IEEE International Performance Computing and Communications Conference-IPCCC-2017, December 2017, San Diego, CA.

Unsupervised Machine Learning in 5G Networks for Low Latency Communications

Eren Balevi, *Member, IEEE* and Richard D. Gitlin, *Life Fellow, IEEE*
Department of Electrical Engineering
University of South Florida
erenbalevi@mail.usf.edu, richgitlin@usf.edu

***Abstract*—This paper incorporates fog networking into heterogeneous cellular networks that are composed of a high power node (HPN) and many low power nodes (LPNs). The locations of the fog nodes that are upgraded from LPNs are specified by modifying the unsupervised soft-clustering machine learning algorithm with the ultimate aim of reducing latency. The clusters are constructed accordingly so that the leader of each cluster becomes a fog node. The proposed approach significantly reduces the latency with respect to the simple, but practical, Voronoi tessellation model, however the improvement is bounded and saturates. Hence, closed-loop error control systems will be challenged in meeting the demanding latency requirement of 5G systems, so that open-loop communication may be required to meet the 1ms latency requirement of 5G networks.**

***Index Terms*—Machine learning, unsupervised clustering, fog networking.**

I. Introduction

Stringent latency requirements of 5G applications have driven a paradigm shift in the state-of-art 4G networks based on the idea of making cloud closer to the end devices known as fog networking [1], [2]. In fog networking, some nodes, e.g., access points, small cells, routers, mobiles, are specialized as fog nodes in a cloud-to-things continuum [2] to control and provide services to the end devices. A promising idea is to incorporate fog networking into heterogeneous networks (HetNets) that are composed of a high power node (HPN) and many low power nodes (LPNs), where some LPNs are upgraded as fog nodes. This can considerably enhance the performance of the state-of-art HetNets not only mitigating interference among LPNs through the control capability of fog nodes but also provide service to the end devices relying on the storage capability of fog nodes.

Despite the apparent benefits of fog networking in HetNets, this network architecture comes with its own questions such as which LPN nodes should become fog nodes, and what should be the number of fog nodes? In this paper, the former question is discussed, i.e., the locations of LPNs that are upgraded to fog nodes are found assuming that the number of fog nodes are given as *a priori* information and the locations of all LPNs within a cell are known. More precisely, LPNs that are upgraded to fog nodes are determined based on an unsupervised soft clustering machine learning algorithm [3]. Accordingly, LPNs are clustered so that the leaders of each cluster, i.e., cluster-heads, are upgraded to fog nodes. There has to be a metric to govern clustering and the metric here is to reduce latency.

There are many different soft clustering approaches for different machine learning applications, however, those papers are not focused on reducing latency in wireless networks, e.g., see [3] and references there. Furthermore, fog networking is now in an early stage and there is no prior art that specifies the locations of fog nodes in the network [2]. This paper fills this gap in this quite inspiring and appealing domain of fog networking.

II. Network Model and Problem Formulation

Fog networks consists of a data plane and a control plane. In the data plane, fog computing and the associated computing services strive to achieve client objectives via its unique features such as dense geographical distribution, local resource pooling, latency reduction and backbone bandwidth savings to achieve better quality of service (QoS). In the control plane, fog networking can coordinate many devices to mitigate interference, which, in the scope of this paper, are the LPNs. A representative heterogeneous networking arrangement that shows a cloud server, fog nodes, LPNs and a HPN is shown in Fig. 1.

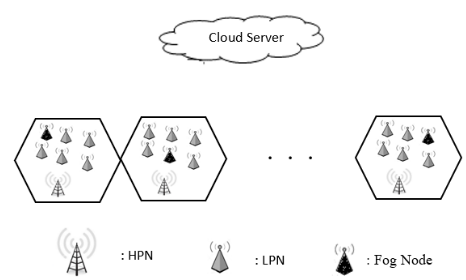

Fig. 1. An illustration of fog networking in a heterogeneous network.

This network model inherently raises the question of the locations of fog nodes and their service area. Regarding the service area of fog nodes, the simplest approach is to employ the Voronoi tessellation model so that each LPN selects a fog node at the closest Euclidean distance. Indeed, this approach corresponds to the K-means hard clustering algorithm in machine learning [3]. The main problem with this model is that the closest Euclidean distance channel may be of poor quality, which degrades the performance of the communication, and thus increases the latency. Based on this motivation, a soft clustering algorithm that reduces the latency is discussed in the next section so that any LPN can be a fog node according to the quality of the channel and one LPN can be probabilistically connected to many fog nodes. Notice that it is well-known that a soft-clustering algorithm performs better than hard clustering [3].

III. A Novel Fuzzy Clustering Algorithm

Assume that the number of fog nodes and the LPN are known *a priori*, and there is a data set composed of the geographical locations of the LPNs such as $X = \{x_1, x_2, \cdots, x_N\} \in R^2$ where N is the total number of LPNs and K of them will be upgraded to fog nodes. Hence, this data set is clustered with the following objective function so that the leader of the clusters or cluster-heads give the locations of the fog nodes as $F = \{f_1, f_2, \cdots, f_K\} \in R^2$.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} F(\gamma_{nk} f(x_n, f_k)) \qquad (1)$$

The optimization of (1) forms the clusters where γ_{nk} shows the probability of connection between one LPN and a fog node, i.e., $\gamma_{nk} \in [0, 1]$. Note that if this was a hard clustering K-means algorithm, γ_{nk} would be either 0 or 1. In particular, $f(x_n, f_k)$ measures the similarity of any data point x_n for $n = 1, 2, \cdots, N$ with a fog node f_k for $k = 1, 2, \cdots, K$, and multiplying it with γ_{nk} constitutes the objective function.

The primary aim of maximizing (1) is to determine a clustering that reduces the latency within the network. To do so, the LPNs are clustered according to their channel strength, which means that each LPN associates with fog nodes that have channels above a certain quality. It is important to emphasize that each LPN can be associated from multiple fog nodes depending on the soft clustering algorithm whose details are presented below.

Algorithm-1: Low latency clustering algorithm

1) Set the number of fog nodes that will be upgraded from LPNs and the number of LPNs that are given as *a priori* information.
2) Specify the quality of channels among all LPNs.
3) Find the fog nodes according to the channel quality.
4) Determine the probability of connection between fog nodes and LPNs according to the channel quality.

A simulation is performed to evaluate the efficiency of the proposed algorithm. Within this scope, it is assumed that there are 8 fog nodes and 100 LPNs that are not specialized as fog nodes within a single cell. According to the proposed algorithm, each LPN is connected to all fog nodes probabilistically depending on the channel conditions. This scheme is compared with the simplest but the practical one in which each LPN is connected to only one fog node that has closest Euclidean distance with itself known as Voronoi tessellation model. When SNR is fixed at 5dB without any loss of generality, the comparison of the proposed clustering and the Voronoi tessellation model is given in Fig. 2 regarding latency in terms of bandwidth. As can be seen, the proposed clustering has a significant latency advantage for low bandwidths. It is worth emphasizing that the proposed algorithm achieves 1 ms latency requirement of 5G applications at 1 GHz bandwidth at 5dB. Notice that increasing SNR improves latency.

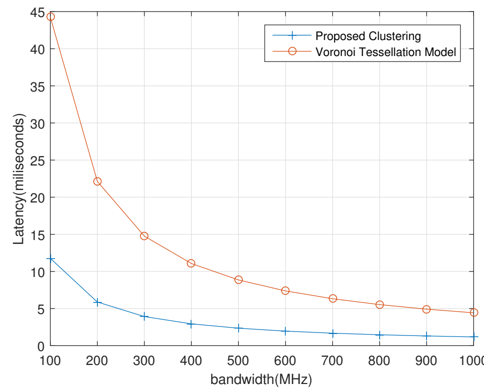

Fig. 2. Latency of the proposed algorithm compared to the Voronoi model as a function of bandwidth for $K = 8$.

One further important point is that the latency calculations are performed with open-loop communications, which means that there is no ACK mechanism for transmitted packets. It is clear that all latency values will be at least doubled when an ACK is required since unsuccessful transmission requires a retransmission and an additional ACK that increases the latency more than two-fold. Based on these discussions, it is certainly challenging to decrease the latency to 1 ms. Our results show that the latency decreases with bandwidth up to a point and after this point the latency saturates and does not decrease further. As a result, it can be deduced that open loop communication is likely necessary to meet the challenging 1 ms requirement in addition to intelligent clustering algorithms. So, to achieve 1 ms latency, it may be necessary to replace the widely used automatic repeat request (ARQ) or hybrid automatic repeat request (HARQ) with open loop communication at the expense of reliability. One promising approach that can be employed to address the network reliability problem is the recently proposed Diversity Coding-Network Coding (DC-NC) which is based on the synergistic combination of diversity coding and network coding [4]. A thorough latency analysis combining the proposed clustering and DC-NC to minimize the latency in fog networking will be considered in future work.

References

[1] F. Bonomi et al., "Fog Computing and its Role in the Internet of Things", *Proc. Wksp. Mobile Cloud Computing*, Helsinki, Finland, pp. 13-16, Aug. 2012.

[2] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities", *IEEE Internet Things J.*, vol. PP, no. 99, pp. 1-1, 2016.

[3] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, Aug. 2006.

[4] N. I. Sulieman, E. Balevi, K. Davaslioglu, and R. D. Gitlin, "Diversity and Network coded 5G fronthaul wireless networks for ultra-low latency communication", PIMRC 2017.