FEDBSE: FEDERATED BAYESIAN STATE EVOLUTION FOR OPEN-CLIENT MEDICAL IMAGE SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated medical image segmentation enables collaborative learning across multiple institutions without sharing raw patient data. However, most existing methods assume a fixed set of participating clients before training, which does not reflect real-world medical collaboration networks where new institutions may continue to join after deployment. Restarting federated training introduces additional communication and optimization costs, while updating the shared model may degrade performance on existing clients. To address this issue, we propose FedBSE, a federated Bayesian state evolution framework for medical image segmentation with open client participation. FedBSE explicitly separates shared segmentation knowledge from client-specific variations and freezes the shared components after the initial federated training stage. For each newly arriving client, the method uses local labeled data and the current state prior to derive a closed-form estimate in a low-dimensional client-state space. This avoids iterative optimization of the full segmentation network and keeps the predictors of existing clients unchanged. The inferred client state and its local posterior uncertainty are then adopted to update the server-maintained state distribution through Gaussian moment matching. The updated distribution serves as the prior for subsequent clients. Experiments on three public multi-center medical image segmentation datasets show that FedBSE achieves competitive segmentation performance while substantially reducing new-client onboarding costs. It also enables the client-state distribution to evolve continuously as new clients arrive.

1 INTRODUCTION

Medical image segmentation is important for lesion identification, quantitative analysis, and clinical decision-making. Robust segmentation models often require multi-institutional data to handle domain shifts caused by differences in imaging devices, acquisition protocols, patient populations, and annotation practices Rieke et al. (2020); Li et al. (2021). However, privacy and data governance constraints often prevent raw medical data from being centrally shared Kaissis et al. (2020). Federated learning enables multi-center collaboration by keeping data local and exchanging only model parameters or statistical information Sheller et al. (2020). Most existing federated segmentation methods assume a fixed set of clients throughout training Li et al. (2020); Karimireddy et al. (2020). This assumption is often unrealistic, as new hospitals or imaging centers may join after deployment and introduce new data distributions Liu et al. (2021); Jiang et al. (2023). We therefore study open-client federated medical image segmentation, where the segmentation task and label space remain unchanged, but the client set can expand over time. After initial federated training, the system must adapt to newly arriving clients with local annotations while preserving the performance of existing clients. As illustrated in Fig. 1, conventional retraining-based adaptation and the proposed FedBSE follow different strategies for handling newly arriving clients.

Without a dedicated client integration mechanism, the system typically has two choices: restarting federated training or directly deploying the existing model to the new client. Restarting federated training can use the new client's annotations, but it requires existing institutions to participate in multiple communication rounds and high-dimensional parameter updates McMahan et al. (2017). Updating the shared model with the new client's data may also degrade performance on existing clients. This couples new-client adaptation with existing-client stability Nguyen et al. (2026). Another option is to apply the existing model to unseen clients and rely on out-of-federation general-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

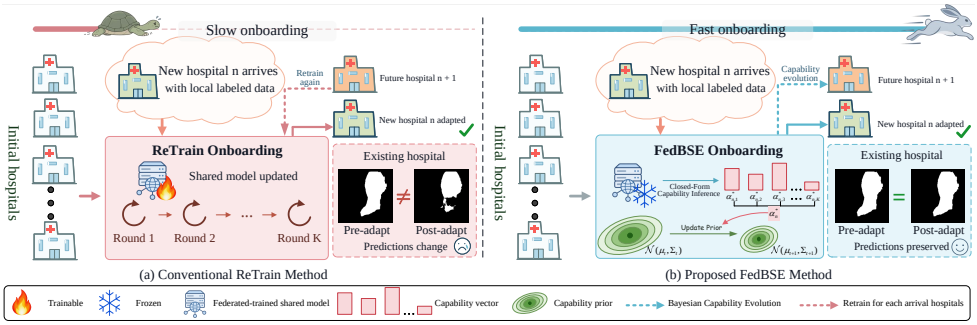


Figure 1: Comparison of retraining-based adaptation and FedBSE. (a) Retraining updates the shared model for new clients. (b) FedBSE performs low-dimensional capability inference while keeping the shared model fixed.

ization Ma et al. (2024). This avoids additional federated optimization, but it cannot use the new client's local annotations or convert its information into knowledge that can benefit future clients. Therefore, open-client expansion must consider three aspects: **adapting to new clients**, **preserving existing-client performance**, and **accumulating knowledge as clients arrive sequentially**.

A key challenge is that traditional shared-parameter federated frameworks encode both cross-institutional shared knowledge and client-specific variations in the same high-dimensional parameters. Incorporating information from new institutions often requires updating the shared model, which couples new-client adaptation with existing-client stability. This increases the cost of high-dimensional optimization and communication, and may also change the prediction functions of existing clients. As a result, it is difficult to maintain both adaptation efficiency and service stability during continuous expansion. To address this issue, we propose FedBSE, a Federated Bayesian State Evolution framework based on Decoupled Modeling for Open-client Federated Segmentation. FedBSE explicitly separates client-specific variations from high-dimensional shared parameters and reformulates new-client integration as inference and evolution in a low-dimensional capability space.

Specifically, FedBSE consists of three interconnected components: Initial Capability Learning (ICL), Closed-Form Capability Inference (CCI), and Bayesian Capability Accumulation (BCA). ICL jointly learns the shared segmentation backbone, the variation basis generator, and client-specific capability coefficients during initial federated training. It also establishes the cross-client capability distribution. After training, the shared components are frozen. When a new client arrives, CCI combines its local annotations with the current capability prior and derives the low-dimensional capability coefficients in closed form under a local quadratic approximation. This avoids iterative optimization of the full segmentation network. Since the shared components and the capability coefficients of existing clients remain unchanged, onboarding a new client does not alter their prediction functions. BCA then uses the inferred capability coefficients and local uncertainty to update the server-side capability distribution through Gaussian moment matching. The updated distribution serves as the prior for subsequent clients. In this way, FedBSE replaces repeated high-dimensional model re-optimization with low-dimensional capability inference and statistical state updates.

To validate these designs, we evaluate FedBSE under both leave-one-client-out and sequential-arrival settings. FedBSE achieves competitive new-client segmentation performance with substantially lower onboarding overhead. It also preserves the prediction functions of existing clients during onboarding. Sequential-arrival experiments further show that accumulated capability information can support the inference of subsequent clients. The main contributions are summarized as follows:

- We formulate open-client federated medical image segmentation, where new institutions may join after the initial federated training. To support this setting, we decouple cross-client shared segmentation knowledge from client-specific distribution shifts through a low-dimensional capability space, turning high-dimensional model adaptation into low-dimensional client inference without retraining the shared model.
- We devise a structured capability modeling framework that jointly learns a shared segmentation backbone, input-dependent variation bases, and client-specific capability coefficients. The variation bases capture reusable directions of cross-client variation, while the capability coefficients describe how each client combines these directions. This design pro-

vides an explicit and compact representation of client heterogeneity instead of absorbing it into shared network weights.

- We develop a closed-form capability inference and Bayesian accumulation mechanism for newly arriving clients. A new client is adapted in closed form under a Gaussian-prior regularized local quadratic model, without iterative optimization of the shared model. The inferred capability and local uncertainty are then incorporated into the global capability distribution through Gaussian moment matching, enabling continuous capability accumulation while keeping existing client predictors unchanged.

2 Related Work

2.1 Federated Learning with Predefined Client Sets

Existing federated learning studies mainly focus on data heterogeneity under predefined client sets. Methods based on proximal regularization Li et al. (2020), control variates Karimireddy et al. (2020), and personalized aggregation Zhang et al. (2021) have been used to reduce client drift and optimization bias caused by non-IID data Li et al. (2020); Karimireddy et al. (2020); Sun et al. (2025). In federated medical image segmentation, existing studies improve generalization across known institutions by preserving local statistics, calibrating features or predictions, enhancing multi-level features, and aligning cross-site representations Li et al. (2021); Xu et al. (2022); Wang et al. (2022); Xie et al. (2024); Zhao et al. (2026). FedEvi further uses model uncertainty to estimate client reliability and adjust aggregation weights Chen et al. (2024). Other studies have also considered heterogeneity caused by imbalanced label distributions Galati et al. (2024). These methods improve robustness and personalization for predefined client sets. However, their learning and model updates still rely on a known set of clients, making them difficult to directly extend to continuously arriving clients after deployment.

2.2 Federated Learning with Dynamic Client Expansion

Existing studies on post-deployment clients mainly focus on unseen-domain generalization, post-deployment personalization, and dynamic client onboarding. Federated domain generalization methods train only on source clients and aim to improve performance on unseen sites. For example, FedDG expands cross-client distributions in the frequency space for medical image segmentation Liu et al. (2021), while later methods explore adversarial domain generation, feature-style transfer, and adaptive frequency interpolation Che et al. (2025); Ren et al. (2026); Wang et al. (2026). These methods do not use labeled data from newly arriving clients. Post-deployment personalization methods instead adapt models to new clients. IOP-FL performs personalized prediction for external clients through test-time model routing Jiang et al. (2023), while ODPFL-HN generates personalized models for unlabeled new clients using a hypernetwork Amosy et al. (2024). However, these methods do not explicitly consider knowledge accumulation across sequential client arrivals.

Recent work has started to study dynamic client populations. Chang et al. consider changes in the federated objective when clients join or leave and use adaptive initialization to accelerate re-optimization Chang et al. (2024). pFedDSH studies progressive client onboarding under a fixed task and label space while preserving existing-client performance Nguyen et al. (2026). In contrast, FedBSE considers continuously arriving medical clients with labeled local support sets. It freezes the shared components after initial federated training, performs low-dimensional capability inference for each new client, and updates the capability distribution through Gaussian moment matching. This allows subsequent clients to use an updated prior without restarting federated training of the shared model.

3 Method

3.1 Problem Formulation

In real-world multi-institutional medical image segmentation, federated learning allows multiple medical institutions to collaboratively train segmentation models without sharing raw data. However, the set of participating institutions is not fixed. As new hospitals or data centers join, both the

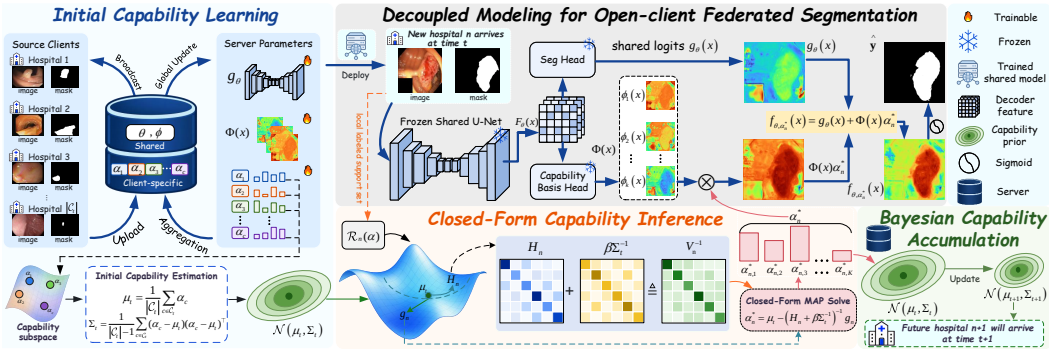


Figure 2: Overall framework of FedBSE for open-client federated medical image segmentation. The framework separates shared segmentation knowledge from client-specific variations, enabling efficient adaptation to newly arriving clients and continuous capability accumulation over time.

client set and the corresponding family of data distributions expand dynamically. Different institutions may use different imaging devices and scanning protocols, and may involve different patient populations and lesion distributions. Therefore, a newly added client may introduce a new data distribution and change the optimization objective of the existing federated segmentation model. Let $\mathcal{C}(t)$ and $\mathcal{P}(t) = \{\mathbb{P}_c \mid c \in \mathcal{C}(t)\}$ denote the set of clients participating in federated medical image segmentation and their corresponding family of data distributions at time t, respectively. For a client c, the risk $\mathcal{R}_c(\theta)$ denotes the expected segmentation loss of model parameters θ on its local medical images and annotations. Conventional federated learning optimizes $\theta_t^* = \arg\min_\theta \sum_{c\in\mathcal{C}(t)} \mathcal{R}_c(\theta)$ over the current distribution family. When a new medical institution joins, the distribution family expands from $\mathcal{P}(t)$ to $\mathcal{P}(t+1)$, and the joint optimization objective becomes

$$\theta_{t+1}^* = \arg\min_\theta \left(\sum_{c\in\mathcal{C}(t)} \mathcal{R}_c(\theta) + \mathcal{R}_{\text{new}}(\theta) \right), \tag{1}$$

where $\mathcal{R}_{\text{new}}$ denotes the segmentation risk of the newly added institution on its local medical image distribution.

Since the segmentation model parameters $\theta \in \mathbb{R}^D$ are typically high-dimensional, the newly added risk term can change the local geometry of the joint objective near the original optimum. In particular, directions associated with large-magnitude Hessian eigenvalues correspond to high local curvature, along which the objective is more sensitive to parameter changes. When a new client introduces a different data distribution, its risk term may change both the local gradients and curvature. This can shift the optimum in the high-dimensional weight space. As a result, dynamic client expansion may require many shared parameters to be updated, leading to costly global re-optimization.

3.2 Decoupled Modeling for Open-client Federated Segmentation

To avoid re-optimizing the entire segmentation model whenever a new medical institution joins, we decouple the *cross-institution shared segmentation representation* from *institution-specific distribution variations*. For medical image segmentation, each client c has a local private dataset $\mathcal{D}_c = \{(x_i^c, y_i^c)\}_{i=1}^{N_c}$, where $x_i^c \in \mathbb{R}^{H\times W\times d}$ denotes an input medical image and $y_i^c \in \{0,1\}^{H\times W\times L}$ denotes its pixel-level segmentation label, with L being the number of classes. In a standard segmentation model, a neural network with parameters θ is represented as $g_\theta(x) \in \mathbb{R}^{H\times W\times L}$, whose output gives the predicted logits for each pixel over different classes. To support the addition of new clients, we introduce a client-specific low-dimensional capability variable α_c and decompose the segmentation model into a shared representation and a client-specific perturbation:

$$f_{\theta,\phi,\alpha_c}(x) = g_\theta(x) + \Phi(x)\alpha_c, \tag{2}$$

where $g_\theta(x) \in \mathbb{R}^{H\times W\times L}$ denotes the logits produced by the shared backbone, and $\alpha_c \in \mathbb{R}^K$ denotes the client-specific low-dimensional capability coefficients. $\Phi(x) = \{\Phi_k(x)\}_{k=1}^K$ denotes the variation bases shared across clients, where each $\Phi_k(x) \in \mathbb{R}^{H\times W\times L}$ represents a structured variation pattern, such that $\Phi(x)\alpha_c = \sum_{k=1}^K \alpha_{c,k}\Phi_k(x)$. Thus, $\Phi(x)\alpha_c \in \mathbb{R}^{H\times W\times L}$ represents the

structured low-dimensional perturbation of client c relative to the shared representation. Specifically, $\Phi(x)$ is generated by a shared basis generator from shared image features, and its learning process is described in the next section. Finally, the segmentation prediction for client c is defined as $p_{\theta,\phi,\alpha_c}(x) = \sigma(f_{\theta,\phi,\alpha_c}(x))$, where $\sigma(\cdot)$ denotes the corresponding output activation function.

The overall framework of FedBSE is illustrated in Fig. 2. Through Decoupled Modeling for Open-client Federated Segmentation, the shared backbone captures common segmentation knowledge, while client-specific differences are represented in a low-dimensional capability space. During **Initial Capability Learning**, the shared components and client capability coefficients are jointly learned. After training, the shared components are frozen. When a new client arrives, FedBSE performs **Closed-Form Capability Inference** under a Gaussian-prior regularized local quadratic model, without updating the high-dimensional shared parameters. The inferred capability and its local uncertainty are then used to update the server-side capability distribution through Gaussian moment matching. The updated capability state, obtained through **Bayesian Capability Accumulation**, serves as the prior for subsequent clients. In this way, FedBSE supports continuous client expansion through low-dimensional capability inference and capability accumulation, while keeping the shared predictor and existing client-specific parameters unchanged.

3.3 Initial Capability Learning

In the initial stage, we learn the variation patterns shared across clients and the low-dimensional representation of each client over these patterns. To this end, we jointly learn the shared segmentation representation, shared variation bases, and client-specific capability coefficients. Specifically, let $z_\theta(x)$ denote the intermediate features extracted by the shared backbone. We introduce a shared variation basis generator $h_\phi(\cdot)$ parameterized by ϕ and define

$$\Phi(x) = h_\phi\big(z_\theta(x)\big) = \{\Phi_k(x)\}_{k=1}^{K}, \tag{3}$$

where $\Phi_k(x) \in \mathbb{R}^{H\times W\times L}$ represents the k-th structured variation basis. Together with Eq. 2, $\Phi(x)$ describes the structured variation directions shared across clients, while α_c specifies the combination weights of client c along these directions. Thus, different clients share the same variation bases, while their distributional differences are represented by low-dimensional capability coefficients.

Let $\mathcal{C}_t$ be the set of clients in the initial stage, where each client $c \in \mathcal{C}_t$ has a private dataset $\mathcal{D}_c$. During this stage, we jointly optimize the shared backbone parameters θ, the variation basis generator parameters ϕ, and the capability coefficients α_c of all clients. The objective function is defined as

$$\min_{\theta,\phi,\{\alpha_c\}} \sum_{c\in\mathcal{C}_t} \left[\widehat{\mathcal{R}}_c(\theta,\phi,\alpha_c) + \frac{\beta}{2}(\alpha_c-\mu)^\top\Sigma^{-1}(\alpha_c-\mu)\right], \tag{4}$$

where $\widehat{\mathcal{R}}_c(\theta,\phi,\alpha_c) = \frac{1}{N_c}\sum_{i=1}^{N_c}\mathcal{L}\big(p_{\theta,\phi,\alpha_c}(x_i^c), y_i^c\big)$ represents the empirical segmentation risk on client c, and $\mathcal{L}$ denotes the segmentation loss function. Here, $\mu \in \mathbb{R}^K$ and $\Sigma \in \mathbb{R}^{K\times K}$ denote the mean and covariance of the current capability distribution, respectively, and β is the regularization weight. To impose a statistical constraint on the client capability coefficients, we assume that they follow a shared Gaussian distribution, $\alpha_c \sim \mathcal{N}(\mu,\Sigma)$. Given μ and Σ, the corresponding quadratic form is $\frac{1}{2}(\alpha_c-\mu)^\top\Sigma^{-1}(\alpha_c-\mu)$. We incorporate this term into the training objective with the weighting factor β. Therefore, the objective above can be viewed as a Gaussian-prior regularized joint optimization of the shared model, variation bases, and client capability coefficients. This constraint encourages capability coefficients from different clients to share a common statistical structure rather than behave as independent free parameters. It also provides prior information for capability inference on new clients.

During initial training, μ and Σ are updated from the current statistics of the client capability coefficients and are used as the capability prior for subsequent optimization. After training, we use the learned $\{\alpha_c\}$ to estimate the initial capability distribution. The mean capability is defined as $\mu_t = \frac{1}{|\mathcal{C}_t|}\sum_{c\in\mathcal{C}_t}\alpha_c$, and the capability covariance is estimated as

$$\Sigma_t = \frac{1}{|\mathcal{C}_t|}\sum_{c\in\mathcal{C}_t}(\alpha_c-\mu_t)(\alpha_c-\mu_t)^\top + \epsilon I, \tag{5}$$

where ϵI is a small stabilization term that ensures that the covariance is non-singular for subsequent inverse operations. After the initial capability learning stage, we freeze the shared backbone parameters θ and the variation basis generator parameters ϕ, while retaining the capability distribution

state (μ_t, Σ_t). Here, μ_t represents the center of the client capability distribution, while Σ_t captures cross-client variation. When a new client joins, only its low-dimensional capability coefficients are inferred, while the shared backbone and variation basis generator remain fixed.

3.4 Closed-Form Capability Inference

In the previous subsection, we learned the shared backbone parameters θ, the variation basis generator parameters ϕ, and the capability distribution prior (μ_t, Σ_t) from the initial client set $\mathcal{C}_t$. Accordingly, the client capability variable is modeled by $\alpha \sim \mathcal{N}(\mu_t, \Sigma_t)$. When a new client n joins, we freeze the shared parameters θ and ϕ and infer only its capability coefficients α_n in the low-dimensional capability space. The capability coefficients of existing clients remain unchanged, thereby preserving their prediction functions and avoiding re-optimization of the high-dimensional shared parameters. For the new client n, the empirical risk is defined as

$$\widehat{\mathcal{R}}_n(\alpha) = \frac{1}{N_n} \sum_{i=1}^{N_n} \mathcal{L}\big(p_{\theta,\phi,\alpha}(x_i^n), y_i^n\big). \tag{6}$$

Given the Gaussian capability prior, the inference problem for the new client is formulated as a Gaussian-prior regularized objective:

$$\alpha_n^* = \arg\min_{\alpha} \left[\widehat{\mathcal{R}}_n(\alpha) + \frac{\beta}{2} (\alpha - \mu_t)^\top \Sigma_t^{-1} (\alpha - \mu_t) \right], \tag{7}$$

where the second term imposes the Gaussian prior structure, and β controls the strength of the prior constraint.

Since $\alpha \in \mathbb{R}^K$ and $K \ll D$, this optimization is much lower-dimensional than optimization in the shared model parameter space. To avoid multiple rounds of iterative optimization for α, we construct a local quadratic approximation of the empirical risk around the current prior mean μ_t. Let $g_n = \nabla_\alpha \widehat{\mathcal{R}}_n(\alpha)\big|_{\alpha=\mu_t}$ denote the gradient of the empirical risk at $\alpha = \mu_t$, where $g_n \in \mathbb{R}^K$. Further, let H_n denote a positive semi-definite local curvature approximation of $\widehat{\mathcal{R}}_n(\alpha)$ at $\alpha = \mu_t$, where $H_n \in \mathbb{R}^{K \times K}$. Then,

$$\widehat{\mathcal{R}}_n(\alpha) \approx \widehat{\mathcal{R}}_n(\mu_t) + g_n^\top (\alpha - \mu_t) + \frac{1}{2} (\alpha - \mu_t)^\top H_n (\alpha - \mu_t). \tag{8}$$

Substituting this approximation into the regularized objective yields a quadratic function of α. Taking its derivative and setting it to zero gives the closed-form solution

$$\alpha_n^* = \mu_t - \big(H_n + \beta \Sigma_t^{-1}\big)^{-1} g_n. \tag{9}$$

Here, H_n describes the local curvature of the new-client risk in the capability space, while Σ_t^{-1} describes the prior constraint along different capability directions. Since H_n is positive semi-definite and Σ_t is positive definite, $H_n + \beta \Sigma_t^{-1}$ is positive definite for $\beta > 0$. Thus, the local quadratic objective has a unique solution. The solution can be viewed as a single low-dimensional curvature-aware update from μ_t under the capability prior.

In practice, the gradient g_n can be obtained with one forward and backward pass. We use a positive semi-definite Gauss–Newton approximation for H_n and compute the capability update by solving a $K \times K$ linear system. The process only infers the low-dimensional capability variable and keeps the shared parameters θ and ϕ fixed. It therefore performs capability inference rather than model retraining. By reducing new-client adaptation to low-dimensional statistical inference, the framework can expand to new clients without updating the shared model.

3.5 Bayesian Capability Accumulation

In the previous subsection, a newly added client n obtained its capability coefficient $\alpha_n^\star$ through closed-form inference under the current Gaussian capability prior, while the shared parameters θ and ϕ remained fixed. This inference is based on the current capability state $\alpha \sim \mathcal{N}(\mu_t, \Sigma_t)$, where $\mu_t \in \mathbb{R}^K$ and $\Sigma_t \in \mathbb{R}^{K \times K}$ denote the mean and covariance of the current client capability distribution, respectively. However, $\alpha_n^\star$ only describes the capability estimate of the new client.

Without updating (μ_t, Σ_t), the information provided by this client cannot be used by subsequent clients. Therefore, we construct a local Gaussian approximation for the new client's capability and use it to update the global capability distribution. Based on the local quadratic approximation in the previous subsection, the local capability uncertainty is defined as $V_n = \left(H_n + \beta\Sigma_t^{-1}\right)^{-1}$. Combined with the closed-form estimate $\alpha_n^\star$, the local Gaussian capability approximation of the new client can be expressed as $q_n(\alpha) = \mathcal{N}(\alpha_n^\star, V_n)$. Here, V_n measures the local uncertainty of capability inference. When the new client's data provides strong information along a capability direction, the corresponding local curvature is high and the variance in V_n is small. Conversely, weak information leads to greater uncertainty. Thus, $(\alpha_n^\star, V_n)$ jointly describes the capability estimate and its local uncertainty for the new client.

Let M_t denote the number of clients accumulated by the system at time t. To incorporate the capability information of the new client into the global capability distribution, we use Gaussian moment matching to update its mean and covariance. The updated capability mean is

$$\mu_{t+1} = \frac{M_t\mu_t + \alpha_n^\star}{M_t + 1}. \tag{10}$$

The corresponding update for the capability covariance is

$$\Sigma_{t+1} = \frac{M_t}{M_t+1}\Sigma_t + \frac{1}{M_t+1}V_n + \frac{M_t}{(M_t+1)^2}\left(\alpha_n^\star - \mu_t\right)\left(\alpha_n^\star - \mu_t\right)^\top. \tag{11}$$

This update contains three components. The first term preserves the existing capability variation represented by the current distribution. The second term accounts for the local uncertainty of the new client's capability inference. The third term captures the deviation between the new client's capability mean and the current global mean. Thus, the updated capability distribution incorporates both the location of the new client in the capability space and the uncertainty of its local inference.

Unlike direct precision accumulation, Σ_t represents capability variation across clients rather than the estimation uncertainty of a single shared latent variable. Therefore, Σ_t does not need to decrease monotonically as new clients are added. The capability distribution may become more concentrated when a new client is similar to existing clients, while the variance along some capability directions may increase when the new client introduces new distributional differences. This update allows the system to continuously incorporate information from new clients without treating cross-client variation as uncertainty that must decrease over time. After the update, the server stores the new capability distribution state $(\mu_{t+1}, \Sigma_{t+1})$ and sets $M_{t+1} = M_t + 1$. The shared parameters θ and ϕ, together with the capability coefficients of existing clients, remain unchanged. When another new client joins, $(\mu_{t+1}, \Sigma_{t+1})$ serves as its new capability prior for closed-form inference. In this way, client onboarding forms a continuous loop of "capability inference–capability accumulation–prior update," allowing the system to expand to new clients without retraining the shared model.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate FedBSE on three medical image segmentation benchmarks: **EndoPolyp** (Polyp) for endoscopic polyp segmentation, **RIF** for retinal fundus optic disc and cup segmentation and **PMR** for T_2-weighted MRI prostate segmentation. EndoPolyp contains four public data sources with $\{1000, 380, 196, 612\}$ image-mask pairs, respectively, which are treated as four clients. Following the split in Dong et al. (2023), all images and masks are resized to 384×384. RIF contains four retinal fundus sites with $\{101, 159, 400, 400\}$ annotated images, respectively Wang et al. (2022). Following Liu et al. (2021), an 800×800 region containing the optic disc is cropped from each image and resized to 384×384, with the same processing applied to the corresponding mask. PMR contains six public prostate MRI data sources with $\{261, 384, 158, 468, 421, 175\}$ annotated 2D slices. We follow the preprocessing in Liu et al. (2020) and the site partitioning in Liu et al. (2021). All images are resampled and center-cropped to 384×384. Each PMR and RIF site is split into training and test sets at an $80\%/20\%$ ratio.

Open-client Protocol. In the leave-one-client-out evaluation, one site is treated as the newly arriving client, while the remaining sites form the initial federation. The held-out site is completely excluded from initial federated training. After deployment, methods that require adaptation can use

Table 1: New-client segmentation results (%) on Polyp, RIF and PMR under the leave-one-client-out setting. L0–L5 denote the held-out sites, and Avg. is the unweighted client average. Best results are shown in bold.

Dataset	Adaptation	Method	Dice							IoU							Onboarding
			L0	L1	L2	L3	L4	L5	Avg.	L0	L1	L2	L3	L4	L5	Avg.	Time (s)
Polyp	None	FedAvg	71.23	91.70	67.48	69.72	–	–	75.03	62.26	84.86	60.00	60.68	–	–	66.95	–
		FedDG	73.37	92.52	70.18	66.36	–	–	75.61	63.18	86.18	62.18	58.13	–	–	67.42	–
		FedEvi	65.99	93.56	73.27	74.04	–	–	76.72	55.57	88.75	66.57	64.94	–	–	68.96	–
	Local	FedBN	72.83	87.87	72.87	67.08	–	–	75.16	62.50	81.77	62.73	59.91	–	–	66.73	63.76
		FedAPA	**75.34**	94.29	71.29	72.26	–	–	78.29	**66.37**	89.05	65.24	63.36	–	–	71.01	35.47
		FedBCS	72.16	93.75	64.84	74.92	–	–	76.42	61.58	88.29	58.03	66.26	–	–	68.54	107.09
		FedBSE	73.88	**94.42**	**73.60**	**77.15**	–	–	**79.77**	64.86	**89.51**	**66.81**	**68.64**	–	–	**72.46**	**2.67**
RIF	None	FedAvg	79.05	78.18	87.26	79.41	–	–	80.98	68.25	66.47	78.19	68.93	–	–	70.46	–
		FedDG	84.79	78.90	87.96	87.32	–	–	84.74	75.74	67.43	79.29	79.30	–	–	75.44	–
		FedEvi	83.22	80.94	88.23	82.66	–	–	83.76	73.63	69.97	79.68	74.71	–	–	74.50	–
	Local	FedBN	72.75	71.21	88.37	89.17	–	–	80.38	61.23	60.83	80.81	81.34	–	–	71.05	61.55
		FedAPA	**86.82**	81.95	87.91	88.04	–	–	86.18	**77.66**	71.26	79.28	79.77	–	–	76.99	48.73
		FedBCS	80.49	76.01	**90.54**	**89.85**	–	–	84.22	70.81	67.98	**83.20**	**82.80**	–	–	76.19	133.67
		FedBSE	85.24	**83.83**	89.12	87.29	–	–	**86.37**	75.23	**73.69**	81.17	78.39	–	–	**77.12**	**2.42**
PMR	None	FedAvg	76.38	83.84	91.30	78.27	83.43	83.69	82.82	63.10	72.60	84.00	64.39	72.32	73.17	71.60	–
		FedDG	**86.86**	83.36	85.26	76.63	88.46	**86.35**	84.49	**76.96**	72.01	74.55	62.62	79.44	**76.12**	73.62	–
		FedEvi	78.32	84.48	91.04	78.70	86.69	83.66	83.81	65.28	73.61	84.27	64.96	76.66	72.15	72.82	–
	Local	FedBN	70.59	77.93	87.22	75.16	84.50	70.45	77.64	54.07	65.32	77.36	64.23	74.14	54.56	64.95	81.16
		FedAPA	84.70	83.29	91.31	80.84	89.79	85.03	85.83	73.69	74.04	84.01	69.46	81.17	74.39	76.13	54.14
		FedBCS	77.71	84.70	88.33	77.41	82.36	76.21	81.12	69.14	76.05	80.05	68.22	73.23	67.12	72.30	144.27
		FedBSE	83.85	**85.15**	**91.55**	**81.32**	**90.05**	85.83	**86.29**	72.06	**76.45**	**84.43**	**70.92**	**81.97**	75.34	**76.86**	**2.85**

its local annotated training data. Performance is evaluated on the corresponding test set. Each site is held out in turn. We report Dice and IoU for each site and their unweighted averages across clients. Sequential-arrival experiments are conducted separately to study the effect of capability accumulation over time.

Comparison Methods. The proposed method is compared with six post-deployment adaptation methods. FedAvg McMahan et al. (2017), FedDG Liu et al. (2021), and FedEvi Chen et al. (2024) perform no target-client adaptation and are directly evaluated on the newly arriving client. FedBN Li et al. (2021), FedAPA Sun et al. (2025), and FedBCS Zhao et al. (2026) use the new client's local annotations for adaptation. FedBN updates local model parameters and BN statistics for five epochs. FedAPA updates aggregation weights for five local epochs while keeping the source models fixed. FedBCS adapts the pre-arrival global model for five epochs and then merges it with the original global model using sample-size-based averaging. ReTrain performs five rounds of FedAvg retraining with both existing and newly arriving clients. These methods mainly differ in whether new-client annotations are used and which parameters are updated after deployment.

Evaluation Metrics. We use Dice Dice (1945) and IoU Taha & Hanbury (2015) to evaluate segmentation performance, and onboarding time to measure new-client adaptation efficiency. Onboarding time includes only the local adaptation computation after deployment, excluding initial training and inter-client communication. Higher Dice and IoU indicate better segmentation performance, while lower onboarding time is better.

4.2 Comparison with Federated Baseline Methods

Table 1 reports the segmentation results on the held-out clients. FedBSE achieves the highest average Dice and IoU on all three datasets and obtains the best performance on 8 of the 14 held-out clients. Among the methods using new-client annotations for adaptation, FedBSE also has the lowest average onboarding time. Performance varies across clients, and some baseline methods perform better on individual sites. Overall, FedBSE achieves strong average segmentation performance with low onboarding cost. Figure 3 shows representative segmentation results. Zero-adaptation methods rely only on models trained on the initial clients and do not use annotations from new clients. Local adaptation methods use these annotations through iterative model updates. FedBSE also uses labeled data from new clients, but limits adaptation to the low-dimensional capability space without updating the high-dimensional shared model.

4.3 Ablation Studies

We conduct ablation studies from five aspects: decoupled capability modeling, new-client inference, Bayesian capability accumulation, shared-model freezing, and sensitivity analysis. The last

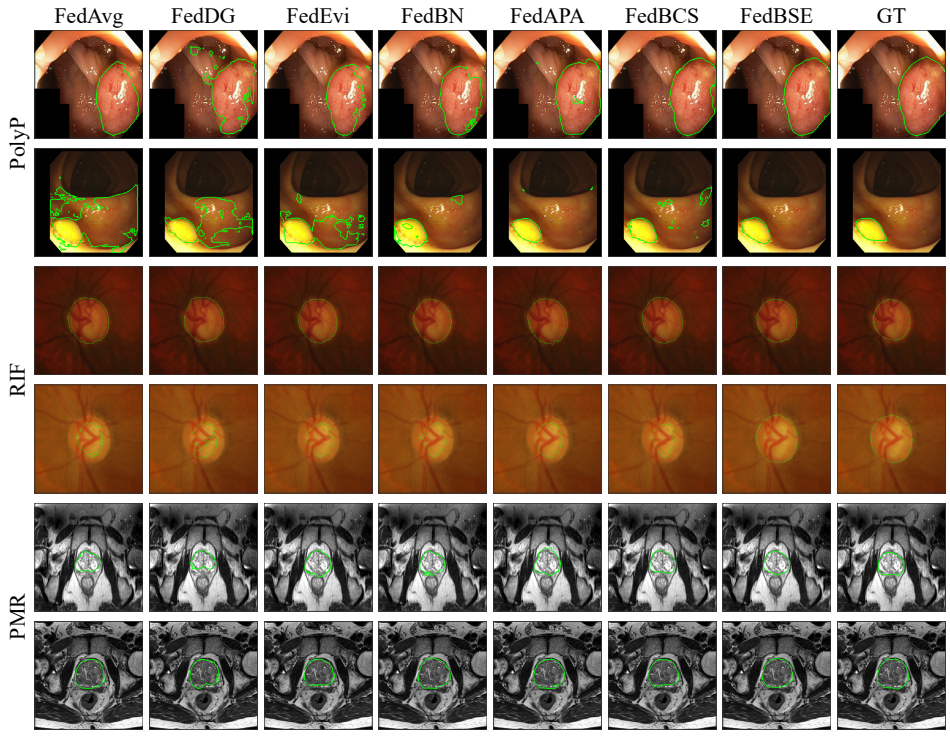


Figure 3: Qualitative segmentation results on Polyp (top), RIF (middle), and PMR (bottom). Green contours indicate foreground boundaries.

three are discussed in the appendix. For decoupled capability modeling, we evaluate the effects of the capability space, input-dependent variation basis, and capability prior. **w/o Capability Space** removes the client-specific term $\Phi(x)\alpha_c$ and retains only the shared model. **w/o Input-dependent Basis** replaces $\Phi(x)$ with an input-independent shared basis. **w/o Capability Prior** removes the regularization term $(\alpha_c - \mu)^\top \Sigma^{-1}(\alpha_c - \mu)$. For new-client inference, we compare the proposed closed-form inference with alternative adaptation strategies. **Gradient-based Adaptation** freezes the shared components and optimizes only the capability coefficients using gradient descent. **Full-model Fine-tuning** fine-tunes the full model on the new client. **First-order Inference** removes the curvature term by setting $H_n = 0$. **Diagonal Curvature** uses only the diagonal approximation of H_n. **Full Gauss–Newton** uses the full positive semi-definite Gauss–Newton approximation. We report Dice, IoU, and onboarding time to evaluate adaptation performance and efficiency. Table 2 shows the effectiveness of these key components.

5 Conclusion

This paper proposes FedBSE to address continuous client expansion after the deployment of federated medical image segmentation models. FedBSE decouples shared segmentation knowledge from client-specific variations. After initial federated training, the shared components are frozen. New clients are then adapted through closed-form inference in a low-dimensional capability space. The inferred capability information is further used to update the server-side capability state, providing a dynamic prior for subsequent clients. Experiments on Polyp, RIF and PMR show that FedBSE achieves the highest average segmentation performance among the compared methods with only a few seconds of onboarding time, while preserving existing client predictions. Sequential-arrival experiments further show that accumulated capability information supports subsequent client inference.

Table 2: Ablation study of FedBSE on Polyp.

Component	Variant	Polyp		
		Dice ↑	IoU ↑	Time (s) ↓
Capability Modeling	w/o Capability Space	75.03	66.95	–
	w/o Input-dependent Basis	78.06	69.61	–
	w/o Capability Prior	78.79	70.54	–
	Full FedBSE	**79.77**	**72.46**	–
New-Client Inference	Gradient-based Adaptation	78.36	71.02	50.76
	Full-model Fine-tuning	**81.55**	**74.74**	126.18
	First-order Inference	77.96	70.85	**2.35**
	Diagonal Curvature	79.18	71.92	2.49
	Full Gauss–Newton (FedBSE)	79.77	72.46	2.67

AI USE STATEMENT

In this work, we used large language models to improve the grammar and clarity of the manuscript and to assist with code revision. The authors designed the overall code structure and wrote the main implementation. These language models were not used to generate experimental data. All AI-assisted content was reviewed by the authors, who take full responsibility for the entire work.

ETHICS STATEMENT

This work uses publicly available datasets for federated medical image segmentation. We cite the original dataset sources and follow their stated terms of use. The study was conducted in accordance with the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

The main paper describes the proposed method, datasets, and evaluation metrics. The appendix provides detailed mathematical derivations and additional ablation results. We will release the code upon acceptance to support reproducibility.

REFERENCES

Ohad Amosy, Gal Eyal, and Gal Chechik. Late to the party? on-demand unlabeled personalized federated learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2184–2193, 2024.

Zhan-Lun Chang, Dong-Jun Han, Rohit Parasnis, Seyyedali Hosseinalipour, and Christopher Brinton. Federated learning with dynamic client arrival and departure: Convergence and rapid adaptation via initial model construction, 2024. arXiv preprint arXiv:2410.05662.

Haoxuan Che, Yifei Wu, Haibo Jin, Yong Xia, and Hao Chen. FedDAG: Federated domain adversarial generation toward generalizable medical image analysis. *IEEE Transactions on Medical Imaging*, 44(12):4931–4944, 2025.

Jiayi Chen, Benteng Ma, Hengfei Cui, and Yong Xia. FedEvi: Improving federated medical image segmentation via evidential weight aggregation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, volume 15010 of *Lecture Notes in Computer Science*, pp. 361–372, 2024.

Lee R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3): 297–302, 1945.

Bo Dong, Wenhai Wang, Deng-Ping Fan, Jinpeng Li, Huazhu Fu, and Ling Shao. Polyp-pvt: Polyp segmentation with pyramid vision transformers. *CAAI Artificial Intelligence Research*, 2:9150015, 2023.

Francesco Galati, Rosa Cortese, Ferran Prados, Marco Lorenzi, and Maria A. Zuluaga. Federated multi-centric image segmentation with uneven label distribution. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, volume 15010 of *Lecture Notes in Computer Science*, pp. 350–360, 2024.

Meirui Jiang, Hongzheng Yang, Chen Cheng, and Qi Dou. IOP-FL: Inside-outside personalization for federated medical image segmentation. *IEEE Transactions on Medical Imaging*, 42(7):2106–2117, 2023.

Georgios A. Kaissis, Marcus R. Makowski, Daniel Rückert, and Rickmer F. Braren. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 2 (6):305–311, 2020.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143, 2020.

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*, volume 2, pp. 429–450, 2020.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. FedBN: Federated learning on non-IID features via local batch normalization. In *International Conference on Learning Representations*, 2021.

Quande Liu, Qi Dou, Lequan Yu, and Pheng Ann Heng. Ms-net: multi-site network for improving prostate segmentation with heterogeneous mri data. *IEEE transactions on medical imaging*, 39 (9):2713–2724, 2020.

Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. FedDG: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1013–1023, 2021.

Mengmeng Ma, Tang Li, and Xi Peng. Beyond the federation: Topology-aware federated learning for generalization to unseen clients. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 33794–33810, 2024.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017.

Thinh Nguyen, Le Huy Khiem, Van-Tuan Tran, Khoa D. Doan, Nitesh V. Chawla, and Kok-Seng Wong. Onboarding without forgetting: Hypernetwork personalization with data-free replay for personalized federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Findings*, pp. 7728–7736, 2026.

Yihan Ren, Yanfeng Li, Jia Sun, Houjin Chen, Xiaochen Ma, and Bijuan Ren. FedCA: Federated domain generalization for medical image segmentation via cross-client feature style transfer and adaptive style alignment. *Expert Systems with Applications*, 311:131394, 2026.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

Micah J. Sheller, Brandon Edwards, G. Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R. Colen, and Spyridon Bakas. Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10(1):12598, 2020.

Yuxia Sun, Aoxiang Sun, Siyi Pan, Zhixiao Fu, and Jingcai Guo. FedAPA: Server-side gradient-based adaptive personalized aggregation for federated learning on heterogeneous data. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 6219–6226, 2025.

Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: Analysis, selection, and tool. *BMC Medical Imaging*, 15(1):29, 2015.

Donghao Wang, Yingchun Cui, Mingyang Li, Heran Xi, and Jinghua Zhu. FedFAT: Frequency adpative interpolation for federated domain generalization on heterogeneous medical images. *Pattern Recognition*, 172:112459, 2026.

Jiacheng Wang, Yueming Jin, and Liansheng Wang. Personalizing federated medical image segmentation via local calibration. In *European Conference on Computer Vision*, volume 13681, pp. 456–472, 2022.

Luyuan Xie, Manqing Lin, Siyuan Liu, ChenMing Xu, Tianyu Luan, Cong Li, Yuejian Fang, Qingni Shen, and Zhonghai Wu. pFLFE: Cross-silo personalized federated learning via feature enhancement on medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, volume 15010 of *Lecture Notes in Computer Science*, pp. 599–610, 2024.

An Xu, Wenqi Li, Pengfei Guo, Dong Yang, Holger R. Roth, Ali Hatamizadeh, Can Zhao, Daguang Xu, Heng Huang, and Ziyue Xu. Closing the generalization gap of cross-silo federated medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20866–20875, 2022.

Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization. In *International Conference on Learning Representations*, 2021.

Xingyue Zhao, Wenke Huang, Xingguang Wang, Haoyu Zhao, Linghao Zhuang, Anwen Jiang, Guancheng Wan, and Mang Ye. Divide, conquer and unite: Hierarchical style-recalibrated prototype alignment for federated medical segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28760–28768, 2026.

A Appendix

A.1 Theoretical Foundations

This section analyzes the fundamental theoretical properties of the proposed open-client federated segmentation framework. Unless otherwise specified, $\|\cdot\|_2$ denotes the Euclidean norm, $\|v\|_A^2 \triangleq v^\top Av$, and $\preceq$ denotes the Loewner partial order.

Theorem 1 (Local Sensitivity of the Weight-space Optimum to Client Expansion). Let the joint risk at time t be

$$F_t(\theta) \triangleq \sum_{c\in\mathcal{C}_t} \mathcal{R}_c(\theta), \tag{12}$$

and let θ_t^* be an interior local optimum. Assume that F_t is twice continuously differentiable in a neighborhood of θ_t^* and satisfies

$$\nabla^2 F_t(\theta_t^*) \succeq mI, \qquad m > 0. \tag{13}$$

Consider the perturbed objective induced by a newly arriving client:

$$F_t^\varepsilon(\theta) = F_t(\theta) + \varepsilon\mathcal{R}_n(\theta), \tag{14}$$

where $\varepsilon > 0$ denotes the perturbation strength. Assume that $\mathcal{R}_n(\theta)$ is twice continuously differentiable in a neighborhood of θ_t^*. Then, for sufficiently small ε, there exists a local optimum $\theta_{t,\varepsilon}^*$ near θ_t^* such that

$$\theta_{t,\varepsilon}^* - \theta_t^* = -\varepsilon\left[\nabla^2 F_t(\theta_t^*)\right]^{-1}\nabla\mathcal{R}_n(\theta_t^*) + o(\varepsilon). \tag{15}$$

Furthermore,

$$\|\theta_{t,\varepsilon}^* - \theta_t^*\|_2 \le \frac{\varepsilon}{m}\|\nabla\mathcal{R}_n(\theta_t^*)\|_2 + o(\varepsilon). \tag{16}$$

Proof. Define

$$G(\theta,\varepsilon) = \nabla F_t(\theta) + \varepsilon\nabla\mathcal{R}_n(\theta). \tag{17}$$

Since θ_t^* is an interior local optimum of F_t,

$$G(\theta_t^*, 0) = \nabla F_t(\theta_t^*) = 0. \tag{18}$$

Moreover,

$$\nabla_\theta G(\theta_t^*, 0) = \nabla^2 F_t(\theta_t^*) \succ 0, \tag{19}$$

and is therefore nonsingular. By the implicit function theorem, there exists a differentiable stationary-point branch $\theta^*(\varepsilon)$ around $\varepsilon = 0$ satisfying

$$G(\theta^*(\varepsilon), \varepsilon) = 0, \qquad \theta^*(0) = \theta_t^*. \tag{20}$$

Since the Hessian of the perturbed objective is continuous in (θ, ε) and $\nabla^2 F_t(\theta_t^*) \succ 0$, it remains positive definite for sufficiently small ε. Thus, $\theta^*(\varepsilon)$ is a local optimum, denoted by $\theta_{t,\varepsilon}^*$.

Differentiating $G(\theta^*(\varepsilon), \varepsilon) = 0$ with respect to ε at $\varepsilon = 0$ gives

$$\nabla^2 F_t(\theta_t^*) \frac{d\theta^*(\varepsilon)}{d\varepsilon}\bigg|_{\varepsilon=0} + \nabla \mathcal{R}_n(\theta_t^*) = 0. \tag{21}$$

Hence,

$$\frac{d\theta^*(\varepsilon)}{d\varepsilon}\bigg|_{\varepsilon=0} = -\left[\nabla^2 F_t(\theta_t^*)\right]^{-1} \nabla \mathcal{R}_n(\theta_t^*). \tag{22}$$

Using the first-order expansion of $\theta^*(\varepsilon)$ around $\varepsilon = 0$ yields

$$\theta_{t,\varepsilon}^* - \theta_t^* = -\varepsilon \left[\nabla^2 F_t(\theta_t^*)\right]^{-1} \nabla \mathcal{R}_n(\theta_t^*) + o(\varepsilon). \tag{23}$$

Since $\nabla^2 F_t(\theta_t^*) \succeq mI$, we have

$$\left\| \left[\nabla^2 F_t(\theta_t^*)\right]^{-1} \right\|_2 \leq \frac{1}{m}. \tag{24}$$

The stated sensitivity bound then follows from the operator norm inequality. □

This result shows that the risk introduced by a new client can shift the optimum of the high-dimensional shared parameters. When the local curvature is weak or the new client introduces a large gradient perturbation, the shared optimum can become more sensitive to client expansion. This theorem provides theoretical motivation for freezing the shared components in our framework. Re-optimizing the shared model after each client arrival may repeatedly shift the optimum, increasing optimization costs and potentially affecting the predictions of existing clients.

Theorem 2 (Local Closed-form Capability Inference). For client c, let

$$g_c = \nabla_\alpha \widehat{\mathcal{R}}_c(\alpha)\big|_{\alpha=\mu}, \tag{25}$$

and consider the following local quadratic surrogate around the capability prior mean μ:

$$\widetilde{\mathcal{R}}_c(\alpha) = \widehat{\mathcal{R}}_c(\mu) + g_c^\top \Delta + \frac{1}{2}\Delta^\top H_c \Delta, \tag{26}$$

where

$$\Delta = \alpha - \mu, \qquad H_c \succeq 0. \tag{27}$$

Here, H_c denotes a positive semi-definite local curvature approximation. Given the Gaussian capability prior $\alpha \sim \mathcal{N}(\mu, \Sigma)$, where $\Sigma \succ 0$, consider the Gaussian-prior regularized objective

$$J_c(\alpha; \mu, \Sigma) = \widehat{\mathcal{R}}_c(\alpha) + \frac{\beta}{2}\|\alpha - \mu\|_{\Sigma^{-1}}^2, \tag{28}$$

where $\beta > 0$ controls the strength of the prior regularization. If client-specific variations can be represented by the shared variation basis as $\Phi(x)\alpha_c$, while the shared parameters θ and ϕ remain frozen, then client adaptation can be restricted to the low-dimensional variable α_c. Under the local quadratic surrogate, the unique minimizer is

$$\alpha_c^* = \mu - \left(H_c + \beta\Sigma^{-1}\right)^{-1} g_c. \tag{29}$$

Proof. Replacing the empirical risk in the regularized objective with its local quadratic surrogate and ignoring terms independent of α gives

$$J_c(\alpha; \mu, \Sigma) \approx g_c^\top \Delta + \frac{1}{2}\Delta^\top \left(H_c + \beta\Sigma^{-1}\right)\Delta. \tag{30}$$

Since $H_c \succeq 0$, $\Sigma \succ 0$, and $\beta > 0$, we have

$$H_c + \beta\Sigma^{-1} \succ 0. \tag{31}$$

Therefore, the local quadratic objective has a unique minimizer. Setting its gradient with respect to Δ to zero gives

$$g_c + \left(H_c + \beta\Sigma^{-1}\right)\Delta = 0. \tag{32}$$

Thus,

$$\Delta = -\left(H_c + \beta\Sigma^{-1}\right)^{-1} g_c. \tag{33}$$

Substituting $\Delta = \alpha_c^* - \mu$ gives the stated closed-form solution. □

This theorem shows that, when the shared variation basis can represent client-specific differences, new-client adaptation can be restricted to a low-dimensional capability space without modifying the high-dimensional shared parameters. It directly supports the proposed Closed-Form Capability Inference module: a new client only needs to infer its low-dimensional capability variable, avoiding iterative retraining of the full segmentation network.

Theorem 3 (Moment-Preserving Capability Accumulation). Suppose the system has accumulated M_t clients by time t, and let the current capability distribution be approximated by

$$Q_t = \mathcal{N}(\mu_t, \Sigma_t), \tag{34}$$

where $\Sigma_t \succ 0$. For a newly arriving client n, let its local Gaussian capability approximation be

$$q_n(\alpha) = \mathcal{N}\left(\alpha_n^\star, V_n\right), \tag{35}$$

where

$$V_n = \left(H_n + \beta\Sigma_t^{-1}\right)^{-1} \succ 0. \tag{36}$$

Consider the mixture distribution

$$\widetilde{Q}_{t+1} = \frac{M_t}{M_t+1} Q_t + \frac{1}{M_t+1} q_n. \tag{37}$$

If Gaussian moment matching is used to approximate $\widetilde{Q}_{t+1}$ by

$$Q_{t+1} = \mathcal{N}(\mu_{t+1}, \Sigma_{t+1}) \tag{38}$$

with the same first and second central moments, then

$$\mu_{t+1} = \frac{M_t\mu_t + \alpha_n^\star}{M_t+1}, \tag{39}$$

and

$$\Sigma_{t+1} = \frac{M_t}{M_t+1}\Sigma_t + \frac{1}{M_t+1}V_n + \frac{M_t}{(M_t+1)^2}\left(\alpha_n^\star - \mu_t\right)\left(\alpha_n^\star - \mu_t\right)^\top. \tag{40}$$

Furthermore, $\Sigma_{t+1} \succ 0$.

Proof. By the definition of the mean of a mixture distribution,

$$\mathbb{E}_{\widetilde{Q}_{t+1}}[\alpha] = \frac{M_t}{M_t+1}\mu_t + \frac{1}{M_t+1}\alpha_n^\star = \frac{M_t\mu_t + \alpha_n^\star}{M_t+1} = \mu_{t+1}, \tag{41}$$

Using the covariance decomposition for a mixture distribution,

$$\mathrm{Cov}_{\widetilde{Q}_{t+1}}(\alpha) = \frac{M_t}{M_t+1}\Sigma_t + \frac{1}{M_t+1}V_n + \frac{M_t}{(M_t+1)^2}(\alpha_n^\star - \mu_t)(\alpha_n^\star - \mu_t)^\top. \tag{42}$$

Therefore, Gaussian moment matching gives

$$\Sigma_{t+1} = \mathrm{Cov}_{\widetilde{Q}_{t+1}}(\alpha). \tag{43}$$

Since $\Sigma_t \succ 0$, $V_n \succ 0$, and

$$(\alpha_n^\star - \mu_t)(\alpha_n^\star - \mu_t)^\top \succeq 0, \tag{44}$$

the first two terms are positive definite and the last term is positive semi-definite. Since their weights are positive, $\Sigma_{t+1} \succ 0$. □

This conclusion indicates that Bayesian Capability Accumulation is not a simple precision accumulation process. Gaussian moment matching preserves the capability variation represented by the existing distribution, the local uncertainty of the new client, and the shift of the new client from the

current capability mean. Therefore, Σ_{t+1} is not required to decrease monotonically as new clients arrive. The theorem guarantees that the first two moments of the mixture are preserved by the Gaussian approximation. It also ensures that $(\mu_{t+1}, \Sigma_{t+1})$ defines a proper Gaussian distribution that can serve as the prior for subsequent client inference.

Theorem 4 (Asymptotic Consistency of Capability Moments). Let the initial stage contain a fixed number M_0 of clients, with capability statistics (μ_0, Σ_0). Let the capabilities of subsequently arriving clients, $\{\alpha_n\}_{n=1}^{\infty}$, be independent and identically distributed according to $\mathbb{P}$, with a finite second moment: $\mathbb{E}_{\mathbb{P}}\left[\|\alpha\|_2^2\right] < \infty$. Suppose that closed-form capability inference gives $\alpha_n^\star = \alpha_n + e_n$, where the average squared inference error satisfies

$$\frac{1}{T}\sum_{n=1}^{T} \|e_n\|_2^2 \to 0 \qquad \text{a.s.} \tag{45}$$

Further assume that the average local Gaussian uncertainty vanishes:

$$\left\| \frac{1}{T}\sum_{n=1}^{T} V_n \right\|_F \to 0 \qquad \text{a.s.} \tag{46}$$

If the server continuously applies the moment-matching update in Theorem 3, then, after T new clients have arrived and as $T \to \infty$,

$$\mu_T \to \mathbb{E}_{\mathbb{P}}[\alpha] \qquad \text{a.s.,} \;\; \Sigma_T \to \mathrm{Cov}_{\mathbb{P}}(\alpha) \qquad \text{a.s.} \tag{47}$$

Proof. After T new clients have arrived, repeated application of the mean update in Theorem 3 gives

$$\mu_T = \frac{M_0\mu_0 + \sum_{n=1}^{T} \alpha_n^\star}{M_0 + T}. \tag{48}$$

Substituting $\alpha_n^\star = \alpha_n + e_n$ gives

$$\mu_T = \frac{M_0\mu_0}{M_0 + T} + \frac{T}{M_0 + T}\left(\frac{1}{T}\sum_{n=1}^{T}\alpha_n + \frac{1}{T}\sum_{n=1}^{T} e_n\right). \tag{49}$$

As $T \to \infty$,

$$\frac{M_0\mu_0}{M_0 + T} \to 0. \tag{50}$$

By the strong law of large numbers,

$$\frac{1}{T}\sum_{n=1}^{T}\alpha_n \to \mathbb{E}_{\mathbb{P}}[\alpha] \qquad \text{a.s.} \tag{51}$$

Moreover, by the Cauchy–Schwarz inequality,

$$\left\| \frac{1}{T}\sum_{n=1}^{T} e_n \right\|_2 \le \sqrt{\frac{1}{T}\sum_{n=1}^{T}\|e_n\|_2^2} \to 0 \qquad \text{a.s.} \tag{52}$$

Therefore, $\mu_T \to \mathbb{E}_{\mathbb{P}}[\alpha]$ a.s.

For the covariance, define the second raw moment

$$S_T \triangleq \Sigma_T + \mu_T\mu_T^\top. \tag{53}$$

Since the update in Theorem 3 preserves the first and second moments, repeated moment matching gives

$$S_T = \frac{M_0\left(\Sigma_0 + \mu_0\mu_0^\top\right) + \sum_{n=1}^{T}\left(V_n + \alpha_n^\star{\alpha_n^\star}^\top\right)}{M_0 + T}. \tag{54}$$

The contribution of the fixed initial client set vanishes as $T \to \infty$. Moreover,

$$\alpha_n^\star{\alpha_n^\star}^\top = \alpha_n\alpha_n^\top + \alpha_n e_n^\top + e_n\alpha_n^\top + e_n e_n^\top. \tag{55}$$

Using the Cauchy–Schwarz inequality,

$$\left\| \frac{1}{T} \sum_{n=1}^{T} \alpha_n e_n^\top \right\|_F \le \sqrt{\frac{1}{T} \sum_{n=1}^{T} \|\alpha_n\|_2^2} \sqrt{\frac{1}{T} \sum_{n=1}^{T} \|e_n\|_2^2} \to 0 \qquad \text{a.s.} \tag{56}$$

Similarly,

$$\left\| \frac{1}{T} \sum_{n=1}^{T} e_n \alpha_n^\top \right\|_F \to 0 \qquad \text{a.s.} \tag{57}$$

Also,

$$\left\| \frac{1}{T} \sum_{n=1}^{T} e_n e_n^\top \right\|_F \le \frac{1}{T} \sum_{n=1}^{T} \|e_n\|_2^2 \to 0 \qquad \text{a.s.} \tag{58}$$

Together with

$$\left\| \frac{1}{T} \sum_{n=1}^{T} V_n \right\|_F \to 0 \qquad \text{a.s.,} \tag{59}$$

and the strong law of large numbers

$$\frac{1}{T} \sum_{n=1}^{T} \alpha_n \alpha_n^\top \to \mathbb{E}_{\mathbb{P}} \left[\alpha \alpha^\top \right] \qquad \text{a.s.,} \tag{60}$$

we obtain

$$S_T \to \mathbb{E}_{\mathbb{P}} \left[\alpha \alpha^\top \right] \qquad \text{a.s.} \tag{61}$$

Finally, since

$$\Sigma_T = S_T - \mu_T \mu_T^\top, \tag{62}$$

it follows that

$$\Sigma_T \to \mathbb{E}_{\mathbb{P}} \left[\alpha \alpha^\top \right] - \mathbb{E}_{\mathbb{P}}[\alpha] \mathbb{E}_{\mathbb{P}}[\alpha]^\top = \mathrm{Cov}_{\mathbb{P}}(\alpha) \qquad \text{a.s.} \tag{63}$$

□

This theorem shows that, under vanishing average inference error and local Gaussian uncertainty, the online capability distribution asymptotically recovers the first- and second-order statistics of the underlying client capability population. It therefore provides a conditional long-term consistency guarantee for capability distribution evolution under continuous client arrival. As more clients join and the inference uncertainty decreases, the server-maintained prior can progressively approach the underlying cross-client statistical structure.

Overall, Theorem 1 provides theoretical motivation for freezing the shared components by showing that new-client perturbations can shift the optimum of the high-dimensional shared parameters. Theorem 2 supports Closed-Form Capability Inference by showing that new-client adaptation can be reduced to a low-dimensional closed-form solution under a local quadratic approximation. Theorem 3 supports Bayesian Capability Accumulation by showing that Gaussian moment matching preserves the first two moments of the accumulated capability distribution. Theorem 4 further establishes the asymptotic consistency of the evolving capability moments under vanishing inference error and uncertainty. Together, these results provide theoretical support for the proposed open-client framework, covering shared-parameter freezing, low-dimensional client adaptation, and continuous server-side capability accumulation.

A.2 Practical Protocol for Experiments

To ensure consistency between the experimental protocol and the actual implementation, the main experiments use a leave-one-client-out setting. Sequential client arrivals are evaluated separately. In each leave-one-client-out task, one site is held out as a newly arriving client, while the remaining sites form the initial client set $\mathcal{C}_t$. Data from the held-out site is completely unavailable during the initial federated training stage.

For each client $c \in \mathcal{C}_t$, its local data $\mathcal{D}_c = \{(x_i^c, y_i^c)\}$ remains private. Client capability coefficients are initialized to zero. A non-singular capability prior is obtained using covariance stabilization. At

the beginning of each communication round, the server estimates (μ, Σ) from the current capability coefficients of all initial clients. This prior remains fixed during local updates. The server broadcasts the shared backbone parameters, the variation basis generator parameters, and the current capability prior (μ, Σ) to all initial clients. Each client uses Adam to locally minimize

$$\widehat{\mathcal{R}}_c(\theta, \phi, \alpha_c) + \frac{\beta}{2}(\alpha_c - \mu)^\top \Sigma^{-1}(\alpha_c - \mu), \tag{64}$$

where $\widehat{\mathcal{R}}_c(\theta, \phi, \alpha_c)$ denotes the empirical segmentation risk of client c. Each client updates the shared backbone parameters, the variation basis generator parameters, and its own capability coefficients. The server performs equal-weight averaging only on the uploaded shared parameters. The capability coefficients are not aggregated as shared model parameters. Instead, they remain client-specific and are updated across communication rounds. At the end of each round, clients send their current low-dimensional capability coefficients to the server. The server then estimates (μ, Σ) for the next round. After the initial federated training stage, the shared backbone and variation basis generator are frozen. The initial capability state (μ_t, Σ_t) is then estimated from $\{\alpha_c\}_{c \in \mathcal{C}_t}$.

Upon deployment, the held-out site joins the system as a new client n. It receives the frozen shared components and the current capability state (μ_t, Σ_t), but no data from other clients. Its training partition is used as a local labeled support set. This set is used to compute the empirical risk, the gradient g_n, and the positive semi-definite curvature approximation H_n. The test partition is used only for final evaluation. The client obtains its capability coefficients by solving a $K \times K$ linear system:

$$\alpha_n^* = \mu_t - \left(H_n + \beta \Sigma_t^{-1}\right)^{-1} g_n, \tag{65}$$

and computes the corresponding local covariance:

$$V_n = \left(H_n + \beta \Sigma_t^{-1}\right)^{-1}. \tag{66}$$

The new client uploads only (α_n^*, V_n) to the server. Raw data and gradients are not transmitted. Given the current number of accumulated clients M_t, the server updates $(\mu_{t+1}, \Sigma_{t+1})$ using Gaussian moment matching. For numerical stability, the statistical covariance Σ_t is kept unchanged, while a stabilized covariance $\Sigma_t + \epsilon I$ is used when matrix inversion is required. The updated capability state is then used as the prior for subsequent new clients. The default protocol does not modify the shared parameters or the capability coefficients of existing clients. Therefore, their prediction functions remain unchanged after a new client joins.

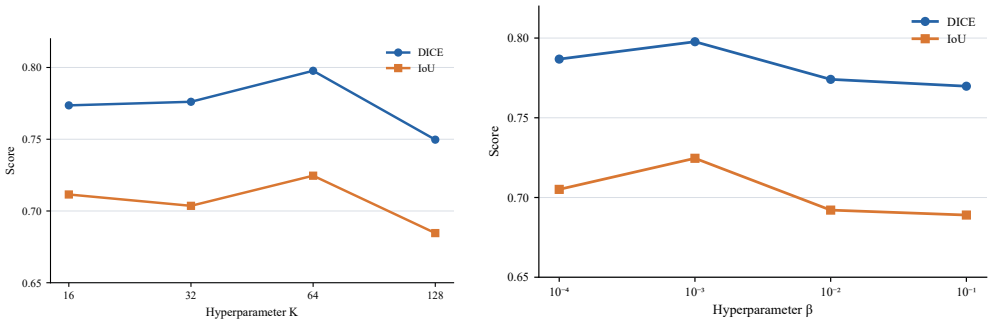
In implementation, g_n is obtained from a forward and backward pass. The curvature matrix H_n is approximated using a positive semi-definite Gauss–Newton approximation. The resulting $K \times K$ linear system is solved using Cholesky decomposition rather than an explicit matrix inverse. A diagonal curvature approximation can also be used when lower computational cost is required. The onboarding stage does not update the shared components and does not require multi-round optimization of the capability coefficients. Sequential-arrival experiments are conducted separately from the main leave-one-client-out experiments to study the effect of capability state accumulation. Both the fixed-prior and cumulative-prior branches use the same initial checkpoint, arrival sequence, and local data. The fixed-prior branch always uses the initial capability state for each new client. In contrast, the cumulative-prior branch uses the state updated after the previous client is integrated. The two branches are identical for the first arriving client and differ only from the second client onward. The main experiments report Dice, IoU, and client-wise macro-averages for each held-out site. We also report the local onboarding time after deployment. Since the default protocol does not update the shared parameters or the capability coefficients of existing clients, we further verify that their predictive performance remains unchanged after new-client integration. The sequential-arrival experiment compares the fixed-prior and cumulative-prior branches under a preset arrival order. These results are not interpreted as average cumulative gains across different random seeds or arrival orders.

A.3 Implementation Details.

For Polyp, the initial training uses 150 communication rounds with three local epochs per round. For PMR, we use 400 communication rounds with one local epoch per round. In the main experiments, the batch size is 6, the initial learning rate is 10^{-3}, the capability dimension is $K = 64$, and $\beta = 10^{-3}$. All experiments are conducted on NVIDIA RTX 4090 GPUs.

Table 4: Ablation of shared-model freezing during new-client onboarding.

Variant	New-client Dice ↑	Existing-client Change ↑	Time (s) ↓
Update Backbone	80.76	-5.93	44.19
Update Basis Generator	80.91	-0.88	24.40
Update All Shared Components	**81.49**	-5.75	47.12
Freeze Shared Model (FedBSE)	79.77	**0.00**	**2.67**



(a) Effect of Capability Dimension K

(b) Effect of Prior Weight β

Figure 4: Sensitivity analysis of FedBSE with respect to the capability dimension K and prior weight β.

A.4 Supplementary Ablation Studies

Bayesian Capability Accumulation. This ablation is conducted under the sequential-arrival setting. **Fixed Prior** always uses the initial capability state (μ_0, Σ_0). **Mean Update Only** updates μ_t while keeping Σ_t fixed. **w/o Uncertainty** removes the local uncertainty term V_n. **w/o Mean-shift Term** removes $\frac{M_t}{(M_t+1)^2}(\alpha_n^\star - \mu_t)(\alpha_n^\star - \mu_t)^\top$. **Full Moment Matching** uses the complete capability-state update.

As shown in Table 3, using a fixed prior gives the lowest Dice and IoU of 65.77% and 53.91%, respectively. Updating only the capability mean improves them to 70.19% and 60.10%. Further modeling the covariance brings additional gains. Removing either the uncertainty term or the mean-shift term reduces performance compared with full moment matching. The full update achieves the best Dice of 72.90% and IoU of 64.32%, showing that both local uncertainty and cross-client capability variation contribute to capability accumulation.

Table 3: Ablation of Bayesian capability accumulation under sequential client arrival.

Variant	Dice ↑	IoU ↑
Fixed Prior	65.77	53.91
Mean Update Only	70.19	60.10
w/o Uncertainty	72.65	63.91
w/o Mean-shift Term	72.81	64.15
Full Moment Matching	**72.90**	**64.32**

Shared-Model Freezing. We further evaluate the effect of freezing the shared components after deployment. **Update Backbone** updates the shared backbone after a new client arrives. **Update Basis Generator** updates only the variation-basis generator. **Update All Shared Components** updates both shared components. The default FedBSE keeps all shared components fixed. We report new-client performance, existing-client performance change, and onboarding time.

Table 4 shows that updating shared components can improve new-client accuracy, but at the cost of higher onboarding time and reduced stability on existing clients. Updating all shared components gives the highest new-client Dice of 81.49%, but decreases existing-client performance by 5.75% and requires 47.12 seconds. In contrast, FedBSE achieves 79.77% Dice with only 2.67 seconds of onboarding time, while keeping existing-client performance unchanged. These results support freezing the shared components when both onboarding efficiency and existing-client stability are important.

Sensitivity Analysis. We study the sensitivity of FedBSE to the capability dimension $K \in \{16, 32, 64, 128\}$ and the prior weight $\beta \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. The results shown in Figure

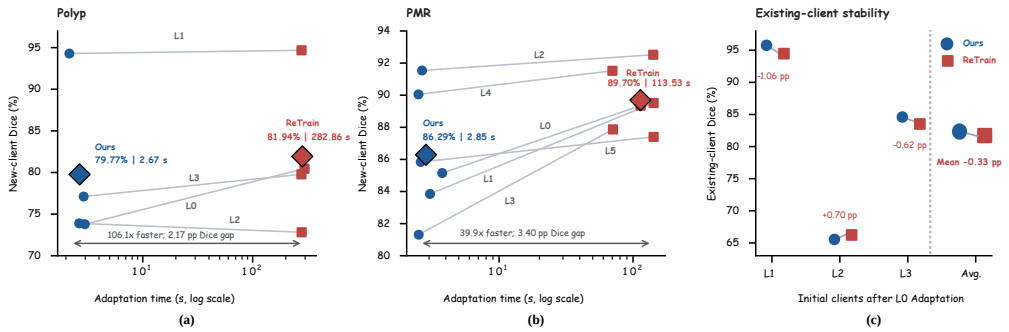


Figure 5: Accuracy, onboarding efficiency, and stability of existing clients. **a–b**: Comparison between FedBSE and 5-round ReTrain on the held-out Polyp and PMR clients. Onboarding time is shown on a logarithmic scale, and diamonds denote the client averages. **c**: Dice scores of existing clients before and after the arrival of Polyp L0.

4 demonstrate that FedBSE remains stable across a broad range of K and β. A moderate capability dimension provides a good balance between representation capacity and model complexity, while an appropriate prior weight helps regularize client-specific capability inference. Performance degrades when K is too small or when β is set too large or too small, indicating that both the capability-space size and prior strength affect adaptation quality.

A.5 Analysis of Accuracy, Onboarding Efficiency, and Stability of Existing Clients

The main comparison results show that the proposed method can onboard new clients within a few seconds. To further examine the trade-off between accuracy and onboarding efficiency, Fig. 5(a)–(b) compares the proposed method with ReTrain on the same held-out clients. When a new client arrives, ReTrain requires all initial clients to come back online and resumes federated training. The shared model is then updated using data from both existing and new clients. Both methods use the same initial model and held-out client, and differ only in how the new client is handled after arrival.

The results show that resuming federated training can further improve new-client accuracy, but requires existing clients to come back online and the shared model to be optimized again. In contrast, the proposed method achieves comparable segmentation performance with much lower onboarding time. This makes it more suitable when existing clients are difficult to reconnect or new clients continuously join the federation.

Besides onboarding time, retraining may also affect existing-client predictions. Fig. 5(c) compares the performance of the initial clients before and after a new client joins. The proposed method does not update the shared backbone, variation basis generator, or capability coefficients of existing clients during onboarding. Therefore, their prediction functions remain unchanged. In contrast, ReTrain updates the shared model and causes different levels of performance drift across the initial clients, with a slight overall decrease. This indicates that shared-model updates for new clients may affect the prediction stability of existing clients. By keeping existing prediction paths fixed, the proposed method reduces onboarding cost while avoiding direct changes to existing-client prediction functions.

A.6 Analysis of Sequential Arrival and Capability Accumulation.

The previous experiments treat each new client arrival as an independent event. In open federated systems, however, new clients may arrive sequentially over time. To examine whether capability information from earlier clients can support later client inference, we construct two branches: a *fixed-state* branch and an *accumulated-state* branch, as shown in Fig. 6. Both branches use the same initial checkpoint, client arrival order, and local data. The shared backbone and variation basis generator are frozen in both cases.

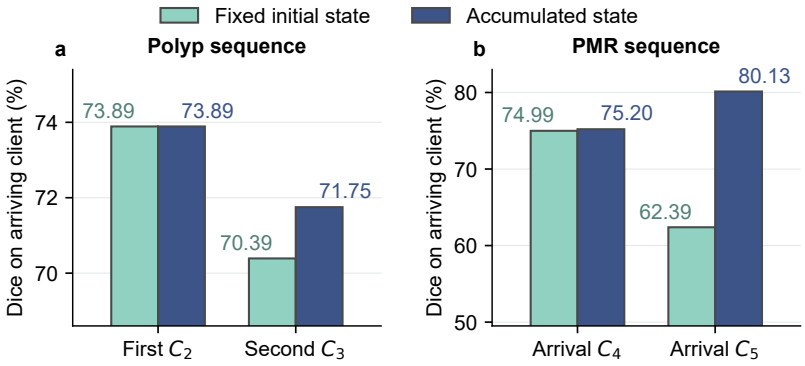


Figure 6: Sequential client arrival under fixed and accumulated capability states. **a**: Polyp with the arrival sequence $C_2 \rightarrow C_3$. **b**: PMR with the arrival sequence $C_4 \rightarrow C_5$.

The fixed-state branch always uses the initial state (μ_0, Σ_0) for each new client. In contrast, the accumulated-state branch uses the updated state (μ_t, Σ_t) from the previous step as the prior for the next client. Since both branches start from the same capability state, their results are identical for the first arriving client. From the second client onward, the accumulated-state branch can use capability information from previously arrived clients.

The results show that capability accumulation affects the inference of later clients, although the effect varies across clients. This indicates that information from earlier clients can be transferred to subsequent inference through the updated capability state.