

Dynamic Resource Allocation for Multi-User Goal-oriented Communications at the Wireless Edge

Francesco Binucci¹, Paolo Banelli², Paolo Di Lorenzo², Sergio Barbarossa²

¹Department of Engineering, University of Perugia, Via G. Duranti 93, 06128, Perugia, Italy

²DIET department, Sapienza University of Rome, via Eudossiana 18, 00184, Rome, Italy

email: francesco.binucci@studenti.unipg.it, paolo.banelli@unipg.it, {paolo.dilorenzo, sergio.barbarossa}@uniroma1.it

Abstract—This paper proposes a wireless, goal-oriented, multi-user communication system assisted by edge-computing, within the general framework of Edge Machine Learning (EML). Specifically, we consider a set of mobile devices that, exploiting convolutional encoders (CE), namely the encoder part of the convolutional auto-encoders (CAE), send compressed data units to an edge server (ES) that performs a specific learning task, such as image classification. The training of both the CEs and the ES classification networks is performed in a off-line fashion, employing a cross-entropy loss, regularized by the mean squared error of the CAE expanded output. Then, exploiting such goal-oriented architecture, and employing a Lyapunov optimization framework, we considered the joint management of computation and transmission resources for the overall system. In particular, we considered a Multi-User Minimum Energy Resource Allocation Strategy (mu-MERAS), which provides the optimal resource allocation for both the devices and the ES, in a energy-efficient perspective. Simulation results highlight a classical EML trade-off between energy, latency, and accuracy, as well as the effectiveness of the proposed approach to adaptively manage resources according to wireless channels conditions, computing requests, and classification reliability.

Index Terms—Goal-Oriented communications, auto-encoder, Lyapunov Optimization, Edge Machine Learning

I. INTRODUCTION

The deployment of 6G is going to radically change the concept of mobile networks, from a pure communication perspective to a key enabler of pervasive and “zero-latency” artificial intelligence controlling several new services, among which Industry 4.0, autonomous driving, augmented reality, and Internet of Things [1]. As such, 6G requires the design of a *holistic* system in which communication, computation, learning, and control are jointly orchestrated to achieve new target levels of reliability, energy efficiency, and sustainability. To this aim, edge machine learning (EML) is becoming a hot research topic [2]–[5], enabling mobile devices (MDs) to opportunistically offload learning tasks to ESs. The crux of EML is a joint management of computational and communication resources at the wireless edge, which can optimize a quantity among energy consumption, latency, or learning accuracy, while guaranteeing specific application constraints on the others [5]. In this framework, *communications are typically goal-oriented*, i.e., take place to fulfil the goal dictated by the specific application, and must be designed to transmit the

minimum information that is necessary to perform the learning task with target levels of quality of service. This requires the development of new communication schemes that can move beyond the classical Shannon paradigm, whose management is a challenging research topic [6].

Related works. Within the EML literature, the authors in [5] describe an offloading strategy that allows to save transmission resources by dynamically allocating the quantization bits used by MDs to transmit their data to the ES, for a specific learning task. More recently, compression at the MD has been proposed by means of the Information Bottleneck (IB) principle [7], which, however, admits a closed form solution only in special cases (e.g., Gaussian IB) and it is not suitable for those scenarios (e.g., image classification) where a closed-form for the stochastic compression does not exist. In this situation, the Variational-IB is more appealing, as in [8], which proposes a goal-oriented compression (GOC) framework. Also, goal-oriented semantic communications were recently proposed in [9], [10]. However, none of these works considered dynamical resources allocation strategies for goal-oriented communications.

Our contributions. we propose a *multi-user*, goal-oriented communication system where multiple MDs exploit banks of Convolutional-Encoders (CEs) (i.e., the encoding section of Convolutional Auto-Encoders) to (dynamically) compress data-units (DUs) that are offloaded to an ES, which has to perform multiple, user-independent, learning tasks, such as image classification, by exploiting banks of convolutional neural networks (CNNs) that are matched to the CEs. Exploiting this dynamical source compression architecture and Lyapunov optimization, we design a multi-user Minimum Energy Resource Allocation Strategy (mu-MERAS), whose aim is to jointly minimize the total energy consumption of MDs and ES, under latency and accuracy guarantees. Simulation results highlight the effectiveness of the proposed approach, as well as the resulting trade-offs between energy, latency, and accuracy.

II. SYSTEM MODEL AND TRAINING PROCEDURE

The system scenario, reported in fig.1, considers K MDs that wirelessly offload to an ES generic, possibly different, learning tasks. Before transmission, the DUs are dynamically compressed by a bank of tunable CEs, which implement a goal-oriented data-size reduction such that each MD transmits

This work was supported by MIUR under the PRIN Liquid Edge Contract.

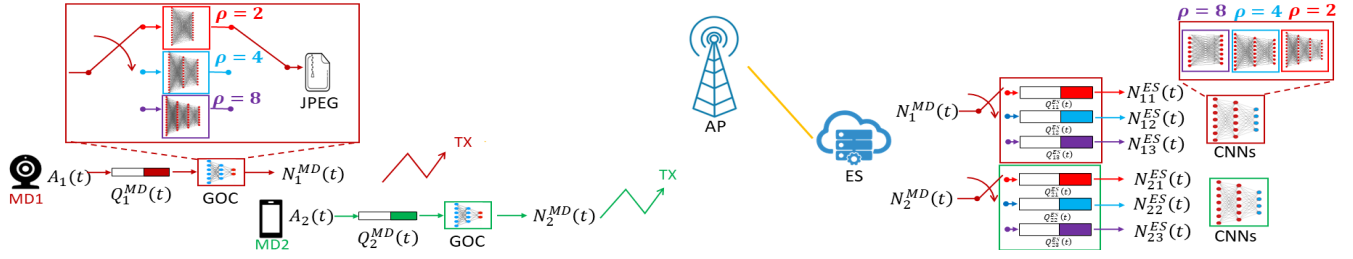


Fig. 1. System Model

the lowest possible amount of data, while ensuring a prescribed learning performance. This idea is formalized in the IB principle, which maps the above requirements into an Information Theoretic perspective [11]. However, due the intractability issues of the statistical quantities (e.g. the Mutual Information) that make the Bottleneck solvable in a closed form only on some special cases [7], some practical approximation should be considered.

A. Compression-oriented training procedure

The joint training of CE and CNN, which is fed by the compressed representation, could be based on the minimization of the *categorical cross-entropy* [12] between the ground truth Y and the prediction \hat{Y} , provided by the CE-CNN classifier. Despite its simplicity, the effectiveness of the proposed training finds its roots in [13], which demonstrates that Cross Entropy can be viewed as a proxy for the *mutual information* involved in the IB principle. Herein, we improve the tuning capability of the resource management, by modifying the CE-CNNs classification in two ways:

- inspired by [14], we penalize the cross entropy training loss by the *Mean Squared Error* (MSE) between the original signal X (e.g., the image to be classified) and the corresponding (re-) expanded CAE output \hat{X} .
- in the bank of parallel encoders, we consider both *short* and *deep* CEs, which are characterized by different complexity (i.e., energy, latency) versus accuracy tradeoffs.

Summarising, the learning procedure reads as

$$\min_{\theta, \phi} \frac{1}{N_t} \sum_{n=1}^{N_t} EL(Y_n, \hat{Y}_n, \phi, \theta) + \lambda L_{mse}(X_n, \hat{X}_n, \theta), \quad (1)$$

where θ and ϕ represent the MD-CAE and the ES-CNN parameters, respectively, EL is the categorical cross Entropy Loss, and L_{mse} is the MSE loss, weighted by the regularization parameter λ . Thus, considering for all the users the same image classification task, after jointly training the CE-CNN structure, we selected by cross-validation the best λ , as the one that maximizes the classification accuracy in the test-set.

B. Communication-oriented compression

After the CE step, each MD employs also a further compression stage, which allows to further zip the (pseudo) DUs at the encoder output, thus resulting in a considerable saving of transmission resources. Obviously, this implies an additional

(energy) computational cost for both the MDs compressions and the ES decompression, which is necessary at the ES for a reliable (CE-based) classification. Specifically, we focused on JPEG because it is efficiently supported by most of the available programming languages and, despite its (slightly) lossy nature, we verified it does not affect considerably the ultimate learning performances.

C. Latency Model

The latency of the system is captured by the congestion of some specific physical queues [15]. The system evolves in (discretized) time-slots indexed by t , with a fixed duration τ . For each user in the system, we define two kinds of queues:

- 1) *MD compression and transmission queues*, which collect the DUs (e.g., images) that are waiting to be compressed and transmitted to the ES for inference processing.
- 2) An *ES computation queue* for each possible compression factor employable by each MD.

We also reasonably assume that:

- i) each DU at the MD queue, has to be (specifically) compressed and transmitted during the same time-slot.
- ii) while the device transmits some DUs, it may also simultaneously compress other DUs, which (necessarily) have to be successively transmitted within the same time-slot.

The number of DUs that the k -th device could transmit during the t -th time-slot is expressed by

$$N_k^{tx}(t) = \left\lfloor \frac{\tau R_k(t)}{M(\rho_k(t))N(\rho_k(t))} \right\rfloor, \quad (2)$$

where $R_k(t)$ and $\rho_k(t)$ are the specifically chosen transmission rate and compression factor, respectively; $M(\rho_k(t))$ is the DU size associated with a specific compression factor $\rho_k(t) \in \mathcal{S}_k$, and $N(\rho_k(t))$ are the corresponding bits used (on average) to encode a pixel of the (JPEG-compressed) CE output.

The number of DUs that a MD could (computationally) process during the t -th time slot is given by

$$N_k^c(t) = \lfloor \tau f_k^d(t) J_d(\rho_k(t)) \rfloor, \quad (3)$$

where $J_d(\rho_k(t))$ denotes the number of DUs compressed in a clock cycle, which depends on the compression factor $\rho_k(t)$, while $f_k^d(t)$ is the MD clock frequency selected for the t -th time slot. Thus, by assumption *i*), the total number of DUs that can be offloaded by the k -th MD during the t -th slot is

$$N_k^{MD}(t) = \min(N_k^{tx}(t), N_k^c(t)). \quad (4)$$

Consequently, the queue $Q_k^{MD}(t)$ at the k -th MD evolves as

$$Q_k^{MD}(t+1) = \max(0, Q_k^{MD}(t) - N_k^{MD}(t)) + A_k(t), \quad (5)$$

where $A_k(t)$ models the DUs arrival process, whose statistical knowledge is not required in Lyapunov optimization [15].

For simpler mathematical tractability, we assume the ES manages L_k queues for each MD, each one associated to one of the compression factors available in \mathcal{S}_k . The evolution of the i -th queue, for the k -th MD, is described by

$$Q_{ki}^{ES}(t+1) = \max(0, Q_{ki}^{ES}(t) - N_{ki}^{ES}(t)) + \min(N_k^{MD}(t), Q_k^{MD}(t)) \mathbb{1}_i\{\rho_k(t)\}, \quad (6)$$

where $\mathbb{1}_i\{\rho_k(t)\}$ denotes the indicator function $\mathbb{1}\{\rho_k(t) = s_{ki}(t)\}$, which models the arrival of new DUs in the i -th queue only if the MD chooses the i -th compression factor, while $N_{ki}^{ES}(t)$ is the number of processed DUs in the t -th slot, expressed by

$$N_{ki}^{ES}(t) = \lfloor \tau f_{ki}^s(t) J_{ki}^s(t) \rfloor. \quad (7)$$

The quantity J_{ki}^s in (7) is a conversion factor to map the number of DUs, transmitted by the MD, in the equivalent number of clock cycles requested by the ES for their processing.

To set-up a *latency constraint* for each MD, we define an overall queue, which takes into account the overall (MD plus ES) computational load. Since the ES can perform parallel processing of DUs hosted in different queues, a (worst) latency constraint for the k -th MD should consider the longest ES queue, as highlighted by

$$Q_k^{tot}(t) = Q_k^{MD}(t) + \max_i\{Q_{ki}^{ES}(t)\}. \quad (8)$$

However, to respect an average latency constraint, as we do in the following, it makes more sense to consider the average length of the parallel queues, which reads as:

$$Q_k^{tot}(t) = Q_k^{MD}(t) + \sum_{i=1}^{L_k} p_i Q_{ki}^{ES}(t), \quad (9)$$

where p_i is the probability to employ the i -th compression factor, which can be estimated by an online sample mean. By the Little's law [16], imposing an upper-bound Q_k^{avg} on the long-term average of (9), is equivalent to impose a constraint on the average delay $D_k^{avg} = Q_k^{avg} / \bar{A}_k$, where $\bar{A}_k = \mathbb{E}\left\{\frac{A_k(t)}{\tau}\right\}$ is the average data arrival rate at MD k .

D. Energy model

Assuming a capacity-achieving wireless system, the transmission energy $E_k^{tx}(t)$ associated to the k -th device can be inferred inverting the Shannon capacity formula [17], i.e.,

$$E_k^{tx}(t) = \tau \frac{B_k N_0}{h_k(t)^2} \left(e^{\left(\frac{R_k(t) \ln(2)}{B_k}\right)} - 1 \right). \quad (10)$$

For the computations, we employ the same energy model for both the ES and the MD, expressed by [18]

$$E_k^{comp}(t) = \tau \kappa f_k^d(t)^3, \quad E_s(t) = \tau \kappa f_c(t)^3, \quad (11)$$

where the constant κ represents the effective switched capacitance of the processor [18]. For simplicity we assume the same κ for all the devices and the server, although this is not strictly necessary. Finally, the overall energy spent by the system during the t -th time-slot is expressed as:

$$E_k^{tot}(t) = \sum_{k=1}^K [E_k^{comp}(t) + E_k^{tx}(t)] + E_s(t). \quad (12)$$

E. Accuracy model

The resource optimization strategy must be designed to satisfy inference accuracy constraints. To this aim, we resort to a model-based approach, where the accuracy can be cast in the optimization problem as a (discrete) function of the compression factor, by employing a look up table, say, $G(\rho_k(t))$, which stores in each entry the accuracy associated with each specific compression factor (cf., Fig.2).

III. DYNAMIC RESOURCE ALLOCATION VIA LYAPUNOV STOCHASTIC OPTIMIZATION

Exploiting the system model described in Section II, we propose herein a Multi-User Minimum Energy Resource Allocation Strategy (mu-MERAS). To this aim, we define the following long-term optimization problem

$$\begin{aligned} \min_{\Phi(t)} \quad & \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \sum_{k=1}^K \mathbb{E}\{E_k^{tot}(t)\} \\ \text{s.t.} \quad & (a) \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E}\{Q_k^{tot}(t)\} \leq Q_k^{avg} \\ & (b) \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E}\{G(\rho_k(t))\} \geq G_k^{avg} \\ & (c) 0 \leq R_k(t) \leq R_k^{max} \\ & (d) f_c(t) \in \mathcal{F}_s, \quad f_k^d(t) \in \mathcal{F}_k^d, \quad \rho_k(t) \in \mathcal{S}_k \\ & (e) \sum_{k=1}^K \sum_{i=1}^{L_k} f_{ki}^c(t) \leq f_c(t), \quad f_{ki}^c(t) \geq 0, \end{aligned} \quad (13)$$

where the long term system energy consumption is minimized under long-term latency and accuracy constraints (a)-(b). The other constraints (c)-(f) define the feasible set for the variables to be dynamically allocated, which are collected in the vector $\Phi(t) = [\{R_k(t)\}_{k=1}^K, \{\rho_k(t)\}_{k=1}^K, \{\{f_{ki}^c(t)\}_{i=1}^{L_k}\}_{k=1}^K, f_c(t)]$. Problem (13) is complicated to solve, since the expectation is taken with respect to wireless channels and data arrivals, whose statistics are supposed to be unknown a priori. Thus, in the sequel, we will exploit Lyapunov stochastic optimization [15], which enables to transform the long-term optimization in (13) into a pure stability problem, which can be solved in a per-slot fashion. Then, to deal with the long-term constraints (a)-(b), we define for each MD the *virtual queues* $Z_k(t)$ and $Y_k(t)$, which evolve as:

$$\begin{aligned} Z_k(t+1) &= \max(0, Z_k(t) + \mu_k(Q_k(t+1) - Q_k^{avg})) \\ Y_k(t+1) &= \max(0, Y_k(t) + \nu_k(G_k^{avg} - G(\rho_k(t)))) \end{aligned} \quad (14)$$

where the parameters μ_k and ν_k are positive step-sizes, used to control the convergence speed of the algorithm. Now, introducing the *Lyapunov Function*

$$L(t) = \frac{1}{2} \sum_{k=1}^K [Y_k(t)^2 + Z_k(t)^2], \quad (15)$$

letting $\Theta(t) = [\{Z_k(t)\}_{k=1}^K, \{Y_k(t)\}_{k=1}^K]$, and defining $\Delta(t) = \mathbb{E}\{L(t+1) - L(t)|\Theta(t)\}$, we obtain the following Lyapunov Drift plus penalty function

$$\Delta_p(t) = \Delta(t) + V\mathbb{E}\{E_{tot}(t)|\Theta(t)\}. \quad (16)$$

The minimization of (16) implies the optimization of the objective function in (13) (e.g., energy), while respecting the long-term constraints, whose violation is modelled by the congestion of the virtual and physical queues [15]. The parameter V in (16) allows to explore the trade-off between the objective function minimization and the margin on the constraints in (13), giving more or less importance to the term we want to optimize. Interestingly, the minimization of (16) can be decoupled over the ES and MD optimization variables (details are omitted due to lack of space). The results of the two separate optimization steps are given in the following.

A. MDs resource allocation

Since the MDs do not cooperate (or interfere), we can independently optimize the Lyapunov drift (16) for each device. Thus, omitting the temporal index t for notation simplicity, for a fixed compression factor ρ_{ki} , the optimal rate can be computed in closed-form by

$$R_k^*(\rho_{ki}, f_k^d) = \left[\frac{B_k}{\ln(2)} \ln \left(\frac{Q_{ki}^{TX} h_k^2}{W(\rho_{ki}) V \ln(2) N_0} \right) \right]_0^{R_k^{max}} \times \mathbb{1}(Q_{ki}^{TX}(t) > 0) \quad (17)$$

where $W(\rho_{ki}) = M(\rho_{ki})N(\rho_{ki})$ and

$$Q_{ki}^{TX} = (L_k + 1)\mu_k^2(Q_k^{MD} - Q_{ki}^{ES}) + \mu_k Z_k.$$

Since the compression factors ρ_k and the frequencies f_k^d assumes values on discrete sets with low cardinality, the overall problem can be solved by an exhaustive search, computing (17) for any pair (f_{ki}^d, ρ_{ki}) , and then selecting the triple $T_k^* = (R_{ki}^*, f_{ki}^d, \rho_{ki})$ that minimizes the Lyapunov drift (16).

B. ES resource allocation

At the ES, we get the following optimization problem:

$$\begin{aligned} \min_{\Phi_s} \quad & - \sum_{k=1}^K \sum_{i=1}^{L_k} \tau Q_{ki}^S J_{ki}^s f_{ki}^c + \tau V \kappa(f_c)^3 \\ \text{s.t.} \quad & 0 \leq f_{ki}^c \leq \min \left(f_c, \frac{Q_{ki}^{ES}}{\tau J_{ki}^s} \right), f_c \in \mathcal{F}_s \\ & \sum_{k=1}^K \sum_{i=1}^L f_{ki}^c \leq f_c, \end{aligned} \quad (18)$$

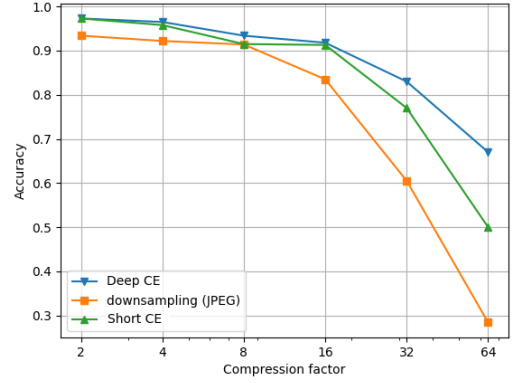


Fig. 2. Accuracy vs Compression Factor without Communication section

TABLE I
CHANNEL PARAMETERS

$D_{max}(m)$	$B(kHz)$	$f_0(GHz)$	σ_0^2
500	2500	9	2.72×10^{-14}

where $\Phi_s = [\{f_{ki}^c\}_{i=1}^{L_k}\}_{k=1}^K, f_c]$ and $Q_{ki}^S = (L_k + 1)\mu_k^2 Q_{ki}^{ES} + \mu_k Z_k$, and whose objective is to select the best server computation frequency f_c , and optimally partition it among the different queues. For any fixed server frequency f_c , (18) becomes the classical (fractional) knapsack problem, which can be easily solved by a linear algorithm [19] and whose solution gives the optimal division of f_c . Thus, solving the problem for each possible clock frequency f_c , we select the best ES resource allocation, by choosing f_c^* (and its associated partitioning) that minimizes the objective function in (16).

IV. SIMULATION RESULTS

We tested the proposed approach by simulations on the GTSRB dataset [20], considering images of 256×256 pixels. All the results are obtained using a bank of pre-trained CE-CNN classifiers, one for each compression factor, picking the λ s in (1), with the best classification accuracy. To assess the effectiveness of our goal-oriented communication scheme, we compared the performance obtained employing short- and deep-CE before the classification CNN, with those obtained by resizing the pseudo-images by a trivial down-sampling process with anti-aliasing. As expected, Fig.2 shows that goal-oriented CE compression schemes reach a greater accuracy, motivating their employment in our framework. However, we underline that the results in Fig.2 do not consider the wireless communication aspects, with the associated energy/latency issues. Thus, it is not possible to exclude that in some cases it could make sense to use also pure down-sampling, thus granting further flexibility to the proposed resource management.

We considered a fast fading channel, characterized by the Jakes-Clark autocorrelation function [21], with an average path-loss computed according to the *ABG model* [22]. Although this is not necessary, we used the same channel (statistical) conditions for all the MDs. In all the simulations we have $K = 5$ MD, each of them equipped with a clock in $\mathcal{F}_d = \{0.1, 0.2, \dots, 0.9, 1\} \times 1.4$ GHz, while the ES clock can choose in $\mathcal{F}_s = \{0.1, 0.2, \dots, 0.9, 1\} \times 4.5$ GHz. Both

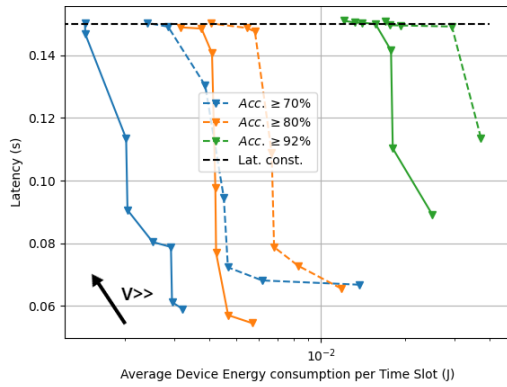


Fig. 3. MD average Energy latency trade-off. CE (solid) vs down-sampling.

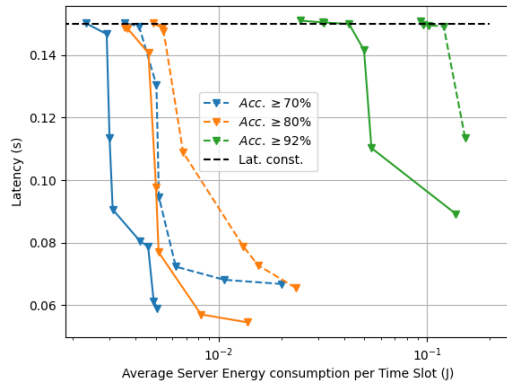


Fig. 4. ES Energy latency trade-off.

MDs and ES energy consumption have been modelled by a switched capacitance $\kappa = 1.097 \times 10^{-27} \left(\frac{s}{cycles}\right)^3$.

We tested the mu-MERAS optimization strategy in the channel scenario reported in Tab.I. In this situation the MDs experience a large channel attenuation, thus resulting in a considerable transmission energy/latency. Fig.3 witnesses how the employment of the Convolutional Encoder compression leads to a lower energy consumption from the MDs perspective. Indeed, from Fig.2, we note that the CEs get acceptable accuracy values also with the highest compression factor, thus allowing to satisfy the accuracy and latency constraints by transmitting small Data Units, i.e., choosing $\rho \in \{32, 64\}$. On the other hand, the employment of a downsampling-based compression technique, forces the resource allocation strategy to choose lower compression factors, with a consequent higher transmission cost.

Looking at Fig.4, we note that the choice of higher compression factors, as it is possible when employing the CEs, it is also advantageous from the ES perspective. Indeed, smaller DUs require lower computational resources to get classified, thus resulting in a lower energy consumption also from the ES perspective.

V. CONCLUSION AND FUTURE DIRECTIONS

We proposed a classification-oriented communication architecture, based on CEs and CNNs, together with a resource

management policy, that proved to be effective and flexible, to trade energy for latency and classification accuracy, in a multi-user scenario served by the same edge server. Future research directions may include the use of multiple servers, cooperative tasks such as in federated learning architectures, as well as further refining of the proposed approach, which may include partial-offloading of the computational task as well as user-focused or server focused energy management.

REFERENCES

- [1] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6g: A comprehensive survey," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 334–366, 2021.
- [2] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, 2019.
- [3] S. Wang, T. Tuor *et al.*, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. of IEEE INFOCOM 2018*, 2018, pp. 63–71.
- [4] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," *IEEE communications magazine*, vol. 58, no. 1, pp. 19–25, 2020.
- [5] M. Merluzzi, P. Di Lorenzo, and S. Barbarossa, "Wireless edge machine learning: Resource allocation and trade-offs," *IEEE Access*, vol. 9, pp. 45 377–45 398, 2021.
- [6] E. C. Strinati and S. Barbarossa, "6g networks: Beyond shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107930, 2021.
- [7] F. Pezone, S. Barbarossa, and P. Di Lorenzo, "Goal-oriented communication for edge learning based on the information bottleneck," *accepted to ICASSP-2022*. [Online]. Available: <https://arxiv.org/pdf/2202.12639.pdf>
- [8] J. Shao, Y. Mao, and J. Zhang, "Learning task-oriented communication for edge inference: An information bottleneck approach," *arXiv preprint arXiv:2102.04170*, 2021.
- [9] M. Jankowski, D. Gündüz, and K. Mikołajczyk, "Wireless image retrieval at the edge," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 89–100, 2020.
- [10] Z. Weng, Z. Qin, and G. Y. Li, "Semantic communications for speech recognition," *arXiv preprint arXiv:2107.11190*, 2021.
- [11] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.
- [12] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.
- [13] M. Boudiaf *et al.*, "A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses," in *European Conference on Computer Vision*. Springer, 2020, pp. 548–564.
- [14] L. Le, A. Patterson, and M. White, "Supervised autoencoders: Improving generalization performance with unsupervised regularizers," in *Advances in Neural Inf. Proc. Systems*, vol. 31. Curran Associates, Inc., 2018.
- [15] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool Publ., 2010.
- [16] J. D. Little, "A proof for the queuing formula: $L = \lambda w$," *Operations research*, vol. 9, no. 3, pp. 383–387, 1961.
- [17] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [18] T. Burd and R. Brodersen, "Processor design for portable systems," *Journal of VLSI Signal Processing*, vol. 13, 11 1996.
- [19] M. Assi and R. A. Haraty, "A survey of the knapsack problem," in *2018 Int. Arab Conf.on Information Tech.(ACIT)*, 2018, pp. 1–6.
- [20] J. Stallkamp *et al.*, "The german traffic sign recognition benchmark: a multi-class classification competition," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 1453–1460.
- [21] A. F. Molisch, *Wireless Communications*. Wiley-IEEE Press, 2011, ch. Wireless channel.
- [22] S. Sun *et al.*, "Propagation path loss models for 5g urban micro-and macro-cellular scenarios," in *2016 IEEE 83rd Vehicular Techn. Conf. (VTC Spring)*, 2016, pp. 1–6.