Joint Cache Placement and Request Routing Optimization in Heterogeneous Cellular Networks

Marisangila Alves, Guilherme Piêgas Koslovski
Graduate Program in Applied Computing - Santa Catarina State University - UDESC
marisangila.alves@edu.udesc.br, guilherme.koslovski@udesc.br

***Abstract*—The 5G Quality-of-Service (QoS) objectives contributed to the Heterogeneous Cellular Network (HCN) evolution, dictating that applications can rely on low-latency and high-bandwidth networks. However, concurrent requests of large amount of multimedia data generate a burden on the backhaul and fronthaul networks due to redundant retransmissions and pose challenges for achieving the QoS objectives. Although mobile network operators can place content closer to the HCN edge to improve the overall QoS indicators, there are still challenges to design a cache policy aware of limited storage capacity, different content popularity, device mobility, and network congestion. This work innovates by introducing a cooperative policy to join caches placement and routing users' requests atop an HCN. By combining networking and cache QoS requirements, the policy balances the fronthaul network load and dynamically maps the caches to HCN resources. We formulated the cache policy through linear programming and in-depth evaluated its performance using extensive simulation scenarios. The results indicate that the proposed network-aware policy decreases the network latency, even when subject to changes in content popularity distribution and total HCN storage capacity.**

***Index Terms*—5G, latency, cache, request routing, placement**

I. Introduction

Mobile devices are largely present in daily activities and have become the most used form of Internet access for end-users. In this context, an evolution in mobile networks is happening to support new applications and services. The Fifth Generation Technology Standard (5G) networks are being implemented and are becoming effectively available in some countries posing new management and administrative challenges to Mobile Network Operators (MNOs) [1]. Among the major QoS requirements defined by 5G, the ultra low latency and the high throughput between end-users and cloud or edge based services deserve to be highlighted. The former opens the opportunity to popularize applications as virtual and augmented reality, Industry 4.0, autonomous vehicles, among others, which have a strict latency requirement [2], while the latter is required by applications based on data transfer operations and mobile video traffic [3].

The physical and logical proximity between resources (services, storage and computing) and end-users in HCN is essential to deliver the 5G QoS requirements [4], [5]. Specifically, the placement of caches on Radio Access Network (RAN), Multi-Access Edge Computing (MEC), and low-power nodes are natural choice to decrease end-to-end latency, increase application's throughput, and to reduce the replicated content load in backhaul network [6], [7]. In this scenario, we claim that the cache placement and requests routing must be jointly performed to deliver QoS for new and running applications. First and foremost, the mobility of end-users on the RAN infrastructures poses a challenge regarding the dynamic routing of data between mobile devices, cache replicas eventually placed on Base Stations (BSs), and external repositories accessed through the backhaul network. Secondly, the heterogeneity of applications requires distinct cache configurations to host multiple concurring users (e.g. a data-sharing application requires more storage cache, while a web page server may require more memory).

Although the specialized literature largely focused on developing caching policies to cache placement [8] and data routing approaches to improve the QoS [9]–[14], the existing approaches do not consider cooperation [8], [9], [15]. Some approaches consider the cooperation only BSs neighbors (one-hop) [10], [11] or decrease the search for content in RAN through hierarchical cooperation [12], [13]. There are strategies which consider multi-hops request routing and cooperation; Nonetheless, the mobility is not considered [14]. Indeed, some proposals ignore the fact that the network can be used by many applications, not just for delivering the services managed by the cache system.

In this sense, this work proposes a cooperative policy aiming the joint placement of caches and users' requests routing on HCNs, which objective is to minimize the latency. The main contributions are three-fold: *(i)* The policy innovates by applying well-known TCP fundamentals to infer formation about the network infrastructure at application layer, specifically bandwidth and Round-Trip Time (RTT) values. By combining networking and cache QoS requirements, the policy balances the network load (to help avoiding network congestion) and performs a dynamic cache to HCN resource mapping (Sec. III) considering the actual link capacity, instead of only analyzing the maximum link bandwidth. *(ii)* The proposed strategy is based on multi-commodity flow problem and formulated as an Linear Programming (LP) model (Sec. IV). *(iii)* Along with the traditional discussions on data distribution and cache storage capacities (Sec. II), the simulation results highlighted that the efficient requests routing can improve the cache hit metric, while simultaneously ensuring the upper-bound latency requirement (4 ms [3]) in most cases (Sec. V). Furthermore, the network-aware feature successfully chose paths (from RAN caches or backhaul to Mobile Devices (MDs)) on different scenarios without significantly impacting latency indicators.

978-1-6654-9792-3/22/$31.00

II. Related work

The work [8] is pioneering in cache placement in wireless networks. The authors formulated an optimal integer LP model and developed a greedy algorithm for minimizing the latency while caching data with restrictions on capacity storage and distribution popularity. In [9] the authors developed a greedy algorithm for joint content placement and request routing, and elaborated two congestion-aware cases while analyzing uncached paths (cloud based) and cached paths (HCN based). The authors in [15] considered a trade-off between using backhaul and RAN, proposing the joint mobility-aware user association and content placement. However, the works [8], [9], [15] do not consider cooperation, in other words, these approaches search the content only in the BSs which the user is associated with. We argue that cooperation between HCN resources (multi-hop) can increase the total BSs storage capacity as well as the optimization search space, latter improving the cache hit and latency indicators, as discussed in Sec. V-D.

The works [10]–[14] considered the cooperation between BSs. While [10] developed a joint content placement and one-hop routing request proposal, [11] proposed the one-hop device-to-device cooperation for jointly implementing the content placement and delivery. The work [12] designed a joint content placement and multi-tier request routing, while [13] developed a framework to deliver content based on the multi-hop cooperation between Small Base Stations (SBSs) and device-to-device networks. The authors in [14] designed the joint cache placement and multi-hop request routing atop unreliable networks. Although these works proposed multi-hops cache policies, they are fundamentally hierarchical, which bounds the content search and consequently the cache hit indicators. Our cache policy has no restrictions on connection hierarchy and analyzes the BSs as a complete set of candidates.

Finally, some approaches consider total available bandwidth for composing the capacity link constraint, ignoring the other competitive network flows, which can cause congestion on links and paths [10]–[12], [14], [15]. We argue that the network representation must be performed based on dynamic RTT values, similar to consolidate TCP literature [16], [17].

III. Cache placement on HCN resources

The cache policy must consider the Service Level Agreement (SLA) requirements specified for each application in two distinct moments: *(i)* initially, when a request is submitted; and *(ii)* periodically to verify the SLA concordance and eventually reconfigure the network paths for guaranteeing the QoS.

A. Graph representation

Given a graph $G(V, E)$, the set of vertices V is composed of base stations (BS), representing Macro Base Station (MBS) and SBS, as well as the mobile devices (MD), caches (C), and external repositories (S, e.g. clouds, edges, hereafter termed backhaul). We resorted to an extended graph technique [18], [19] to combine physical (e.g. BSs, MDs) and logical (e.g. caches, network path, content distribution) information into a single graph. Potential cache repositories are logically connected to backhaul (for retrieving the original data) and to cache-enabled BSs.

A cache service c requires $c_k^s \in \mathcal{N}+$ storage resources, a minimum buffer size defined by c_k^b, and $c_k^{thp} \in \mathcal{N}+$ minimum end-to-end throughput to be efficiently provisioned. In turn, a base station $i \in BS$ has a total storage resource capacity donated by b_i^s (0 indicates that caching is not enabled). It is important to note that the graph composition and its attributes (residual capacities and parameters) represent a snapshot of the infrastructure. Each new snapshot will eventually contain differences from the previous one that must be considered by the policy. Finally, the cache policy combines the new requests with those previously allocated to perform a complete reconfiguration (placement and requests routing), whenever the model's parameters allow, as described as follow.

B. Mobile devices, caches and base stations connectivity

The HCN topology and BS coverage radius are modeled based on geographical distances. Each BS i and MD u has a pair of (x, y) coordinates associated to it, and a function $dis(.) \in \mathcal{R}+$ is applied to account for the cartesian distance between two pairs of coordinates. There is a directional connection $iu \in E$ if $dis(x_i, y_i, x_u, y_u) \leq D_i$, where $D_i \in \mathcal{R}+$ denotes the maximum coverage radius distance for each BS i in the HCN scenario. The connectivity between cache servers and BSs follows the same rationale regarding the directional connection, however guided by a populating algorithm. Initially, any existing algorithm for populating caches based on content access popularity can be applied for distributing content on BSs [20], composing the parameter γ_{ik}, which indicates that a cache k can be potentially placed on $i \in BS$. In turn, the backhaul for retrieving the original data for any cache k is represented by a single entry point S. Finally, $r_{uk} \in \{0, 1\}$ indicates if a MD u is requiring a cache $k \in C$.

C. Perspectives of MNOs and cache providers

For dealing with the dynamism of HCN traffic, the model embraces the TCP congestion control knowledge, specifically on time-sensitive variants [16], [17], and it is constructed based on RTT values. It is worthwhile to highlight that the model considers the management at application layer of TCP/IP stack, however, it employs well-defined concepts from the transport layer. Each $ij \in E$ has a RTT associated to represent the latest sample (denoted by rtt_{ij}). Given the latest RTT, the current estimated throughput [16] for a cache $k \in C$ atop a link $ij \in E$ is defined by $thp_{ijk}^{cur} = \frac{c_k^b}{rtt_{ij}} \in \mathcal{N}+$. The difference between the current estimated throughput and the SLA requirement is represented by $thp_{ijk}^{diff} = thp_{ijk}^{cur} - c_k^{thp} \in \mathcal{N}$. In this sense, the model innovates by defining a network-aware cache policy based on estimation of the actual link capacity, instead of only considering the maximum link bandwidth (as it is usually an HCN classified information - Sec. II). The estimation of actual link (or path) capacity follows the end-to-end design principle of TCP congestion control algorithms enabling a feasible use in users-competitive

scenarios, composed of multiple services and applications. Consequently, the proposed policy is agnostic to concurrent traffic (e.g. applications, data transfer) on network links and paths which are not administrated by the MNO.

The communication mobility is a requirement for 5G, however it poses challenges in guaranteeing network-related QoS. By increasing (or even varying) the distance between devices and BSs, the quality of communication signal is directly affected and, in some cases, the mobility can result on connectivity handovers between SBS. Such facts can lead to packet losses, a factor that directly impacts the RTT. It is possible to deduce that the RTT is related to the distance between the user and the SBSs and furthermore, when the RTT exhibits an increasing (decreasing) trend along the time, we can infer the impact on distance (and vice-versa) [21]. In turn, the RTT obtained in the intermediate paths, that is, through a wired connection, also varies according to the link load [22].

IV. Linear Programming model

The cache policy's objective is to simultaneously maximize the network and cache resources usage while guaranteeing the QoS indicators for those cache providers who requested SLA requirements. The main objective is achieved by jointly performing the cache placement and the request routing. We recur to LP to formally analyze and represent the model.

A. Variables and objective

The LP relays on traditional multi-commodity flow problem for representing the network configuration, jointly considering the capacity of caches and QoS requirements defined by cache providers. In this sense, two binary variables are used: x_{ijk} denotes if the cache content k is flowing through a link $ij \in E$, while y_{ik} indicates the effective use of a cache service k hosted by $i \in BS$ (by setting the value 1, and 0 otherwise). In other words, $y_{ik} = x_{kik}; \forall k \in C, i \in BS$. It is worth mentioning that Eqs. (2)-(4) guarantee that there is a single active path between a cache and an MD.

To achieve the MNOs' and cache providers' perspectives a minimization-based objective function is defined by Eq. (1). Both terms of the objective function aim at load balancing the demands atop the available residual resources. Although applications indicate a minimum throughput requirement (c_k^{thp}), the policy can select higher values (thp_{ijk}^{cur}) based on current HCN load. In this sense, Eq. (1) aims at decreasing the requested-to-allocated throughput ratio to avoid congestion in the network and potentially allocating more requests. Finally, $\delta \to 0$ is a small positive constant to avoid dividing by zero when a cache is not currently offered by a given BS.

$$min \sum_{u \in MD} \sum_{i \in BS} \sum_{k \in C} \frac{(b_i^s - c_k^s) \times r_{uk} \times y_{ik}}{b_i^s \times \gamma_{ik} + \delta} + \sum_{u \in MD} \sum_{ij \in E} \sum_{k \in C} \frac{c_k^{thp}}{thp_{ijk}^{cur}} \times x_{ijk} \times r_{uk} \quad (1)$$

It is worth noting that the decision variables are closely related, that is, the cache placement is defined as a function of the request routing decision, and vice-versa. In addition, the first term from Eq. (1) aims at reducing the number of active cache replicas and avoids the use of the backhaul link, while the second term balances the network load giving priority to high throughput links, hence decreasing the RTT.

B. Constraints

A set of flow- and QoS-related constraints must be satisfied while accounting the LP objective function. Initially, the flow-related constraints are given by Eqs. (2)-(4). While Eq. (2) ensure that all flows will be routed inside the HCN, Eqs. (3) and (4) indicate that the cache data flows to the mobile device.

$$\sum_{i \in BS} x_{jik} \times r_{uk} - \sum_{i \in BS} x_{ijk} \times r_{uk} = 0; \quad \forall j \in BS, \forall k \in C, \forall u \in MD \quad (2)$$

$$\sum_{i \in BS} x_{kik} \times r_{uk} - \sum_{i \in BS} x_{ikk} \times r_{uk} = 1; \quad \forall k \in C, \forall u \in MD \quad (3)$$

$$\sum_{i \in BS} x_{uik} \times r_{uk} - \sum_{i \in BS} x_{iuk} \times r_{uk} = -1; \quad \forall k \in C, \forall u \in MD \quad (4)$$

$$\sum_{u \in MD} \sum_{k \in C} c_k^s \times y_{ik} \times r_{uk} \leq b_i^s; \forall i \in BS \quad (5)$$

$$(x_{ijk} \times r_{uk}) \times c_k^{thp} \leq thp_{ijk}^{cur} \times (x_{ijk} \times r_{uk}); \quad \forall ij \in E, \forall k \in C, \forall u \in MD \quad (6)$$

$$\sum_{i \in BS} y_{ik} \times r_{uk} = 1; \forall k \in C, \forall u \in MD \quad (7)$$

Eqs. (5) and (6) are used to guarantee the QoS requested by cache providers. For performing the cache placement, the model must assure that the hosting HCN components have enough storage capacity (Eq. 5), while the requests routing must guarantee the requested throughput (Eq. 6). Specifically, Eq. (5) accounts the BS storage capacity considering that a cache k can be concurrently accessed by multiple requests. In turn, Eq. 7 ensures that a requested is attended just by one cache source. This approach combined with the latency-oriented formulation (Sec. III-C) aims at decreasing the backhaul pressure while placing the cache content atop fronthaul BSs. However, it is interesting to notice that the policy enables the use of replicas whenever it is necessary to achieve the load balance target by the objective function (Eq. (1)).

V. Simulations and discussions

As a proof-of-concept, we implemented the LP model with the Gorubi Optimizer (9.1), as well as a discrete event simulator (Python 3.9)[1].

A. Simulation parameters

1) HCN configuration: The simulation scenario comprises 2 MBSs each composed of 15 SBSs [11], [23], and the coverage radius (b_i^d) for SBSs is 70 meters [8], [11], [13]. With

[1]The source code is available at https://github.com/marischatten/modeling.

regarding the storage capacity of BSs, the upper-bound limit is defined as 4 GB and 20 GB for SBS and MBS, respectively, except for the scenario described in Sec. V-C2.

2) Devices mobility: The number of MDs is limited to 200 and one MD can be simultaneously connected up to 2 SBSs following the characteristics of ultra-dense networks [24]. At each event, an MD can randomly move around [8] [15] [13] up to 10 meters from the current cartesian point. Initially, each network connectivity $ij \in E$ has rtt_{ij} set as 1 ms [3]. The RTT evolution over the events is driven by two complementary rules, one for the fronthaul optical network and other for the MD. For the fronthaul optical network, the RTT increases exponentially based on links' load [22], while for MD-to-BS connectivity the RTT is based on MDs mobility and cartesian distances [21]. In other words, the rtt_{ij} value for each $ij \in E$ remains between 1 ms and 2 ms, while the RTT for the whole path is given by the sum of all composing links.

3) QoS requirements and caches configurations: For a cache k, a provider can configure c_k^s, c_k^b and c_k^{thp} QoS parameters defining minimum values for storage, buffer size, and network throughput, respectively. While c_k^s is uniformly selected from $\{2, 4, 8\}$ GB values to represent distinct cache services, c_k^b is set as 48 Mb, and c_k^{thp} is defined as 100 Mbps [3]. A total of 100 caches is available and the initial content popularity is defined using a Zipf distribution [25] with $\alpha = 0.8$, except for the scenario defined in Sec. V-C1. Finally, the requests (indicated by r_{uk}) are submitted following a Poisson distribution with $\lambda = 5$ [9] and once provisioned a cache service remains active up to 10 events.

B. Metrics

A set of metrics commonly discussed by the specialized literature (reviewed in Sec. II) was selected to represent MNO and cache providers' perspectives.

1) Cache hit and miss: While cache hit denotes all requests attended with data storage from HCN BSs, the cache miss value indicates all requests sent to the external cloud storage. Intuitively, we can observe that the policy should maximize cache hits to decrease the network delay.

2) Network delay: At each event a snapshot of all HCN resources and MDs is taken and the network latency of all allocated requests is accounted to figure out the efficient of the cache policy in this MNO and cache provider perspective. As the policy acts at the application layer, the network latency is obtained from RTT values [16], [17]. In this sense, the network delay can indicate the HCN saturation points and the efficiency of the load balancing approach.

3) BSs storage loads: The cache policy aims at balancing storage loads, however, such decision can impact on network delay. For investigating this correlation, we collected the percentage of BS storage usage at the end of each event.

C. Simulation Results

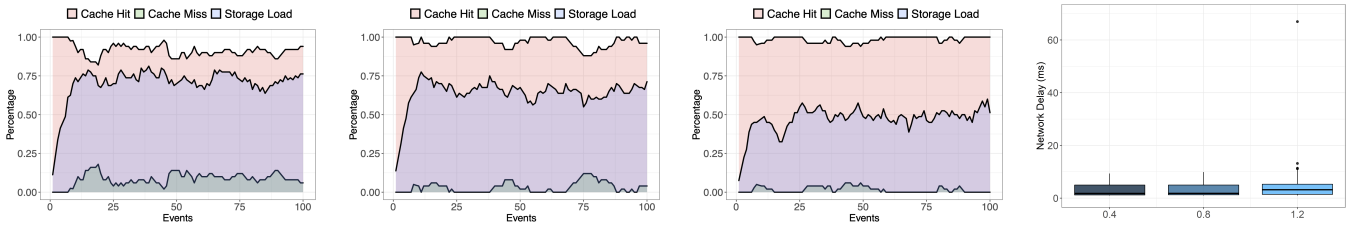
1) Content popularity: The Zipf content access popularity in driven by α value [25], and the higher the value, the greater the concentration of requests on a small subset of data content. To represent distinct scenarios of content access, we varied α with 0.4, 0.8 and 1.2. The Figures 1(a)-1(d) present the simulation results for the cache popularity scenario.

With an analysis as a whole, Fig. 1 indicates that small α values result on small cache hit ratio, the greater the α values, the lower is the cache miss. This fact is justified due to the possibility of allocating more content in caches: the lower the α value, the more varied the requested content, and consequently more storage capacity is used. In other words, for $\alpha = 0.4$ (Fig. 1(a)) the maximum storage load and cache miss were 0.81 and 0.18, respectively, while the minimum cache hit was 0.82. On the other hand, for $\alpha = 1.2$ (Fig. 1(c)) the maximum storage load was 0.6 and the cache hit values increased when compared to lower α values. Thus, more requests were served in cache due to the concentration of content popularity. In turn, the maximum cache misses values were 0.06 and 0.18 for $\alpha = 1.2$ and $\alpha = 0.4$, respectively.

Regarding the cache policy objectives, it is possible to observe a correlation between the content popularity and the use of storage capacity, which triggers effects on the routing of requests (fronthaul and backhaul). This behavior is intensified as the value of α increases. However, even with the lower use of storage capacity and greater concentration of content popularity, it is still possible to observe the occurrence of cache misses potentially originated from the network state, which had a direct impact on the policy decision regarding the routing of requests. Although the number of cache misses is much smaller for the value of $\alpha = 1.2$, there were still requests that were not met in HCN caches. In other words, the network paths between the MDs and the cache repositories were overloaded and the policy chose to deliver the content directly from the backhaul, which in this scenario is advantageous from the perspective of the cache provider and the end user. This behavior emphasizes the importance of a network-aware policy.

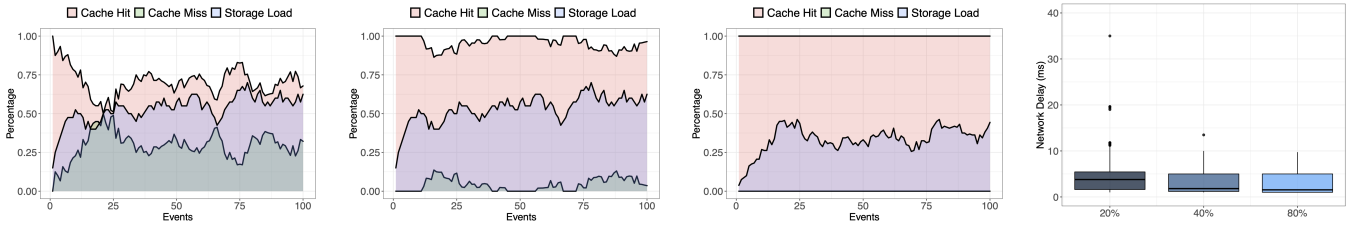
Fig. 1(d) shows that the network delay distribution was not changed, even varying the content popularity. Based on the Pearson coefficient, no statistical correlation was found between the latency and the variation in content popularity. Specifically analyzing Fig. 1(d), for all cases, 50% of the sample remained less than or equal to 4 ms (a requirement determined by [3]). In the last quartile, the network delay remained less than 9.3 ms in the best case and 67 ms in the worst case. This discrepancy is caused by the popularity concentration in specific cache contents, eventually overloading network links or specific cache repositories. Furthermore, the 95^{th} percentile of the sample remained less than 5.9 ms in the best case (Fig. 1(a)) and 7.2 ms in the worst case (Fig. 1(c)).

2) Heterogeneous storage resources: The total HCN storage capacity varied between 20%, 40% and 80% of the total cache library size (requested by MDs). In other words, this parameter represents the amount of storage capacity that is available for the policy analysis. Within a larger configuration (80%) the policy has greater freedom of choice over replicas, while a stricter configuration (20%) requires greater control over the network. The rationale is to allow the cache policy



(a) $\alpha = 0.4$. (b) $\alpha = 0.8$. (c) $\alpha = 1.2$. (d) Network delay.

Fig. 1. Results for the cache popularity scenario. By varying the α parameter distinct scenarios are composed [25].



(a) 20%. (b) 40%. (c) 80%. (d) Network delay.

Fig. 2. Results for the heterogeneous storage resources. The total HCN storage capacity is a percentage of the total library cache size.

to dynamically decide between creating cache replicas or reorganizing the requests routing.

Fig. 2 summarizes the results for this scenario. As expected, cache hits and misses are directly related with total storage capacity and the greater the storage resource, the greater the chance that requests will be allocated in caches. However, the analysis indicates that for HCNs composed of BSs with large storage capacity the use decreases, demonstrating the efficiency of the proposed policy when routing the requests to avoid the over-provisioning of cache replicas. Specifically for HCN total storage capacity of 20% (Fig. 2(a)), the maximum cache miss was 0.5, and the maximum value for cache hit was achieved in the first event due to Zipf initial configuration allied with the low number of requests allocated at this moment. Additionally, the maximum storage load was 0.70, although with a lower value for cache hit. In turn, the maximum storage load was identified as 0.46 for the scenario with 80% of total storage capacity, as given by Fig. 2(c). In summary, it is important to mention that there is a strong statistical correlation between the cache hit and the variation storage capacity (Pearson correlation coefficient of 0.76) and consequently the greater the total storage space, the greater the chance of allocating cached contents.

Fig. 2(d) demonstrates the network delay. The policy balanced the cache placement (first term of Eq. (1)) and the request routing (second term of Eq. (1) combined with network-related constraints) resulting in low network delay. Moreover, for all cases, 50% of the sample remained with latency values less than 4 ms (a requirement defined by [3]). In addition, when analysing the 95^{th} percentile of each sample, the latency remained less than 5.9 ms in the best case (Fig. 2(c)) and 7.3 ms in the worst case (Fig. 2(a)). Following the Pearson coefficient, no statistical correlation was found between the latency and the variation in total storage capacity.

In general, the policy prioritizes the cache placement atop BSs, avoiding overloading both the fronthaul and backhaul networks whenever possible. However, for scenarios with large amounts of total storage capacity (80% and $\alpha = 0.8$), the policy may prioritize the first term of Eq. (1) faced to network QoS requirements (second term). In this sense, the weight of the total storage capacity may unbalance the policy's decision, consequently placing all content in cache (eventually with replicas). When decreasing the total storage capacity of BSs, less possibilities of cache placement and routing paths are available, however even with such constrained scenario, the results pattern are consistent. In fact, the cache policy demonstrated to be adaptive regarding the total storage capacity.

D. Key observations

A network-aware cache policy is essential to increase the overall QoS indicators. This observation is originates from both content popularity and heterogeneous storage capacity scenarios. The policy model was malleable enough to accommodate different Zipfs configurations as well as distinct HCN total storage capacities. Specifically, the network delay analysis indicated that multi-hop request routings atop collaborative BSs can decrease the latency perceived by MDs.

Cache placement is a challenging task, even with a large number of storage resources. Initially, the cache hit values were governed by the Zipf α configuration: the higher the α value, the greater the concentration of MDs requesting the same set of data content. Consequently, a small number of caches must be allocated atop BSs for guaranteeing the QoS. On the other hand, when decreasing the α value, the content

variability is increased and, hence, more storage resource is needed. In fact, by increasing the content popularity, the cache miss indicators are increased too.

The cache policy remains latency-aware even when routing requests through the backhaul. We observed that the storage load never reached the total storage capacity. It means that the policy chooses to retrieve the cache content from the backhaul, even with storage capacity available at the BSs. This phenomenon is justified by the dynamic nature of the scenario, as well as new requests placement and routing reconfigurations decisions taken by the policy. Moreover, the cache policy is guided by dynamically accounted RTT values, which can indicate a temporal overload of fronthaul links or path. Despite this phenomenon, the cache policy remains network-aware, and we observed that the latency remains stable for different content popularity and storage capacities. This behavior is related with the multi-hop approach target by the model, which individually considers all intermediate links composing a path.

VI. Conclusion

The use of caches closer to MDs is a common approach to meet the 5G requirements. Despite all benefits introduced by positioning caches on HCN BSs, this scenario brought a set of challenges to MNOs. Specifically, the total storage capacity of BSs is a limited and highly requested resource, and multiple applications with distinct network QoS requirements are sharing the HCN fronthaul. In this context, we formulate an optimal network-aware cache policy. The model is based on cooperative storage between BSs and proposed the joint cache placement and request routing. Moreover, the network representation is based on RTT values, following the consolidated literature on TCP congestion control algorithms. Extensive simulations demonstrated that the cache policy decreases the network latency, even when subject to changes in content popularity distribution and total HCN storage capacity. The cache policy successfully analyzed multi-hop paths from the HCN to decide between retrieving data through the backhaul or placing replicas on BSs. Independently of the choice, the request routes are constantly reorganized to load balance and decrease the latency. As future work, the model can be extended to deal with other QoS requirements (e.g. processing capacity, delay-sensitive applications) and comparative metrics from the specialized literature, while a second line indicates an implementation on testbeds.

Acknowledgement: This work was developed at LabP2D/UDESC and funding by the National Council for Scientific and Technological Development (CNPq), by the Coordination for the Improvement of Higher Education Personnel (CAPES), and by the Santa Catarina State Research and Innovation Support Foundation (FAPESC).

References

[1] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5G: RAN, core network and caching solutions," *IEEE Comm. Surveys and Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018.

[2] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5g wireless networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[3] I. T. U. ITU, "Minimum requirements related to technical performance for imt-2020 radio interface(s)," Tech. Rep., 2017.

[4] J. G. Andrews, "Seven ways that hetnets are a cellular paradigm shift," *IEEE Communications Magazine*, vol. 51, no. 3, pp. 136–144, 2013.

[5] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 142–149, 2013.

[6] Q.-V. Pham, F. Fang, V. N. Ha, M. J. Piran, M. LE, L. B. LE, W.-J. Hwang, and Z. Ding, "A survey of multi-access edge computing in 5g and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116 974–117 017, 2020.

[7] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.

[8] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 8402–8413, 2013.

[9] M. Dehghan, B. Jiang, A. Seetharam, T. He, T. Salonidis, J. Kurose, D. Towsley, and R. Sitaraman, "On the complexity of optimal request routing and content caching in heterogeneous cache networks," *IEEE/ACM Trans. on Networking*, vol. 25, no. 3, pp. 1635–1648, 2017.

[10] L. Pu, L. Jiao, X. Chen, L. Wang, Q. Xie, and J. Xu, "Online resource allocation, content placement and request routing for cost-efficient edge caching in cloud radio access networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 8, pp. 1751–1767, 2018.

[11] W. Jiang, G. Feng, and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1382–1393, 2017.

[12] X. Li, X. Wang, K. Li, Z. Han, and V. C. M. Leung, "Collaborative multi-tier caching in heterogeneous networks: Modeling, analysis, and design," *IEEE Trans. on Wireless Comm.*, vol. 16, pp. 6926–6939, 2017.

[13] M. Sheng, C. Xu, J. Liu, J. Song, X. Ma, and J. Li, "Enhancement for content delivery with proximity communications in caching enabled wireless networks: architecture and challenges," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 70–76, 2016.

[14] Y. Song, T. Wo, R. Yang, Q. Shen, and J. Xu, "Joint optimization of cache placement and request routing in unreliable networks," *Journal of Parallel and Distributed Computing*, vol. 157, pp. 168–178, 2021.

[15] D. Harutyunyan, A. Bradai, and R. Riggio, "Trade-offs in cache-enabled mobile networks," in *2018 14th International Conference on Network and Service Management (CNSM)*, 2018, pp. 116–124.

[16] L. S. Brakmo, S. W. O'Malley, and L. L. Peterson, "Tcp vegas: New techniques for congestion detection and avoidance," *SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, p. 24–35, Oct. 1994.

[17] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "Bbr: Congestion-based congestion control," *ACM Queue*, vol. 14, September-October, pp. 20 – 53, 2016.

[18] M. Chowdhury, M. R. Rahman, and R. Boutaba, "Vineyard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Transactions on Networking*, vol. 20, pp. 206–219, 2012.

[19] F. R. de Souza, C. C. Miers, A. Fiorese, M. D. de Assunção, and G. P. Koslovski, "Qvia-sdn: Towards qos-aware virtual infrastructure allocation on sdn-based clouds," *Journal of Grid Computing*, 2019.

[20] T. M. Ayenew, D. Xenakis, N. Passas, and L. Merakos, "Cooperative content caching in mec-enabled heterogeneous cellular networks," *IEEE Access*, vol. 9, pp. 98 883–98 903, 2021.

[21] Y. Tian, K. Xu, and N. Ansari, "Tcp in wireless environments: Problems and solutions," *IEEE Com. Magazine*, vol. 43, pp. S27–S32, 2005.

[22] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN Systems*, vol. 17, no. 1, pp. 1–14, 1989.

[23] A. Khreishah, J. Chakareski, and A. Guaraibeh, "Joint caching, routing, and channel assignment for collaborative small-cell cellular networks," *Journal on Selected Areas in Comm.*, vol. 34, pp. 2275–2284, 2016.

[24] M. Kamel, W. Hamouda, and A. Youssef, "Ultra-dense networks: A survey," *IEEE Comm. Surveys Tutorials*, vol. 18, pp. 2522–2545, 2016.

[25] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: evidence and implications," in *Proc. of IEEE INFOCOM*, vol. 1, 1999, pp. 126–134 vol.1.