservative.

### 3.1. Gradient Vector Fields of Generalized Linear Models

Let $\mathcal{G}(\mathbb{R}^n,\mathbb{R}) \subsetneq \mathcal{C}^1(\mathbb{R}^n,\mathbb{R})$ denote the class of functions $f : \mathbb{R}^n \to \mathbb{R}$ of the form

$$f(x) = \sum_{i=1}^{m} \sigma(\langle x, z_i\rangle) \tag{5}$$

where $m$ is a positive integer, $z_i \in \mathbb{R}^n$, and $\sigma \in \mathcal{C}^1(\mathbb{R})$. Such functions arise in statistics and optimization when learning generalized linear models. For example, when $\sigma(t) = \ln(1 + e^{-t})$, (5) is effectively the loss function used in logistic regression.

We further define $\mathcal{G}_\perp(\mathbb{R}^n,\mathbb{R}) \subsetneq \mathcal{G}(\mathbb{R}^n,\mathbb{R})$ to be the set of functions of the form (5) where $\{z_i\}_{i=1}^m$ are mutually orthogonal. We then have the following result.

**Theorem 3** *Let $f \in \mathcal{G}_\perp(\mathbb{R}^n,\mathbb{R})$ be as in (5). Let $\phi_i(t) = \|z_i\|^2\sigma'(t)$. For all $k \geq 2$,*

$$(\nabla f)^k(x) = \nabla \left( \sum_{i=1}^{m} \int_0^{\langle x, z_i\rangle} \sigma'(\phi_i^{k-1}(t))dt \right). \tag{6}$$

*Thus, $\nabla\mathcal{G}_\perp(\mathbb{R}^n,\mathbb{R})$ is $\infty$-conservative and closed under self-composition.*

**Proof** Let $V = \nabla f$. We claim that for all $k \geq 1$,

$$V^k(x) = \sum_{i=1}^{m} \sigma'(\phi_i^{k-1}(\langle x, z_i\rangle))z_i$$

where $\phi_i^0$ is the identity function. We will show this inductively. This clearly holds for $k = 1$. By the inductive hypothesis, we then have

$$\begin{aligned} V^{k+1}(x) &= \sum_{i=1}^{m} \sigma'(\langle V^k(x), z_i\rangle)z_i \\ &= \sum_{i=1}^{m} \sigma'\left( \left\langle \sum_{j=1}^{m} \sigma'(\phi_i^{k-1}(\langle x, z_j\rangle))z_j, z_i \right\rangle \right) z_i \\ &= \sum_{i=1}^{m} \sigma'\left( \|z_i\|^2\sigma'(\phi_i^{k-1}(\langle x, z_i\rangle)) \right) z_i \\ &= \sum_{i=1}^{m} \sigma'(\phi_i^{k}(\langle x, z_i\rangle))z_i. \end{aligned}$$

Here, the second equality follows from the inductive hypothesis, while the third equality follows from the orthogonality of the $z_i$. Therefore, if we define $h_k : \mathbb{R} \to \mathbb{R}$ via

$$h_k(x) = \sum_{i=1}^{m} \int_0^{\langle x, z_i\rangle} \sigma'(\phi_i^{k-1}(t))dt$$

then by the chain rule,

$$\nabla h_k(x) = \sum_{i=1}^{m} \sigma'(\phi_i^{k-1}(\langle x, z_i\rangle))z_i = V^k(x).$$

■

In order to understand the dynamics of gradient descent on generalized linear models, we now extend Theorem 3 to the function class $I - \gamma\nabla\mathcal{G}_\perp(\mathbb{R}^n, \mathbb{R})$.

**Theorem 4** *Let $f \in \mathcal{G}_\perp(\mathbb{R}^n, \mathbb{R})$ be as in (5). For fixed $\gamma \in \mathbb{R}$, let $\psi_i(t) = t - \gamma\|z_i\|^2\sigma'(t)$. For all $k \geq 2$,*

$$(I - \gamma\nabla f)^k(x) = x - \gamma\nabla\left(\sum_{i=1}^m \int_0^{\langle x, z_i\rangle} \sigma'(\psi_i^{k-1}(t))dt\right). \tag{7}$$

*Thus, $I - \gamma\nabla\mathcal{G}_\perp(\mathbb{R}^n, \mathbb{R})$ is $\infty$-conservative and closed under self-composition.*

**Proof** The proof is nearly identical to the proof of Theorem 3. Let $V(x) = x - \gamma\nabla f(x)$. A slight modification of the inductive argument in the proof of Theorem 3 implies that

$$V^k(x) = x - \gamma\sum_{i=1}^m \sigma'(\psi_i^{k-1}(\langle x, z_i\rangle))z_i.$$

By the chain rule, this implies that

$$V^k(x) = x - \gamma\nabla\left(\sum_{i=1}^m \int_0^{\langle x, z_i\rangle} \sigma'(\psi_i^{k-1}(t))dt\right).$$

■

On the other hand, $\nabla\mathcal{G}(\mathbb{R}^n, \mathbb{R})$ is not 2-conservative. Let $f_1(x, y) = e^x, f_2(x, y) = e^{x+y}$, $f_3 = f_1 + f_2$. Note that by Theorem 3, $\nabla f_1, \nabla f_2$ are both $\infty$-conservative. However, by direct computation

$$(\nabla f_3)^2(x, y) = \begin{pmatrix} \exp(e^x + e^{x+y}) + \exp(e^x + 2e^{x+y}) \\ \exp(e^x + 2e^{x+y}) \end{pmatrix} =: \begin{pmatrix} h_1(x, y) \\ h_2(x, y) \end{pmatrix}.$$

One can then verify that $\frac{\partial}{\partial y}h_1(x, y) \neq \frac{\partial}{\partial x}h_2(x, y)$, so by Clairaut's theorem, $\nabla f_3$ is not 2-conservative. Notably, $f_1, f_2$ and $f_3$ are all convex functions, demonstrating that whether $\nabla\mathcal{F}$ is $\infty$-conservative is not determined by whether the class $\mathcal{F}$ is convex.

While $f \in \mathcal{G}_\perp(\mathbb{R}^n, \mathbb{R})$ implies that $\nabla f$ is $\infty$-conservative, exactly characterizing the set of $\infty$-conservative vector fields in $\nabla\mathcal{G}(\mathbb{R}^n, \mathbb{R})$ remains open. Part of the difficulty in this problem comes from the fact that a function $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ can have multiple representations satisfying (5).

## 4. Smooth $k$-Conservative Vector Fields

We now explicitly construct the space of smooth, $k$-conservative vector fields. Given $V \in \mathcal{C}^\infty(\mathbb{R}^n)$, let $J(V) : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ denote its Jacobian, which we can view as an $n \times n$ matrix over $\mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$. If $V \in \mathcal{C}^\infty(\mathbb{R}^n)$, then by the Poincaré lemma (Warner, 1983, Section 4.18), $V$ is $k$-conservative if and only if $J(V^k)$ is symmetric. For $k \geq 1$, we then define $D_k : \mathcal{C}^\infty(\mathbb{R}^n) \to \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}^{n\times n})$ by

$$D_k(V) := J(V^k) - J(V^k)^\intercal. \tag{8}$$

Thus, $V \in \mathcal{C}^\infty(\mathbb{R}^n)$ is $k$-conservative if and only if $D_k(V) = 0$. We may now define the space of smooth, $k$-conservative vector fields by $\mathcal{W}^k(\mathbb{R}^n) := D_k^{-1}(\{0\})$ and $\mathcal{W}^\infty(\mathbb{R}^n) := \cap_{k=1}^\infty \mathcal{W}^k(\mathbb{R}^n)$. We note a few facts about $\mathcal{W}^\infty(\mathbb{R}^n)$:

1. $\mathcal{W}^k(\mathbb{R}^n)$ and $\mathcal{W}^\infty(\mathbb{R}^n)$ are closed in $\mathcal{C}^\infty(\mathbb{R}^n)$ under several natural topologies, like that of uniform convergence of all derivatives on compact sets. To see this, note that $D_k$ is a continuous function in this topology, so $D_k^{-1}(\{0\}) = \mathcal{W}^k(\mathbb{R}^n)$ is closed. Thus, $\mathcal{W}^\infty(\mathbb{R}^n)$ is an intersection of closed sets, and is closed itself.

2. $\mathcal{W}^\infty(\mathbb{R}^n)$ is closed under scalar multiplication. While it contains linear subspaces (such as the space of symmetric linear vector fields, see Section 3), it is not closed under addition. For a simple counter-example, see the end of Section 3.1.

3. While $\mathcal{W}^\infty(\mathbb{R}^n)$ is closed under self-composition, it is not closed under arbitrary composition. See Appendix A for an explicit counter-example.

Some basic open questions on the structure of $\mathcal{W}^\infty(\mathbb{R}^n)$:

1. How does $W^k(\mathbb{R}^n)$ relate to $W^j(\mathbb{R}^n)$ for $k \neq j$? As we show in Appendix A, $\mathcal{W}^k(\mathbb{R}^n) \not\subseteq \mathcal{W}^j(\mathbb{R}^n)$ for $j < k$. More generally, are there smooth vector fields that are $k$-conservative but not $j$-conservative for $j \neq k$?

2. If we restrict to $\mathcal{P}_d(\mathbb{R}^n)$, the zero locus of $D_k$ defines a projective variety over the coefficients of polynomials in $\mathcal{P}_d(\mathbb{R}^n)$. For example, applying (8) to $\mathcal{P}_d(\mathbb{R}^n)$, we find:

   - $\mathcal{W}^1(\mathbb{R}^n) \cap \mathcal{P}_1(\mathbb{R}^n)$ is a hyperplane.
   - $\mathcal{W}^2(\mathbb{R}^2) \cap \mathcal{P}_1(\mathbb{R}^2)$ is a union of two hyperplanes.
   - $\mathcal{W}^3(\mathbb{R}^2) \cap \mathcal{P}_1(\mathbb{R}^2)$ is a union of a hyperplane and a quadric surface.
   - $\mathcal{W}^1(\mathbb{R}^2) \cap \mathcal{W}^2(\mathbb{R}^2) \cap \mathcal{P}_2(\mathbb{R}^2)$ is a quadric surface.

   See Appendix A for the full details on these computations. Can we say anything more general? For example, what is the degree of $\mathcal{W}^k(\mathbb{R}^n) \cap \mathcal{P}_d(\mathbb{R}^n)$?

3. For all $k \geq 1$, define $\rho_k : \mathcal{W}^\infty(\mathbb{R}^n) \to \mathcal{W}^\infty(\mathbb{R}^n)$ via $V \mapsto V^k$. Many of the discussions above can be rephrased in terms of properties of this map. For example, Theorem 3 implies that $\rho_k$ is an endomorphism on $\nabla\mathcal{G}_\perp(\mathbb{R}^n, \mathbb{R})$. Are there other important function classes for which $\rho_k$ is an endomorphism? More broadly speaking, we may also wish to understand the image of $\rho_k$. Note that this is important for federated learning, as according to the discussion in Section 2, this will govern what kinds of dynamics of FEDAVG are possible in settings where clients have $\infty$-conservative loss functions.

## 5. Conservatism and Lifting

In this section, we show that if $V$ is $k$-conservative, then many properties of $V$ "lift" to the vector field $V^k$. This will further allow us to show that if $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ has a $k$-conservative gradient field and $\nabla h = (\nabla f)^k$, then many properties of $f$ will lift to $h$.

Due to their importance in optimization, we will focus on geometric notions related to convexity. Under smoothness, such notions can be rephrased in terms of eigenvalues of Jacobian matrices. Thus, we first prove a result concerning how eigenvalues of such matrices lift under self-composition of vector fields.

**Proposition 5** *Suppose $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ and $\nabla f$ is $j$-conservative for $1 \leq j \leq k$, with $(\nabla f)^j = \nabla g_j$. Then for all such $j$, the function $g_j$ is smooth and satisfies:*

1. *Suppose there are $\alpha, \beta \geq 0$ such that for all $x$, $\alpha I \preceq J(\nabla f)(x) \preceq \beta I$. Then for all $x$,*
$$\alpha^k I \preceq J(\nabla g_j)(x) \preceq \beta^k I.$$
2. *Suppose there is some $\lambda \geq 0$ such that for all $x$, $-\lambda I \preceq J(\nabla f)(x) \preceq \lambda I$. Then for all $x$,*
$$-\lambda^k I \preceq J(\nabla g_j)(x) \preceq \lambda^k I.$$

*Items 1 and 2 also hold if we change $\preceq$ to $\prec$ throughout.*

**Proof** Since $f$ is smooth, $(\nabla f)^j \in \mathcal{C}^\infty(\mathbb{R}^n)$. Since $(\nabla f)^j = \nabla g_j$ (and in particular, $g_j$ is differentiable), we must have $g_j \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$.

For Item 1, we proceed inductively. For $k = 1$, the result holds by assumption. For the inductive step, let $2 \leq k \leq K$, and assume the result holds for $k - 1$. Let $J_j(x)$ denote the Jacobian of $\nabla g_j$ at a point $x$. By the chain rule,

$$J_j(x) = J_1(\nabla g_{j-1}(x)) J_{j-1}(x). \tag{9}$$

By the inductive hypothesis, we have

$$\alpha^{j-1} I \preceq J_{j-1}(x) \preceq \beta^{j-1} I$$

and by our assumptions on $\nabla f$, we have

$$\alpha I \preceq J_1(\nabla g_{j-1})(x) \preceq \beta I.$$

Since $J_j(x)$ is symmetric (as it is the Jacobian of a gradient field), its eigenvalues are therefore products of eigenvalues of $J_{j-1}(x)$ and $J_1(\nabla g_{j-1})(x)$. Hence, its maximum eigenvalue is at most $\beta^j$, and its minimum eigenvalue is at most $\alpha^j$.

The proof of Item 2 follows in a similar way, noting that by the inductive hypothesis, the matrices on the right-hand side of (9) will have eigenvalues in the ranges of $[-\lambda, \lambda]$ and $[-\lambda^{j-1}, \lambda^{j-1}]$. Since $J_j(x)$ is symmetric, its eigenvalues are products of the eigenvalues of the matrices in the right-hand side of (9), and the result follows. ■

**Remark 6** *Note the critical role of symmetry in the argument above. In $\mathbb{R}^n$, $J(V^k)$ is symmetric if and only if $V$ is $k$-conservative. Thus, $k$-conservatism is exactly the condition required for us to reason about how the eigenvalues of $J(V^k)$ relate to that of $J(V)$.*

We will use Proposition 5 to show that iterating $\infty$-conservative vector fields preserves geometric properties, including Lipschitz continuity, as in the following definition.

**Definition 7** *A vector field $V \in \mathcal{C}^1(\mathbb{R}^n)$ is $\beta$-Lipschitz continuous if for all $x \in \mathbb{R}^n$, $\|J(V)(x)\| \leq \beta$. $V$ is Lipschitz continuous if there is some $\beta$ for which $V$ is $\beta$-Lipschitz continuous.*

In the definition above, $\|\cdot\|$ refers to the operator norm induced by the $\ell_2$ norm on $\mathbb{R}^n$, viewing $J(V)(x)$ as an $n \times n$ matrix over $\mathbb{R}$. In the following, we let $\mathcal{L}(\mathbb{R}^n) \subsetneq \mathcal{V}(\mathbb{R}^n)$ denote the set of Lipschitz continuous vector fields. Proposition 5 directly implies the following result.

**Corollary 8** *Let $\mathcal{F} \subsetneq \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ be the set of (a) smooth, strongly convex functions, (b) smooth, strictly convex functions, or (c) smooth, convex functions. Then $\nabla\mathcal{F} \cap \mathcal{W}^\infty(\mathbb{R}^n)$ and $\nabla\mathcal{F} \cap \mathcal{W}^\infty(\mathbb{R}^n) \cap \mathcal{L}(\mathbb{R}^n)$ are closed under self-composition.*

**Proof** This follows directly from Proposition 5. In particular, for (1), if $f$ is smooth and strongly convex, then there is some positive $\alpha$ such that $\alpha I \preceq J(\nabla f)(x)$ for all $x$. Since $\nabla f \in \mathcal{W}^\infty(\mathbb{R}^n)$, for all $k \geq 1$, there is some $g_k$ such that $\nabla g_k = (\nabla f)^k$. By Proposition 5, we have $\alpha^k I \preceq J(\nabla g_k)(x)$, so $g_k$ is smooth and strongly convex. If $\nabla f$ is also Lipschitz continuous, then there is some $\beta$ for which $J(\nabla f)(x) \preceq \beta I$ for all $x$, and a similar argument shows that $\alpha^k I \preceq J(\nabla g_k)(x) \preceq \beta^k I$.

The case of convex or strictly convex follows in an analogous manner, as they correspond (respectively) to the bounds $0 \preceq J(\nabla f)(x)$ and $0 \prec J(\nabla f)(x)$ (for all $x$), which is also preserved under $k$-fold composition by Proposition 5. ■

Thus, convexity "lifts" under self-composition of the associated gradient vector field: If $f$ is smooth, convex, and $\nabla f$ is $k$-conservative, then $(\nabla f)^k = \nabla g$ for some smooth, convex function $g$. Next, we consider vector fields $V = I - (I - \gamma\nabla f)^k$ where $\gamma > 0$ (induced by gradient descent). In the following lemma, we show that if $V$ is $\infty$-conservative and $V^k = \nabla h_k$, then $h_k$ inherits smoothness and critical points from $f$.

**Lemma 9** *Let $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ and $\gamma \in \mathbb{R}_{>0}$. Suppose that $(I - \gamma\nabla f)$ is $j$-conservative for $1 \leq j \leq k$. Then $V_k := I - (I - \gamma\nabla f)^k$ is conservative, and if $\nabla h_k = V_k$ then (1) $h_k$ is smooth, and (2) if $\nabla f(y) = 0$, then $\nabla h_k(y) = 0$.*

**Proof** For (1), $h_k$ is clearly differentiable. Moreover, $\nabla h_k = V_k \in \mathcal{C}^\infty(\mathbb{R}^n)$, as smoothness is preserved under addition and composition. Hence, $h_k \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$. For (2), note that $(I - \nabla f)(y) = y - \nabla f(y) = y$. Therefore, $\nabla h_k(y) = y - (I - \nabla f)^k(y) = y - y = 0$. ■

In fact, many geometric properties important to optimization (such as convexity) are also inherited by $h_k$, provided that $\gamma$ is not too large, as in the following.

**Lemma 10** *Suppose $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ and $\nabla f$ is $\beta$-Lipschitz continuous. Suppose that for some $\gamma \in \mathbb{R}_{>0}$, $(I - \gamma\nabla f)$ is $j$-conservative for $1 \leq j \leq k$, with $\nabla h_k = I - (I - \gamma\nabla f)^k$. Then:*

1. *If $f$ is $\alpha$-strongly convex and $\gamma \leq 2(\alpha + \beta)^{-1}$ then $h_k$ is $(1 - \lambda^k)$-strongly convex and $\nabla h_k$ is $(1 + \lambda^k)$-Lipschitz continuous for $\lambda = 1 - \gamma\alpha$.*

2. *If $f$ is convex and $\gamma \leq 2\beta^{-1}$ then $h_k$ is convex and $\nabla h_k$ is 2-Lipschitz continuous. If $\gamma \leq \beta^{-1}$, then $\nabla h_k$ is 1-Lipschitz continuous.*

3. *If $f$ is strictly convex and $\gamma < 2\beta^{-1}$ then $h_k$ is strictly convex.*

4. *If $f$ is $\delta$-weakly convex for $\delta \leq \beta$ and $\gamma \leq 2\beta^{-1}$, then $h_k$ is $(\lambda^k - 1)$-weakly convex and $\nabla h_k$ is $(1 + \lambda^k)$-Lipschitz continuous for $\lambda = 1 + \gamma\delta$.*

**Proof** This is a direct consequence of Proposition 5. For (1), by assumption we have $\alpha I \preceq J(\nabla f)(x) \preceq \beta I$ for all $x$, and therefore $-\lambda \preceq J(I - \gamma\nabla f)(x) \preceq \lambda I$ for all $x$ where $\lambda = 1 - \gamma\alpha$. By Proposition 5, we have that for all $x$

$$-\lambda^k I \preceq J((I - \gamma\nabla f)^k)(x) \preceq \lambda^k I$$

and so

$$0 \prec (1-\lambda^k)I \preceq J(\nabla h_k)(x) \preceq (1+\lambda^k)I.$$

The remaining parts of the lemma are proved in an analogous way using Proposition 5 and basic algebraic manipulations. ■

## 6. Convergence Rates in Federated Learning

We now use our machinery above to understand the convergence of FEDAVG in various settings. Recall that the server update at each round is given by $x_{t+1} = x_t - \eta V_s(x_t)$, where the "server vector field" $V_s$ is given by (2). Throughout, we assume that each client $c$ performs $k$ steps of gradient descent with learning rate $\gamma > 0$ on their loss function $f_c$. We make the following assumption.

**Assumption 1** *For all c, $f_c \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ and $I - \gamma\nabla f_c$ is j-conservative for $1 \leq j \leq k$.*

This leads to the following result on sufficient conditions for $V_s$ to be conservative.

**Theorem 11** *Under Assumption 1, $V_s$ is a conservative vector field. Moreover, if $V_s = \nabla f_s$, then $f_s$ is smooth and the* FEDAVG *server update in (1) is equivalent to the following gradient descent step:*

$$x_{t+1} = x_t - \eta\nabla f_s(x_t). \tag{10}$$

**Proof** By Assumption 1, for $c = 1, \ldots, C$, there is some function $h_c$ such that $\nabla h_c = (I - \gamma\nabla f_c)^k$. We can then define $q_c : \mathbb{R}^n \to \mathbb{R}$ by $q_c(x) := \frac{1}{2}\|x\|^2 - h_c(x)$. By construction, $\nabla q_c = I - (I - \gamma\nabla f)^k$, so that $V_s = C^{-1}\sum_{c=1}^C \nabla q_c$. Therefore, $V_s = \nabla f_s$ where $f_s = C^{-1}\sum_{c=1}^C q_c$. The smoothness of any $f_s$ such that $V_s = \nabla f_s$ follows directly from Lemma 9. ■

In this setting, if we have some understanding of $f_s$ (for example, whether $f_s$ is convex), we can immediately apply centralized optimization results to derive convergence results for FEDAVG. To better understand the structure of $f_s$, we will use Lemma 10. Since this requires Lipschitz continuity, we make the following assumption.

**Assumption 2** *For all c, $\nabla f_c$ is $\beta$-Lipschitz continuous.*

Under Assumptions 1 and 2, Lemma 10 lifts geometric properties of the client loss functions $f_c$ to the function $f_s$. Combining this with Theorem 11, we can translate convergence rates for gradient descent to convergence rates for FEDAVG in strongly convex and convex settings. We make no direct assumptions on client heterogeneity. Throughout, we let $f_s$ be a function such that $V_s = \nabla f_s$, as guaranteed by Theorem 11.

**Theorem 12** *Suppose Assumptions 1 and 2 hold, and that for all c, $f_c$ is $\alpha$-strongly convex. Then $f_s$ has a unique minimizer $x_s^*$, and if $\gamma = 2(\alpha+\beta)^{-1}$, $\eta = 1$, the iterates $\{x_t\}_{t=0}^\infty$ of* FEDAVG *satisfy*

$$\|x_t - x_s^*\| \leq \left(\frac{\beta-\alpha}{\beta+\alpha}\right)^{kt} \|x_0 - x_s^*\|. \tag{11}$$

**Proof** This follows directly by combining Theorem 11 and Lemma 10 with well-known convergence rates for smooth, strongly convex functions (for example, see (Nesterov, 2003, Theorem 2.1.15)). See Appendix B.1 for more details. ■

The convergence rate in (11) was shown first by Malinovskiy et al. (2020, Theorem 2.11), which extended to non-conservative gradient vector fields. The salient difference is that under under our assumptions, the limit point $x_s^*$ is actually the global minimizer of some strongly convex function. As we discuss below, this allows us to immediately derive analogous results for variants of FEDAVG that apply other server optimizers.

When $k = 1$, this recovers the convergence of gradient descent on $f_{avg} = C^{-1}\sum_{c=1}^{C} f_c$. Hence, FEDAVG with $k > 1$ yields an exponential improvement in convergence (with respect to $k$), but may not converge to the minimizer $x^*$ of $f_{avg}$. To understand this discrepancy, one could analyze $\|x_s^* - x^*\|$. A tight upper bound was given for strongly convex quadratic functions by Charles and Konečný (2021, Lemma 5). A bound in the general strongly convex setting (not assuming $k$-conservatism) was given by Malinovskiy et al. (2020, Theorem 2.14), though whether this bound can be improved under Assumption 1 is an open question.

We now give a convergence rate for FEDAVG in the convex setting.

**Theorem 13** *Suppose Assumptions 1 and 2 hold, and that for all c, $f_c$ is convex with finite minimizer. Then $f_s$ has a finite minimizer $x_s^*$, and if $\gamma = \beta^{-1}$, $\eta = 1$, the iterates $\{x_t\}_{t=0}^{\infty}$ of* FEDAVG *satisfy*

$$f_s(x_t) - f_s(x_s^*) \leq \frac{1}{2t}\|x_0 - x_s^*\|^2. \tag{12}$$

**Proof** This follows by combining Theorem 11 and Lemma 10 with well-known convergence rates for smooth, convex functions (for example, see (Bubeck, 2015, Theorem 3.3)). See Appendix B.2 for more details. ■

To the best of our knowledge, Theorem 13 is the first result showing that FEDAVG exhibits convergent behavior on a class of functions, even with fixed learning rates and $k > 1$. Unlike Theorem 12, it is not clear that the convergence in (12) is "faster" (in some sense) than the convergence of gradient descent on $f_{avg}$. Such analysis is an open and important problem.

### 6.1. Extensions to Other Methods

These techniques allow us to transfer convergence rates for many optimization algorithms to federated learning under the same assumptions as Theorems 12 and 13; We can directly analyze any federated learning algorithm where the server update in (3) is replaced with another optimizer (as proposed by Reddi et al. (2021)). If we treat $V_s(x_t)$ as an estimate of the gradient of the loss function, applying gradient descent leads us to the update step in (3). Given any first-order "server optimization" method SERVEROPT, we can generalize (3) via

$$x_{t+1} = \text{SERVEROPT}(V_s(x_t)). \tag{13}$$

For example, SERVEROPT could be gradient descent with momentum or an adaptive method such as Adagrad (Duchi et al., 2011; McMahan and Streeter, 2010). These two choices of SERVEROPT lead to FEDAVGM (Hsu et al., 2019) and FEDADAGRAD (Reddi et al., 2021) respectively, and can lead to improved empirical convergence. Note that under Assumption 1, (13) becomes $x_{t+1} =$

SERVEROPT($\nabla f_s(x_t)$), which is simply first-order optimization. Thus, convergence rates for SERVEROPT can be translated to a converge rates for the federated optimization update in (13). Notably, this directly implies that in certain settings, there are algorithms which converge faster than FEDAVG to the same point.

For example, in the same settings as Theorem 12, we can improve convergence by using gradient descent with heavy-ball momentum instead of just gradient descent. By an almost identical proof to Theorem 12, we have the following result.

**Theorem 14** *Let $\{x_t\}_{t=0}^{\infty}$ be the iterates of (13) where* SERVEROPT *is gradient descent with heavy-ball momentum. Under the same setting as Theorem 12, for some choice of parameters of* SERVEROPT*, we have*

$$\|x_t - x_s^*\| \leq \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^t \|x_0 - x_s^*\| \tag{14}$$

*where $\kappa = (1+\lambda^k)/(1-\lambda^k)$ and $\lambda = (\beta-\alpha)/(\beta+\alpha)$.*

**Proof** The proof is the same as for Theorem 12, but we apply convergence rates for gradient descent with heavy-ball momentum instead (see (Polyak, 1964)). See Appendix B.1 for more details. ■

One can verify that the convergence rate in (14) is faster than (11). We stress that while the same kind of result can be derived for any number of centralized optimization algorithms, the key point is that our observations allows us to leverage existing knowledge of centralized optimization methods in the context of federated learning. In particular, this can enable more informed, theoretically grounded decisions about which choice of optimizer and hyperparameters to use in (13).

## 7. Open Problems

As we have shown, FEDAVG is well-behaved when certain vector fields are $k$-conservative (see Theorems 12 and 13) and can exhibit non-convergent, circular behavior when they are not (Section 2.1). Better characterizations of when FEDAVG exhibits convergent behavior (or fails to do so) is an important open problem. Similarly, we have only scratched the surface on how the dynamics of the client loss functions lift to the server dynamics. While many convexity-adjacent properties lift (Lemma 10), one can show that many natural properties (including being bounded below) do not lift. What about properties such as the Polyak-Łojasiewicz condition (Karimi et al., 2016)? More generally, we would like to characterize which properties lift and use this to better understand the behavior of FEDAVG on a larger class of functions.

Another important area is understanding the empirical effectiveness of methods such as FEDAVG. As discussed by Wang et al. (2021), theoretical convergence rates of federated learning methods often do not improve upon centralized rates for algorithms such as gradient descent. While Theorem 12 shows that FEDAVG accelerates convergence to a non-optimal point, it is unclear whether Theorem 13 implies a similar acceleration. More generally, is there some sense in which the limit point $x_s^*$ is a useful point of convergence, either for learning a global model, or as a starting point for personalization?

Finally, the dynamics presented in Section 2.1 point to a fundamental failure of methods such as FEDAVG, to our knowledge the first observation of cyclic dynamics in the literature on FEDAVG. This mirrors non-conservative dynamics arising from many GAN training methods (Mescheder et al., 2018). Can we use insights from training multi-agent systems (such as GANs) to create better federated learning methods, or even to simply design better client loss functions?

## References

K. A. Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloé M Kiddon, Jakub Konečný, Stefano Mazzocchi, Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards federated learning at scale: System design. In *SysML 2019*, 2019. URL https://arxiv.org/abs/1902.01046.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Zachary Charles and Jakub Konečný. Convergence and accuracy trade-offs in federated learning and meta-learning. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, 2021.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In Paolo Frasconi, Niels Landwehr, Giuseppe Manco, and Jilles Vreeken, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 795–811, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46128-1.

Saber Malekmohammadi, Kiarash Shaloudegi, Zeou Hu, and Yaoliang Yu. An operator splitting view of federated learning. *arXiv preprint arXiv:2108.05974*, 2021.

Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtarik. From local SGD to local fixed-point methods for federated learning. In *International Conference on Machine Learning*, pages 6692–6701. PMLR, 2020.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

Reese Pathak and Martin J Wainwright. Fedsplit: an algorithmic framework for fast federated optimization. *Advances in Neural Information Processing Systems*, 33:7057–7066, 2020.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=LkFG3lB13U5.

Michael Spivak. *Calculus on manifolds: a modern approach to classical theorems of advanced calculus*. CRC press, 2018.

Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

Frank W. (Frank Wilson) Warner. Foundations of differentiable manifolds and Lie groups, 1983.

## Appendix A. In-Depth Examples

In this section, we give some in-depth examples regarding the $k$-conservatism of vector fields in $\mathcal{C}^\infty(\mathbb{R}^2)$. Note that for $V \in \mathcal{C}^\infty(\mathbb{R}^2)$, $D_k(V)$, as defined in (8), is a $2 \times 2$ anti-symmetric matrix over $\mathcal{C}^\infty(\mathbb{R}^2, \mathbb{R})$. Thus, when setting $D_k(V) = 0$, it suffices to consider a single off-diagonal entry. In a slight abuse of notation, in this section we will identify $D_k(V)$ with either off-diagonal entry of $D_k(V)$. Note that this is well-defined up to a factor of $-1$.

### A.1. Linear Vector Fields

Recall that $\mathcal{P}_1(\mathbb{R}^n)$ denotes the set of linear vector fields. Let $V \in \mathcal{P}_1(\mathbb{R}^n)$ be of the form $V(x, y) = (ax + by, cx + dy)$. Then we have the following equations (where we consider only the non-zero off-diagonal entries of $D_k$):

$$\begin{aligned}
D_1(V) &= b - c \\
D_2(V) &= (b - c)(a + d) \\
D_3(V) &= (b - c)(a^2 + ad + bc + d^2) \\
D_4(V) &= (b - c)(a + d)(a^2 + 2bc + d^2).
\end{aligned}$$

If $V$ is conservative, then $b = c$ and these equations all vanish. Comparing $D_1$, $D_2$, and $D_3$, we see that 2-conservative vector fields need not be conservative nor 3-conservative. For example, if we take

$$V(x) = \begin{pmatrix} 1 & 2 \\ 1 & -1 \end{pmatrix} x$$

then $V$ is 2-conservative and 4-conservative, but not conservative or 3-conservative.

Recall that in Section 3, we defined $\mathcal{S}_1(\mathbb{R}^n)$ to be the set of symmetric linear vector fields. While $\mathcal{S}_1(\mathbb{R}^n)$ is closed under self-composition, it is not closed under arbitrary composition. is To see this, consider the symmetric linear vector fields

$$V_1(x) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x, \quad V_2(x) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} x.$$

Then $V_1, V_2 \in \mathcal{W}^\infty(\mathbb{R}^n)$. However, $V_1 \circ V_2 \notin \mathcal{W}^\infty(\mathbb{R}^n)$ since

$$V_1(V_2(x)) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} x$$

which is a non-symmetric linear map.

Notably, $\mathcal{P}_1(\mathbb{R}^n)$ contains vector fields that are $j$-conservative but not $k$-conservative for $k < j$. For $j \geq 2$, consider the vector field given by $V_j(x) = A_j x$ where

$$A_j(x) = \begin{pmatrix} \cos(\theta_j) & \sin(\theta_j) \\ -\sin(\theta_j) & \cos(\theta_j) \end{pmatrix}, \quad \theta_j = \frac{\pi}{j}.$$

This is the vector field that rotates vectors by an angle of $\pi/j$. Since $V_j^k$ is conservative precisely when $V_j^k$ is symmetric, $V_j^k$ is conservative if and only if $\sin(k\theta_j) = 0$. Thus, $V_j$ is $k$-conservative if and only if $j$ divides $k$.

### A.2. Gradient Vector Fields of Cubic Polynomials

Consider the vector space $\mathcal{P}_3(\mathbb{R}^2, \mathbb{R})$ containing polynomials of the form

$$f(x, y) = ax^3 + bx^2y + cxy^2 + dy^3$$

for $a, b, c, d \in \mathbb{R}$. All such $f$ satisfy $D_1(\nabla f) = 0$ (as $\nabla f$ is conservative). By direct computation, taking only the off-diagonal entries of $D_k$, we get

$$D_2(\nabla f) = g_1x^3 + g_2x^2y + g_3xy^2 + g_4y^3.$$

for $g_1, g_2, g_3, g_4 \in \mathbb{R}[a, b, c, d]$ defined by

$$\begin{aligned} g_1 &= -4b(3ac - b^2 + 3bd - c^2) \\ g_2 &= 4(3a - 2c)(3ac - b^2 + 3bd - c^2) \\ g_3 &= 4(2b - 3d)(3ac - b^2 + 3bd - c^2) \\ g_4 &= 4c(3ac - b^2 + 3bd - c^2). \end{aligned}$$

One can then verify that these equations vanish simultaneously if and only if

$$g(a, b, c, d) = 3ac - b^2 + 3bd - c^2 = 0.$$

Thus, the set of 2-conservative functions in $\nabla\mathcal{P}_3(\mathbb{R}^2, \mathbb{R})$ is the hypersurface given by the zero locus of $g$. Since this zero locus is not closed under addition, the set of 2-conservative vector fields in $\nabla\mathcal{P}_3(\mathbb{R}^2, \mathbb{R})$ is not closed under addition either.

An analogous computation shows that the set of 3-conservative function is given by the zero locus of 8 homogeneous polynomials of degree 7, each of which is divisible by $g$. Therefore, all 2-conservative vector fields in $\nabla\mathcal{P}_3(\mathbb{R}^2, \mathbb{R})$ are also 3-conservative.

## Appendix B. Detailed Proofs

### B.1. Proof of Theorems 12 and 14

**Proof** Fix $c$ such that $1 \leq c \leq C$. By Lemma 9, there is some function $h_c \in$ such that $\nabla h_c = I - (I - \gamma \nabla f_c)^k$. Let $\lambda = (\beta - \alpha)/(\beta + \alpha)$. By Assumption 2, Lemma 10, and our assumption on $\gamma$, we find that $h_c$ is $(1 - \lambda^k)$-strongly convex and $\nabla h_c$ is $(1 + \lambda^k)$-Lipschitz continuous.

Note that the server vector field $V_s$ in (2) is therefore given by $V_s = \nabla f_s$ where

$$f_s(x) = \frac{1}{C} \sum_{c=1}^{C} h_c(x).$$

By basic properties of strong convexity and Lipschitz-continuity, we find that $f_s$ is $(1 - \lambda^k)$-strongly convex and $\nabla f_s$ is $(1 + \lambda^k)$-Lipschitz continuous. In particular, it has a unique minimizer $x_s^*$.

For Theorem 12, applying standard results on the convergence of gradient descent on smooth strongly convex functions (in particular, see (Nesterov, 2003, Theorem 2.1.15)), we find that the iterates of gradient descent with learning rate of $\eta = 1$ on $f_s$ produces iterates $\{x_t\}_{t=0}^{\infty}$ such that

$$\|x_{t+1} - x_s^*\| \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^t \|x_o - x_s^*\|$$

where $\kappa = (1 + \lambda^k)/(1 - \lambda^k)$. Some simple algebraic manipulation implies

$$\frac{\kappa - 1}{\kappa + 1} = \left(\frac{\beta - \alpha}{\beta + \alpha}\right)^k$$

proving the result.

For Theorem 14, we apply standard results on the convergence of gradient descent with heavy-ball momentum (see Polyak (1964)). In particular, by setting the learning rate $\eta$ by

$$\eta = \frac{4}{\left(\sqrt{1 + \lambda^k} + \sqrt{1 - \lambda^k}\right)^2}$$

and the momentum parameter $m$ by

$$m = \max\left\{\left|1 - \sqrt{\eta(1 - \lambda^k)}\right|, \left|1 - \sqrt{\eta(1 + \lambda^k)}\right|\right\}^2$$

we obtain the desired convergence rate. ■

### B.2. Proof of Theorem 13

**Proof** Fix $c$ such that $1 \leq c \leq C$. By Lemma 9, there is some function $h_c \in$ such that $\nabla h_c = I - (I - \gamma \nabla f_c)^k$. By Assumption 2, Lemma 10, and our assumption on $\gamma$, we find that $h_c$ is convex and 1-Lipschitz continuous. By Theorem 9 and our assumption that $f_c$ has a finite minimizer, $h_c$ has a finite minimizer as well.

Note that the server vector field $V_s$ in (2) is therefore given by $V_s = \nabla f_s$ where

$$f_s(x) = \frac{1}{C}\sum_{c=1}^{C} h_c(x).$$

By basic properties of convexity and Lipschitz-continuity, we find that $f_s$ is convex and 1-Lipschitz continuous. Moreover, the average of convex functions with finite minimizers must have a finite minimizer, so $f_s$ also has some finite minimizer $x_s^*$. By applying standard results on the convergence of gradient descent on smooth convex functions (in particular, see (Bubeck, 2015, Theorem 3.3)), we find that the iterates of gradient descent with learning rate of $\eta = 1$ on $f_s$ produces iterates $\{x_t\}_{t=0}^{\infty}$ such that

$$f_s(x_t) - f_s(x_s^*) \leq \frac{1}{2t}\|x_0 - x_s^*\|.$$

■

## Appendix C. Closed Integral Curves in Federated Learning

In this appendix we present calculations that demonstrate the possibility of closed integral curves in federated learning with nonconvex client losses. The existence of losses of higher regularity than those presented here (e.g. convex or satisfying the PL condition) whose server dynamics admit closed integral curve solutions is an interesting open question. We suspect that examples like this can be transferred to some higher regularity classes, but clearly not all. For example, Charles and Konečný (2021) demonstrate that such integral curves are impossible for quadratic functions (under minor assumptions on learning rates).

Our example dynamics take place in $\mathbb{R}^2$, and we focus on the case of $C = 2$ clients. For $c = 1, 2$ we define a family of functions by

$$f_c(x,y) := f_c^{(1)}(x,y) + f_c^{(2)}(x,y), \tag{15}$$

where

$$f_c^{(1)}(x,y) := \min\left(\frac{\alpha_c}{2}\left(y - y_c\right)^2 + \frac{\beta_c}{2}\left(x - x_c\right)^2, 1\right),$$

$$f_c^{(2)}(x,y) := \min\left(\frac{\alpha_c}{2}\left(y + y_c\right)^2 + \frac{\beta_c}{2}\left(x + x_c\right)^2, 1\right).$$

We will see that carefully selecting two functions from this family and performing full-gradient FEDAVG on these clients will yield server dynamics with closed integral curves. First, note that for any $x_c$ and $y_c$, $\alpha_c$ and $\beta_c$ can be chosen such that the domains of attraction of the terms $f_c^{(1)}$ and $f_c^{(2)}$ are non-overlapping. One can verify that setting $\gamma = 5, \delta = 0.05$, and letting $\alpha_1 = \delta, \beta_1 = \gamma, x_1 = y_1 = 1$, or $\alpha_2 = \gamma, \beta_2 = \delta, x_2 = -1, y_2 = 1$ satisfies this requirement. Let these choices define the functions $f_1$ and $f_2$.

Now, assume we perform FEDAVG with fixed learning rate $\eta > 0$ for some sufficiently large number of local steps $k$. We assume these clients follow full gradient descent, and we choose $k$ large enough so that the clients following full-gradient descent on the losses $f_1$ and $f_2$ converge to a stationary point, independent of starting point. This can be guaranteed in our setting by setting $k = O\left(\eta^{-1}\right)$, with (easily computable) constant depending on $\gamma$ and $\delta$.

Notice that by assuming clients "run until convergence", the form of the server vector field $V_s$ (defined in (2)) becomes quite simple. We define the following domains in the $xy$-plane:

$$\begin{aligned}\mathbf{I} &= \{(x,y) : f_1^{(1)}(x,y) < 1\},\\ \mathbf{II} &= \{(x,y) : f_1^{(2)}(x,y) < 1\},\\ \mathbf{III} &= \{(x,y) : f_2^{(1)}(x,y) < 1\},\\ \mathbf{IV} &= \{(x,y) : f_2^{(2)}(x,y) < 1\}.\end{aligned} \tag{16}$$

It is straightforward (though tedious) to verify that our choices of $\gamma, \delta$ above ensure $\mathbf{I} \cap \mathbf{IV}, \mathbf{I} \cap \mathbf{III}, \mathbf{II} \cap \mathbf{III},$ and $\mathbf{II} \cap \mathbf{IV}$ are all nonempty. With these regions defined, a straightforward computation shows that the server vector field $V_s$ is given by:

$$V_s(x,y) = \begin{cases} (1-x,-y) & (x,y) \in \mathbf{I} \cap \mathbf{IV} \\ (-x,1-y) & (x,y) \in \mathbf{I} \cap \mathbf{III} \\ (-1-x,-y) & (x,y) \in \mathbf{II} \cap \mathbf{III} \\ (-x,-1-y) & (x,y) \in \mathbf{II} \cap \mathbf{IV} \\ (1-x,1-y) & (x,y) \in \mathbf{I} \cap (\mathbf{III}^c \cup \mathbf{IV}^c) \\ (-1-x,1-y) & (x,y) \in \mathbf{III} \cap (\mathbf{I}^c \cup \mathbf{II}^c) \\ (-1-x,-1-y) & (x,y) \in \mathbf{II} \cap (\mathbf{III}^c \cup \mathbf{IV}^c) \\ (1-x,-1-y) & (x,y) \in \mathbf{IV} \cap (\mathbf{I}^c \cup \mathbf{II}^c) \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

We define a flow along this vector field in the usual manner, by the ODE

$$\frac{d}{dt}\,(x(t),y(t)) = V_s(x,y).$$

That the dynamics of FEDAVG will admit closed integral curves in this setting can now be readily seen, either by inspecting Fig. 1 or explicitly following a closed trajectory. The dynamics of FEDAVG (as in (1)) correspond to discretizing the ODE above with some step-size $\eta$. That is, FEDAVG maps a point $(x_t, y_t)$ to $(x_{t+1}, y_{t+1}) := (x_t, y_t) + \eta V_s(x_t, y_t)$. Under this discretization, letting $(x_0, y_0) = (0, 1)$ and choosing $\eta = 1$ yields a closed trajectory of period 8. Further, the choice of discretization does not affect the nature of the closed curve, only its period, as is clear from Fig. 1.