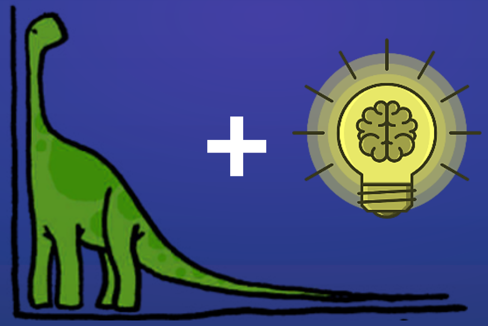
Wireless Network
INTELLIGENCE @ the *EDGE*



Mehdi Bennis
Associate Professor
Head of Intelligent Connectivity and Networks/Systems
Centre for Wireless Communications
University of Oulu, Finland

5G: Evolution or Revolution?

- The evolutionary part of 5G (eMBB) has made great strides focusing primarily on the use of high frequency bands + Numerology.
  - 5G standardization has been going full steam with the 1st 5G new radio (NR) milestone for non-standalone and subsequent releases.
- Fundamentals of ultra-reliable and low-latency communication (URLLC) at the network level (catalyst of the 5G **revolution**) are not well understood.
  - 3GPP focused on radio instead of a E2E codesign of sensing/communication/controlling/computing/actuating

- Networks getting very complex to manage
- Emergence of new breed of devices and high-stake applications driven by Robotics & Autonomous Systems, human-machine/brain-machine interaction, multi-sensory AI, 3D Imaging [+ unforeseen applications..]

→Edge intelligence key for unlocking full potential of 5G+!

ML/AI Changing Our Lives ...@ *What Cost?*

Good News

Today's AI successfully *recognizes faces*, *diagnoses diseases*, *predicts rainfall*, consumer preferences + much more.

- Deep NN are SOTA for ML tasks and *revolutionized* our lives
  - *Thanks to more data and compute power*

Bad News

Modern NN architectures are compute, space and power-hungry.

- Cloud-Run: *Computationally intensive* → difficult to deploy on embedded devices with limited hardware resources + tight power budget
- Centralized + Offline training
- Do not reliably quantify prediction confidence
- Easy to fool changing slightly the input (GANs) -- adversarial examples
- No privacy guarantees
- Dominant paradigm: Dumb devices w/ *always-on cloud-connectivity*

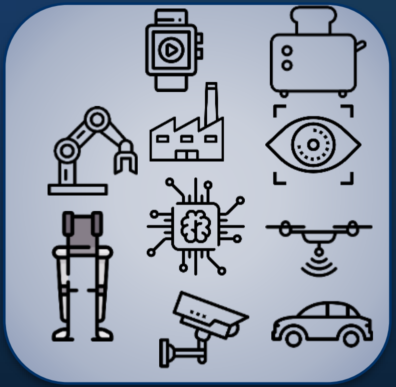
Machine translation

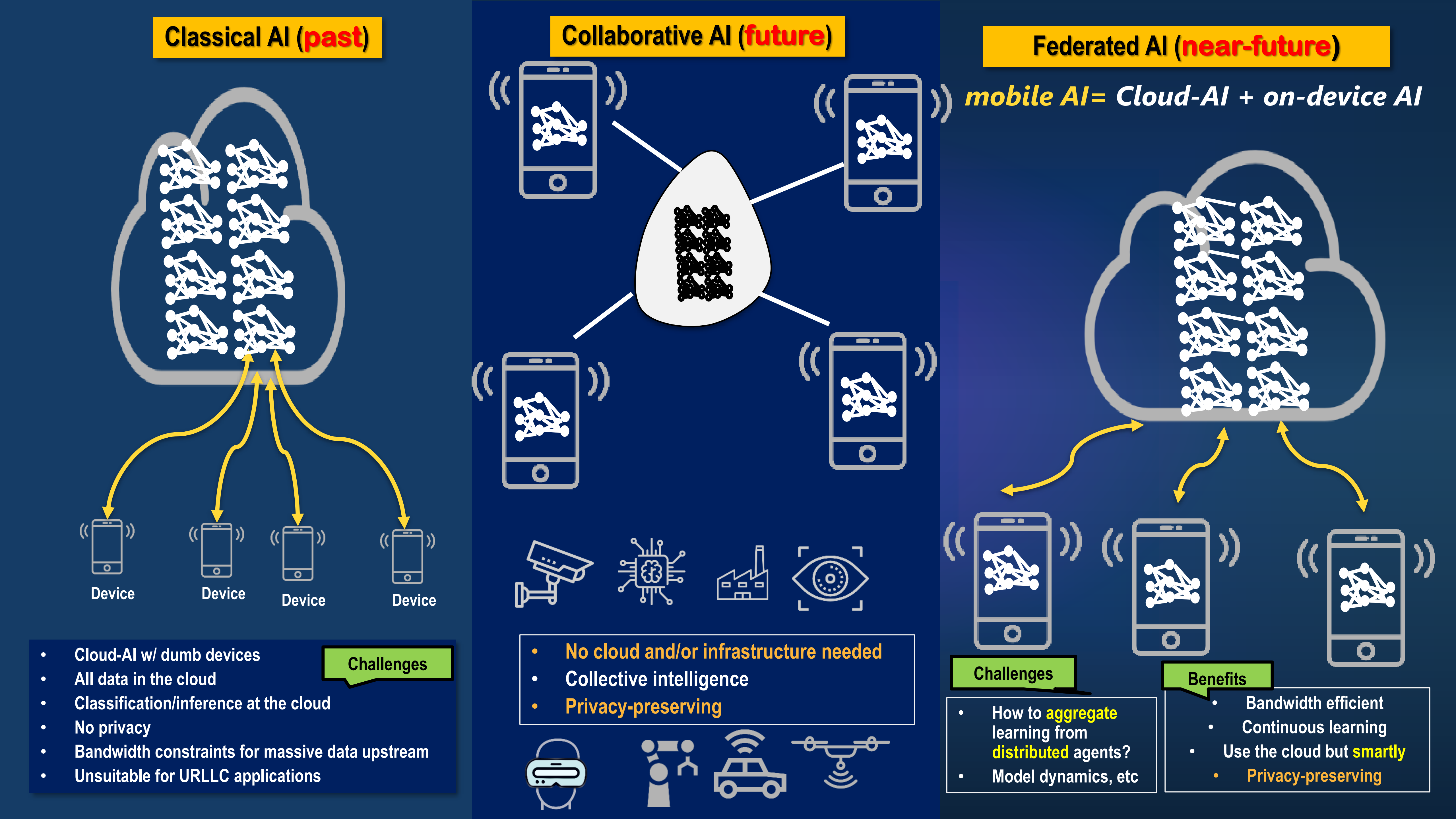
Face Recognition

AlphaGo

Medical Diagnosis

Unfit for the new breed of intelligent devices & high-stake applications



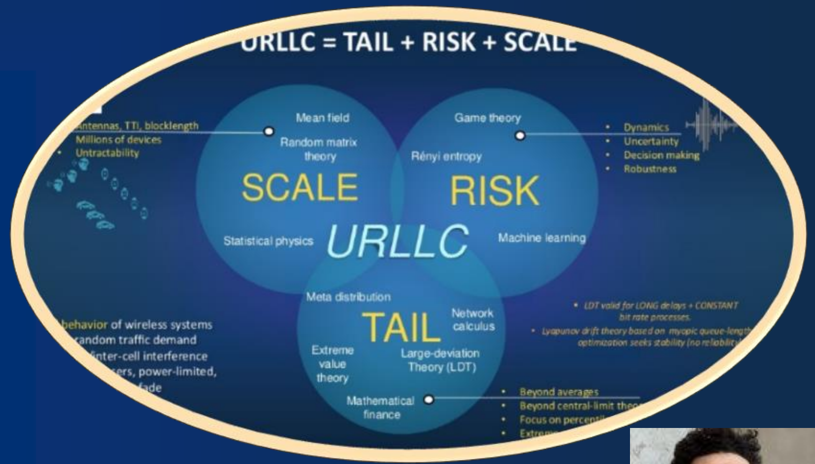

Classical AI (past)
Device
Device
Device
Device
Cloud-AI w/ dumb devices
All data in the cloud
Classification/inference at the cloud
No privacy
Bandwidth constraints for massive data upstream
Unsuitable for URLLC applications
Challenges
Collaborative AI (future)
No cloud and/or infrastructure needed
Collective intelligence
Privacy-preserving
Federated AI (near-future)
mobile AI= Cloud-AI + on-device AI
Challenges
How to aggregate learning from distributed agents?
Model dynamics, etc
Benefits
Bandwidth efficient
Continuous learning
Use the cloud but smartly
Privacy-preserving

URLLC Meets AI

ENABLERS
- Short TTI
- Caching
- Densification
- Grant-free + NOMA
- UAV/UAS
- MEC/FOG/MIST
- Network Coding
- **On-device ML**
- Slicing

ENABLERS
- Short TTI
- Spatial diversity
- Network Coding
- Caching, MEC
- Multi-connectivity
- Grant-free + NOMA
- **On-device machine learning**
- Slicing

ENABLERS
- Finite Blocklength
- Packet duplication
- HARQ
- Multi-connectivity
- Slicing
- Network Coding
- Spatial diversity
- Slicing



ML for Communication (ML4C) – Current Focus

- ML @ Physical layer
  - Accurate knowledge of RF environment (channel effect, propagation models, fault monitoring)
  - Optimized use of RF environment (improved MCS, resource scheduling, spatial encoding schemes for MU-MIMO, reduced power consumption)
  - Channel detection and decoding (data-driven useful for non well-established channel models)
  - Learn how to cancel FD self-interference
- ML @ network and application layer
  - Resource slicing, caching popular contents, routing, etc
  - Traffic classification
  - Spectrum sensing (generate new examples to augment a dataset to train a classifier)
  - Community detection

Mostly **data-driven** + **centralized** + **blackbox** based solutions

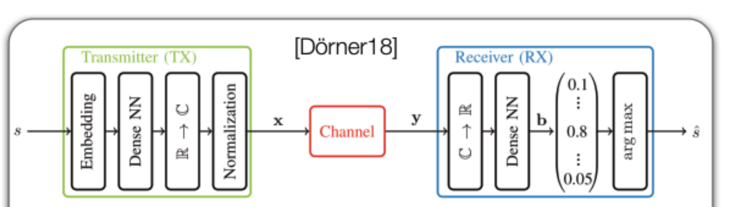
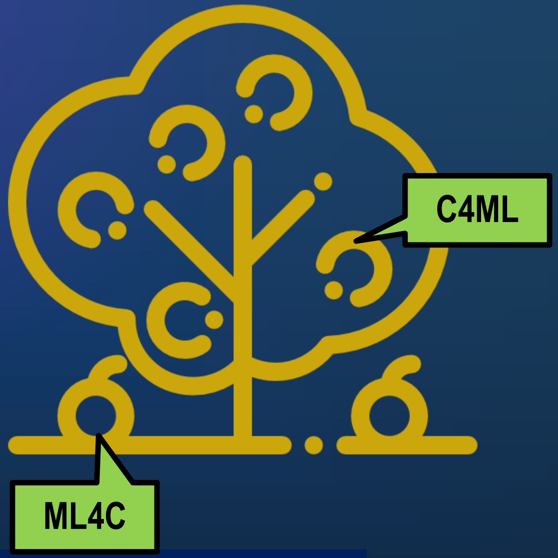


Fig. 2: Representation of a communications system as an autoencoder



What about C4ML? → Distributed Edge Intelligence

Wireles Edge Intelligence

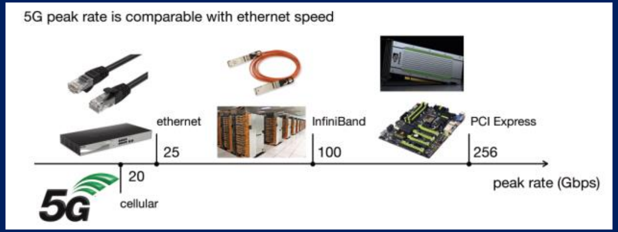
- Edge intelligence (EI) is a nascent research field which requires a major departure from centralized cloud-based *training/inference/control* approaches
- *Towards a system design* where edge devices communicate and exchange their learned models (not their private/raw data) to build a centralized trained model—subject to:
  - latency
  - reliability
  - privacy
  - Memory/compute/power constraints
  - accuracy.

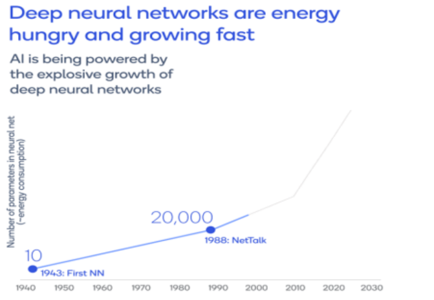
ML+Wireless codesign needed across full stack from application through hardware + improved efficiency of NN *yielding*:

- Latency reduction via local inference
- Bandwidth efficiency
- Immediacy
- Privacy-preserving

Communication Bottleneck

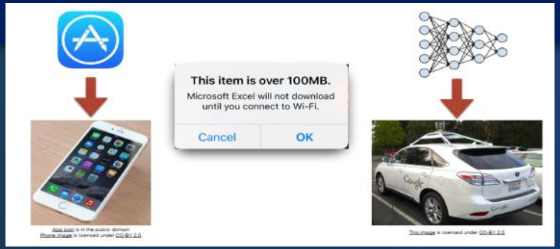
- 5G peak rate comparable with Ethernet speed
- Synchronization latency due to network connectivity, power and computing constraints

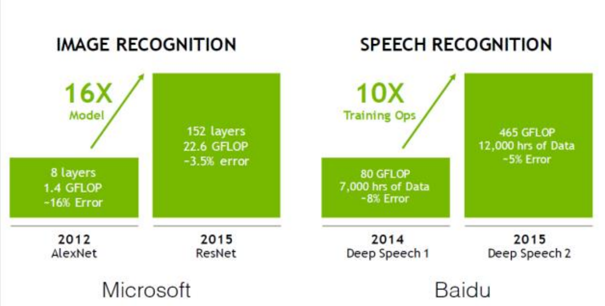




Model Size

- Hard to distribute large models through OTA
- Smaller model improve inference speed, require less arithmetic operations and computation cycles + less memory reference cycles





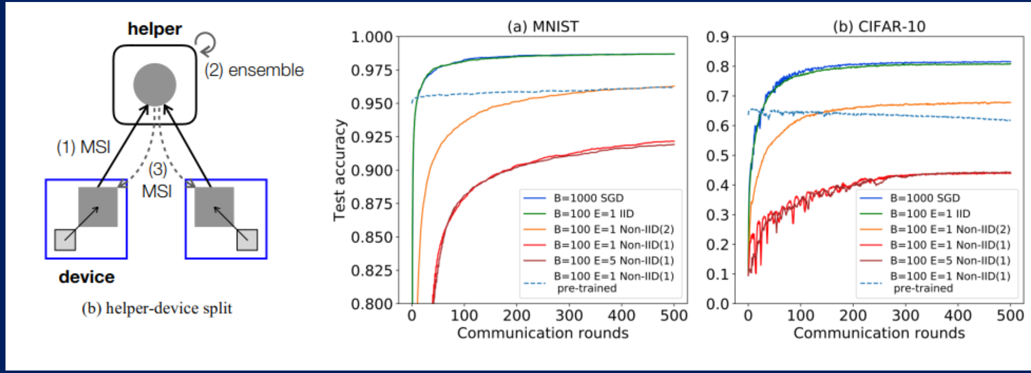
Energy Efficiency

- AlphaGo: 1920 CPUs + 280 GPUs, $3000 electric bill per game
- Running large NNs require significant memory bandwidth to fetch the weights

larger model => more memory reference => more energy

Statistical Challenge: Non-IID Data

- Unrealistic to assume local data on each edge device is always IID
- Non-IID training dataset degrades accuracy [Zhao18] due to weight divergence
  - Distribute small amount of globally shared data
  - Accuracy vs. centralization tradeoff



[Zhao18] Y. Zhao et al., "Federated Learning with Non-IID Data," arXiv preprint

Inference Speed

- Many applications require low-latency, realtime inference such as self-driving vehicles and AR glasses
- Very long training time limits ML researcher's productivity

	Error rate	Training time
ResNet18:	10.76%	2.5 days
ResNet50:	7.02%	5 days
ResNet101:	6.21%	1 week
ResNet152:	6.16%	1.5 weeks

Challenges & Requirements

Non-stationary systems & parameters

Compute-memory-power hungry

Prediction Confidence

Privacy

Limited Data

Data Split

Model Size

Model split

Offline Training

Uncertainty

Easy to fool

Intermittent connectivity

Non-convex loss function

Centralized Training

Training Time

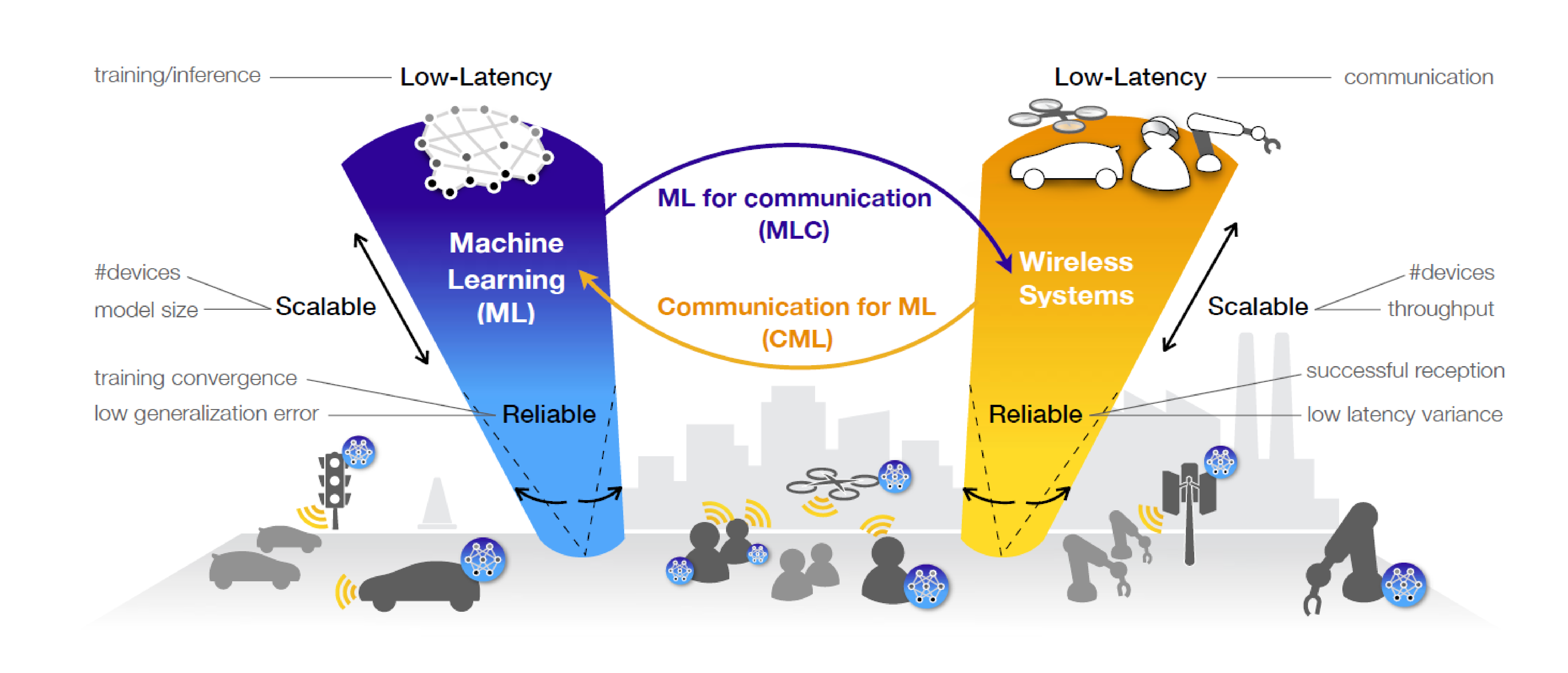
+ Robustness to

- known unknowns
- unknown unknowns

Requirements

- Accuracy
- Latency
- Energy
- Reliability
- Scalability
- Capacity
- Sample Complexity

Our Vision



"Wireless Network Intelligence at the Edge" https://arxiv.org/pdf/1812.02858.pdf

<u>Some</u> Fundamental Questions

Q1. How do resource-constrained devices *collectively train* a high-quality centralized model in a decentralized manner? for **different master/slave and slave-slave architectures**.

Q2. How do learners utilize <u>correlation information</u> balancing computational complexity, communication cost and prediction accuracy? How to learn while **personalizing** to each user/task?

Q3. How do learners address the notorious challenge of non-convexity in neural network training?

- How to enable beyond-average, *reliable* and *low-latency* AI?

Q4. Can we adapt the optimization algorithms to <u>converge faster</u> with less communication rounds?

Q5. How many communication-computing rounds are needed for a given target % accuracy?

  - Computing is cheaper than communication

Q6. What is the impact of <u>no. of learners</u> (participating devices), <u>no. of samples</u> per leaner and no. of <u>local iterations</u>? What is the impact of <u>learning rates</u>, data sample <u>freshness/importance</u>, etc.?

- What information to be shared between nodes based on communication-delay constraints?

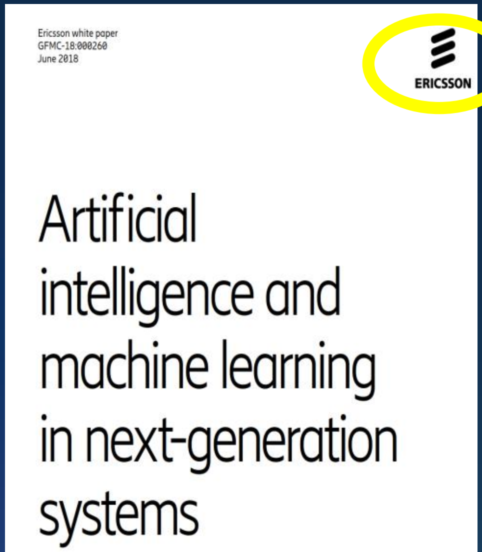
Q7. How to carry out decision making under risk and uncertainty for resource-constrained devices? How to model dynamics, uncertainty (DL ignores <u>uncertainty</u>) → Bayesian DL

+ Neural architectural split & Intelligence split from device-edge-cloud

FL-Wireless Ramifications

An ML model may have million parameters

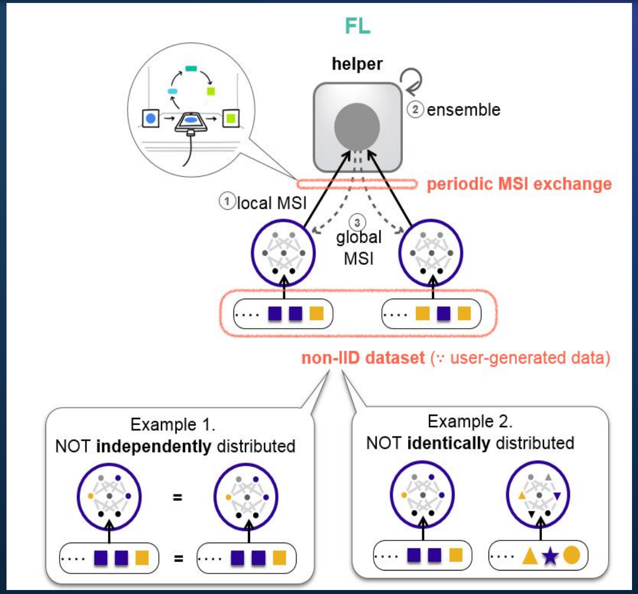
- Model updating is bandwidth consuming especially for 1000X edge devices
- Slowest node or straggler
- Lack of synchronization and asynchronous updates
- Moving nodes
- Noisy/interfered links
- Sample importance/Freshness
- Data quantity vs. quality

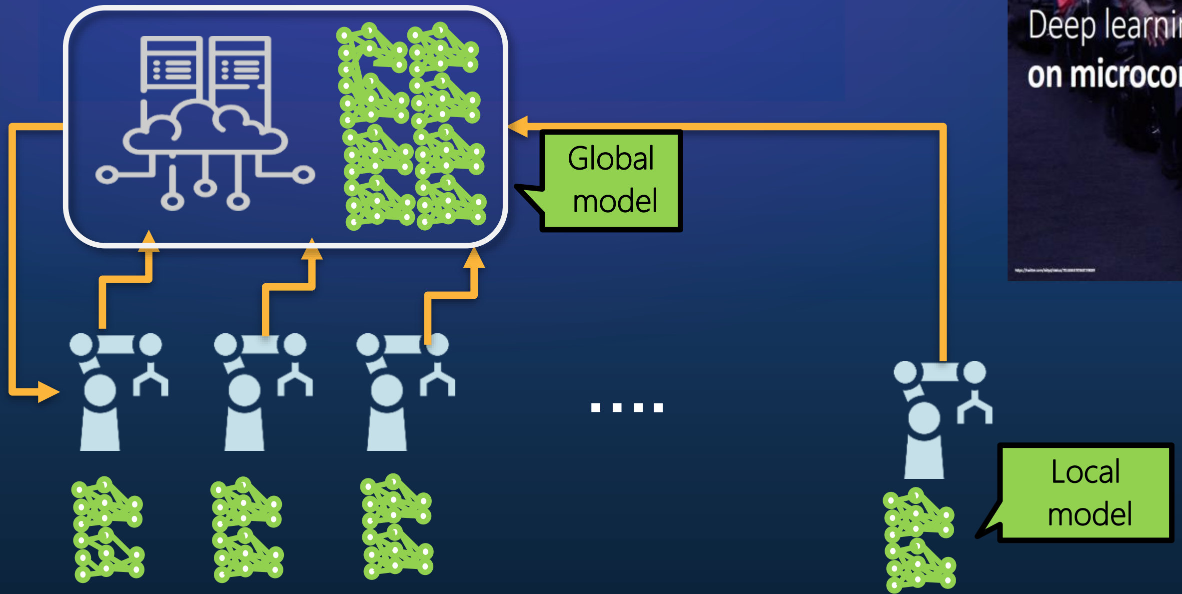


Ericsson white paper
GFMC-18:000260
June 2018

ERICSSON

Artificial intelligence and machine learning in next-generation systems





Federated learning for Reliable V2V

Use case = URLLC-V2X + distributed FL

Challenge: latency distribution is needed!

$$\Pr(q_u(t) \geq q_0) \leq \epsilon$$

Solutions:

- Locally but lack of samples (latency↑)
- Remotely (RSU) but violate latency constraints (reliability↑ but latency↑)
- Synchronous vs. asynchronous UL (latency↑) .

Key Idea.

- Instead of vehicles uploading their data to the cloud/RSU, every vehicle locally uploads its model to RSU

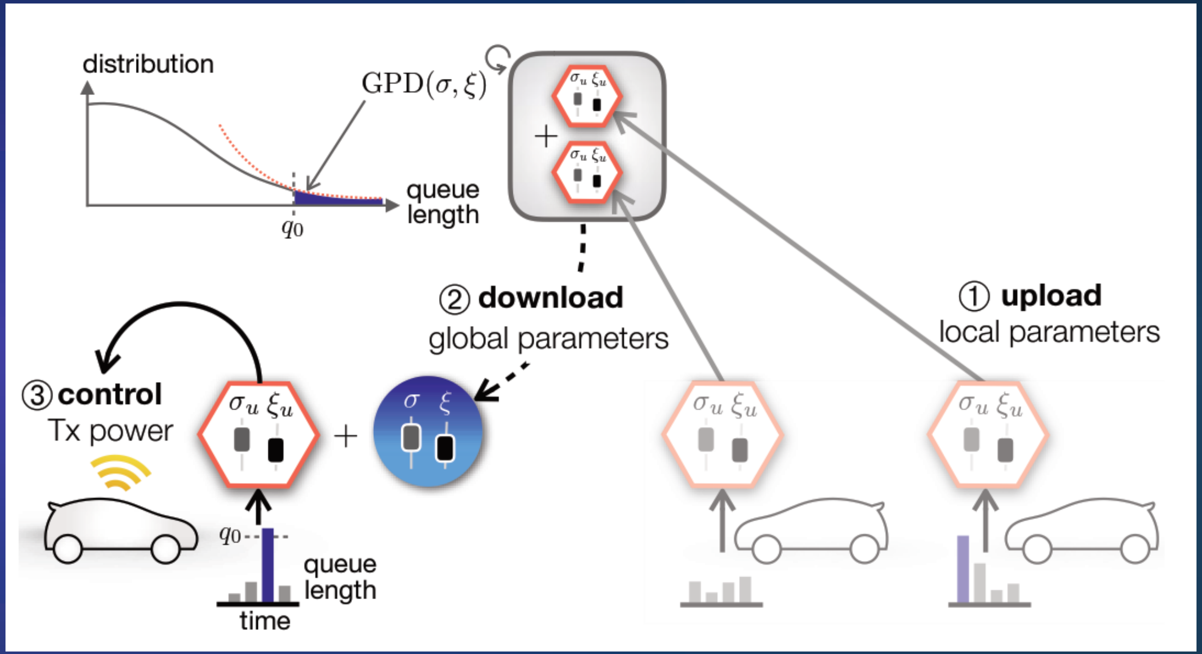
  →**Model-driven ML**

- RSU does model averaging and brodcasts/multicasts to vehicles.

Benefits.

1. FL is a lower latency + Higher reliability enabler ☺
2. Works even during **connectivity loss** ☺

FL: Decentralized training without centralizing training data!



Modeling Extreme Queue Lengths Using Extreme Value Theory

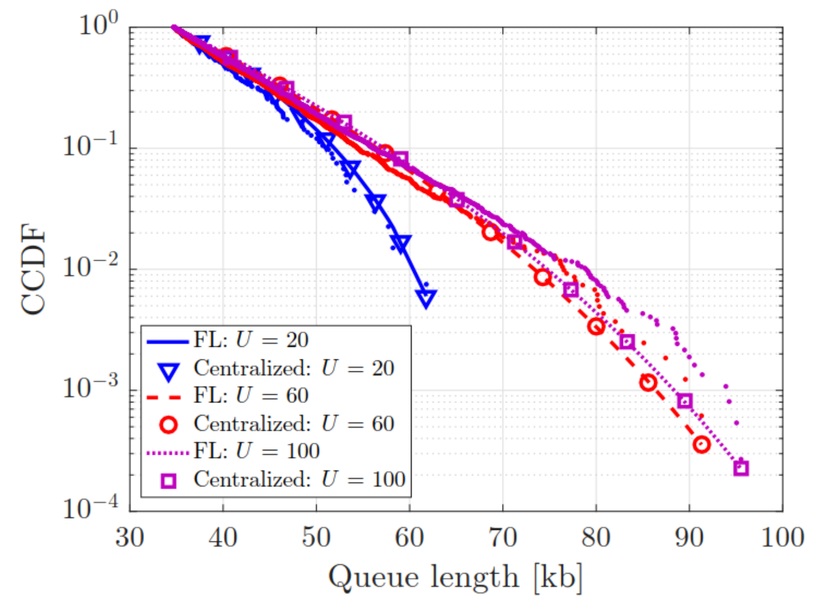
$$G_M^d(m) = \begin{cases} \frac{1}{\sigma}(1+\xi m/\sigma)^{-1-1/\xi} & \text{for } \xi \neq 0, \\ \frac{1}{\sigma}e^{-m/\sigma} & \text{for } \xi = 0, \end{cases}$$

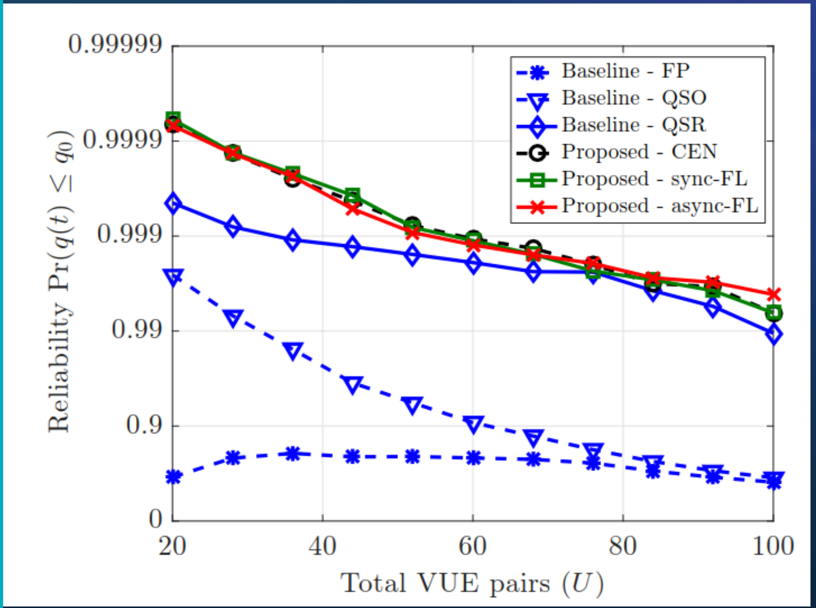
→ Parameter estimation via MLE

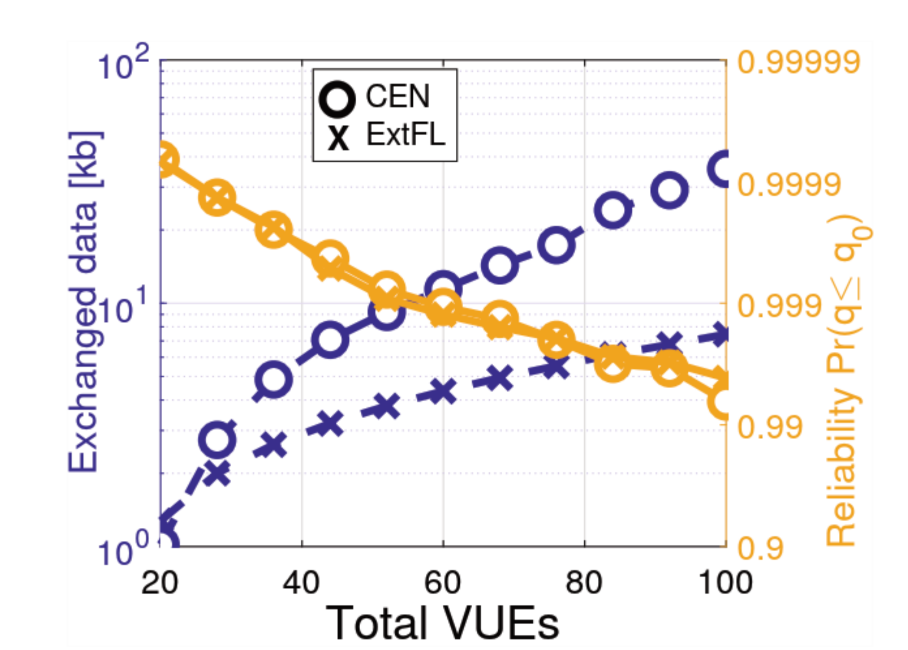
$$\min_{d \in \mathcal{D}(\mathcal{Q})} \quad f^d(\mathcal{Q}) = -\frac{1}{|\mathcal{Q}|}\sum_{Q \in \mathcal{Q}} \log G_X^d(Q),$$

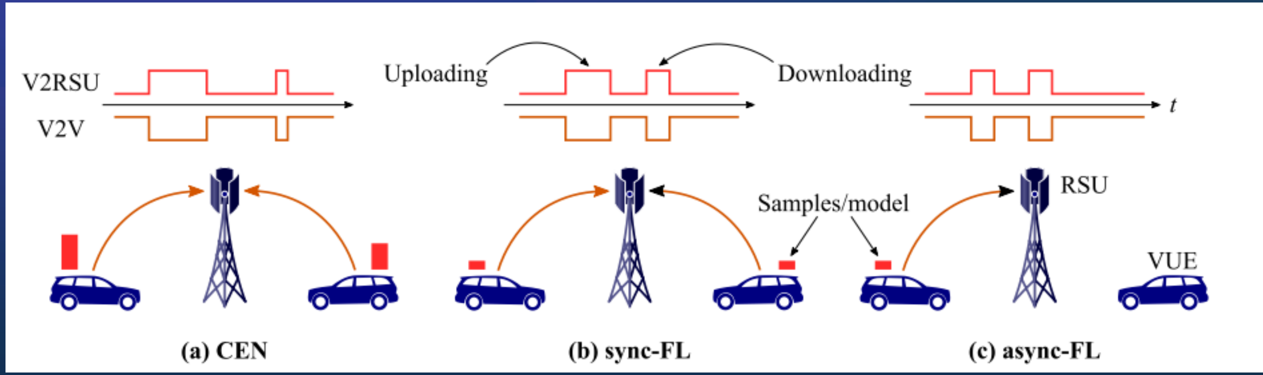
Online+distributed training

Federated learning for Reliable V2V









S. Samarakoon, et al, "**Federated learning for Ultra-reliable low-latency V2V communications**," *in proc. of* IEEE GLOBECOM 2018, Abu-dhabi, UAE.

FL: What is Next?

- Learning global model is great but not sufficient
  - Need to adapt to local dynamics → multi task FL // Transfer learning..
- Model size can quickly become the bottleneck
  - Other ideas needed (e.g. Distillation)
- Beyond Maximum Likelihood to different distances (e.g., Wasserstein)
- Often times, training data becomes outdated
  - Smart sampling of data with a cost
  - Active learning
- From federated learning to federated **control** (e.g. For drones, robots)

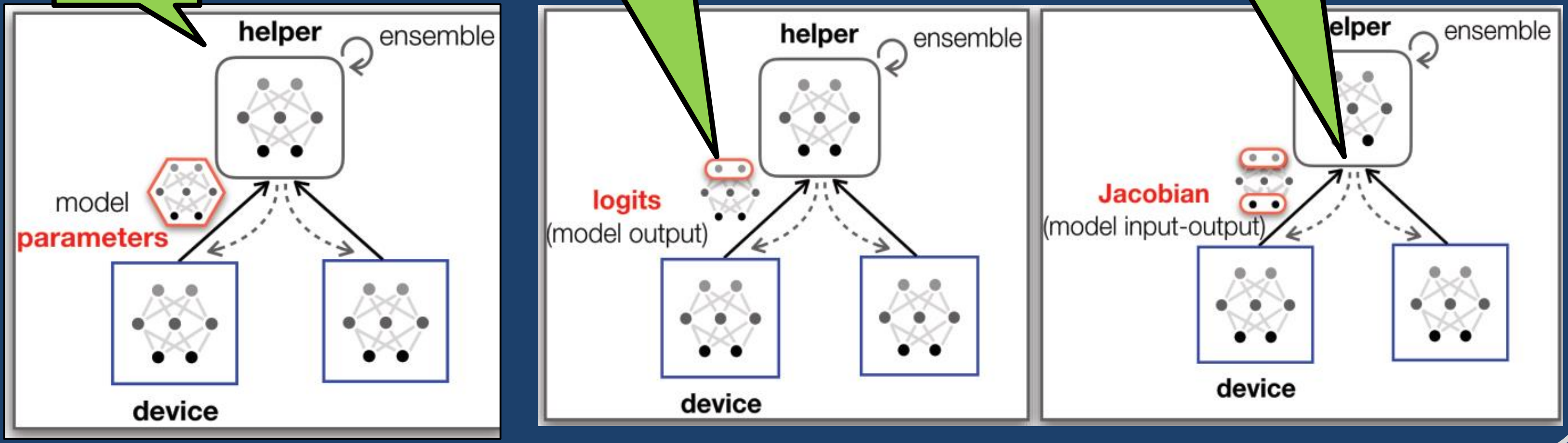
Many Extensions

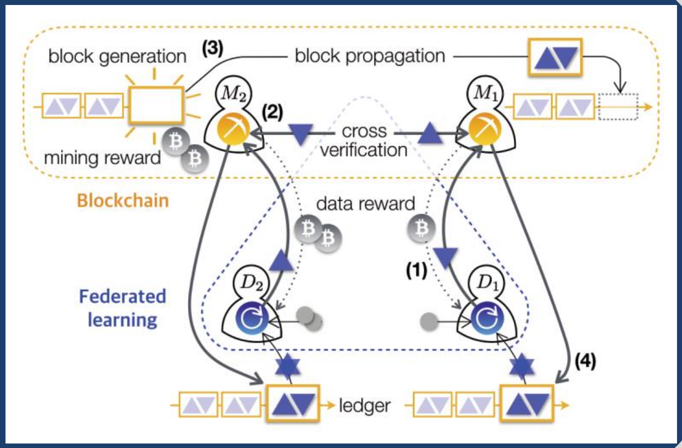
FL

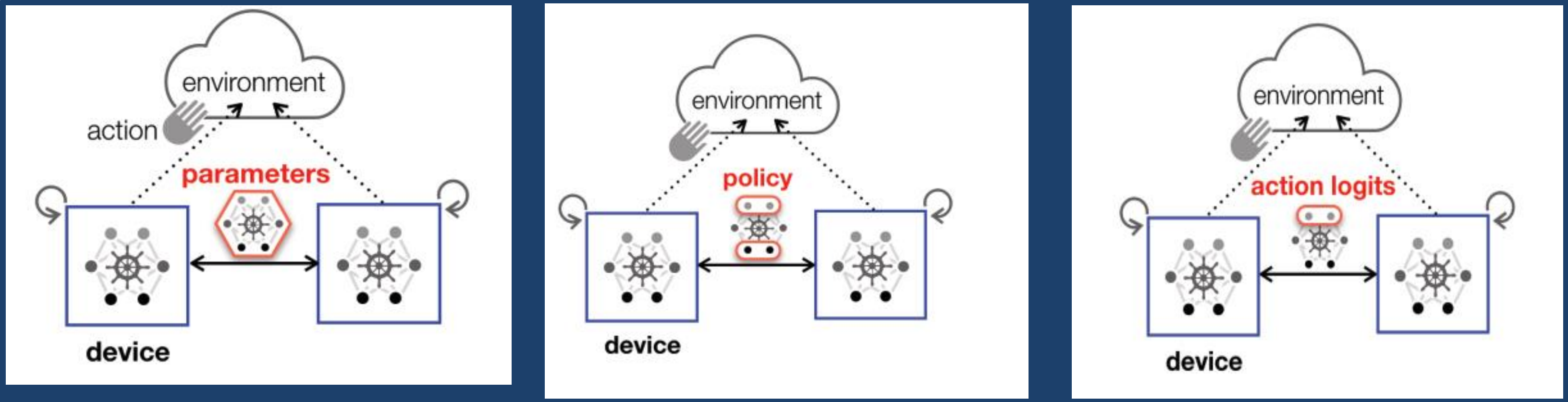
Co-Distillation

Jacobian dist.

How **different** data owners train their models?

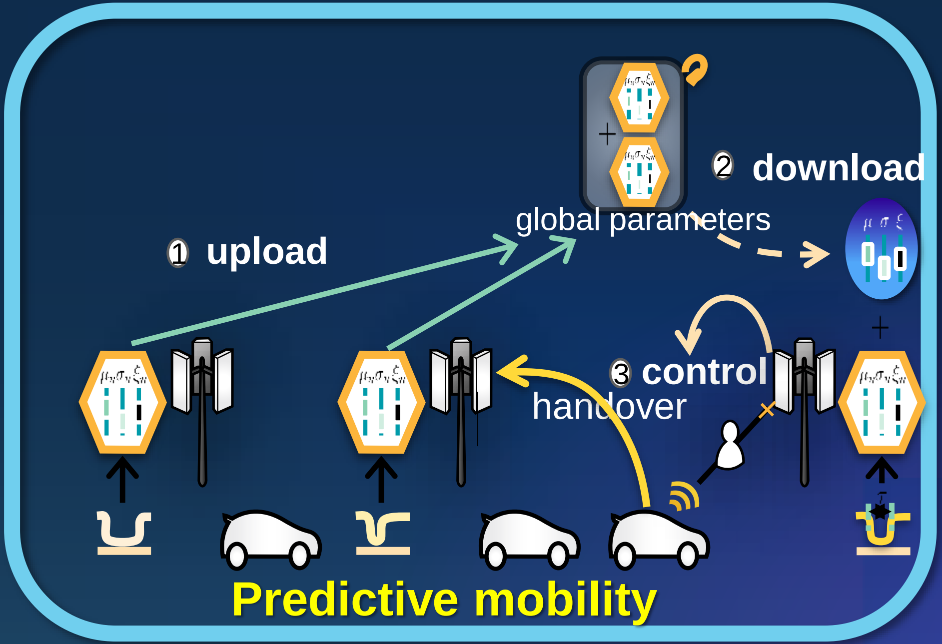


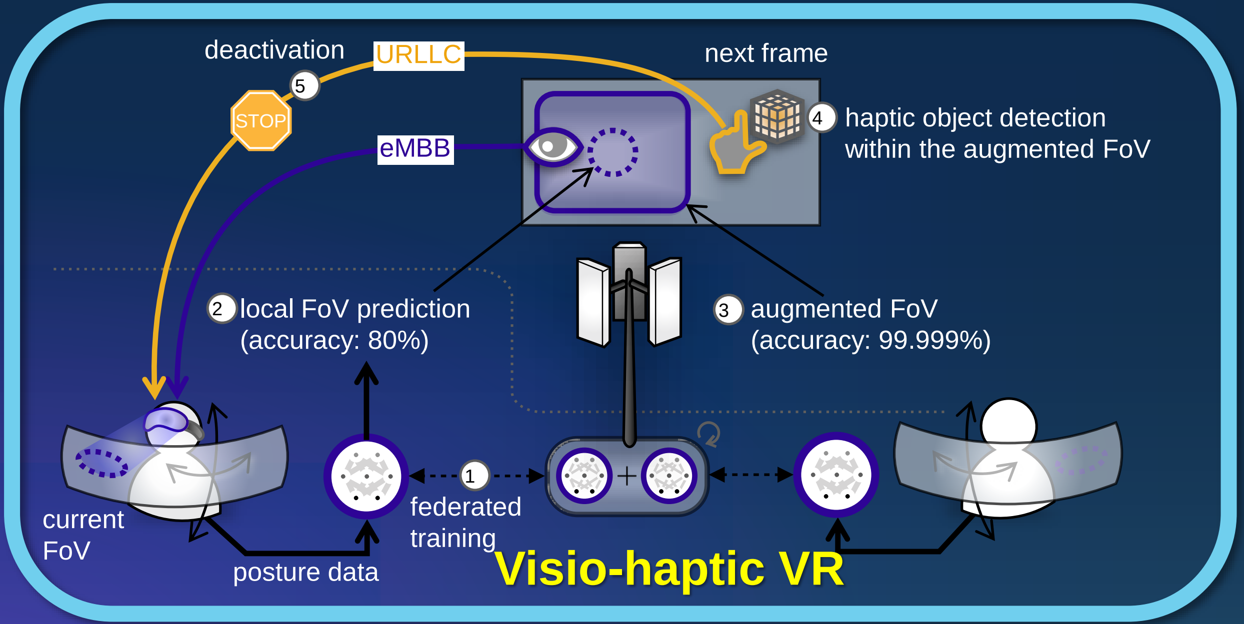


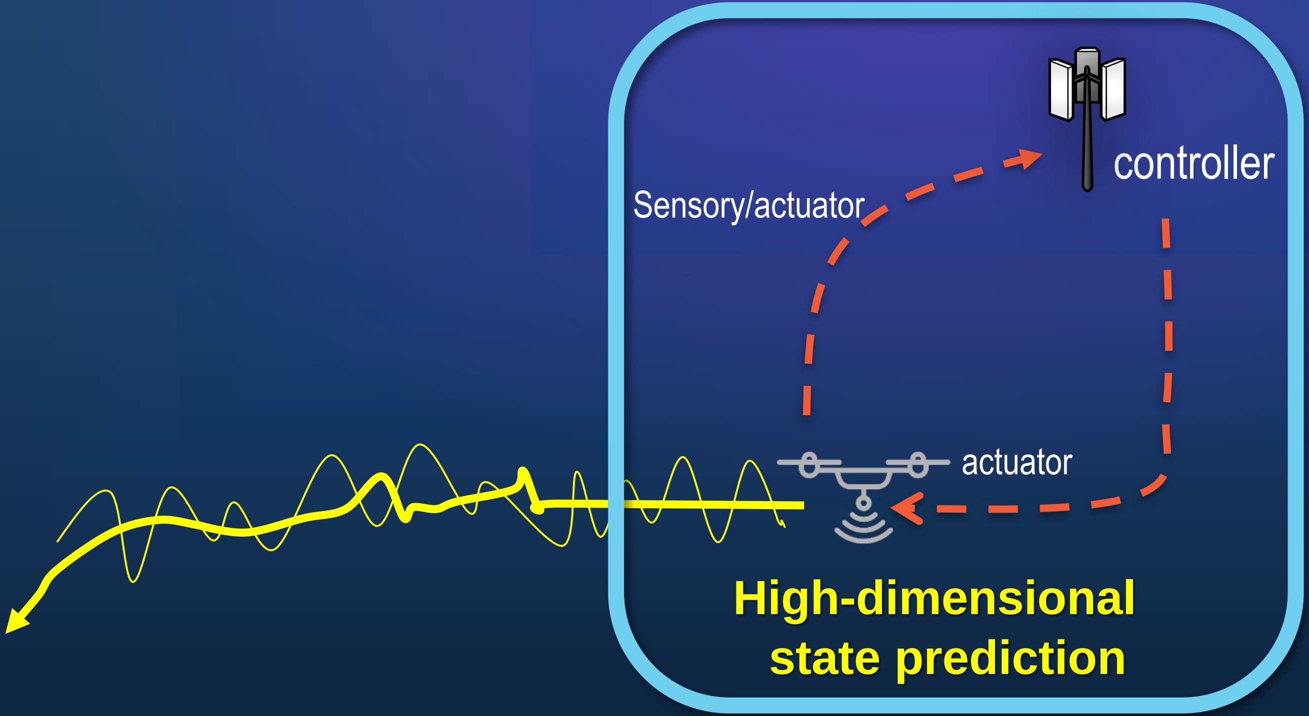


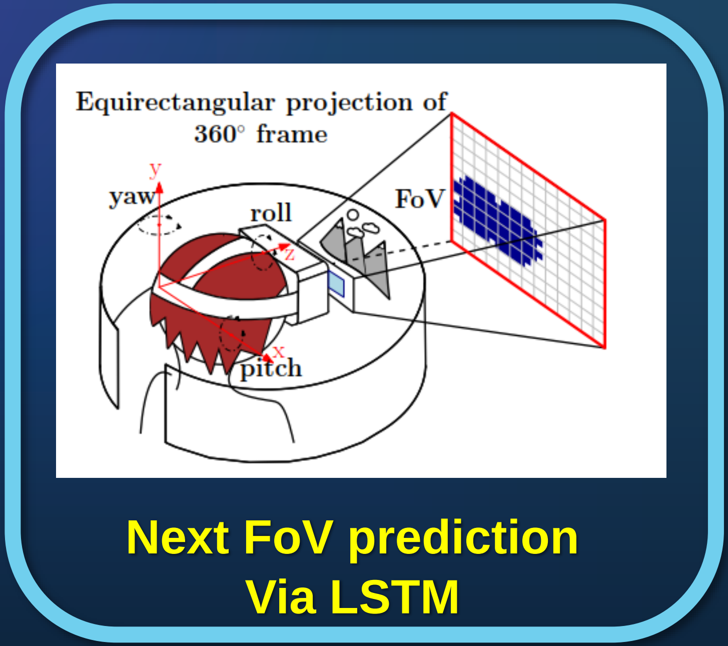
- J. Park, et al. "on-device FL via Blockchain and its Latency Analysis," IEEE Comm. Letter, 2018

- "Federated Distillation and Augmentation under non IID private data," NIPS, Montreal, 2018


1 upload
2 download
global parameters
3 control
handover
Predictive mobility

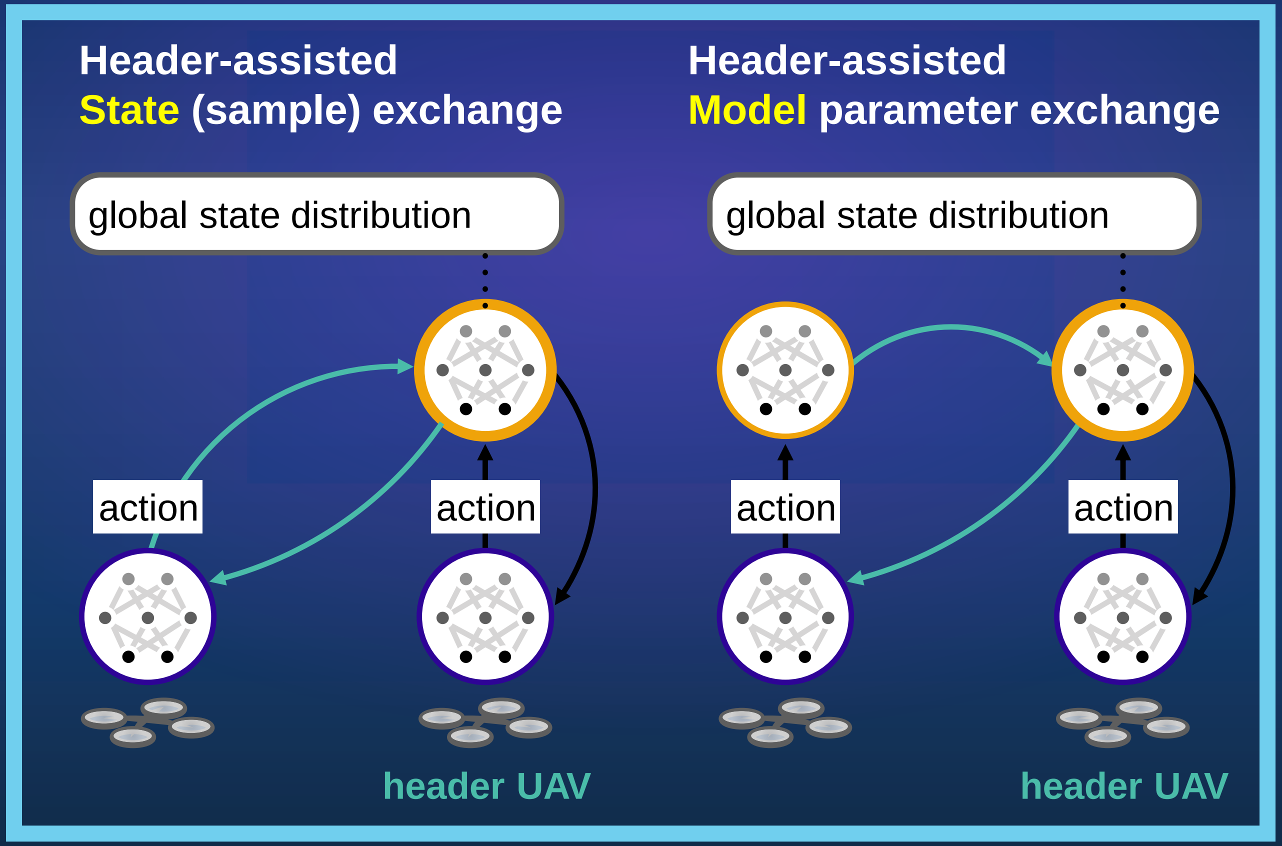

deactivation
URLLC
next frame
5
STOP
eMBB
4 haptic object detection
within the augmented FoV
2 local FoV prediction
(accuracy: 80%)
3 augmented FoV
(accuracy: 99.999%)
current
FoV
1
federated
training
posture data
Visio-haptic VR


controller
Sensory/actuator
actuator
High-dimensional
state prediction


Equirectangular projection of
360° frame
yaw
y
roll
FoV
z
x
pitch
Next FoV prediction
Via LSTM

From data-driven **communication** to data-driven **control**

URLL”C=control” over Wireless



Parting Comments

- Distributed edge intelligence will unlock full potential of 5G (and beyond)
- Lots remain to be studied at many levels and across many domains:
  - Architectural (data split, model split)
  - Algorithmic, mathematical tools needed (Back to school)
  - Hardware (codesign needed)

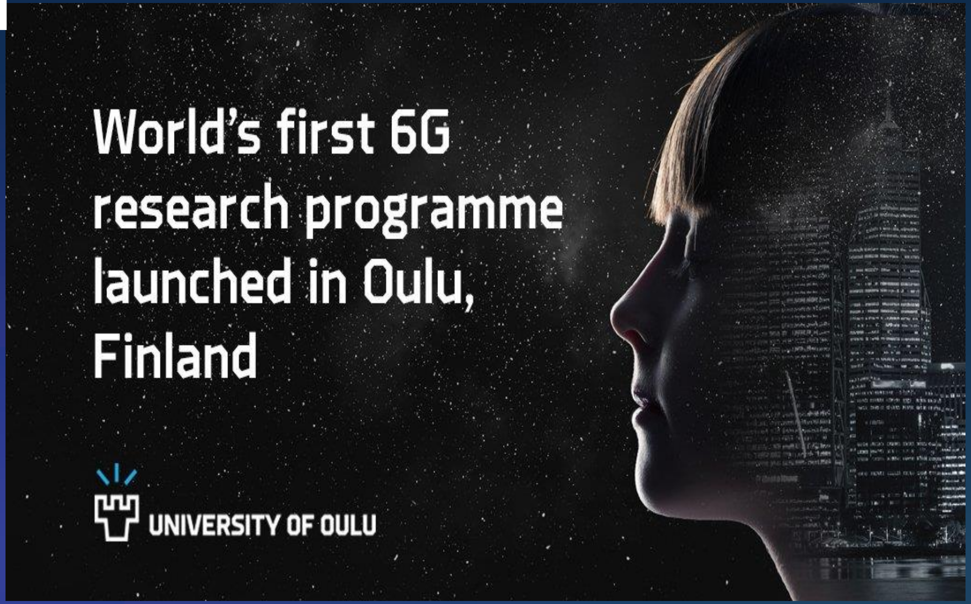
→ Quest for Robust & Mission-critical AI

"Nowhere close to true intelligence"

Call for Collaboration

- URLLC 2.0
- ML/AI

6G
6G WIRELESS SUMMIT
Levi • Lapland • Finland
24-26 March 2019



Thank You

NOKIA Bell Labs

UNIVERSITY OF OULU

ACADEMY OF FINLAND

https://sites.google.com/view/dr-mehdi-bennis/research