Quality-Aware Caching, Computing and Communication Design for Video Delivery in Vehicular Networks

Ting-Yen Kuo, Ming-Chun Lee, *Member, IEEE* and Ta-Sung Lee, *Fellow, IEEE*
Institute of Communications Engineering, National Yang Ming Chiao Tung University, Taiwan
Email: {cissykuo.cm08g, mingchunlee}@nctu.edu.tw, tslee@mail.nctu.edu.tw

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9838410

***Abstract*—To satisfy the increasing demands of wireless traffic in vehicular networks, how to significantly improve vehicular networks becomes a critical issue. Motivated by the potential benefits of jointly using edge-computing and edge-caching in vehicular networks, this paper considers investigating the quality-aware caching, computing and communication (3C) optimization for video delivery in vehicular networks. By incorporating the quality-awareness with 3C models, we formulate a quality-aware joint 3C optimization problem. Then, by considering the practice that the caches might be pre-determined along with the formulated 3C problem, we obtain the quality-aware joint 2C optimization problem with caching. We propose effective approaches to solve these two problems. Simulation results show that our proposed approaches can outperform the reference schemes significantly.**

I. Introduction

Due to the increasing demands of wireless traffic in vehicular networks, how to significantly improve the vehicular networks becomes a critical issue [1]. Among those newly proposed solution technologies for improving vehicular networks, edge-computing and edge-caching have been deemed as two of the most promising technologies, as they can effectively increase throughput and reduce latency [2], [3].

To bring benefits of edge-computing and edge-caching to vehicular networks, topics in edge-computing and edge-caching have been intensively investigated. However, the edge-caching and edge-computing in vehicular networks were investigated individually as they were motivated by different applications. Recently, this has been changed as many new applications require the computational process based on large amount of data, and it has been shown that combining caching with computing can bring significant benefits [8]–[11].

Being one of the major sources for wireless traffic in vehicular networks, video service can benefit significantly from jointly considering caching, computing and communication (3C) [4], [5]. This is because (i) different user satisfaction can be obtained by videos with different qualities [6]; (ii) video delivery with different qualities require the transcoding of the video [7]; and (iii) delivering videos with quality-awareness can improve the overall network performance as videos with different qualities has different communication requirements [6]. However, the investigation of quality-aware 3C optimization for vehicular networks is far from sufficient as the current literature might not fully address the issues in 3C optimization and the quality-awareness has not been considered much in the context of 3C optimization. For example, studies in [8]–[11], [14], [15] mainly focused on jointly optimizing computing and caching in roadside units (RSUs) of the vehicular network, while the caching and computing abilities of vehicles were overlooked. Refs. [10]–[12] considered caching and computing for both RSU and vehicles. However, [10] and [11] focused on the joint computing and caching design without optimizing the communications, while [12] focused on caching design only without jointly optimizing computing and communications. Ref. [13] considered the joint 3C optimization for both vehicles and RSUs, but assuming that only the idle vehicles can cache and compute videos. To the best of our knowledge, the quality-awareness was considered only in [15]. However, [15] considered only 3C ability in RSUs.

In this paper, we consider the quality-aware 3C optimization in vehicular networks, where both the RSUs and vehicles have 3C capabilities. We assume a vehicle can have multiple requests at a time as the vehicles, e.g., buses and cars, can have more than a single passenger. Observing that delivering videos with different qualities can lead to different network utilities and 3C requirements, and that different RSUs and vehicles have different 3C capabilities, we formulate a joint 3C optimization problem considering all the heterogeneity. Given this 3C optimization problem, we discuss how to modify the problem into feasible problems under different network conditions. Furthermore, by considering the practical assumption that the cached videos can be pre-determined, we propose a quality-aware joint computing and communication (2C) optimization problem with given cached content. To the best of our knowledge, such problem is also a critical but unexplored problem.

As both the 2C and 3C optimization problems are mixed integer nonlinear problems, which are non-trivial to solve,

This work was partially supported by the"Center for Open Intelligent Connectivity" under the Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) of Taiwan, and partially supported by the Ministry of Science and Technology (MOST) of Taiwan under grants MOST 110-2634-F-009-028, MOST 110-2224-E-A49-001, MOST 110-2622-E-A49-004, MOST 110-2222-E-A49-006, and MOST 110-2221-E-A49-025-MY2. This work was also funded in part by Qualcomm through a Taiwan University Research Collaboration Project.

we propose effective solution approaches for them. Since the formulated 2C optimization is a subproblem of the 3C optimization, we first solve the 2C problem by judiciously transforming it to an integer problem which can be reformulated as the submodular maximization with multidimensional knapsack constraints, and thus be solved effectively. Then, given the solution approach of the 2C problem, we propose a heuristic approach that can provide effective solution to the 3C optimization problem. We evaluate the proposed 3C and 2C designs by simulations. Results show that our proposed designs can outperform the reference schemes significantly.

II. Network Model

A. Network Architecture

We consider a vehicular network which includes a base station (BS), several RSUs, buses, and cars. We assume the network is split by the RSUs such that the area covered by different RSUs are non-overlapped; thus, the area covered by a RSU is called a cluster. We consider that the network wants to provide video services to vehicles (i.e., buses and cars), where a content with a given quality would be delivered to a vehicle when the content is requested by the vehicle. We consider the quality-awareness for the network. Thus, a content could be transcoded (if needed) to different formats with different qualities before delivery. We assume a vehicle can have multiple requests at a time, which corresponds to the practice that the in-vehicle WiFi could exist and there might exist multiple passengers in a vehicle.

We consider the RSU, buses, and cars all have the 3C capabilities so that the 3C can be conducted by any of them. We assume vehicles in a cluster can only be served by the RSU of the cluster via a direct RSU-to-vehicle (R2V) link and by the vehicles in the same cluster via vehicle-to-vehicle (V2V) communications. We assume both the RSU and vehicles can only transcode and deliver the content cached by them due to limited infrastructure support.[1] We assume that the frequency reuse approach is used for the clusters so that different clusters would not interfere with one another. By the above considerations, we can thus focus on a single cluster at a time-slot, where we assume that the vehicles are invariant within each time-slot.

We assume a RSU and $N(t) = N_b(t) + N_c(t)$ vehicles exist in the cluster at time-slot t, where $N_b(t)$ and $N_c(t)$ are the numbers of buses and cars in the cluster at time t, respectively. Since the vehicles can be the transmitter and/or the receiver, we in the reminder of the paper denote n as the index of the vehicle n serving as the receiver who has requests to satisfy. On the other hand, we denote m as the the index of the vehicle/node m serving as the transmitter to deliver videos. Note that for convenience, we let $M = N + 1$ be the index for the RSU which only provides service without having any request. Since satisfying a request might need 3C functionalities, we in the following introduce the corresponding models.

[1]Although this might result in that some requests cannot be served by the RSU and vehicles, they might still be handled via the BS with low quality which serves as the backup solution for the outages.

B. Communication Model

To define the communication model, we need to define who to associate with whom and what to communicate in the network. Therefore, in our communication model, we denote $y_{m,n}^{(c,f,f')}(t) \in \{0,1\}$ as the decision indicator which indicates that if $y_{m,n}^{(c,f,f')}(t) = 1$, vehicle n should receive content c with quality f from node m, where the delivered content is transcoded from content c with quality f' cached in node m; otherwise $y_{m,n}^{(c,f,f')}(t) = 0$. We assume the communication links in a cluster share the total bandwidth B at each time-slot, and different links are assigned with orthogonal frequency resource. We then denote the amount of bandwidth allocated to the link for delivering content c with quality f from node m to vehicle n as $b_{m,n}^{(c,f)}(t)$ and assume the transmit power of an established link of node m is P_m. It follows that when $n \neq m$ the link rate $R_{m,n}^{(c,f)}(t)$ is expressed as

$$R_{m,n}^{(c,f)}(t) = b_{m,n}^{(c,f)}(t) \cdot \log_2 \left(1 + \frac{P_m |h_{m,n}(t)|^2}{N_0 b_{m,n}^{(c,f)}(t)}\right), \quad (1)$$

where N_0 is the noise power density and $h_{m,n}$ is the channel coefficient between vehicle n and node m. By (1), when $n \neq m$, the transmission delay can then be expressed as

$$d_{\mathrm{t},m,n}^{(c,f)}(t) = \frac{\left(\sum_{f'} y_{m,n}^{(c,f,f')}(t)\right) \cdot s^{(f)}}{R_{m,n}^{(c,f)}(t)}, \quad (2)$$

where $s^{(f)}$ is the segment size (in terms of bits) for the content with quality f. It should be noticed that when $m = n$, the request can be satisfied by using the content in the own cache of vehicle n. As a result, in this case, the transmission delay is 0 and the corresponding transmission power is also 0.

C. Computing Model

Since a content might be transcoded from a format to another, we need computing model for such transcoding. We assume that the transcoding can be either upgrading the low-quality content to higher quality via the so-called super-resolution technique for higher utility or downgrading the high-quality content to lower quality via the low-resolution technique, e.g., video compressing, for reducing the communication requirement [15].

Then, denoting the computing power allocated for the transcoding due to delivering content c with quality f from node m to vehicle n as $l_{m,n}^{(c,f)}(t)$, it follows that the computing delay for the corresponding transcoding can be expressed as

$$d_{\mathrm{c},m,n}^{(c,f)}(t) = \frac{\sum_{f'} y_{m,n}^{(c,f,f')}(t) \cdot \mathbf{F}(f,f') \cdot s^{(f')}(t)}{l_{m,n}^{(c,f)}(t)}, \quad (3)$$

where $\mathbf{F}(f,f')$ is the required amount of CPU cycles for transcoding a bit from quality f' to f. Note that since no transcoding is needed when $f = f'$, we have $\mathrm{F}(f,f') = 0$ in this case. We assume the computing power is limited. As a result, we let $l_{CPU}^{\max}$ be the maximum computing power that can be consumed by a node.

D. Caching Model

We assume there are in total C different content in the library and assume the number of different qualities of a content is F. We denote $Q^{(f)}$ as the size for content c with quality f. We denote the caching capacity of node m as Z_m and assume that Z_m is limited so that it is impossible for a node to cache all contents with all different qualities. Therefore, nodes should cooperate in caching to gain maximal benefits. We denote $x_m^{(c,f)}(t) \in \{0,1\}$ as the caching decision of node m for content c with quality f, where $x_m^{(c,f)}(t) = 1$ indicates that content c with quality f is cached in node m; otherwise, it is not cached in node m. We denote $r_n^c \in \{0,1\}$ as the indicator in which $r_n^c = 1$ indicates that content c is requested by vehicle n; otherwise $r_n^c = 0$.

We assume either the cached content can be updated at each time-slot or the cached content is fixed and is determined at $t = 0$. Although the assumption for the former case is less practical, the design and results under such assumption can serve as an useful reference for the study in the future. The study of the practical cache replacement [17] is considered as a future direction. On the other hand, the latter assumption is relatively practical as caching is commonly conducted in off-peak hours, and is invariant during peak hours.

E. Computing and Communication Pipeline

We consider the time-slot structure in this paper, where the duration of a time-slot is τ. We then assume that both the computation and transmission of a content should be finished within the duration of a single time-slot. Thus, the overall latency between the request and reception of a content is at most 2τ. To facilitate the content delivery in consideration of both computing and communication delays, we adopt the computing and communication pipeline as be shown in Fig. 1, where the computing tasks for requests received at the beginning of time-slot t will be computed in time-slot t, and then delivered in time-slot $t+1$. As a result, the computation and communication do not need to wait for one another which prevents the waste of resource usage. It should be noticed that if there are several requests received in a time-slot, the requests will be processed simultaneously. Note that although the resource allocation, computing and caching decisions are made at the time that the requests are received, since τ is generally small, we can assume that the channel is slow-varying, so the decision made at time-slot t is still effective even though the delivery happens in time-slot $t+1$.

Since the previously described architecture and models are dynamic and decisions are made for every time-slot, we can then focus on the optimization of each time-slot. As a result, we will in the remainder of this paper drop the time index from the notations for brevity.

III. Proposed Quality-Aware 3C Optimization Problem Formulation

In this section, we introduce our problem formulation for the joint 3C design. Our goal is to maximize the total utility of the vehicular network by optimizing bandwidth allocation, computing power allocation, transcoding decision, and caching decision under different system conditions. We will first introduce the fundamental problem formulation of this paper where the full-duplex transmission is assumed with multiple associations, i.e., a vehicle transmits and receives the signals at the same time, and can receive content from more than a single node. Then, problems for different system conditions can be obtained by modifying this fundamental problem.

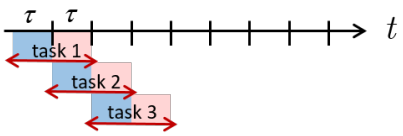

Fig. 1. Computing and communication pipeline.

Assume that the full-duplex transmissions can be used by the network and that a node can be associated to multiple vehicles, the utility maximization problem is given as

$$\max_{\mathbf{X},\mathbf{Y},\mathbf{B},\mathbf{L}} \quad \sum_m \sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \cdot u^{(f)} \tag{4a}$$

$$\text{s.t.} \quad 0 \le \sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \cdot P_m \le P_{\text{ind},m}, \forall m, \tag{4b}$$

$$\sum_m \sum_n \sum_c \sum_f b_{m,n}^{(c,f)} \le B, \tag{4c}$$

$$0 \le b_{m,n}^{(c,f)} \le \sum_{f'} y_{m,n}^{(c,f,f')} \cdot B, \forall f, c, n, m, \tag{4d}$$

$$\sum_n \sum_c \sum_f l_{m,n}^{(c,f)} \le l_{CPU}^{\max}, \forall m, \tag{4e}$$

$$0 \le l_{m,n}^{(c,f)} \le \sum_{f'} y_{m,n}^{(c,f,f')} \cdot l_{CPU}^{\max}, \forall f, c, n, m, \tag{4f}$$

$$\sum_c \sum_f x_m^{(c,f)} \cdot Q^{(f)} \le Z_m, \forall m, \tag{4g}$$

$$\sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \le r_n^c, \forall c, n, m, \tag{4h}$$

$$y_{m,n}^{(c,f,f')} \le x_m^{(c,f')}, \forall f, f', c, n, m, \tag{4i}$$

$$d_{\text{t},m,n}^{(c,f)} \le \tau, \forall f, c, n, m, \tag{4j}$$

$$d_{\text{c},m,n}^{(c,f)} \le \tau, \forall f, c, n, m, \tag{4k}$$

$$x_m^{(c,f)} \in \{0,1\}, \forall f, c, m, \tag{4l}$$

$$y_{m,n}^{(c,f,f')} \in \{0,1\}, \forall f, f', c, n, m, \tag{4m}$$

where $\mathbf{X}$ is the caching decision, $\mathbf{Y}$ is the transcoding decision, $\mathbf{B}$ is the bandwidth allocation, $\mathbf{L}$ is the computing power allocation, $P_{\text{ind},m}$ is the total transmit power constraint for node m and $u^{(f)}$ is the utility with quality f. In this problem, our goal is to maximize the network utility given in (4a) in terms of total bits received by vehicles in the time-slot. Constraint (4b) is to limit the total transmit power. Constraints (4c) and (4d) respectively give the total bandwidth constraint and indicate that a bandwidth can be allocated only if there is a transcoded content for delivery. Similarly, constraints (4e) and (4f) respectively describe the total computing power constraint

and that computing power is allocated only if there is a content needs to be transcoded. Constraint (4g) indicates the caching capacity is limited so that we cannot cache everything in a node. Constraint (4h) indicates that there is no association between vehicle n and node m for content c if there is no request for content c from vehicle n. Constraint (4i) mentions that node m can transcode content c from quality f' to quality f for satisfying the request of vehicle n for content c only if node m caches content c with quality f'. Finally, (4j) and (4k) are the communication and computing delay constraints mentioned in Sec. II-E.[2]

We note that if we consider the scenarios that the caching results are pre-determined at time-slot $t = 0$, and then focus on optimizing the computing and communications in consideration of the pre-determined caching results, this can be achieved simply by considering (4) with that the caching indicator $\mathbf{X}$ is treated as a given parameter of the problem. Such problem is called quality-aware 2C design problem in this paper. We note that the problems formulated in this section are mixed integer convex problems. Although there exist some optimization toolboxes, such as CVX, that can handle this problem with small-scale, due to the high complexity (in terms of dimension) of our problem, those standard solvers generally fail in our case. Therefore, in next section, we propose an effective design approach that can solve the formulated problem much more efficiently than the standard solvers.

IV. Proposed Quality-Aware 3C Design Approach

In this section, we present the proposed approach that can solve (4) efficiently. We first solve the subproblem obtained by fixing the caching policy $\mathbf{X}$ in (4). Then, based on the solution approach of the subproblem, we can update $\mathbf{X}$ via using a proposed heuristic approach with low complexity.

We first present the solution approach of the subproblem obtained by fixing $\mathbf{X}$. Specifically, we observe that when $\mathbf{X}$ is given, we can immediately obtain a reduced feasible set $\mathcal{Y}$ of $y_{m,n}^{(c,f,f')}, \forall m, n, c, f, f'$ according to the constraint in (4i). Then, we denote $b_{\text{min},m,n}^{(c,f,f')}$ and $l_{\text{min},m,n}^{(c,f,f')}$ as the minimum amount of bandwidth and computing power, respectively, if we want to deliver content c with quality f transcoded from quality f' in node m to vehicle n, where $b_{\text{min},m,n}^{(c,f,f')}$ and $l_{\text{min},m,n}^{(c,f,f')}$ are obtained by solving $\tau = \frac{s^{(f)}}{b_{\text{min},m,n}^{(c,f,f')} \log_2\left(1+\frac{P_m |h_{m,n}(t)|^2}{N_0 b_{\text{min},m,n}^{(c,f,f')}}\right)}$ and $\tau = \frac{\mathbf{F}(f,f')s^{(f')}}{l_{\text{min},m,n}^{(c,f,f')}}$, respectively. Note that solving these two equalities is easy as they are convex. It follows that with the obtained $b_{\text{min},m,n}^{(c,f,f')}$ and $l_{\text{min},m,n}^{(c,f,f')}$ for all m, n, c, f, f', the subproblem can be transformed into an integer problem given as follows:

$$\max_{\mathbf{Y}} \quad \sum_m \sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \cdot u^{(f)} \tag{5a}$$

$$\text{s.t.} \quad 0 \le \sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \cdot P_m \le P_{\text{ind},m}, \forall m, \tag{5b}$$

$$\sum_m \sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} b_{\text{min},m,n}^{(c,f,f')} \le B, \tag{5c}$$

$$\sum_n \sum_c \sum_f \sum_{f'} y_{m,n}^{(c,f,f')} l_{\text{min},m,n}^{(c,f,f')} \le l_{CPU}^{\max}, \forall m, \tag{5d}$$

$$\sum_f \sum_{f'} y_{m,n}^{(c,f,f')} \le r_n^c, \forall c, n, m, \tag{5e}$$

$$y_{m,n}^{(c,f,f')} \in \{0,1\}, \quad \text{if } y_{m,n}^{(c,f,f')} \in \mathcal{Y}, \tag{5f}$$

$$y_{m,n}^{(c,f,f')} = 0, \quad \text{if } y_{m,n}^{(c,f,f')} \notin \mathcal{Y}. \tag{5g}$$

The intuition of (5) is that if we know the cost for delivering the content c with quality f from node m to vehicle n, then the remaining problem is to find the best approach we can have such that the utility is maximized. Although problem (5) is a convex integer problem and can be solved by some standard solvers, we further propose an effective solution with submodularity and knapsack interpretation. To do this, we first reformulate problem (5) into the form of monotone the submodular maximization with multidimensional knapsack constraints as follows [18]:

$$\max_{\mathbf{Y}} \quad f_{\text{func}} = \sum_j y_j u_j \tag{6a}$$

$$\text{s.t.} \quad \sum_j y_j w_{ij} \le W_i, \forall i, \tag{6b}$$

$$y_j \in \{0,1\}, \quad \text{if } y_j \in \mathcal{Y}, \tag{6c}$$

$$y_j = 0, \quad \text{if } y_j \notin \mathcal{Y}, \tag{6d}$$

where the constraints from (5b) to (5e) in (5) are combined into (6b); y_j represents the transcoding decision of item j, w_{ij} is the weight value for the item j in knapsack constraint i, and W_i is the weight constraint of the i-th knapsack constraint. It should be noticed that there are in total $(2m+1+m\times n\times c)$ knapsack constraints and $(m \times n \times c \times f \times f')$ items. The solution approach of (6) then generally follows the approach in [18], which is briefly described as follows. Starting with an empty set $S = \varnothing$, for each step, we compute the marginal gain of adding item $j, \forall j$ to S. Then, we greedily choose the best item to add in each step until adding any of the remaining items would violate the constraints.

We note that since we can compute $b_{\text{min},m,n}^{(c,f,f')}$ and $l_{\text{min},m,n}^{(c,f,f')}$ for all m, n, c, f, f' in advance of solving (6), the solution of (6) along with them give us a design of $\mathbf{Y}$, $\mathbf{B}$, and $\mathbf{L}$ of (4) when $\mathbf{X}$ is given. Furthermore, such solution is indeed the solution for the proposed quality-aware 2C design in this paper. With this, the remaining is to find an effective $\mathbf{X}$ such that an effective solution of (4) is obtained.

To do this, we propose a heuristic approach that finds the effective $\mathbf{X}$ by using the preferences of the vehicles and the network. The proposed approach is as following. Suppose we choose to cache contents only with quality f. Then, since we are given the requests of vehicles, i.e., $r_n^c, \forall n, c$, we let RSU gradually cache the content starting from the content

[2]We can modify (4) to fit different system conditions, namely, the full-duplex and/or half-duplex transmissions with single/multiple associations, and the modified problems can still be solved by our proposed solution approach in this paper. However, the details are omitted due to page limitation.

that is requested the most times by the users until the cache space is used up, namely, we let RSU cache those contents that are the mostly requested by vehicles. Then, for a vehicle n, we let the vehicle first caches the content according to whether the content has been requested or not. If yes, the content should be cached. However, depending on whether the vehicle has sufficient cache space or not, there could be two possible situations. First, if the vehicle can cache all the content that are relevant to its requests, it first cache all the relevant content, and then uses the remaining cache space (if any) to conduct the caching starting from the content that has been least cached in the network. On the other hand, if the vehicle cannot cache all the content that are relevant, then it would conduct the caching starting from the relevant contents that have been least cached in the network until the cache space is used up.

With the above procedure, we can obtain a solution of $\mathbf{X}$ for a given f. Then, with the obtained $\mathbf{X}$, we solve (6) to obtain the corresponding $\mathbf{Y}$, $\mathbf{B}$, and $\mathbf{L}$. Finally, we note that we still need to decide what quality to cache. This is done by trying all different possible qualities, and then select the one that gives the maximal utility. Note that since we assume all nodes in the network cache the content with the same quality, exhaustive search for the best quality f is not difficult. Also, we note that although this heuristic approach can work efficiently and can provide good improvement when compared to the reference schemes (see our simulation results in next section), this approach is clearly a suboptimal approach, and thus finding a more advanced solution approach is considered as one of our future works.

V. Computer Simulations

A. Simulation Setup

We consider the following setup for simulations. We consider a RSU is located at the center of the roadside of the service area with size 200×6 m^2, which corresponds to an ordinary straight road. We assume the vehicles are distributed uniformly at random in the service area and the numbers of buses N_b and cars N_c are uniformly selected from the set $\{4, 5, 6\}$ for each time-slot, indicating that the vehicles can leave and enter the cluster. We assume that the number of requests in each bus is a Poisson random variable with mean value equal to 15; the number of requests in each car is a truncated Poisson random variable with mean and maximal values equal to 3 and 9, respectively. We assume different vehicles have different distributions to request the content, and the distributions are generated using the generator proposed in [19]. We assume the number of videos in the library is $C = 200$ and assume there are 4 different video qualities for delivery. Therefore, we let the corresponding $Q^{(f)}$, $u^{(f)}$ and $s^{(f)}$ of different qualities to be set as $20, 40, 80, 160$ MB, $50, 100, 200, 400$ B and $50, 100, 200, 400$ B respectively. We set $\mathbf{F}(f, f')$ to $0, 1.5, 2, 2.5$ CPU cycles per bits when $|f - f'|$ is equal to $0, 1, 2, 3$, respectively.

We let the maximal total transmit powers of the RSU and the vehicles be 30 dBm and 20 dBm, respectively. We let the transmit power of each link of the RSU and the vehicles be 20 dBm 10 dBm, respectively. We consider different caching capacities for RSU and vehicles and consider three different cases, denoted as large, medium, and small cases, respectively. The caching capacities of the RSU, the bus, and the car in the large case are 3 GB, 1 GB, and 0.5 GB, respectively; in the medium case are 2 GB, 0.65 GB, and 0.35 GB, respectively; and in the small case are 1.5 GB, 0.5 GB, and 0.25 GB, respectively. We consider $B = 100$ MHz; $l_{CPU}^{\max} = 200$ GHz; $\tau = 10^{-3}$ sec; and $N_0 = -174$ dBm/Hz.

We consider the large-scale fading model, where only the pathloss and shadowing are considered, and assume the small-scaling fading is eliminated by some diversity approach. The pathloss model used in the simulations is given as $PL = P_{L_0} + 10\gamma \log_{10}(\frac{d}{d_0}) + X_{SF}$ where $d \geq 1$ is the distance between the transmitter and receiver; P_{L_0} is the pathloss calculated using the free-space pathloss model with reference distance $d_0 = 1$ m and $f = 2.8$ GHz; $\gamma = 3.5$ is the pathloss factor; and $X_{SF} \sim \mathcal{N}(0, 6^2)$ is the lognormal shadowing with 0 dB and 6 dB for mean and standard deviation, respectively.

B. Simulation Results

In this subsection, we evaluate the proposed design and conduct comparisons with some reference schemes. We consider the performance evaluations under two scenarios, where the caching results $\mathbf{X}$ can be dynamically updated or that $\mathbf{X}$ are pre-determined. When considering the caches can be updated dynamically, we compare our proposed design with the reference schemes described in the following. The first two reference schemes, labelled as "cache low/high", randomly select the content to cache and consider caching content only with the lowest/highest quality. The next two reference schemes, labelled as "cache low/high by preference", follow the similar approach as the proposed caching policy in Sec. IV; the only differences lie in that here vehicles selfishly cache only content with the lowest/highest quality according to the vehicles' requests, i.e., there is no selection of quality and no collaboration among the nodes as compared to the proposed caching policy. The final reference scheme, labelled as "pure random cache", randomly chooses the content and quality to cache. It should be noted that these reference schemes only determine the caching approach, and the computing and communication design of them are still determined by using the proposed quality-aware 2C design in Sec. IV. This implies that the comparisons with them are from the caching aspect.

The results are shown in Fig. 2, where the full-duplex transmissions are implemented with multiple association. We observe that the proposed design can outperform all the reference schemes. Besides, we see that the "cache low by preference" scheme can perform closely to the proposed design. This is because even though all the cached contents are with lowest quality, our proposed quality-aware 2C design can still appropriately manage the transcoding, computing, and communication such that the overall utility is good. On the other hand, when caching only high-quality content, e.g., the "cache high by preference" scheme, since caching content with

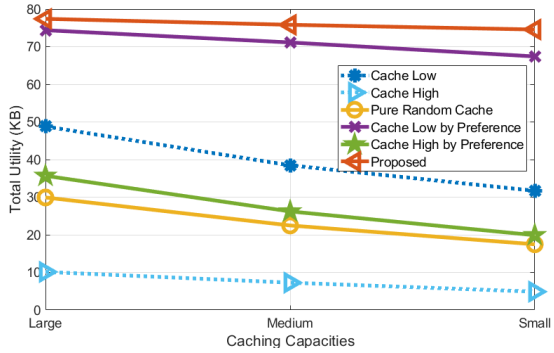

Fig. 2. Evaluation of the proposed quality-aware 3C optimization.

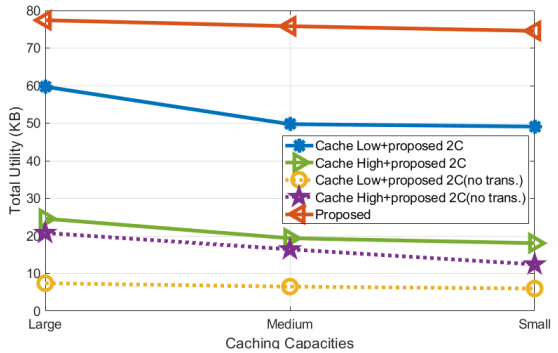

Fig. 3. Evaluation of the proposed quality-aware 2C optimization.

high quality might occupy too much cache space, it ultimately leads to worse performance as the transcoding benefit might not be effectively leveraged in this case.

In Fig. 3, we consider the case that $\mathbf{X}$ are pre-determined and evaluate the proposed quality-aware 2C design under multiple association. Specifically, we consider the following schemes for comparisons. The first type of schemes use the proposed quality-aware 2C design with that the cached contents of a vehicle (or RSU) are determined by the caching the contents with lowest/highest quality according to the individual/global preference of the vehicle (or RSU), i.e., each vehicle (or RSU) caches its most preferred contents until the cache space is used up. This type of schemes is labelled as "cache low/high + proposed 2C". The second type of schemes consider adopting the same caching approach as the first-type schemes without transcoding. In other words, the second-type schemes do not consider any transcoding, but directly deliver the cached content. Finally, we use the proposed 3C design as the upper bound reference scheme.

From Fig. 3, we observe that the schemes with the proposed quality-aware 2C design can significantly outperform the schemes without transcoding. This validates the efficacy of our design and shows the benefit of quality-aware computing. We also observe that the proposed quality-aware 3C design outperforms the proposed quality-aware 2C design significantly. This indicates that if we can refresh the caches from time-to-time, we can obtain significant benefits as compared to the case that the caches are fixed. This motivates the design of cache replacement scheme that is one of our future directions.

VI. Conclusion

This paper investigated the quality-aware 3C optimization for video delivery in vehicular networks. The investigation included both the joint 3C optimization and the 2C optimization with pre-determined caching. Effective solution approaches were proposed, and simulations were conducted to evaluate the proposed designs. Results showed that our proposed approaches can significantly improve the network performance.

References

[1] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang and Y. Zhou, "Heterogeneous Vehicular Networking: A Survey on Architecture, Challenges, and Solutions," *IEEE Commun. Surveys Tut.*, vol. 17, no. 4, pp. 2377-2396, 2015.

[2] Y. Mao, C. You, J. Zhang, K. Huang and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2322-2358, 2017.

[3] L. Li, G. Zhao and R. S. Blum, "A Survey of Caching Techniques in Cellular Networks: Research Issues and Challenges in Content Placement and Delivery Strategies," *IEEE Commun. Surveys Tut.*, vol. 20, no. 3, pp. 1710-1732, 2018.

[4] M. Jiau, S. Huang, J. Hwang and A. V. Vasilakos, "Multimedia Services in Cloud-Based Vehicular Networks, "*IEEE Intell. Transp. Syst. Mag.*, vol. 7, no. 3, pp. 62-79, 2015.

[5] M.-C. Lee and A. F. Molisch, "Asymptotic Delay–Outage Analysis for Noise-Limited Wireless Networks with Caching, Computing, and Communications," *IEEE 2022 ICC*, May 2022.

[6] M. Choi, J. Kim and J. Moon, "Wireless Video Caching and Dynamic Streaming Under Differentiated Quality Requirements," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1245-1257, June 2018.

[7] M. Tang, L. Gao, and J. Huang. "Communication, Computation, and Caching Resource Sharing for the Internet of Things," *IEEE Commun. Mag.,* vol. 58, no. 4, pp. 75-80, Apr. 2020.

[8] Z. Ning et al., "Intelligent Edge Computing in Internet of Vehicles: A Joint Computation Offloading and Caching Solution," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 4, Apr. 2021.

[9] L. T. Tan, R. Q. Hu and L. Hanzo, "Twin-Timescale Artificial Intelligence Aided Mobility-Aware Edge Caching and Computing in Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol.68, no. 4, pp. 3086-3099, 2019.

[10] L. T. Tan and R. Q. Hu, "Mobility-Aware Edge Caching and Computing in Vehicle Networks: A Deep Reinforcement Learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10190-10203, 2018.

[11] G. Qiao, S. Leng, S. Maharjan, Y. Zhang and N. Ansari, "Deep Reinforcement Learning for Cooperative Content Caching in Vehicular Edge Computing and Networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 247-257, Jan. 2020.

[12] C. Tang, C. Zhu, H. Wu, Q. Li, and J. J. P. C. Rodrigues, "Towards Response Time Minimization Considering Energy Consumption in Caching Assisted Vehicular Edge Computing," *IEEE Internet of Things J.*, 2021.

[13] Z. Lyu and Y. Wang, "Learning-Based Demand-Aware Communication Computing and Caching in Vehicular Networks," *IEEE WCNCW*, pp. 1-6, 2019.

[14] Y. He, N. Zhao and H. Yin, "Integrated Networking, Caching, and Computing for Connected Vehicles: A Deep Reinforcement Learning Approach," *IEEE Trans. Veh. Technol.*,vol. 67, no. 1, pp. 44-55, Jan. 2018.

[15] S. M. A. Kazmi et al., "Infotainment Enabled Smart Cars: A Joint Communication, Caching, and Computation Approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8408-8420, Sept. 2019.

[16] K. Hayat, "Multimedia Super-resolution via Deep Learning: A Survey," *Dig. Sig. Process.*, vol. 81, pp. 198-217, 2018.

[17] M.-C. Lee, H. Feng, and A. F. Molisch, "Dynamic Caching Content Replacement in Base Station Assisted Wireless D2D Caching Networks," *IEEE Access*, vol. 8, pp. 33909-33925, Feb. 2020.

[18] Y. Wang, Y. Liy, and K.-L. Tan, "Efficient Streaming Algorithms for Submodular Maximization with Multi-Knapsack Constraints," arXiv:1706.04764v1.

[19] M.-C. Lee, A. F. Molisch, N. Sastry, and A. Raman, "Individual Preference Probability Modeling and Parameterization for Video Content in Wireless Caching Networks," *IEEE Trans. Netw.*, 2019.