IBM **Semiconductors**

Decentralized Collaborative Machine Learning Framework with Democracy, Diversity, and Privacy

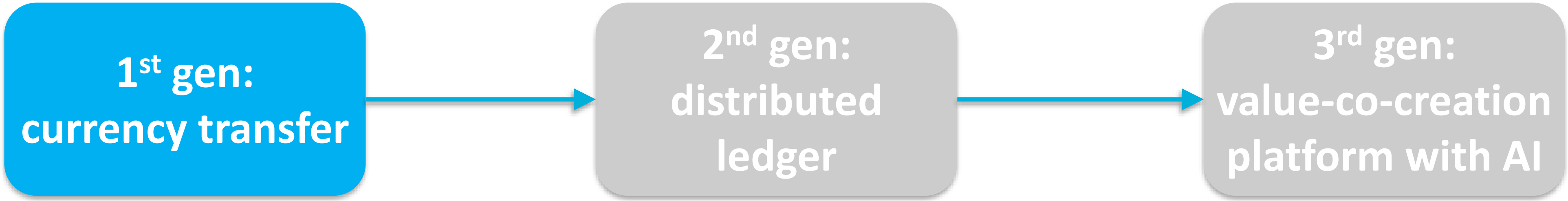
Tsuyoshi Idé (“Ide-san” 井手 剛)
Head of Data Science, IBM Semiconductors

Presented at the 2023 International workshop Blockchain Kaigi (Kobe, Japan, Oct. 2023).

Agenda

- Introduction: Blockchain as value co-creation platform
- Decentralized collaborative learning framework
- Secure decentralized aggregation
- Network topology design
- Summary

1st gen Blockchain: Designed specifically for currency transfer



1st gen: currency transfer

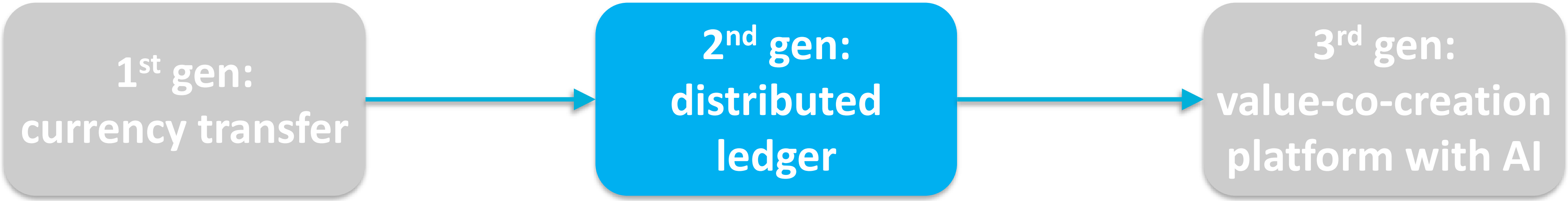
2nd gen: distributed ledger

3rd gen: value-co-creation platform with AI

- Blockchain 1.0: Bitcoin
  - Designed specifically for currency transfer
  - Verifying a transaction is trivial: just by checking account balances
  - A unique consensus algorithm is used (“proof-of-work”)
- Limitations
  - Unable to handle general business transactions
  - Proof-of-work lacks a deterministic guarantee

bitcoin

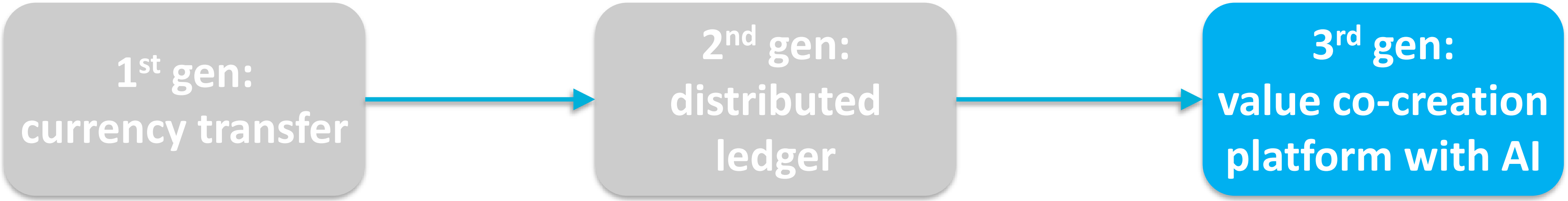
2nd gen Blockchain: General-purpose business transaction management platform


1st gen: currency transfer
2nd gen: distributed ledger
3rd gen: value-co-creation platform with AI

- Blockchain 2.0: Smart-Contract-enabled transaction management platform
  - Designed to be able to handle “general” business transactions
  - Traditional consensus algorithm (e.g. PBFT) is typically used
- Limitations
  - Validating smart contracts is not straightforward (c.f. money transfer)
  - No “knowledge discovery” elements: only perform predefined routines

ethereum

3rd gen Blockchain: Towards AI-integrated value co-creation platform



- Blockchain 3.0: Value co-creation platform
- "Value co-creation": Share data, and collaboratively develop new insights that cannot be accessed when looking at your own data alone
- AI/machine learning provides a systematic means for value co-creation

Three requirements of value co-creation platform: Democracy, diversity, privacy

- Democracy
  - All participants are equal
  - No dictator/central server that controls everything
- Diversity
  - All participants are not the same
  - They wish to have insights customized to each
- Privacy
  - All participants can keep own data secret
  - Collaborative learning is not communism

Agenda

- Introduction: Blockchain as value co-creation platform
- Decentralized collaborative learning framework
- Secure decentralized aggregation
- Network topology design
- Summary

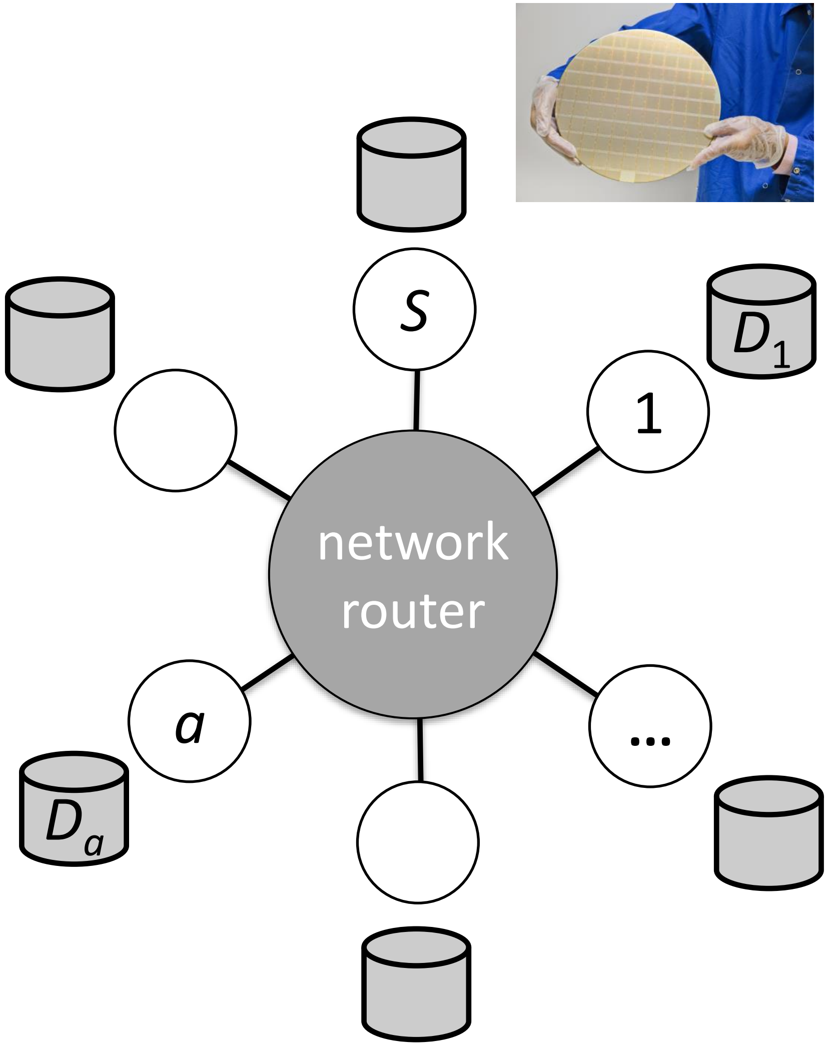
These three requirements are naturally translated into specific machine learning problems

- Democracy → decentralized
  - All participants are equal
  - No dictator/central server that controls everything
- Diversity → multi-task
  - All participants are not the same
  - They wish to have insights customized to each
- Privacy
  - All participants can keep own data secret
  - Collaborative learning is not communism

**Blockchain 3.0
as
multi-task learning
with decentralization
and privacy
constraints**

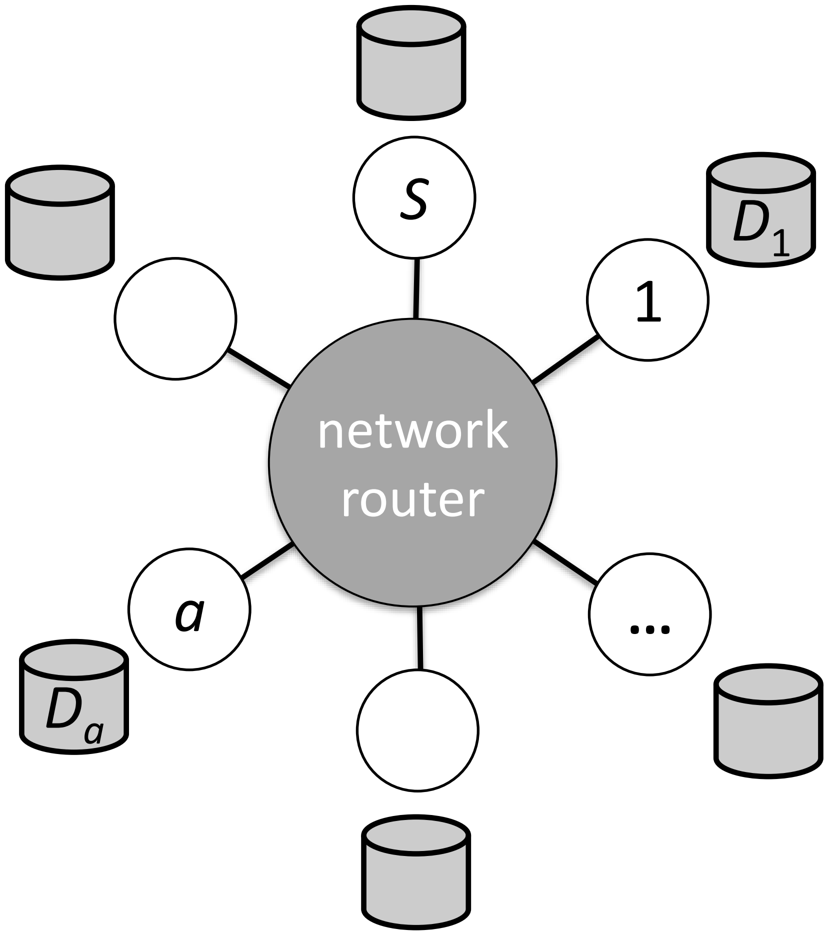
Use-case example: Collaborative training of anomaly detection models of semiconductor manufacturing tools

- Why collaborative?
  - Anomalies are rare. Collecting as many anomaly examples as possible is critical for a high accuracy.
- Why multi-task?
  - A one-size-fit-all model is typically not useful as the operating conditions are different.
- Why privacy/decentralized?
  - Information on failures is highly confidential. They don’t want to disclose raw anomaly data.
  - They don’t want send sensitive data to the server, either.



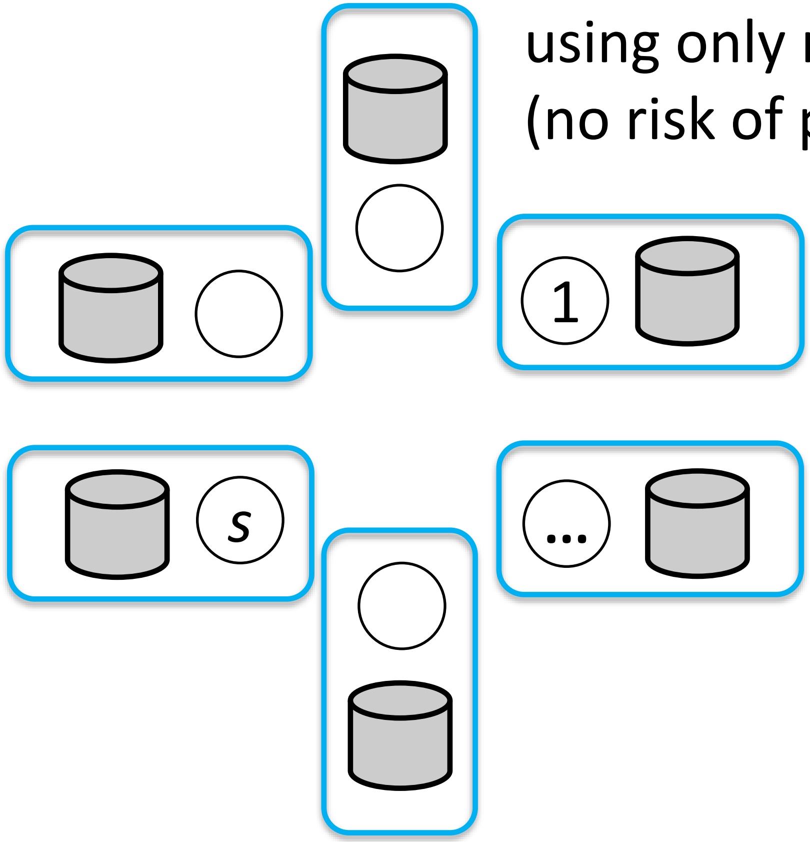
(For ref.) We focus on multi-task density estimation as a concrete machine learning problem

- Each participant (*a*=1,.., *S*) has a dataset D^a privately
  - $\mathcal{D}^a = \{\boldsymbol{x}^{a(1)}, \boldsymbol{x}^{a(2)}, \ldots, \boldsymbol{x}^{a(N_a)}\}$
- The model in this case is the probability density function (pdf) of observed data ***x***
  - ***x***: real-valued multi-dimensional vector
- No central server. Only P2P communication is allowed according to a given network topology
- All the participants share the motivation of refining their model by leveraging other participants' knowledge



Model training can be done by iterating global and local updates in maximum likelihood

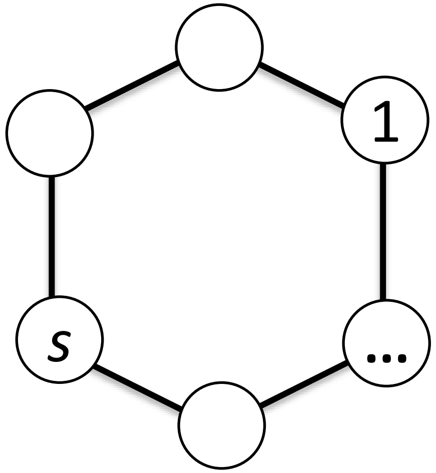
Local updates:
compute statistics locally using only my own data (no risk of privacy breach)



Iterates until convergence

Global consensus:

- Aggregate local statics (under some risk of privacy breach)
- Obtain globally optimal models



Question: How do we achieve decentralized and privacy-preserving training?

- The original maximum likelihood algorithm does not consider either decentralized or privacy-preserving aspects.

- Two research questions to be answered

How do we securely aggregate local statistics?

Can we optimize communication network for faster aggregation?

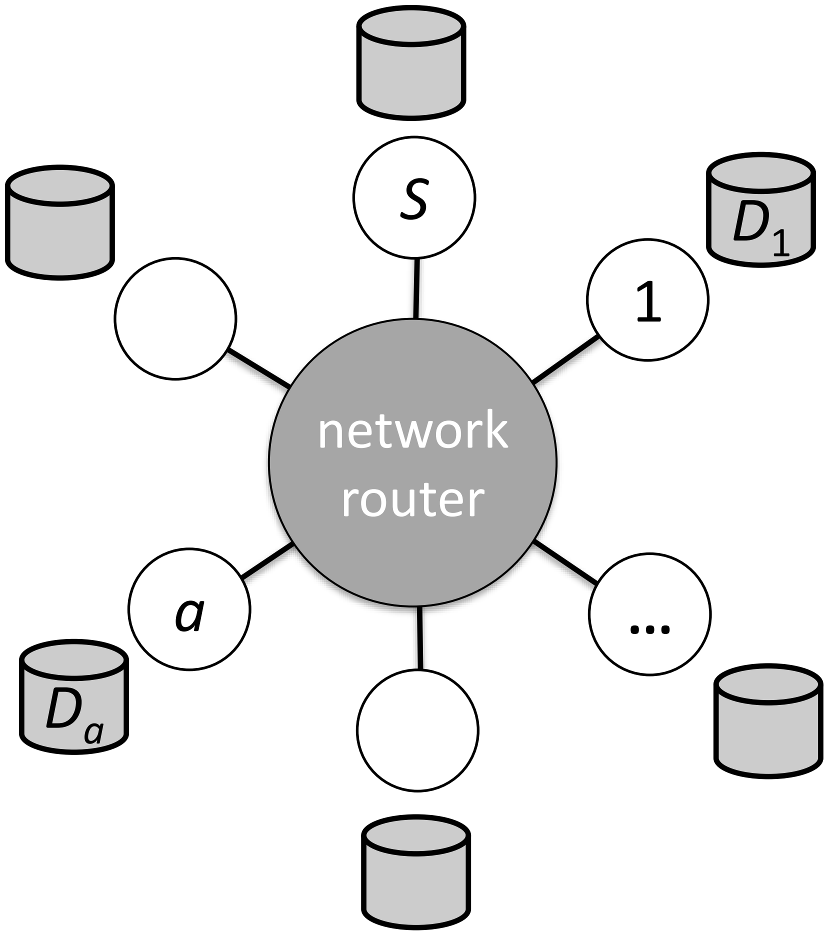
Agenda

- Introduction: Blockchain as value co-creation platform
- Decentralized collaborative learning framework
- Secure decentralized aggregation
- Network topology design
- Summary

Secure aggregation problem

- Problem: Compute summation
  - $\bar{c} = c_1 + c_2 + \ldots + c_S$
  - c_a (a=1, .., S): A statistic (or a datum) computed locally by participant a

- Easy? Not really, when only P2P communications are allowed
  - Broadcasting your data to all?
    - ✓ No! Total privacy breach
  - Select a leader to let her compute?
    - ✓ No! What if she is a bad guy?



Existing privacy-preservation approaches have issues in decentralized setting

- Encryption-based
  - Decentralization is nontrivial
  - Can be serious computational bottleneck
    - ✓ Great for one-time business transactions
    - ✓ Not designed for iterative machine learning algorithms

- "Noise-based" (differential privacy)
  - Typically needs central authority
  - Noise variance blows up in the multi-party setting as a result of aggregation
  - Learning models can be suboptimal due to noise

Our solution to secure aggregation problem

Dynamical consensus + **Secret sharing**

Dynamical consensus

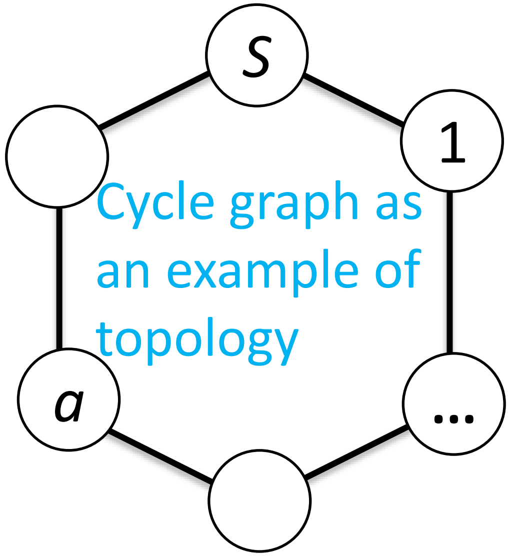
- Repeat P2P communication so a certain Markov transition is performed
- Stationary state of the Markov chain converges to the aggregated value (magical!)

Secret sharing

- Random chunking with probabilistic privacy guarantee
- Securer (but slow) alternative:
  - Shamir's secret sharing combined with dynamical consensus

Dynamical consensus algorithm:
Leveraging Markovian dynamics for aggregation

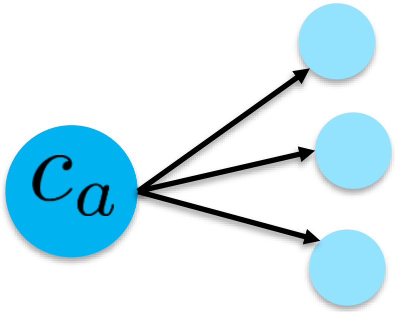
- Algorithm: Each participant repeat an update until convergence
  - $$c_a \leftarrow c_a + \epsilon \sum_{j=1}^{S} \mathsf{A}_{a,j}(c_j - c_a)$$ Communicate only with connected peers
  - **A**: Network topology (= adjacency matrix of the graph)
  - ε: A small positive constant
- Upon convergence, each participant ends up having
  - $$\bar{c} = \sum_{a=1}^{S} c_a = \underbrace{\mathbf{1}}_{S\text{-dimensional vector of ones}}{}^{\top} \boldsymbol{c}$$
- Why? Because the update is the same as multiplying a matrix, whose leading eigenvector is the **1** vector. (→ see the paper)



Cycle graph as an example of topology

Random chunking algorithm:
Applying aggregation to each random split

- Each participant randomly splits their datum into N_c chunks
  - $$\bar{c} = \sum_{a=1}^{S} c_a$$



$$c_a = c_a^{[1]} + c_a^{[2]} + c_a^{[3]}$$

- Do dynamic consensus N_c times and sum
  - $$\bar{c} = \bar{c}^{[1]} + \bar{c}^{[2]} + \bar{c}^{[3]}$$
- Need to shuffle node IDs every time upon starting aggregation
  - This is for a node not to receive all the chunks
  - Security guarantee becomes thus probabilistic

Random chunking algorithm trades off cryptographic security guarantee for computational efficiency

- Shamir's secret sharing (SSS) allows performing aggregation without revealing any raw data

- In random chunking, privacy guarantee is probabilistic. But it is a few orders of magnitude faster than SSS

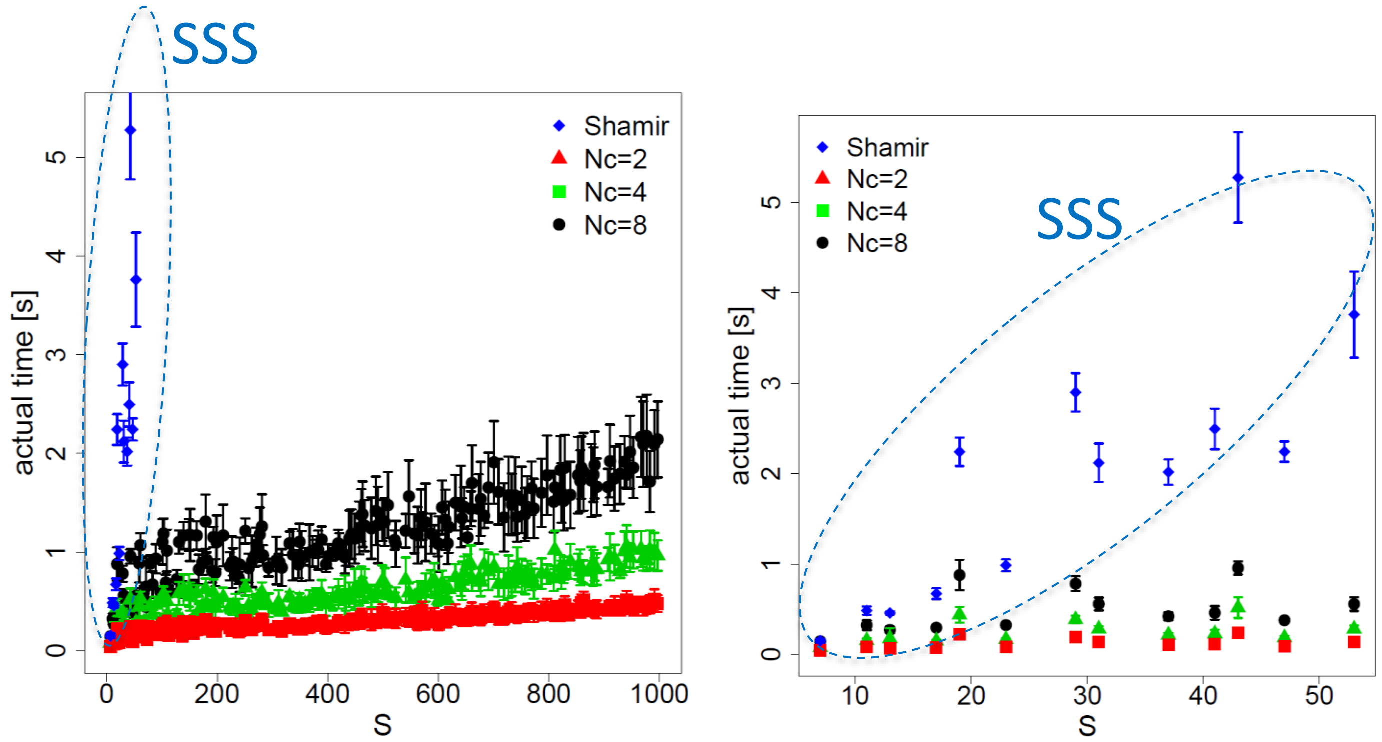


Fig. 6. Actual computation time for aggregation on the 3-regular expander graph. The right panel covers the range of $7 \leq S \leq 53$.

Random chunking algorithm has probabilistic security guarantee (proof → [Ide & Raymond, SMDS 21])

scenario	breach probability bound per node	parameters
independent	$\leq (S-1)\left(\frac{d_a}{S-1}\right)^{N_\mathrm{C}}$	d_a: Node degree of the a-th node N_C: The number of splits
collusion	$\leq \exp\left\{-N_\mathrm{C}\left(1-\frac{d_a}{S-N_\mathrm{L}}\right)^{N_\mathrm{L}}\right\}$	N_L: The number of colluded nodes
eavesdropping	$\leq \exp\left\{-N_\mathrm{C}\left(1-\frac{N_\mathrm{E}}{E-d_a+1}\right)^{d_a}\right\}$	N_E: The number of tapped edges E: The total number of edges of the graph

Tsuyoshi Idé, Rudy Raymond, "Decentralized Collaborative Learning with Probabilistic Data Protection," In Proceedings of the 2021 IEEE International Conference on Smart Data Services (SMDS 21, September 5-10, 2021), pp.234-243.

Agenda

- Introduction: Blockchain as value co-creation platform
- Decentralized collaborative learning framework
- Secure decentralized aggregation
- Network topology design
- Summary

Spectral stricture of W_ϵ governs convergence speed in dynamical consensus

- The dynamical consensus algorithm can be viewed as repeated multiplication of a matrix $\mathbf{W}_\epsilon \equiv \mathbf{I} - \epsilon(\mathbf{D} - \mathbf{A})$
  - ○ **A**: adjacency matrix; **D**: degree matrix
  - ○ **D – A** is known as the graph Laplacian

dynamical consensus update

$$c_a \leftarrow c_a + \epsilon \sum_{j=1}^{S} \mathsf{A}_{a,j}(c_j - c_a)$$

- The spectral structure of $\mathbf{W}_\varepsilon$ governs convergence speed
  - ○ Critical quantity is the “spectral gap”: $\lambda_1 - \lambda_2$
    - ✓ The difference between the 1st and the 2nd largest eigenvalues of $\mathbf{W}_\varepsilon$

- <u>Question</u>: How do I choose the topology, so the spectral gap is as large as possible while keeping the probability of privacy breach low?

Deep mathematical result in graph theory helps find a good compromise between privacy and convergence speed

- The topology should be
  - as sparse as possible for privacy protection
  - as dense as possible for faster convergence
- A class of graphs called the *expander graph* is an ideal compromise
  - Known as a sparse approximation of the complete graph
- Remarkable property of the expander graph
  - By Cheeger’s inequality, we have

$$\Delta_\lambda \triangleq \lambda_1 - \lambda_2 \geq \epsilon \frac{\alpha^2}{2d} \quad (d\text{-regular expander graphs})$$

- α: lower bound of a quantity called the expansion coefficient
- δ: relative error allowed

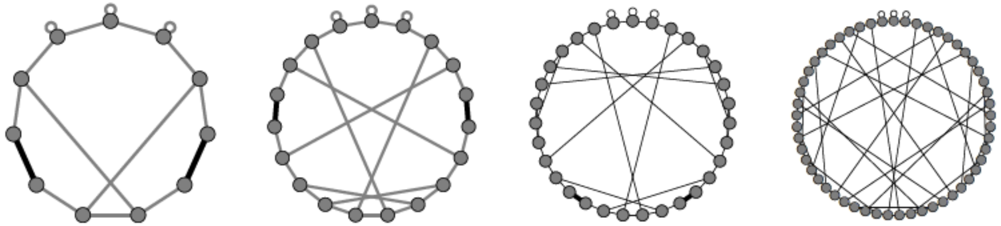
from which we can evaluate the number of iterations as

$$t \sim \mathrm{O}\left(\frac{\ln(\sqrt{S}/\delta)}{|\ln(1-\Delta_\lambda)|}\right)$$

logarithmic convergence w.r.t. # participants

Expander graph drastically improves convergence speed for aggregation.

- "Cycle with inverse chord" is a known instance of expander graph.
  - Cycle graph + some edges



- Speedup is drastic: expander vs. cycle
  - S is the number of network participants.
  - Expander: comp. time $\sim \log S$
  - Cycle: $\sim S^2$

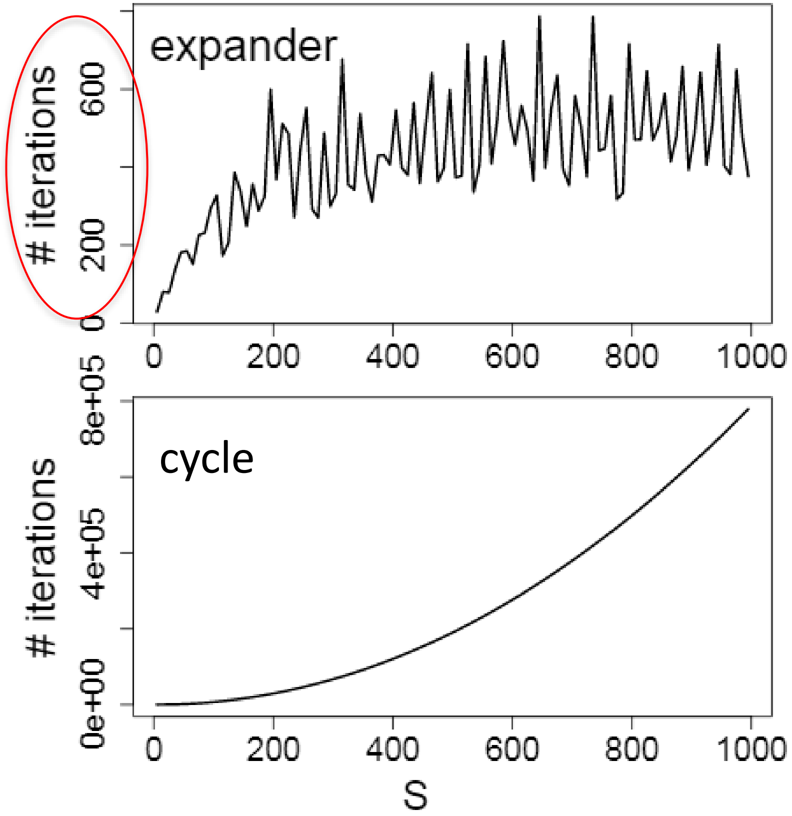


Fig. 5. Comparison of the number of iterations t for $\delta = 10^{-3}$.

Agenda

- Introduction: Blockchain as value co-creation platform
- Decentralized collaborative learning framework
- Secure decentralized aggregation
- Network topology design
- Summary

Summary

- Decentralized collaborative learning is a generalization of the original concept of blockchain.

- It can be formalized as multi-task learning under decentralization and privacy constraint.

- Random chucking can be a practical alternative to slow cryptographic algorithms

- Expander graph as network topology achieves drastic speed up in secure aggregation.

Future research topics

- Learning under network failures
  - The current model assumes perfect synchronization. Evaluating robustness under network failure and extending the algorithm to handle asynchronous communication is important.
- Meta-agreement issues
  - In addition to computed numerical statistics, there are several things that require participants’ consensus
    - ✓ Choice of the algorithm, dimensionality, topology, etc.
- External data privacy
  - We focused on privacy guarantees among network participants. Evaluation of privacy leakage when, e.g., externally selling the learned model is an open question.
- Randomness in graph spectra
  - The expander graph provides an excellent convergence rate in dynamical consensus, but it introduces some unpredictability in the graph spectra.
- Security analysis
  - The random chunking algorithm combined with the dynamic consensus algorithm appears to have more flexibility than traditional cryptographic methods. We need to study further the pros and cons of those methods.
- Use-cases
  - Finally, we need to develop practical use-cases where the decentralized architecture is truly useful. The lightweight probabilistic privacy guarantee seems suitable in IoT applications, but more study is needed.

Thank you!