Aggregation on Learnable Manifolds for Asynchronous Federated Optimisation

Archie Licudi[2,1] **Anshul Thakur**[1] **Soheila Molaei**[1] **Danielle Belgrave**[3] **David A. Clifton**[1,4]
[1]Institute of Biomedical Engineering, University of Oxford, UK
[2] Imperial College London, UK
[3]GlaxoSmithKline, London, UK
[4]Oxford-Suzhou Institute of Advanced Research (OSCAR), Suzhou, China

Abstract

Asynchronous federated learning (FL) with heterogeneous clients faces two key issues: curvature-induced loss barriers encountered by standard linear parameter interpolation techniques (e.g. FedAvg) and interference from stale updates misaligned with the server's current optimisation state. To alleviate these issues, we introduce a geometric framework that casts aggregation as curve learning in a Riemannian model space and decouples trajectory selection from update conflict resolution. Within this, we propose AsyncBezier, which replaces linear aggregation with low-degree polynomial (Bézier) trajectories to bypass loss barriers, and OrthoDC, which projects delayed updates via inner-product–based orthogonality to reduce interference. We establish framework-level convergence guarantees covering each variant given simple assumptions on their components. On three datasets spanning general-purpose and healthcare domains, including LEAF Shakespeare and FEMNIST, our approach consistently improves accuracy and client fairness over strong asynchronous baselines; finally, we show that these gains are preserved even when other methods are allocated a higher local compute budget.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. Copyright 2026 by the author(s).

1 INTRODUCTION

In recent years, Federated Learning (FL) has seen a wave of research interest (Zhang et al., 2021; Xu et al., 2023) for its ability to keep data in private silos and achieve collaborative model training without necessitating the disclosure of sensitive information. This has been particularly notable in the healthcare sector (Rieke et al., 2020; Soltan et al., 2023; Molaei et al., 2024), where practitioners must balance evolving data-privacy legislation against the need for high-performance models in high-stakes settings. In particular, FL studies optimisation problems of the form:

$$\min_{\Theta \in \mathcal{M}^{\Theta}} \mathcal{L}(\Theta) := \frac{1}{M} \sum_{i=1}^{M} w_i \mathbb{E}_{(X,y)\sim p_i}[\ell(y; X, \Theta)] \quad (1)$$

for some vector of client weights $\mathbf{w} \in \mathbb{R}^M$ and some set of client risk functions $\mathcal{L}_i$, corresponding to the expected value of loss ℓ over the client data distribution p_i. Each client has access only to $\mathcal{L}_i$ and must collaboratively find a minimum Θ. In the original FedAvg algorithm, this is accomplished by taking a simple arithmetic mean of client models trained by SGD (McMahan et al., 2017).

Where clients have differing dataset sizes or computational resources, some participants will consistently compute training steps faster than others (Pfeiffer et al., 2023), leading to long idle times in the synchronous FedAvg paradigm. This motivates consideration of *asynchronous updates* (Xie et al., 2020), where clients are able to submit their results and receive an updated global model to continue training immediately. In this setting, distributional heterogeneity between client datasets poses a more severe challenge, since conflicting updates can no longer be reconciled by waiting for all clients. Despite this, most FL systems in use today rest on the assumption that the linear interpolation of client models produces a strong multi-task model. In

the irregular and non-convex loss landscapes of neural networks (Li et al., 2018), this assumption can fail as "barriers" of higher loss are encountered when averaging along straight lines.

Related Work There have been many proposals since to mitigate the effects of client heterogeneity and asynchronous update staleness. Li et al. (2020) is a notable example, which adds a *proximal* L^2 regularisation term to the client losses; this principle is used in the asynchronous setting by Xie et al. (2020). Nguyen et al. (2022) takes the simple step of buffering updates to increase training stability, where Wang et al. (2022) aims to homogenise clients by scaling the number of local epochs each client performs according to the delay with which its updates are received, as well as down-weighting the contribution of updates according to this metric-based "staleness" value. Unlike the methods above, Zheng et al. (2017) directly modifies the update rule, approximating the gradient at the current point via a first-order Taylor expansion around the stale gradient. A number of literature proposals are based on adaptive optimisation at the server-side (Wang et al., 2024; Reddi et al., 2021) and seek to delay-correct these momentum terms (Shi et al., 2024; Wang et al., 2024), but they maintain the same linear connectivity assumption as the aforementioned.

Where methods do explicitly consider mode connection geometry, it is usually indirect, via flatness-aware minimisation (Sun et al., 2023) or whole manifold learning (Grinwald et al., 2024), and neither approach tackles loss barriers explicitly. A final approach which seeks to improve the linear connection quality is Wang et al. (2020), performing neuron alignment (Tatro et al., 2020) before aggregation to factor out permutation equivariance in layers; we find, however, that the number of epochs for which each client trains in the standard federated setting almost never leads to misaligned models, suggesting that this is only appropriate for the direct model fusion problem (Li et al., 2023).

Our Contributions We present a novel family of algorithms in full Riemannian generality (Nickel and Kiela, 2018; Bonnabel, 2013; Li and Ma, 2022) that relaxes this linear assumption to the existence of an arbitrary low-loss geodesic. Marking a departure from prior art, we dynamically learn these "aggregation manifolds" with a modified two-step training process, for which we provide a framework-level convergence result. From these foundations, we propose the AsyncBezier algorithm for asynchronous optimisation as a simple implementation where polynomial mode connections are directly learned as low-loss 1-manifolds and the novel OrthoDC staleness correction rule is deployed to factor out update directions which conflict with the global optimisation trajectory. Finally, we implement a comprehensive empirical testing suite using an asynchronous fork of the Flower FL library (Beutel et al., 2022), demonstrating that our proposal is able to consistently outperform existing literature baselines on the canonical benchmark datasets FEMNIST, LEAF Shakespeare, and CXR8.

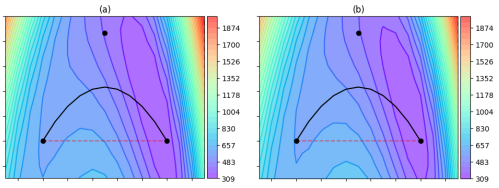

Figure 1: Quadratic Bézier mode connections learned during the federated training of LeNet-5, projected onto a 2-d loss landscape. Plot (a) shows cross-entropy loss w.r.t. a local training set and (b) w.r.t. the global test set, exhibiting the same geometric features in both.

2 BACKGROUND

2.1 Mode Connectivity

Different local minima (*modes*) in parameter space are often connected by simple polynomial curves of low average loss, revealing a large, highly-connected subspace of good solutions (Garipov et al., 2018; Lubana et al., 2023). These polynomial mode connections often exist between heterogenous multi-task models even where the linear connection fails, and are consistently able to find paths of lower average loss than the straight line, suggesting natural curvature to this solution subspace (Zhou et al., 2023) which an aggregation mechanism should respect.

Figure 1 shows the advantage of taking loss landscape curvature into account when merging local and global parameters. At an aggregation step, we learn a low-loss quadratic Bézier path between the models. Although any order of curve could be chosen, Zhou et al. (2023) demonstrates that single-bend paths are usually expressive enough to avoid non-convexities. Accordingly, we choose quadratics as the simplest family that are smooth everywhere, keeping curve learning steps to the same complexity as a normal parameter update step. To produce the plot, the local and global parameters are projected onto two vertices of the standard 2-simplex, with the learned control point projected onto the third. The learned curvature-aware connection and the naive linear connection are plotted in solid black and dashed red respectively, with contours showing the loss function in the plane of this simplex.

In both cases, we see a configuration reminiscent of

figures in Garipov et al. (2018), although the effect is less pronounced given the comparatively few epochs in a single federated learning step: the piecewise linear optimisation trajectory of local training has navigated around a "barrier" in parameter space of higher loss that the naive linear aggregation function is not expressive enough to capture, but the quadratic curve is. Work such as Izmailov et al. (2018) and Guo et al. (2023) has examined the positive relationship between choosing models from the midpoint of mode connections and the flatness of minima, conjectured to be correlated with a model's generalisation ability (Haddouche et al., 2025; Caldarola et al., 2022).

2.2 Riemannian Optimisation Preliminaries

We begin by briefly recalling the key mathematical components of Riemannian Gradient Descent (Bonnabel, 2013):

Definition 1 (Riemannian Gradient)**.** *Let $f : \mathcal{M} \to \mathbb{R}$ be a real-valued C^∞ function w.r.t. a **Riemannian manifold (R-Manifold)** $\mathcal{M}$. Then we write $\operatorname{grad} f(x) \in T_x\mathcal{M}$ to denote the unique tangent such that, for all $v \in T_x\mathcal{M}$*

$$Df_x(v) = \langle \operatorname{grad} f(x), v \rangle \tag{2}$$

Definition 2 (Exponential Map)**.** *Letting γ_v denote the unique geodesic from x with initial tangent vector v, we define the Riemannian exponential map:*

$$\exp_x(v) := \gamma_v(1) \tag{3}$$

This generalises the idea in Euclidean space of stepping along a straight line towards a point to the setting of R-manifolds. Since geodesics are constant-speed, we have the desirable quality that $d(x, \exp_x(v)) \equiv \|v\|$ where d denotes the induced Riemannian metric on $\mathcal{M}$.

Definition 3 (Metric-Preserving Transport)**.** *Letting $x, y \in \mathcal{M}$ we write $P_{x\to y} : T_x\mathcal{M} \to T_y\mathcal{M}$ to denote the **parallel transport** map with respect to the Levi-Civita connection. This map has the **(Riemannian) metric-preserving** property:*

$$\forall v, w \in T_x\mathcal{M}, \quad \langle P_{x\to y}[v], P_{x\to y}[w] \rangle_x = \langle v, w \rangle_y \tag{4}$$

The technical definition of parallel transport in general terms is beyond the scope of this paper, as this property is the only one we actively use (along with the guaranteed existence of such a map for any $x, y \in \mathcal{M}$). $P_{x\to y}$ is not always the only function with this property, but it is the unique one which also introduces no **torsion** to the underlying manifold (Lee, 1997).

Riemannian GD then proceeds with a simple generalisation of the Euclidean GD update rule:

$$\theta^{t+1} \leftarrow \exp_{\theta^t}(\eta \operatorname{grad}(\theta^t)) \tag{5}$$

for some learning rate $\eta \in (0, \infty)$; it is clear how this can be used to generalise Euclidean FedAvg to the Riemannian context, and we can similarly lift the two main paradigms of handling asynchronicity to manifolds. More precisely, the issue of grad being computed against θ^τ for $\tau < t$ can be solved by trusting the learned *position* or *tangent*, exemplified by FedAsync (Xie et al., 2020) and ASGD (Dean et al., 2012) respectively. We can express these in general Riemannian terms, letting g^τ denote the learned stochastic pseudo-gradient and $\hat{\theta}^\tau := \exp_{\theta^\tau}(g^\tau)$ the learned model:

$$\theta^{t+1} \longleftarrow \exp_{\theta^t}(\eta \exp^{-1}_{\theta^t}(\hat{\theta}^\tau)) \qquad \text{(AsyncPos)}$$

$$\theta^{t+1} \longleftarrow \exp_{\theta^t}(\eta P_{\theta^\tau \to \theta^t}[g^\tau]) \qquad \text{(AsyncTan)}$$

Here, and for the remainder of the paper, we may abuse notation and write $\exp^{-1}$ to denote *any* member of the preimage when outside the injectivity radius. Other "delay correcting" update rules can be lifted to the Riemannian case, provided their Euclidean assumptions have manifold analogues. For example, DC-ASGD becomes:

$$\theta^{t+1} \longleftarrow \exp_{\theta^t}\left(\eta P_{\theta^\tau \to \theta^t}[g^\tau + \operatorname{Hess} f(x)[\exp^{-1}_{\theta^\tau}(\theta^t)]\right)$$

The outer product of tangent vectors as an unbiased estimator for the Hessian trick used in the original Euclidean formulation can also be applied to our Riemannian version since the operation occurs in tangent space. In Euclidean space, this "stepping vector" can be expressed as a simple linear combination of the ones for AsyncPos and AsyncTan, but this necessitates flatness of the underlying manifold. Given the variety of update rules in the literature, we black-box this choice in our general framework: we assume a function that takes g^τ and outputs a staleness-corrected tangent direction. We then propose a new geometric rule for AsyncBezier.

An important assumption in the convergence proofs of Bonnabel (2013) and Li and Ma (2022) is the *geodesic-completeness* of the manifold $\mathcal{M}_\Theta$: that the exponential map is defined on the entire tangent space. By the Hopf-Rinow theorem, this implies *geodesic-connectedness*: that any two points on the manifold are connected by a geodesic path. Whilst in practice this assumption could be relaxed for e.g. punctured manifolds, most sensible choices of optimisation manifold will be complete, so we adopt this convention for guaranteed existence of $\exp_\theta(\operatorname{grad}_\theta \mathcal{L})$ and $\exp^{-1}_\theta(\hat{\theta})$.

3 THE ASYNCMANIFOLD FRAMEWORK

We may define the "aggregation problem" of AsyncFL as finding the path in parameter space $\gamma : [0, 1] \to \mathcal{M}_\Theta$

between the local and global models and the step size $\eta_g \in [0, 1]$ such that $\gamma(\eta_g)$ is in a low point of both the local and global loss landscapes. The most common paradigm for choosing γ is the *Linear Mode Connectivity* hypothesis: independent neural network minima are often connected by straight lines of low-loss, so γ is simply the straight line $\Theta^{\text{local}} \leftrightarrow \Theta^{\text{global}}$. This assumption often fails to hold, however, although minima may still be connected by polynomial curves (Lubana et al., 2023). Some authors consider a stronger hypothesis that extends to entire low-loss submanifolds connecting more than two minima (Benton et al., 2021), but these approaches based on flat simplicial complexes can encounter the same problem of loss barriers. Instead we make a more immediate generalisation of straight-line connectivity to the Riemannian context that both allows for dynamic adaptation to the solution space geometry and maintains the semantic richness of a manifold learning framework: that there exists a (low-loss) submanifold of $\mathcal{M}_\Theta$ on which the geodesic connection of minima is low-loss. In particular, this subsumes the *Polynomial Mode Connectivity* hypothesis, as we notice that the graph of a Bézier curve is a 1-dimensional submanifold, on which the geodesics trivially follow the polynomial in $\mathbb{R}^\Theta$. An important class of manifolds where the geodesics coincide with a polynomial curve but maintain the dimensionality of $\mathcal{M}_\Theta$ are the **ε-tunnels** (Dold et al., 2025): ε-balls extruded along a Bèzier curve. This enables a variant of **Sharpness-Aware Minimisation (SAM)** (Caldarola et al., 2022) for curve learning, which seeks to improve generalisation ability by increasing solution volume.

With the aggregation problem cast as curve learning, we may now present our proposed solution. We specify the AsyncManifold family of algorithms, where the learned aggregation manifold is arbitrary, and provide a particular implementation in AsyncBezier, where we directly learn geodesics as (quadratic) Bézier curves; finally, we provide a convergence result for the framework, agnostic to the choice of manifold.

Training Step (Client) Given a particular global model Θ^t, the goal of the client is to learn a submanifold (with boundary) $\mathcal{M}_\phi$ of parameter space $\mathcal{M}_\Theta$ around Θ^t.

Our framework rests on a key observation: a wide class of submanifolds can be learned with standard gradient-based methods by choosing a geodesically-connected smooth manifold (with boundary) $\mathcal{M}$ and learning a smooth map $\iota_\phi : \mathcal{M} \to \mathcal{M}_\Theta$, inducing a Riemannian structure on $\mathcal{M}$ by pulling back the metric along the embedding. We call $\mathcal{M}$ equipped with a metric depending smoothly on ϕ the Riemannian manifold $\mathcal{M}_\phi := (\mathcal{M}, g_\phi)$.

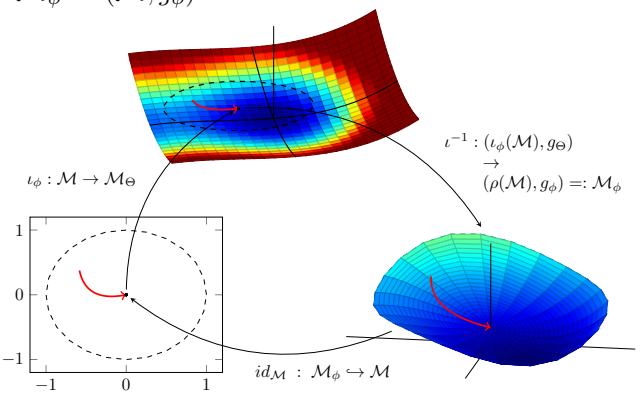

Figure 2: Illustration of our approach to manifold learning. $\mathcal{M} = D_1(\mathbb{R}^2)$ maps into parameter space $\mathcal{M}_\Theta = \mathbb{R}^3$ by the learned embedding. $\iota_\phi(\mathcal{M})$ inherits a Riemannian structure from $\mathcal{M}_\Theta$ via the subspace metric, distorted by ι_ϕ, which is in turn isometric to a retraction of $\mathcal{M}$ equipped with the pullback metric (in this illustration, the retraction $\rho = id$). The curvature of this $\mathcal{M}_\phi$ space thus induces a lower-loss curved path in $\mathcal{M}$, and hence $\mathcal{M}_\Theta$ under the embedding. †

We learn parametrised realisations of $\mathcal{M}$ in $\mathcal{M}_\Theta$ by choosing a smooth map $\iota : \mathcal{M}_\Phi \times \mathcal{M} \to \mathcal{M}_\Theta$, for some R-manifold $\mathcal{M}_\Phi$. This ι has two important features: first, for every $\Theta \in \mathcal{M}_\Theta$, there exists a unique $\phi_\Theta \in \mathcal{M}_\Phi$ such that $\iota_{\phi_\Theta}(\mathcal{M}) = \{\Theta\}$ - inducing a subspace $\mathcal{M}_\Phi^0$ homeomorphic to $\mathcal{M}_\Theta$. This "compression" property is necessary because the FL optimisation state at any given time is a single point in $\mathcal{M}_\Theta$. We are not attempting collaborative learning of a single low-loss manifold, rather an ephemeral low-dimensional manifold on which to do aggregation; to apply the framework to federated manifold learning we choose $\mathcal{M}_\Theta$ as the space of *manifold* parameters and choose a simpler structure for parameter space aggregation (e.g. Bézier curves).

Second, ι_ϕ should be an immersion wherever $\phi \notin \mathcal{M}_\Phi^0$ - this ensures that the pullback metric from $\mathcal{M}_\Theta$ will always induce a Riemannian structure on $\mathcal{M}_\phi$ as soon as the local and global models diverge. Where ι_ϕ is not injective, we will abuse notation and write $\iota_\phi^{-1}(\Theta)$ to mean any member of the Θ preimage.

We may now optimise this embedding using standard Riemannian SGD on $\mathcal{M}_\Phi$. For this, we must choose a sampling distribution $\mathbf{P}$ over $\mathcal{M}$ which approximates the uniform distribution on the geodesic connecting

†In this figure, we have shown $\mathcal{M}_\Theta$ with Riemannian structure corresponding to the loss landscape for illustration purposes - this will not be the case in general and usually the Riemannian structure of $\mathcal{M}_\Theta$ is defined without ℓ. Since evaluating the loss function is costly, we induce a new geometry of $\mathcal{M}_\phi$ via distortions in ι_ϕ

$\iota_\phi^{-1}(\Theta^t)$ to the distinguished *local model* $\omega \in \mathcal{M}$. Starting from Φ_{Θ^t} for the received global model Θ^t, ϕ is then trained against the objective:

$$\min_\phi \mathbb{E}_{S\sim\mathbf{P}}\left[F_i(X;\iota_\phi(S),\Theta^t)\right] := \min_\phi \mathbb{E}_{S\sim\mathbf{P}}\left[\ell_i(X;\iota_\phi(S)) + \frac{\mu}{2}\left\|\iota_\phi(S)-\Theta^t\right\|^2\right] \tag{6}$$

Optimisation proceeds by general Riemannian gradient descent on $\mathcal{M}_\Phi$, sampling $S_k \sim \mathbf{P}_k$ at local batch k - this is possible by the smoothness of the cost function on $\mathcal{M}_\Theta$. After K total rounds of optimisation, the reparametrisation vector

$$v_i^t \in T\mathcal{M}_\Phi := \left(\exp_{\phi_{\Theta^t}}\right)^{-1}(\phi^K) \tag{7}$$

is transmitted back to the server.

Remark. *To perform stochastic analysis we must, separately to any differentiable structure, endow $\mathcal{M}$ and $\mathcal{M}_\Theta$ with probability measures. ι_ϕ must be measurable with respect to them, but the pushforward and latent measures on $\iota_\phi(\mathcal{M})$ need not coincide. In particular, sampling from the uniform distribution on $\iota_\phi(\mathcal{M})$ with respect to the $\mathcal{M}_\Theta$ measure may be possible only by computing a corrected non-uniform distribution on $\mathcal{M}$.*

AsyncBezier learns the aggregation path directly as a low-degree polynomial, representing the simplest choice under this framework of a 1-dimensional $\mathcal{M}$. We set $\mathcal{M} := [0,1]$ and $\mathcal{M}_\Phi = (\mathcal{M}_\Theta)^n$ to be the space of control points for degree-n Bèzier curves in the model space $\mathcal{M}_\Theta$[‡]. ι is then defined by the numerically-stable *de Casteljau's formula* (Delgado et al., 2023) which, for the quadratic case we focus on, is:

$$\iota : (\mathcal{M}_\Theta)^3 \times [0,1] \longrightarrow \mathcal{M}_\Theta \qquad A,B,C,t \longmapsto (1-t)^2A + 2t(1-t)B + t^2C \tag{8}$$

Notice that ι_ϕ is thus almost everywhere an embedding. We then fix the parametrisation such that $\iota_\phi(0) = \Theta^t$ and $\omega := 1$. $\mathbf{P}$ is set to the Dirac delta at 1 for the first K_1 rounds, forcing movement away from the global model, followed by $\mathcal{U}[0,1]$ for the subsequent $K - K_1$. After K_1 rounds, the experiments of Section 4 freeze both endpoints, but freezing the "local" endpoint is optional.

Correction Step (Server) At time step τ, the server receives v_i^t from client i. Since Θ^τ is out of synchronisation with Θ^t, we need a framework for correcting this staleness. To achieve this, we fix a function $\pi : \mathcal{M}_\Theta^2 \times T\mathcal{M}_\Phi \to T\mathcal{M}_\Phi$, mapping learned gradient and a (Θ^t, Θ^τ) pair to the delay-corrected gradient, ensuring that the $\iota_\phi(\mathcal{M})$ this induces always contains Θ^τ. We can view this π as inducing a weak form of smooth fibre bundle from the total space:

$$S_{\phi,\Theta^t} := \left\{\left(\Theta^\tau, \iota_{(\pi_\phi(\Theta^\tau|\Theta^t)}(x)\right) \mid \Theta^\tau \in \mathcal{M}_\Theta, x \in \mathcal{M}\right\}$$

In particular, for AsyncBezier, $\iota_\phi(\mathcal{M}) \cong \mathcal{M}$ for all $\phi \notin \mathcal{M}_\Phi^0$, which is true almost everywhere. The "optimal" bundle would be one where each $\Theta \in \mathcal{M}_\Theta$ is associated with $\mathcal{M}_\phi$ for the optimal ϕ, but this would define an intractable π. Instead, the AsyncManifold framework black-boxes the optimisation (given Θ^t) of the initial client ϕ from the perspective of the server and ensure that the transformation to delay-corrected parameters is simple to reason about. This π can thus be seen as approximating the true gradient at Θ^t via the learned geodesic; convergence is guaranteed as long as its error is at most a constant factor worse than the parallel transport of the curve tangent at Θ^t to Θ^s.

For AsyncBezier, we propose a π incorporating a novel delay-correction procedure that directly leverages a general principle of Riemannian geometry: *orthogonality*. One method deployed successfully in multi-task learning is the (sequential) **Gradient Surgery** approach of Yu et al. (2020). This algorithm considers client update tangents Δ_1, Δ_2 to "conflict" if they have an obtuse angle between them (i.e. $\langle\Delta_1,\Delta_2\rangle_{\mathcal{M}_\Theta} < 0$). Where updates conflict, Δ_1 will be projected into the orthogonal complement subspace of Δ_2, hence any action of Δ_1 in direct opposition to Δ_2 will be cancelled, whilst preserving orthogonal movement. Inspired by this work, we propose the OrthoDC formula, for tunable hyperparameter $\vartheta \in [-1,1]$ and global drift vector $\Delta^g := \exp^{-1}_{\phi_{\Theta^t}}(\phi_{\Theta^\tau})$:

$$\pi(\Theta^t,\Theta^\tau,\Delta) := \begin{cases} \Delta - \mathrm{proj}_{\Delta^g}(\Delta) & \frac{\langle\Delta,\Delta^g\rangle}{\|\Delta\|\cdot\|\Delta^g\|} \le \vartheta \\ \Delta & \text{otherwise} \end{cases}$$

Where $\mathrm{proj}_{\mathbf{b}}(\mathbf{a}) := \frac{\langle\mathbf{a},\mathbf{b}\rangle}{\langle\mathbf{b},\mathbf{b}\rangle}\mathbf{b}$. Setting $\vartheta = 0$ recovers traditional gradient surgery, while $\vartheta = 1$ retains only the update component orthogonal to the global drift. Using $\vartheta = 1$ thus conceptually "factors out" the difference between the Pos and Tan approaches on $T_\Theta\mathcal{M}_\Theta$; factoring out the difference in tangent space leads to the approaches coinciding exactly on flat (Euclidean) manifolds, but only up to the first order otherwise. Finally, the server computes

$$\psi^\tau \longleftarrow \exp_{\phi_{\Theta^\tau}}(\pi(\Theta^t,\Theta^\tau,v_i^t)) \tag{9}$$

Aggregation Step (Server) With a final manifold $\mathcal{M}_{\psi^\tau}$ chosen, we find the tangent vector $v^\tau := \exp^{-1}_{\iota_{\psi^\tau}^{-1}(\Theta^\tau)}(\omega)$ and transition to the next global model by moving part-way along the exponential map. We first define

$$S^{t,\tau} := 1 + \alpha\left(\left\|\Theta^\tau - \hat{\Theta}^\tau\right\| / \left\|\Theta^t - \Theta^\tau\right\| - 1\right) \tag{10}$$

[‡]For geodesic-completeness it can be useful to recall that $\mathcal{M}$ can be sensibly extended to $\mathbb{R}$.

(where $\hat{\Theta}^\tau := \exp_{\Theta^\tau}^{\psi^\tau}(v^\tau)$) for some decay strength hyperparameter $\alpha \in [0,1]$, and finally define the new global model:

$$\Theta^{\tau+1} \longleftarrow \left(\iota_{\psi^\tau} \circ \exp_{\iota^{-1}(\Theta^\tau)}\right)\left(S^{t,\tau} \cdot w_{i_\tau} \eta_g^\tau v^\tau\right) \quad (11)$$

for some global learning rate $\eta_g^\tau \in (0,1]$. This integrates a *staleness penalty*, inspired by Wang et al. (2022), to down-weight desynchronised updates. Clients that are perfectly sequential should have an approximately constant $S^{t,\tau}$ (decaying as the gradient magnitude decreases over time), with faster clients being up-weighted and slower ones down-weighted.

We recall that geodesics are arc-length parametrised and step size in this exponential map is measured according to the $\mathcal{M}_\Theta$ metric pulled back to $\mathcal{M}$. For AsyncBezier, we achieve this by reparametrisation by instead scaling by $\tilde{\eta}_g^\tau$ chosen to ensure that:

$$\left\|\exp_\Theta^{-1}\left(\iota_\phi\left(S^{t,\tau} \cdot w_{i_\tau} \tilde{\eta}_g^\tau\right)\right)\right\|_{\mathcal{M}_\Theta} = \left\|S^{t,\tau} \cdot w_{i_\tau} \eta_g^\tau \exp_\Theta^{-1}(\iota_\phi(1))\right\|_{\mathcal{M}_\Theta} \quad (12)$$

Meta-Aggregation Step (Server) Finally, the server may choose to perform **Stochastic Weight Averaging (SWA)** (Izmailov et al., 2018), where learning rate schedules are fixed or cyclic and the final returned model is an average of models from throughout the latter stages of the learning process. This is done by Karcher mean on $\mathcal{S}$, the server-side manifold. This can, much like $\mathcal{M}$, be embedded into $\mathcal{M}_\Theta$ *a priori* or by learning a parametric ι_{ξ^*} such that:

$$\xi^* = \arg\min_{\xi \in \Xi} \left[\sum_{t \in A} \min_{x \in \iota_\xi(\mathcal{S})} d_\Theta(\Theta^t, x)\right] \quad (13)$$

For some subset of model indices $A \subset [T]$. d_Θ here denotes any metric on $\mathcal{M}_\Theta$, which may or may not coincide with the induced Riemannian one.

See Appendix C for a complete pseudocode description of the AsyncBezier algorithm.

3.1 Convergence Analysis

We may now present our main result on convergence of the framework in general terms; see Appendix A for precise details of the assumptions made for our Riemannian AsyncFL paradigm.

Theorem 1 (Convergence of AsyncManifold)**.** *The* AsyncManifold *algorithm, with no SWA, assumptions as above, and the local learning rate* $\eta_l = \mathcal{O}(1/\max\{2C_1, \sqrt{T}\})$*, converges with:*

$$\frac{1}{T}\sum_{t=0}^{T} \mathbb{E}\left\|\operatorname{grad}\mathcal{L}(\Theta^t)\right\|^2 \leq \mathcal{O}\left(\frac{\lambda_{min}}{Q\eta_g\sqrt{T}}\left[\mathcal{L}(\Theta^0) - \mathbb{E}\mathcal{L}(\Theta^T)\right]\right) + \mathcal{O}\left(\frac{\lambda_{min}}{\sqrt{T}}(C_2 + 2C_3)\right) \quad (14)$$

Where C_1, C_2, C_3 are constants as defined in the proof.

Proof. See Appendix A for details. □

4 EXPERIMENTAL ANALYSIS

We develop a fork of the Flower FL framework (Beutel et al., 2022) which handles asynchronous client updates, evaluating AsyncBezier against a number of baseline methods across a variety of datasets.

4.1 Models and Datasets Used

We focus on three datasets, each representing a different style of task and architecture best-suited to it. We choose two datasets from LEAF (Caldas et al., 2019), the canonical FL benchmark suite, and augment them with a challenging hospital-scale radiography task as an application to the computational health domain. Statistical heterogeneity of the partition into $N = 30$ local instances is modelled in a distinct way for each dataset, representing real-world diversity. The datasets and models under study here are comparatively small compared to contemporary big data pipelines, but this is representative of the typical FL setup where resource-constrained edge devices are collaborating to train models for offline inference. For full details and explanations of each scenario, please see Appendix B.

FEMNIST (Cohen et al., 2017): The canonical OCR dataset on 62 handwritten characters, using preprocessed versions from the LEAF suite. Heterogeneity is introduced synthetically by assigning a different proportion of the samples for each character label to each client, according to a 0.5-Dirichlet distribution. The model architecture is a simple 2-conv, 2-dense CNN.

Shakespeare (Caldas et al., 2019): Again from LEAF, performing character-level sequence prediction on a corpus of Shakespeare plays. No synthetic heterogeneity is used, since every play is already assigned wholly to a single client. For this task, we apply a small, 6-head, GPT 2-like (Radford et al., 2019) transformer.

CXR8 (Wang et al., 2017): Black-and-white chest X-Ray images, labelled for 8 conditions (including *cardiomegaly* and *pneumothorax*) as a multi-hot vector. Similar to Shakespeare, there is heterogeneity already embedded in the data, and we assign the scans from each patient wholly to a single client. We test fine-tuning a ShuffleNet V2 (×1.5) (Ma et al., 2018), using PyTorch's pre-trained ImageNet (Deng et al., 2009) weights.

The proposed AsyncBezier is then evaluated against 4 representative baselines: FedAsync (Xie et al., 2020), DC-ASGD (Xie et al., 2020), FedBuff (Nguyen et al.,

2022), and AsyncFedED (Wang et al., 2022). In addition, to evaluate its influence on our proposal's performance, we implement the standard FedAsync algorithm with the OrthoDC correction rule, terming this FedOrtho where $\vartheta = 1$ and FedGS where $\vartheta = 0$. We differentiate between two versions of our proposed algorithm, with AsyncBezierED using $\alpha = 1$ in the staleness decay parameter and $\alpha = 0$ used otherwise; choice of ϑ, on the other hand, is treated as a tunable hyperparameter, found by line search over $\{-1, 0, 1\}$. For the purposes of side-by-side comparison in this paper, we focus only on those methods which are at their core "SGD-like" in the update rule, so exclude those proposals which introduce momentum terms and further associated hyperparameters to tune.

For these datasets, we focus on the instantiation of AsyncBezier on Euclidean ambient manifolds, i.e. $\mathcal{M}_\Theta = \mathbb{R}^\Theta$, and leave evaluation of the "full Riemannian" version on real-world data to future work.

4.2 Results

Table 1 shows the test set accuracy results for both our proposal and the baseline methods over the Shakespeare and FEMNIST datasets, with Table 2 showing the macro AUROC and AUPRC results for CXR8. To give an accurate impression of the balance between accuracy at convergence and speed to reach a target error level, we choose an error (defined as 1 - AUROC for CXR8) threshold e close to the converged value and report T_e, the number of communication rounds at which this threshold is reached.

Each model was trained for 360 communication rounds (720 total epochs, mean 24/client), with $e = 0.20, 0.50, 0.25$ for the FEMNIST, Shakespeare, and CXR8 datasets respectively. Each scenario was repeated with three different random seeds, with the means and standard deviations across runs being reported in the table.

We can make the following observations: **(1)** The optimal choice of delay-corrected update rule is sensitive to dataset. In particular, we see that different values of ϑ are optimal for AsyncBezier on different problems, illustrating the ways in which the geometric relationships between clients are task-dependent. **(2)** AsyncBezier (with optimal choice of α) always outperforms FedAsync, with an average +1.05% performance and -54 epochs to target error. **(3)** Indeed, our proposal outperforms every other baseline on every metric (by an average +.17% performance advantage vs. the runner-up with -9 epochs) other than CXR8 AUPRC, where it ranks 3rd behind AsyncFedED and FedGS. The disparity between AUROC and AUPRC results may be attributed to the difficulty of this task, especially for the lightweight ShuffleNet model, reflected in the poor overall performance of AUPRC scores, with high class imbalance and some conditions significantly harder to detect than others. The CXR8 results nonetheless provide a useful benchmark on a less well-studied real-world dataset. Future work should evaluate AsyncManifold specialised to larger models capable of higher baseline AUPRC, where the solution space geometry may exhibit more stable characteristics for our method to take advantage of. **(4)** The proposals based on OrthoDC usually outperform FedAsync, but the gains of AsyncBezier cannot solely be attributed to this since they still consistently have an advantage of an average +.41% performance and -25 epochs vs. FedGS/FedOrtho. **(5)** Indeed, our proposal is the only method other than DC-ASGD to outperform naive FedAsync on every dataset. Our proposal also outperforms DC-ASGD on every dataset, by an average of .31% accuracy/AUROC and 13 communication rounds. In general, we can attribute the superior performance to the greater fitness of our *quadratic mode connection* hypothesis to dataset geometries than that of *linear mode connection*.

4.2.1 Client Fairness

Under both statistical and size heterogeneity, it is important to ensure that no client's performance is neglected simply because that client is under-represented in the global loss computation. We term this desirable property *client fairness* (Mohri et al., 2019), and it is particularly relevant in the healthcare setting, where clients will often correspond to hospitals with different patient demographics (Rieke et al., 2020).

Following Thakur et al. (2025), we borrow two classical econometric formulae for calculating the "inequality" of a sampled distribution that goes beyond simple variance analysis: the *Gini Coefficient* and *Theil Index* (see Appendix B.4). Figure 3 shows these values computed according to the Accuracy/Macro AUROC value distribution for the best performing global model across the decentralised client validation sets; we note that the two metrics broadly agree on the ordering of methods, with the Theil index showing slightly more sensitivity.

There is comparatively little consistent variation amongst the methods, with FedOrtho, FedGS, and DC-ASGD in particular all close together. The -ED variants both show a consistent poorer performance and higher variance than their respective non-scaling counterparts (most noticeable in AsyncBezierED), which is expected due to their intentional down-weighting (to varying degrees) of certain straggling clients.

Our proposal (with $\alpha = 0$) consistently shows a slight improvement over all other baselines, with an average

(a) FEMNIST		
Method	**Test Acc. (%)**	T_e
FedAsync	85.01 ± 0.11	137 ± 6.6
FedOrtho	84.83 ± 0.08	133 ± 3.4
FedGS	85.38 ± 0.14	149 ± 1.2
DC-ASGD	85.25 ± 0.17	135 ± 1.6
FedBuff	84.62 ± 0.35	174 ± 2.1
AsyncFedED	85.48 ± 0.29	114 ± 5.7
AsyncBezier	$\mathbf{85.82 \pm 0.14}$	130 ± 2.6
AsyncBezierED	85.67 ± 0.14	$\mathbf{114 \pm 0.5}$

(b) Shakespeare		
Method	**Test Acc. (%)**	T_e
FedAsync	50.60 ± 0.06	296 ± 10.0
FedOrtho	52.76 ± 0.54	202 ± 14.0
FedGS	52.87 ± 0.18	209 ± 11.0
DC-ASGD	52.01 ± 0.06	230 ± 8.5
FedBuff	50.84 ± 0.34	287 ± 13.0
AsyncFedED	53.03 ± 0.29	188 ± 7.0
AsyncBezier	52.07 ± 0.05	209 ± 2.0
AsyncBezierED	$\mathbf{53.13 \pm 0.13}$	$\mathbf{164 \pm 2.5}$

Table 1: Percentage test set accuracy across methods for the FEMNIST and Shakespeare datasets.

CXR8 Macros			
Method	**Test Macro AUROC**	**Test Macro AUPRC**	T_e
FedAsync	77.93 ± 0.01	25.72 ± 0.21	140 ± 6.0
FedOrtho	77.91 ± 0.13	25.90 ± 0.29	134 ± 2.5
FedGS	77.85 ± 0.10	$\mathbf{26.31 \pm 0.18}$	141 ± 6.0
DC-ASGD	78.32 ± 0.39	26.06 ± 0.34	146 ± 1.5
FedBuff	77.82 ± 0.02	25.89 ± 0.14	172 ± 2.0
AsyncFedED	77.45 ± 0.08	25.37 ± 0.13	144 ± 2.5
AsyncBezier	$\mathbf{78.44 \pm 0.04}$	26.12 ± 0.11	132 ± 6.8
AsyncBezierED	77.89 ± 0.12	26.11 ± 0.12	$\mathbf{116 \pm 9.0}$

Table 2: Macro AUROC and AUPRC scores for each method across the 8 conditions in the CXR8 dataset.

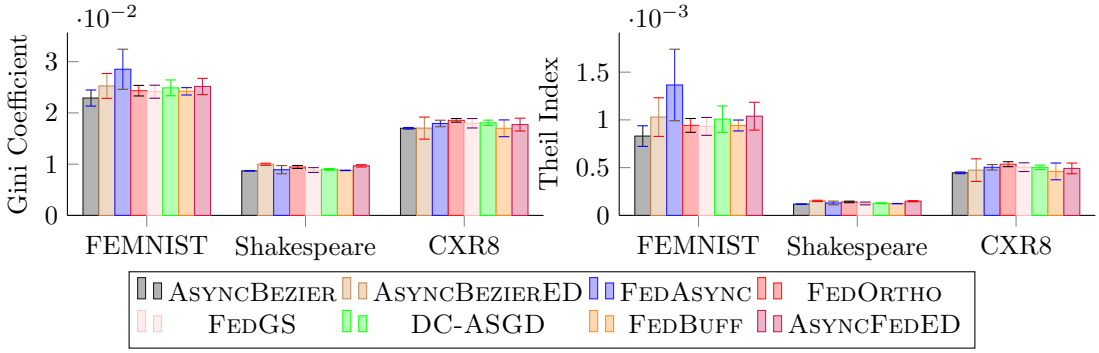

Figure 3: Bar plots of (unweighted) Gini Coefficient and Theil Index computed for each method over the model performance on each client's validation set.

of 4.7×10^{-4} Gini coefficient and 4.0×10^{-5} Theil index. We conjecture this may be attributable to the generalisable minima-seeking behaviour of the curve learning process. This shows the clear promise of our framework for applications in the aforementioned medical contexts, along with its strong performance on the CXR8 dataset.

4.2.2 Effect of Local Epoch Counts

An important consideration when weighing the use of AsyncBezier is whether the computational overhead of curve-fitting epochs is justified, given that these epochs could instead be allocated to standard pointwise SGD." To investigate the effect of increased local SGD epochs on final method performance, we re-run FEMNIST training on each of the methods (excluding AsyncFedED) for $T = 360$ communication rounds with each of four different epoch counts. For AsyncBezier we use $\min(K, 2)$ curve-fitting epochs when running with K SGD epochs.

Figure 4 shows the results of this investigation. As expected, every method sees mean gains of 1.33% when

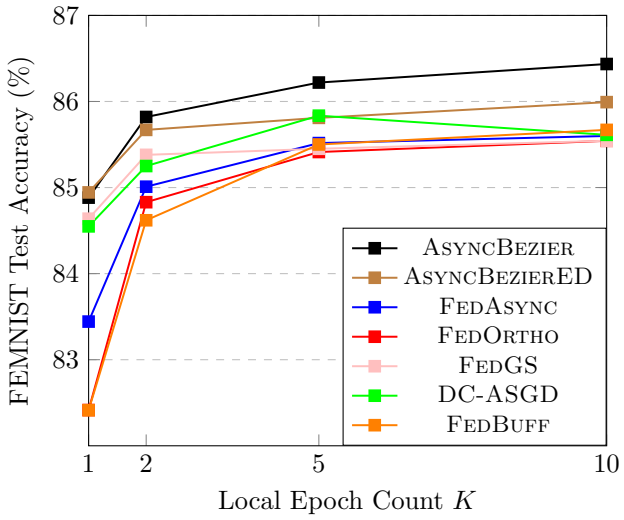

Figure 4: Accuracy of each method on FEMNIST after 360 communication rounds by local epoch count

moving from 1 to 2 epochs and .45% from 2 to 5, attributable in both cases to larger step sizes allowing greater progress towards convergence in the fixed T. When moving from 5 to 10 epochs, however, the gains for most methods are minimal ($\mu = .09\%$), with DC-ASGD even seeing a decline in accuracy of .22%, attributable to client heterogeneity leading to divergences in the local gradients becoming compounded with the increased time between synchronisation steps. Crucially for this evaluation, our method outperforms every baseline at every K value; indeed, even our $K = 2$ variant surpasses all other methods regardless of their local epoch count. In particular, it is more efficient to spend 2 epochs in pointwise SGD and 2 epochs in our curve learning procedure (as in the main results of this section) than it is to spend 5 total epochs in pointwise SGD and proceed by any other proposal. Furthermore, our method shows the greatest ability to take advantage of more local epochs, being the only one to reach over 86% accuracy at higher counts. This suggests an improved capacity to handle divergent local gradients due to our consideration of local solution space geometry.

5 CONCLUSION AND FUTURE WORK

In this paper, we have developed AsyncBezier, a new AsyncFL algorithm augmenting SGD-based methods with greater knowledge of client loss landscape geometry, and proven its convergence by situating it within our AsyncManifold Riemannian framework. Our proposal is supported by a novel staleness correction method derived from orthogonal complement projection to minimise conflicting updates from heterogenous clients. In evaluations of both CNN and Transformer architectures on general-purpose and healthcare datasets, our proposal is shown to be empirically superior to strong baselines in terms of both accuracy, AUROC, and fairness. Whilst our method does introduce computational overhead compared to FedAsync, we have shown in Section 4.2.2 that our curve learning procedure makes better use of computation budget for higher epoch counts than pure pointwise SGD.

Future work would include deeper analyses of more complex implementations of the AsyncManifold framework, especially on non-Euclidean underlying manifolds, including providing stronger convergence bounds with more specific method-wise assumptions. Applications of AsyncBezier to real-world healthcare contexts are a natural next step given the promising CXR8 results, particularly on larger clusters with many resource-constrained clients. Another important future direction for our framework, especially in this domain, is combining it with for ensuring differential privacy.

Acknowledgements

DAC was funded by an NIHR Research Professorship; a Royal Academy of Engineering Research Chair; and the InnoHK Hong Kong Centre for Cerebro-cardiovascular Engineering (COCHE); and was supported by the National Institute for Health Research (NIHR) Oxford Biomedical Research Centre (BRC) and the Pandemic Sciences Institute at the University of Oxford. This work was also supported by the NVIDIA Academic Grant Program.

References

Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2021.106775. URL https://www.sciencedirect.com/science/article/pii/S0950705121000381.

Chenhao Xu, Youyang Qu, Yong Xiang, and Longxiang Gao. Asynchronous federated learning on heterogeneous devices: A survey. *Computer Science Review*, 50:100595, 2023. ISSN 1574-0137. doi: https://doi.org/10.1016/j.cosrev.2023.100595. URL https://www.sciencedirect.com/science/article/pii/S157401372300062X.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletarì, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N. Galtier, Bennett A. Landman, Klaus Maier-Hein, Sébastien Ourselin, Micah Sheller, Ronald M. Summers, Andrew Trask, Daguang Xu,

Maximilian Baust, and M. Jorge Cardoso. The future of digital health with federated learning. *npj Digital Medicine*, 3(1), September 2020. ISSN 2398-6352. doi: 10.1038/s41746-020-00323-1. URL http://dx.doi.org/10.1038/s41746-020-00323-1.

Andrew A. S. Soltan, Anshul Thakur, Jenny Yang, Anoop Chauhan, Leon G. D'Cruz, Phillip Dickson, Marina A. Soltan, David R. Thickett, David W. Eyre, Tingting Zhu, and David A. Clifton. Scalable federated learning for emergency care using low cost microcomputing: Real-world, privacy preserving development and evaluation of a COVID-19 screening test in UK hospitals. *medRxiv*, 2023. doi: 10.1101/2023.05.05.23289554. URL https://www.medrxiv.org/content/early/2023/05/11/2023.05.05.23289554.

Soheila Molaei, Anshul Thakur, Ghazaleh Niknam, Andrew Soltan, Hadi Zare, and David A Clifton. Federated learning for heterogeneous electronic health records utilising augmented temporal graph attention networks. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1342–1350. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/molaei24a.html.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Kilian Pfeiffer, Martin Rapp, Ramin Khalili, and Jörg Henkel. Federated learning for computationally constrained heterogeneous devices: A survey. *ACM Computing Surveys*, 55(14s):1–27, July 2023. ISSN 1557-7341. doi: 10.1145/3596907. URL http://dx.doi.org/10.1145/3596907.

Cong Xie, Sanmi Koyejo, and Indranil Gupta. Asynchronous federated optimization, 2020. URL https://arxiv.org/abs/1903.03934.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 6391–6401, Red Hook, NY, USA, 2018. Curran Associates Inc. URL https://dl.acm.org/doi/10.5555/3327345.3327535.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems 2 (MLSys 2020)*, 2020. URL https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

John Nguyen, Kshitiz Malik, Hongyuan Zhan, Ashkan Yousefpour, Michael Rabbat, Mani Malek, and Dzmitry Huba. Federated learning with buffered asynchronous aggregation. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 3581–3607. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/nguyen22b.html.

Qiyuan Wang, Qianqian Yang, Shibo He, Zhiguo Shi, and Jiming Chen. AsyncFedED: Asynchronous federated learning with euclidean distance based adaptive weight aggregation, 2022. URL https://arxiv.org/abs/2205.13797.

Shuxin Zheng, Qi Meng, Taifeng Wang, Wei Chen, Nenghai Yu, Zhi-Ming Ma, and Tie-Yan Liu. Asynchronous stochastic gradient descent with delay compensation. In *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 4120–4129. PMLR, 2017. URL https://proceedings.mlr.press/v70/zheng17b/zheng17b.pdf.

Yujia Wang, Shiqiang Wang, Songtao Lu, and Jinghui Chen. FADAS: Towards federated adaptive asynchronous optimization. In *Proceedings of the 41st International Conference on Machine Learning (ICML 2024)*, 2024. URL https://dl.acm.org/doi/10.5555/3692070.3694190.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations (ICLR 2021)*, 2021. URL https://arxiv.org/abs/2003.00295.

Chang-Wei Shi, Yi-Rui Yang, and Wu-Jun Li. Ordered momentum for asynchronous sgd. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385. URL https://dl.acm.org/doi/10.5555/3737916.3738455.

Yan Sun, Li Shen, Shixiang Chen, Liang Ding, and Dacheng Tao. Dynamic regularized sharpness aware minimization in federated learning: Approaching

global consistency and smooth landscape. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 32991–33013. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/sun23h.html.

Dennis Grinwald, Philipp Wiesner, and Shinichi Nakajima. Federated learning over connected modes. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2024)*, 2024. URL https://arxiv.org/abs/2403.03333.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations (ICLR 2020)*, 2020. URL https://arxiv.org/abs/2002.06440.

N. Joseph Tatro, Pin-Yu Chen, Payel Das, Igor Melnyk, Prasanna Sattigeri, and Rongjie Lai. Optimizing mode connectivity via neuron alignment. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020. URL https://dl.acm.org/doi/abs/10.5555/3495724.3497007.

Weishi Li, Yong Peng, Miao Zhang, Liang Ding, Han Hu, and Li Shen. Deep model fusion: A survey, 2023. URL https://arxiv.org/abs/2309.15698.

Maximilian Nickel and Douwe Kiela. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3779–3788. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/nickel18a.html.

Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, September 2013. ISSN 1558-2523. doi: 10.1109/tac.2013.2254619. URL http://dx.doi.org/10.1109/TAC.2013.2254619.

Jiaxiang Li and Shiqian Ma. Federated learning on riemannian manifolds. *ArXiv*, abs/2206.05668, 2022. URL https://api.semanticscholar.org/CorpusID:249625803.

Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Kwing Hei Li, Titouan Parcollet, Pedro Porto Buarque de Gusmão, and Nicholas D. Lane. Flower: A friendly federated learning research framework, 2022. URL https://arxiv.org/abs/2007.14390.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of DNNs. In *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/hash/be3087e74e9100d4bc4c6268cdbe8456-Abstract.html.

Ekdeep Singh Lubana, Eric J. Bigelow, Robert P. Dick, David Krueger, and Hidenori Tanaka. Mechanistic mode connectivity. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 22965–23004. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/lubana23a.html.

Tailin Zhou, Jun Zhang, and Danny H.K. Tsang. Mode connectivity in federated learning with data heterogeneity. In *2023 57th Asilomar Conference on Signals, Systems, and Computers*, pages 1600–1604, 2023. doi: 10.1109/IEEECONF59524.2023.10476886. URL https://arxiv.org/abs/2309.16923.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence (UAI 2018)*, 2018. URL https://api.semanticscholar.org/CorpusID:3833416.

Hao Guo, Jiyong Jin, and Bin Liu. Stochastic weight averaging revisited. *Applied Sciences*, 13(5), 2023. ISSN 2076-3417. doi: 10.3390/app13052935. URL https://www.mdpi.com/2076-3417/13/5/2935.

Maxime Haddouche, Paul Viallard, Umut Şimşekli, and Benjamin Guedj. A PAC-Bayesian Link Between Generalisation and Flat Minima. In *ALT 2025 - 36th International Conference on Algorithmic Learning Theory*, pages 1–31, Milan, Italy, February 2025. URL https://hal.science/hal-04455639.

Debora Caldarola, Barbara Caputo, and Marco Ciccone. Improving generalization in federated learning by seeking flat minima. In *Computer Vision – ECCV 2022: 17th European Conference*, page 654–672, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-20049-6. doi: 10.1007/978-3-031-20050-2_38. URL https://doi.org/10.1007/978-3-031-20050-2_38.

John M. Lee. *Riemannian Manifolds: An Introduction to Curvature*. 1997. URL https://api.semanticscholar.org/CorpusID:119659969.

Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao,

Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1223–1231, Red Hook, NY, USA, 2012. Curran Associates Inc. URL https://dl.acm.org/doi/10.5555/2999134.2999271.

Gregory W. Benton, Wesley J. Maddox, Sanae Lotfi, and Andrew Gordon Wilson. Loss surface simplexes for mode connecting volumes and fast ensembling. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 769–779. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/benton21a.html.

Daniel Dold, Julius Kobialka, Nicolai Palm, Emanuel Sommer, David Rügamer, and Oliver Dürr. Paths and ambient spaces in neural loss landscapes. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics (AISTATS 2025)*, 2025. URL https://arxiv.org/abs/2503.03382.

J. Delgado, E. Mainar, and J.M. Peña. On the accuracy of de casteljau-type algorithms and bernstein representations. *Computer Aided Geometric Design*, 106:102243, 2023. ISSN 0167-8396. doi: https://doi.org/10.1016/j.cagd.2023.102243. URL https://www.sciencedirect.com/science/article/pii/S0167839623000754.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546. URL https://dl.acm.org/doi/10.5555/3495724.3496213.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H. Brendan McMahan, Virginia Smith, and Ameet Talwalkar. LEAF: A benchmark for federated settings. *Workshop on Federated Learning for Data Privacy and Confidentiality*, 2019. URL https://arxiv.org/abs/1812.01097.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters, 2017. URL https://arxiv.org/abs/1702.05373.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. ChestX-Ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3462–3471. IEEE, July 2017. doi: 10.1109/cvpr.2017.369. URL http://dx.doi.org/10.1109/CVPR.2017.369.

Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV 2018)*. Springer, 2018. URL https://arxiv.org/abs/1807.11164.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848. URL https://api.semanticscholar.org/CorpusID:57246310.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2019. URL https://arxiv.org/abs/1902.00146.

Anshul Thakur, Soheila Molaei, Patrick Schwab, Danielle Belgrave, Kim Branson, and David A. Clifton. Optimising clinical federated learning through mode connectivity-based model aggregation. In Yingzhen Li, Stephan Mandt, Shipra Agrawal, and Emtiyaz Khan, editors, *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pages 163–171. PMLR, 03–05 May 2025. URL https://proceedings.mlr.press/v258/thakur25a.html.

Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. 2019. URL https://arxiv.org/abs/1909.06335.

Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7252–7261. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/yurochkin19a.html.

Amartya Sen and James Foster. *On Economic Inequality*. Oxford University Press, 12 1973. ISBN 9780198281931. doi: 10.1093/0198281935.001.0001. URL https://doi.org/10.1093/0198281935.001.0001.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [**Yes**] *AsyncFL setting discussed in introduction, see Appendix A for precise assumptions made and a clear convergence result.*
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [**Yes**] *Relevant discussion of epoch counts can be found in the experimental analysis section*
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [**Yes**] *Code uploaded to git repo.*

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [**Yes**] *See Appendix A*
   (b) Complete proofs of all theoretical results. [**Yes**] *See Appendix A*
   (c) Clear explanations of any assumptions. [**Yes**] *See Appendix A*

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [**Yes**] *Code uploaded to git repo.*
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [**Yes**] *See Appendix B*
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [**Yes**] *Explained alongside relevant figures*
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [**Yes**] *See Appendix B.*

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [**Yes**] *The LEAF datasets, original CXR8 paper, and PyTorch ImageNet weights are all cited where they are mentioned.*
   (b) The license information of the assets, if applicable. [**Not Applicable**]
   (c) New assets either in the supplemental material or as a URL, if applicable. [**Yes**] *Code uploaded to git repo.*
   (d) Information about consent from data providers/curators. [**Not Applicable**]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [**Not Applicable**] *Anonymisation process discussed in CXR8 original paper*

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [**Not Applicable**]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [**Not Applicable**]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [**Not Applicable**]

Aggregation on Learnable Manifolds for Asynchronous Federated Optimisation: Supplementary Materials

A PROOF OF CONVERGENCE

We begin with the standard assumptions of non-convex optimisation, lifted to the Riemannian context with appropriately adjusted definitions:

Assumption 1 (L-Smooth Loss)**.** *There exists a constant L_Θ such that:*

$$\|\operatorname{grad}\mathcal{L}(X) - P_{X\to Y}[\operatorname{grad}\mathcal{L}(Y)]\| \leq L_\Theta \|X - Y\|$$

For all $X, Y \in \mathcal{M}_\Theta$

Assumption 2 (Bounded Loss Gradient)**.** *There exists some constant G such that $\|\operatorname{grad}\mathcal{L}(\Theta)\| \in [0, G]$ for all $\Theta \in \mathcal{M}_\Theta$. The unbiased gradient estimates used for stochastic local steps should also have norm upper bounded by G.*

For simplicity in this paper, we will adopt the following "weakly homogenous" setting, which assumes that stochastic gradients w.r.t. $\mathcal{L}_i$ are an unbiased estimator for $\operatorname{grad}\mathcal{L}$.

Assumption 3 (Unbiased Client Heterogeneity)**.** *We have that the local stochastic gradients of the cost function, taken across both the choice of client index and the local entropy during the training step, are unbiased estimators for the global cost. In particular, the expectation of the local stochastic gradient equals the true global gradient.*

Formally, the cost function in question in the previous assumption is the once whose variance is bounded in:

Assumption 4 (Bounded Stochastic Divergence from Geodesic)**.** *Suppose that local steps at time step t are taken against the cost function:*

$$\tilde{G}_t(\phi) := \int_{\tilde{\mathcal{M}}_t \subseteq \mathcal{M}} \mathcal{L}(\iota_\phi(x))\, dp_t(x) \tag{15}$$

For some probability distribution p_t on $\tilde{\mathcal{M}}_t$, chosen as some subset of $\mathcal{M}$. Then there exist some constants σ_1, σ_2 such that:

$$\mathbb{E}\left\|\operatorname{grad}\tilde{G}_t(\phi) - \operatorname{grad}G(\phi)\right\|^2 \leq \sigma_1^2 + \sigma_2^2 \|\operatorname{grad}G(\phi)\|^2$$

Where:

$$G(\phi) := \int_1^0 \mathcal{L}(\iota_\phi(\gamma_t(\lambda)))\, d\lambda \tag{16}$$

For γ_t the geodesic connecting $\iota_\phi^{-1}(\Theta^t) \to \omega$.

This modification to the standard bounded stochastic variance assumption seems quite strong on $(n > 1)$-dimensional manifolds, but can be achieved in a number of ways leveraging smoothness and shrinking off-geodesic volume. This is a product of the "ephemerality" of the learned manifolds being used to compute steps rather than as part of an effort to learn a low-loss manifold in itself.

Next, we need to bound the reasonableness of functions chosen in the AsyncManifold instantiation:

Assumption 5 (Lipschitz and Bounded Curvature Embedding)**.** *There exists a constant M_Φ such that, for all $x, y \in \mathcal{M}$ and $\phi, \psi \in \mathcal{M}_\Phi$:*

$$\|\iota(x,\phi) - \iota(y,\psi)\| \leq M_\Phi \|(x,\phi) - (y,\psi)\| \tag{17}$$

ι should also be L-smooth, and from this we have L-smoothness of the lifted loss:

$$\left\|\operatorname{grad}(\mathcal{L}\iota)(\phi,x) - P_{(\psi,y)}^{(\phi,x)}[\operatorname{grad}(\mathcal{L}\iota)(\psi,y)]\right\| \leq L_\Phi \|X - Y\|$$

Finally, the operator norm of the second fundamental form (geodesic curvature) of ι_ϕ should be uniformly bounded for any $\phi \in \mathcal{M}_\Phi$ and any $x \in \mathcal{M}$:

$$\left\|\mathrm{I\!I}_{\iota_\phi}(x)\right\|_{op} \leq C \tag{18}$$

Assumption 6 (Embedding Immersivity)**.** *$\iota_\omega : \mathcal{M}_\Phi \to \mathcal{M}_\Theta$ should be an immersion for any $\omega \in \mathcal{M}$. This ensures local injectivity of the differential map, and we furthermore enforce that the smallest eigenvalue of its adjoint is bounded everywhere uniformly above zero by $\sqrt{|\lambda_{min}|}$.*

The following assumption quantifies the "well-behavedness" of our delay correction procedure: we should finish with a stepping tangent which is at most a constant times worse as an approximation to $\operatorname{grad}\mathcal{L}(Y)$ than the parallel transport:

Assumption 7 (Delay Correction Quality)**.** *Let $\gamma_{\Theta,\phi}$ denote the $\iota_\phi^{-1}(\Theta) \to \omega$ geodesic for a given parametrisation ϕ and let $(\iota \circ \gamma)_{\Theta,\phi}$ denote its embedding into $\mathcal{M}_\Theta$. Then there exists some constant Q such that, for any $\phi \in \mathcal{M}_\Phi$, $X, Y \in \mathcal{M}_\Theta$:*

$$\begin{aligned} &\left\langle \operatorname{grad}\mathcal{L}(Y), (\iota \circ \gamma)'_{Y,\pi(X,Y,\phi)}(Y) \right\rangle \\ &\geq Q \left\langle \operatorname{grad}\mathcal{L}(Y), P_{X \to Y}[(\iota \circ \gamma)'_{X,\phi}(X)] \right\rangle \end{aligned} \tag{19}$$

We ensure that clients will always participate with at most finite gaps:

Assumption 8 (Bounded Staleness)**.** *Suppose an update from client i arrives at time τ, with the local copy of the client model being Θ^t. Then $\mathbb{E}\left[\|\Theta^\tau - \Theta^t\| \mid \Theta^t\right] \leq S \max_{t' \in [t..\tau-1]} \left\|\gamma'_{\phi_k^{t'}}(0)\right\|$.*

Note that the above constraint is immediately implied by *client ergodicity* where, as $T \to \infty$, every client participates infinitely often in the updates, with non-vanishing probability. In the heterogenous client distribution setting, this ergodicity assumption would be required explicitly to ensure convergence of the global loss.

For completeness, we reproduce the statement of the theorem, with the full definition of the constants $C_{\{1,2,3\}}$:

Theorem 1 (Convergence of AsyncManifold)**.** *The* AsyncManifold *algorithm, with no SWA, assumptions as above, and the local learning rate $\eta_l = \mathcal{O}(\frac{1}{\max\{2C_1, \sqrt{T}\}})$, converges with:*

$$\begin{aligned} \frac{1}{T}\sum_{t=0}^{T} \mathbb{E}\left\|\operatorname{grad}\mathcal{L}(\Theta^t)\right\|^2 \leq\ & \mathcal{O}\left(\frac{\lambda_{min}}{Q\eta_g\sqrt{T}}\left[\mathcal{L}(\Theta^0) - \mathbb{E}\mathcal{L}(\Theta^T)\right]\right) \\ & + \mathcal{O}\left(\frac{\lambda_{min}}{\sqrt{T}}(C_2 + 2C_3)\right) \end{aligned}$$

Where:

$$\begin{aligned} C_1 &:= \frac{(1+\sigma_2^2)L_\Phi}{2} - KL_\iota^2(L_\Theta + GC)\left(\frac{1}{6} + \frac{\eta_g(1+\alpha(S^{-1}-1))}{4Q}\right) \\ C_2 &:= \frac{1}{\beta}\left[(1-\alpha)\bar{\eta}_g L_\Theta S K^2 L_\iota^2 G^2 + \alpha L_\Theta K^2 L_\iota^2 G^2\right] \\ C_3 &:= KL_\Phi\sigma_1^2 \qquad\qquad \beta := 1 + \alpha(S^{-1}-1) \end{aligned}$$

Proof of Theorem 1. We assume that the manifold parameters are trained by Riemannian SGD on $\mathcal{M}_\Phi$ for K steps against the loss function:

$$G_{i,\Theta}(\phi) := \int_0^1 \mathcal{L}(\gamma_\phi(t))\, dt \tag{20}$$

Where i is a given client index and $\gamma_\phi : [0,1] \to \mathcal{M}_\Theta$ is the constant-speed (scaled) geodesic connecting ω and $\iota_\phi^{-1}(\Theta^t)$, embedded under ι_ϕ. Notice that, by L-smoothness of the lifted loss and the fundamental theorem of calculus, this is L-smooth. By Assumption (1), we may bound the loss at ϕ from below:

$$\mathcal{L}_i(\gamma_\phi(t)) \geq \mathcal{L}_i(\gamma_\phi(0)) + \left\langle \operatorname{grad}\mathcal{L}_i(\gamma_\phi(0)), t\gamma'_\phi(0)\right\rangle - \frac{L_\Theta + GC}{2}t^2\left\|\gamma'(0)\right\|^2 \tag{21}$$

Where the GC term comes from the difference $|\mathcal{L}(\exp_\Theta(t\gamma'(0))) - \mathcal{L}(\gamma(t))| \leq G \left\|\exp_\Theta(t\gamma'(0)) - \gamma(t)\right\|$, which in turn is bounded by $\frac{GC}{2} t^2 \left\|\gamma'(0)\right\|^2$ due to Assumption 5. Integrating over t to find a bound on G:

$$G_{i,\Theta}(\phi) \geq \mathcal{L}_i(\gamma_\phi(0)) + \left\langle \operatorname{grad} \mathcal{L}_i(\gamma_\phi(0)), \gamma'_\phi(0) \right\rangle \int_0^1 t \, dt - \frac{L_\Theta}{2} \left\|\gamma'_\phi(0)\right\|^2 \int_0^1 t^2 \, dt \tag{22}$$

$$= \mathcal{L}_i(\Theta) + \frac{1}{2} \left\langle \operatorname{grad} \mathcal{L}_i(\gamma_\phi(0)), \gamma'_\phi(0) \right\rangle - \frac{L_\Theta + GC}{6} \left\|\gamma'_\phi(0)\right\|^2 \tag{23}$$

We can bound the expectation for ϕ_k:

$$\mathbb{E} G_{i,\Theta}(\phi_k) \geq \mathbb{E}\mathcal{L}_i(\Theta) - \underbrace{\left[\frac{1}{2}\mathbb{E}\left\langle -\operatorname{grad} \mathcal{L}_i(\Theta), \gamma'_{\phi_k}(0) \right\rangle + \frac{L_\Theta + GC}{6} \mathbb{E}\left[\left\|\gamma'_{\phi_k}(0)\right\|^2\right]\right]}_{\Delta_1} \tag{24}$$

Similarly, we can use the learning procedure to bound $G_\Theta(\phi))$ from above. By smoothness and the bounded variance Assumption 4:

$$\mathbb{E} G_{i,\Theta}(\phi_{k+1}) \leq G_{i,\Theta}(\phi_k) - \eta_l \left\langle -\operatorname{grad} G_{i,\Theta}(\phi_k), \mathbb{E} g_{i,k} \right\rangle + \frac{\eta_l^2 L_\Phi}{2} \mathbb{E}\left[\left\|g_{i,k}\right\|^2\right] \tag{25}$$

$$\leq G_{i,\Theta}(\phi_k) - \eta_l \left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\|^2 + \frac{\eta_l^2 L_\Phi}{2} \left[(1+\sigma_2^2)\left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\|^2 + \sigma_1^2\right] \tag{26}$$

$$= G_{i,\Theta}(\phi_k) - \left(\eta_l - \eta_l^2 \frac{(1+\sigma_2^2) L_\Phi}{2}\right) \left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\|^2 + \eta_l^2 \frac{\sigma_1^2 L_\Phi}{2} \tag{27}$$

Telescoping the sum of $G(\phi_k) - G(\phi_{k+1})$ over $[K]$ yields:

$$\mathbb{E}[G_{i,\Theta}(\phi_k)] \leq G_{i,\Theta}(\phi_0) - \underbrace{\left(\eta_l - \eta_l^2 \frac{(1+\sigma_2^2) L_\Phi}{2}\right) \sum_{k=0}^{K} \mathbb{E}\left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\|^2 + \eta_l^2 \frac{K L_\Phi \sigma_1^2}{2}}_{\Delta_2} \tag{28}$$

Recalling that ϕ_0 is a point parametrisation, we have that $G_\Theta(\phi_0) = \mathcal{L}(\Theta)$. We can now combine these bounds, noticing that $\mathcal{L}(\Theta) - \Delta_1 \leq \mathcal{L}(\Theta) - \Delta_2$, hence $\Delta_1 \geq \Delta_2$:

$$\frac{1}{2}\mathbb{E}\left\langle -\operatorname{grad} \mathcal{L}_i(\Theta), \gamma'_{\phi_k}(0) \right\rangle + \frac{L_\Theta + GC}{6} \mathbb{E}\left[\left\|\gamma'_{\phi_k}(0)\right\|^2\right] \geq \left(\eta_l - \eta_l^2 \frac{(1+\sigma_2^2) L_\Phi}{2}\right) \sum_{k=0}^{K} \mathbb{E}\left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\|^2 - \eta_l^2 \frac{K L_\Phi \sigma_1^2}{2} \tag{29}$$

We can now apply the smoothness of $\mathcal{L}$ on $\mathcal{M}_\Theta$ to yield an upper bound in similar form to (21):

$$\mathbb{E}\mathcal{L}(\Theta^{t+1}) \leq \mathcal{L}(\Theta^t) - \underbrace{Q\eta_g \mathbb{E}\Sigma^s(\alpha) \left\langle -\operatorname{grad} \mathcal{L}_i(\Theta^t), P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)] \right\rangle}_{T_1} + \eta_g^2 \frac{L_\Theta + GC}{2} \mathbb{E}\left[\left\|\gamma'_{\phi_k^t}(0)\right\|^2\right] \tag{30}$$

$$\text{where } \Sigma^s(\alpha) := 1 + \alpha \max\left[\frac{\left\|\gamma'(0)_{\phi_k^t}\right\|}{\|\Theta^s - \Theta^t\|} - 1, s - 1\right] \tag{31}$$

Where we convert to a parallel transport term with Assumption 7. Rearranging T_1:

$$\begin{aligned}
T_1 &= \bar{\eta}_g \Sigma^s(\alpha) \left\langle -\operatorname{grad} \mathcal{L}_i(\Theta^s) + P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}_i(\Theta^t)] - P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}_i(\Theta^t)], \mathbb{E} P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)] \right\rangle \\
&\geq \bar{\eta}_g \Sigma^s(\alpha) \left\langle -P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}_i(\Theta^t)], \mathbb{E} P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)] \right\rangle \\
&\quad + \bar{\eta}_g \Sigma^s(\alpha) \left\langle -\operatorname{grad} \mathcal{L}_i(\Theta^s) + P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}_i(\Theta^t)], \mathbb{E} P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)] \right\rangle \\
&= \bar{\eta}_g \Sigma^s(\alpha) \left\langle -\operatorname{grad} \mathcal{L}_i(\Theta^t), \bar{\eta}_g \mathbb{E}\gamma'_{\phi_k^t}(0) \right\rangle - \underbrace{\bar{\eta}_g \Sigma^s(\alpha) \left\langle \operatorname{grad} \mathcal{L}_i(\Theta^s) - P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}(\Theta^t)], \mathbb{E} P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)] \right\rangle}_{T_2}
\end{aligned} \tag{32}$$

We choose the global learning rate η_g to ensure that $\left\|\eta_g \gamma'_{\phi_k}(0)\right\| = \left\|\bar{\eta}_g \exp_{\Theta}^{-1}(\iota(\phi_k, \omega))\right\|$. By the Lipschitz property of embedded diameter and the fact that ϕ_0 is a point parametrisation, we have that:

$$\left\|\exp_{\Theta}^{-1}(\iota(\phi_k, \omega))\right\| \leq L_\iota \left\|\exp_{\phi_0}^{-1}(\phi_k)\right\| \leq L_\iota \eta_l \sum_{k=0}^{K} \left\|\operatorname{grad} G_{i,\Theta}(\phi_k)\right\| \tag{33}$$

Where the last inequality is by the geodesic triangle and AM-GM inequalities. This enables us to continue bounding T_2:

$$T_2 \leq \Sigma^s(\alpha(\left\|\operatorname{grad} \mathcal{L}_i(\Theta^s) - P_{\Theta^t \to \Theta^s}[\operatorname{grad} \mathcal{L}_i(\Theta^t)]\right\| \cdot \left\|\mathbb{E} P_{\Theta^t \to \Theta^s}[\gamma'_{\phi_k^t}(0)]\right\| \tag{34}$$

$$\leq \left((1-\alpha) + \alpha \frac{\left\|\gamma'(0)_{\phi_k^t}\right\|}{\left\|\Theta^s - \Theta^t\right\|}\right) L_\Theta \left\|\Theta^s - \Theta^t\right\| \cdot \left\|\mathbb{E}\gamma'_{\phi_k^t}(0)\right\| \tag{35}$$

$$\leq (1-\alpha)\left[L_\Theta \sum_{i\in[t..s]} \eta_g \left\|\gamma'_i(0)\right\| \left\|\mathbb{E}\gamma'_s(0)\right\|\right] + \alpha \left\|\mathbb{E}\gamma'_{\phi_k^t}(0)\right\|^2 \tag{36}$$

$$\leq (1-\alpha)\left[L_\Theta \sum_{i\in[t..s]} \left[\bar{\eta}_g K L_\iota^2 \eta_l^2 \sum_{k\in[K]} \left\|\operatorname{grad} G_{i,\Theta^i}(\phi_k^i)\right\|^2\right]\right] + \alpha \left\|\mathbb{E}\gamma'_{\phi_k^t}(0)\right\|^2 \tag{37}$$

$$\leq (1-\alpha)\eta_l^2 \bar{\eta}_g L_\Theta S K^2 L_\iota^2 G^2 + \alpha \eta_l^2 L_\Theta K^2 L_\iota^2 G^2 \tag{38}$$

Substituting (38) into (32), then accumulating into (30) along with (29):

$$\mathbb{E}\mathcal{L}(\Theta^{t+1}) \leq \mathcal{L}(\Theta^t) - 2Q\eta_g \Sigma^s(\alpha)\left(\eta_l - \eta_l^2 \frac{(1+\sigma_2^2)L_\Phi}{2}\right) \sum_{k=0}^{K} \mathbb{E}\left\|\operatorname{grad} G_{i,\Theta^t}(\phi_k^t)\right\|^2 \tag{39}$$

$$+ \left(\eta_g^2 \frac{L_\Theta + GC}{2} + 2Q\Sigma^s(\alpha)\eta_g \frac{L_\Theta + GC}{6}\right) \mathbb{E}\left\|\gamma'_{\phi_k^t}(0)\right\|^2 \tag{40}$$

$$+ 2Q\Sigma^s(\alpha)\eta_g \eta_l^2 \frac{K L_\Phi \sigma_1^2}{2} + Q\eta_g T_2 \tag{41}$$

$$\leq \mathcal{L}(\Theta^t) + 2Q\Sigma^s(\alpha)\eta_g \eta_l^2 \underbrace{K L_\Phi \sigma_1^2}_{C_3}$$

$$+ Q\bar{\eta}_g \eta_l^2 \Sigma^s(\alpha) \underbrace{\frac{1}{1+\alpha(S^{-1}-1)}\left[(1-\alpha)\bar{\eta}_g L_\Theta S K^2 L_\iota^2 G^2 + \alpha L_\Theta K^2 L_\iota^2 G^2\right]}_{C_2}$$

$$- 2Q\eta_g \eta_l \Sigma^s(\alpha)\left(1 - \eta_l C_1\right) \sum_{k=0}^{K} \mathbb{E}\left\|\operatorname{grad} G_{i,\Theta^t}(\phi_k^t)\right\|^2 \tag{42}$$

$$\text{where } C_1 := \frac{(1+\sigma_2^2)L_\Phi}{2} - K L_\iota^2 (L_\Theta + GC)\left(\frac{1}{6} + \frac{\eta_g(1+\alpha(S^{-1}-1))}{4Q}\right) \tag{43}$$

We rearrange and telescope over $[T]$ to find a convergence bound in terms of the Riemannian gradient on $\mathcal{M}_\Phi$:

$$\frac{1}{T}\sum_{t=0}^{T} \mathbb{E}\left\|\operatorname{grad} G_{i,\Theta^t}(\phi^t)\right\|^2 \leq \frac{\mathcal{L}(\Theta^0) - \mathbb{E}\mathcal{L}(\Theta^T)}{2TQ\eta_l\eta_g(1+\alpha(S^{-1}-1))\left(1-\eta_l C_1\right)} + \eta_l \frac{C_2 + 2C_3}{2(1-\eta_l C_1)} \tag{44}$$

We need now to translate this to a bound w.r.t. $\mathcal{M}_\Theta$. Recalling that the differential (and hence its adjoint) are linear operators, by a standard linear algebraic argument we have:

$$\left\|(D\iota_\omega)^*_\phi(v)\right\|^2 = \left\langle (D\iota_\omega)^*_\phi(v), D\mathcal{L}_\Theta(v)\right\rangle = \left\langle v, ((D\iota_\omega)(D\iota_\omega)^*)_\phi(v)\right\rangle \geq \lambda_{\min}\left\|v\right\|^2 \tag{45}$$

For $\lambda_{\min}$ the smallest absolute eigenvalue of $D(\iota_\omega)_\Theta$. Accordingly:

$$\left\|\operatorname{grad} G_{i,\Theta^t}(\phi^t)\right\|^2 = \left\|D(\iota_\omega)^*_\phi[\operatorname{grad} \mathcal{L}_i(\Theta^t)]\right\|^2 \geq \frac{1}{\lambda_{\min}}\left\|\operatorname{grad} \mathcal{L}_i(\Theta^t)\right\|^2 \tag{46}$$

We substitute (46) into (44), simply multiplying by $\lambda_{\min}$ (bounded above zero by Assumption 6).

Notice that we have used $\mathcal{L}$ without considering the proximal term in this analysis. This is because our result bounds the loss gradient on $\mathcal{M}_\Theta$ at Θ^t by bounding the loss gradient on ϕ at ϕ^t_{init} - hence the proximal and raw losses coincide when evaluated at this point, so we can conclude a bound on the raw loss immediately, although we have technically abused notation referring to the client optimising over $\mathcal{L}$. The result then follows from an appropriate choice for η_l. □

B EXPERIMENTAL DETAILS

Experiments were run on two Nvidia RTX GPUs (1x 5070, 1x 3070), each simulating 15 clients. For each method implemented, we use a local Adam optimiser on the FedProx objective (per Equation 6) for 2 epochs with $\eta_l = \mu = 0.001$, only tuning global parameters of the aggregation framework.

The most influential hyperparameter is the choice of (constant) global learning rate η_g, which for all methods was found by line search over $\{0.25, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 5.0\}$ - see Table 3 for the choices by method and dataset.

B.0.1 Scheduler details

Since each device simulates multiple clients, we need to implement scheduler that emulates the message order of a realistic asynchronous federation. Although a deterministic queue could emulate perfectly resource-equal clients where computation time is exactly proportional to dataset size, we opt for a stochastic version that reproduces the starvation behaviour of small-vs-large dataset clients and more accurately captures real-world delay variations.

1. Clients are pushed into a priority queue based on

$$\texttt{num\_local\_batches} + k \cdot \texttt{simulation\_elapsed\_time}$$

   k here is an empirical "batches-per-second" constant which we set $k = 2$, approximately the average number of training batches the available GPUs can execute per second when operating in parallel, divided by the number of clients. This is equivalent to maintaining the queue strictly by `num_batches` and decaying the keys of long-waiting clients over time. The client at the back of this queue will be approximately the client which "has the most local batches remaining", but larger-dataset clients will be slightly more starved by this approach than a deterministic algorithm which assumes instantaneous communication and server-side processing.

2. When each GPU finishes execution of the current client, it chooses the client with index i in the queue as the next task to execute with probability p_i:

$$p_i := \frac{(N-i)}{\sum_{j \leq N} j}$$

We observed little sensitivity of method rankings to the exact scheduling algorithm: earlier experiments used a simpler method but produced similar trends, motivating our decision not to conduct a costly ablation study. Once updates are processed by the GPUs, aggregation is performed on a central server thread and clients immediately dispatched back to the waiting pool with updated model weights.

B.1 FEMNIST

805,263 28×28 black-and-white images, representing a single alphanumeric character (hence one of 62 classes). Samples were heterogenously partitioned into 30 clients according to the Dirichlet distribution ($\alpha = 0.5$) on class labels.

Figure 5 shows the full CNN architecture used for this dataset (ReLU activations not shown).

Contributions from each client are weighted by proportion of dataset seen by that client. For DC-ASGD, the λ_t parameter is set dynamically with $\lambda_0 = 2.0$, as proposed by Zheng et al. (2017). For FedBuff, we use $K = 10$ as recommended in Nguyen et al. (2022).

Global Learning Rates (η_g)			
Method	**FEMNIST**	**Shakespeare**	**CXR8**
FedAsync	3.0	5.0	2.0
FedOrtho	3.0	5.0	2.0
FedGS	1.0	2.5	1.0
DC-ASGD	1.0	2.5	1.0
FedBuff	1.0	2.0	1.0
AsyncFedED	0.25	1.5	0.5
AsyncBezier	0.5	1.5	0.5
AsyncBezierED	0.25	1.0	0.25

Table 3: Global learning rates η_g selected by line search for each method across each validation dataset.

For AsyncFedED, we follow the original paper (Wang et al., 2022), and use $\overline{\gamma} = 1.0, \kappa = 1$ (notice that λ in their notation is subsumed by η_g in ours). Increasing $gam\bar{m}a$ to above 1 increases training stability, but increases wall-clock time far more and results in worse performance in communication round terms. We note that in the early stages, staleness computed according to their Equation (6) can exhibit high variance that can throw training off. Accordingly, we do not compute staleness dynamically until after a short "warm-up" period, using the modified:

$$\tilde{\gamma}(i,t) = \begin{cases} \gamma(i,t) & t > 10 \\ \overline{\gamma} & \text{otherwise} \end{cases} \tag{47}$$

AsyncFedED is unique among methods tested in using adaptive per-client epoch counts. All our convergence rate results are computed according to communication round count as opposed to wall-clock time, but we do not notice much advantage given to the method, which achieves similar results to other baselines when measured according to communication rounds, despite taking far greater wall-clock time than FedAsync. We can possibly attribute this to the reduced performance of the FedAsync update rule as the number of local epochs increases outweighing any task-balancing issues.

For AsyncBezier, we set $\vartheta = 1$, using the "orthogonalising" version of the OrthoDC update rule.

B.1.1 On Label Heterogeneity

For completeness, we provide a brief background to and motivation for our choice of 0.5-Dirichlet label heterogeneity in this simulation. Non-uniform label partitioning is a widely-accepted method of simulating heterogeneity in federated learning studies (Hsu et al., 2019), where the samples of K-categorical data $(\mathbf{x}, y)$ for $y \in [K]$ are distributed amongst clients so each client will have the majority of its data drawn from a small subset of possible y values; in the case of FEMNIST, this corresponds to each client having the majority of its data drawn from a small subset of handwritten characters. The Dirichlet partitioning method we use is adapted from the description in Yurochkin et al. (2019). This is the same method implemented in Flower's `DirichletPartitioner`, although we do not use the balancing feature enabled there by default. Suppose we have N clients and K classes:

1. For each label $y \in [K]$, we sample a point $(p_{y,1}, ..., p_{y,N})$ on the standard $(N-1)$-simplex (i.e. $\sum_{i \in [N]} p_{y,i} = 1$) using the *symmetric Dirichlet distribution* with p.d.f.:

$$\mathbf{p}_\alpha(p_1, ..., p_N) = \frac{\Gamma(\alpha N)}{\Gamma(\alpha)^N} \prod_{i=1}^{N} p_i^{\alpha - 1}$$

$\alpha \in [0, \infty)$ here is called the **concentration parameter** and dictates the partition imbalance. A smaller α value gives a more heterogeneous partitioning: $\alpha = 0$ gives perfect imbalance, where there will be exactly one i such that $p_i = 1$ and the rest will be 0, $\alpha = 1$ leaves the distribution equivalent to (normalised) uniform, and as $\alpha \to \infty$ the variance of the distribution $\to 0$ and the partitioning tends towards perfect equality. We follow the aforementioned paper and use $\alpha = 0.5$, a midpoint that gives realistic imbalance whilst not

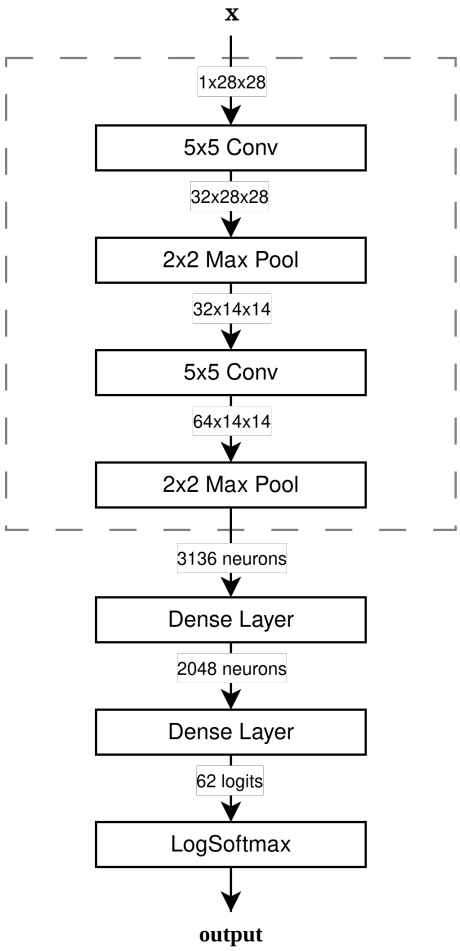

Figure 5: CNN Architecture for FEMNIST

collapsing entirely into a multi-task learning problem; this choice of symmetric $\alpha = 0.5$ is also widespread as the "Jeffrey's prior" for Bayesian inference on Dirichlet data with unknown concentration.

2. For the training samples D_y corresponding to each label $y \in [K]$, we partition them uniformly at random so client i receives $\lfloor p_{y,i}|D_y| \rfloor$ samples.

We make some statistical observations about our particular setting, which uses $\alpha = 0.5$, $N = 30$, and $K = 62$:

- The expected mean over all clients is that $\sim 52\%$ of its data is drawn from a pool of 9 characters and $\sim 91\%$ of its data drawn from a pool of 30 characters. We would further expect that every client (i.e. expectation of minimum amongst a population of 30 clients) will have $> 50\%$ of its data drawn from a pool of 12 and $> 90\%$ from a pool of 34. This illustrates the way in which $\alpha = 0.5$ introduces significant yet realistic and tractable heterogeneity.
- This strategy (when balancing is not used) samples the Dirichlet proportion vector independently for each K and does not make any attempt to balance the dataset sizes assigned to each client, so each client will not only exhibit statistical heterogeneity, but also heterogeneity in dataset size. For our choice of parameters, the smallest-dataset client is expected to have around half the size as the largest-dataset client ($\mathbf{E}[\sum_{y\in[K]} p_{i,y}] \approx 1.40$ for smallest vs ≈ 2.86 for largest).

B.2 Shakespeare

The dialogue lines are first separated by speaker and then windowed into 80-character sequences, for a total of 4,027,181 samples drawn from 35 plays. We allocate each play wholly to a distinct client - since there are 30 clients, 5 will receive 2 plays each, simulating real-world clients which have a disproportionate share of the samples.

We use the nanoGPT framework [https://github.com/karpathy/nanoGPT] to build a GPT-2 like character-level transformer with 6 layers, 6 heads, a 128-dimensional embedding, and dropout $p = 0.1$. We train for next-character prediction given an 80-character input sequence. Most hyperparameters other than η_g remain the same, but:

- For AsyncFedED we maintain $\overline{\gamma} = 1$, which gives far superior performance when compared to $\overline{\gamma} = 3$ (and at faster wall-clock).
- For AsyncBezier, we instead set $\vartheta = 0$, using the "gradient surgery" version of the OrthoDC update rule.

B.3 CXR8

This is a dataset of 112,120 128×128 black-and-white chest X-Ray images. 8 conditions (*Atelectasis*, *Cardiomegaly*, *Effusion*, *Infiltration*, *Mass*, *Nodule*, *Pneumonia*, *Pneumothorax*) are labelled for and the model is trained to detect their presence, encoded as a multi-hot vector to allow for co-incidence. The data is drawn from scans of 30,805 patients, with each assigned wholly to one of 30 clients.

For CXR8, we use the ShuffleNet V2 architecture (Ma et al., 2018), expanded to the $\times 1.5$ version. We use the weights available from PyTorch (https://docs.pytorch.org/vision/main/models/generated/torchvision.models.shufflenet_v2_x1_5) which have been pre-trained on the general-purpose ImageNet dataset. The CXR8 images are then rescaled to 128×128 and reshaped to 3 channels in order to match ImageNet input before being used to fine-tune the model.

ϑ remains $= 0$ for AsyncBezier and $\overline{\gamma} = 1$ for AsyncFedED.

B.4 Fairness Calculations

For completeness, we provide the method to compute the Gini Coefficient and Theil Index as used in Figure 3; both definitions are sourced from Sen and Foster (1973). The Gini Coefficient is a measure of pairwise variance in a sample $X = \{x_1, ..., x_N\}$, normalised by the sample mean $\bar{X}$:

$$\text{Gini}(X) := \frac{1}{2N^2\bar{X}} \sum_{i \in [N]} \sum_{j \in [N]} |x_i - x_j| \tag{48}$$

Intuitively, it measures the difference in area between the plot of cumulative relative "wealth" (here, the values of x_i) against cumulative proportion of the population for the observed sample and the plot that would be yielded from the uniform distribution between minimum and maximum values (a straight line).

The Theil Index is a measure derived instead from information theory, quantifying the difference between the Shannon entropy of the observed distribution of proportional "wealth" and the entropy of the same uniform distribution:

$$\text{Theil}(X) := \frac{1}{N\bar{X}} \sum_{i \in [N]} x_i \log\left(\frac{x_i}{\bar{X}}\right) \tag{49}$$

We note that these are both simply measures of concentration for X's distribution, but this is a valid proxy for inequality as distance from the "most equal" uniform distribution.

C FULL PSEUDOCODE OF THE PROPOSED ASYNCBEZIER ALGORITHM

Algorithm 1: AsyncBezier (Server)

Hyperparameters: Global learning rate schedule η_g^τ, staleness decay $\alpha \in [0,1]$, client weights $\{w_i\}$, client count N.

Initialize model parameters Θ^0 and dispatch initial weights to each of the clients $i = 1, ..., N$;
for $\tau = 0, 1, 2, \ldots, T$ **do**
  Receive $(v_i^t, t) \leftarrow \text{ClientStep}_{i_\tau}(\Theta^t, t)$ from some client i_τ;
  $\psi^\tau \leftarrow (\exp_{\phi_{\Theta^\tau}} \circ \text{OrthoDC}_\vartheta)(\Theta^t, \Theta^\tau, v_i^t)$;
  Compute candidate $\hat{\Theta}^\tau \leftarrow \iota_{\psi^\tau}(1)$;
  Compute staleness penalty $S^{t,\tau}$ according to Equation (10);
  Compute pulled-back learning weight $\tilde{\eta}_g^\tau$ according to Equation (12);
  $\Theta^{\tau+1} \leftarrow \iota_{\psi^\tau}\left(S^{t,\tau}\, w_{i_\tau}\, \tilde{\eta}_g^\tau\right)$;
  Send $\Theta^{\tau+1}$ as new parameters to client i_τ;
end
if *Using SWA with ensemble indices* I **then**
  return $\arg\min_{\Theta \in \mathcal{M}_\Theta} \sum_{i \in I} \left\|\Theta - \Theta^i\right\|^2$;
else
  return Θ^T;
end

Algorithm 2: $\text{ClientStep}_i(\Theta^t, t)$

Input: Current global model Θ^t, time step t
Hyperparameters: Local epochs K, warm-up epochs $K_1 < K$, proximal weight μ.

Initialise control points $\phi^{(0)} = (A, B, C) \leftarrow (\Theta^t, \Theta^t, \Theta^t)$;
Define Bézier map $\iota_\phi(\lambda) := (1-\lambda)^2 A + 2\lambda(1-\lambda)B + \lambda^2 C$;
for $k = 1, \ldots, K$ **do**
  Sample minibatch $X \sim \mathcal{D}_i$;
  if $k \le K_1$ **then**
    Compute the update vector $v_{i,k} \leftarrow \nabla_C F_i(X;\, \iota_{\phi^{(k-1)}}(1),\, \Theta^t)$; `// Defined in Equation (6).`
  else
    Sample S_k according to $\mathcal{U}[0,1]$;
    `// If freezing both endpoints, take derivative w.r.t.` B `only.`
    Compute the update vector $v_{i,k} \leftarrow \nabla_{B,C} F_i(X;\, \iota_{\phi^{(k-1)}}(S_k),\, \Theta^t)$;
  end
  Update $\phi^{(k)} \leftarrow \phi^{(k-1)} - \eta_l v_{i,k}$;
end
Compute reparametrisation vector $v_i^t \leftarrow \exp^{-1}_{\phi^{(0)}}(\phi^{(K)})$;
return (v_i^t, t) to the server;

Algorithm 3: $\text{OrthoDC}_\vartheta(\Theta^t, \Theta^\tau, v_i^t)$

Input: Stale global model Θ^t, current global model Θ^τ, client update vector v_i^t.
Hyperparameters: Conflict threshold $\vartheta \in [-1, 1]$.

Compute global drift $\Delta^g \leftarrow \exp^{-1}_{\phi_{\Theta^t}}(\phi_{\Theta^\tau})$;

return $\begin{cases} \Delta - \text{proj}_{\Delta^g}(\Delta) & \text{if } \frac{\langle \Delta, \Delta^g \rangle}{\|\Delta\| \cdot \|\Delta^g\|} \le \vartheta \\ \Delta & \text{otherwise} \end{cases}$; `// where` $\text{proj}_{\mathbf{b}}(\mathbf{a}) = \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{\langle \mathbf{b}, \mathbf{b} \rangle}\, \mathbf{b}$