Leveraging Social Behaviour of Users Mobility and Interests for Improving QoS and Energy Efficiency of Content Services in Mobile Community Edge-clouds

Vu San Ha Huynh[a] and Milena Radenkovic[b]
School of Computer Science, University of Nottingham, Nottingham, U.K.

Keywords: Mobile Community Edge-clouds, Network Coding, Content Caching, Multilayer Spatial-temporal Locality.

Abstract: Community network edge-clouds have been attracting significant interests over the recent years with the emergence of ubiquitous networked devices embedded in our daily activities and increasingly widespread fully-distributed heterogeneous networks of smart edges offering various applications and services in real time. This paper proposes EdgeCNC, a novel joint multilayer adaptive opportunistic network-coding algorithm integrated with adaptive opportunistic content caching service. EdgeCNC exploits the multilayer spatial-temporal locality of users' mobility and interests in community network edge-clouds in order to select highly suitable set of contents to forward, cache and network code to highly suitable set of nodes in order to enhance QoS, reduce data transmissions and improve energy efficiency. We perform a multi-criteria evaluation of EdgeCNC performance in realistic Foursquare New York scenario of mobile community edge-clouds against the benchmark and competitive protocols in the face of dynamically changing users' publish-subscribe and mobility patterns. We show that EdgeCNC achieves higher success ratio and data transmission efficiency while keeping lower delays, packet loss and energy consumption compared to the competitive and benchmark protocols.

1 INTRODUCTION

Community network edge-clouds (Selimi et al., 2019) aim to respond to the rapidly increasing demands for network connectivity and real-time distributed services in rural and urban communities. According to (Cisco Index, 2017), global network traffic will be threefold over the next few years and the provision of intelligent and adaptive content services (Radenkovic et al., 2018) closer to the local interest of dynamic geo-temporal clusters of mobile users will be needed to deal with the increasing traffic demand. This paper aims to further improve the reliability and scalability of the intelligent dynamic edge-cloud networks by reducing data transmission and improving energy efficiency which play a vital role regarding the resource constraints and battery of users' mobile devices.

We propose EdgeCNC, a novel integration scheme which combines adaptive and opportunistic network coding service CafNC (Radenkovic and Zakhary, 2012) and adaptive edge caching CafRepCache (Radenkovic et al., 2018) to improve the QoS and energy efficiency of content services in mobile community edge-clouds. We argue that adaptive network coding (Radenkovic and Zakhary, 2012) and adaptive content caching (Radenkovic et al., 2018) will enable high-performance efficiency of content services while reducing the number of sending packets, avoiding congestion and minimising energy consumption in dynamic mobile community edge-clouds. We envisage that EdgeCNC is desirable in heterogeneous dynamic geo-temporal clusters of mobile user scenarios such as mobile personal clouds in pervasive health services (Radenkovic and Huynh, 2016) and cognitive privacy for personal mobile clouds (Radenkovic, 2016). Building on CafNC and CafRepCache, EdgeCNC leverages the social power-law behaviour of users' mobility and interest to select highly suitable set of contents to forward, cache and network code in highly suitable set of relaying nodes while being adaptive to dynamically changing resource and energy availability. We evaluate and

[a] https://orcid.org/0000-0002-5472-5328
[b] https://orcid.org/0000-0003-4000-6143

Huynh, V. and Radenkovic, M.
Leveraging Social Behaviour of Users Mobility and Interests for Improving QoS and Energy Efficiency of Content Services in Mobile Community Edge-clouds.
DOI: 10.5220/0009415103820389
In *Proceedings of the 10th International Conference on Cloud Computing and Services Science (CLOSER 2020)*, pages 382-389
ISBN: 978-989-758-424-4

compare EdgeCNC with state-of-the-art in realistic Foursquare New York scenario of mobile community edge-clouds. We show that EdgeCNC enhances QoS of content services such as edge-to-edge delay, packet loss as well as improves the energy efficiency of communications between edge-clouds.

The paper begins by providing an overview of the related work in Section 2. We then introduce the adaptive multilayer opportunistic network coding mechanism integrated with adaptive caching in Section 3. Section 4 evaluates the performance of different network-coded forwarding and caching protocols in opportunistic networks. The conclusion is given in Section 5.

2 RELATED WORK

Network coding approaches (Kouvelas et al., 1998; Radenkovic, 2004; Radenkovic and Zakhary, 2012) for dynamic communities (i.e. clusters) have attracted significant interest in recent years due to their potential to improve network throughput, reduce delay and increase data transmission efficiency. HubCode (Ahmed and Kanhere, 2009) is proposed as a static forwarding and network coding approach that utilises highly central nodes as message relays in order to improve the probability of message delivery. While using highly central nodes or hubs as relays may help to deliver messages to the destinations, CafNC (Radenkovic and Zakhary, 2012) shows that these hubs may become overloaded when there is an increasing congestion level, resulting in significant packet loss and achieve even lower delivery success ratios. In this paper, we integrate CafNC's solution and further refine it with energy awareness to choose the relaying nodes that are capable of receiving coded contents without becoming congested and being energy depleted. (Muhammad and Kang, 2018) states that network coding is beneficial in content-centric networks which fundamentally consider multiple publish-subscribe content-delivery solutions, then propose a linear-coding techniques CCCN (Muhammad and Kang, 2018). However, CCCN does not support capturing and predicting the spatial-temporal locality of content requests, thus the network coding rate calculation is not sufficiently fine-grained.

Authors in (Manzoor et al., 2019) propose a proactive caching which measures mobility prediction uncertainty and content request frequency to predict the most promising prefetching node, thus eliminating redundancy and improving cache resource efficiency. The proposed approach achieves improvement compared to reactive schemes; however, these gains are limited due to the dynamics of users' mobility patterns and content traffic demand. Authors in (Said et al., 2018) propose a proactive caching algorithm that caches the contents from highly influential users within a close group or community which is discovered by clustering coefficient based genetic algorithm. (Said et al., 2018) assumes the assist of infrastructure-based knowledge and does not support the dynamic network topologies due to the users' mobility.

In order to identify and predict more accurately dynamic spatial-temporal clusters of mobile social publisher-subscriber, thus improve caching and network coding performance, exploitation of users' contextual information for social power-law behaviour of users' mobility and content interests is needed. Research on today's social networks (Dabirmoghaddam et al., 2014; Oliveira et al., 2017) has questioned the validity of the Independent Reference Model (IRM) and Zipf's law model (Cha et al., 2007) of content requests which suggested that the content is requested scattered randomly and independently over time. Instead, (Dabirmoghaddam et al., 2014; Oliveira et al., 2017) argue that content requests often exhibit both temporal and spatial locality which is shown in the real-world Foursquare New York dataset that we used in this paper.

3 JOINT NETWORK CODING & CACHING SERVICES IN COMMUNITY EDGE-CLOUDS

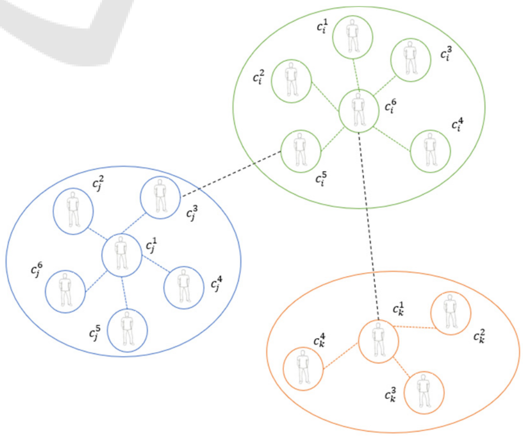

Figure 1: Community Edge-Clouds.

Community edge-clouds (Fig. 1) are fully-distributed self-organised networks of smart edges which form heterogeneous dynamic geo-temporal clusters of users. In this paper, we propose EdgeCNC, a novel joint multilayer network coding-caching mechanism which integrates adaptive opportunistic linear network coding in CafNC (Radenkovic and Zakhary, 2012) with adaptive edge caching service in CafRepCache (Radenkovic et al., 2018) to improve the QoS and energy efficiency of multilayer communications in mobile community edge-cloud networks.

3.1 Overview of Joint Network Coding and Caching Services - EdgeCNC

Adaptive congestion-aware forwarding, replication and caching CafRepCache (Radenkovic et al., 2018) is a multilayer adaptive caching framework in mobile Opportunistic Networks. CafRepCache is able to capture and predict the dynamic spatial-temporal locality of users' mobility, content request patterns and resources as well as the interdependences between them in order to more accurately and more responsively cache contents from dynamic clusters of subscribers (Huynh and Radenkovic, 2019) with dynamic mobility, complex content requests and dynamic resource availability. CafRepCache utilises multi-layer predictive analytics to manage complex dynamic trade-offs between maximising content delivery while reducing delay and resource consumption. This paper extends CafRepCache with adaptive opportunistic network coding to reduce the number of sending packets, improve network throughput and enhance data transmission efficiency.

Adaptive congestion-aware forwarding and opportunistic network coding CafNC (Radenkovic and Zakhary, 2012) is a real-time multilayer network coding mechanism in mobile Opportunistic Networks. CafNC is able to capture, predict and adapt to dynamically changing patterns of users mobility, content interests and resources via multilayer predictive analytics which drives network coding to the most suitable set of contents in the most suitable set of carriers in order to increase content delivery success ratio while minimizing delay, reducing redundancy and waste of network resources in multi-source, multi-path and multicast forwarding algorithm. CafNC adaptively balances the complex dynamic trade-off between under and over-utilised coding nodes, as missing encoding opportunities may potentially cause increased delays and lower success ratios while extensive coding at the time of high congestion may result in significant packet loss (Radenkovic and Zakhary, 2012). CafNC encodes together contents that share the same interest or are sent to the same dynamic geo-temporal cluster of subscribers. This paper refines CafNC with an energy-aware network coding rate controlled by the network coding threshold metric in order to improve the data transmission and energy efficiency of communications in community edge-clouds.

EdgeCNC is built on CafRepCache (Radenkovic et al., 2018) to allow integration with CafNC (Radenkovic and Zakhary, 2012) and enable energy awareness which enhances the QoS while reducing the number of sending packets and improving the energy efficiency of mobile community edge-cloud networks. In line with CafNC (Radenkovic and Zakhary, 2012) and CafRepCache (Radenkovic et al., 2018), EdgeCNC is adaptive, fully-distributed, opportunistic, collaborative and is able to perform multilayer spatial-temporal predictive analytics and heuristics of multivariate mixed data (e.g. mobility, content and resource) as well as collaborate and exchange its local observations with other neighbour nodes in order to detect and exploit coding and caching opportunities in the accurate and responsive manner without the need of global knowledge.

3.2 Energy-Aware Network Coding in EdgeCNC

We refine CafNC (Radenkovic and Zakhary, 2012) with energy consideration by proposing a novel energy-aware network coding threshold metric in order to improve data transmission and energy efficiency. In line with CafNC (Radenkovic and Zakhary, 2012), we model linear network coding process in EdgeCNC system as a network G consisting of a set of nodes N and a set E of edges. A set of contents that can be requested in the network is denoted as O. A node $n_i \in N$ receives c_k^i (the encoded form of content $o_k \in O$) directly from a relay node and linearly combined encoded symbol $(c_n, \tau_i c_i + \tau_n c_n)$ from another node in which requested content is segmented linearly into the encoded symbols and the encoded symbols are received simultaneously. The coding vector is defined as a set of coefficients associated with each coding point. We define the transform matrices T as follow:

$$T = \begin{bmatrix} 1 & 0 \\ \tau 1 & \tau 2 \end{bmatrix} \quad (1)$$

We consider φ_i, as the symbol received by the caching point or intermediary node n_i. The node can then recover all N encoded symbols by resolving the linear equation:

$$\begin{bmatrix} c_1 \\ c_2 \\ c_n \end{bmatrix} = T^{-1} \begin{bmatrix} \varphi_i \\ \varphi_2 \\ \varphi_n \end{bmatrix} \quad (2)$$

Energy efficiency plays a vital role in EdgeCNC solution due to the constraint in the battery life of users' mobile devices. Thus, we propose to refine the NCThreshold from CafNC (Radenkovic and Zakhary, 2012) to include energy awareness (in addition to mobility, resources and contents). The EnergyNCThreshold parameter to control the coding rate is resolved based on energy availability E_i. EnergyNCThreshold is a configurable parameter between 0 and 1 measured as:

$$\text{EnergyNCThreshold} = \begin{cases} 1, & E_i = 0 \\ \dfrac{\text{cost}_k}{\gamma E_i}, & E_i > 0 \end{cases} \quad (3)$$

in which E_i is the currently available energy level of a node n_i, cost_k is the energy cost to transmit signals, forward and code the content o_k. γ is a control parameter which will be set depending on the importance of a node such that $0 < \gamma \leq 1$. For example, γ will be set to low if a node n_i is important and has to be protected from being battery drainage.

EdgeCNC justifies the number of coding a node carries out based on EnergyNCRate parameter:

$$\text{EnergyNCRate} = \text{EdgeCNCTotalUtil}^{t}_{i,k} * (1 - \text{EnergyNCThreshold}) \quad (4)$$

in which EdgeCNCTotalUtil is the total multi-layer utility of EdgeCNC in line with (Radenkovic and Zakhary, 2012) and (Radenkovic et al., 2018).

Table 1: EdgeCNC pseudocode.

EdgeCNC – Arrival of Content Request/Interest

When the request/interest of a content is received at a node:
for each Contact c **do**:
c.socialHeur = exchSocHeur (Contact, Contacts.reputation)
c.resourceHeur = exchResHeur (Contact, Contacts.reputation)
c.contentHeur = exchPopularity(Contact, Contacts.reputation)
c.calculateUtility(c.heuristics)
ListUtils.insert(Contact.Utility)
end for
(x_0, x_1) = in-networkCaching&NetworkCoding(ListUtils)
if x_0 == 1
Node is set to be a cache candidate of the Content
if(EnergyNCRate >= 0.5 & lenth(contents) ≥ EnergyNCThreshold) **do** NCodeCache(contents) **else** cache(contents)
else if x_1 == 1
forward Interest to Contact with highest utility value
if(EnergyNCRate >= 0.5 & lenth(contents) ≥ EnergyNCThreshold) **do** NCodeForward(content) **else** forward(content)
end if

In line with CafNC (Radenkovic and Zakhary, 2012), we argue that when EnergyNCRate is high, the level of congestion is relatively low, EdgeCNC increases the probability of coding with low risk of packet loss. When EnergyNCRate is low, there may be a high level of congestion, EdgeCNC stricts the coding criteria as the risk of high packet loss. Packets of the content are compared to the EnergyNCThreshold. If the number of packets for the content exceeds the threshold, then the node network codes the contents and sends its coded contents to the corresponding neighbour node. If the threshold is higher than the number of content packets, EdgeCNC forwards the contents without coding them. The caching node generates a coding vector and performs a linear combination of the packets cached that are targeted to the same dynamic cluster of subscribers. The subscribers will collect packets and when the number of these packets reaches a certain number, the subscribers will decode them and deliver them to the upper network layer. We provide the pseudo-code of EdgeCNC in Table 1.

4 EVALUATION

This section discusses the performance of EdgeCNC across multiple metrics against state-of-the-art network coded forwarding and caching protocols including HubCode (Ahmed and Kanhere, 2009), CafNC (Radenkovic and Zakhary, 2012), CCCN (Muhammad and Kang, 2018) and CafRepCache (Radenkovic et al., 2018). The overall performance is measured by different criteria: success ratio, latency, packet loss, number of nodes coding, network coding cost (measured as the ratio between encoding/decoding time and total time) and average energy consumption.

We use real-world mobility trace (Scott et al., 2009) and real-world content requests Foursquare New York trace (Dingqi et al., 2015) in ONE simulator (Keränen et al., 2009). We run our simulation for 7 days with a total of 100 nodes and 10^5 contents in the network. In line with (Cha et al., 2007), we vary the content size from 1MB to 8.4MB while the request packet size ranges from 8kB to 128 kB. In this paper, we fix the number of content publishers (15% of nodes) while varying the number of Foursquare users (subscribers) who have interests in the contents. All experiments are repeated ten times and averaged to give us statistically sufficient diversity to evaluate EdgeCNC performance in different publish-subscribe contexts.

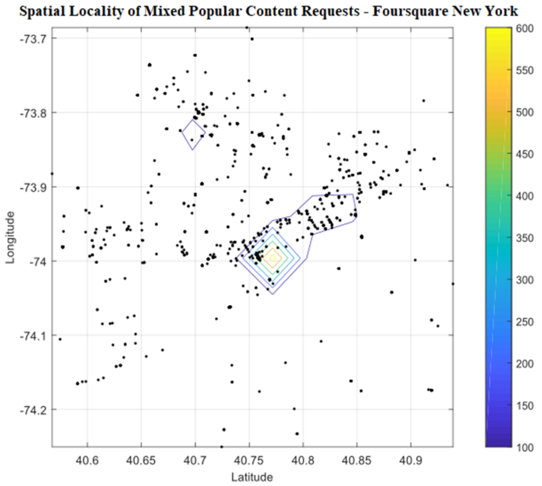

Figure 2a: Spatial locality of mixed popular content requests in Foursquare New York.

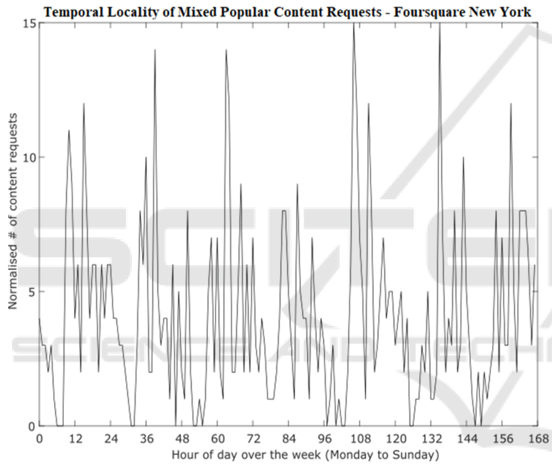

Figure 2b: Temporal locality of mixed popular content requests in Foursquare New York.

Fig. 2a & b show the spatial and temporal correlation of content traffic (i.e. temporal requests pattern of mobile subscribers) for different contents in different locations in Foursquare dataset. The locations of mobile subscribers feature different degrees of similarity in the content request such that two locations which are relatively close to each other have similar request patterns compared to that of another location which is far away from others (Fig. 2a). This shows that there is a strong correlation between the geographical diversity of the users and their requested contents. Fig. 2b shows the temporal locality of content traffic during weekdays and weekend that the content is not requested randomly and independently, rather a content might be of particular interest at a certain period of time interval before its popularity gradually decreases.

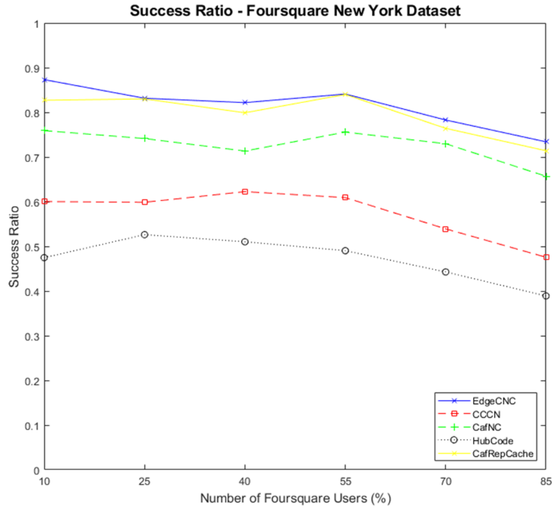

Figure 3: Edge-to-edge success ratio vs. Number of Foursquare users.

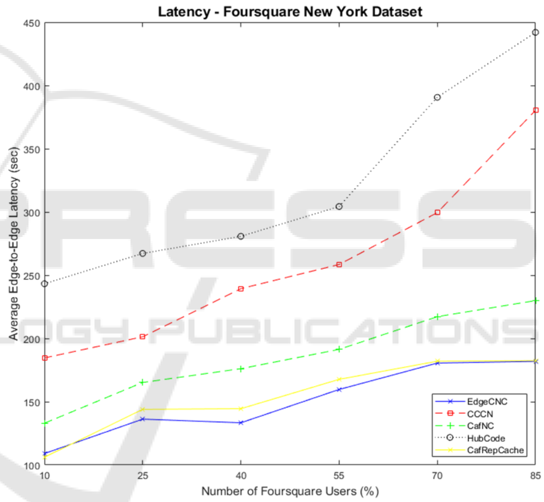

Figure 4: Average edge-to-edge latency vs. Number of Foursquare users.

As EdgeCNC integrates multilayer adaptive opportunistic CafNC and CafRepCache, it is able to utilise multilayer multidimensional predictive analytics of in-network delays and congesting rates of nodes and their ego networks in order to minimise delays of dynamic edge-to-edge content retrieval in community edge-cloud services. EdgeCNC extends CafRepCache with adaptive network coding mechanism to reduce the number of sending packets and thus improve overall network throughput. EdgeCNC refines network coding rate in CafNC with energy awareness (in addition to mobility, resources, contents) which enhances the QoS while reducing the number of sending packets and avoiding energy depletion. Furthermore, EdgeCNC's integrated adaptive content caching mechanism allows the requested contents to be mainly found in the caching points rather than getting the content from the publishers, thus reduce significantly the edge-to-edge latency. HubCode and CCCN have the worst performance (below 60% success ratio, above 6.1 min delay) due to its static and non-adaptive network coding in the face of increasing congestion levels.

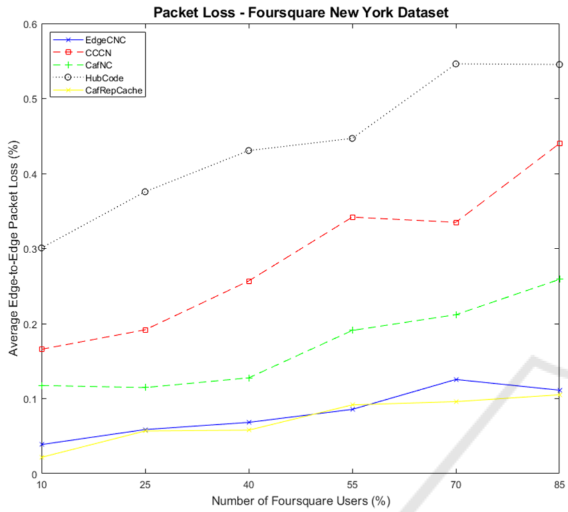

Figure 5: Edge-to-edge packet loss vs. Number of Foursquare users.

Fig. 5 shows the packet loss that EdgeCNC and other competing protocols experience in the face of increasing congestion levels through the increasing number of mobile Foursquare users. EdgeCNC, CafRepCache and CafNC achieve the lowest packet loss rates (below 18%) followed by CCCN (43%) and HubCode (56%). This is because EdgeCNC, CafRepCache and CafNC utilise fully-distributed predictive analytics and heuristics of how likely the nodes and their ego networks are about to congest, size of the buffer, the bandwidth available and computational resources of a node in order to avoid forwarding and network coding to congested nodes or regions where congestion may happen. CafRepCache without network coding has relatively lower packet loss compared to EdgeCNC and CafNC as dropping network-coded packets, even occasionally, may result in increased packet loss. HubCode has lowest QoS performance in terms of packet loss rates (55.4%) as it constantly network codes the contents, even in the event of congestion which increases significantly the packet loss compared to only forwarding a single packet. CCCN has significantly higher packet loss rate compared to EdgeCNC, CafRepCache and CafNC as CCCN does not support congestion-aware and energy-aware network coding.

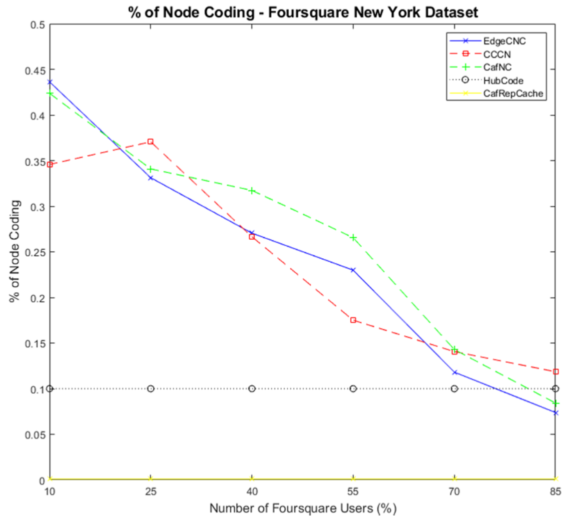

Figure 6: Average % of node coding vs. Number of Foursquare users.

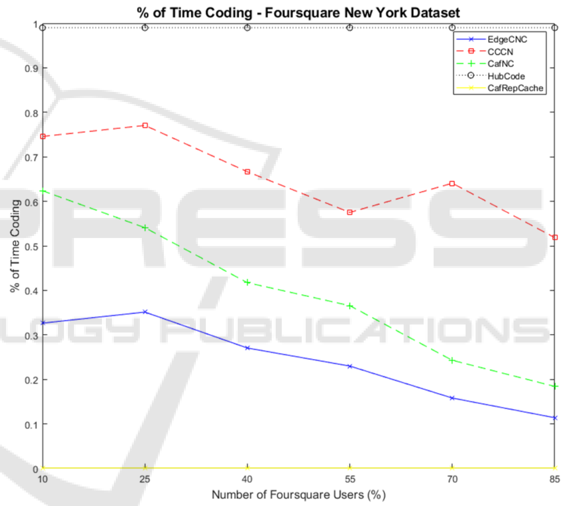

Figure 7: Average % of Time Coding vs. Number of Foursquare users.

Fig. 6 and 7 show the average percentage of network coding nodes and the average percentage of time duration out of total simulation time a node performs network coding in EdgeCNC, CCCN, CafNC and HubCode in the face of the increasing number of mobile Foursquare users. We observe that EdgeCNC and CafNC use 18-21% of the nodes to encode/decode traffic 29-35% of the time. HubCode is 400% worse off than others as it uses 10% of the nodes to network code traffic all of the time (99%). HubCode, as a static forwarding coding approach, misses many coding opportunities even when congestion is low. When the congestion level is increased, HubCode forwards and codes at a static rate that increases the risk of packet loss.

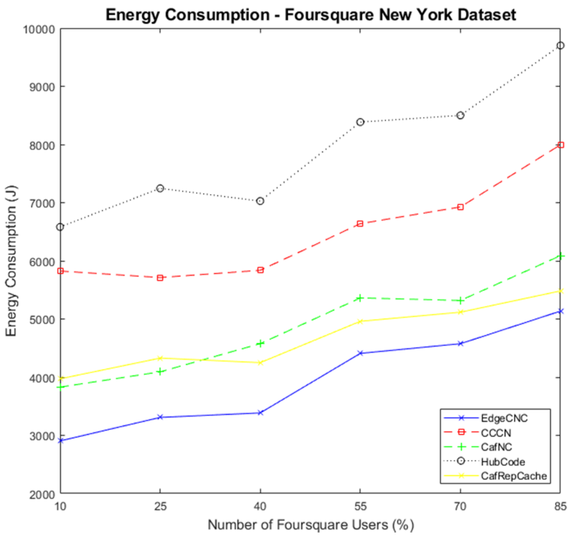

Figure 8: Average Energy Consumption (J) vs. Number of Foursquare users.

Fig. 8 shows that EdgeCNC consumes significantly less average energy (5095J) compared to HubCode (9824J) and CCCN (8012J) due to EdgeCNC utilising multilayer predictive analytics and heuristics for dynamic resources and energy (in addition to mobility and contents) which allows it to improve data transmission efficiency, reduce waste of network resources and energy consumption (while the HubCode and CCCN do not support congestion and energy awareness). EdgeCNC's energy consumption is less than 10-20% compared to CafRepCache (5412J) and CafNC (6039J). This is because EdgeCNC extends CafRepCache and CafNC with adaptive network coding and caching which reduce significantly the number of sending packets, thus improve energy efficiency. EdgeCNC profits from its refined energy-aware network coding rate in order to cache and network code contents of dynamic spatial-temporal clusters of subscribers in a more energy-efficient manner, thus minimise redundancy and decrease average energy consumption.

5 CONCLUSIONS

This paper proposes EdgeCNC, a novel adaptive opportunistic joint network coding and caching mechanism in mobile community edge-clouds. EdgeCNC integrates multilayer adaptive network coding scheme CafNC (Radenkovic and Zakhary, 2012) and adaptive content caching service CafRepCache (Radenkovic et al., 2018) in order to improve QoS and energy efficiency for community edge-clouds. EdgeCNC is able to exploit the multilayer spatial-temporal locality of users' mobility, interests and resources in order to select highly suitable set of contents to forward, cache and network code in highly suitable set of nodes. Note that our focus is to design an algorithm across trusted collaborators in the community network edge-clouds, thus malicious-behaviour protection is out of the scope of this paper. We performed an extensive evaluation of EdgeCNC in real-world connectivity traces and use a realistic dynamic changing content request dataset in order to compare EdgeCNC performance versus competitive network coded forwarding and caching algorithms: HubCode (Ahmed and Kanhere, 2009), CCCN (Muhammad and Kang, 2018), CafNC (Radenkovic and Zakhary, 2012) and CafRepCache (Radenkovic et al., 2018). We show that our proposal performs better than the state-of-the-art solutions, it improves the content delivery success ratio while reducing the number of data transmissions, resulting in lower packet loss, delay and energy consumption.

We envisage that the jointly adaptive network coding-caching mechanism proposed in this paper will be applied to many application scenarios of mobile community edge-clouds that include both social and other complex types of networks.

REFERENCES

Ahmed S. and Kanhere S. *Hubcode: message forwarding using hub-based network coding in delay tolerant networks*. Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems ACM, p. 288–296, 2009.

Cha M., Kwak H., Rodriguez P., Ahn Y.-Y., and Moon S. *I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system*. Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, ser. IMC '07, p. 1–14. 2007

Dabirmoghaddam A., Barijough M. M., and Garcia-LunaAceves J. *Understanding optimal caching and opportunistic caching at 'the edge' of information-centric networks*. Proc. ICN. USA, pp. 47–56. 2014.

Dingqi Y., Daqing Z., Vincent W. Z., Zhiyong Y. *Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs*. IEEE Trans. on Systems, Man, and Cybernetics: Systems, (TSMC), 45(1), p. 129-142, 2015.

Huynh, V. S. H. and Radenkovic, M. *Interdependent Multi-layer Spatial Temporal-based Caching in Heterogeneous Mobile Edge and Fog Networks*. Proceedings of the 9th International Conference on Pervasive and Embedded Computing and Communication Systems PECCS. P. 34 – 45. 2019.

Index, C. V. N. *Global mobile data traffic forecast update, 2016–2021 white paper*. Cisco: San Jose, CA, USA. 2017.

Karbaschi G., Viana A., Martin S., and Al Agha K. *On using network coding in multi hop wireless networks*. Personal, Indoor and Mobile Radio Communications, IEEE 20th International Symposium. p. 420–424. 2009.

Keränen A., Ott J. and Kärkkäinen T. *The ONE Simulator for DTN Protocol Evaluation*. SIMUTools'09: 2nd International Conference on Simulation Tools and Techniques. Rome. 2009.

Kouvelas, I., Hardman, V. and Crowcroft, J. *Network adaptive continuous-media applications through self organised transcoding*. Proceedings of the Network and Operating Systems Support for Digital Audio and Video. 1998.

Manzoor, S., Mazhar, S., Asghar, A., Mian, A. N., Imran, A. and Crowcroft, J. *Leveraging mobility and content caching for proactive load balancing in heterogeneous cellular networks*. Transactions on Emerging Telecommunications Technologies. 2019.

Muhammad B., Kang S. G. *Network-Coding Approach for Information-Centric Networking*. IEEE Systems Journal, p.1–10, 2018.

Nicosia V., Tang J., Mascolo C., Musolesi M., Russo G., and Latora V. *Graph metrics in temporal networks*. Springer Temporal Networks, p. 15–40. 2013.

Oliveira E. M. R, Viana A., Naveen K., Sarraute C. *Mobile data traffic modeling: Revealing temporal facets*. Computer Networks, 112:176–193, 2017.

Radenkovic, M. *A framework for building and deploying the multiparty audio service for collaborative environments*. Presence: Teleoperators & Virtual Environments, 13(6), pp.708-725. 2004.

Radenkovic, M. *Cognitive privacy for personal clouds*. Mobile Information Systems, 2016.

Radenkovic, M. and Huynh, V. S. H. *Low-cost mobile personal clouds*. International Wireless Communications and Mobile Computing Conference (IWCMC) (pp. 836-841). 2016. IEEE.

Radenkovic M., Huynh V.S.H. and Manzoni P. *Adaptive real-time predictive collaborative content discovery and retrieval in mobile disconnection prone networks*. IEEE Access, 6, pp.32188-32206. 2018.

Radenkovic M. and Zakhary S. *Flexible and dynamic network coding for adaptive data transmission in dtns*. IEEE 8th International Wireless Communications and Mobile Computing Conference (IWCMC), p. 567–573, 2012.

Said, A., Shah, S., Farooq, H., Mian, A., Imran, A. and Crowcroft, J. *Proactive Caching at the Edge Leveraging Influential User Detection in Cellular D2D Networks*. Future Internet, 10(10), p.93. 2018.

Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, A. Chaintreau. *CRAWDAD data set Cambridge/haggle*. May 2009.

Selimi, M., Cerdà-Alabern, L., Freitag, F., Veiga, L., Sathiaseelan, A. and Crowcroft, J. *A lightweight service placement approach for community network micro-clouds*. Journal of Grid Computing, 17(1), pp.169-189. 2019.