IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 7 May 2023, accepted 22 May 2023, date of publication 26 May 2023, date of current version 5 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3280411

SURVEY

Network for Distributed Intelligence: A Survey and Future Perspectives

CLAUDIA CAMPOLO[1,2], (Senior Member, IEEE), ANTONIO IERA[3], (Senior Member, IEEE), AND ANTONELLA MOLINARO[1,2,4], (Senior Member, IEEE)

[1]DIIES Department, University Mediterranea of Reggio Calabria, 89124 Reggio Calabria, Italy
[2]Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT), 43124 Parma, Italy
[3]DIIES Department, University of Calabria, 87036 Arcavacata, Italy
[4]Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190 Gif-sur-Yvette, France

Corresponding author: Antonio Iera (antonio.iera@dimes.unical.it)

This work was supported in part by the European Union through the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU (Telecommunications of the Future) by the Program RESTART under Grant PE00000001.

ABSTRACT To keep pace with the explosive growth of Artificial Intelligence (AI) and Machine Learning (ML)-dominated applications, distributed intelligence solutions are gaining momentum, which exploit cloud facilities, edge nodes and end-devices to increase the overall computational power, meet application requirements, and optimize performance. Despite the benefits in terms of data privacy and efficient usage of resources, distributing intelligence throughout the cloud-to-things continuum raises unprecedented challenges to the network design. Distributed AI/ML components need high-bandwidth, low-latency connectivity to execute learning and inference tasks, while ensuring high-accuracy and energy-efficiency. This paper aims to explore the new challenging distributed intelligence scenario by extensively and critically scanning the main research achievements in the literature. In addition, starting from them, the main building blocks of a network ecosystem that can enable distributed intelligence are identified and the authors' views are dissected to provide guidelines for the design of a ''future network for distributed Intelligence''.

INDEX TERMS Artificial intelligence, cloud continuum, distributed intelligence, machine learning, network.

I. INTRODUCTION

Fueled by the increasing amount of data, generated by massively deployed connected devices (up to 29.3 billions by 2023 according to Cisco's annual report [1]), Artificial Intelligence (AI) algorithms are significantly advancing decision making in many real-world application domains, ranging from smart manufacturing [2] to immersive experience [3], from autonomous driving [4] to healthcare [5].

As a branch of AI, Machine Learning (ML), relies on two main phases: *(i) learning*, which trains a model based on an input dataset, and *(ii) inference*, which provides knowledge/prediction upon the trained model. The conventional approach so far was to execute both training and inference in the cloud, by leveraging the computing capabilities of high-performing data centers. Nevertheless, the increasing demand for running training [6] and inference procedures [7] is outpacing the increase in computation power of existing centralized (cloud) infrastructures. Moreover, supporting frequent transmission of huge amount of training data towards the cloud is a challenging task even for wired links [8].

Such circumstances coupled with the responsiveness, security and privacy demands of a large set of ML-based applications, are pushing towards *distributed intelligence* solutions, leveraging computing resources spread from the cloud to the edge, and even extending to the *deep* edge, encompassing (resource-constrained) embedded devices [9]. For instance, model training can run in parallel over multiple distributed heterogeneous devices [6], and the execution of inference models can be sequentially split over multiple chained nodes [10].

The associate editor coordinating the review of this manuscript and approving it for publication was Hosam El-Ocla.

A. RELEVANT CHALLENGES

Distributing AI workloads entails deciding *how many* computing resources to dedicate and *where* to allocate them; *such*

decisions cannot overlook the status of communication links and the overall network dynamics. Indeed, (big) amount of data, e.g., huge raw datasets and (portion of) ML models as well as small-sized results of inference need to be moved across the network and possibly transformed to readily construct and distribute knowledge.

Such a new context raises unprecedented challenges to the network design; in fact, the network behaviour can highly affect the performance of applications built upon distributed intelligence.

On the one hand, the network may be a bottleneck, e.g., in case of distributed training [19], when the reliability and the latency of huge data exchanges among learning nodes over multiple iterations may be undermined by lossy and low-bandwidth channels. On the other hand, if properly designed, the network can boost distributed AI performance. Indeed, entities hosting AI-related components (e.g., data, models) can be chained to ensure reliable, low-latency and efficient data exchanges. Programmable network nodes on the path towards end-host applications may perform in-network Artificial Neural Network (ANN) processing [20], while reducing the amount of moved data and its latency.

B. CONTRIBUTIONS OF THIS SURVEY

From these considerations, a strongly felt need clearly emerges to develop future network solutions to better support distributed intelligence applications.

Nonetheless, so far, studies that relate AI and ML with next-generation networks mostly consider intelligence as a key enabler to improve network performance, in a perspective of "*AI for networks*", e.g., [21], [22], [23], [24], and [25]. Reason for this is undoubtedly the expected tremendous complexity of sixth-generation (6G) systems, which will likely be much denser (i.e., in terms of number of access points, users and devices), more heterogeneous (in terms of technologies), and will support a variety of fascinating applications with stricter performance requirements w.r.t. the fifth-generation (5G) [26]. Thus, solutions utilizing sophisticated AI/ML techniques are being designed to enable the cognitive management of network functionalities and to dynamically and promptly adapt offered services in an automated fashion, based on changes in user needs, environmental conditions, and business goals [11].

Only a few recent pioneering works recognized the need for a shift from a "network of information" to a "network of intelligence". They promote new communication primitives and network functionalities aimed to meet accuracy and latency constraints of AI/ML workloads [12], thus matching the scope of the so-called *"network for AI"* vision.

To the best of our knowledge, the literature is still missing a comprehensive analysis of the key design issues for future networks supporting AI, with special focus on the key requirements of emerging distributed intelligence solutions. Recently, some works have been published which address such topics, among which [11], [12], [13], [14], [15], [16], [17], [18]. However, as summarized in Table 1, the existing works either focus on specific network domains, e.g., wireless access [16], [18], edge domain [15], or they consider few communication (transport) protocols [17] and few radio resource allocation management approaches (i.e., power and bandwith allocation) [18], or miss a detailed overview of the state-of-the-art [11], [12], [13], [14].

In particular, taking a cue from the works summarized in Table 1 and going further, we aim to provide an *end-to-end perspective*, by identifying network design principles that can effectively support distributed intelligence over both the radio access and core network segments, in agreement with the vision of fifth generation (5G) and upcoming 6G systems. Such principles and the relevant scanned literature solutions span from radio resource allocation and innovative physical layer techniques, to both evolutionary and disruptive routing and forwarding mechanisms in the core network domain and orchestration and management approaches. To this aim, a comprehensive survey of the literature about network-related solutions to support distributed intelligence is shared along with our visions. We expect our investigation to fuel research efforts towards the design of a new network ecosystem aimed both at supporting distributed intelligence *by design* and at actively contributing to its widespread adoption.

C. ORGANIZATION OF THIS SURVEY

The remainder of this work is organized as follows. Section II scans some representative distributed intelligence solutions. The relevant network requirements and issues are dissected in Section III together with some solutions to the above issues coming from the AI community. Sections IV, V, VI discuss the technologies we deem relevant as key enablers of a future end-to-end network supporting distributed intelligence, in the wireless access domain, in the core network and at the orchestration layer, respectively. Section VII reviews the latest progress of the industry standardization and projects on developing network solutions to support the deployment of distributed intelligence. Then, Section VIII summarizes the main findings of the surveyed literature, provides guidelines for future research directions as well as pinpoints additional open issues. Conclusive remarks are reported in Section IX.

II. DISTRIBUTED INTELLIGENCE: REPRESENTATIVE IMPLEMENTATIONS

Several options have been devised for distributing training and inference workloads across multiple devices. In the targeted context, such devices are not limited to machines within data centers, but may span the cloud-to-things continuum, as graphically sketched in Fig. 1. In this section, some of the most representative solutions, which we deem may impact the network performance and be affected, in their turn, by the network performance, are recalled.

A. PARALLEL TRAINING

To speed up training and cope with increasing complexity and sizes of Deep Neural Network (DNN) models and training

TABLE 1. Differences between this manuscript and the closest related works.

Ref., year	Perspective	Distributed intelligence overview	Literature overview	Standardization efforts	Targeted scope and focus
[11], 2021	AI for network; Network for AI	Concisely provided	Not provided	Not scanned	Early results on intelligence-defined networking, compute-communication co-design and distributed AI in 6G
[12], 2021	Network for AI	Not provided	Not provided	Not scanned	Evolution towards the network of intelligence with early results
[13], 2021	AI for network; Network for AI	Not provided	Not provided	Not scanned	Early identification of the main pillars of a learning-oriented network architecture
[14], 2021	Network for AI	Not provided	Not provided	Not scanned	Detailed design of a new ICN-based networking architecture for AI with early evaluation
[15], 2022	AI for network; Network for AI	Provided	Provided	Scanned	(Wireless) Edge-distributed ML co-design among solutions to address communication inefficiencies for distributed ML
[16], 2021	Network for AI	Provided	Provided	Not scanned	Wireless communication techniques for efficient deployment of distributed learning
[17], 2021	Network for AI	Provided	Provided	Not scanned	Analysis of communication optimization techniques for distributed intelligence from the networking community (with focus on transport protocols only) and the AI community perspectives
[18], 2023	Network for AI	Provided	Provided	Not scanned	Overview of communication optimization techniques for distributed learning from the networking community (with focus on radio resource management only) and the AI community perspectives
This work	**Network for AI**	**Provided with focus on relevant requirements**	**Provided**	**Scanned**	**End-to-end perspective on network design for distributed intelligence**

datasets, full advantage of parallel nodes can be taken by partitioning the data, the model, or a combination of them [6], [27]. The *data-parallel* approach assumes to partition the data and feed the different portions to a set of distributed nodes, all implementing the same model. Alternatively, a *model-parallel* approach can be applied in which the entire dataset is used by each node that operates on different parts of the model, and the final model results from the aggregation of the various parts.

Initially proposed for data centers with multiple workers, the same approach has been recently borrowed in the edge computing domain, where several nodes can enforce the aforementioned parallel tasks [28], [29]. For instance, in [30], such techniques are applied to Convolutional Neural Networks (CNNs) models for video surveillance tasks.

B. MODEL SPLITTING

A further distributed framework, that can be applied to both training and inference, is *model splitting*, in which different (at least two) portions of a complex ML model are executed sequentially in different processing nodes [31]. The peculiarity of this type of distributed learning is that each node does not train an instance of the whole model and model layers are processed sequentially. In the envisioned context, for instance, end-devices may run the initial computation-friendly model layers of the DNN and then, send the intermediate results to edge nodes and cloud facilities to feed the remaining computation-heavier layers and produce the final outputs [10], [32], [33].

The model splitting idea builds upon the fact that the data size at some intermediate layers of a DNN is significantly smaller than that of raw input data. Hence, on the one hand, it is possible to reduce the transmission latency and the incurred amount of exchanged data compared to cloud-only DNN implementations. On the other hand, model splitting oversteps the limitations in terms of computing and storage capabilities of edge devices, which are usually not able to fully deploy and run large deep network models (e.g., containing up to millions of parameters) [34]. In addition, model splitting protects user privacy by transmitting partially processed data rather than transmitting raw data [35].

How to split the model is a critical decision, since it affects the resulting computational cost and communication overhead. If not properly selected, it can cause the data amplification effect [34], according to which the size of intermediate output data of the DNN can be larger than that of the input data.

C. FEDERATED LEARNING

Rather than sending raw data to a remote server, in Federated Learning (FL) the model is (entirely, unlike in Model Splitting) locally trained on their own data by distributed devices,

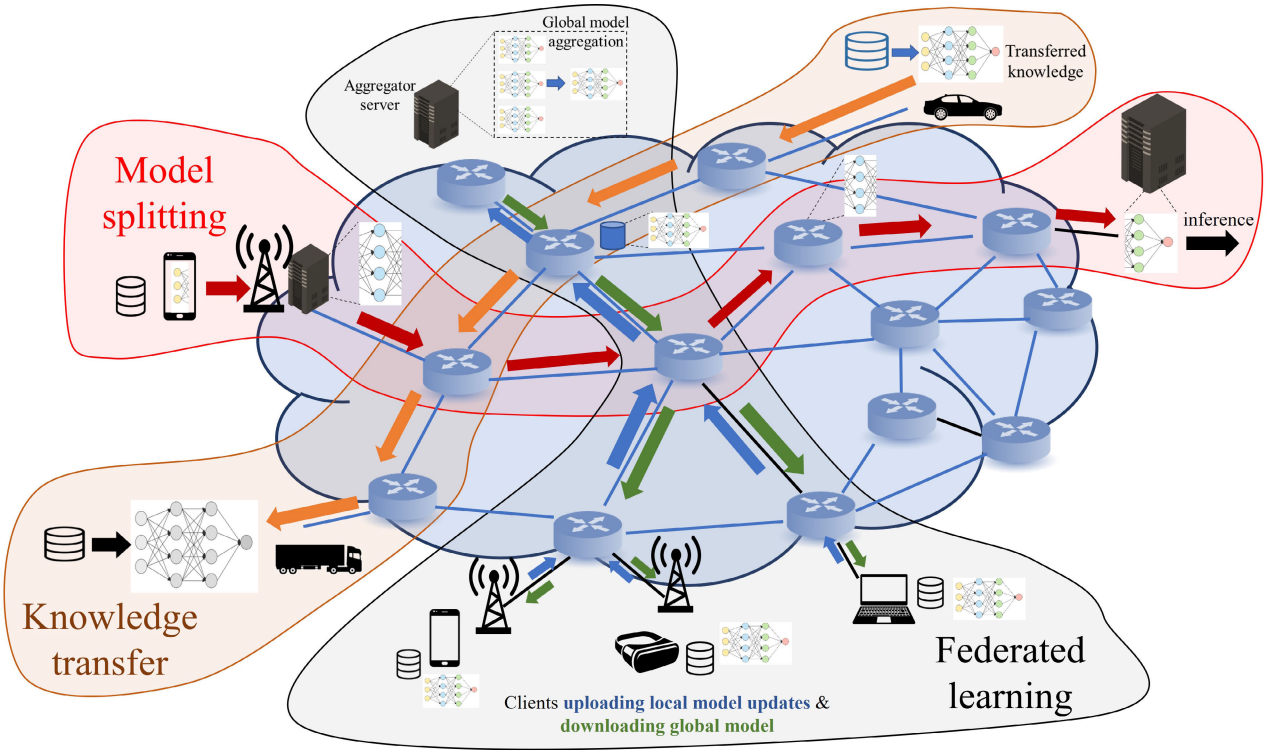

FIGURE 1. Representative distributed intelligence solutions deployed across the continuum. End-devices and network nodes included in the shaded areas are those involved in specific distributed intelligence solutions, i.e., model splitting, knowledge transfer and federated learning. Some of the network nodes, besides forwarding AI-related data and relevant models, can cache them as well as execute (portions of) them.

referred to as clients. The latter ones then share the relevant parameters (e.g., NN weights) with a server, which aggregates the model and pushes it back to the clients. The procedure is reiterated over several rounds until the desired accuracy level is achieved [36], [37].

Initially proposed by Google [36], Federated Learning allows minimizing user privacy leakage, since raw data are not required for the training at the Federated Learning aggregator. As a further benefit, latency due to data offloading is reduced compared to a centralized learning approach and network resources for data exchange saved.

The single point of failure (the single server) is one of the main weaknesses of the Federated Learning approach. Indeed, the central aggregator may not always be available and reliable when the number of edge devices is large, and especially in some application scenarios, e.g., cooperative driving and robotics [38]. Hence, decentralized server-less Federated Learning solutions, also referred to as *swarm learning*, are gaining momentum [39], [40], [41]. In such decentralized solutions, devices interact with their neighbors through device-to-device (D2D) communications, and consensus mechanisms are needed to ensure the achievement of a common learning goal.

Finally, coupling Federated and Split Learning may be contemplated when resource-constrained clients are involved and training and deployment of the full model is poorly feasible [42].

D. TRANSFER LEARNING

In particular usage scenarios, e.g., the medical domain [43], accessing the massive data necessary to train Deep Learning models is very expensive, or even impossible. In other cases, data may available in a number of disjoint physical locations, e.g., those collected from users' personal devices for the sake of activity recognition [44] or from smart devices in smart manufacturing context [45].

By mimicking human behavior in which it is possible to apply knowledge learned in one domain to solve problems in a different domain, Transfer Learning enables the transfer of knowledge and learning between devices [46]. Although not fully equivalent to the distributed approaches mentioned so far, Transfer Learning also entails the distribution of data/models among different nodes spanning different domains.

E. DISTRIBUTED REINFORCEMENT LEARNING

Reinforcement Learning involves a sequential decision-making procedure, where a learner takes (possibly randomized) actions in a stochastic environment over a sequence of time steps, and aims to maximize the long-term cumulative rewards received from the interacting environment according to a given policy.

Although initially conceived for the single-learner tasks, multiple learners can be foreseen which are coordinated by a central controller, whenever scalability becomes an issue,

TABLE 2. Network requirements to support data exchange in representative distributed intelligence (DI) solutions.

DI solution	Exchanged data	Bandwidth	Latency
Parallel Training	*Data-parallel*: Huge raw datasets once and towards each learner; *Model-parallel*: Medium sized-intermediate results once by each learner	*Data-parallel*: Very high; *Model-parallel*: Medium	Medium
Model Splitting	*Training*: Medium-sized intermediate results over hundreds of iterations (for feed-forward and backward propagations); *Inference*: Medium-sized intermediate results over a single iteration	*Training*: High; *Inference*: Low	*Training*: Medium; *Inference*: Very low
Federated Learning	Global model from the FL aggregator to the FL clients over hundreds of iterations; Model updates from each FL client to the FL aggregator over hundreds of iterations	*Global model*: High; *Model updates*: Medium	Medium
Transfer Learning	(partially) trained models between different domains	Medium	Low
Distributed Reinforcement Learning	Small-sized gradient aggregations from the coordinator to the learners over millions of iterations; Small-sized updated weights from each learner towards the controller over millions of iterations	*Gradient aggregations*: High; *Updated weights*: High	High

e.g., in robotics [47], drones-based [48] and autonomous vehicles applications [49]. To this aim, the latter one must exchange information with all learners, by collecting their rewards and local observations, or, broadcasting the policy to them [50].

III. NETWORK REQUIREMENTS AND ISSUES FOR DISTRIBUTED INTELLIGENCE

When considering distributed intelligence spanning different nodes, achieving high accuracy does not only matter. Indeed, its achievement also depends on the specific network deployment and can be affected by the peculiar network requirements of each distributed intelligence solution.

A. NETWORK REQUIREMENTS

Whatever the distributed intelligence approach and the motivation to implement it, it is essential to properly connect the nodes that perform the different tasks in order to streamline operations. It is possible to opt for systems with a single central point of aggregation, or based on tree structures, or even systems leveraging fully distributed nodes, possibly organized into clusters, without hierarchical relationships. The different architectural models and their relevant deployed topology differently impact on: the obtainable accuracy, the system resilience and security/privacy degree, the overall computing load of nodes, and, what we are most interested in, the communication footprint and network performance.

Table 2 reports the network requirements in terms of communication footprint (bandwidth) and latency to support specific data exchange (e.g., dataset, entire model, model weights, etc.) in representative distributed intelligence solutions.

1) BANDWIDTH

Distributed intelligence solutions have been mainly conceived to partition the computing load and/or reduce the communication footprint. Some of the aforecited approaches may still require to transfer bandwidth-hungry datasets among different nodes, as in the case of parallel training solutions. However, also when datasets are not exchanged, heavy models demanding high bandwidth may be transferred, ranging in size from hundreds of MBytes to several GBytes [51], for instance in the case of Transfer and Federated Learning.

Additionally, model updates may also imply a huge network load, if frequently (e.g., at every iteration) and massively transmitted (e.g., by multiple nodes), as for Distributed Reinforcement Learning and Federated Learning solutions. Each model update may have as many parameters as the model itself. Natural Language Processing models usually have hundreds of millions of parameters. For instance, the well-known GPT-3 model has 175 billion parameters corresponding to over 350 GBytes [52]. Training such kind of models by exchanging the aforementioned amount of data per each communication round is challenging, especially over wireless channels.

In case of model splitting, data outputs of intermediate DNN layers could overwhelm the network and deteriorate the learning/inference performance due to a wrong choice of the splitting points.

2) LATENCY

In distributed intelligence solutions, latency is composed of two main contributions: the *communication latency* due to exchanges of data (whatever the type, as per Table 2) over the network, and the *computing latency*, due to the execution of (portions of) training/inference models.

In general, keeping a short latency is more crucial for inference compared to training operations. Indeed, inference often needs to be executed in near real-time to provide prompt reaction to events, e.g., 200 ms to get predictions from ML models for voice assistants, or below 10 ms when tactile Internet and autonomous driving operations are considered.

B. MAIN ISSUES

Compared to centralized solutions, distributed intelligence solutions exhibit specific issues which need to be addressed.

1) HETEROGENEITY OF DEVICES

Devices to be involved in the distributed training and inference procedures may span the cloud continuum, hence

encompassing Internet of Things (IoT) devices as well as edge/cloud nodes. Heterogeneity may concern the computing/battery capabilities, as well as the experienced connectivity conditions. In practical distributed learning setups, some clients are stragglers and cannot send their updates regularly, either because: *(i)* they cannot finish their computation within a prescribed deadline due to limitations of their computing capabilities, or *(ii)* they experience poor and/or intermittent connectivity.

On the one hand, stragglers may significantly deteriorate the convergence of distributed learning procedures as the computed local updates become stale. On the other hand, if they are not selected to contribute to the training procedures, the model quality could be extremely low, especially if their number is high [53].

2) MOBILITY OF DEVICES

In several upcoming applications, learning algorithms will rely on data provided/generated by either terrestrial [54] or aerial mobile devices [55]. Raw data retrieval under high mobility conditions may face poor robustness due to fast fading and may be hindered by short-lived connectivity. The mobility of clients could become a concern also for Federated Learning, when model updates need to be iteratively exchanged over multiple rounds. The issue is less exacerbated in the case of inference, where time dynamics are typically smaller. However, it could happen that inference split may be interrupted during the movement of mobile users from one edge access point to another [56].

3) LACK OF INTEROPERABILITY

Different distributed AI components should be able to transparently collaborate to serve the same purpose, in case of parallelization and serialization of training and inference workloads. However, unlike in centralized AI deployments, interoperability is severely hindered by fragmented and mainly application-specific solutions [57].

A typical approach to achieving the targeted requirements and solving some of these issues comes from the AI community and consists in considering the network as a possible bottleneck and the solution sought is to adapt the distributed learning techniques to make them work better despite the limits imposed by the network. Most of the solutions of this type in the literature try to act on the data, models, and information managed by learning/inference algorithms in such a way as to overcome the issues experienced by distributed training and inference solutions over the continuum.

Model compression techniques, such as pruning and quantization, help reducing the computational complexity of DNNs [58], [59], to enable their execution even in constrained devices at the deep edge, while reducing latency and energy consumption, although at the expenses of a lower accuracy.

More in detail, quantisation strategies provide a low precision representation of weights, gradients or activations to reduce the total number of bits transmitted in each update and thus, reduce the incurred communication footprint and latency transfer [59]. Sparsification techniques prevent irrelevant updates from being transmitted by, for example, removing redundant information and only transmitting the important values from local estimates. Knowledge distillation techniques transfer knowledge learnt from a larger model or ensemble of models, such as output predictions, feature activations, or correlations between feature maps, to train a smaller model. Knowledge distillation techniques can be applied in Federated Learning to send soft-label predictions, instead of heavier updated models or gradients [60].

The analysis of the aforementioned communication-efficient techniques is extensively surveyed in the literature, for instance in [18], [59], [61], and [62]. As stated in [19], such solutions require changes to applications and may hurt the accuracy performance of models. For this very reason, the design of solutions coming from the network community is advocated as the real game changer to enable distributed intelligence solutions.

In alignment with such a perspective, one must start from understanding *the actual impact of the aforementioned requirements and issues on the network design*, and consequently move from the current concept of networks that already support distributed services well towards that of *networks for distributed AI-driven services*, which will surely characterize future 6G ecosystems.

To this aim, in the following Sections we will discuss potential network enablers to adequately support distributed intelligence. For each technical enabler, we briefly discuss potentials, by scanning relevant representative solutions in the literature, whenever possible, and then, outline possible limitations and open issues.

IV. ENABLERS AND SOLUTIONS IN THE RADIO ACCESS

Intense research activities have been recently carried out to exploit ML for optimizing various procedures in wireless communication networks (e.g., handover, radio resource allocation) [21], [22]. Less attention instead, has been devoted to assess the impact of ML techniques in practical wireless communication systems. For instance, whenever learning is offloaded to the edge/cloud, the transfer of huge datasets could easily burden the uplink air interface. The same holds for model updates sent by mobile devices acting as Federated Learning clients.

Communication solutions aimed to make the best of the limited radio resources are strongly needed to enable effective distributed intelligence in upcoming 6G systems [16], [88], [89].

However, only very recently efforts have been devoted to explore how adapting, optimizing, and arranging wireless networks can contribute to implementing ML techniques [90]. Some of these solutions are discussed in the following, by scanning some representative works in the literature. The solutions analysed in this Section and summarized in Table 3 refer to the Radio Access Network (RAN) of future wireless systems.

TABLE 3. Overview of literature solutions targeting distributed intelligence in the Radio Access Network.

Ref., year	Enabler	DI solution	Main contribution	Methodology
[63], 2020	Semantic communications	Edge learning	Importance-aware scheduling exploiting channel diversity and data diversity simultaneously	Simulations and validation with a real dataset
[64], 2020	Semantic communications	Edge learning	Data-importance aware Automatic Repeat-reQuest which adapts retransmission decisions to both data importance and reliability	Simulations and validation with a real dataset
[65], 2020	Semantic communications	FL	Joint data selection and communication resource allocation algorithm based on the data importance	Simulations and validation with a real dataset
[66], 2020	Semantic communications	FL	Scheduling policy jointly accounting for the staleness of the received model parameters and the instantaneous channel qualities to improve the FL convergence rate	Monte Carlo simulations
[67], 2020	NOMA	FL	NOMA used for the uplink transmission of FL model updates	Simulations
[68], 2020	NOMA	FL	Scheduling policy and power allocation applied to improve NOMA-assisted FL performance	Simulations
[69], 2021	NOMA	FL	NOMA-assisted transmission of FL model updates after a fixed time, before aggregation	Monte Carlo simulations
[70], 2022	NOMA	FL	NOMA-assisted model updates delivery aimed at minimizing the convergence round and to maximize the user access fairness	Analytical expressions validated by Monte Carlo simulations
[71], 2022	NOMA	FL	NOMA-assisted simultaneous uplink model updates transmission and joint minimization of the total energy consumption and FL convergence latency, also through Wireless Power Transfer of clients	Simulations and validation with real datasets
[72], 2021	NOMA	FL	NOMA-assisted FL for UAV swarms	Simulations and training with a real dataset
[73], 2020	AirComp	FL	Fast model update aggregation through analog modulated local models/gradients simultaneously transmitted by client devices	Simulation results and validation with a real dataset
[74], 2023	AirComp	Model Splitting	Interplay between MIMO-based over-the-air computation and neural network for split learning	Numerical results and validation with real datasets
[75], 2022	RIS	FL	RIS-assisted power minimization of FL clients	Simulations
[76], 2021	RIS + AirComp	FL	Joint optimization of RIS configuration and client selection	Simulations and validation with a real dataset
[77], 2020	RIS + AirComp	FL	FL model aggregation boosted by a RIS-aided simultaneous access scheme	Illustrative results
[78], 2021	RIS + AirComp	FL	RIS-assisted mitigation of the signal magnitude misalignment of AirComp	Simulations and validation with a real dataset
[79], 2021	RIS + AirComp	FL	Multiple RISs leveraged to reduce the signal distortion of AirComp	Simulations and validation with a real dataset
[80], 2022	RIS + AirComp	FL	Hybrid learning approach with high-computing devices performing local training and the others directly transmitting to the aggregator raw datasets; all exploit RIS-assisted AirComp	Simulations and validation with a real dataset
[81], 2021	D2D	FL	Small data samples exchanged with neighbors to improve the quality of data spread among clients	Experimental testbed with real datasets and IoT devices
[82], 2020	D2D	FL	Data sharing over D2D links to adjust the computation loads at devices and reshape the data distribution for enhancing the training speed and accuracy	Simulations and validation with a real dataset
[83], 2020	D2D	FL	Consensus-based mechanism enabled through D2D interactions	Experimental IIoT testbed
[84], 2021	D2D	Model Splitting	Inference splitting over devices in proximity	Experimental testbeds with real datasets
[38], 2021	D2D + AirComp	FL	Local model consensus through D2D aided by AirComp	Simulations and validation with a real dataset
[85], 2022	Relaying	FL	Local updates delivered to the aggregator by poorly connected clients with the help of their neighbors	Simulations
[86], 2022	Relaying	FL	UAV-assisted model updates delivery coupled with UAV trajectory and device scheduling optimization	Simulations and validation with a real dataset
[87], 2022	Relaying + AirComp	FL	Relay-assisted over-the-air FL to counteract the communication straggler issue	Simulations and validation with a real dataset

A. SEMANTIC COMMUNICATIONS

Conventional communication techniques assume data bits being of equal importance. For learning, instead, some samples within a training dataset may be more important than others. Therefore, data can be delivered more efficiently over wireless links by differentiating the usefulness of training data samples. In this context, Age of Information [91] is a further possible metric to consider for assessing information significance.

Semantic communications appear extremely promising in such a context, when informative messages need to be transmitted for training/inference procedures. They refer to a paradigm shift for the wireless system design from data-oriented communication (i.e., maximizing communication rate or reliability based on Shannon theory) to goal-oriented communications targeting effective task execution among distributed network nodes [92].

1) STATE-OF-THE-ART

The work in [63] proposes a data-importance aware user scheduling algorithm for communication-efficient edge machine learning. A classifier is trained at the edge server by utilizing the data distributed at multiple edge devices. The scheduling decision is based on a data importance indicator, which both incorporates the signal-to-noise ratio and data uncertainty. Likewise, re-transmission is devised in [64], which selectively re-transmits a data sample based on its uncertainty, which helps learning. Besides raw data collection, a similar philosophy can be applied for the acquisition of learning relevant information in Federated Learning as in [65]. Similarly, in [66], based on a metric termed the Age of Update, a scheduling policy is proposed to improve the Federated Learning efficiency that jointly accounts for the staleness of the received parameters and the instantaneous channel qualities.

2) LESSONS LEARNT AND ROAD AHEAD

Some of the aforementioned solutions, although promising to reduce the bandwidth pressure over wireless links and to improve accuracy and latency performance, entail further advancements towards more sophisticated and powerful wireless transceivers. Semantic extraction is necessary to understand what information is of interest before transmitting it; this, in turn, may require AI models.

B. NON-ORTHOGONAL MULTIPLE ACCCESS

The conventional orthogonal multi-access (OMA) schemes are inefficient to handle massive learners competing for radio resources, as in the case of Federated Learning. The required radio resources linearly scale with the number of edge devices that participate in the learning process. On the contrary, Non-Orthogonal Multiple Access (NOMA) allows multiple devices to transmit simultaneously on the same channel, so that the data rate is increased and the communication latency is reduced [93]. This is crucial to make training convergence faster and to improve communication efficiency.

1) STATE-OF-THE-ART

The seminal work in [67] proposes NOMA for model update in Federated Learning. Clients are capable of transmitting their trained parameters simultaneously, while the base station decodes the users' messages by utilizing Successive Interference Cancellation (SIC). There, it shown that NOMA outperforms a traditional time division multiplexing access approach. The proposal also adaptively compresses gradient values according to either sparsification or quantization. A scheduling policy and power allocation scheme using NOMA is proposed in [68] in order to maximize the weighted sum data rate under practical constraints during the entire learning process.

The Compute-then-Transmit NOMA (CT-NOMA) protocol is introduced and optimized in [69]. According to CT-NOMA, users terminate concurrently the local model training and then, after a fixed time, simultaneously transmit the trained parameters to the central aggregator. The work in [70] formulates a multi-objective optimization problem adopted to minimize the convergence round and to maximize another crucial metric in the NOMA domain, namely the user access fairness.

In [71] NOMA is applied coupled with wireless power transfer, which is adopted by the base station to power the end-devices. An optimization problem is formulated with the aim of minimizing a system-wise cost that includes the total energy consumption and the overall latency for the Federated Learning convergence.

In [72] a group of unmanned aerial vehicles (UAVs) use their collected data to train their respective local models, which are then aggregated into a global model. There, NOMA enables the follower-UAVs to send their local models to the leader-UAV simultaneously over a same resource block.

2) LESSONS LEARNT AND ROAD AHEAD

Despite the neat advantages of NOMA to improve uplink transmissions, its concrete implementation is still an open issue. Existing works mainly consider ideal SIC, which is not always the case. Moreover, more sophisticated access solutions, such as hybrid NOMA/OMA configurations, which could possibly enhance the scalability of Federated Learning, may definitely be a subject matter of future work.

C. AirComp

AirComp [94] has been recently developed as a new air-interface solution, which merges the concurrent data transmission from multiple devices and performs ''over-the-air'' data aggregation, by exploiting the inherent waveform superposition property of a multi-access channel.

1) STATE-OF-THE-ART

In [73] AirComp is proposed for fast model update aggregation. The Federated Learning server directly receives

the aggregated version of analog modulated local models/gradients simultaneously transmitted by devices. Such a scheme allows simultaneous access and hence, can dramatically reduce multi-access latency compared to the OMA schemes, better scaling with the number of clients, without significant loss of the learning accuracy. AirComp coupled with MIMO is applied to split learning in [74].

2) LESSONS LEARNT AND ROAD AHEAD

Accurate channel estimation and strict synchronization of participating devices, although hard to achieve under mobility and highly dynamic channel conditions, are mandatory for an effective AirComp implementation.

Although the AirComp approach can significantly improve the performance of model aggregation for Federated Learning, it may still suffer from unfavorable signal propagation conditions over the wireless links, such as deep fading. Since all local parameters are uploaded via noisy concurrent transmissions, the unfavorable propagation error can prevent from achieving a high accuracy of the aggregated global model. Hence, it is typically not deployed in a stand-alone manner, but instead coupled with other techniques, as discussed in the following.

D. RECONFIGURABLE INTELLIGENT SURFACES

Thanks to their capability of proactively modifying the wireless communication environment, Reconfigurable Intelligent Surfaces (RISs) have become a prominent technology to mitigate a wide range of challenges encountered in wireless networks [95]. More in detail, the large number of low-cost passive reflecting elements of a RIS can adjust the phase shift of the incident signal and thus, altering the propagation of the reflected signal. The signal reflected by the RIS can be constructively superposed with the signal over the direct link to boost the received signal power, by compensating for the power loss over long distances and/or obstructed propagation paths. Moreover, compared with conventional active relays, RISs usually do not require dedicated energy supplies for operation.

1) STATE-OF-THE-ART

RISs have been extensively leveraged to improve the Federated Learning performance. Battery constraints drive the design of the proposal in [75] where the total transmit power minimization of clients is targeted with the assistance of the RIS, for a sustainable Federated Learning implementation. RISs can also help counteracting the AirComp limitations in a Federated Learning context. The work in [76] proposes a unified framework to jointly optimize RIS configurations and client selection.

A novel simultaneous access scheme empowered by RIS is proposed in [77] to develop a smart radio environment and hence, to boost the performance of model aggregation. In [78] a RIS is deployed to mitigate the signal magnitude misalignment of AirComp during model aggregation at the server, due to a unfavorable propagation environment. Similarly, in [79] multiple geo-distributed RISs are deployed to enhance the parameter aggregation from IoT devices to the base station in an efficient manner.

A hybrid learning approach is proposed in [80]. There, devices with high computing capabilities are selected to learn locally, whereas the others upload their datasets to the base station for remote aggregation on behalf of them. To enhance spectrum efficiency, both the updated models and the raw data are transmitted concurrently over the simultaneous transmitting and reflecting RIS-assisted multiple access channels.

2) LESSONS LEARNT AND ROAD AHEAD

The RIS design and deployment *per se* is still a challenging topic. RISs comprise a large number of reconfigurable phase shifts to be optimized, as well as of devices' transmit beamformers and of the base station's receive beamformer, also under imperfect CSI and under fast-varying channel environments. Deep Reinforcement Learning can be applied to properly adapt the RIS configuration by learning about the environment [96].

E. D2D COMMUNICATIONS AND RELAYING

D2D communications have emerged as a promising technology for optimizing spectral efficiency in future cellular networks [97]. They exploit the proximity of devices for efficient utilization of available radio resources, improving data rates, unburdening the network infrastructure and reducing latency and energy consumption. In addition, devices experiencing poor connectivity (e.g., those at the edge of the cell) can forward their transmissions to the base station by establishing a D2D link with a device in proximity acting as a relay. In addition, in the envisioned context, resource-constrained devices can offload training/inference tasks to resource-rich devices in proximity through D2D communications [35].

1) STATE-OF-THE-ART

In [81] nodes can exchange small data samples with trusted neighbors, calculate similarities among datasets locally, and report them to the Federated Learning aggregator, so to improve the quality of data spread among clients. The authors in [82] propose a solution aiming to minimize the total delay for the FL model training, by optimizing the radio resource allocation for both D2D data sharing and distributed model training. The work in [83] leverages the cooperation of devices that perform data operations inside the network by iterating local computations and mutual interactions via consensus-based methods. In [38] AirComp is adopted to facilitate the local model consensus in a D2D communication manner.

Instead of designing a client selection mechanism for Federated Learning, or optimizing resource allocation to balance client participation, the authors in [85] introduce a relaying mechanism that takes into account the nature of individual clients' connectivity to the aggregator and ensures that,

in case of poor connectivity, their local updates are delivered to the aggregator with the help of their neighboring clients acting as relays. A similar approach is foreseen in [87] and [98], where a relay-assisted over-the-air Federated Learning scheme is proposed to counteract the communication straggler issue. With similar purposes, the work in [86] proposes to leverage a UAV as a flying aggregator when no terrestrial base station is available. There, the authors propose to jointly design UAV trajectory and device scheduling, in order to ensure that all devices (also those experiencing poor channel conditions) have the opportunity to successfully participate in distributed training.

Collaboration among devices in proximity is also beneficially exploited for inference splitting in [84], where close devices exchange data via the Web real-time communication protocol. Similar approaches are surveyed in [99] and references therein.

2) LESSONS LEARNT AND ROAD AHEAD

Despite the huge literature leveraging D2D communications for distributed intelligence solutions, to the best of our knowledge, most of them fail to investigate how D2D communications can actually be designed to match distributed inference needs.

Moreover, as for other applications, still the exploitation of direct short-range communications, relaying and task offloading among devices in proximity entail the definition of proper incentive mechanisms.

F. SHORT-PACKET COMMUNICATIONS

Control commands and sensor status updates with ultra-reliable and low-latency communications requirements are normally conveyed in short packets, in the order of hundreds of bits. Short-packet communications can be also leveraged to transfer inference results, which need to be promptly transmitted to feed into the decision-making process.

In short-packet communications, the decoding error probability at a receiver is not negligible, due to a finite block length, and to the fact that both the thermal noise and the channel distortion are not easily averaged. Moreover, the associated packet control information is not negligible compared to the short payload.

Despite the recent advancements in information theory, several issues still need to be addressed to make the transmission of short packets efficient and reliable [100] and no solution still exists specifically meant to address *inference delivery*.

G. MULTICASTING

Point-to-multipoint communications are expected to support distributed intelligence. Multiple devices may, in fact, simultaneously need to be queried or to receive data. For instance, multiple clients are instructed by the Federated Learning server to locally perform model training, thus becoming the simultaneous recipients of both initial global model and updated parameters. Furthermore, multiple devices, e.g., surveillance cameras in a smart city, may be the simultaneous recipients of updated inference models (e.g., for face recognition).

Proper network primitives (e.g., multicast) are required to efficiently and effectively forward data over radio links towards the nodes involved in the learning/inference process.

So far, broadcast communications are mainly assumed, with the exception of a few works that specifically mention multicast interactions, e.g., [101], [102], [103], and [104]. In [101] and [102], a global model is sent in multicast to a set of Federated Learning clients. There, a basic multicasting scheme is envisioned according to which the throughput is assumed to be bounded by that of the client experiencing the worst channel conditions. No further details are provided about how the multicast group is formed and maintained, and how transmissions are actually performed over the radio interface (e.g., over which channel, with which periodicity).

Efforts should be devoted to practically implement the above procedures, also in compliance with the Third Generation Partnership Project (3GPP) 5G Multicast Broadcast Services (MBS) specifications in Release 17 and beyond [105]. Moreover, improvements over the basic legacy multicasting procedures can be envisioned, e.g., through dynamic subgrouping [106] applied to the Federated Learning clients, in order to speed-up the global model delivery.

V. ENABLERS AND SOLUTIONS IN THE CORE NETWORK

Datasets, models, intermediate/final inference results may need to traverse the continuum, hence entailing a huge traffic to be routed beyond the wireless edge domain across the core network, when properly steered across different end-points.

In the following, solutions addressing these issues on top of existing infrastructures are discussed and summarized in Table 4. Some of them, widely known in the 5G context, shall likely be re-engineered to handle distributed intelligence. Others are emerging as key innovations of beyond-5G systems and we believe that can serve the aforementioned purpose.

A. SOFTWARE-DEFINED NETWORKING

Routing and forwarding protocols have undergone a deep transformation in recent years owing to the Software-Defined Networking (SDN) paradigm [122]. By decoupling the control plane from the data plane, and moving the former to a logically centralized entity, the Controller, SDN allows abstracting network functions (e.g., routing, load balancing) from the underlying network nodes, which become simple forwarding elements. Thanks to the network-wide view of the Controller about link status and network nodes under its control, sophisticated mechanisms for traffic control and resource management can be more flexibly deployed.

Huge research efforts have been devoted to improve SDN performance through ML techniques, as surveyed e.g., in [123]. However, only recently, interesting research works have started to address how a programmable control

TABLE 4. Overview of literature solutions targeting distributed intelligence in the core network.

Ref., year	Enabler	DI solution	Main contribution	Methodology
[107], 2021	SDN	FL	Auction method and SDN management to create a trading strategy for an FL server and clients	Simulations with realistic dataset
[108], 2021	SDN	FL	Forwarding graph optimization for FL	Emulation
[109], 2021	SDN	FL	SDN-assisted communication among FL controllers	Emulation
[110], 2019	ICN	Generic learning and inference	Preliminary design of NDN architecture for distributed AI in mobile environments	Theoretical
[111], 2020	ICN	Generic learning and inference	Analysis of the benefits of NDN for edge AI	Theoretical
[112], 2022	ICN	FL	Client discovery and data delivery supported by NDN semantic-rich naming and in-network caching	Simulations
[113], 2022	ICN	FL	Communication-efficient framework for FL based on the pub/sub model targeting Internet of Vehicles	Experimental test-bed
[14], 2021	ICN, SDN	Generic learning and inference	Software-defined service-centric three-layers framework for in-network intelligence	Small-scale emulation
[114], 2019	Programmable data plane	Inference for classification	Framework for in-network classification, mapping learning algorithms to a match-action pipeline	Software and hardware implementation
[115], 2018	Programmable data plane	Inference Model Splitting	NN models quantization and execution of such quantized models on both network processor-based SmartNICs and programmable switching chips	Experimental test-bed
[116], 2018	Programmable data plane	Generic Learning	Lightweight lossy-compression algorithm for floating-point gradient values and NIC-integrated compression accelerator	Experimental test-bed
[117], 2021	Programmable data plane	Inference for smart network telemetry	Distribution of neurons of an ANN into multiple switches instead of running an entire ANN in a single device	Optimization and implementation in SmartNICs
[118], 2021	Programmable data plane	Inference for classification	In-network classification that addresses the issue of creating simple and reasonably accurate ML models to be deployed into a programmable data plane with minor performance degradation	Software and hardware implementation
[119], 2019	Programmable data plane	Data parallelism	In-network aggregation of model updates from multiple distributed workers	Experimental test-bed
[120], 2019	Programmable data plane	Distributed RL	In-switch acceleration acting over packet payloads to facilitate the gradient aggregation	Experimental test-bed
[121], 2022	Programmable data plane	Inference Model Splitting	Splitting of a DNN over UPFs of a 3GPP network	Emulation

plane, through the SDN paradigm, can support functionalities related to distributed intelligence applications. Some of them are scanned below.

1) STATE-OF-THE-ART

In [107], the authors devise a solution for Federated Learning, in which the server acts as the auction buyer while the clients function as sellers. There, the SDN Controller is used to create an overlay network in which server and clients can perform auction bidding and product provision. Also, in [108] the potential of SDN Controllers are exploited to orchestrate the optimal forwarding graph for several slices in order to optimize communications associated with Federated Learning in software-defined IoT networks. Instead, the authors of [109] address two key aspects of a mobile IoT network, i.e., security and seamless connectivity for data delivery. They propose to exploit an SDN-assisted Federated Learning approach to predict the users' demands for a particular content in order to improve content placement decisions. SDN facilitates privacy in the communication channels between Federated Learning controllers.

2) LESSONS LEARNT AND ROAD AHEAD

The aforementioned works are among the first examples that are emerging from the literature, in which SDN Controllers are designed to support distributed intelligence applications. There is still much to do, since their role in orchestrating the overall distributed intelligence implementation is still marginal. By way of example, two areas are identified in which studies of this kind show great potential.

The first field of investigation considers the role of an SDN controller to support a Federated Learning server in the client selection phase. By leveraging SDN, *(i)* not only the memory and computation capabilities of the clients, but also the delays on the core network path that divides them from the server can be taken into account during the selection phase, and *(ii)* the load conditions on the core network links can be continuously monitored and dynamically adapted in the view of an improvement in the entire Federated Learning performance.

The second area of research considers the capability of the SDN Controller to set multicast forwarding rules inside the switches' forwarding tables. This will facilitate the

delivery of the same model updates to several clients, whenever point-to-multipoint communications encompass wired links besides the radio interface. Indeed, in recent years, huge efforts have concentrated on designing multicast supported controllers, and the related interfaces to program them, for distributing many types of contents [124]. The full potential of such techniques to support collective communications for distributed intelligence solutions still needs to be unveiled.

Another interesting application area is that of in-network learning, in which an SDN Controller could dynamically support ensemble learning implemented via the deployment of various Weak Learners (WL) within programmable switches. As known, a WL produces a classifier which is only slightly more accurate than random classification. The role of the Controller could be the wise and dynamic routing of data flows between WLs, in order to optimally distribute the learning tasks across a programmable data plane of the core network, see Section V-C.

B. SEMANTIC ROUTING

Making the core network aware of the type of data traffic exchanged among players of the various distributed intelligent processes could make it more supportive for these processes. In this respect, the network could differentiate the handling of packets, exchanged during learning and inference procedures, based on the nature of the carried content and the originating applications. In the literature, there are several solutions to manage packets in a differentiated way. Some of them are designed to be implemented in small and private Internet Protocol (IP) domains, others to be used across multiple domains over the Internet. Some require clean-slate solutions, others can be supported via current Internet extensions or hybrid solutions.

In particular, *semantic routing* represents a promising direction to explore for the case of data exchange related to distributed intelligence applications. It is intended as the process of routing packets that contain IP addresses with additional semantics, possibly using that information to perform policy-based routing or other enhanced routing functions [125].

Different techniques have been proposed which allow flexibly modifying the packet treatment behaviour, i.e., the forwarding decisions. This can be done by either adding information into IP packet headers to adequately instruct network nodes, or by modifying addresses or even interpreting them differently. Several methods are being studied to extend the semantics of IP headers [125]. Unfortunately, to the best of our knowledge, despite the potential advantages deriving from using this approach to identify and optimally treat packets related to distributed intelligence applications, literature on this subject is still lacking, being the semantic routing technology a brand new topic.

It is our opinion that IP packets originated by any given distributed intelligence service, if embedded with differentiated semantic information, can bring great benefits to the service itself. For instance, packets belonging to a given dataset can be forwarded towards a node, which is equipped with the right capabilities to run the training procedure upon them. Similarly, an inference output, which is needed for fast decision making, can be sent over a low-latency path; instead, a huge dataset can be carried over a low-congested and high-bandwidth path. In the case of model splitting, for example, the instructions contained in the packets' header could appropriately guide intermediate data from one node to the next, in order to execute different model portions sequentially.

What we believe should be pursued is to create adequate semantics, specific to flexibly support the traffic associated with distributed intelligence, and accordingly redirect it to the optimal endpoint or over the path meeting its service quality requirements [125].

C. PROGRAMMABLE DATA PLANE

The flexibility addressed in the previous two subsections calls for data plane programmability. It entails a network device to expose the low-level packet processing logic to the control plane, through standardized Applications Programming Interfaces (APIs), to be systematically, rapidly, and comprehensively reconfigured. We support the idea that good results could come from a wise joint use of network control plane virtualization (via SDN) and data plane virtualization techniques, for example by using switches or Network Interface Cards (NICs) that are programmable, e.g., via the P4 language [126], [127].

Programmable network switches, like other network devices and NICs, have been identified among the main enablers for the transition of intelligence into the data plane [114], [128]. They can play a key role for applications like real-time flow classification and detection of network traffic anomalies, thus acting as *on-path* Neural Network (NN) accelerators, which avoid additional data transfer towards purpose-built *off-path* dedicated hardware.

1) STATE-OF-THE-ART

In this view, there is a wide body of literature starting from initial works [115], [129], which hypothesized the implementation of NNs and the execution of in-network inference within programmable network devices. Most of the works deal with ML models that run entirely on single network devices [114], [116]. In [114], the authors introduce a framework for in-network classification, in which they map both supervised and unsupervised algorithms to a match-action pipeline, and discuss the applicability of such implementations.

The obvious risk is overloading the devices themselves and subtracting resources usually dedicated to packet processing and forwarding, thus reducing their performance levels. For this reason, other interesting works are beginning to appear, such as [117], which proposes distributed in-network intelligence based on distributed NNs, and addresses the challenges of neuron specification, placement, and chaining in

switches, by considering smart network telemetry as a use case. In [118], the focus is on deploying ML trained models for classification, showing how to express them into the P4 language's primitives and focusing on in-network classifier for an Intrusion Detection System as a proof-of-concept.

The work in [119] addresses the network bottleneck issue due to the heavy exchange of model updates in parallel training. Distributed workers send their model updates over the network, where an (integer) aggregation primitive implemented in a programmable data plane switch, sums the updates and distributes only the resulting value. A P4 program distributes aggregation across multiple stages of the switch ingress pipeline. An in-switch aggregation accelerator is also proposed in [120], but to reduce the gradient aggregation overhead in Distributed Reinforcement Learning training.

In [121] the authors propose to enhance the data plane of a 3GPP network with the aim to enable in-transit Deep Learning inference services over extended User Plane Functions (UPFs). Required extensions of the control plane and the management plane to support the envisioned data plane modifications are also discussed, e.g., in terms of interfaces and service discovery.

2) LESSONS LEARNT AND ROAD AHEAD

Despite the inherent benefits of in-network acceleration, a particularly relevant open issue is related to the choice of how to split a trained DNN model so that it can best fit data plane functions along a forwarding path. This is addressed by a recent paper [121], which deals with the problem of finding valid strategies for splitting and distributing DNNs within programmable network devices in the light of the growing complexity of large-scale NN models. The research in this field is in its infancy and efforts in this direction are still required.

D. INFORMATION-CENTRIC NETWORKING

Departing from the host-centric IP model that is oblivious of the content of exchanged packets, Named Data Networking (NDN) [130], one of the most prominent Information-centric networking (ICN) instantiations [131], conveys semantic-rich names and attributes in exchanged packets, natively enabling a more conscious data delivery. NDN also natively implements in-network caching and, if properly extended [132], can support in-network processing, thus enabling in-network intelligence at a wide extent.

Overall, such features make NDN a candidate networking solution to support distributed intelligence workloads. Clearly, the support of distributed intelligence requires additional functionalities in the NDN data plane, besides forwarding.

1) STATE-OF-THE-ART

In [110] it was early argued that through native in-network caching and name-based forwarding, NDN can facilitate the orchestration of distributed AI components. The potential benefits are more extensively discussed in [111].

In-network caching can play a crucial role for both training and inference phases. Inference results can be cached if deemed of interest for several end-points, e.g., a given traffic sign on a driving lane that needs to be detected by multiple vehicles. Also data for training or intermediate trained results (e.g., in case of model splitting) can be cached to quickly recover from packet losses, which may occur over unreliable and/or congested links.

The authors in [112] propose NDN to improve client discovery and data exchange procedures in Federated Learning. An expressive and flexible naming scheme allows declaring the capabilities of heterogeneous clients in a uniform semantic-rich manner. Moreover, it is proven that multicast data delivery and in-network caching save precious network resources. This is especially useful when exchanging huge global models over potentially congested and/or bandwidth-limited lossy backhaul links. The potential of ICN in Federated Learning is also investigated in [113], where it is coupled with the Kafka publish/subscribe framework to support Internet of Vehicles applications.

Finally, ICN can be combined with SDN to realize a service-centric architecture for in-network intelligence orchestration, as argued in [14].

2) LESSONS LEARNT AND ROAD AHEAD

Initial attempts to effectively support distributed intelligence applications by leveraging disruptive technologies to enhance the IP data plane exist. However, it is well known that the practical large-scale deployment of disruptive ICN-based solutions is still far from being a reality [133], unless to consider greenfield environments.

Moreover, there is much room for further theoretical investigations. For example, model popularity can be used to support caching decisions. In this respect, a popular trained model can be cached into those network nodes that are more likely to be traversed by input data, and the model placement can be dynamically updated based on requests [7]. Overall, novel caching policies should be defined to match the delivery requirements of distributed inference and learning tasks.

VI. END-TO-END ORCHESTRATION AND MANAGEMENT

The design of effective network ecosystems meeting the requirements of distributed intelligence goes well beyond the scope of the radio access and core network solutions scanned in the previous sections, and entails further actions. Whenever an intelligent application, built upon AI/ML learning/inference, requests the execution of a workload, its lifecycle management (i.e., initial configuration, placement, maintenance, update, delete, etc.) must be performed.

Some instantiations of such solutions are discussed in this Section and summarized in Table 5.

A. COMPUTING AND COMMUNICATION CO-DESIGN & ORCHESTRATION

Policies are needed to jointly orchestrate and manage computing, caching, and communication (3C) resources across

TABLE 5. Overview of literature solutions targeting end-to-end orchestration and management for distributed intelligence.

Ref., year	Enabler	DI solution	Main contribution	Methodology
[134], 2022	Co-design and orchestration	Generic inference	Reinforcement learning-based inference offloading and model selection • *Targets: to minimize response time while meeting accuracy requirements*	Test-bed
[7], 2021	Co-design and orchestration	Generic inference	Distributed and dynamic allocation algorithm for the newly concept of inference delivery network • *Targets: to minimize cost made of inference latency and accuracy*	Numerical results
[135], 2022	Co-design and Orchestration	Learning Model Splitting	Heuristic solving an optimization problem aimed at selecting the data to be used, choosing the distributed DNNs and physical nodes to run their layers • *Targets: to minimize energy consumption while meeting time and accuracy requirements*	Numerical results, Test-bed
[136], 2022	Co-design and orchestration	Learning Model Splitting	Split points selection accounting for time-varying network throughput and computing resources • *Targets: to jointly minimize energy consumption and training time*	Experimental test-bed with IoT devices
[137], 2021	Co-design and orchestration	Inference Model Splitting	ML model partitioned across different layers in the presence of multiple MEC servers • *Targets: to minimize the inference latency*	Simulations and realistic ML model cost profiles measurements
[138], 2021	Co-design and orchestration	FL	Joint client selection and RB allocation • *Targets: to minimize the accuracy loss*	Simulations and validation with real datasets
[139], 2021	Co-design and orchestration	FL	Energy and latency-aware resource management and client selection • *Targets: to minimize cost made of inference latency and accuracy*	Simulations
[140], 2021	Co-design and orchestration	FL	Joint client selection and bandwidth allocation algorithm supported by O-RAN • *Targets: to minimize the convergence time to achieve a certain model accuracy*	Simulations
[141], 2021	Co-design and orchestration	FL	Intra-FL and inter-FL service bandwidth allocation • *Targets: to minimize the training round duration*	Simulations
[142], 2021	Virtualization	Generic inference	OMA LwM2M-based virtual intelligent object	Test-bed
[143], 2023	Virtualization	FL	MQTT and OMA LwM2M-based client discovery	Test-bed
[59], 2022	Virtualization	Generic learning and inference	Analysis of communication-efficient strategies for DI solutions	Theoretical
[144], 2022	Virtualization + SDN	Generic learning and inference	Liquid-specific and flexible software-defined network architecture for AI applications in 6G	Theoretical

the cloud-to-things continuum in order to optimize distributed AI/ML workloads [11], [59].

Joint 3C orchestration for distributed ML workloads unveil specific challenges compared to more traditional orchestration problems. On the one hand, the seamless support of end-to-end distributed pipelines of AI components cannot rely on the conventional orchestration of the computing infrastructure, designed orthogonally to the network architecture. In fact, if data exchange occurs in myopic manner, then the network risks becoming the bottleneck for distributed intelligence. On the other hand, the policies meant to decide where to place a given intelligence task should not only minimize data collection latency, computation times and/or energy consumption, but also provide the targeted accuracy in training and inference.

In this regard, a key issue is the choice of the proper model instance to perform a given learning/inference task, among the multiple ones that may be available throughout the

continuum. Lighter model versions can be run by constrained end-devices at the expenses of lower accuracy, while more sophisticated models can be available at large data centers.

Additionally, according to the deployed distributed intelligence application, the following possible decisions may need to be taken, e.g., when and where a model (re-)training procedure shall be triggered; which clients shall be selected for a Federated Learning task; among which nodes and where an inference/training model shall be split.

1) STATE-OF-THE-ART

The work in [134] proposes to employ online reinforcement learning to orchestrate Deep Learning services for multi-users over the end-edge-cloud system, in order to better understand the system dynamics. The offloading policy aims to minimize response time while providing sufficient accuracy.

In [7] the authors propose the concept of *inference delivery networks*, i.e., networks of computing nodes that coordinate to perform inference tasks. The proposal targets the following problems: *(i)* where placing the models for serving a certain inference task among a set of nodes, and *(ii)* how selecting their size/complexity among the available alternatives, while achieving the best trade-off between latency and accuracy. Similarly, the work in [135] proposes a framework allowing to jointly decide *(i)* which data using for learning, *(ii)* which DNN structure employing, and *(iii)* which physical nodes, and resources therein, using. The proposal aims at minimizing the energy consumption while meeting a target maximum learning time and desired learning quality. Unlike the work in [7], the latter targets the more challenging and resource-intensive learning phase, and splits a model across multiple devices.

The authors in [136] propose a solution, which aims to jointly accelerate model training time and minimize energy consumption in resource-constrained IoT devices, through the proper selection of model splitting points. It accounts for the time-varying network throughput and the computing resources of involved devices. Model splitting is also the focus in [137], where an inference model is partitioned in a 5G infrastructure across the end devices, multiple edge server and the cloud, in order to minimize the inference latency.

Orchestrating computing and communication resources is also crucial for Federated Learning. Indeed, client selection is critical to determine the training time and model accuracy, and it cannot be performed by overlooking (wireless) channel conditions and its dynamics and clients' computing/battery capabilities. Several client selection schemes have been proposed, e.g., [101] and [145], which consider the aforementioned criteria. However, tighter coupling is required between client selection and related resource assignment (e.g., transmission power, radio resource blocks), which should be performed in a joint manner. In [138], a joint client selection and resource allocation policy is proposed for Federated Learning under communication limitations and imperfect CSI.

In general, selecting more clients may reduce the overall training latency. Optimizing the tradeoff between maximizing the number of clients and minimizing the overall energy consumption is the target of the study in [139]. There, appropriate resources, in terms of CPU frequency and transmission power, are allocated to the selected clients. An algorithm is proposed in [140] that jointly selects the best clients and allocates the right amount of bandwidth at each round, by considering their data, computing power, and channel gain. The algorithm can be implemented as an application in the open radio access network (O-RAN) architecture, to collect information about the data (e.g., the numbers of data and data classes) as well as the channel gain, and the available computing power of potential clients.

In [141] the co-existence is considered of multiple Federated Learning services sharing common wireless resources. A two-level resource allocation framework is proposed, which aims to minimize the round length by optimizing bandwidth allocation among the clients of each Federated Learning service, and distributing bandwidth resources among multiple simultaneous services.

2) LESSONS LEARNT AND ROAD AHEAD

Despite the aforementioned works, in the recent literature the coupling of distributed intelligence orchestration with the network data plane is still loose. For instance, in case of model splitting, decisions shall be taken along with the configuration of the proper routing path across the nodes selected to host the different NN layers. The interplay of orchestration mechanisms with SDN routing policies is highly recommended.

A more holistic approach for jointly orchestrating network and computing resources can build upon network slicing, with a slice tailored to the specific demands of distributed AI applications, while sharing the (programmable) network infrastructure with other services [146].

Another interesting perspective is early discussed in [147]. Understanding the quality of input data (e.g., subject to anomalies, discrepancies, and noisy components) can be exploited for selective sensing, in order to reduce the input data volume to be transferred over the network. There, in the healthcare context, the potentials of a cross-layered sense-compute co-optimization are presented to jointly improve sensing, computation, and communication aspects of ML-based applications over the end-edge-cloud continuum.

B. VIRTUALIZATION

The aforementioned decisions cannot be taken without collecting sufficient information about capabilities of end-devices and network nodes, as well as about AI/ML models. For instance, to enable smart selection of the best Federated Learning clients [101], [145], candidate devices can be asked by the aggregator to provide information about their capabilities as well as the experienced link quality. The same holds for the identification of the nodes which should host a partitioned model in case of splitting.

Virtualization techniques, initially proposed in the Information Technology domain, can serve this purpose, being recently inherited within the IoT context [148] and evolving towards the broader digital twin concept.

1) STATE-OF-THE-ART

In [142] a virtualization layer hosted at the network edge is proposed, which is in charge of the semantic description of AI-embedded IoT devices' capabilities. The virtual replicas expose and augment the cognitive capabilities of the corresponding (potentially constrained) physical devices, in order to feed intelligent IoT applications. Properly customized data models defined according to the Open Mobile Alliance (OMA) Lightweight Machine-to-Machine (LwM2M) semantics [149] are leveraged to this purpose. The choice falls into OMA LwM2M being it specifically conceived for resource-constrained devices. A similar approach is followed in [143] for the semantic description of the client capabilities in an edge-based Federated Learning context involving IoT devices as learners.

The authors in [59] propose the use of digital twins as virtual replica of a distributed training system. They can provide a platform for real-time analysis to learn the peculiarities of the system, and target inefficiencies in communication.

The need for joint 3C and learning (3C-L) resource allocation approaches is theoretically discussed in [144], where a liquid model for 6G is proposed. According to the vision discussed there, 3C-L resources provided by edge nodes may be virtualized, also with the support of digital twins, with the aim of improving resource pooling.

2) LESSONS LEARNT AND ROAD AHEAD

Different players involved in the network design and deployment for supporting distributed intelligence should achieve a consensus on the proper data models/semantics to adopt. Furthermore, although they foster the crucial issue of interoperability and enable more judicious orchestration policies, digital replicas of AI components still may increase the communication load to keep the state of their physical counterparts in real-time. Thus, their instantiation and whole lifecycle management need to be carefully planned.

VII. STANDARDIZATION INITIATIVES AND PROJECTS

Synergies among network-related standardization activities and AI/ML forums are advocated to accelerate advancements in both domains.

The International Telecommunications Unit (ITU) Focus Group on Technologies for Network 2030 (FG-NET2030) pioneeringly identified the pervasive distribution of AI as a crucial use case for future networks [150].

Efforts are underway within the Internet Research Task Force (IRTF) COmputing in the Network Research Group (coinrg) to push new network protocol designs to efficiently federate decentralized computing resources, also to support emerging AI/ML workloads [151].

The one6G Association outlines distributed federated AI among the key enabling technologies that constitute the pillars of the evolution towards 6G [152].

Further initiatives are encouraged which recognize the need for a novel network design to satisfy the AI demands and to optimize the AI performance, and not the other way around only. Indeed, the investigation of AI to optimize the network performance are targeted, for instance, by ITU [153] and 3GPP [154].

Along with the advocated path, 3GPP prospective work items for the upcoming 5G Release 18 are aimed to support AI applications. For instance, the document in [155] covers use cases and potential requirements for 5G system to support AI/ML model distribution and transfer (download, upload, updates, etc.). This is an interesting contribution coming from a standardization group, but still too germinal.

Dynamic distribution of intelligence is the subject matter of several recent projects funded by the European Commission. For instance, this is the case of the DAIS [156] and DEDICAT 6G [157] projects. However, a few of them, among which the AI@EDGE project [158], claims to address network design for AI.

VIII. MAIN FINDINGS, GUIDELINES AND OPEN ISSUES

A. FINDINGS AND GUIDELINES

The conducted analysis unveils that, despite the infancy of the distributed intelligence topic, a huge amount of works have been published in the literature. Starting from it, the following main considerations can be summarized.

1) RESEARCH SOLUTIONS IN THE RAN DEFINITELY OUTNUMBER THOSE IN THE CORE NETWORK

It can be observed that the design of solutions specifically targeting the wireless (edge) domain mainly catalyzed the interest of the networking community. This trend was somehow expected, given the wireless channel dynamics and the resource-constrained nature of end-devices. Moreover, this is also due to the fact that most of the distributed intelligence deployments consider a two-layer architecture involving end-devices and an edge server only. However, in the near future the compute continuum will become a reality and even network nodes could contribute to (portion of) training and inference procedures. Hence, additional efforts are required, on the one hand, to improve distributed intelligence-related traffic steering, and on the other hand, to allow in-network NN execution, without penalizing forwarding procedures.

2) COUPLING OF DIFFERENT COMMUNICATION/NETWORKING TECHNIQUES IS HIGHLY ADVISED

It can be further stated that to effectively overstep the limitations of wireless communications, different techniques need to be blended to improve the performance of distributed intelligence solutions over the RAN. This is the case of AirComp coupled with either RIS or D2D communications

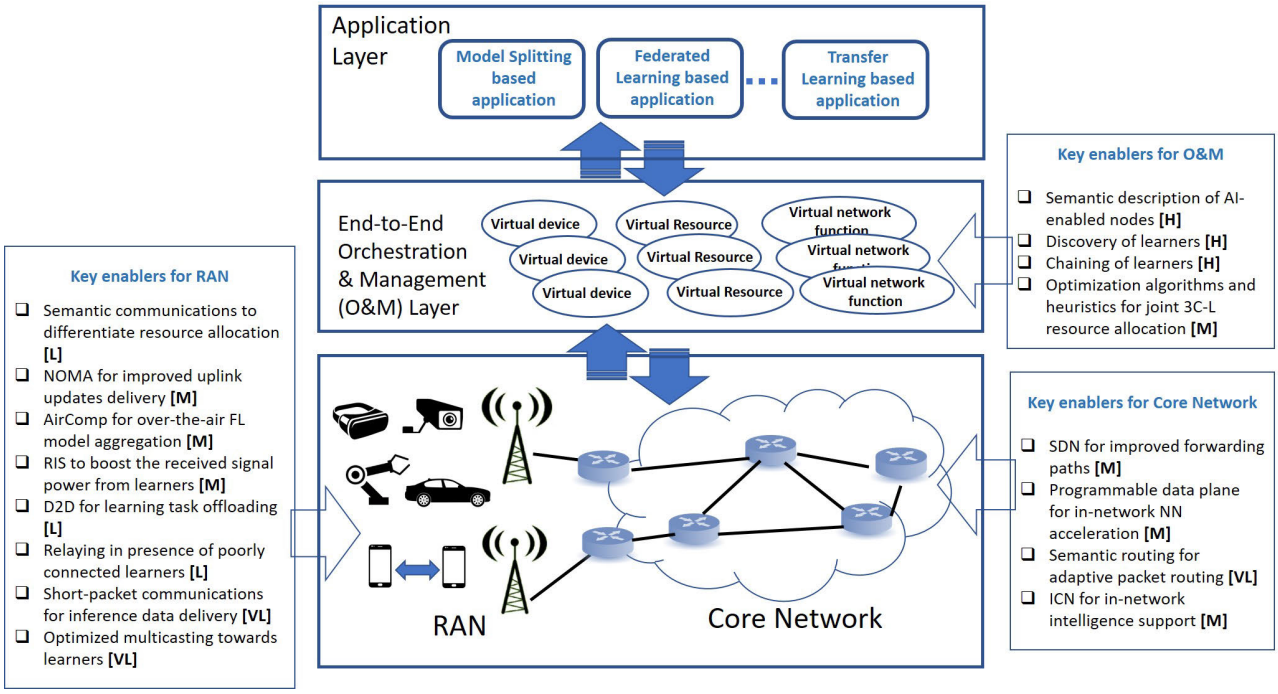

FIGURE 2. Key enablers in different network segments (level of maturity for distributed intelligence support: [VL] = very low, [L] = low, [M] = medium, [H] = high).

and relaying. Synergies between SDN and ICN appear also extremely promising.

3) NETWORK SOLUTIONS SO FAR MAINLY TARGETED FEDERATED LEARNING AND, MORE IN GENERAL, LEARNING PROCEDURES

We noticed that the majority of works focused on the Federated Learning approach, given the inherent benefits it promises and, likely, the push from the industries. However, although less computation- and bandwidth-hungry, also inference will be massively and frequently requested, and it would likely be pervasive for the network, likely as IoT was in the last decade. Hence, inference orchestration and network optimization solutions for it should be conceived.

4) SEVERAL SOLUTIONS, ALTHOUGH PROMISING, ARE STILL GERMINAL

Not all the solutions have the same maturity level, as shown in Fig. 2, where the identified enablers are graphically sketched to provide an end-to-end perspective of a future network supporting distributed intelligence.

In the RAN, there is much room for the design of D2D communication techniques specifically treating distributed intelligence-related data. The same comment holds for short-packet, multicast and semantic communications.

Furthermore, robust solutions against mobility of devices are needed in the RAN segment. For instance, predicting the mobility of source devices may be crucial to improve data (i.e., datasets, model updates) collection, making it reliable also through opportunistic procedures [13].

In the core network segment, we identified semantic routing as a prominent enabler for distributed intelligence, but to the best of our knowledge, no work is available yet. Moreover, despite the maturity of the SDN paradigm *per se*, its potential in dealing with distributed intelligence is still overlooked.

Overall, whatever the network segment, a more conscious distributed intelligence-related data delivery and network procedures adaptation (including 'casting' primitives) to their peculiar needs and features would make the difference.

5) CO-DESIGN OF DISTRIBUTED INTELLIGENCE AND NETWORK OPERATIONS IS MANDATORY

The model splitting point decision, as well as the Federated Learning client selection cannot disregard the (wireless) network dynamics. The behaviour of radio resource allocation schemes and routing protocols should be tightly coupled with the decision concerning the distributed intelligence deployment to appropriately trade-off among accuracy, latency and bandwidth performance.

6) REALISTIC AND COMPREHENSIVE EVALUATION FRAMEWORKS ARE MISSING

A further limitation identified while scanning the literature is the lack of evaluation platforms which couple accurate and realistic simulators, focusing on link/network-layer performance, with the validation of model quality through realistic datasets. Results are typically achieved separately or by loosing in accuracy of the achieved results. A few works target experimental test-beds; although being quite representative of realistic deployments, they barely scale to up to

hundreds/thousands of devices, which represent the more likely case for several distributed intelligence (especially learning) solutions.

B. ADDITIONAL OPEN ISSUES

For the sake of completeness, some socio-economic aspects are also worth discussing which deserve further investigations, although outside the main scope of the manuscript.

1) SECURITY

Distributed AI services further challenge the design of security and privacy-preserving mechanisms. Therefore, to ensure a broad social acceptability of the paradigm, mutual trustworthiness among involved entities, proper access control and authentication schemes for end-users, and secure routing mechanisms need to be enforced, considering that raw (sensitive) data, model (parameters) and intermediate outputs need to be exchanged.

2) SUSTAINABILITY

It is important to consider implications related to environmental sustainability. The overall AI lifecycle should be devised to be distributed throughout the continuum with an eye to the carbon footprint reduction [159]. To this aim, for instance, orchestration should opportunistically deploy AI workloads where green sources are available, increasingly leveraged by telco and cloud providers to power their infrastructures.

3) BUSINESS MODELS

As for business-related issues, the provisioning of distributed AI services on top of existing networks also entails revising the traditional value chain. Either providers of AI-based applications may interact with network operators and cloud/edge providers to offer the aforementioned services, or new operators may enter the scene to offer such kind of distributed and possibly green AI services. Solid business models should be conceived accordingly, which may provide new revenue opportunities and stimulate cooperation among all players in the envisioned ecosystem.

IX. CONCLUSIVE REMARKS

In this paper, we first identified the most representative distributed intelligence solutions and the main relevant issues, with special focus on networking aspects. Then, we provided a comprehensive and end-to-end analysis of the key enablers, from the RAN to the core, for the design of a future network supporting distributed intelligence. Despite the infancy of the topic, several research works can be found in the literature. This paper focused on classifying the most representative solutions for each identified enabler, without having the claim to be exhaustive, but with the aim to provide a valuable support to newcomers to the topic as well as to experienced researchers both from the AI and the networking communities.

Indeed, the intriguing challenges for building future network ecosystems supporting distributed intelligence are multidisciplinary and span different research areas having different maturity levels. Hence, synergies among different communities need to be fostered.

REFERENCES

[1] *Cisco Annual Internet Report 2018–2023*, Cisco, San Jose, CA, USA, 2020.
[2] J. Wan, J. Yang, Z. Wang, and Q. Hua, "Artificial intelligence for cloud-assisted smart factory," *IEEE Access*, vol. 6, pp. 55419–55430, 2018.
[3] T. Huynh-The, Q.-V. Pham, X.-Q. Pham, T. T. Nguyen, Z. Han, and D.-S. Kim, "Artificial intelligence for the metaverse: A survey," *Eng. Appl. Artif. Intell.*, vol. 117, Jan. 2023, Art. no. 105581.
[4] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4316–4336, Jul. 2021.
[5] G. Rong, A. Mendez, E. Bou Assi, B. Zhao, and M. Sawan, "Artificial intelligence in healthcare: Review and prediction case studies," *Engineering*, vol. 6, no. 3, pp. 291–301, Mar. 2020.
[6] J. Verbraeken, "A survey on distributed machine learning," *ACM Comput. Surv.*, vol. 53, no. 2, pp. 1–33, 2020.
[7] T. S. Salem, G. Castellano, G. Neglia, F. Pianese and A. Araldo, "Towards inference delivery networks: Distributing machine learning with optimality guarantees," in *Proc. 19th Medit. Commun. Comput. Netw. Conf. (MedComNet)*, Jun. 2021, pp. 1–8.
[8] Z. Cheng, X. Fan, M. Liwang, M. Min, X. Wang, and X. Du, "Hybrid architectures for distributed machine learning in heterogeneous wireless networks," 2022, *arXiv:2206.01906*.
[9] D. Rosendo, A. Costan, P. Valduriez, and G. Antoniu, "Distributed intelligence on the edge-to-cloud continuum: A systematic literature review," *J. Parallel Distrib. Comput.*, vol. 166, pp. 71–94, Aug. 2022.
[10] Y. Matsubara, M. Levorato, and F. Restuccia, "Split computing and early exiting for deep learning applications: Survey and research challenges," *ACM Comput. Surveys*, vol. 55, no. 5, pp. 1–30, May 2023.
[11] S. Talwar, N. Himayat, H. Nikopour, F. Xue, G. Wu, and V. Ilderem, "6G: Connectivity in the era of distributed intelligence," *IEEE Commun. Mag.*, vol. 59, no. 11, pp. 45–50, Nov. 2021.
[12] F. R. Yu, "From information networking to intelligence networking: Motivations, scenarios, and challenges," *IEEE Netw.*, vol. 35, no. 6, pp. 209–216, Nov. 2021.
[13] J. Pan, L. Cai, S. Yan, and X. S. Shen, "Network for AI and AI for network: Challenges and opportunities for learning-oriented networks," *IEEE Netw.*, vol. 35, no. 6, pp. 270–277, Nov. 2021.
[14] X. Li, R. Xie, F. R. Yu, T. Huang, and Y. Liu, "Advancing software-defined service-centric networking toward in-network intelligence," *IEEE Netw.*, vol. 35, no. 5, pp. 210–218, Sep. 2021.
[15] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6G: Vision, enabling technologies, and applications," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 5–36, Jan. 2022.
[16] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, "Distributed learning in wireless networks: Recent progress and future challenges," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3579–3605, Dec. 2021.
[17] S. Ouyang, D. Dong, Y. Xu, and L. Xiao, "Communication optimization strategies for distributed deep neural network training: A survey," *J. Parallel Distrib. Comput.*, vol. 149, pp. 52–65, Mar. 2021.
[18] X. Cao, T. Basar, S. Diggavi, Y. C. Eldar, K. B. Letaief, H. V. Poor, and J. Zhang, "Communication-efficient distributed learning: An overview," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 851–873, Apr. 2023.
[19] Z. Zhang, C. Chang, H. Lin, Y. Wang, R. Arora, and X. Jin, "Is network the bottleneck of distributed training?" in *Proc. Workshop Netw. Meets AI ML*, Aug. 2020, pp. 8–13.
[20] M. C. Luizelli, R. Canofre, A. F. Lorenzon, F. D. Rossi, W. Cordeiro, and O. M. Caicedo, "In-network neural networks: Challenges and opportunities for innovation," *IEEE Netw.*, vol. 35, no. 6, pp. 68–74, Nov. 2021.
[21] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 4th Quart., 2019.
[22] S. Hu, X. Chen, W. Ni, E. Hossain, and X. Wang, "Distributed machine learning for wireless communication networks: Techniques, architectures, and applications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1458–1493, 3rd Quart., 2021.

[23] H. Yang, A. Alphones, Z. Xiong, D. Niyato, J. Zhao, and K. Wu, "Artificial-intelligence-enabled intelligent 6G networks," *IEEE Netw.*, vol. 34, no. 6, pp. 272–280, Nov. 2020.
[24] D. M. Gutierrez-Estevez, M. Gramaglia, A. D. Domenico, G. Dandachi, S. Khatibi, D. Tsolkas, I. Balan, A. Garcia-Saavedra, U. Elzur, and Y. Wang, "Artificial intelligence for elastic management and orchestration of 5G networks," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 134–141, Oct. 2019.
[25] S. Zhang and D. Zhu, "Towards artificial intelligence enabled 6G: State of the art, challenges, and opportunities," *Comput. Netw.*, vol. 183, Dec. 2020, Art. no. 107556.
[26] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 55–61, Mar. 2020.
[27] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, Q. Le, and A. Ng, "Large scale distributed deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–12.
[28] J. Chen and X. Ran, "Deep learning with edge computing: A review," *Proc. IEEE*, vol. 107, no. 8, pp. 1655–1674, Aug. 2019.
[29] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 869–904, 2nd Quart., 2020.
[30] J. Chen, K. Li, Q. Deng, K. Li, and P. S. Yu, "Distributed deep learning model for intelligent video surveillance systems with edge computing," *IEEE Trans. Ind. Informat.*, early access, Apr. 4, 2019, doi: 10.1109/TII.2019.2909473.
[31] O. Gupta and R. Raskar, "Distributed learning of deep neural network over multiple agents," *J. Netw. Comput. Appl.*, vol. 116, pp. 1–8, Aug. 2018.
[32] E. Li, Z. Zhou, and X. Chen, "Edge intelligence: On-demand deep learning model co-inference with device-edge synergy," in *Proc. Workshop Mobile Edge Commun.*, Aug. 2018, pp. 31–36.
[33] J. Shao and J. Zhang, "Communication-computation trade-off in resource-constrained edge inference," *IEEE Commun. Mag.*, vol. 58, no. 12, pp. 20–26, Dec. 2020.
[34] H. Li, C. Hu, J. Jiang, Z. Wang, Y. Wen, and W. Zhu, "JALAD: Joint accuracy-and latency-aware deep structure decoupling for edge-cloud execution," in *Proc. IEEE 24th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2018, pp. 671–678.
[35] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019.
[36] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.
[37] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140699–140725, 2020.
[38] Y. Shi, Y. Zhou, and Y. Shi, "Over-the-air decentralized federated learning," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2021, pp. 455–460.
[39] E. T. M. Beltrán, M. Q. Pérez, P. M. S. Sánchez, S. L. Bernal, G. Bovet, M. G. Pérez, G. M. Pérez, and A. H. Celdrán, "Decentralized federated learning: Fundamentals, state-of-the-art, frameworks, trends, and challenges," 2022, *arXiv:2211.08413*.
[40] L. Barbieri, S. Savazzi, M. Brambilla, and M. Nicoli, "Decentralized federated learning for extended sensing in 6G connected vehicles," *Veh. Commun.*, vol. 33, Jan. 2022, Art. no. 100396.
[41] Y. Xiao, Y. Ye, S. Huang, L. Hao, Z. Ma, M. Xiao, S. Mumtaz, and O. A. Dobre, "Fully decentralized federated learning-based on-board mission for UAV swarm system," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3296–3300, Oct. 2021.
[42] C. Thapa, P. C. M. Arachchige, S. Camtepe, and L. Sun, "SplitFed: When federated learning meets split learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 8, 2022, pp. 8485–8493.
[43] A. Sufian, A. Ghosh, A. S. Sadiq, and F. Smarandache, "A survey on deep transfer learning to edge computing for mitigating the COVID-19 pandemic," *J. Syst. Archit.*, vol. 108, Sep. 2020, Art. no. 101830.
[44] L. Valerio, A. Passarella, and M. Conti, "Accuracy vs. traffic trade-off of learning IoT data patterns at the edge with hypothesis transfer learning," in *Proc. IEEE 2nd Int. Forum Res. Technol. Soc. Ind. Leveraging Better Tomorrow (RTSI)*, Sep. 2016, pp. 1–6.
[45] K. I. Wang, X. Zhou, W. Liang, Z. Yan, and J. She, "Federated transfer learning based cross-domain prediction for smart manufacturing," *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 4088–4096, Jun. 2022.
[46] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2009.
[47] Y. Wang, M. Damani, P. Wang, Y. Cao, and G. Sartoretti, "Distributed reinforcement learning for robot teams: A review," 2022, *arXiv:2204.03516*.
[48] F. Venturini, F. Mason, F. Pase, F. Chiariotti, A. Testolin, A. Zanella, and M. Zorzi, "Distributed reinforcement learning for flexible and efficient UAV swarm control," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 3, pp. 955–969, Sep. 2021.
[49] M. Spryn, A. Sharma, D. Parkar, and M. Shrimal, "Distributed deep reinforcement learning on the cloud for autonomous driving," in *Proc. IEEE/ACM 1st Int. Workshop Softw. Eng. for AI Auto. Syst. (SEFAIAS)*, May 2018, pp. 16–22.
[50] T. Chen, K. Zhang, G. B. Giannakis, and T. Basar, "Communication-efficient policy gradient methods for distributed reinforcement learning," *IEEE Trans. Control Netw. Syst.*, vol. 9, no. 2, pp. 917–929, Jun. 2022.
[51] N. Tonellotto, A. Gotta, F. M. Nardini, D. Gadler, and F. Silvestri, "Neural network quantization in federated learning at the edge," *Inf. Sci.*, vol. 575, pp. 417–436, Oct. 2021.
[52] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and A. Askell, "Language models are few-shot learners," in *Proc. Adv. Neur. Inf. Process. Sys.*, vol. 33, 2020, pp. 1877–1901.
[53] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained IoT devices," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 1–24, Jan. 2022.
[54] T. Zeng, O. Semiari, M. Chen, W. Saad, and M. Bennis, "Federated learning on the road autonomous controller design for connected and autonomous vehicles," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10407–10423, Dec. 2022.
[55] Y. Liu, J. Nie, X. Li, S. H. Ahmed, W. Y. B. Lim, and C. Miao, "Federated learning in the sky: Aerial-ground air quality sensing framework with UAV swarms," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9827–9837, Jun. 2021.
[56] A. Xie and Y. Peng, "Improving the quality of inference for applications using chained DNN models during edge server handover," in *Proc. IEEE/ACM 7th Symp. Edge Comput. (SEC)*, Dec. 2022, pp. 516–520.
[57] E. Ramos, R. Morabito, and J. Kainulainen, "Distributing intelligence to the edge and beyond [research Frontier]," *IEEE Comput. Intell. Mag.*, vol. 14, no. 4, pp. 65–92, Nov. 2019.
[58] X. Qi and C. Liu, "Enabling deep learning on IoT edge: Approaches and evaluation," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 367–372.
[59] C. Mwase, Y. Jin, T. Westerlund, H. Tenhunen, and Z. Zou, "Communication-efficient distributed AI strategies for the IoT edge," *Future Gener. Comput. Syst.*, vol. 131, pp. 292–308, Jun. 2022.
[60] C. Wu, F. Wu, L. Lyu, Y. Huang, and X. Xie, "Communication-efficient federated learning via knowledge distillation," *Nature Commun.*, vol. 13, no. 1, pp. 1–8, Apr. 2022.
[61] Y. Shi, K. Yang, T. Jiang, J. Zhang, and K. B. Letaief, "Communication-efficient edge AI: Algorithms and systems," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2167–2191, 4th Quart., 2020.
[62] J. Park, S. Samarakoon, A. Elgabli, J. Kim, M. Bennis, S. Kim, and M. Debbah, "Communication-efficient and distributed learning over wireless networks: Principles and applications," *Proc. IEEE*, vol. 109, no. 5, pp. 796–819, May 2021.
[63] D. Liu, G. Zhu, J. Zhang, and K. Huang, "Data-importance aware user scheduling for communication-efficient edge machine learning," *IEEE Trans. Cognit. Commun. Netw.*, vol. 7, no. 1, pp. 265–278, Mar. 2021.
[64] D. Liu, G. Zhu, Q. Zeng, J. Zhang, and K. Huang, "Wireless data acquisition for edge learning: Data-importance aware retransmission," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 406–420, Jan. 2021.
[65] Y. He, J. Ren, G. Yu, and J. Yuan, "Importance-aware data selection and resource allocation in federated edge learning system," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13593–13605, Nov. 2020.
[66] H. H. Yang, A. Arafa, T. Q. S. Quek, and H. Vincent Poor, "Age-based scheduling policy for federated learning in mobile edge networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8743–8747.

[67] H. Sun, X. Ma, and R. Q. Hu, "Adaptive federated learning with gradient compression in uplink NOMA," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 16325–16329, Dec. 2020.

[68] X. Ma, H. Sun, and R. Q. Hu, "Scheduling policy and power allocation for federated learning in NOMA based MEC," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–7.

[69] P. S. Bouzinis, P. D. Diamantoulakis, and G. K. Karagiannidis, "Wireless federated learning (WFL) for 6G networks—Part II: The compute-then-transmit NOMA paradigm," *IEEE Commun. Lett.*, vol. 26, no. 1, pp. 8–12, Jan. 2022.

[70] I. Mrad, R. Hamila, A. Erbad, and M. Gabbouj, "Joint learning and optimization for federated learning in NOMA-based networks," *Pervas. Mobile Comput.*, vol. 89, Feb. 2023, Art. no. 101739.

[71] Y. Wu, Y. Song, T. Wang, L. Qian, and T. Q. S. Quek, "Non-orthogonal multiple access assisted federated learning via wireless power transfer: A cost-efficient approach," *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2853–2869, Apr. 2022.

[72] Y. Song, T. Wang, Y. Wu, L. Qian, and Z. Shi, "Non-orthogonal multiple access assisted federated learning for UAV swarms: An approach of latency minimization," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2021, pp. 1123–1128.

[73] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.

[74] Y. Yang, Z. Zhang, Y. Tian, Z. Yang, C. Huang, C. Zhong, and K. Wong, "Over-the-air split machine learning in wireless MIMO networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 1007–1022, Apr. 2023.

[75] Q. N. Le, L. Bariah, O. A. Dobre, and S. Muhaidat, "Reconfigurable intelligent surface-enabled federated learning for power-constrained devices," *IEEE Commun. Lett.*, vol. 26, no. 11, pp. 2725–2729, Nov. 2022.

[76] H. Liu, X. Yuan, and Y. A. Zhang, "Reconfigurable intelligent surface enabled federated learning: A unified communication-learning design approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7595–7609, Nov. 2021.

[77] K. Yang, Y. Shi, Y. Zhou, Z. Yang, L. Fu, and W. Chen, "Federated machine learning for intelligent IoT via reconfigurable intelligent surface," *IEEE Netw.*, vol. 34, no. 5, pp. 16–22, Sep. 2020.

[78] Z. Wang, J. Qiu, Y. Zhou, Y. Shi, L. Fu, W. Chen, and K. B. Letaief, "Federated learning via intelligent reflecting surface," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 808–822, Feb. 2022.

[79] W. Ni, Y. Liu, Z. Yang, H. Tian, and X. Shen, "Federated learning in multi-RIS-aided systems," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9608–9624, Jun. 2022.

[80] W. Ni, Y. Liu, H. Tian, Y. C. Eldar, and K. Huang, "SemiFL: Semi-federated learning empowered by simultaneously transmitting and reflecting reconfigurable intelligent surface," in *Proc. IEEE Int. Conf. Commun.*, May 2022, pp. 5104–5109.

[81] S. Wang, M. Lee, S. Hosseinalipour, R. Morabito, M. Chiang, and C. G. Brinton, "Device sampling for heterogeneous federated learning: Theory, algorithms, and implementation," in *Proc. INFOCOM IEEE Conf. Comput. Commun.*, May 2021, pp. 1–10.

[82] X. Cai, X. Mo, J. Chen, and J. Xu, "D2D-enabled data sharing for distributed machine learning at wireless network edge," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1457–1461, Sep. 2020.

[83] S. Savazzi, M. Nicoli, and V. Rampa, "Federated learning with cooperating devices: A consensus approach for massive IoT networks," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4641–4654, May 2020.

[84] Y. Huang, X. Qiao, W. Lai, S. Dustdar, J. Zhang, and J. Li, "Enabling DNN acceleration with data and model parallelization over ubiquitous end devices," *IEEE Internet Things J.*, vol. 9, no. 16, pp. 15053–15065, Aug. 2022.

[85] M. Yemini, R. Saha, E. Ozfatura, D. Gündüz, and A. J. Goldsmith, "Semi-decentralized federated learning with collaborative relaying," 2022, *arXiv:2205.10998*.

[86] M. Fu, Y. Shi, and Y. Zhou, "Federated learning via unmanned aerial vehicle," 2022, *arXiv:2210.10970*.

[87] Z. Lin, H. Liu, and Y. A. Zhang, "Relay-assisted cooperative federated learning," *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 7148–7164, Sep. 2022.

[88] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.

[89] D. Shome, O. Waqar, and W. U. Khan, "Federated learning and next generation wireless communications: A survey on bidirectional relationship," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 7, p. e4458, Jul. 2022.

[90] S. Sorour, U. Mohammad, A. Abutuleb, and H. Hassanein, "Returning the favor: What wireless networking can offer to AI and edge learning," 2020, *arXiv:2006.07453*.

[91] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Found. Trends Netw.*, vol. 12, no. 3, pp. 162–259, 2017.

[92] E. Calvanese Strinati and S. Barbarossa, "6G networks: Beyond Shannon towards semantic and goal-oriented communications," *Comput. Netw.*, vol. 190, May 2021, Art. no. 107930.

[93] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan, and V. K. Bhargava, "A survey on non-orthogonal multiple access for 5G networks: Research challenges and future trends," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 10, pp. 2181–2195, Oct. 2017.

[94] A. Sahin and R. Yang, "A survey on over-the-air computation," *IEEE Commun. Surveys Tuts.*, early access, Apr. 5, 2023, doi: 10.1109/COMST.2023.3264649.

[95] Y. Liu, X. Liu, X. Mu, T. Hou, J. Xu, M. Di Renzo, and N. Al-Dhahir, "Reconfigurable intelligent surfaces: Principles and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1546–1577, 3rd Quart., 2021.

[96] C. Huang, R. Mo, and C. Yuen, "Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1839–1850, Aug. 2020.

[97] F. Jameel, Z. Hamid, F. Jabeen, S. Zeadally, and M. A. Javed, "A survey of device-to-device communications: Research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2133–2168, 3rd Quart., 2018.

[98] Z. Lin, H. Liu, and Y.-J. A. Zhang, "Relay-assisted over-the-air federated learning," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2021, pp. 1–7.

[99] W. Ren, Y. Qu, C. Dong, Y. Jing, H. Sun, Q. Wu, and S. Guo, "A survey on collaborative DNN inference for edge intelligence," 2022, *arXiv:2207.07812*.

[100] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultrareliable, and low-latency wireless communication with short packets," *Proc. IEEE*, vol. 104, no. 9, pp. 1711–1726, Sep. 2016.

[101] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.

[102] X. Chen, G. Zhu, Y. Deng, and Y. Fang, "Federated learning over multihop wireless networks with in-network aggregation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4622–4634, Jun. 2022.

[103] T. T. Vu, H. Quoc Ngo, T. L. Marzetta, and M. Matthaiou, "How does cell-free massive MIMO support multiple federated learning groups?" in *Proc. IEEE 22nd Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sep. 2021, pp. 401–405.

[104] Y. Lin and S. Luo, "Poster: Accelerate cross-device federated learning with semi-reliable model multicast over the air," in *Proc. IEEE 29th Int. Conf. Netw. Protocols (ICNP)*, Nov. 2021, pp. 1–2.

[105] V. K. Shrivastava, S. Baek, and Y. Baek, "5G evolution for multicast and broadcast services in 3GPP release 17," *IEEE Commun. Standards Mag.*, vol. 6, no. 3, pp. 70–76, Sep. 2022.

[106] E. F. Pupo, C. C. González, L. Atzori, and M. Murroni, "Dynamic multicast access technique in SC-PTM 5G networks: Subgrouping with OM/NOM," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2022, pp. 1–6.

[107] E. Seo, D. Niyato, and E. Elmroth, "Auction-based federated learning using software-defined networking for resource efficiency," in *Proc. 17th Int. Conf. Netw. Service Manage. (CNSM)*, Oct. 2021, pp. 42–48.

[108] S. K. P. Tam and S. Math, "Efficient resource slicing scheme for optimizing federated learning communications in software-defined IoT networks," *J. Internet Comput. Services*, vol. 22, no. 5, pp. 27–33, 2021.

[109] V. Balasubramanian, M. Aloqaily, M. Reisslein, and A. Scaglione, "Intelligent resource management at the edge for ubiquitous IoT: An SDN-based federated learning approach," *IEEE Netw.*, vol. 35, no. 5, pp. 114–121, Sep. 2021.

[110] D. Aguiari, A. Ferlini, J. Cao, S. Guo, and G. Pau, "Poster abstract: C-continuum: Edge-to-cloud computing for distributed AI," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 1053–1054.
[111] C. Campolo, G. Lia, M. Amadeo, G. Ruggeri, A. Iera, and A. Molinaro, "Towards named AI networking: Unveiling the potential of NDN for edge AI," in *Proc. AdHocNow*, 2020, pp. 16–22.
[112] M. Amadeo, C. Campolo, A. Iera, A. Molinaro, and G. Ruggeri, "Client discovery and data exchange in edge-based federated learning via named data networking," in *Proc. IEEE Int. Conf. Commun.*, May 2022, pp. 2990–2995.
[113] S. Bano, N. Tonellotto, P. Cassara, and A. Gotta, "KafkaFed: Two-tier federated learning communication architecture for Internet of Vehicles," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops Other Affiliated Events (PerCom Workshops)*, Mar. 2022, pp. 515–520.
[114] Z. Xiong and N. Zilberman, "Do switches dream of machine learning? Toward in-network classification," in *Proc. 18th ACM Workshop Hot Topics Netw.*, Nov. 2019, pp. 25–33.
[115] D. Sanvito, G. Siracusano, and R. Bifulco, "Can the network be the AI accelerator?" in *Proc. Morning Workshop In-Network Comput.*, Aug. 2018, pp. 20–25.
[116] Y. Li, J. Park, M. Alian, Y. Yuan, Z. Qu, P. Pan, and R. Wang, "A network-centric hardware/algorithm co-design to accelerate distributed training of deep neural networks," *IEEE/ACM MICRO*, Oct. 2018, pp. 175–188.
[117] M. Saquetti, R. Canofre, A. F. Lorenzon, F. D. Rossi, J. R. Azambuja, W. Cordeiro, and M. C. Luizelli, "Toward in-network intelligence: Running distributed artificial neural networks in the data plane," *IEEE Commun. Lett.*, vol. 25, no. 11, pp. 3551–3555, Nov. 2021.
[118] B. M. Xavier, R. S. Guimarães, G. Comarela, and M. Martinello, "Programmable switches for in-networking classification," in *Proc. IEEE Conf. Comput. Commun.*, May 2021, pp. 1–10.
[119] A. Sapio, M. Canini, C.-Y. Ho, J. Nelson, P. Kalnis, C. Kim, A. Krishnamurthy, M. Moshref, D. R. K. Ports, and P. Richtárik, "Scaling distributed machine learning with in-network aggregation," 2019, *arXiv:1903.06701*.
[120] Y. Li, I. Liu, Y. Yuan, D. Chen, A. Schwing, and J. Huang, "Accelerating distributed reinforcement learning with in-switch computing," in *Proc. ACM/IEEE 46th Annu. Int. Symp. Comput. Archit. (ISCA)*, Jun. 2019, pp. 279–291.
[121] J. He, H. Wu, X. Xiao, R. Bassoli, and F. H. P. Fitzek, "Functional split of in-network deep learning for 6G: A feasibility study," *IEEE Wireless Commun.*, vol. 29, no. 5, pp. 36–42, Oct. 2022.
[122] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
[123] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, C. Wang, and Y. Liu, "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 393–430, 1st Quart., 2019.
[124] S. Islam, N. Muslim, and J. W. Atwood, "A survey on multicasting in software-defined networking," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 355–387, 1st Quart., 2018.
[125] D. King and A. Farrel, "A survey of semantic internet routing techniques," Internet Eng. Task Force (IETF), Draft-King-Irtf-Semantic-Survey-01, Tech. Rep., 2022.
[126] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, Jul. 2014.
[127] O. Michel, R. Bifulco, G. Rétvári, and S. Schmid, "The programmable data plane: Abstractions, architectures, algorithms, and applications," *ACM Comput. Surveys*, vol. 54, no. 4, pp. 1–36, 2021.
[128] T. Mai, S. Garg, H. Yao, J. Nie, G. Kaddoum, and Z. Xiong, "In-network intelligence control: Toward a self-driving networking architecture," *IEEE Netw.*, vol. 35, no. 2, pp. 53–59, Mar. 2021.
[129] G. Siracusano and R. Bifulco, "In-network neural networks," 2018, *arXiv:1801.05731*.
[130] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 66–73, Jul. 2014.
[131] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 1024–1049, 2nd Quart., 2014.
[132] M. Amadeo, C. Campolo, A. Molinaro, and G. Ruggeri, "IoT data processing at the edge with named data networking," in *Proc. Eur. Wireless 24th Eur. Wireless Conf.*, May 2018, pp. 1–6.
[133] A. Rahman, D. Trossen, D. Kutscher, and R. Ravindran, *Deployment Considerations for Information-Centric Networking (ICN)*, document RFC 8763, 2020.
[134] S. Shahhosseini, D. Seo, A. Kanduri, T. Hu, S.-S. Lim, B. Donyanavard, A. M. Rahmani, and N. Dutt, "Online learning for orchestration of inference in multi-user end-edge-cloud networks," *ACM Trans. Embedded Comput. Syst.*, vol. 21, no. 6, pp. 1–25, Nov. 2022.
[135] F. Malandrino, C. F. Chiasserini, and G. di Giacomo, "Efficient distributed DNNs in the mobile-edge-cloud continuum," *IEEE/ACM Trans. Netw.*, early access, Nov. 24, 2022, doi: 10.1109/TNET.2022.3222640.
[136] E. Samikwa, A. D. Maio, and T. Braun, "ARES: Adaptive resource-aware split learning for Internet of Things," *Comput. Netw.*, vol. 218, Dec. 2022, Art. no. 109380.
[137] S. Wang, X. Zhang, H. Uchiyama, and H. Matsuda, "HiveMind: Towards cellular native machine learning model splitting," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 2, pp. 626–640, Feb. 2022.
[138] M. M. Wadu, S. Samarakoon, and M. Bennis, "Joint client scheduling and resource allocation under channel uncertainty in federated learning," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 5962–5974, Sep. 2021.
[139] L. Yu, R. Albelaihi, X. Sun, N. Ansari, and M. Devetsikiotis, "Jointly optimizing client selection and resource management in wireless federated learning for Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 6, pp. 4385–4395, Mar. 2022.
[140] H. Ko, J. Lee, S. Seo, S. Pack, and V. C. M. Leung, "Joint client selection and bandwidth allocation algorithm for federated learning," *IEEE Trans. Mobile Comput.*, vol. 22, no. 6, pp. 3380–3390, Jun. 2023.
[141] J. Xu, H. Wang, and L. Chen, "Bandwidth allocation for multiple federated learning services in wireless edge networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2534–2546, Apr. 2022.
[142] C. Campolo, G. Genovese, A. Iera, and A. Molinaro, "Virtualizing AI at the distributed edge towards intelligent IoT applications," *J. Sensor Actuator Netw.*, vol. 10, no. 1, p. 13, Feb. 2021.
[143] C. Campolo, G. Genovese, G. Singh, and A. Molinaro, "Scalable and interoperable edge-based federated learning in IoT contexts," *Comput. Netw.*, vol. 223, Mar. 2023, Art. no. 109576.
[144] T. Yang, M. Qin, N. Cheng, W. Xu, and L. Zhao, "Liquid software-based edge intelligence for future 6G networks," *IEEE Netw.*, vol. 36, no. 1, pp. 69–75, Jan. 2022.
[145] S. Abdulrahman, H. Tout, A. Mourad, and C. Talhi, "FedMCCS: Multicriteria client selection model for optimal IoT federated learning," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4723–4735, Mar. 2021.
[146] W. Wu, C. Zhou, M. Li, H. Wu, H. Zhou, N. Zhang, X. S. Shen, and W. Zhuang, "AI-native network slicing for 6G networks," *IEEE Wireless Commun.*, vol. 29, no. 1, pp. 96–103, Feb. 2022.
[147] A. Kanduri, S. Shahhosseini, E. Kasaeyan Naeini, H. Alikhani, P. Liljeberg, N. Dutt, and A. M. Rahmani, "Edge-centric optimization of multimodal ML-driven eHealth applications," 2022, *arXiv:2208.02597*.
[148] M. Nitti, V. Pilloni, G. Colistra, and L. Atzori, "The virtual object as a major element of the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1228–1240, 2nd Quart., 2016.
[149] *Open Mobile Alliance, Lightweight Machine to Machine Technical Specification Core; v1_1-20180612-c*, Open Mobile Alliance (OMA), 2018.
[150] *Focus Group on Technologies for Network 2030, Additional Representative Use Cases and Key Network Requirements for Network 2030*, document ITU FG-NET2030, Jun. 2020. Accessed: May 29, 2023. [Online]. Available: https://datatracker.ietf.org/rg/coinrg/about/
[151] *IETF Computing in the Network Research Group (COINRG)*. Accessed: May 29, 2023. [Online]. Available: https://datatracker.ietf.org/rg/coinrg/about/
[152] *One6G White Paper: 6G Technology Overview*. Accessed: May 29, 2023. [Online]. Available: https://www.itu.int/en/ITU-T/focusgroups/ml5g/Pages/default.aspx
[153] *ITU-T Focus Group on Machine Learning for Future Networks including 5G*.
[154] *Architecture Enhancements for 5G System (5GS) to Support Network Data Analytics Services*, document TS 23.288, V16.2.0, Rel. 16, 3GPP, Dec. 2019.
[155] *Technical Specification Group Services and System Aspects; Study on Traffic Characteristics and Performance Requirements for AI/ML Model Transfer in 5GS; Release 18*, document TR 22.784, V17.1.1, 3GPP, Dec. 2021. Accessed: Feb. 28, 2023.

[156] *DAIS, Distributed Artificial Intelligent System*. Accessed: Feb. 28, 2023. [Online]. Available: https://dais-project.eu/
[157] *DEDICAT6G, Dynamic Coverage Extension and Distributed Intelligence for Human Centric Applications With Assured Security, Privacy, and Trust: From 5G to 6G*. Accessed: Feb. 28, 2023. [Online]. Available: https://dedicat6g.eu/
[158] *AI@edge, A Secure and Reusable Artificial Intelligence Platform for Edge Computing in Beyond 5G Networks*. Accessed: Feb. 28, 2023. [Online]. Available: https://aiatedge.eu/
[159] C.-J. Wu, "Sustainable AI: Environmental implications, challenges and opportunities," in *Proc. Mach. Learn. Syst.*, vol. 4, 2022, pp. 795–813.

CLAUDIA CAMPOLO (Senior Member, IEEE) is currently an Associate Professor of telecommunications with the University Mediterranea of Reggio Calabria, Italy. Her research interests include vehicular networking, 5G/6G, and future internet architectures.

ANTONIO IERA (Senior Member, IEEE) is currently a Full Professor of telecommunications with the University of Calabria, Italy. His research interests include next generation mobile and wireless systems and the Internet of Things.

ANTONELLA MOLINARO (Senior Member, IEEE) is currently a Full Professor of telecommunications with the University Mediterranea of Reggio Calabria, Italy, and Université Paris-Saclay, France. Her current research interests include 5G and beyond networks, connected vehicles, and the future internet.

Open Access funding provided by 'Università della Calabria' within the CRUI CARE Agreement