Toward Energy-Efficient Federated Learning over 5G+ Mobile Devices

Dian Shi, Liang Li, Rui Chen, Pavana Prakash, Miao Pan, and Yuguang Fang

Abstract

The continuous convergence of machine learning algorithms, 5G and beyond (5G+) wireless communications, and artificial intelligence (AI) hardware implementation hastens the birth of federated learning (FL) over 5G+ mobile devices, which pushes AI functions to mobile devices and initiates a new era of on-device AI applications. Despite the remarkable progress made in FL, huge energy consumption is one of the most significant obstacles restricting the development of FL over battery-constrained 5G+ mobile devices. To address this issue, in this article, we investigate how to develop energy- efficient FL over 5G+ mobile devices by making a tradeoff between energy consumption for "working" (i.e., local computing) and that for "talking" (i.e., wireless communications) in order to boost the overall energy efficiency. Specifically, we first examine energy consumption models for GPU computation and wireless transmissions. Then, we overview the state of the art of integrating FL procedure with energy-efficient learning techniques (gradient sparsification, weight quantization, pruning, etc.). Finally, we present several potential future research directions for FL over 5G+ mobile devices from the perspective of energy efficiency.

Introduction

Machine learning (ML), particularly deep learning (DL), is one of the most disruptive technologies the world has witnessed in the last few years. Unfortunately, cloud-centric ML generates tremendous traffic and also causes serious privacy concerns, which is not suitable for many resource-constrained applications. In order to scale and move beyond cloud-centric ML, Google has introduced federated learning (FL), the currently popular distributed ML paradigm, which aims to enable mobile devices to collaboratively learn a joint global ML model without sharing their privacy-sensitive raw data [1]. With FL, distributed data stakeholders (e.g., mobile devices) only need to periodically upload their updated local models to the aggregation server for global updates instead of uploading their potentially private raw data, thus significantly lowering the risk of privacy leakage. However, stakeholders in many IoT applications, like smart devices, are resource-constrained in terms of spectrum, energy, computing, and storage, which makes FL for such on-device applications highly challenging. Recent successes in 5G and beyond (5G+) technology [2, 3] can further facilitate the implementation of FL over mobile devices. First, due to the advance of hardware design, 5G+ mobile devices are usually armed with ever increasingly high-performance computation units, such as the central processing units (CPUs) and graphics processing units (GPUs), which enable them to host computation-intensive learning tasks. Furthermore, the 5G standard has also embraced computing capability, such as multi-access edge computing (MEC), paving the way for performing computation for edge intelligence, and hence building an effective wireless network architecture to support viable FL. Moreover, 5G+ wireless transmissions feature a very high data rate and ultra-low latency, which can be leveraged to tackle the communication bottleneck issue for local model updates during training. Such a combination of 5G+ and FL prompts tremendous successful applications over 5G mobile devices, including keyboard prediction [4], cardiac event prediction, financial risk management, and so on.

While deploying FL over 5G+ mobile devices is promising in having so many interesting applications, FL and 5G+ mobile devices cannot be easily married for fruitful use without friction. Severe challenges are foreseeable, of which energy consumption is the dominant concern. On one hand, executing on-device computing and performing local model updates are both resource-hungry, inducing a significant surge of energy consumption on mobile devices and hence draining significant battery power. Thus, the first mountain we have to climb is to improve the system energy efficiency in order to prolong the lifetime of mobile devices during training, where the energy consumption usually comes from both the local computing and wireless communications. On the other hand, for FL over 5G+ mobile devices, there will be a trade-off between computing and communication over resource-constrained mobile devices. This stems from our observation on the comparable energy consumption for on-device training with high-performance processors and wireless transmissions with advanced communication techniques. For example, to transmit a ResNet-50 model, a commonly used deep network

Dian Shi, Rui Chen, Pavana Prakash, and Miao Pan are with the University of Houston; Liang Li is with Beijing University of Posts and Telecommunications; Yuguang Fang is with the University of Florida.

Digital Object Identifier: 10.1109/MWC.003.2100028

1536-1284/22/$25.00 © 2022 IEEE

for image classification, with approximately 100 MB parameters via 100 Mb/s 5G wireless uplinks typically consumes 30J for Industrial Internet of Things (IIoT) devices [5]. This is comparable to the energy consumption for performing a single-step local training on one GPU (e.g., 30J for NVIDIA Tesla V100 on the ImageNet dataset [6]). Therefore, how to make a trade-off between computing and communications in order to accommodate realistic computing environments is another critical issue when deploying FL over 5G+ mobile devices.

Motivated by the aforementioned challenges, we plan to investigate energy-efficient FL over 5G+ mobile devices in this article. Our goal is to enable effective and efficient local training on 5G+ mobile devices while minimizing the overall energy consumption for FL over 5G+ mobile devices for both involved communications and computing. To this end, we first give an overview on FL over 5G+ mobile devices and discuss the energy consumption models. Then we study the local computing and wireless transmission co-design from the long-term learning perspective, where we make a trade-off between the two parts simultaneously. In addition, several advanced techniques, such as gradient sparsification, gradient quantization, weight quantization, model pruning, and dynamic batch sizing, are integrated into the proposed design to further reduce the overall energy consumption for FL over 5G+ mobile devices. Specifically, gradient sparsification, gradient quantization, and dynamic batch sizing are mainly used to save communication energy, while weight quantization and model pruning reduce the required computing energy consumption to cope with insufficient computing resources. It will be ideal that all those techniques can be integrated to enable energy-efficient FL over resource-constrained 5G+ mobile devices. Finally, we conclude this article with discussions on potential research directions for energy-efficient FL over 5G+ mobile devices.

Backgrounds and Energy Models

FL over 5G+ Mobile Devices

As an emerging decentralized learning paradigm, FL takes advantage of the computing resources across massive numbers of participants. Specifically, all participants collaboratively contribute to one global learning task in a distributed manner with continuous interactions for model parameter updates. With FL over 5G+ mobile devices, all 5G+ mobile devices (smartphones, laptops, automatic vehicles, etc.) can serve as participants, and a server such as a gNodeB acts as the aggregator. In particular, a 5G server first broadcasts a current global model to the participating 5G+ mobile devices in FL. After receiving the global model, a 5G+ mobile device conducts the local on-device training based on the local data and its computing capability. Second, when a 5G+ mobile device finishes its local training in this round, it will upload its local model updates (i.e., gradients) to the server via wireless links with 5G+ techniques. Finally, the server does the aggregation over all the received local gradients to update the global model and then feeds it back to the participating mobile devices for the next-round training. The above procedures are repeated until obtaining a converged global model, which can be deployed by 5G+ devices for future utilization. A typical paradigm of FL over 5G+ mobile devices, including local computation and wireless communications, is shown in Fig. 1.

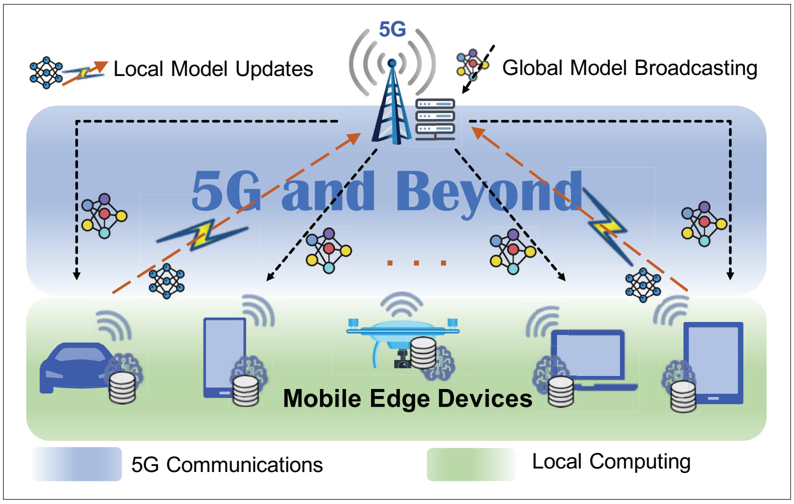


FIGURE 1. The illustration of FL over 5G+ mobile devices.

FL over 5G+ mobile devices has become a natural way to implement the artificial intelligence (AI) at the edge, like the keyboard prediction introduced by Google [4]. Such a combination pushes AI functions to mobile devices, which provides a flexible and convenient approach to conducting a learning task, especially for some real-time and lifelong learning applications. However, deploying FL over 5G+ mobile devices raises tremendous challenges and difficulties, and energy consumption is one of the most significant issues. Unlike other central servers with wired connections, mobile devices have limited energy resources due to limited battery power. Furthermore, the energy consumption of wireless transmissions is not encountered in learning scenarios with wired servers. Both make it extremely difficult for 5G+ mobile devices to handle the energy-hungry training tasks. In light of this, it is worthwhile to investigate FL's energy model over 5G+ mobile devices to deal with the energy-saving issue.

Communication and Computing Energy Models

With FL, all 5G+ mobile devices contribute to one unified global model by continuously transferring the local model updates with the centralized aggregator. In this process, the energy consumption of the 5G+ mobile devices mainly comes from the wireless transmissions of the model updates and the local computation executed on them. Hence, brief descriptions of the communication energy model and the computing energy model are given as follows.

Communication: All participating 5G+ mobile devices transmit their computed local model updates to the central aggregator through wireless transmissions, which corresponds to the communication energy consumption. Note that conducting an entire FL task training procedure with multiple communication rounds usually takes several minutes. In this situation, the channel condition may not remain the same all the time and may suffer from fluctuations. Therefore,

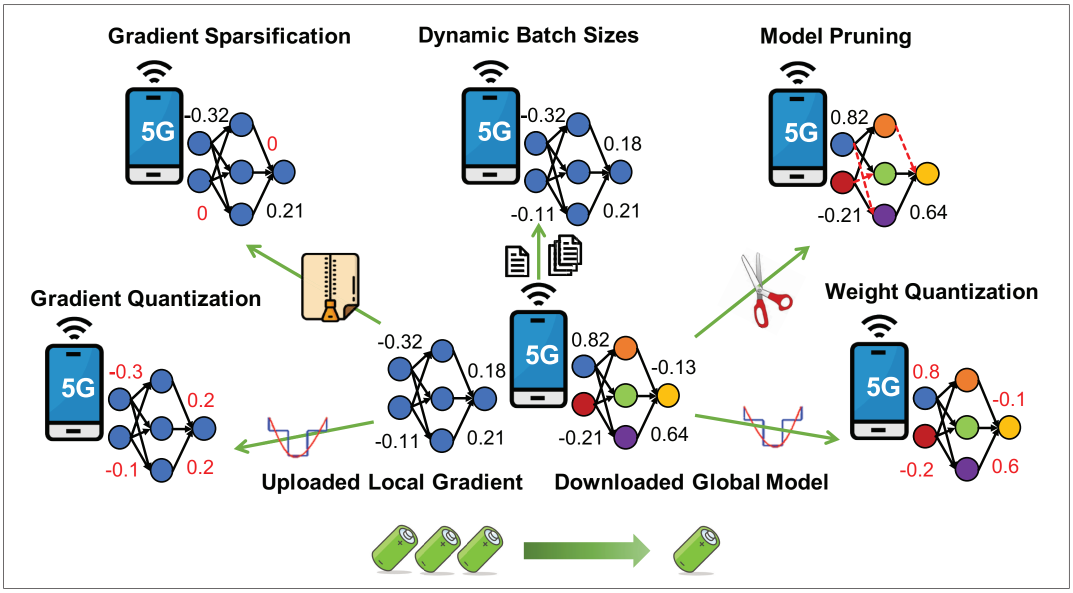


FIGURE 2. Illustrative scenarios for energy-efficient FL over 5G+ mobile devices.

one possible way to model the transmission rate with dynamic channel conditions for each device throughout the entire training process is to take the expectation over the channel variations. Furthermore, according to the Shannon–Hartley theorem, both the bandwidth and the transmission power impact the transmission rate, and the power of additive white Gaussian noise (AWGN) needs to be considered as well. Accordingly, the overall energy consumed by each device for wireless transmissions during the training process can be formulated as the product of the required number of global communication rounds and the energy consumption in a single round. Here, the one-round energy consumption is related to the transmission power, the transmission rate, and the model size.

According to the characteristics of the energy model mentioned above, two possible methods can be adopted to save the energy consumption on the communication part in the entire training process, namely, decreasing the required rounds of global communications and reducing the communication workloads per round. Given this observation, several approaches can be implemented to save the communication overhead. On one hand, global synchronizations can be done after several local computing iterations to decrease the communication frequency (i.e., the federated averaging). Similarly, the number of required global communication rounds can also be reduced by gradually increasing the batch size throughout the training. Since the total amount of data computation required for FL convergence is relatively fixed, these two methods greatly reduce the number of communication rounds by increasing the computing load in each communication round. On the other hand, model compression technologies (e.g., model sparsification and quantization) can greatly help reduce the size of the local model to be transmitted, thus saving the communication energy in each round.

Computing: With the ever increasing popularity of smart devices equipped with high-performance GPUs, 5G+ mobile devices can undertake heavy computations, even for deep learning tasks. However, due to the powerful computational capability and the massively parallel architecture of the GPU, the computational energy consumption has been a significant burden for the learning scenarios, especially for training tasks implemented on the mobile devices with limited battery power. Thus, more research efforts are needed to investigate the computing energy model and the corresponding computing energy-saving strategies. Here, we assume that the 5G+ mobile devices are equipped with GPUs, which are widely assumed in modern learning training. Specifically, the GPU computation architecture involves the memory modules referring to data fetching and the core modules referring to the data calculation. Under this architecture, the voltage and the frequency of the corresponding modules can be controlled independently.

The energy consumption for computing a mini-batch of data in one local iteration can be calculated as the product of the execution time and the runtime power. Here, the execution time is determined by the device-dependent parameters, like the memory frequency and the core frequency, and the task-specific information, such as the number of cycles for data fetching and calculation [7]. Similarly, the runtime power is also affected by device-dependent parameters, including the frequency and the voltage, and coefficients related to the specific learning tasks. Hence, the total computation energy consumption can be computed as the product of energy consumption for one local iteration and the total number of iterations. Currently, model compression techniques, such as pruning and weight quantization, together with the corresponding hardware co-design, can alleviate the burden of computing energy consumption in one local iteration because the needed number of cycles for data fetching and calculation is decreased. The total number of local iterations will increase mildly due to the falling of the model precision in each local iteration, but overall, the total computational energy will be saved.

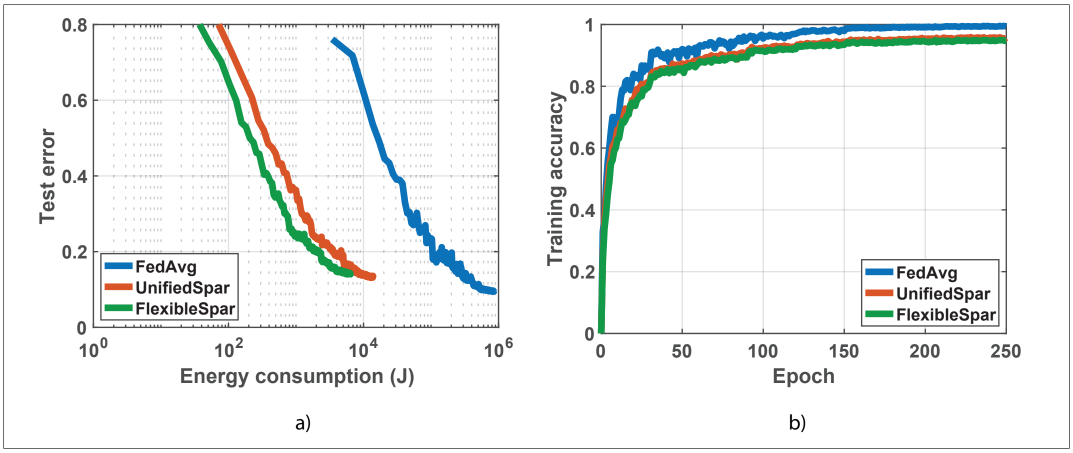


FIGURE 3. Performance evaluation for FL with gradient sparsification: a) test error vs. energy consumption; b) training accuracy vs. epoch.

Energy-Efficient FL via Communication and Computing Co-Design

As illustrated in the previous section, the energy consumption of FL over 5G+ mobile devices mainly comes from two parts: on-device local computing and wireless communications for model updates. Thus, in this section, we focus on how to integrate various technologies (gradient sparsification, gradient quantization, weight quantization, model pruning, etc.) and develop corresponding communication and computing co-design to reduce overall energy consumption for FL over 5G+ mobile devices. In this section, we survey some recent developments focusing on the communication and computing energy consumption co-design, which are illustrated in Fig. 2.

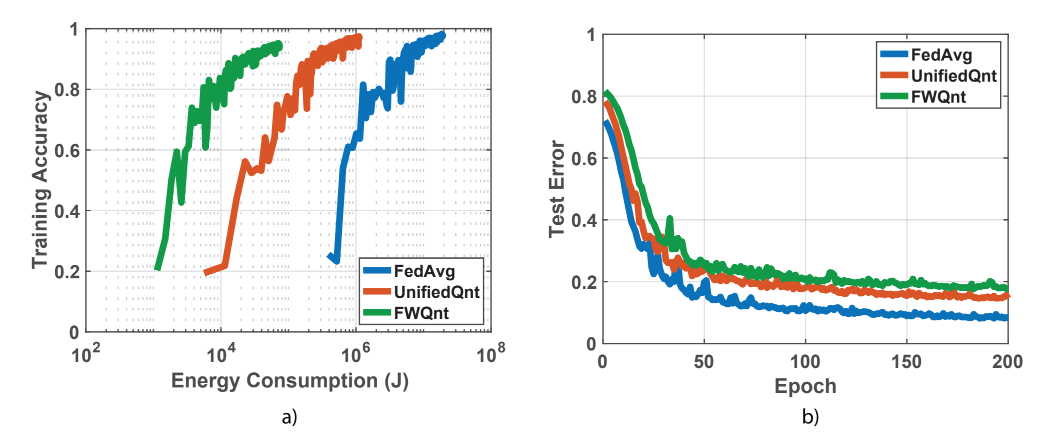
Gradient Sparsification to Reduce Communication Energy

FL communication energy consumption can be reduced by integrating the training algorithm with two state-of-the-art communication compression strategies, namely, local computations and gradient sparsification. The former allows more local computations to be performed on the 5G+ mobile device between every two global model updates, thereby reducing the total number of communication rounds. The latter lets participants upload only a fraction of gradients with significant magnitudes, thereby reducing the communication payload in each round. Furthermore, error compensation is applied at each participant after every model update to accelerate the global convergence by accumulating the error that arises from only uploading sparse approximations of the gradient updates, ensuring all gradient information eventually gets aggregated.

The convergence results for our FL algorithm indicate that, from the learning perspective, the gradient sparsity magnitudes of all the participants jointly impact global convergence and communication complexity. Given a target model accuracy, lower sparsity results in a larger number of communication rounds, potentially involving more communications to converge. Also, increasing the number of local iterations within a reasonable range is likely to reduce the overall communication complexity, but imposes more computing burdens on mobile devices in each communication round. In a realistic edge computing environment, these two types of compression factors implicitly determine the energy consumption for participating 5G+ mobile devices by affecting the payload required for transmissions and the workload required for processing, respectively. The above findings reveal that there is an intertwined trade-off between communication and computing, controlled by the compression factors. Thus, they need to be well balanced for each participant to accommodate the specific environment. This can be achieved by formulating a compression control problem using the derived convergence bound from the long-term learning perspective [8], with the goal of optimizing the overall energy efficiency for FL on 5G+ mobile devices over wireless networks. We consider an FL scenario with the ResNet20 deep model on the CIFAR-10 dataset, and the parameters of the global model are initialized by the Xavier method. As shown in Fig. 3a, the flexible sparsification-based FL method (*FlexibleSpar*), which considers the heterogeneity of participating 5G+ mobile devices and provides the flexible sparsification strategies derived from an elaborated compression control algorithm, consume less energy than the other methods. Specifically, *UnifiedSpar* makes every participant compress the gradients with a unified sparsity, regardless of the heterogeneous communication condition, and *FedAvg* is the original FL algorithm. Furthermore, Fig. 3b shows the convergence rate in terms of training epochs, which indicates that *FlexibleSpar* exhibits very similar behavior to *UnifiedSpar* in terms of convergence rate and final accuracy, both of which slightly underperform the baseline approach, FedAvg [1].

Gradient Quantization to Reduce Communication Energy

Similar to the gradient sparsification technique, another efficient compression method to reduce the communication energy consumption is gradient quantization. After completing the local training in one round, the computed gradients should be uploaded to the aggregator for the global updates. With gradient quantization, instead of uploading all raw gradients, we can also quantify the computed local gradients with low precision (i.e., a small number of bits), thus reducing the



a) b)

FIGURE 4. Performance evaluation for FL with weight quantization: a) training accuracy vs. energy consumption; b) test error vs. epoch.

communication load per round. In this way, a high gradient quantization level corresponds to low communication energy consumption per round. In contrast, the required global rounds increase due to the precision reduction, thus increasing the total computing energy. Therefore, the gradient quantization level needs to be controlled to make a trade-off between the communication and computing for the overall energy efficiency. It should be noted that local updates (i.e., gradients) refer to the communication part. Specifically, gradient quantization can help reduce the gradient size and thus save communication energy. However, when we download the global model to the device, the weight quantization is related to the forward pass in the local computing part, which is discussed in the following subsection.

Weight Quantization to Reduce Computing Energy

Weight quantization is regarded as a promising solution to decrease the energy consumption in local on-device training. It reduces the model complexity and thus computing energy via representing the model parameters with small bit widths (e.g., 8-bit or 16-bit fixed point numbers) during the training forward pass on 5G+ mobile devices. As model weights are represented by small bit-widths, the total data size is reduced, which saves the energy costs of memory accesses. The computational costs are also reduced because fixed-point arithmetic, such as 8-bit integer, consumes 30 times less energy than 32-bit floating point arithmetic. Considering device heterogeneity, the quantization selections for different participants vary. However, the quantization induces information loss during training. The errors between the quantized and original values make the FL model converge to a neighborhood of the optimal solution. Smaller quantization levels (i.e., bit representations) lead to higher error and push the FL model further away from the minima. In this way, given a model accuracy, total communication rounds are related to the average effect of quantization strategies across mobile devices. The participants with limited computing resources or good wireless conditions prefer more aggressive quantization strategies (smaller bit widths) to reduce the energy costs. In contrast, those who face long transmission delays would leverage higher-precision training to reduce the total communication rounds and the overall energy consumption, at the cost of more computing burden.

As a result, one can consider a joint design for flexible quantization selection and bandwidth allocation to capture the trade-off between the local computing and communications and minimize the overall energy consumption for FL training within the allowed deadline [9]. One example of a stochastic quantization scheme could first determine a quantization set based on different quantization levels, and then map the model weights to the nearest quantization point with high probability. The participating devices could determine different quantization levels depending on their device capabilities and the targeted model accuracy. Meanwhile, the server allocates the wireless bandwidth to the participants considering both the channel conditions and participants' computing capabilities. This process terminates when it reaches a certain global model accuracy level. Figure 4a shows the overall energy consumption for the FL training procedure under different learning mechanisms, and the settings of FL are the same as those in Fig. 3. For a fixed number of training iterations, those mechanisms equipped with quantization (UnifiedQnt and FWQnt) consume less energy than FedAvg without quantization. Specifically, the scheme FWQnt considers the impacts of the quantization on FL convergence, device heterogeneity, and wireless channel conditions, and enhances the energy efficiency with ×2–×100 less energy consumption than the other schemes in the FL training process under the same accuracy level. Moreover, the convergence rates for the corresponding schemes are also shown in Fig. 4b, where both quantization-based approaches slightly underperform FedAvg.

Model Pruning to Reduce Computing Energy

Another efficient learning technique that can be integrated into the FL process is model pruning, which can save the computing energy during training with proper underlying code design. Generally speaking, model pruning compresses the model by removing less contributing weights and connections, while retaining the performance of the original dense models. This will significantly affect the computing energy per iteration. First, computations are performed on an incrementally sparser model, reducing the total number of energy-intensive memory accesses. Second, pruning is performed across all neural network layers, including each convolutional layer. Typically, since convolutional layers dominate the overall energy consumption in a deep neural network, pruning leads to computing energy saving from the major energy consuming layers of the model. However, a more aggressive pruning strategy with higher pruning frequency slows down convergence due to network structure change, resulting in more communication iterations. Therefore, the pruning parameters need to be carefully designed to balance the trade-off between computation and communications for overall energy efficiency. Furthermore, reducing the model size also saves the inference time and energy, enabling the pruning method to be much more suitable for FL training over 5G+ mobile devices.

Typically, since convolutional layers dominate the overall energy consumption in a deep neural network, pruning hence leads to computing energy saving from the major energy consuming layers of the model. However, a more aggressive pruning strategy with higher pruning frequency, slows down convergence due to network structure change, resulting in more communication iterations.

Moreover, the pruning can be coupled with the quantization technique in FL training to address the restrictions in implementing deep neural networks on resource-constrained 5G+ mobile devices. Essentially, quantization requires a smaller number of bits to represent each pruned connection, thereby reducing memory, bandwidth, and energy consumption. To enable a communication-efficient and mobile-device-compatible FL process, we develop a three-fold compression of double quantization along with a model pruning approach [10]. In particular, the gradients and weights of uplink and downlink models are appropriately quantified, and the gradual pruning of the received model is utilized to reduce the computing and communication loads. This has a dual advantage of reducing the communication time and energy, and reducing the memory bandwidth due to fewer memory accesses. Therefore, we can reduce the model redundancy and make the FL process computation, storage, and communication sufficiently efficient to deploy large-sized deep neural networks over 5G+ mobile devices.

Other Techniques to Save Energy

The total energy consumption can also be reduced by capturing the intrinsic training dynamics, such as dynamically adjusting the batch size [11]. Specifically, we can interpret the stochastic gradient descent (SGD) training process as integrating a stochastic differential equation (SDE) whose "noise scale" is related to the batch size selection. Small batch size theoretically corresponds to large-scale random fluctuations, which can help explore the parameter space to avoid trapping in local minima at the initial stage in the FL problem. At a later stage, small-scale fluctuations (large batch size) are more desirable to fine-tune the parameters when a promising region of parameter space is reached. Therefore, gradually increasing batch size in the training process with a well-designed increment strategy can help reduce the communication rounds in the FL training process.

Moreover, gradually increasing batch size also leads to positive effects on computational energy saving. Due to the GPU's parallelism property, the local computing energy is no longer proportional to the batch size. Accordingly, the energy consumption of unit data calculation is relatively small for large batch training, especially executing on 5G+ mobile devices with multiple GPUs. Furthermore, it has been theoretically and experimentally demonstrated that both the fixed batch sizing approach and the dynamic batch sizing approach need similar data epochs. In this case, thanks to the benefit of large batch training at a later stage, the training approach with dynamic batch sizes is more energy-efficient. Furthermore, a larger increment factor of batch size decreases the number of required communication rounds but increases computation in each round. Therefore, a batch size control scheme catering to the GPU computing performances and wireless communication conditions of mobile devices can be further developed to balance the computing and communications, thus achieving energy-efficient FL over 5G+ mobile devices.

Balancing Communications and Computing to Reduce Overall Energy Consumption

As aforementioned, the trade-off always exists between communications and computing. Therefore, it is widely expected that both computing and communication parts need to be considered in the energy-efficient FL training process. When integrating both in the FL training process, the needed number of global communication rounds is a critical element. One possible way to approximate the needed global rounds is to conduct the FL theoretical convergence analysis. After identifying the specific dataset and the training model, the bound of the required number of communications can be derived based on the required training accuracy under some mathematical assumptions. Note that some learning settings (batch size, local iteration numbers, etc.) also impact the required number of communications. Accordingly, the total energy consumption corresponding to the required number of global rounds can be obtained, which exhibits a global view of the overall energy consumption in the FL training process. Despite employing the computing- or communication-efficient learning techniques, the energy efficiency cannot be significantly improved without elaborately controlling the key training parameters from a global perspective. This requires us to optimize the training parameters, so that communications and computing can be well balanced. This is why, in the above subsections, we first study efficient communication (e.g., gradient sparsification, gradient quantization and dynamic batch sizing) or efficient local computing (e.g., weight quantization and pruning [12]) methods in order to find the effective schemes. We then integrate these efficient methods with resource allocation strategies to strike a good balance between communications and computing, thus minimizing the energy consumption for FL over 5G+ mobile devices.

Challenges and Future Research Directions

Although there are a few pioneering research works done on energy efficiency for FL over 5G+ mobile devices, the relevant study is still in its infancy and requires more thorough investigation. In this section, we summarize some existing challenges and potential future directions.

Energy-Efficient FL over Heterogeneous Mobile Devices

Most of the existing energy-efficient FL algorithms assume a certain level of homogeneity of mobile devices, while they may be considerably different

The heterogeneity across participating devices may lie in local training data, computing capability, and wireless channel condition, and so on, which require flexible and customizable training strategies for each participant. Thus, it is expected to develop advanced design methodology for efficient FL over 5G+ mobile devices across multi-dimensional heterogeneity.

in practice. The heterogeneity across participating devices may lie in local training data, computing capability, wireless channel conditions, and so on, which require flexible and customizable training strategies for each participant. Thus, it is expected to develop advanced design methodology for efficient FL over 5G+ mobile devices across multi-dimensional heterogeneity. Accordingly, the quantization granularity, pruning strategy, and compression levels for different participants can vary to accommodate their realistic environments for energy saving. For example, a device with powerful GPUs but poor channel conditions may choose to compute more and communicate less and vice versa. In these situations, it is also significant to optimize the personalized compression or pruning strategy, adapting to the heterogeneous devices' capability of total energy saving by balancing communications and computing.

Energy-Efficient FL under Flexible Aggregation

The practical scenarios with heterogeneous data and devices demand higher requirements on the aggregation strategies. On one hand, the updated local models across devices may differ in size or even structure, which invalidates the current FL aggregation schemes (e.g., FedAvg [1]) if averaging model parameters directly. Thus, we need to investigate more powerful and more flexible aggregation schemes for FL over heterogeneous 5G+ mobile devices, while mitigating the energy consumption to the same extent as the baseline FL algorithms. On the other hand, due to the non-IID (independent and identically distributed) data sources and different precision for local models, each participant may have different contributions to the global model in terms of accuracy and convergence. Therefore, from the energy perspective, only a proportion of the participants need to contribute their models in each communication round, or the aggregation will be performed in an asynchronous way [13], thus reducing more energy compared to the original inefficient aggregation method. Overall, flexible and asynchronous aggregations are efficient methods to further reduce energy consumption, which deserves further investigation.

Energy-Efficient FL with Privacy Preservation

One of the inherited features of FL over 5G+ mobile devices is the privacy preservation of the users' sensitive raw data. Unfortunately, the private information can still be inferred from local updates communicated between user devices and the aggregation server with some recently developed attack mechanisms. Therefore, such distributed learning, which needs to exchange intermediate model parameters, brings a significant design challenge for privacy protection. A common strategy to enhance the privacy and security for participating users is to introduce perturbations into the FL training framework, such as adding the noise. Fortunately, some early research works have already shown that pruning, quantization, and other energy-efficient methods can introduce randomness into the FL training process and provide the enhanced privacy guarantee, that is, intrinsic privacy preservation. Under such conditions, energy-saving strategies and privacy protections are perfectly combined, where the privacy will be preserved without additional noisy computation. However, the related research is still in its initial stages and needs deeper exploration, especially for the differential privacy preservation, where the privacy level can be precisely quantified.

Extensive Applications of Energy-Efficient FL

With the massive growth of personal data with end users and the rapid popularization of the power-efficient mobile edge devices, FL over 5G+ mobile devices can be applied to a large number of applications [14] and can be involved in every area of daily life. With energy-efficient training strategies, FL over 5G+ mobile devices is perfectly compatible with lifelong on-device learning that requires a constant training process and is battery-driven. For example, it can be applied to some real-time assisting services like the voice UI, keyboard prediction, and some low-latency control scenarios such as gaming and automated guided vehicles [15]. Moreover, along with the exponential improvement in on-device AI capabilities, more sensing data from smart sensors like cameras, microphones, and compasses can be effectively utilized in IIoT, e-health, finance, social networks, and so on.

Conclusion

This article has studied FL over 5G+ mobile devices to address the issue of energy consumption during the FL training process. We have investigated how to properly conserve energy and allocate resources during FL training. We start with introduction of wireless communications and on-GPU computing models. Then we discuss several energy-efficient training techniques, including gradient sparsification, gradient quantization, weight quantization, model pruning, and dynamic batch sizing, to save energy. At the same time, the resource allocation strategies are adapted to reasonably manage energy resources by balancing communications and computing energy consumption. We conduct extensive simulations to demonstrate the efficacy of the techniques mentioned above for FL over 5G+ mobile devices. Finally, we have presented some existing design challenges and the corresponding research directions.

Acknowledgments

The work of D. Shi, R. Chen, P. Prakash, and M. Pan was partially supported by the U.S. NSF under grant CNS-2107057. The work of Y. Fang was partially supported by the U.S. NSF under grant CNS-2106589.

References

[1] B. McMahan *et al.*, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *Proc. Int'l. Conf. AI and Statistics*, Fort Lauderdale, FL, Apr. 2017.

[2] W. Saad, M. Bennis, and M. Chen, "A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems," *IEEE Network*, vol. 34, no. 3, May/June 2019, pp. 134–42.

[3] N. Kato *et al.*, "Ten Challenges in Advancing Machine Learning Technologies Toward 6G," *IEEE Wireless Commun.*, vol. 27, no. 3, June 2020, pp. 96–103.

[4] A. Hard *et al.*, "Federated Learning for Mobile Keyboard Prediction," arXiv:1811.03604, June 2018.

[5] 3GPP, "Technical Specification Group Services and System Aspects; Release 15 Description; Summary of Rel-15 Work Items," Technical Spec. (TS) 21.915, 09 2019, v. 2.0.0; https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3389

[6] P. Goyal *et al.*, "Accurate, Large Minibatch Sgd: Training Ima-

genet in 1 Hour," arXiv:1706.02677, Apr. 2018.

[7] X. Mei *et al.*, "Energy Efficient Real-Time Task Scheduling on Cpu-Gpu Hybrid Clusters," *Proc. IEEE INFOCOM*, Atlanta, GA, May 2017.

[8] L. Li *et al.*, "To Talk or to Work: Flexible Communication Compression for Energy Efficient Federated Learning Over Heterogeneous Mobile Edge Devices," *Proc. IEEE INFOCOM*, Virtual, May 2021.

[9] R. Chen *et al.*, "Energy Efficient Federated Learning over Heterogeneous Mobile Devices via Joint Design of Weight Quantization and Wireless Transmission," arXiv:1406.2661, Dec. 2021.

[10] P. Prakash *et al.*, "IoT Device Friendly and Communication Efficient Federated Learning via Joint Model Pruning and Quantization," *IEEE IoT J.*, Jan. 2022. DOI: 10.1109/JIOT.2022.3145865.

[11] D. Shi *et al.*, "To Talk or to Work: Dynamic Batch Sizes Assisted Time Efficient Federated Learning over Future Mobile Edge Devices," Submit to *IEEE Trans. Wireless Commun.*, 2022.

[12] S. Han, H. Mao, and W. J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding," *Proc. Int'l. Conf. Learning Representations*, San Juan, Puerto Rico, May 2016.

[13] Z. M. Fadlullah and N. Kato, "HCP: Heterogeneous Computing Platform for Federated Learning Based Collaborative Content Caching Towards 6G Networks," *IEEE Trans. Emerging Topics in Computing*, Apr. 2020.

[14] L. Li *et al.*, "A Review of Applications in Federated Learning," *Computers & Industrial Engineering*, Sept. 2020, p. 106854.

[15] R. Lu *et al.*, "5G Vehicle-to-Everything Services: Gearing Up for Security and Privacy," *Proc. IEEE*, vol. 108, no. 2, Nov. 2020, pp. 373–89.

Biographies

Biographies for all authors were not available at the time this issue went to press.