Exploiting Caching and Cross-Layer Transitions for Content Delivery in Wireless Multihop Networks

Mousie Fasil, Sabrina Müller, Hussein Al-Shatri, and Anja Klein
Communications Engineering Lab, Technische Universität Darmstadt,
Merckstrasse 25, 64283 Darmstadt, Germany
Email:{m.fasil, s.mueller, h.shatri, a.klein}@nt.tu-darmstadt.de

***Abstract*—One of the key challenges in wireless communications is handling the ever growing traffic demand. A large fraction of this traffic is induced by popular content. One way to face this challenge is mobile content caching which improves the system performance by caching content closer to the user. The benefit of content caching depends on the applied content delivery strategy. In this paper, we investigate a scenario where multiple destinations are concurrently requesting a content, which is already cached at mobile devices and then delivered over a wireless multihop network. We propose a content delivery framework which jointly exploits content already cached at mobile devices as well as switching between mechanisms at the physical layer and the network layer in order to optimally deliver the content to all destinations under changing network conditions. In our framework, we use a unified graph model to jointly model the network, the cached content and different mechanisms at the lower three layers. From the unified graph model, an optimization problem is formulated, which is used to find the optimal content delivery strategy. In our numerical evaluation, we show the combined gain of caching and the capability of switching between mechanisms by comparing with conventional schemes which either cannot switch between mechanisms or do not exploit caching.**

I. Introduction

Today, multimedia content is increasingly consumed through wireless networks, which is a challenge considering that such a content consumes a large fraction of the scarce resources in wireless networks. Recent advances in wireless technologies tackle this challenge of limited resources. First, 5G improves the spectral efficiency [1] through advancement of physical layer and medium access layer techniques. Second, device-to-device (D2D) communication [2], which enhances infrastructure-based wireless networks, further improves resource utilization with respect to the reuse of resources. However, the demand for multimedia content is increasing rapidly, and it is estimated that multimedia content will contribute to three-fourths of the overall mobile traffic [3], hence is gonna consume a large part of the available resources.

Therefore, another approach to reduce the impact of multimedia content on wireless resources is required. Content caching is such an approach, where content is stored closer to the user, e.g., at a base station or even directly at mobile devices [4]. Recent research showed that content caching at mobile devices increases the achievable throughput in wireless networks [5]. Caching in wireless networks deals with various research questions, such as, what to cache, where to cache and how to design caching policies [6], [7], [8]. Nevertheless, the amount of work on how to deliver cached content is limited, especially in wireless networks with content already cached at mobile devices. Hence, this paper investigates content delivery where content is already cached at mobile devices with a focus on wireless multihop networks.

The goal is to devise a content delivery strategy which utilizes content cached at mobile devices, while taking into account the underlying network conditions like the network topology, the available resources and the channel quality. Some work on content delivery of cached content is available for wired networks, which has been investigated under the paradigm of information-centric networking [9], [10]. In [11], an analytical content delivery cross-layer framework for wired networks is proposed, which serves multiple users by using network coding and caching techniques. A combination of caching policies and content delivery in wired networks has for example been investigated in [12], where distributed caching algorithms for minimizing the bandwidth costs in a hierarchical cache network with a tree structure are proposed.

A few steps have been made to take into account lower layer aspects into the delivery of cached content in wireless networks. In [7], message coding and multicasting for a single-hop scenario with one source and multiple destinations is considered. Additionally, the authors show that by caching a fraction of the content directly at the destinations, a common message can be forwarded to all users by utilizing multicasting, to improve the achieved rate. In [13], joint caching and routing algorithms for small cell networks are proposed, which minimize the traffic at the macro base station by jointly selecting the cache content of small base stations and assigning each user request to a small base station that has the requested content in their cache. The small base station then delivers the content by utilizing a single-hop unicast transmission. In [14], content replication and routing in multihop networks are considered jointly and scaling laws for the required link capacity are presented. In [15], the authors combine caching and multipath routing to improve the reliability in wireless multihop networks. In [16], a cross-layer approach for a cache-enabled wireless relaying network is presented, where the goal is to minimize the content delivery time.

However, the works mentioned above are not considering changing network conditions at the lower three layers. These changes can be due to variations in the network topology at the network layer (NET), in the availability of resources

978-3-901882-90-6/17 ©2017 IFIP

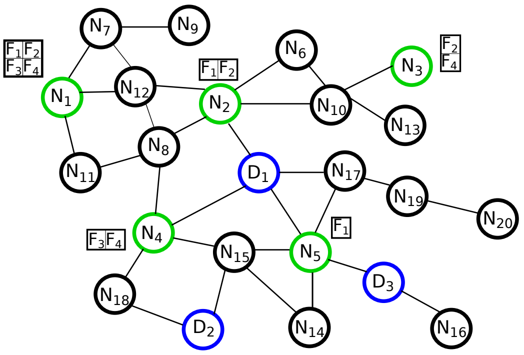

Figure 1: Destinations D_1, D_2 and D_3, marked in blue, request content F, which is distributed over the wireless multihop network, where nodes with the complete or part of the content in the cache are marked in green.

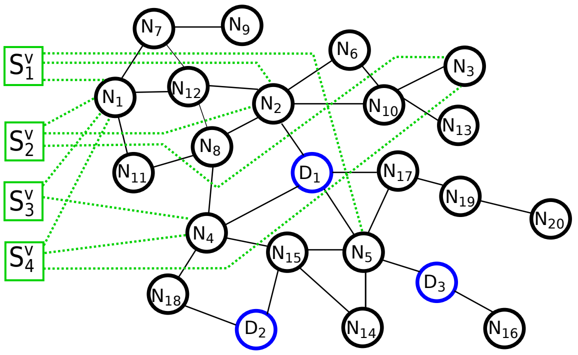

Figure 2: For the four parts F_1, F_2, F_3 and F_4 of content F, four virtual sources S^v_1, S^v_2, S^v_3 and S^v_4 are introduced, which are connected via virtual links with nodes that have the respective part in their cache, indicated by green dotted lines.

at the medium access layer (MAC) and the ever changing quality of the wireless channel at the physical layer (PHY). We propose a content delivery framework which jointly exploits content cached at mobile devices and enables switching, i.e. transitions, between mechanisms at the lower three layers in a wireless multihop network. By this approach, the traffic demand can be handled in a much more flexible manner by delivering content to destinations from close-by nodes which have already cached the content. At the same time, by exploiting transitions, our approach allows to adapt to changing network conditions. Exemplary transitions are the switching between different network support structures at the NET and switching between different transmission mechanisms at the PHY.

In this paper, we consider a scenario in which mobile devices with limited caches have either the complete or a fraction of a content in their cache. Furthermore, the mobile devices form a wireless multihop network in which the content is simultaneously requested by multiple destinations within the network. We propose the following approach for each part of the content, cached at several mobile devices: Determine a mobile device which has the part in the cache. At the same time, obtain the best combination of mechanisms at the lower three layers to deliver the content by taking into account the changing network conditions. For this purpose, a modeling is required which can exploit the cached content in the wireless multihop network and which utilizes the capabilities at the lower three layers. Thus, we present a unified graph model which represents the network, the cached content and the lower three layers. From the unified graph model, we formulate an optimization problem, which corresponds to a multi-source multi-destination sum rate maximization problem. Specifically, the solution of the optimization problem corresponds to the combination of sources from which the cached content should be retrieved and the combination of mechanisms at the lower three layers that should be used in order to best deliver the requested content to all destinations, such that the maximum sum rate is achieved in the system. Consequently, our proposed cross-layer content delivery framework (XCDF) for wireless multihop networks jointly exploits cached content within the network and adapts to changing network conditions by performing transitions.

The remainder of the paper is organized as follows. In Section II, the problem statement is discussed and the graph-based model of the content delivery framework is presented. Based on the unified graph model, a sum rate maximization problem for multiple sources and multiple destinations is formulated in Section III. The simulation results are discussed in Section IV, where the impact of caching and the benefit of transitions are evaluated. Moreover, a comparison between the proposed approach and conventional schemes without the possibility to perform transitions or exploit caching is shown. The paper is concluded in Section V.

II. Problem Statement and System Model

In this section, the content delivery problem and the graph-based model are discussed. First, the wireless multihop scenario is introduced. Next, the modeling of the cached content in the graph-based model is presented. Thereafter, transmission mechanisms at the PHY are incorporated into the unified graph and the scheduling at the MAC for the unified graph is described. Finally, the NET aspects are discussed.

A. *Scenario and Assumptions*

A wireless multihop network consisting of half-duplex nodes is considered. In this scenario, multiple destinations simultaneously request a content, which consists of multiple parts. Furthermore, it is assumed that the content is already cached among nodes in the network, which requested or forwarded the content previously. Hence, a destination may either request the complete content or missing parts of the content. The network is modeled as a graph with nodes $\mathcal{N}^{\text{phy}} = \{N_1, \ldots, N_I, D_1, \ldots, D_M\}$ and links $\mathcal{L}^{\text{phy}} = \{1, \ldots, L\}$. Here, the nodes $\{N_1, \ldots, N_I\} \subset \mathcal{N}^{\text{phy}}$ may cache the complete content or parts of it. The nodes $\{D_1, \ldots, D_M\} \subset \mathcal{N}^{\text{phy}}$ correspond to the destinations which request the complete content or missing parts of it.

The scenario is illustrated exemplarily in Fig. 1, where the content F is composed of four disjoint parts, denoted by

F_1, F_2, F_3, F_4. Five nodes, N_1, N_2, N_3, N_4, N_5, have parts of the content cached and three destination nodes D_1, D_2, D_3 request the complete content $F = \{F_1, F_2, F_3, F_4\}$. As shown in Fig. 1, different nodes have different parts of the content, e.g., N_2 has F_1 and F_2 in the cache, while N_4 has F_3 and F_4 in the cache. Destinations may request any part of a content from any node which has the respective part of the content in its cache. However, a destination cannot request a part of the content from more than one node. In addition, each destination requires the complete content $F = \{F_1, F_2, F_3, F_4\}$. Therefore, the goal is to determine for each destination and each part of the content from which node that part of the content should be retrieved and through which paths it should be delivered. It has to be noted that these decisions are coupled and depend on the available paths, resources and channel qualities with respect to all destinations. The number of possible forwarders as well as the number of possible paths to the destinations are very high when the number of nodes with cached content is high, which enlarges the complexity of the problem. Thus, a modeling is required which can cope with a high number of nodes with cached content. Specifically, the model needs to take into account the possibility that nodes cache only parts of the content.

B. Modeling of Cached Content

It is assumed that a content F is separable and composed of P parts, $F = \bigcup_{p=1}^{P} F_p$, where the parts are disjoint, i.e., $F_p \cap F_q = \emptyset$, $\forall p \neq q$. We denote by $P_i \in \{0, ..., P\}$ the number of parts of content F cached in node N_i. Since every part F_p of content F can be cached at multiple nodes, each part F_p is modeled as a virtual source S_p^{v} in our unified graph. Each S_p^{v} is connected through virtual links to nodes that have the content cached. The model is illustrated in Fig. 2, where the four parts are modeled as virtual sources S_1^{v}, S_2^{v}, S_3^{v} and S_4^{v}, respectively. Each virtual source is connected through virtual links with nodes that have the corresponding part F_p in their cache, indicated by the green dotted lines. For example, S_1^{v} is connected to N_1, N_2 and N_5.

C. Utilizing Transitions for Content Delivery

In order to ensure an efficient content delivery, the changing network conditions at the lower three layers are taken into account. Moreover, the lower three layers have to be considered and adapted jointly. In order to achieve this, performing transitions at the lower three layers is proposed. In more detail, transitions between wireless transmission mechanisms at the PHY and between tree and butterfly structures at the NET are considered. Consequently, a unified graph is developed which jointly models the content and transmission mechanisms. Based on the unified graph, a conflict-free scheduler at the MAC is proposed, which performs the scheduling taking into account the available transmission mechanisms at the PHY. In the following, the above mentioned aspects are discussed in detail.

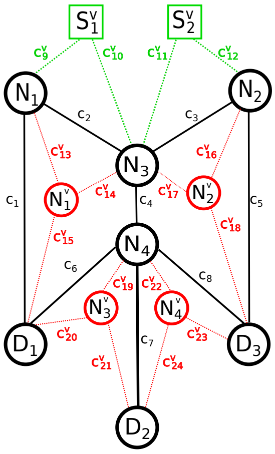

Figure 3: Virtual sources S_1^{v} and S_2^{v} represent parts F_1 and F_2 of content F and are connected through virtual links to nodes which have cached F_1 or F_2, respectively. Virtual nodes N_1^v and N_2^v represent the broadcast of N_1 and N_2, respectively, while N_3^v and N_4^v represent two possible multicasts of N_4.

Transmission mechanisms: In order to utilize the broadcast nature of the wireless medium, we consider all three transmission mechanisms, namely, unicast, multicast and broadcast. Thus, a node can switch between either forwarding the content to one neighbor node using unicast, to a group of neighbor nodes using multicast or to all neighbor nodes using broadcast. In the unified graph, the links in $\mathcal{L}^{\mathrm{phy}}$ represent the unicast transmission mechanism, where a link $l \in \mathcal{L}^{\mathrm{phy}}$ corresponds to the physical connection between two nodes and the respective link capacity is expressed as c_l, which is normalized to one. In order to consider all transmission mechanisms in the graph-based model, node virtualization is applied to represent multicast and broadcast transmissions. Node virtualization, cf. [17], extends a given graph by adding virtual nodes and links to the graph, which together represent either a multicast or a broadcast transmission. For each node with at least two outgoing links, a virtual node is added which is then (i) connected via virtual links to a subset of neighbor nodes to represent multicast transmission or (ii) connected to all neighbor nodes to represent broadcast transmission. This is illustrated in Fig. 3. The black solid edges represent the unicast links between the nodes, e.g., the direct link between N_1 and D_1. A broadcast is represented by introducing a virtual node which is connected to the originating node and all respective neighboring nodes. As an example, virtualization is applied to N_1, where the virtual node N_1^{v} is introduced and connected to the respective neighbors of N_1, namely, N_3 and D_1. The link capacity of the virtual links between N_1^{v} and N_1, N_3 and D_1 are set to the minimum of the unicast links of N_1, given by $c_{13}^{\mathrm{v}} = c_{14}^{\mathrm{v}} = c_{15}^{\mathrm{v}} = \min\{c_1, c_2\}$. Based on the same principle, virtualization can be applied to represent a multicast transmission. However, repeating node virtualization multiple times to represent all possible transmission mechanisms increases the size of the unified graph exponentially. Therefore, a quantized node virtualization is proposed, where each node quantizes there outgoing link capacities. Quantized node virtualization limits the increase in size of the unified graph, while simultaneously conserving the broadcast gain. This is shown in Fig. 3 for node N_4. Here, quantized node virtualization is applied, to represent two possible multicast possibilities for N_4, which are depicted in Fig. 3, given by N_3^{v} and N_4^{v}. In comparison to represent all three possible multicast and the broadcast transmissions, node virtualization has to be applied four times and thus introducing four virtual nodes to the unified graph.

The unified graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ includes the set of physical nodes $\mathcal{N}^{\mathrm{phy}} \subset \mathcal{N}$, the set of physical unicast links $\mathcal{L}^{\mathrm{phy}} \subset \mathcal{L}$, the set of virtual sources $\mathcal{S}^{\mathrm{v}} \subset \mathcal{N}$ representing a content, the set of virtual nodes $\mathcal{N}^{\mathrm{v}} \subset \mathcal{N}$ representing the multicast and broadcast transmission mechanisms, and the set of virtual links $\mathcal{L}^{\mathrm{v}} \subset \mathcal{L}$ to connect virtual sources from $\mathcal{S}^{\mathrm{v}}$ and virtual nodes from $\mathcal{N}^{\mathrm{v}}$ with physical nodes of the network. For each node $N \in \mathcal{N}$ in the unified graph, we denote the sets of its outgoing links and its incoming links as $\mathcal{O}(N) \subset \mathcal{L}$ and $\mathcal{I}(N) \subset \mathcal{L}$, respectively.

Conflict-free Scheduler: Another important aspect is the scheduling of resources at the MAC. Concurrent transmission through the wireless medium may introduce collisions and, hence, degrade the utilization of available resources. A collision occurs when a node receives multiple transmissions at the same time or when a node is transmitting and receiving at the same time. Therefore, a conflict-free scheduling is required, which simultaneously schedules as many nodes as possible while avoiding collisions between nodes. This is achieved by splitting the unified graph $\mathcal{G}$ into K collision-free subgraphs $\mathcal{G}_k = (\mathcal{N}_k, \mathcal{L}_k)$. The proposed conflict-free scheduler determines the subgraphs by scheduling unicast transmissions such that no two adjacent edges are assigned into the same subgraph, which corresponds to the edge coloring problem [18]. Moreover, broadcast transmissions are scheduled such that no two adjacent vertices are assigned into the same subgraph, which corresponds to the vertex coloring problem [18]. The conflict-free scheduler is a heuristic, which tries to obtain a minimal number K of independent subgraphs using the following steps:

1) Determine the degree of the nodes in $\mathcal{G}$.
2) Initialize an empty queue and add the nodes with respect to their degree in descending order.
3) Set $i = 1$
4) Set $\mathcal{G}_i = \emptyset$.
5) Add the first node in the queue to $\mathcal{G}_i$ and remove the first node from the queue.
6) Compare every node in $\mathcal{G}_i$ with the next node in the queue.
7) If there is no conflict, add the node to $\mathcal{G}_i$ and remove it from the queue, else continue.
8) $i = i + 1$
9) Repeat step 4)-8) until queue is empty.

Network Support Structures: A network support structure identifies a set of nodes favorable to establish a communication between source and destination nodes. In this paper, two network support structures are considered, the tree and the butterfly structure [17]. In a tree structure, the aim is to use the smallest number of nodes to forward the content, which allows selected nodes to better utilize the shared wireless medium. The aim of a butterfly structure is to exploit as many independent paths as possible, in order to apply network coding between different contents or different parts of a content flowing through the wireless multihop network. Based on the unified graph and the scheduling, our content delivery framework determines if the content should be delivered through a tree or through butterfly structure.

III. Content Delivery Framework

In this section, a multi-source multi-destination sum rate maximization problem for content delivery is formulated. As already mentioned, it is assumed that the current cache content of the nodes in the network is known. Hence, the proposed content delivery framework can be combined with any caching policy. In the proposed optimization problem, the multiple sources correspond to the virtual sources which represent the content. The content delivery framework aims at maximizing the sum rate in the system by choosing which nodes forward which parts in their cache to the destinations. At the same time, the best combination of mechanisms to forward the content at the lower three layers is determined. Hence, the utility function is formulated as

$$\max \sum_{m=1}^{M} \min_{p} r(m,p), \tag{1}$$

where $r(m,p) \geq 0$ is the rate achieved between virtual source S_p^{v} and destination D_m. The objective in (1) expresses $i)$ that each destination receives each part with the same rate ensuring that the complete content is received and $ii)$ the maximization of the sum rate in the system.

The flow from virtual source S_p^{v} to destination D_m over a link l in the k-th subgraph is defined as $f_l^{(k)}(m,p)$. The outgoing flow from virtual source S_p^{v} to destination D_m over the outgoing virtual link l is $r(m,p)$ if the link is activated, else it is zero. The activation of a virtual link l between virtual source S_p^{v} and destination D_m is indicated with the binary variable $y_l(m,p)$. Hence, the flow constraint for every virtual source is expressed as

$$\sum_{k=1}^{K} f_l^{(k)}(m,p) = \begin{cases} r(m,p), & \text{if } y_l(m,p) = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$
$$p \in \{1,...,P\}, l \in \mathcal{O}(S_p^{\mathrm{v}}), m \in \{1,...,M\}.$$

Since we assume that for a given destination, a given part F_p of the content can only be retrieved from one node, only one outgoing virtual link of a virtual source can be activated for serving D_m. This can be expressed as

$$\sum_{l \in \mathcal{O}(S_p^{\mathrm{v}})} y_l(m,p) \leq 1, \tag{3}$$
$$p \in \{1,...,P\}, m \in \{1,...,M\}.$$

The rate between S_p^{v} and D_m is constrained by the maximum flow in the network [19]. Every node in $\mathcal{N}^{\mathrm{phy}} \cup \mathcal{N}^{\mathrm{v}}$ must uphold the flow conservation, which conveys that any incoming flow into a node must depart from the node. The flow conservation constraint for forwarding nodes is given by

$$\sum_{k=1}^{K} \left(\sum_{l \in \mathcal{O}(N)} f_l^{(k)}(m,p) - \sum_{l \in \mathcal{I}(N)} f_l^{(k)}(m,p) \right) = 0, \tag{4}$$
$$\forall N \in (\mathcal{N}^{\mathrm{phy}} \cup \mathcal{N}^{\mathrm{v}}) \setminus \{D_m\}, p \in \{1,...,P\}, m \in \{1,...,M\}.$$

The flow constraint for destinations is given by

$$\sum_{k=1}^{K} \left(- \sum_{l \in \mathcal{I}(D_m)} f_l^{(k)}(m,p) \right) = -r(m,p), \tag{5}$$
$$m \in \{1,...,M\}, p \in \{1,...,P\}.$$

Every flow in the network is upper bounded by a capacity constraint. The capacity of a link l in subgraph $\mathcal{G}_k$ depends on the link capacity c_l and the duration the link is utilized in the k-th subgraph, which is determined by the timeshare factor τ_k. If a link is part of subgraph $\mathcal{G}_k$, the indicator function $\mathbf{I}_{\mathcal{L}_k}(l)$ is one, else the flow on that link is zero in $\mathcal{G}_k$. The indicator function is written as

$$\mathbf{I}_{\mathcal{L}_k}(l) = \begin{cases} 1, & \text{if } l \in \mathcal{L}_k, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

The capacity constraint is expressed as

$$0 \leq \sum_{p=1}^{P} f_l^{(k)}(m,p) \leq \tau_k \cdot c_l \cdot \mathbf{I}_{\mathcal{L}_k}(l), \tag{7}$$
$$\forall l \in \mathcal{L}, k \in \{1,...,K\}, m \in \{1,...,M\}.$$

This bounds all flows through a link. The timeshare factors are normalized and bounded as follows

$$\sum_{k=1}^{K} \tau_k = 1, \tag{8}$$
$$0 \leq \tau_k \leq 1, \qquad k \in \{1,...,K\}. \tag{9}$$

The sum rate maximization problem for multiple sources and multiple destinations is expressed in (1) - (9). We rewrite the above problem as a binary linear problem. First, the objective function formulated in (1) is a piecewise-linear function and can be reformulated [20] as

$$\max \sum_{m=1}^{M} t_m, \tag{10}$$

where t_m is an auxiliary variable which is constrained by

$$t_m \leq r(m,p), \tag{11}$$
$$m \in \{1,...,M\}, p \in \{1,...,P\}. \tag{12}$$

Furthermore, the constraint in (2) can be written either as one binary non-linear constraint or as a set of three binary linear constraints by applying the big-M method [21]. Here, the latter approach is chosen. The first constraint sets the outgoing flows of S_p^{v} to zero if $y_l(m,p)$ is equal to zero which is expressed as

$$\sum_{k=1}^{K} f_l^{(k)}(m,p) \leq M_1 y_l(m,p), \tag{13}$$

where M_1 is a constant that should be chosen sufficiently large but close enough to the upper bound of $\sum_{k=1}^{K} f_l^{(k)}(m,p)$. Since the maximum flow is bounded by the link capacities in the network and the maximum physical capacity in the unified graph is normalized to one, we can set $M_1 = 1$. Next, two constraints are introduced to enforce that the outgoing flows are set equal to $r(m,p)$ when $y_l(m,p)$ is equal to one. This is formulated as

$$\sum_{k=1}^{K} f_l^{(k)}(m,p) - r(m,p) \leq M_2(1 - y_l(m,p)), \tag{14}$$

and

$$r(m,p) - \sum_{k=1}^{K} f_l^{(k)}(m,p) \leq M_3(1 - y_l(m,p)), \tag{15}$$

where M_2 is set as the upper bound of the left-hand side expression in (14) and M_3 as the upper bound of the left-hand side expression in (15). Since the link capacities in our model are normalized to one, the maximum rate and flows in the network are upper bounded to one and hence $M_2 = M_3 = 1$.

IV. Simulation Results

In this section, the impact of caching and lower layer transitions is evaluated in terms of the achievable sum rate. This is done over 100 snapshots of random networks, where sixteen nodes are uniformly distributed in an area with a map size of 15 m by 5 m. The simulation results are obtained through MATLAB [22], using CVX [23] and Gurobi [24] to solve the binary linear problem. The proposed cross-layer content delivery framework is abbreviated XCDF. XCDF is compared against two schemes which cannot perform transitions. The first one, called BBC, is using the butterfly structure at the NET and broadcast at the PHY. The second scheme, called TUC, is utilizing the tree structure at the NET and unicast at the PHY. In the simulation, it is assumed that the content consists of four parts and that each of the four parts is available in at least one node. Furthermore, the performance of the content delivery framework is evaluated by studying the impact of the average number of parts a node has in the cache. In our simulation, the number P_i of parts a node has in its cache is sampled from a uniform distribution. Four cases are investigated $i)$ each node has one part of the content ($X = 1$) in its cache, $ii)$ each node has less than or equal to two random parts ($X \leq 2$) in its cache, $iii)$ each node has less than or equal to three random parts ($X \leq 3$) in its cache and $iv)$ each node has less than or equal to four random parts ($X \leq 4$) in its cache.

Fig. 4 shows the average sum rate achieved by XCDF over the percentage of nodes with at most X parts of the content in their cache for the four cases of X. In all four cases, it can

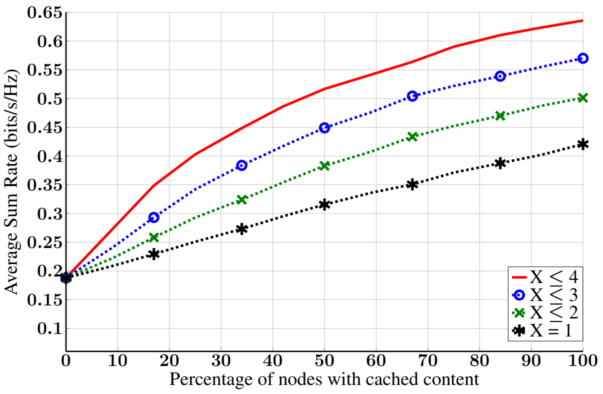

Figure 4: Sum rate achieved by XCDF vs. relative number of nodes with cached content for different X, where X is the maximum number of parts cached.

be observed that an increasing number of nodes with cached content improves the average sum rate. This is due to the fact that the content is getting closer to the destinations as the number of caching nodes increases. Additionally, it can be observed that the performance increases the more parts of a content are cached at each node. In the case of $X \leq 4$, the maximum sum rate achieved is 0.64 bits/s/Hz, which is a gain of 12% compared to $X \leq 3$, a gain of 28% compared to $X \leq 2$ and a gain of 52% compared to $X = 1$. The curves show a concave behavior, except for the case $X = 1$ where the curve follows a linear trend.

In Fig. 5, the average sum rate achieved by XCDF is shown for the cases when 75%, 50%, 25% of nodes in the network have parts of the content in their cache, as well as when the caches of the nodes are empty. XCDF achieves an average sum rate of 0.19 bits/s/Hz without caching. In comparison, the average sum rate increases by 33% when $X = 1$, 54% when $X \leq 2$, 80% when $X \leq 3$ and 112% when $X \leq 4$, for the case that 25% of nodes have cached some parts of the content. This shows the advantage of exploiting caching, but it also shows that the number of parts cached at each node has an impact on the achievable sum rate. As an example, an average sum rate of 0.3 bits/s/Hz is achieved for the case $X = 1$ when 50% of the nodes have cached some parts of the content. Similarly, this can be observed for the case $X \leq 4$ which requires 25% of the nodes with parts of the content in the cache. The reasons is that a nearby node with multiple parts of the content in the cache will provide all parts in the cache to a destination if selected, thus reducing the number of active sources and increasing the amount of available resources. In summary, both factors, i.e., the number of nodes with content in the cache and the amount of parts cached at nodes play a role.

Another advantage of XCDF is the utilization of transitions at the lower layers. Fig. 6 shows the average sum rate over the percentage of nodes with part of the content in the cache for BBC, TUC, and XCDF for the two cases $X = 1$ and $X \leq 4$. When the caches of the nodes are empty, XCDF achieves an average sum rate of 0.19 bits/s/Hz, which is two times higher than for TUC and 1.5 times higher compared to BBC. For the case $X = 1$, XCDF outperforms both TUC and BBC, where XCDF achieves an average sum rate of 0.42 bits/s/Hz, which is a gain of 50% and 40% compared to TUC and BBC, respectively. Moreover, XCDF outperforms both TUC and BBC for the case of $X \leq 4$, where XCDF achieves an average sum rate between 0.19 bit/s/Hz and 0.64 bits/s/Hz. This results in an average gain of 40% compared to TUC and BBC. Thus, Fig. 6 shows the benefit of exploiting caching and performing transitions, since XCDF can adapt to changing network conditions by switching between unicast and broadcast at the PHY and between tree and butterfly at the NET.

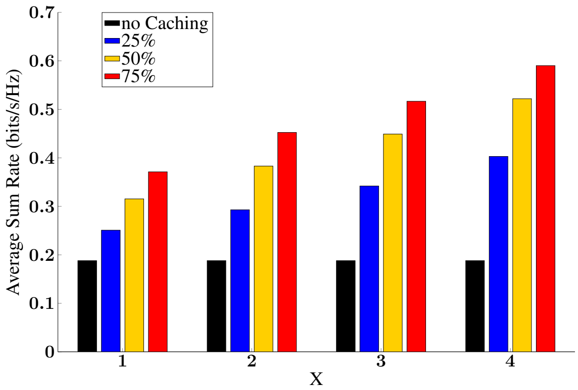

Figure 5: Average sum rate for different X of parts cached at nodes for 0%, 25%, 50% and 75% of nodes with cached content.

V. Conclusion

In this paper, a content delivery framework for wireless multihop networks is proposed. The content delivery framework exploits the fact that content is already cached at different nodes. Furthermore, the framework performs transitions at the lower layers to adapt to changing network conditions. A unified graph model is proposed, which jointly models cached content, network support structures and different transmission mechanisms. The different parts of a content are modeled as virtual sources from which the destinations simultaneously request and retrieve the complete content. Based on the unified graph model, a multi-source multi-destination sum rate maximization problem is formulated. The evaluation of the content delivery framework XCDF shows that by exploiting caching, the average sum rate can be increased by a factor of two to three depending on the amount each node has cached compared to the case that caching is not exploited. Furthermore, as the number of nodes with cached content increases, the sum rate increases since it is more likely that a content is cached close to the destinations. Another advantage of the proposed scheme is the utilization of transitions, which achieves a gain of 40% on average compared to conventional schemes which cannot adapt to changing network conditions. As a next step, the aim is to design a distributed algorithm, which can work solely on information available locally at each node, e.g., cache information and channel information.

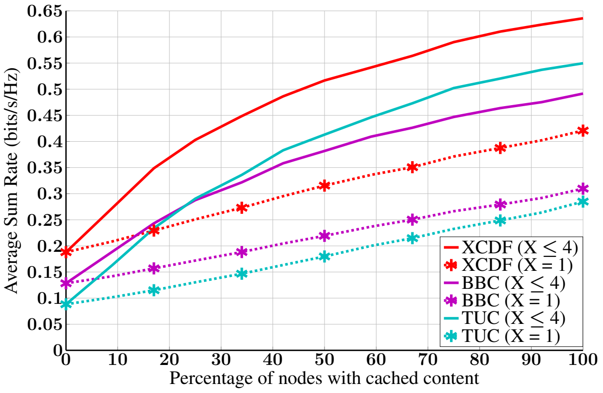

Figure 6: Average sum rate for $X = 1$ and $X \leq 4$ for BBC, TUC and XCDF.

VI. Acknowledgment

This work has been funded by the German Research Foundation (DFG) as part of project B3 and C1 within the Collaborative Research Center (CRC) 1053 – MAKI. The authors would like to thank Michael Zink for his feedback.

References

[1] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, H. Tullberg, M. A. Uusitalo, B. Timus, and M. Fallgren, "Scenarios for 5G mobile and wireless communications: the vision of the METIS project," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 26–35, May 2014.

[2] L. Lei, Z. Zhong, C. Lin, and X. Shen, "Operator controlled device-to-device communications in LTE-advanced networks," *IEEE Wireless Communications*, vol. 19, no. 3, pp. 96–104, June 2012.

[3] "White paper: Cisco visual networking index data traffic forecast update, 2014-2019," Cisco, Tech. Rep., 2015.

[4] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. Leung, "Cache in the air: exploiting content caching and delivery techniques for 5G systems," *IEEE Communications Magazine*, vol. 52, pp. 131–139, 2014.

[5] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE Journal on Selected Areas in Communications*, vol. 34, pp. 176–189, 2016.

[6] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, pp. 2856–2867, 2014.

[7] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "On the average performance of caching and coded multicasting with random demands," in *Proc. IEEE International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 922–926.

[8] S.-W. Jeon, S.-N. Hong, M. Ji, and G. Caire, "Caching in wireless multihop device-to-device networks," in *Proc. IEEE International Conference on Communications (ICC)*, 2015, pp. 6732–6737.

[9] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine*, vol. 50, pp. 26–36, 2012.

[10] C. Fang, F. R. Yu, T. Huang, J. Liu, and Y. Liu, "A survey of energy-efficient caching in information-centric networking," *IEEE Communications Magazine*, vol. 52, pp. 122–129, 2014.

[11] J. Llorca, A. M. Tulino, K. Guan, and D. C. Kilper, "Network-coded caching-aided multicast for efficient content delivery," in *Proc. IEEE International Conference on Communications (ICC)*, 2013, pp. 3557–3562.

[12] S. Borst, V. Gupta, and A. Walid, "Distributed caching algorithms for content distribution networks," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[13] K. Poularakis, G. Iosifidis, and L. Tassiulas, "Approximation algorithms for mobile data caching in small cell networks," *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3665–3677, Oct 2014.

[14] S. Gitzenis, G. S. Paschos, and L. Tassiulas, "Asymptotic laws for joint content replication and delivery in wireless networks," *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 2760–2776, May 2013.

[15] A. Valera, W. K. G. Seah, and S. V. Rao, "Cooperative packet caching and shortest multipath routing in mobile ad hoc networks," in *Proc. IEEE International Conference on Computer and Communications (INFOCOM)*, vol. 1, 2003, pp. 260–269.

[16] L. Xiang, D. W. K. Ng, T. Islam, R. Schober, and V. W. S. Wong, "Cross-layer optimization of fast video delivery in cache-enabled relaying networks," in *IEEE Global Communications Conference (GLOBECOM)*, 2015, pp. 1–7.

[17] M. Fasil, A. Kuehne, and A. Klein, "Node virtualization and network coding: Optimizing data rate in wireless multicast," in *Proc. IEEE International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 573–578.

[18] J. A. Bondy and U. S. R. Murty, *Graph theory with applications*. Citeseer, 1976, vol. 290.

[19] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows : theory, algorithms, and applications*. Prentice Hall, 1993.

[20] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[21] I. Griva, S. G. Nash, and A. Sofer, *Linear and nonlinear optimization*. Siam, 2009.

[22] I. The MathWorks, "MATLAB 2013a," Natick, Massachusetts, United States.

[23] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[24] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2016. [Online]. Available: http://www.gurobi.com