Topics on the Edge

Pushing the fundamental limits of federated learning on the mobile edge

- Federated Learning and Mobile Edge Computing
- Hierarchical FL
- Compute-aware Client Selection

Federated Learning and the Mobile Edge

The Mobile Edge

"Enduring Challenges" of Pervasive Computing

- Resource poverty
- Communication Uncertainty
- Finite Energy
- *Multi-modal interaction*
- *Scarce user attention*
- Lower privacy, security, robustness

Mahadev Satyanarayanan - *Mobile and Pervasive Computing* (15-821/18-843, Every Fall, including Fall 2022)

Federated Learning: Natural Advantages

"Enduring Challenges" of Pervasive Computing

- Resource poverty
- Communication Uncertainty
- Finite Energy
- *Multi-modal interaction*
- *Scarce user attention*
- Lower privacy, security, robustness

Client Selection

Federated Training

Mahadev Satyanarayanan - *Mobile and Pervasive Computing* (15-821/18-843, Every Fall, including Fall 2022)

Federated Learning: Natural Disadvantages

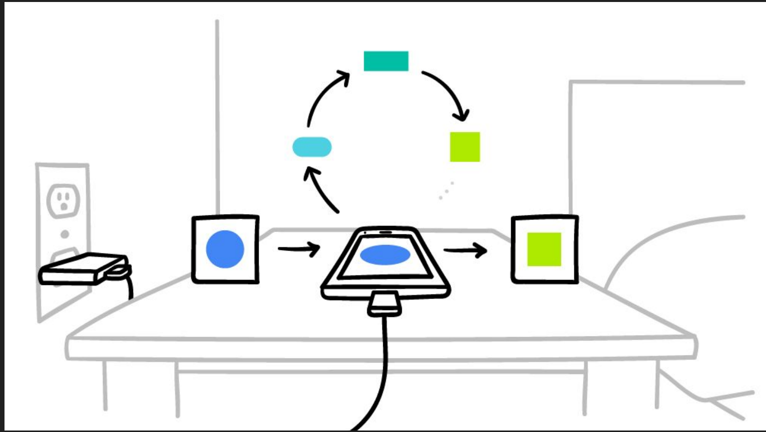
"Enduring Challenges" of Pervasive Computing

- Resource poverty
- Communication Uncertainty
- Finite Energy
- *Multi-modal interaction*
- *Scarce user attention*
- Lower privacy, security, robustness

There will always be a strong case for centralization!

Mahadev Satyanarayanan - *Mobile and Pervasive Computing* (15-821/18-843, Every Fall, including Fall 2022)

Current Federated Learning



Current Applications:

- Next word prediction
- Recommender systems

Not very performance sensitive

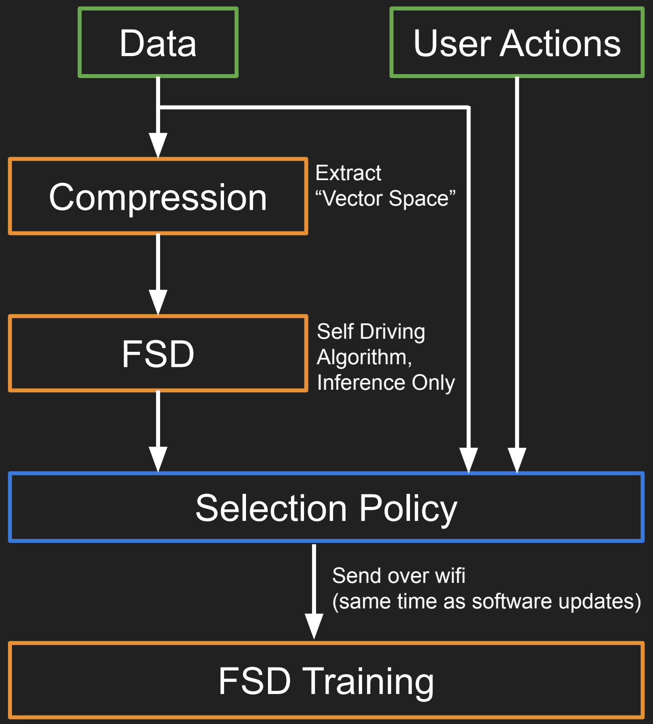
- Recruit only plugged-in, wifi-connected clients
- Research focus is mostly on accuracy, data heterogeneity, etc
- Does not hit the fundamental limits of mobile computing

Google - Federated Learning for Google Keyboard

Why Mobile Edge?

Why (Not) Mobile Edge?

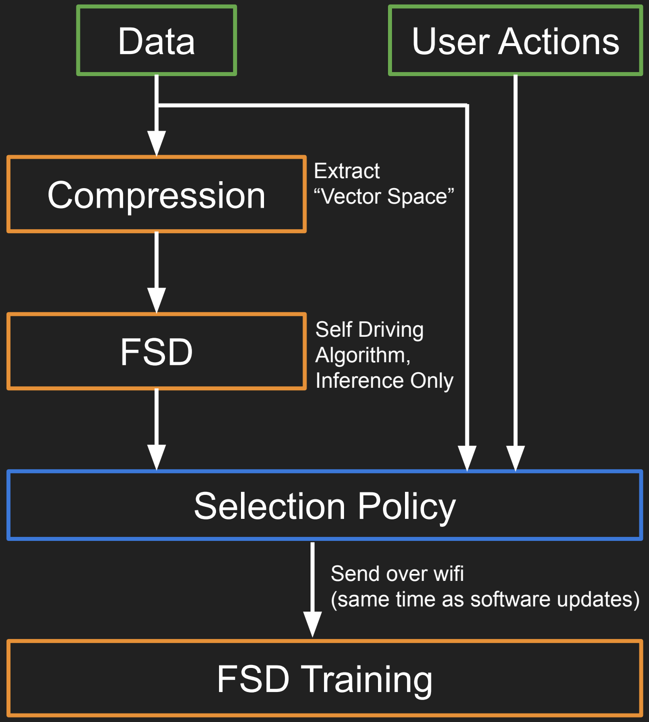
Tesla Full Self Driving Training



Why (Not) Mobile Edge?

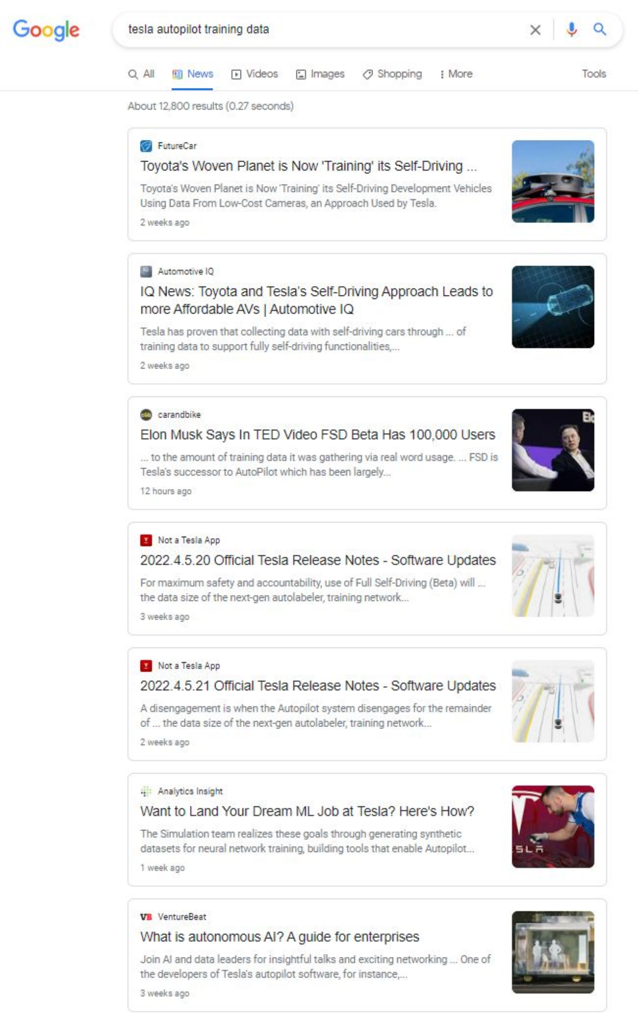
Tesla Full Self Driving Training

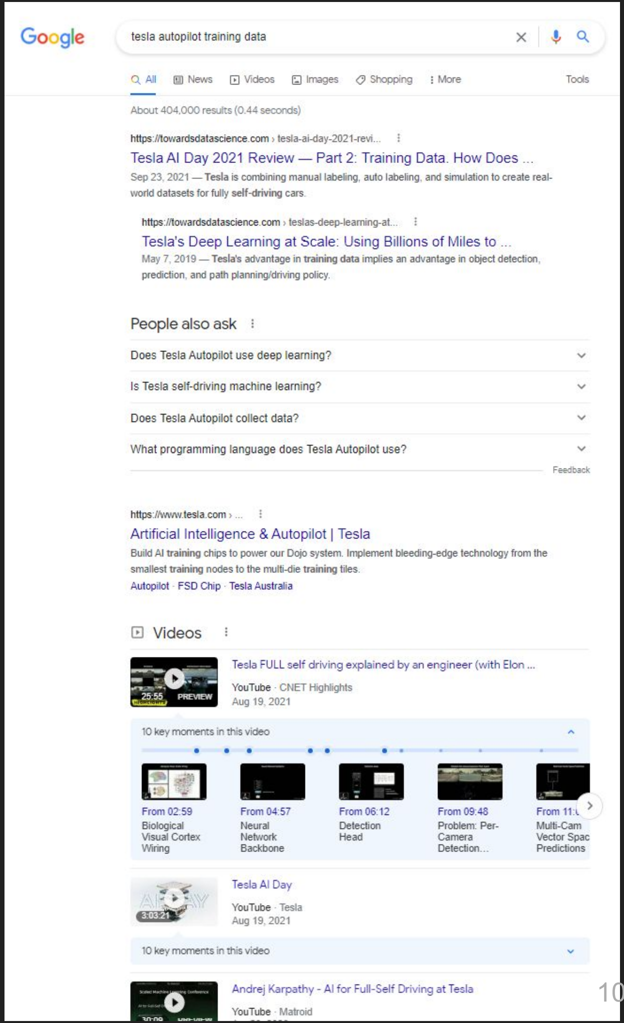
- Edge training is hard/expensive
- Users don't know about privacy
- Users don't care about privacy



No articles about privacy on the first page!

People are only mildly concerned:
[Self-Driving Cars and Data Collection: Privacy Perceptions of Networked Autonomous Vehicles]() (2017, Lujo Bauer's group)





Why Mobile Edge?

- Privacy and Liability
  - Increased awareness of data privacy
  - GDPR and "Personal Data Sovereignty"
  - Liability to data breaches and the difficulty in obtaining cyber insurance

Why Mobile Edge?

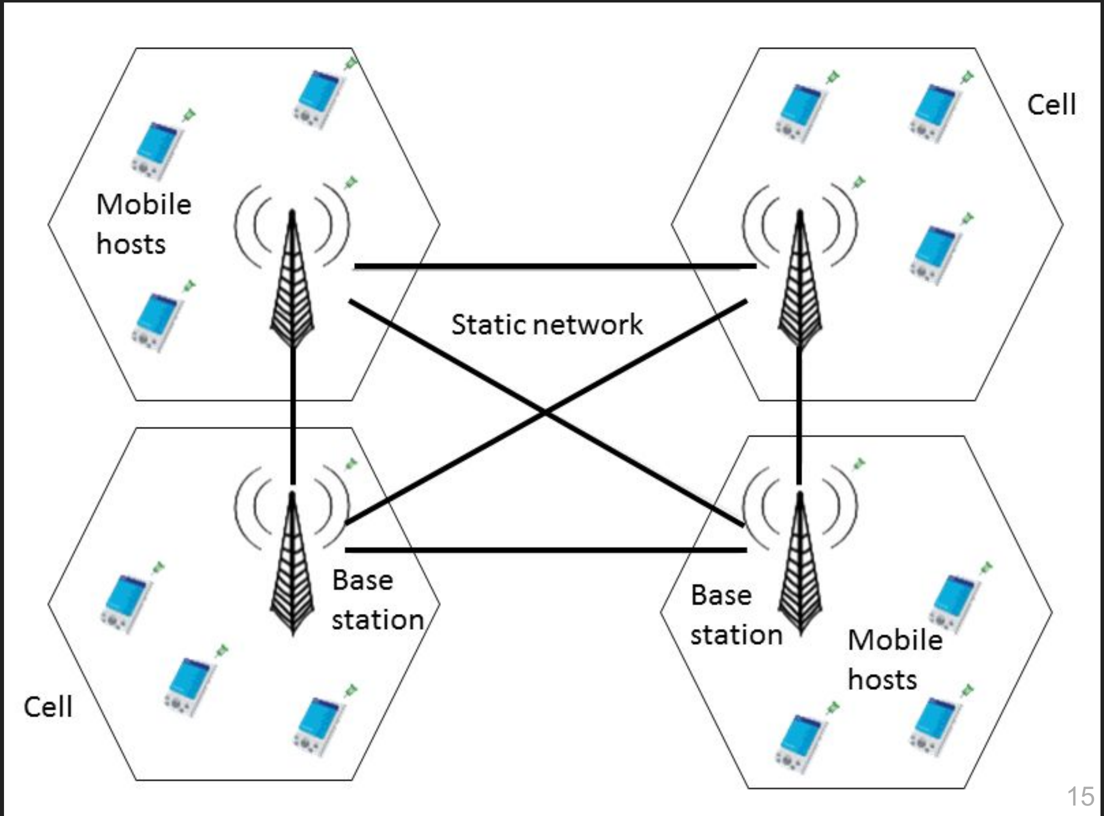
- Privacy and Liability
  - Increased awareness of data privacy
  - GDPR and "Personal Data Sovereignty"
  - Liability to data breaches and the difficulty in obtaining cyber insurance
- Communication Constraints
  - High data rate, i.e. video, LIDAR (easily 100s of gbps)
  - More domain-specific tasks where you need more data from each client

Why Mobile Edge?

- Privacy and Liability
  - Increased awareness of data privacy
  - GDPR and “Personal Data Sovereignty”
  - Liability to data breaches and the difficulty in obtaining cyber insurance
- Communication Constraints
  - High data rate, i.e. video, LIDAR (easily 100s of gbps)
  - More domain-specific tasks where you need more data from each client
- **Rapid Iteration**
  - Relative to training time
  - Without rapid iteration, none of these constraints matter!

Hierarchical FL

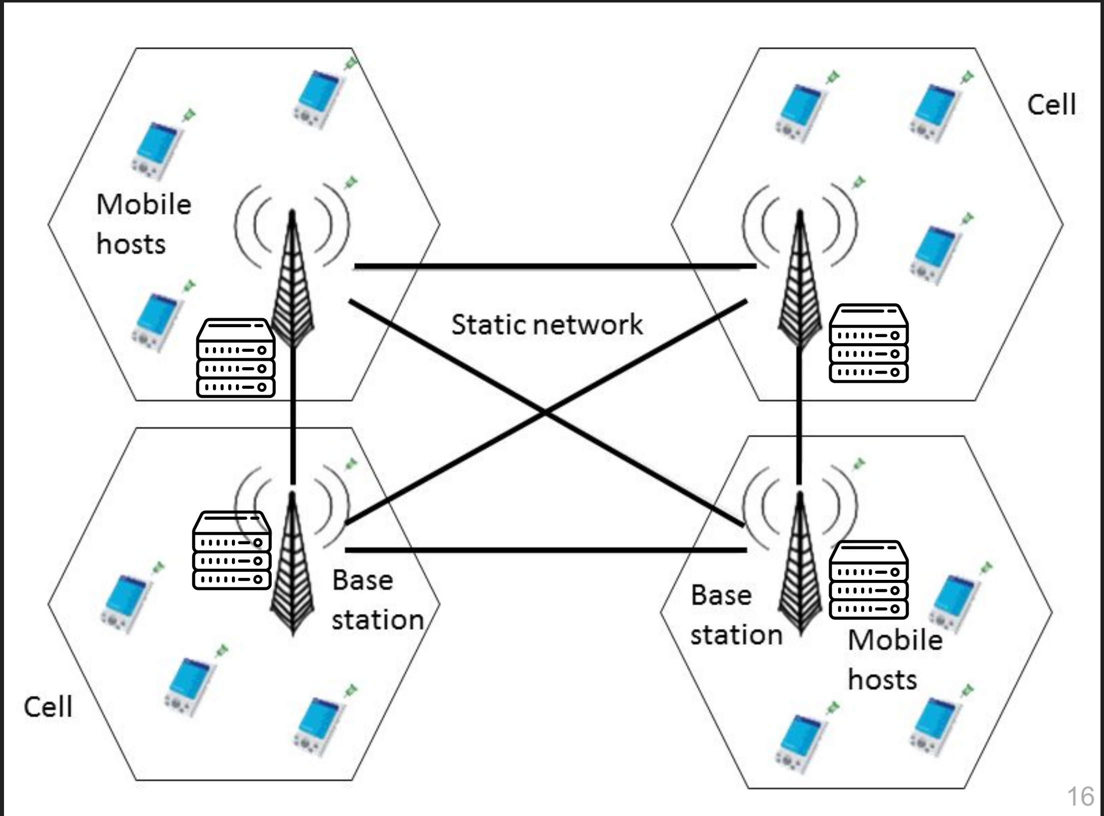
Physical Architecture



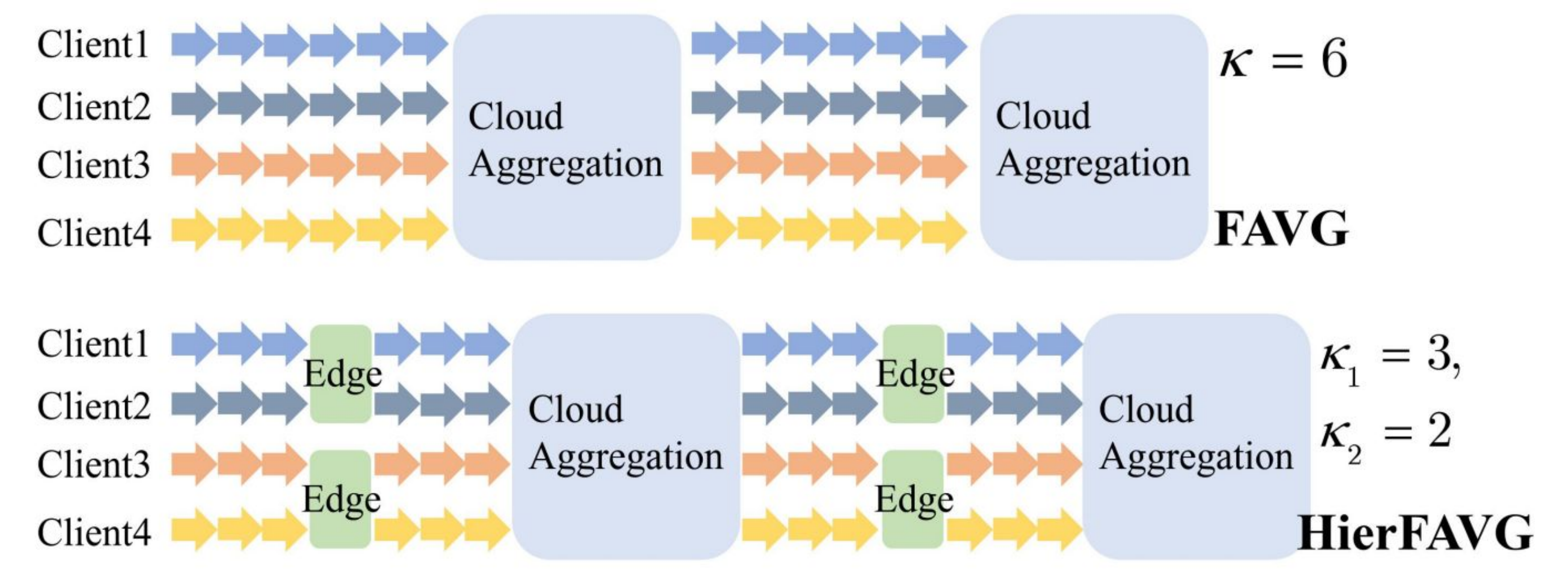
Just add Edge Servers!

[Verizon 5G Edge]

(It's an AWS virtual machine connected to the cellular network somewhere)

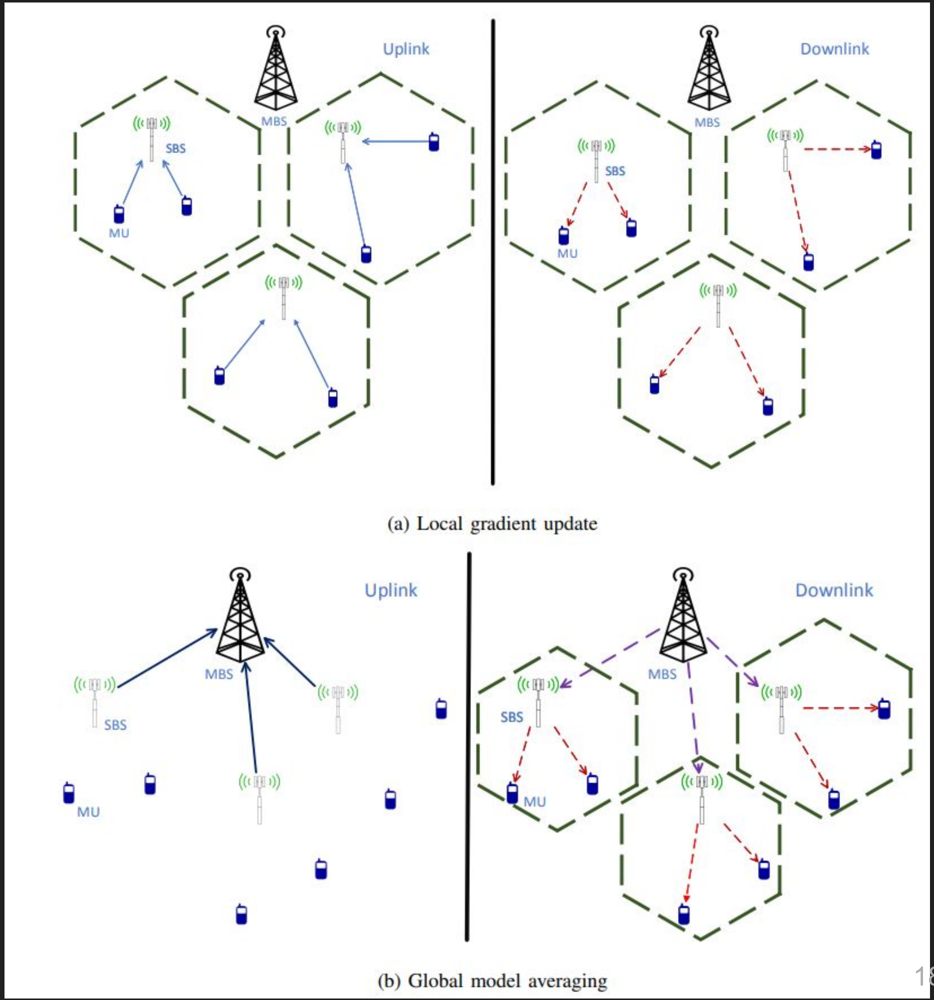


Hierarchical Federated Learning



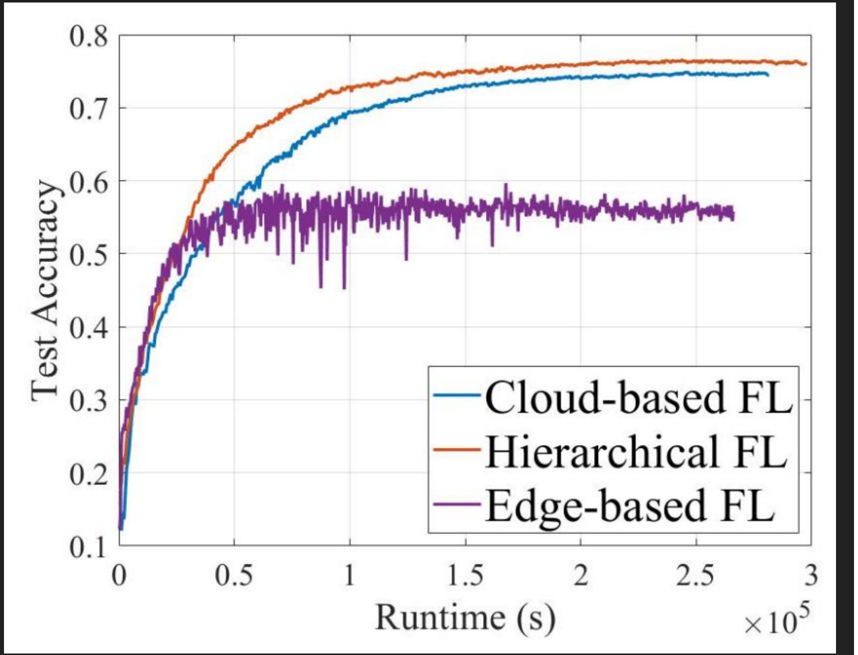
Client-Edge-Cloud Hierarchical Federated Learning (2019)

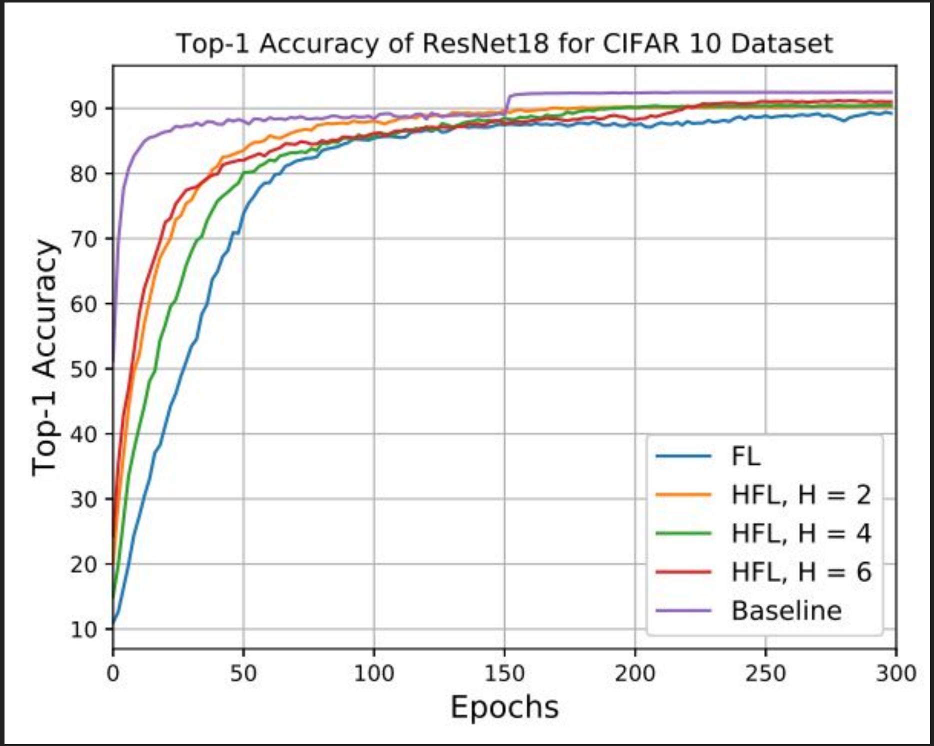
Hierarchical Federated Learning



[Hierarchical Federated Learning Across Heterogeneous Cellular Networks]() (2019)

More Synchronization = Faster Convergence





*experiments use 50 and 28 clients, respectively

Hierarchical Federated Learning

Extensions:

- Arbitrarily many levels (2022)
- Edge aggregation server selection and scheduling (2020)

Possible Ideas:

- Per-cluster adaptation (number of steps, gradient compression, etc.)
- Cluster selection and client selection
- Mixed hierarchical, non-hierarchical FL

Why Not Now?

Hierarchical FL is “canonical,” but…

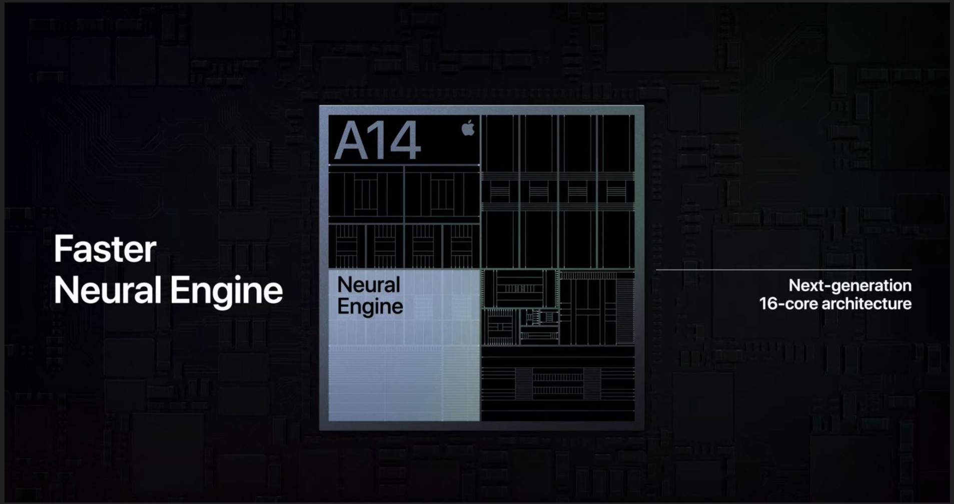
Why Not Now?

Hierarchical FL is “canonical,” but…

- Infrastructure is not fully there yet
- Edge server deployment is difficult
- Data is billed to the mobile user, not the developer. No discount for edge communication.

Compute-Aware Client Selection

Today’s Server, Tomorrow’s Edge



Compute-Aware Client Selection

Most obvious approach: pick all clients that will complete in time.

[Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge](), 2019

Algorithm 3 `Client Selection` in Protocol 2

Require: Index set of randomly selected clients $\mathbb{K}'$

Initialization $\mathbb{S} \leftarrow \{\}$, $T^{\mathrm{d}}_{\mathbb{S}=\emptyset} \leftarrow 0$, $\Theta \leftarrow 0$
while $|\mathbb{K}'| > 0$ **do**
  $x \leftarrow \arg\max_{k \in \mathbb{K}'} \frac{1}{T^{\mathrm{d}}_{\mathbb{S}\cup k} - T^{\mathrm{d}}_{\mathbb{S}} + t^{\mathrm{UL}}_k + \max\{0, t^{\mathrm{UD}}_k - \Theta\}}$
  remove x from $\mathbb{K}'$
  $\Theta' \leftarrow \Theta + t^{\mathrm{UL}}_x + \max\{0, t^{\mathrm{UD}}_x - \Theta\}$
  $t \leftarrow T_{\mathrm{cs}} + T^{\mathrm{d}}_{\mathbb{S}\cup x} + \Theta' + T_{\mathrm{agg}}$
  if $t < T_{\mathrm{round}}$ **then**
    $\Theta \leftarrow \Theta'$
    add x to $\mathbb{S}$
  end if
end while
return $\mathbb{S}$

In summary, `Client Selection` is formulated by the following maximization problem with respect to $\mathbb{S}$:

$$\max_{\mathbb{S}} \quad |\mathbb{S}| \qquad \text{s.t.} \quad T_{\mathrm{round}} \geq T_{\mathrm{cs}} + T^{\mathrm{d}}_{\mathbb{S}} + \Theta_{|\mathbb{S}|} + T_{\mathrm{agg}}. \tag{4}$$

Compute-Aware Client Selection

[FedMCCS: Multicriteria Client Selection Model for Optimal IoT Federated Learning](), 2021

B. Problem Formulation

We formulate our problem as a bilevel maximization with knapsack and other constraints as follows:

$$\max_{X_S} |X_S|$$

subject to

$$\begin{cases} \forall X_{f^i_{z=1}} \sum \mathrm{Util}^{X_{f_z}}_{r\in\{\mathrm{CPU,Memory,Energy}\}} < \mathrm{Budget}^{X_{f_z}}_r [co_1] \\ \forall X_{f^i_{z=1}} \sum\left(T^{X_{f_z}}_d + \mathrm{Util}^{X_{f_z}}_{r=T_{ud}} + T^{X_{f_z}}_{ul}\right) < T[co_2] \end{cases}$$

subject to

$$\max ER_{X_{f^i_{z=1}}} = \left[\frac{|X_{f_z}.l_A|}{|X_{f_z}.l_A| + |X_{f_z}.l_N|} \times 100\right][co_3]. \quad (1)$$

Compute-Aware Client Selection

[FedMCCS: Multicriteria Client Selection Model for Optimal IoT Federated Learning](), 2021

Select as many clients as possible, such that:

(1) we do not exceed the resource budget

(2) we do not exceed the round time

(3) selection also maximizes clients with minority classes

B. Problem Formulation

We formulate our problem as a bilevel maximization with knapsack and other constraints as follows:

$$\max_{X_S} |X_S|$$

subject to

$$\begin{cases} \forall X_{f_{z=1}^{i}} \sum \text{Util}_{r\in\{\text{CPU},\text{Memory},\text{Energy}\}}^{X_{f_z}} < \text{Budget}_r^{X_{f_z}} [co_1] \\ \forall X_{f_{z=1}^{i}} \sum \left(T_d^{X_{f_z}} + \text{Util}_{r=T_{ud}}^{X_{f_z}} + T_{ul}^{X_{f_z}} \right) < T [co_2] \end{cases}$$

subject to

Percent of “abnormal” samples

$$\max ER_{X_{f_{z=1}^{i}}} = \left[\frac{|X_{f_z}.l_A|}{|X_{f_z}.l_A| + |X_{f_z}.l_N|} \times 100 \right] [co_3]. \quad (1)$$

Compute-Aware Client Selection

Open questions:

- How do you select the budget and round time?
- How do you reconcile compute-aware selection with fairness if data is correlated to compute in hard-to-quantify ways?
- How do you know (i.e. predict) the resource usage ahead of time, especially if a device hasn't participated recently or has never participated?

Related work: Runtime Performance Prediction for DL, 2021

Why Not Now?

- Under-studied (probably because it’s really hard to study!)
- Performance-sensitive Federated Learning on the mobile edge not really used or needed yet
- Google: devices are limited in diversity, well-profiled in advance, and developers have root access

Conclusion

Hierarchical FL

- Obvious, simple, useful
- Very easy to implement in the lab
- Very hard to implement in the wild

Compute-aware Client Selection

- Logical, but not so simple
- Not needed until FL becomes more popular (especially by non-privileged parties)