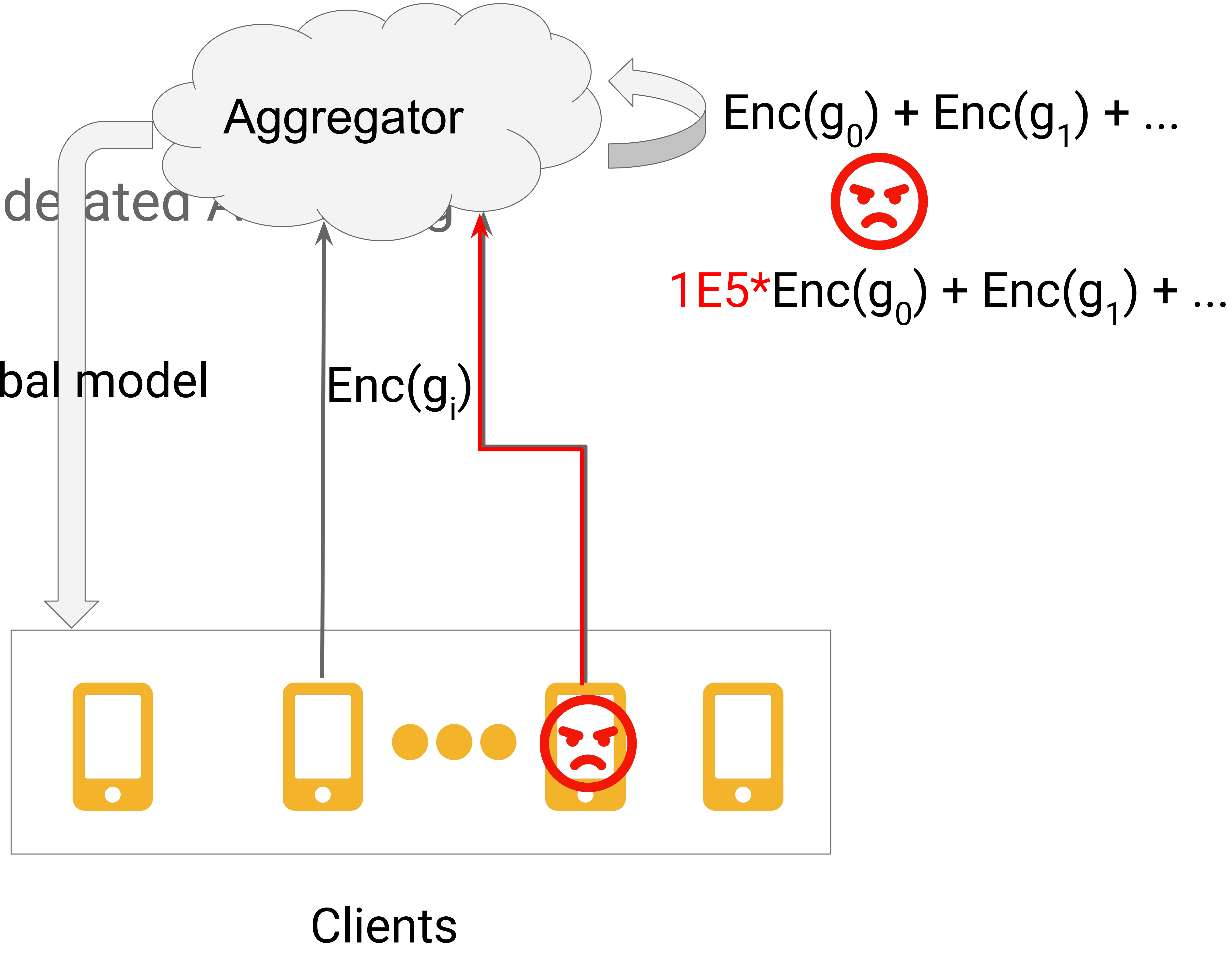
# Towards an Efficient System for Differentially-private, Cross-device Federated Learning

**Kunlong Liu**; Richa Wadaskar; Trinabh Gupta
*University of California, Santa Barbara*

# Gboard’s next word prediction

- Fede ated A g



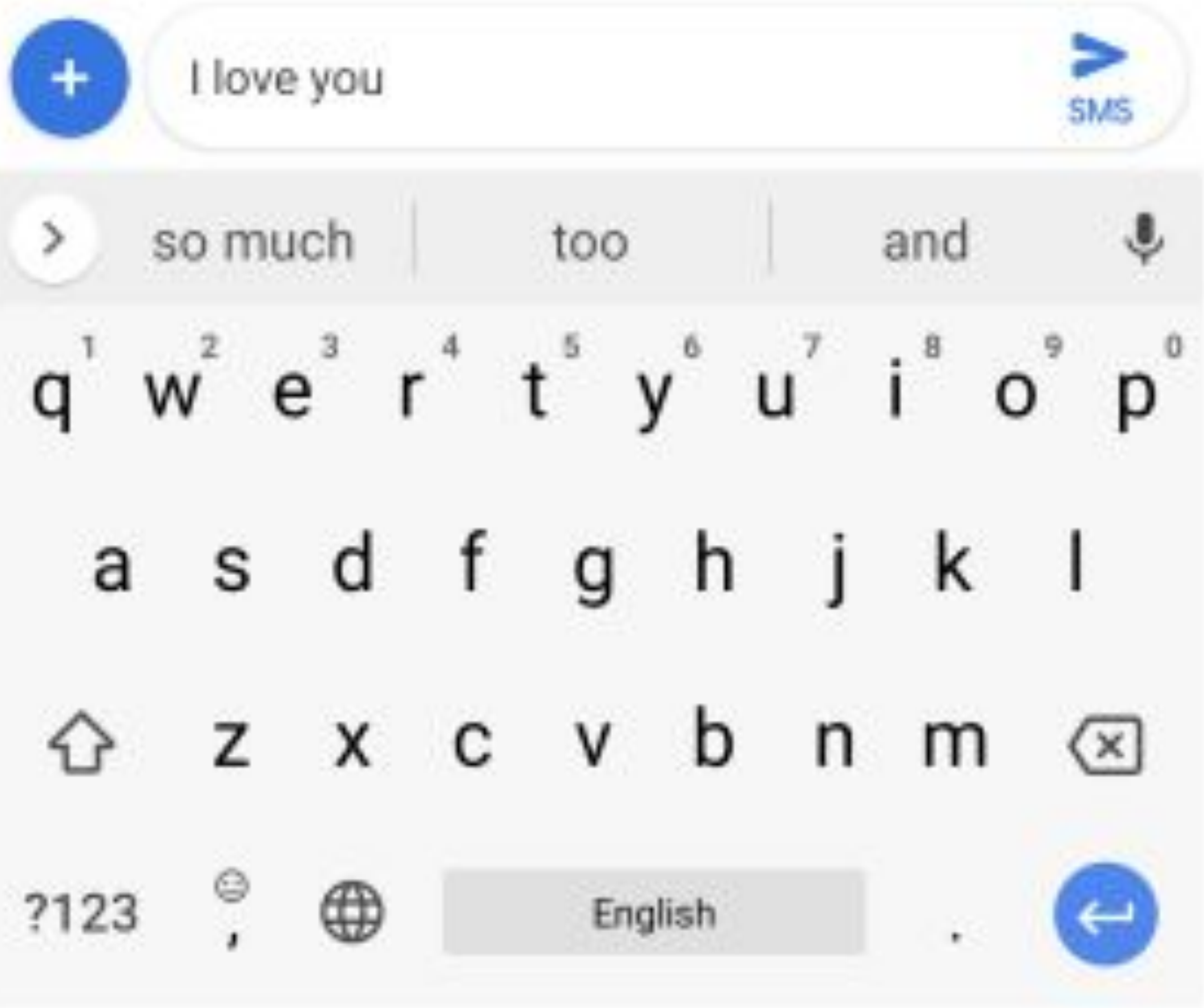


**Fig. 1.** Next word predictions in Gboard. Based on the context “I love you”, the keyboard predicts “and”, “too”, and “so much”.

# Goals

- **Strong guarantees**
    - Differential privacy, even when some clients and the aggregator are both malicious
    - Correctness or robustness of training: bounded gradients
- **Scalability**
    - Scale to million or billion
- **Efficiency**
    - low client-side cost

# Orchard [OSDI ’20]

- **Strong guarantees**
  - Differential privacy, even malicious clients and malicious aggregator
  - Correctness or robustness of training: bounded gradients
- **Scalability**
  - Scale to million or billion
- ~~**Efficiency**~~
  - **High** client-side cost, both computation and network

# Gboard + Orchard [OSDI '20]

- **Setting**
  - 1.4 M parameters, 3000 rounds to converge, on a 6-core laptop
- **Computation**
  - 4 minutes per device per round.
- **Network**
  - 764 MB download per device per round

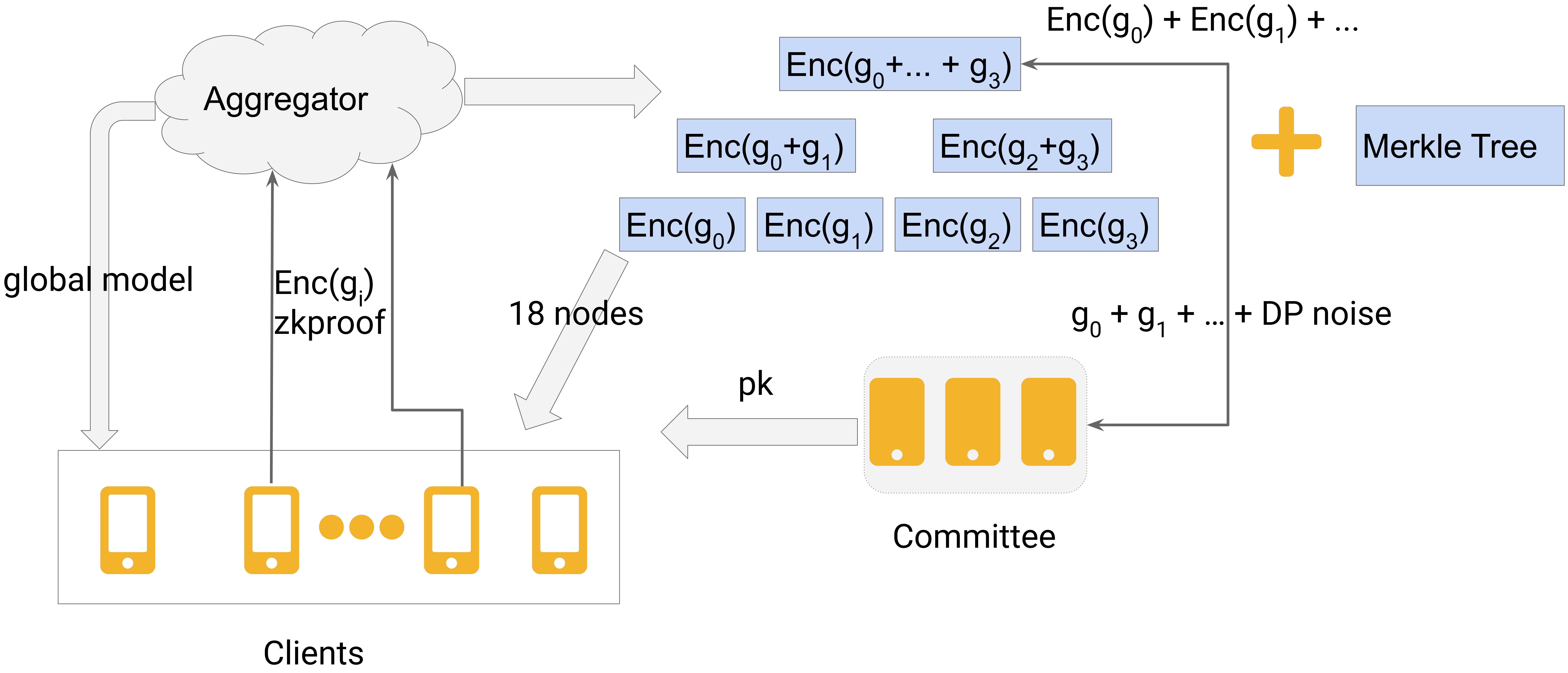
# Gboard + Atom [Our system]

- **Setting**
  - 1.4 M parameters, 3000 rounds to converge, on a 6-core laptop
- **Computation**
  - move ⅓  CPU time to offline phase.
- **Network**
  - 5 MB download per device per round

# The rest of the talk

- **Orchard**
    - architecture, threat model, key performance-related protocols
- **Key ideas of Atom**

# Architecture of Orchard

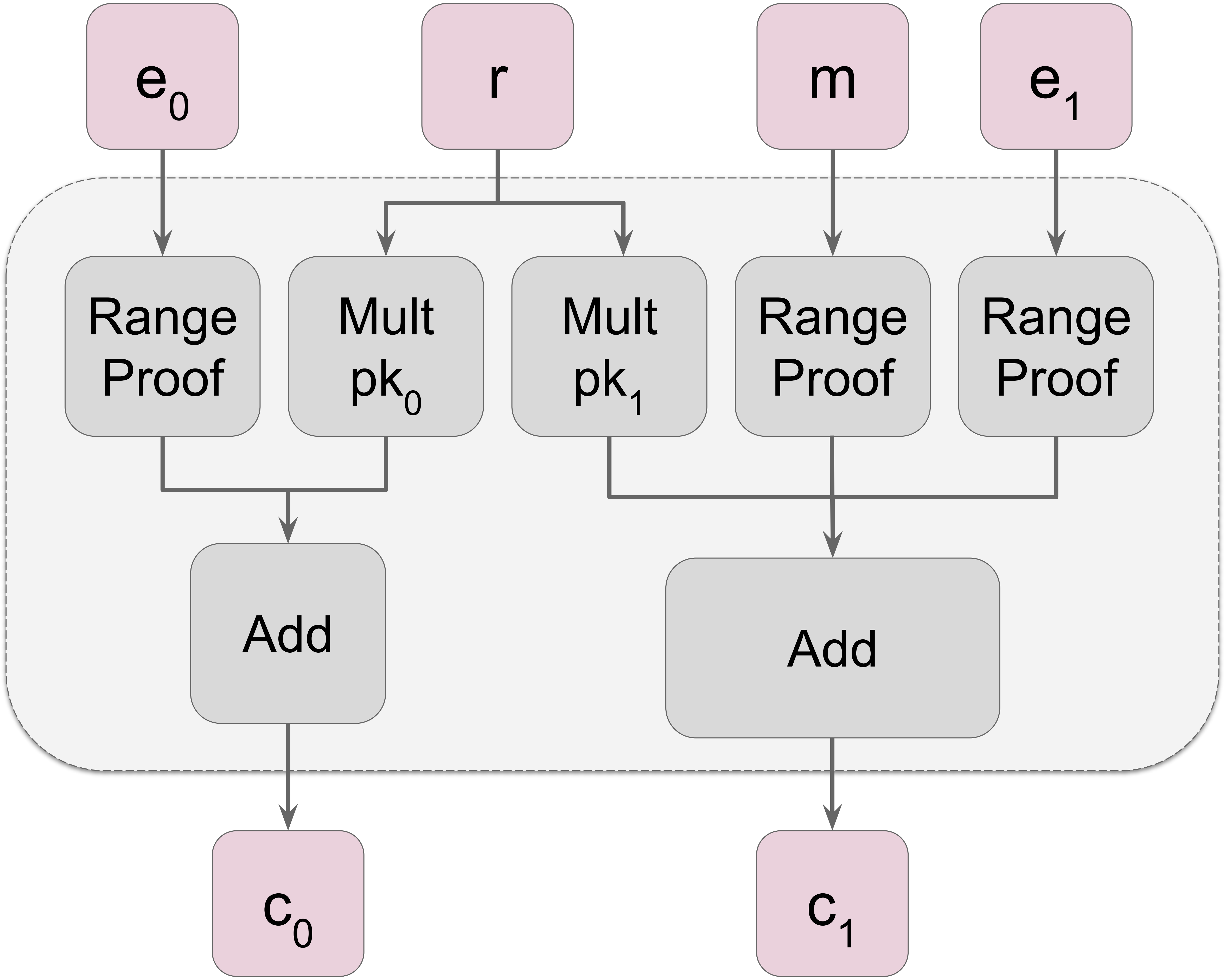


# Threat Model

- **Aggregator: occasionally byzantine(OB)**
  - a rogue system administrator is executing an attack
- **Clients: mostly correct (MC)**
  - a configurable small fraction (1-5%) can be malicious. (million out of billion)
- **Security guarantees**
  - **Privacy** always guaranteed even if the aggregator is malicious
  - **Integrity** guaranteed when aggregator is not malicious

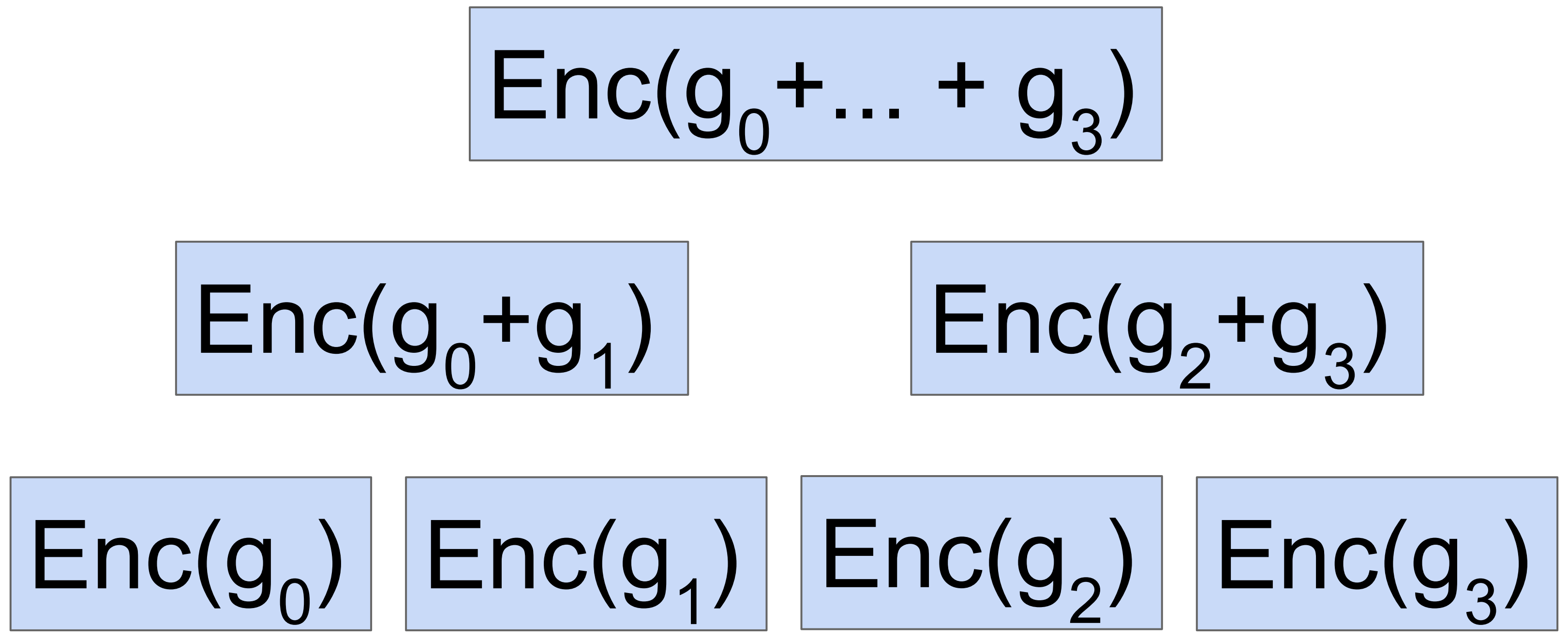
# CPU Bottleneck of Orchard

- zero-knowledge proof for the ciphertext
  - Proof time
    - ~8s for 1 CT (single thread)
    - ~235s for 1.4M parameters (342 CTs) on 6-core (12 threads)



# Network Bottleneck of Orchard

- verifying the summation tree
  - 18 nodes
    6 leaf nodes + 12 non-leaf nodes
  - Network cost
    760MB for 1.4M parameters (342 CTs)



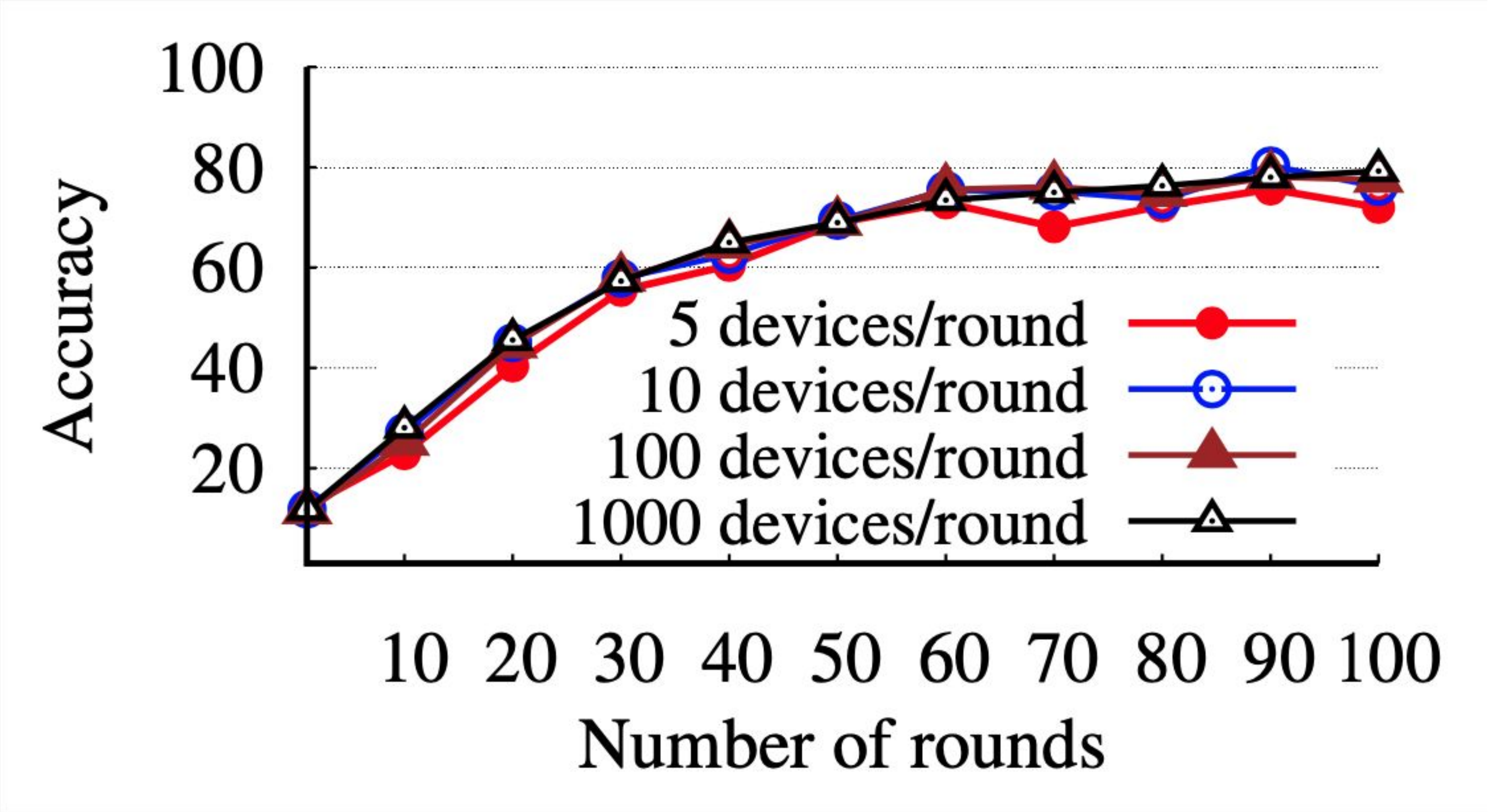
Enc($g_0$+... + $g_3$)

Enc($g_0$+$g_1$) Enc($g_2$+$g_3$)

Enc($g_0$) Enc($g_1$) Enc($g_2$) Enc($g_3$)

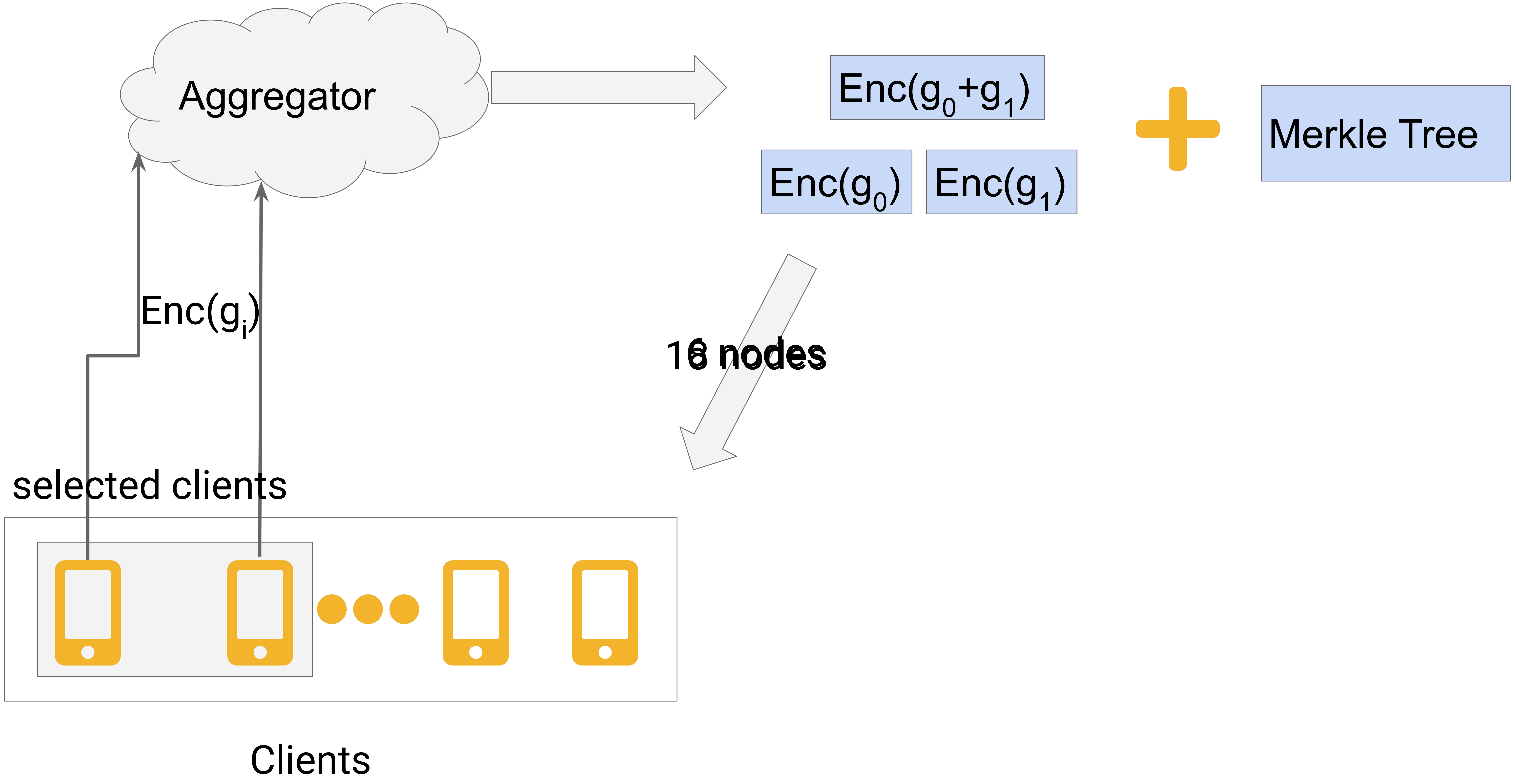
# Idea 1: Switching to stochastic FedAvg

- Full batch gradient descent is not necessary

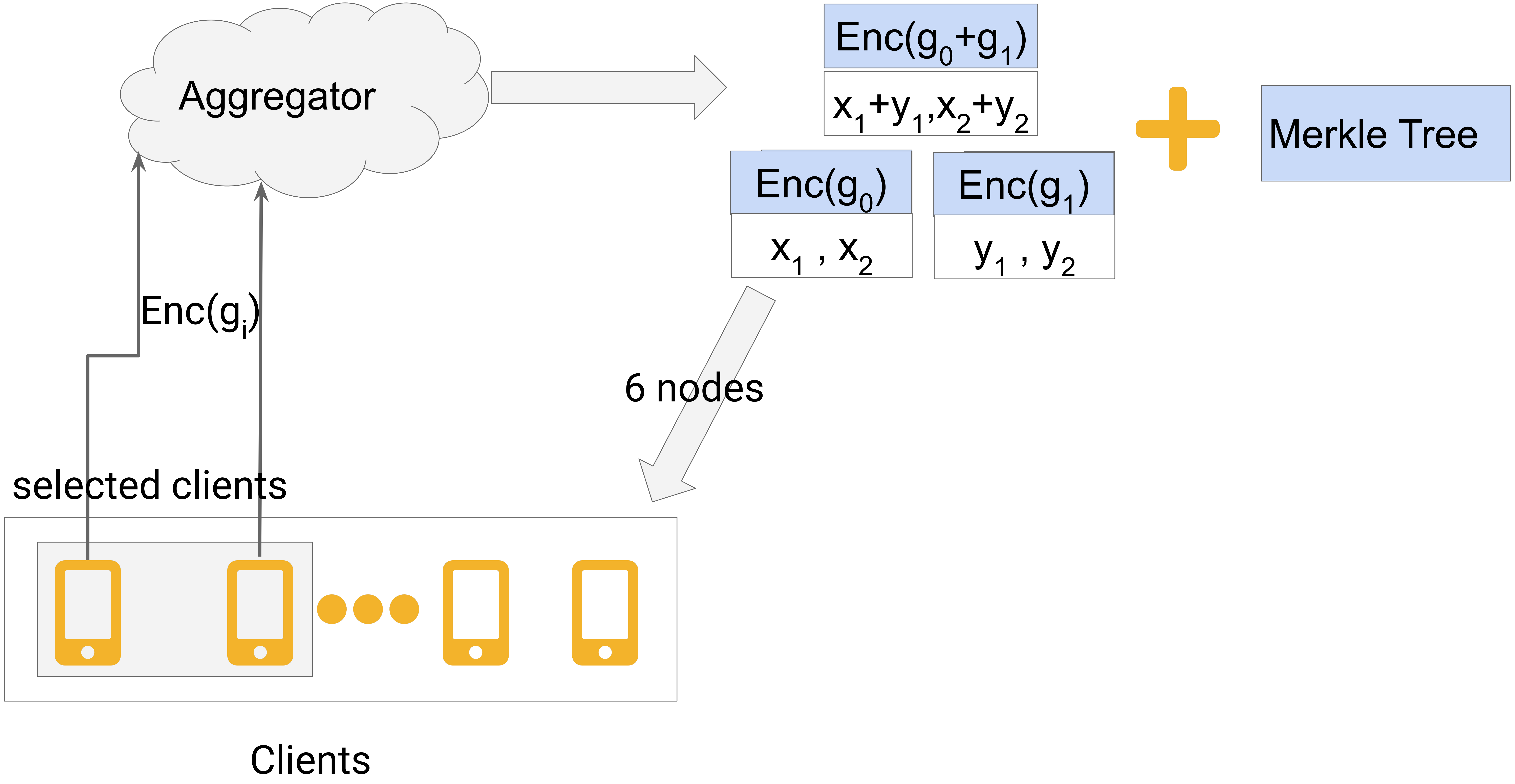


Accuracy of a DNN model on EMNIST dataset

# Switching to stochastic FedAvg



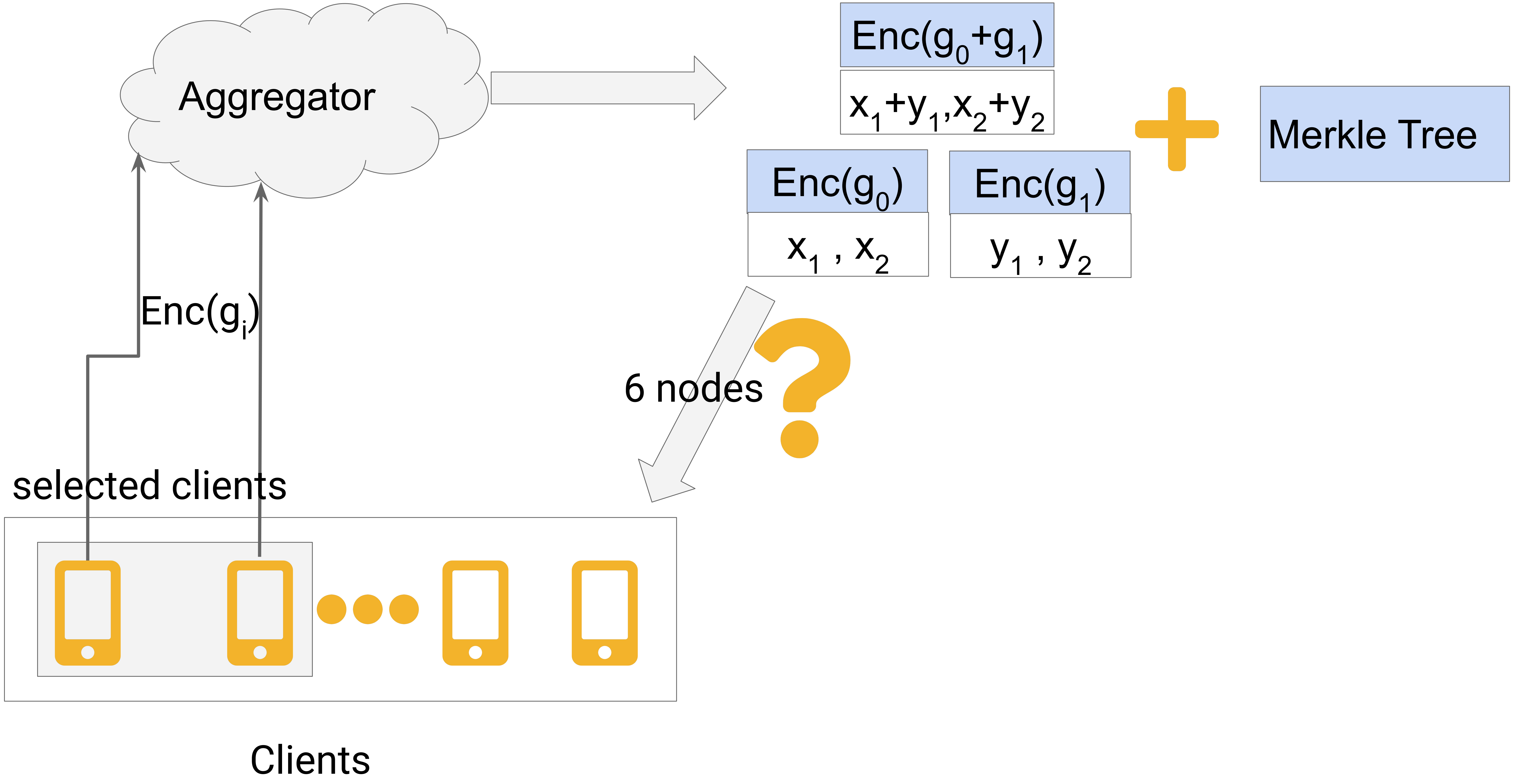
# Switching to stochastic FedAvg



# Cost

- **Network cost saving for Gboard**
  - If the fraction is 1%, 11.24 MB per device.
  - If the fraction is 2%, 22.48 MB per device.
  - If the fraction is 5%, 56.21 MB per device.

# Idea 2: Integrating Polynomial Identity Test



# Polynomial Identity Test

- To check f(x) == 0
- In a prime field F, if a non-zero polynomial f(x) has M degree, it has at most M zero points.
- $Pr[r \leftarrow F; f(r) = 0] \leq \frac{M}{|F|}$

# RLWE Encryption

- Enc(m) = (as+e, bs+e’+m), where a,b,s,e,m are all polynomials.

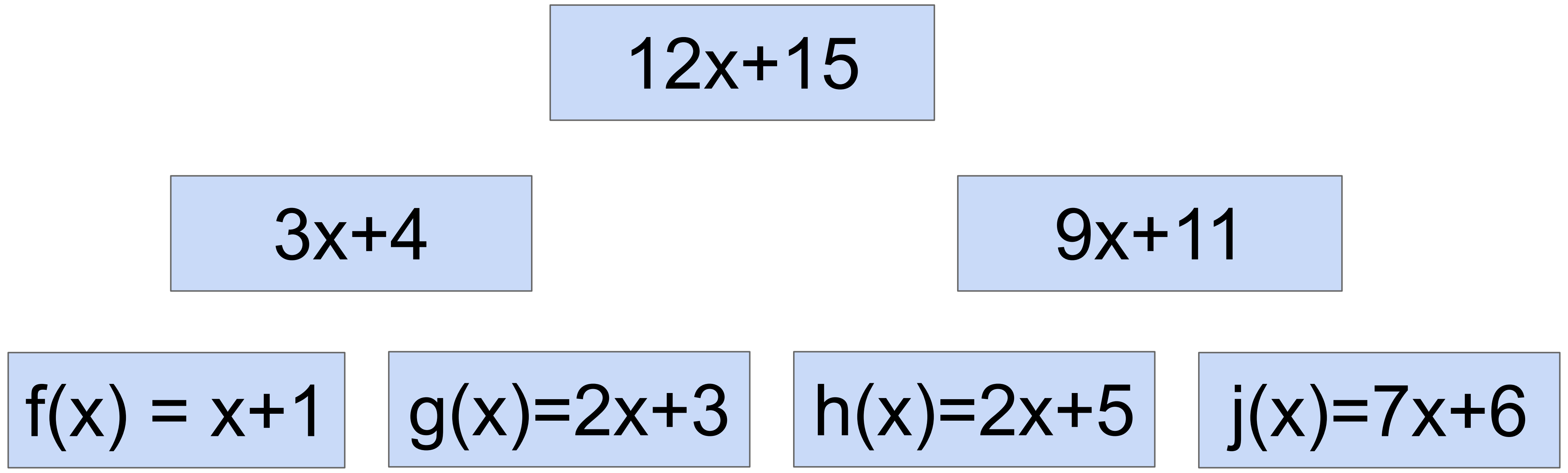
- Enc($g_1$) = ($a_1$, $b_1$), Enc($g_2$) = ($a_2$, $b_2$), Enc($g_1$+$g_2$)=($a_3$, $b_3$)
- Enc($g_1$) + Enc($g_2$) == Enc($g_1$+$g_2$)

⇒ $a_1$+ $a_2$ == $a_3$ and $b_1$+ $b_2$== $b_3$

⇒ $a_1$+ $a_2$- $a_3$ = **0** and $b_1$+ $b_2$ - $b_3$ = **0**

⇒ $a_1$(r) + $a_2$(r) - $a_3$(r) = 0 and $b_1$(r) + $b_2$(r) - $b_3$(r) = 0

# New summation tree

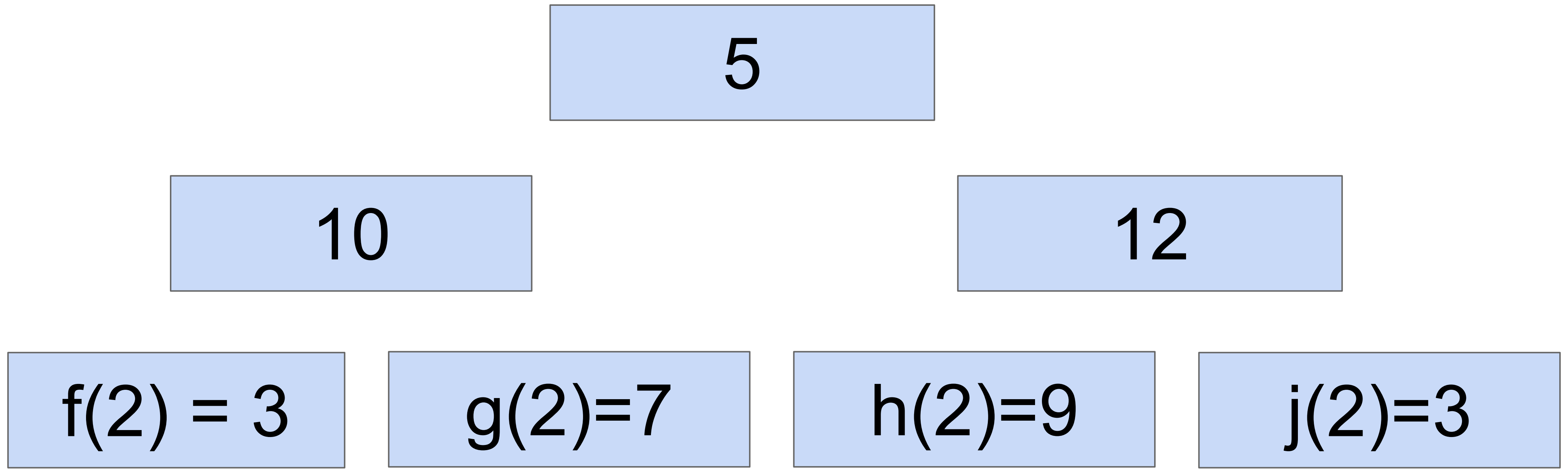


12x+15

3x+4 9x+11

f(x) = x+1 g(x)=2x+3 h(x)=2x+5 j(x)=7x+6

Orchard
mod 17



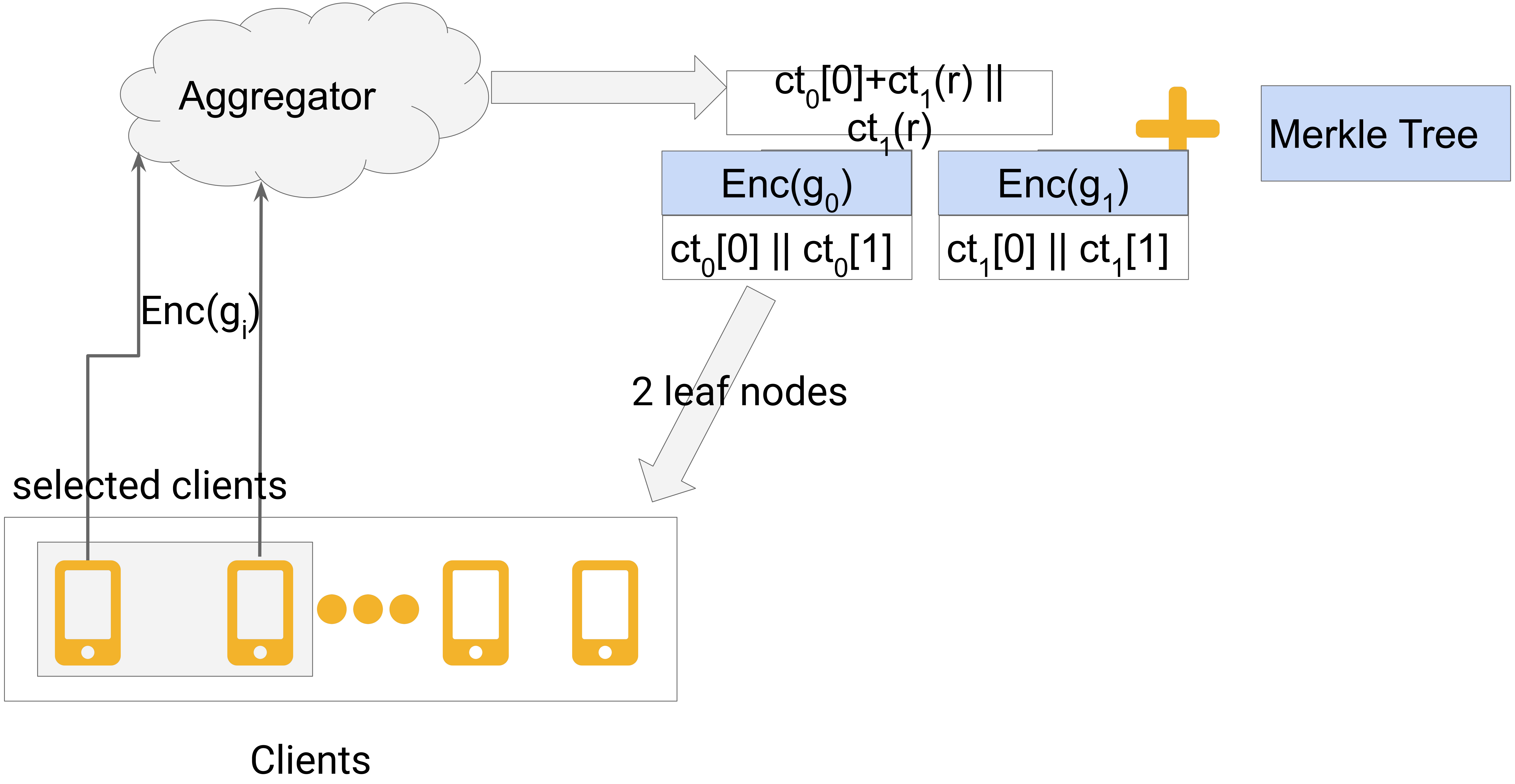
5

10 12

f(2) = 3 g(2)=7 h(2)=9 j(2)=3

Atom
mod 17, r=2

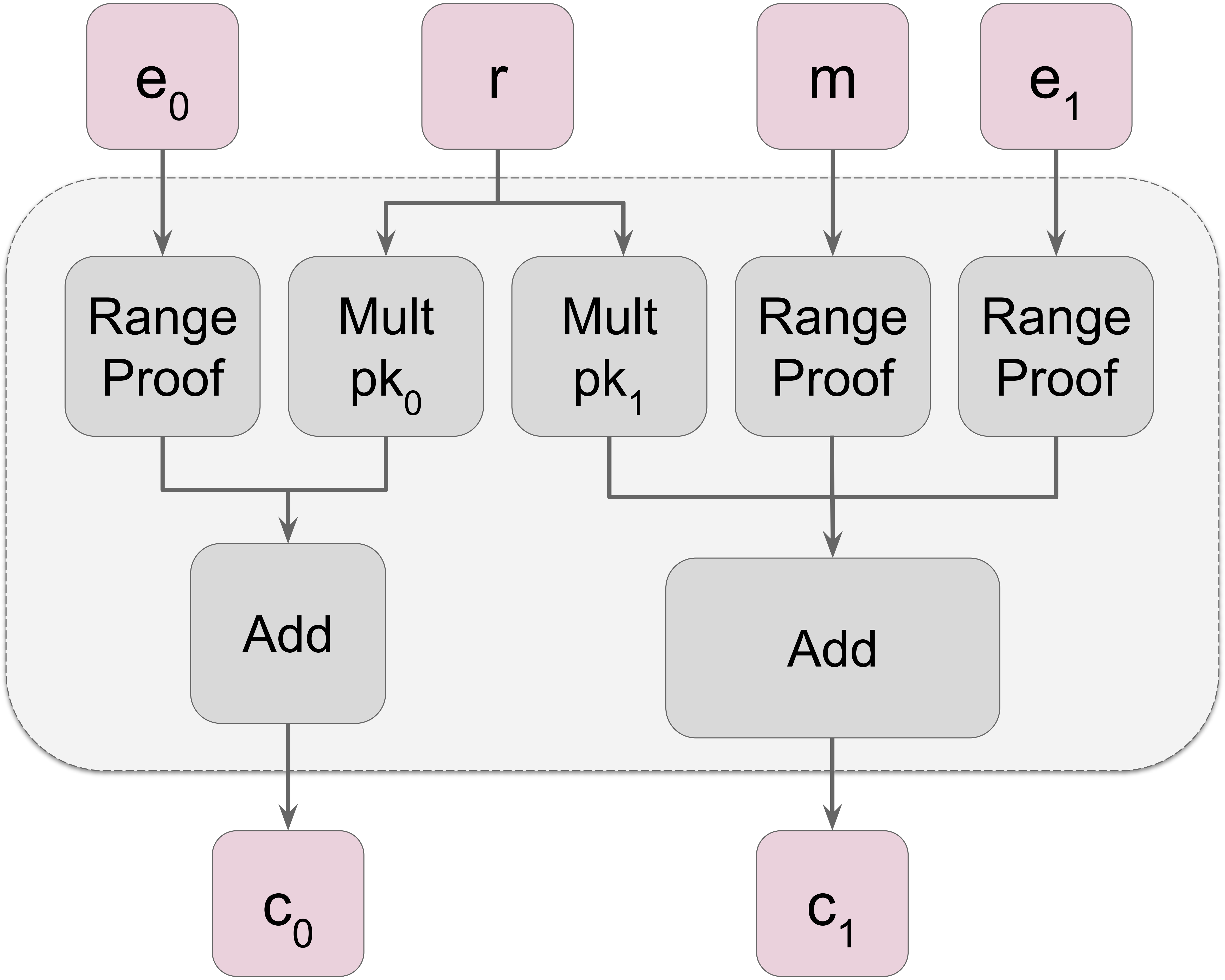
# Cost


Aggregator
$ct_0[0]+ct_1(r)$ || $ct_1(r)$
Enc($g_0$)
$ct_0[0]$ || $ct_0[1]$
Enc($g_1$)
$ct_1[0]$ || $ct_1[1]$
Merkle Tree
Enc($g_i$)
2 leaf nodes
selected clients
Clients

# Idea 3: Splitting into Offline Phase

- **Observation:**

  Ring multiplication most expensive



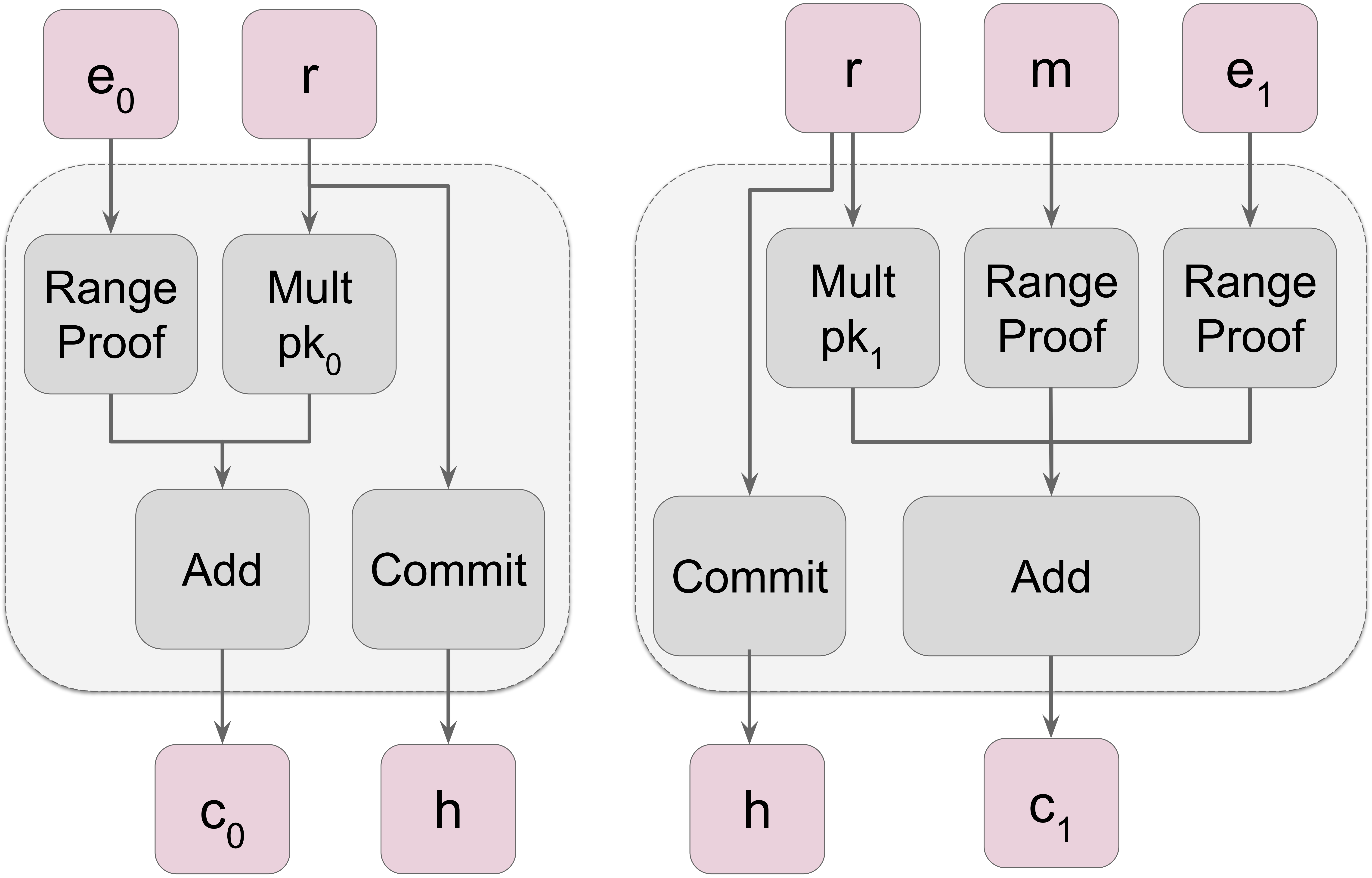
# Splitting into Offline Phase

- **Cost**

  Orchard:~235s for 342 CTs

  Online: ~155.6s for 342 CTs

  Offline: ~141.9s for 342 CTs



# Summary

- **Atom**
  - the same threat model as Orchard
  - scale to billions of clients
  - Improves the per device download
  - Improves the overall training time
- Future work
  - committee