This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

Latency Control in Edge Information Cache and Dissemination for Unmanned Mobile Machines

Shao-Yu Lien, Shao-Chou Hung, and Hsiang Hsu

***Abstract*—Unmanned technologies facilitating human activities have been regarded as the most promising innovation to empower fully automatic and intelligent ecosystems. Targeting at extending the processing capabilities of humans, unmanned mobile machines (UMMs) are designated to devise the optimum action at varying operating conditions, which relies on prompt information provisioning through existing cellular infrastructures, and renders latency control to information acquisition an inevitable challenge. For this purpose, caching information at network edges has been a remedy for substantial latency reduction, which however ignores practical cell deployment inducing imbalanced wireless services to each UMM in the hot-spot and rural areas. In this paper, through formulating the Lyapunov function, an algorithm optimizing the utilization of fronthaul resources while stabilizing each UMM's queue is proposed for edge information cache and dissemination in the hot-spot areas. Furthermore, through formulating the cost measurement as the Cobb-Douglas production function, the optimal beginning time of cache is also derived for UMMs in the rural areas. With the provided analytical foundations and simulation studies, the effectiveness of our latency control scheme is fully demonstrated.**

***Index Terms*—Latency control, UMMs, edge information cache, Lyapunov function, epidemic spreading.**

I. Introduction

UNMANNED mobile machines (UMMs) embracing robots, drones, vehicles, etc. are projected to fundamentally shift the present paradigms in industries, commerce, agriculture, and transportation [1], [2]. These UMMs cruising down the urban areas (such as indoor shopping malls or outdoor streets) or rural areas (mountain districts, or locations unreachable by humans) are designated to optimally process the assigned tasks [3] under varying operating conditions, which relies on prompt and ubiquitous information provisioning through wireless communication/network technologies. This demand consequently renders the exploitation on existing cellular infrastructure a tractable solution [4]. However, the existing cell planning strategies of operators' utilitarian interests target at optimizing the user experience of human-carried devices. As a result, base stations (BSs) may not be deployed uniformly over a geographic area [5], but operators may massively deploy BSs at the areas with a high population (hot-spot areas) and sparsely deploy BSs at the areas with a low population (rural areas), as illustrated in Fig. 1. Since operating on unreachable regions by humans is one of the major use cases of UMMs, the existing cell planning strategies may not fully sustain UMMs without additional designs.

Both in the hot-spot and rural areas, low latency information acquisition is of crucial importance for UMMs [6] to process timing-sensitive tasks. In the hot-spot areas, an UMM is able to obtain all desired information from a cloud server through connecting to a BS(s), as illustrated in Fig. 1. However, acquiring information from a cloud server involves data traversal through a considerable number of routers, switches and gateways at backhaul links (i.e., links between BSs and a cloud server), to lead to an unaffordable latency performance [7]. Recently, this challenge motivates the contrivances to cache frequently accessed information [8] at network edges such as BSs. In this case, when an UMM requests information from a cloud server through connecting to a BS, a BS that has cached this information is able to directly offer requested information to an UMM. As a result, backhaul links can be avoided to significantly improve the latency performance. However, due to limited cache space, each BS may cache a part of information which is distinct from that in other BSs. When an UMM is eager for particular information, although requesting this information from more BSs may increase the probability to successfully obtain desired information, more resources are also consumed at fronthaul links (i.e., link between a BS and a UMM) through using this scheme. On the contrary, the less BSs an UMM acquires information from, the lower probability an UMM can successfully obtain desired information. If none of requested BSs cache desired information, then this information should be provided from the cloud server through backhaul links. Consequently, there is a tradeoff between the latency performance and the amount of resources utilized on the fronthaul links.

In the rural areas, it is likely that an UMM cannot connect to any BS, as illustrated in Fig. 1. To facilitate prompt information dissemination, information cache can be extended from BSs to each UMM. When wireless services from a BS are available, an UMM not only acquires desired information but also further caches this information. Subsequently, when this UMM moves outward coverage of BSs, cached information can be disseminated to other UMMs in physical proximity [9]. Such information cache and dissemination among UMMs through stochastic mobility and contact form a new spreading behavior and

S.-Y. Lien is with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chia-Yi 62102, Taiwan, email: sylien@ccu.edu.tw.

S.-C. Hung is with the MediaTek, Hsinchu 30078, Taiwan, e-mail: d02942008@ntu.edu.tw.

H. Hsu is with the School of Engineering and Applied Science (SEAS), Harvard University, Cambridge, MA 02138, USA, e-mail: hsianghsu@g.harvard.edu.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

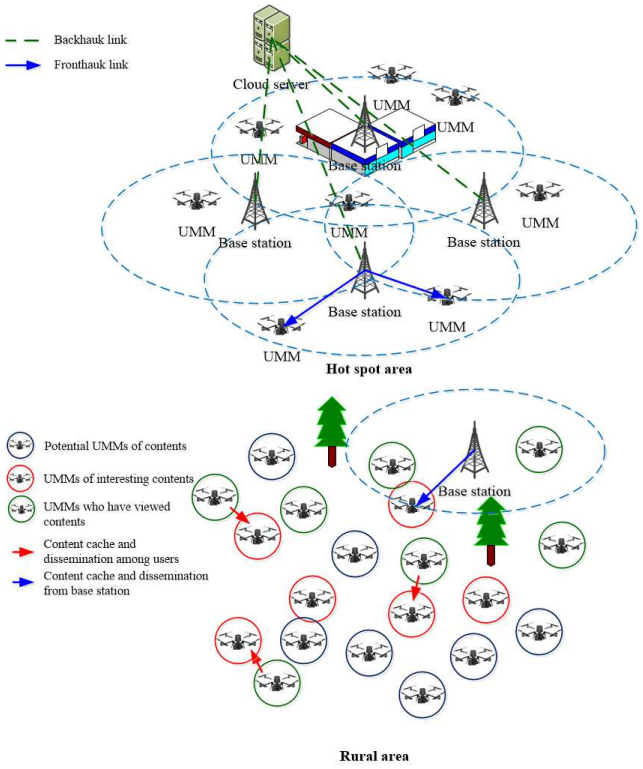

Fig. 1. Operators may massively deploy BSs at the areas with a high population (hot-spot areas) and sparsely deploy BSs at the areas with a low population (rural areas).

dynamics [10], which can be analogous to the well known disease epidemics [11]. To prolong the battery life for each UMM, energy efficiency is of pivotal significance in designs from all the aspects for an UMM. This concern thus conduces an adequate beginning time for each UMM to perform information cache a critical issue.

To address these critical issues, in this paper, we should develop the optimum latency control of information cache and dissemination for UMMs. The contributions of this paper thus include the followings.

- We derive analytical foundations to strike the trade-off between the latency performance and resource utilization at fronthaul links in the hot-spot areas. With the facilitation of *Lyapunov* function [12], the devised design is analytically proven to effectively maximize the fronthaul resource utilization with a bounded latency in information acquisition.
- Furthermore, through formulating the cost of cache as the Cobb-Douglas production function, we analytically derive the optimum control to the beginning time of cache for epidemic information dissemination among UMMs in the rural areas.

Through the analytical and simulation studies, the derived optimum cache time is shown to achieve efficacious information dissemination within a bounded latency.

II. Related Works

Cloud storage/computing to collect and process data around the world at a unified cloud server has been demonstrated to empower ubiquitous knowledge acquisition and global information analysis/management [13]. With the aid of existing cellular infrastructure, this paradigm has been further extended to mobile users, devices, and UMMs to access both location-based and non-location-based information in this decade. However, recent research has revealed severe inefficiency of cloud storage/computing due to scalability [6], [14], and such inefficiency includes 1) growing latency to access information stored/processed at databases whose physical locations could be any place on this planet, 2) growing burdens on backhaul links of mobile networks to forward information between BSs and the cloud server, 3) decreasing spectrum utilization at fronthaul links of mobile networks due to heavy burdens to forward information between BSs and mobile users. These thorny issues consequently drive the development of edge content cache and dissemination [8], [9], [15]–[21].

Caching frequently desired information at network edges has been regarded as a promising innovation to tackle the scalability issue [8], [22] so as to significantly alleviate information acquisition latency and traffic burdens at backhaul links [8], [17], [19]. In the meantime, network edges may further disseminate cached/obtained information/knowledge to other network edges [9], [15], [20]. As information dissemination only occurs among network edges in physical proximity, spectrum could be fully reused in the spatial domain to considerably enhance the spectrum utilization at fronthaul links [16], [18]. Further, recent research also reveals that energy efficiency of mobile networks can be significantly improved with the facilitation of edge content cache and dissemination [21].

Despite considerable discussions on the technical merits in terms of latency, traffic burdens at backhaul links and the spectrum utilization at fronthaul links, an optimum cache design tailored for UMMs utilizing existing cellular infrastructures still remains open. To practice UMMs, we therefore should address this urgent and open challenge.

III. System Model

In the literature, extensive research studies have reveled that a device may have distinct levels of interests in accessing different contents [23]. Owing this fact, denoting the set of overall contents as $\mathcal{K}$ and denoting the number of contents in the set as $\|\mathcal{K}\|$, the Zipt distribution has been shown a general model to capture the popularities of each content in $\mathcal{K}$ [20]. That is, through ranking the popularities of all the contents from the most popular to the least popular as $k = 1, \cdots, \|\mathcal{K}\|$, the probability that an UMM requests for the kth most popular content follows the Zipf distribution $p_k = \frac{1/k^{\xi}}{H_{\|\mathcal{K}\|}}$, where $H_{\|\mathcal{K}\|} = \sum_{k=1}^{\|\mathcal{K}\|} \frac{1}{k^{\xi}}$ and $\xi \in [0, \infty)$ is a skew factor. The definitions of all notations used in this paper are summarized in Table I.

A. Hot-Spot Areas

In the hot-spot areas with massive BS deployment as shown in Fig. 1, denote $\mathcal{A}_t^f$ as the set of total BSs available to be connected by an UMM at time t, which are indexed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

TABLE I
NOTATIONS USED IN THIS PAPER

Notations	Definitions
$\|\cdot\|$	Number of elements in the set $\cdot$
$\mathcal{K}$	Set of overall contents
p_k	Probability of occurrence of the kth popular content
ξ, α	Skew factor of Zipt distribution
$H_{\|\mathcal{K}\|}$	$H_{\|\mathcal{K}\|} = \sum_{k=1}^{\|\mathcal{K}\|} \frac{1}{k^{\xi}}$
$\mathcal{A}_t^f$	Set of total BSs available to be connected by an UMM at time t
A_t^f	Set of BSs connected by an UMM
Z_l	Maximum amount of contents that can be stored in the local cache of a BS
Z_g	Maximum amount of contents that can be stored in the global cache of a BS
$\mathcal{K}_n^l$	Available contents in the local cache
$\mathcal{K}_n^g$	Available contents in the global cache
$\mathcal{K}_t^b$	Set of available contents stored at cloud server at time t
$\mathcal{C}$	Set of contents that an UMM is able to access to
$U_k(t)$	Length of a task queue (requiring the kth most popular content) at time t in an UMM
$a_k(t)$	Number of arriving tasks requiring the kth most popular content
$u_k(t)$	Number of proceed tasks requiring the kth most popular content at time t
λ_k	Expected value of $a_k(t)$
λ_0	Overall task arrival rate
ϖ_l	Processing rates if contents are obtained from BSs
ϖ_g	Processing rates if contents are obtained from cloud server
$l_{bs}(t)$, $a_{bs}(t)$	Numbers of BSs which leave and enter connectable region
$S(t)$	Number of potential viewers of contents
$R(t)$	Number of UMMs who have accessed the content and never request it again
$I(t)$	Contagious UMMs who want but still waiting for the content
δ	Distance of physical proximity
N	Total number of UMMs in the rural areas
L	Range of the rural area
T_C	Beginning time to perform cache
κ_2, κ_1	Processing rates with and without cache
$\tilde{S}(t)$	Normalized susceptible population at time t
$\tilde{R}(t)$	Normalized recovered population at time t
ν	Virality of the content
η	Average number of contacted UMMs per unit time
$\dot{S}(t)$, $\dot{R}(t)$, $\dot{I}(t)$	$\dot{S}(t) = dS(t)/dt$, $\dot{R}(t) = dR(t)/dt$, $\dot{I}(t) = dI(t)/dt$
κ	Socially cooperative sharing coefficient
$n_b(t)$	Amount of contents downloaded from cloud server at t
$\mathbf{1}_{bk}$	An indicator
$n_f(t)$	Number of connected BSs at time t
ϑ	Maximum amount of the contents permitted to download from cloud server
$L(t)$	Lyapunov function
V	Cost weight of utilizing fronthaul links (BSs)

by $n = 1, \cdots, \|\mathcal{A}_t^f\|$, where $\|\mathcal{A}_t^f\|$ is the number of BSs in $\mathcal{A}_t^f$. When an UMM wishes to acquire a content, this UMM connects to a set of BSs $A_t^f \in \mathcal{A}_t^f$ to request for the content. When a BS in A_t^f receives such a request at the fronthaul link, a BS may relay this request to a cloud server and the desired content can be provided through backhaul links. As aforementioned, data traverse through backhaul links may induce unacceptable latency, and an accurate model on such latency on backhaul links may not be analytically tractable. Each BS may perform content cache to avoid utilizing backhaul links. For this purpose, each BS contains two cache spaces; one is known as "local cache" storing location-based contents collected by the BS itself, and another one is known as "global cache" storing global wise contents downloaded from a cloud server. The maximum amounts of contents that can be stored in local and global cache are denoted by Z_l and Z_g, respectively. Therefore, the available contents in the local and global caches of the nth BS can be denoted by $\mathcal{K}_n^l \in \mathcal{K}$ ($\|\mathcal{K}_n^l\| < Z_l$) and $\mathcal{K}_n^g \in \mathcal{K}$ ($\|\mathcal{K}_n^g\| < Z_g$), respectively. Obviously, the set of contents that are cached and can be provided by the nth BS is $\mathcal{K}_n^l \cup \mathcal{K}_n^g$. If an UMM connects to more than one BS, then the set of available contents is extended to $\bigcup_{n \in A_t^f} \left(\mathcal{K}_n^l \cup \mathcal{K}_n^g\right)$. As a result, the more BSs that an UMM connects to, the higher probability that the desired content falls within $\bigcup_{n \in A_t^f} \left(\mathcal{K}_n^l \cup \mathcal{K}_n^g\right)$ as A_t^f is extended. However, if unfortunately the desired content does not fall within $\bigcup_{n \in A_t^f} \left(\mathcal{K}_n^l \cup \mathcal{K}_n^g\right)$, an UMM only relies on a single BS to acquire the content from the cloud server. Consequently, the set of contents that an UMM is able to access to can be generally expressed by

$$\mathcal{C} \triangleq \bigcup_{n \in A_t^f} \left(\mathcal{K}_n^l \cup \mathcal{K}_n^g\right) \bigcup \mathcal{K}_t^b, \tag{1}$$

where we denote $\mathcal{K}_t^b \in \mathcal{K}$ as the set of available contents stored at the cloud server at time t.

Let $U_k(t)$, $k \in \mathcal{K}$, denote the length of a task queue (requiring the kth most popular content) at time t in an UMM. A task in the queue can be processed and removed from the queue only if a content (of the kth most popular) can be obtained from the cloud server or from a BS. The dynamics of $U_k(t)$ can be captured by

$$U_k(t+1) = \left(U_k(t) - u_k(t)\right)^+ + a_k(t), \forall k \in \mathcal{K}, \tag{2}$$

where $a_k(t)$ is the number of arriving tasks requiring the kth most popular content, and $u_k(t)$ is the number of proceed tasks requiring the kth most popular content at time t. The arrival process of $a_k(t)$ depends on practical applications. For example, $a_k(t)$ can be *Gamma* distributed for Internet-of-Things (IoT) use cases, or follows *Poisson* distribution. Generally consider the expected value of $a_k(t)$ as λ_k, which is determined by the associated popularity of the content. That is, the fraction of the task arrival rate also follows the Zipf distribution with skew factor $\alpha \in [0, \infty)$. Therefore, λ_k can be expressed by $\lambda_k = \lambda_0 p_k$, where λ_0 is the overall task arrival rate with every associated popularity of desired contents. Likewise, we generally regard the expected value of $u_k(t)$ as a function of $\mathcal{C}$,

$$\mathbb{E}\left(u_k(t)\right) = f_k(\mathcal{C}) = \begin{cases} \varpi_l, k \in \bigcup\limits_{n \in A_t^f} \left(\mathcal{K}_n^l \cup \mathcal{K}_n^g\right) \\ \varpi_g, k \in \mathcal{K}_t^b \\ 0, \text{if } k \notin \mathcal{C}, \end{cases} \tag{3}$$

where ϖ_l and ϖ_g are the processing rates if the kth most popular contents are obtained from BSs and the cloud server, respectively. That is, $\mathbb{E}\left(u_k(t)\right)$ depends on whether an UMM is able to obtain a content of the popularity k from a BS or from the cloud server. If the contents cannot be obtained, then the processing rate degrades to zero.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

B. Rural Areas

In the rural or human-unreachable areas as shown in Fig. 1, BSs are generally sparsely deployed. In this case, information dissemination mainly relies on opportunistic mobility and contacts among UMMs. To analyze the time dynamics of information dissemination, the non-linear ordinary differential equations (ODE) of epidemic spreading to model the system [11]. Particularly, the susceptible-infected-recovered (SIR) model is adopted as the state evolution equations for the system [24]. To connect the analogue between the SIR model and the composition of UMMs, three compartments $S(t)$, $I(t)$, and $R(t)$ are denoted as the susceptible, infected, and recovered populations of UMMs at time t, as illustrated in Fig. 1. The susceptible UMMs $S(t)$ represent the potential viewers of the content; the infected UMMs $I(t)$ are contagious UMMs who want but still waiting for the content; the recovered UMMs $R(t)$ represent those who have accessed the content and never request it again. An infected UMM can attract a susceptible UMM to access the content when they are in physical proximity within a range of δ. Denote N as the total number of UMMs in the rural areas, *i.e.,* $S(t)+I(t)+R(t)=N$, $\forall t \geq 0$, and these UMMs move randomly in a $L \times L$ square area.

1) Information Cache at BSs: If all UMMs lack the capability to cache the contents, infected UMMs can only acquire the contents from the BSs with limited bandwidth at backhaul links. Although an UMM that has obtained the content from the BSs is able to further disseminate the content to other UMMs when they are within a region of δ, limited bandwidth of backhaul links could be a bottleneck for $u_k(t)$ provided from BSs. However, if contents can be cached at BSs, the processing rate of tasks can be enhanced due to the shift of traffic load from backhaul links to BSs [25]. In this case, the processing rate can be expressed by

$$u_k(t) = \begin{cases} \kappa_1, & t < T_C \\ \kappa_2, & t \geq T_C \end{cases} \tag{4}$$

T_C is the beginning time to cache the content at the BS, and $0 \leq \kappa_1 \leq \kappa_2 \leq 1$. κ_2 and κ_1 are the processing rates with and without cache, respectively. The dynamics of infected population is determined by the pairwise virality of the content, rate of contacts among UMMs and the fraction of susceptible UMMs to the epidemic content. The recovered population is controlled by $u_k(t)$ of the BS. Putting all together, the state equations can be formulated as

$$\begin{cases} \dot{I}(t) = \nu\eta\hat{S}(t)I(t) - u_k(t)I(t), \\ \dot{R}(t) = u_k(t)I(t), \\ \dot{S}(t) + \dot{I}(t) + \dot{R}(t) = 0, \end{cases} \tag{5}$$

where $\hat{S}(t) = S(t)/N$ is the normalized susceptible population at time t; $\nu \in \mathbb{R}$ is the virality of the content, which can be regarded as the penchant for susceptible UMMs to request the content upon contact with infected UMMs in the range δ; $\eta = \pi\delta^2/L^2$ is the average number of contacted UMMs per unit time. The last equation is implied from fixed total population N, where $\dot{S}(t) = dS(t)/dt$, $\dot{R}(t) = dR(t)/dt$, and $\dot{I}(t) = dI(t)/dt$.

2) Information Cache Both at BSs and UMMs: In the previous scheme, direct opportunistic exchange of contents is utilized for content dissemination. Nevertheless, this feature can be further enhanced when cache is extended to UMMs. That is, the recovered UMMs can cache the epidemically spread contents and share them to socially satisfy other UMMs. The population of recovered UMMs can thus be further increased through the cooperative sharing among UMMs. Consequently, the state equations can be formulated as

$$\begin{cases} \dot{I}(t) = \nu\eta\hat{S}(t)I(t) - u_k(t)I(t) - \phi(t)\eta\hat{R}(t)I(t), \\ \dot{R}(t) = u_k(t)I(t) + \phi(t)\eta\hat{R}(t)I(t), \\ \dot{S}(t) + \dot{I}(t) + \dot{R}(t) = 0, \end{cases} \tag{6}$$

where $\hat{R}(t) = R(t)/N$ is the normalized recovered population at time t,

$$\phi(t) = \begin{cases} 0, & t < T_C \\ \kappa, & t \geq T_C \end{cases} \tag{7}$$

and $\kappa \in [0,1]$ is the socially cooperative sharing coefficient, which can be interpreted as the pairwise willingness to share the epidemic content. Each recovered UMM starts to cache the content and provide auxiliary sharing of the content after time T_C. A direct observation from (6) shows that this scheme proliferates the recovered population and therefore alleviates the growth of infected population. Moreover, comparing (5) and (6), only caching contents at BSs is actually a degenerate case of performing cache both at BSs and UMMS when $\kappa = 0$, which means that the total service of the content comes from the BS.

IV. Latency Control to Information Cache and Dissemination in Hot-Spot Areas

A. Problem Formulation

From (3), we observe that the more BSs an UMM connects to, the larger the probability that an UMM is able to successfully acquire the desired content without utilizing the cloud server. However, owning the fact that the number of BSs (and thus the amount of resources at fronthaul links) is limited, it is infeasible to permit an UMM to connect to all the BSs; otherwise, every UMM may suffer from unacceptable latency at fronthaul links due to connection congestion. Therefore, each UMM should connect to as least number of BSs as possible. On the other hand, although an UMM can certainly obtain the desired content from the cloud server, it is infeasible to permit an UMM always to acquire the content from the cloud server, due to the intractably large latency induced by utilizing the backhaul link. Consequently, the latency control to information cache and dissemination can be formulated as the following optimization.

Definition 1. *Denoting $n_b(t)$ as the amount of contents*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

downloaded from the cloud server at t, $n_b(t)$ is defined by

$$n_b(t) \triangleq |\mathcal{K}_t^b| = \sum_{k\in\mathcal{K}} \mathbf{1}_{bk}, \tag{8}$$

where $\mathbf{1}_{bk}$ is an indicator, which equals to 1 if the content is downloaded from the cloud server and 0 if not, i.e.,

$$\mathbf{1}_{bk} = \begin{cases} 1, k \in \mathcal{K}_t^b \\ 0, k \notin \mathcal{K}_t^b. \end{cases} \tag{9}$$

Definition 2. *The time-averaged amount of contents downloaded from the cloud server is defined by*

$$\overline{n_b} \triangleq \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} n_b(t). \tag{10}$$

Optimization 1. *Denoting $n_f(t)$ as the number of connected BSs at time t (i.e., $n_f(t) \triangleq |A_t^f|$), and denote ϑ as the maximum amount of the contents permitted to download from the cloud server, the optimal control is given by*

$$\begin{aligned} \min_{A_t^f, n_b(t)} \quad & \overline{n_f} \triangleq \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} n_f(t) \\ \textit{s.t.} \quad & \textit{(i)}\ \overline{n_b} < \vartheta \\ & \textit{(ii)}\ \lim_{t\to\infty} U_k(t) < B, \forall B < \infty, \forall k \in \mathcal{K}, \end{aligned} \tag{11}$$

The objective of (11) minimizes the time-averaged number of connected BSs, subject to two constraints. The constraint (i) guarantees that the time-averaged amount of contents downloaded from the cloud server does not exceed ϑ, and (ii) further stabilizes the task queue in an UMM.

B. Proposed Optimum Control Scheme

Since (11) involves time dynamics both in the objective and constraints, conventional time-invariant optimization schemes may not be feasible to solve the problem. Fortunately, we may tackle this optimization from the inspiration of stochastic optimization [26]. To take the constraint (ii) into account, a virtual queue $X(t)$ can be introduced with the updating rule $X(t+1) = (X(t) - \vartheta)^+ + n_b(t)$. In this virtual queue, ϑ can be interpreted as the amount of items in the queue that have been served at time t, and $n_b(t)$ can be regarded as arrivals at time t of the queue. To make the virtual queue stable, the necessary and sufficient condition is to restrict the time-averaged amount of arrivals smaller than the amount of served items, i.e., $\overline{n_b} < \vartheta$. As a result, we can convert the constraint (ii) to a dynamic queue, like $U_k(t)$, and a solution solving (11) should both guarantee the stability of $U_k(t)$ and $X(t)$. With the updating rule, we may reformulate (11) to the following optimization.

Optimization 2. *At each time t, after observing the present $X(t)$ and $U_k(t)$, an UMM solves the following optimization to determine the amount of BSs to connect to, and the amount of the contents to be downloaded from the cloud server,*

$$\max_{A_t^f \in \mathcal{A}_t^f, \mathbf{1}_{bk}} \sum_{k\in\mathcal{K}} \left(U_k(t)u_k(t) - X(t)\mathbf{1}_{bk}\right) - \frac{V}{2} n_f(t), \tag{12}$$

where $V > 0$ weights the cost of utilizing BSs (fronthaul links).

Optimization 2 is motivated from the spirit that a queue with a longer length should be served first. In the following lemma and theorem, we show that (12) is analytically tractable, by assuming an extreme case of ϑ=0.

Definition 3. *$g(C)$ representing the increment of (12) by connecting the BSs with the content $C \in \mathcal{C}$ is defined by*

$$g(C) \triangleq \sum_{k\in C} U_k(t)u_k(t) - \frac{V}{2}. \tag{13}$$

Lemma 1. *Consider two BSs, BS1 and BS2, providing the content C_1 and C_2, respectively. If an UMM connects to both BSs, then the order of the selection to BS1 and BS2 do not affect the performance, i.e. $g(C_1) + g(C_2 \setminus C_1) = g(C_2) + g(C_1 \setminus C_2)$, where $C_i \setminus C_j$ denotes subtracting C_j from the set of C_i.*

Proof: The improvement of (12) by first connecting to BS1 owning the content C_1 then BS2 owning C_2 can be expressed as

$$\begin{aligned} & g(C_1) + g(C_2 \setminus C_1) \\ &= \sum_{k\in C_1} U_k(t)u_k(t) - \frac{V}{2} + \sum_{k\in C_2\setminus C_1} U_k(t)u_k(t) - \frac{V}{2} \\ &= \sum_{k\in C_1\cup C_2} U_k(t)u_k(t) - V. \\ &= \sum_{k\in C_2} U_k(t)u_k(t) - \frac{V}{2} + \sum_{k\in C_1\setminus C_2} U_k(t)u_k(t) - \frac{V}{2} \\ &= g(C_2) + g(C_1 \setminus C_2), \end{aligned} \tag{14}$$

which complete the proof. ■

Theorem 1. *(12) is tractable at least using the greedy scheme. That is, an UMM first selects a BS which can provide the largest improvement of (12), then selects the one which can provide the second largest improvement, and so on, until the objective cannot be further maximized.*

Proof: Consider a set of available BSs, $\mathcal{A}^f$, which can be divided into two groups $\mathcal{A}_1^f$ and $\mathcal{A}_2^f$ containing the contents C_1 and C_2, respectively. Without loss of generality, we assume $g(C_1) \geq g(C_2)$, where $g(C)$ is defined in (13). If connecting to both groups can improve the performance, according to **Lemma 1**, then the order of connection does not affect the performance. We then consider another scenario. If connecting to $\mathcal{A}_2^f$ cannot improve the performance after connecting toh $\mathcal{A}_1^f$, *i.e.*, $g(C_2 \setminus C_1) < 0$, then the following equation holds

$$g(C_1) > g(C_1) + g(C_2 \setminus C_1) = g(C_2) + g(C_1 \setminus C_2). \tag{15}$$

The last equality is actually the result of first connecting to $\mathcal{A}_2^f$ and then $\mathcal{A}_1^f$. Therefore, (12) is tractable using the greedy scheme. ■

With the essence paved by **Theorem 1**, we can thus propose the following **Algorithm 1** to effectively solve

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

(12).

Algorithm 1 Fronthaul First Backhaul Last (FFBL)

Require: A set of available BSs $\mathcal{A}_t^f$
Each BS $n \in \mathcal{A}_t^f$ provides a set of contents C_n.
Chosen set $A_t^f = \varnothing$.
Ensure: Select a set of BSs $A_t^f \in \mathcal{A}_t^f$ to minimize $\sum_{n\in A_t^f} g(C_n)$.
1: **repeat**
2: Select a BS with the largest contribution $n^* = arg\max_{n\in\mathcal{A}_t^f} g(C_n)$.
3: **If** $g(C_{n^*}) > 0$
4: $A_t^f = A_t^f \bigcup n^*$
5: $\mathcal{A}_t^f = \mathcal{A}_t^f \setminus n^*$.
6: **else**
7: Terminate
8: **until** $\mathcal{A}_t^f = \varnothing$.

C. Performance Analysis of the Proposed Scheme

In above section, we have demonstrated that the optimum control in **Optimization 1** can be transferred to (12). We have further shown that (12) is tractable with the facilitation of **Lemma 1** and **Theorem 1**. **Algorithm 1** is subsequently proposed to solve (12). In this section, we continue the discussion on the performance of this proposed scheme. For this purpose, the *Lyapunov drift function* in stochastic optimization [26] inspires the derivation of this performance analysis.

Definition 4. *The quadratic Lyapunov function of **Optimization 1** can be defined by*

$$L(t) \triangleq \sum_{k\in\mathcal{K}} U_k^2(t) + X^2(t). \tag{16}$$

The spirit of this definition is that all the queues, including the virtual queue, can be stabilized if the value of $L(t)$ decreases consistently over time.

Definition 5. *The corresponding Lyapunov drift function of **Definition 4** can be defined by $\Delta L(t) = L(t+1) - L(t)$, which captures the increasing (or decreasing) rate of the Lyapunov function.*

To stabilize the queues, the value of the *Lyapunov drift function* is expected to be as negative as possible since a negative drift reduces overall queue length. It suggests that more BSs (and thus more resources at fronthaul links) are utilized to increase the content dissemination at fronthaul links. However, the amount of utilized resources at fronthaul links should be minimized to avoid severe congestion at fronthaul links leading to the unacceptable latency performance. To this end, a *drift-plus-penalty* function can be introduced, which is defined by $\Delta L(t) + Vn_f(t)$, where $n_f(t)$ can be regarded as the amount of utilized resources at fronhaul links (i.e., the number of the connected BSs). $V \geq 0$ is the weighting constant on the $n_f(t)$. Consequently, the *drift-plus-penalty* captures the tradeoff between the utilization of the BSs and the latency performance.

Theorem 2. *The performance of **Optimization 1** is bounded by $\frac{\sum_k B_k + D}{V} + n_f^*$, where $B_k \triangleq \mathbb{E}\left(\varpi_l^2 + \varpi_l + a_k^2(t)\right)$ and $D \triangleq \mathbb{E}\left(\vartheta^2 + |\mathcal{K}|^2\right)$.*

To prove this theorem, we may introduce the following lemma.

Lemma 2. *For positive real numbers X, Y, μ, υ satisfying*

$$Y = \max[X - \mu, 0] + \upsilon, \tag{17}$$

then $Y^2 \leq X^2 + \mu^2 + \upsilon^2 - 2X(\mu - \upsilon)$.

Proof: If $X - \mu < 0$, then the followings hold

$$\begin{aligned} Y^2 = \upsilon^2 &\leq (X-\mu)^2 + \upsilon^2 + 2X\upsilon \\ &= X^2 + \mu^2 + \upsilon^2 - 2X(\mu - \upsilon). \end{aligned} \tag{18}$$

If $X - \mu > 0$, Y^2 satisfies

$$\begin{aligned} Y^2 &= (X-\mu)^2 + \upsilon^2 + 2(X-\mu)\upsilon \\ &= X^2 + \mu^2 - 2X\mu + \upsilon^2 + 2X\upsilon - 2\mu\upsilon \\ &\leq X^2 + \mu^2 + \upsilon^2 - 2X(\mu - \upsilon) \end{aligned} \tag{19}$$

■

With the facilitation of **Lemma 2**, we are ready to proceed to the proof of **Theorem 2**. According to **Lemma 2**, we obtain

$$\begin{aligned} &\mathbb{E}\left(U_k^2(t+1) - U_k^2(t)\right) \\ &\leq \mathbb{E}(u_k^2(t) + a_k^2(t) 2U_k(t)u_k(t) + a_k^2(t) \\ &+ 2a_k(t)\left(U_k(t) - u_k(t)\right)) \\ &\leq B_k - \mathbb{E}\left(2U_k(t)\left(u_k(t) - a_k(t)\right)\right), \forall k \in \mathcal{K} \end{aligned} \tag{20}$$

where $\mathbb{E}\left(u_k^2(t)\right)$ is replaced with the maximum possible value of $\varpi_l^2 + \varpi_l$ which is the mean square value of the amount of acquired contents defined in (3). Likewise,

$$\mathbb{E}\left(X^2(t+1) - X^2(t)\right) \leq D - \mathbb{E}\left(2X(t)\left(\vartheta - n_b(t)\right)\right), \tag{21}$$

where $n_b^2(t)$ is replaced with the maximum possible value $\mathcal{K}$ since the maximum value of $n_b(t)$ occurs when all contents are downloaded from the cloud server. Taking the summation of (20) and (21), we obtain

$$\begin{aligned} \Delta L(t) \leq &\sum_{k\in\mathcal{K}} B_k + D \\ &- 2\sum_{k\in\mathcal{K}} \mathbb{E}\left[U_k(t)\left(u_k(t) - a_k(t)\right) + X(t)\left(\vartheta - n_b(t)\right)\right]. \end{aligned} \tag{22}$$

Adding both side with $Vn_f(t)$ and averaging the result over time, we obtain

$$\begin{aligned} V\overline{n}_f(t) \leq &\sum_{k\in\mathcal{K}} B_k + D + 2\sum_{k\in\mathcal{K}} \mathbb{E}\left(U_k(t)a_k(t) - X(t)\theta\right) \\ &- 2\left(\sum_{k\in\mathcal{K}} \mathbb{E}\left[U_k(t)u_k(t) - X(t)n_b(t)\right] + \frac{V}{2} n_f(t)\right). \end{aligned} \tag{23}$$

We can observe that (12) is exactly to minimize the right-hand side of above inequality. It is noted that expectation is not taken on $U_k(t)$ and $X(t)$, since both of these variables are regarded as constants at time t. Denote the optimal

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

$n_f(t)$ utilization scheme as $n_f^*(t)$. Since $\mathbb{E}\left(u_k(t) - a_k(t)\right)$ and $\mathbb{E}\left(\theta - n_b(t)\right)$ are larger than 0, the upper bound of the average utilization of the BSs can be obtained by $\overline{n}_f \leq \frac{\sum_k B_k + D}{V} + n_f^*$ The first term on the right-hand-side is the excess utilization of the BSs, and a larger excess term leads to a smaller average queue size. The second term on the right-hand-side, n_f^*, is the minimum utilization of BSs such that the queue can be stabilized. Consequently, when n_f^* is achieved through **Algorithm 1**, $\overline{n}_f$ is minimized as well.

V. Latency Control to Information Cache and Dissemination in Rural Areas

As aforementioned, the cost of cache is directly related to the time duration to perform cache. The cost of cache can be formulated into the cost of control $u_k(t)$ in the duration, since cache leads to higher task departure rate of a queue. Besides the cost, the efficient control of cache also depends on the number of recovered UMMs. At the beginning, there is no recovered UMM, causing scarce sharing and inefficient cache. Nevertheless, the number of recovered UMMs increases by time to enhance the sharing. Aiming to determine the optimal control, we exploit optimal control theory [27], [28] to capture the cost of cache and the efficiency of epidemic content dissemination. The goal is to minimize the performance measurement in the form of the Cobb-Douglas production function.

Optimization 3. *The optimum control to the beginning time to perform cache aims at solving the following optimization,*

$$T_C^* = \underset{T_C}{arg\ min} \int_0^{T_f} I(t)^\beta + \frac{1}{\alpha} u_k(t)^\alpha dt \tag{24}$$

where $u_k(t)$ takes the α-power form, $\alpha \geq 0$. $\beta \geq 0$ represents the requirement of network dynamic. T_f is the completion time for observation.

To solve this optimization, we construct the functional Hamiltonian $\mathcal{H}$ by applying Euler-Lagrange equation as

$$\begin{aligned}
&\mathcal{H}\left(I(t), R(t), u_k(t), \Lambda_I(t), \Lambda_R(t)\right) \\
&= I(t)^\beta + \frac{1}{\alpha} u(t)^\alpha \\
&+ \Lambda_I(t) \left[\nu\eta \hat{S}(t) I(t) - u_k(t) I(t) - \phi(t)\eta \hat{R}(t) I(t)\right] \\
&+ \Lambda_R(t) \left[u_k(t) I(t) + \phi(t)\eta \hat{R}(t) I(t)\right]
\end{aligned} \tag{25}$$

from which the co-state variables $\Lambda_I^*(t)$ and $\Lambda_R^*(t)$ are the partial derivatives of the Hamiltonian with respect to $I(t)$ and $R(t)$:

$$\begin{aligned}
\dot{\Lambda}_I^*(t) &= -\frac{\partial \mathcal{H}}{\partial I} = -\beta I(t)^{\beta-1} - \Lambda_I^*(t) \\
&\times \left[\nu\eta \frac{N - 2I(t) - R(t)}{N} - u_k(t) - \phi(t)\eta \frac{R(t)}{N}\right] \\
&- \Lambda_R^*(t) \left[u(t) + \phi(t)\eta \frac{R(t)}{N}\right]
\end{aligned} \tag{26}$$

$$\begin{aligned}
\dot{\Lambda}_R^*(t) &= -\frac{\partial \mathcal{H}}{\partial R} \\
&= \Lambda_I^*(t) \left[\nu\eta \frac{I(t)}{N} + \phi(t)\eta \frac{I(t)}{N}\right] - \Lambda_R^*(t)\phi(t)\eta \frac{I(t)}{N}
\end{aligned} \tag{27}$$

with boundary conditions $\dot{\Lambda}_I^*(T_f) = \dot{\Lambda}_R^*(T_f) = 0$. Assuming that all of the state and co-state variables are according to their values for the optimal control $u_k^*(t)$, we rewrite the Hamiltonian in (25) with the switching function

$$\theta^*(t) := \Lambda_I^*(t) I(t) - \Lambda_R^*(t) I(t) = \left[\Lambda_I^*(t) - \Lambda_R^*(t)\right] I(t) \tag{28}$$

yielding

$$\begin{aligned}
&\mathcal{H}\left(I^*(t), R^*(t), u_k(t), \Lambda_I^*(t), \Lambda_R^*(t)\right) \\
&= I(t)^\beta + \frac{1}{\alpha} u_k(t)^\alpha - \theta^*(t) u_k(t) \\
&+ \eta I^*(t) \left[\Lambda_I^*(t) \left(\nu \hat{S}(t) - \phi(t)\hat{R}(t)\right) + \Lambda_R^*(t)\phi(t)\hat{R}(t)\right]
\end{aligned} \tag{29}$$

By Pontryagin's minimum principle [29], the unconstrained optimal control $U^*(t)$ with free end time T_f that minimizes the performance measure is the solution of the equation $\frac{\partial \mathcal{H}}{\partial u_k} = 0$. From the reformed Hamiltonian in (29), we have $U_k^*(t) = \theta^*(t)^{\frac{1}{\alpha-1}}$. That is, $U_k^*(t)$ can be obtained by solving the state variables in (6), (26) and (27). Moreover, with the acceptable control $u_k(t) \in [0, 1]$, the induced constrained optimal control $u_k^*(t)$ is

$$u_k^*(t) = \begin{cases} 0 & \text{if } \theta^*(t) \leq 0 \\ \theta^*(t)^{\frac{1}{\alpha-1}} & \text{if } \theta^*(t) \in (0, 1) \\ 1 & \text{if } \theta^*(t) \geq 1. \end{cases} \tag{30}$$

We may observe a discrepancy between $u_k(t)$ in (4) and the constrained optimal control $u^*(t)$, since $u^*(t)$ is acquired by presuming that cache can be performed at initial time 0. In (4), we consider a realistic situation where $u_k(t)$ increases only after the optimal caching time T_C^*. With T_C^*, we can bridge the gap between (4) and (30).

VI. Performance Evaluation

A. Simulation Studies of the Hot-Spot Area Cases

For the hot-spot areas, each BS selects Z_l contents among $\mathcal{K}$ to store in the local cache. Since the local cache stores location-based knowledge, these contents may be equally relevant to UMMs. To capture this characteristic, the popularities of Z_l contents is assumed to follow the uniform distribution. Each BS selects Z_g contents to store in the global cache following the Zipf distribution. Due to the mobile nature of UMMs, the number of connectable BSs for each UMM, $N_{bs}(t)$, may vary over time t. To capture this dynamics, $N_{bs}(t)$ is modeled as a queue, that is,

$$N_{bs}(t+1) = \max(N_{bs}(t) - l_{bs}(t), 0) + a_{bs}(t), \tag{31}$$

where $l_{bs}(t)$ and $a_{bs}(t)$ are the numbers of BSs which leave and enter the connectable region (e.g., the distance between an UMM and a BSis less than 150 m) of an UMM at

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

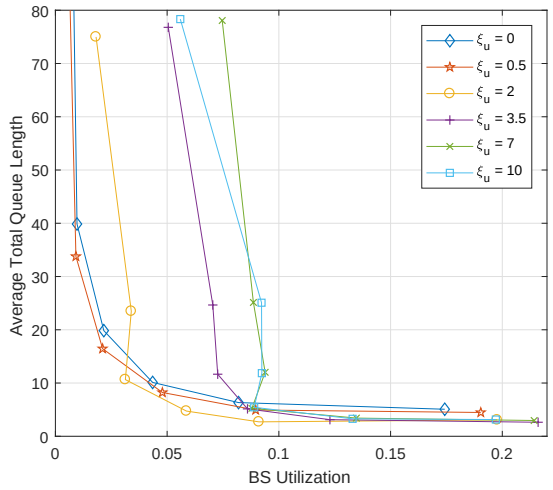

Fig. 2. Average total queue length with different BS utilizations and ξ_u, where the cloud server utilization constraint $\vartheta = 0.3$, $\xi = 5$, $Z_l = 10$ and $Z_g = 10$.

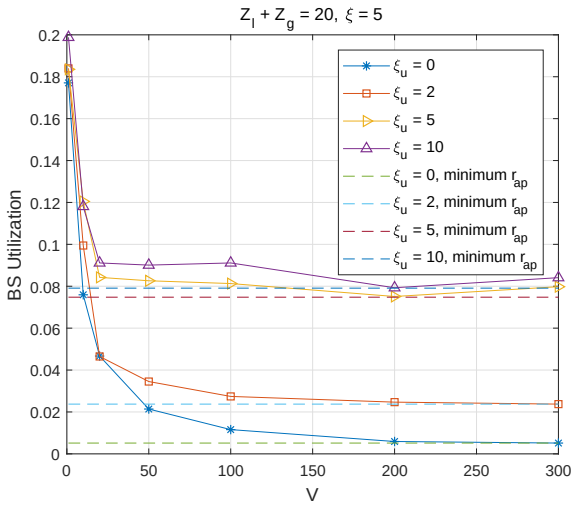

Fig. 3. Minimum BS utilization under different V and ξ_u.

time t, respectively. In this simulation, the velocity of each UMM is 20 m/s [30] (to simulate moderate mobility of drones), and the deployment of BSs follows a homogeneous Poisson Point Process with the density $5 \times 10^{-5}/m^2$. The task arrival process follows a Poisson process as a demonstration example. The mean of the total task arrival rate is normalized to 1, *i.e.*, $\sum_k a_k(t) = 1$. The amount of total available content $\|\mathcal{K}\| = 40$, and processing rates are considered to be $\varpi_l = 2$ and $\varpi_g = 1.5$.

In Fig. 2, we evaluate the optimum performance in terms of the average total queue length of all UMMa under different BS (fronthaul link) utilization values. In this performance evaluation, the skew factor of each UMM ξ_u is adopted to capture whether an UMM demands only a particular sort of contents (low ξ_u) or a variety of contents (high ξ_u). We can observe from Fig. 2 that, the average queue length increases under a low BS utilization. When the BS utilization is low, the fronthaul links become the bottleneck to acquire contents, which thus increases the average queue length. Please note that, a low BS utilization is resulted from a large V value. Since V is the cost of utilizing a BS (fronthaul link), a large V is the result of a scheme preferring to connect to less BSs (fronthaul links), which consequently increases the queue length. Fig. 2 also shows that, a high ξ_u may lead to an extremely large average total queue length under a low BS utilization. This phenomenon is also expected. Since a high ξ_u value suggests that each UMM may demand a variety of contents, if the BS utilization is low, then each UMM may utilize few fronthaul resources. As a result, it is likely that an UMM may not obtain desired information from the accessed BS. This result also aligns with the intuition that there is a tradeoff between the BS utilization and the average queue length.

In Fig. 3, we demonstrate that the proposed FFBL scheme is able to effectively achieve the optimum performance. In [14], the ideal performance of information cache and dissemination at network edges in terms of the minimum BS utilization has been derived as $r_{bs} \triangleq \frac{n_f^*}{\mathbb{E}(\|\mathcal{A}_f^t\|)}$. We can observe from Fig. 3 that, under different skew factors ξ_u, the BS utilization asymptotically approaches to the ideal performance with the increment of V. Aligning with the results in Fig. 2, the BS utilization decreases with the increment of V until the minimum BS utilization is reached. This result suggests that if UMMs desires a variety of contents, we can deploy BSs with a lower density.

In Fig. 4, we further compare the performance of the proposed FFBL scheme with that of the cross-layer cache architecture (CLCA) [31]. In the CLCA [31], each mobile device may connect to all available BSs with the facilitation of the downlink coordinated multi-point transmissions, regardless of the traffic congestion at the fronthaul links. Since there is a tradeoff between the BS utilization and latency performance, an effective cache design should both minimize the BS utilization and the length of a task queue. In this simulation, the *Pareto efficiency* is therefore adopted as the performance metric to evaluate the effectiveness of the proposed scheme, where the Pareto efficiency is defined as the product of the BS utilization and the length of a task queue. Consequently, the Pareto efficiency should be as low as possible. We can observe from Fig. 4 that the proposed scheme outperforms the CLCA under any value of the global skew factor ξ. Under a high global skew factor, information dissemination mainly relies on the backhaul link. Since an UMM only connects to a limited number of BSs if information should be acquired through the backhaul link, the proposed scheme offers a better performance in terms of the Pareto efficiency in this case. On the other hand, under a low skew factor, information is mostly obtained from the global cache and local cache in BSs. In this case, the proposed scheme also optimizes the utilization of fronthaul links as well, to lead to an outstanding Pareto efficiency.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

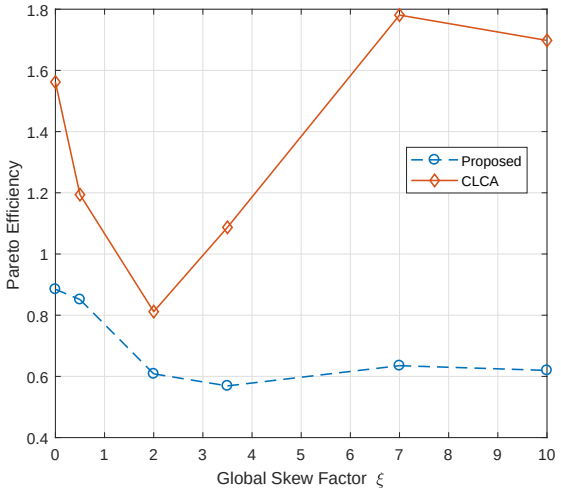

Fig. 4. Pareto Efficiency defined as the product of the BS utilization and the queue length under different global skew factors ξ.

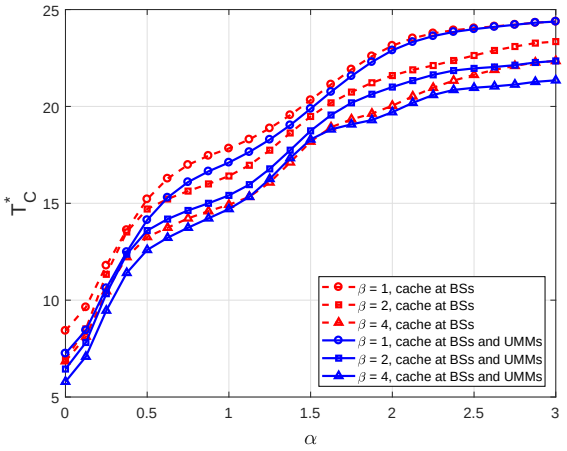

Fig. 5. Optimal caching time T_C^* under different (α, β) configurations, where α stands for the cost of caching and β stands for the requirement of system dynamic.

B. Simulation Studies of the Rural Area Cases

Subsequently, we investigate the optimal beginning time to perform cache with respect to the cost of cache α and the requirement of system dynamics in the UMM network β formed by UMMs and BSs. α could be understood as the scarcity of cache resource in BSs. In system view, a larger amount of contents circulating in BSs sharing the BS storage makes storage capacity more scarce, and thus leads to a larger α. Meanwhile, β represents the requirement of the quality of system dynamics. When the quality is required high, $I(t)$ should be as few as possible, and β becomes large. Moreover, the optimal beginning time to perform cache is also investigated with respect to the virality of the content, as a different system feature other than popularities of the contents. Pertaining to the simulation setup, $N = 1000$ UMMs are moving in a square area with wrap-around condition via Lèvy walk mobility model to account more for practical mobility behavior [32], where the step size and pause time are accounted by a power-law distribution with negative exponent. The step size exponent is set to 1.5 and the pause time exponent is set to 1.38, which fit the real trace-based data collected in [33]. Other parameters for this simulation are $I_0 = 1$, $L = 100$, $\delta = 1$, $\nu = 1$, $\kappa_1 = 0.1$, $\kappa_2 = 0.2$, $\kappa = 0.15$, $T_f = 150$, $\Lambda_I(0) = 20$, $\Lambda_R(0) = 10$.

In Fig. 5, the optimal cache time increases with the increase of α, since if there are many contents circulating among UMMs and (few) BSs, a longer time should be required to decide whether to cache the viral content, in order to optimize the usage of storage capacity at BSs and the UMMs. Nevertheless, since the number of UMMs who have viewed the content grows by time, late caching time implies better chance to efficaciously utilize caching among UMMs for sharing. However, as we require the number of UMMs waiting for the content as few as possible, the optimal caching time becomes early to handle the requirement. These two factors form a trade-off in design optimal caching utilization to serve epidemic contents. Furthermore, we observe that caching both at BSs and UMMs contributes to early cache by starting storing in recovered UMMs to create more sharers of epidemic contents.

Finally, the optimal cache time T_C^* versus virality ν is demonstrated in Fig. 6. Obviously, a highly viral content advances the optimal cache time to handle suddenly massive requirements; while for lowly viral content, the optimal cache time is late since it takes a longer time for the content to infect enough UMMs to help serve the UMMs. For different situations of cache cost and the requirement of system dynamic, our previous discussion still holds; that is, large α delays optimal cache time and large β advances it. The virality of an epidemic content has more pragmatic meaning than merely the popularity, since prefect and centralized traffic monitoring to obtain popularity is often arduous and not timely. Therefore, virality holds a chance as a more realistic feature when it comes to UMMs, and an important system characteristic when designing information sharing networks. It is obvious that for a small change in virality, e.g. from 1.5 to 2.0, the optimal cache time could vary largely.

VII. Conclusion

In this paper, a latency control to optimize the resource utilization at fronthaul links under a given queue length constraint (latency constraint) for UMMs in the hot-spot areas, and to determine the optimum cache time for autonomous information dissemination among UMMs in the rural areas is analytically derived. Given the imposed cost to perform content cache and dissemination, the popularity of the contents and the virality of the contents, the proposed control scheme is able to optimize the performance in spite of mobility of UMMs. The proposed scheme thus offers essential foundations for the frontier of information cache and dissemination designs of UMMs under practical cellular infrastructure deployment.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2018.2834619, IEEE Transactions on Industrial Informatics

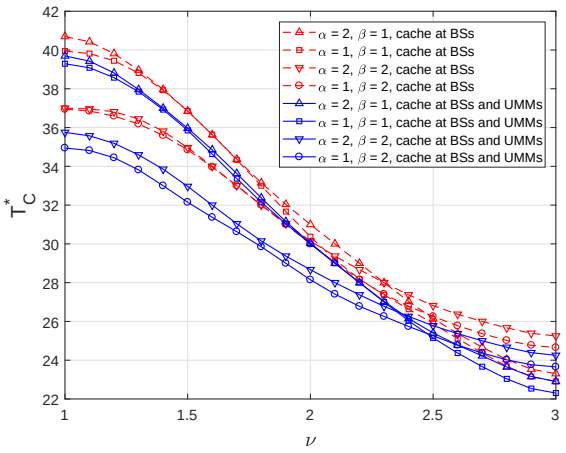

Fig. 6. Optimal caching time versus virality ν under different (α, β) configurations.

In fact, performing information cache and dissemination at network edges also facilitates a high energy efficiency in cellular infrastructures. Since the exploitation of the backhaul link is alleviated, energy consumptions at backhaul routers/switches can be abated. However, energy consumptions at fronthaul links may still remain. To minimize latency, an UMM may utilize as many fronthaul link resources as possible, and therefore there is a tradeoff between latency and energy efficiency when information cache and dissemination are applied. A future work of this research consequently should aim at achieve the optimum tradeoff between latency and energy efficiency.

ACKNOWLEDGEMENT

This research is supported by National Chung-Shan Institute of Science and Technology and Ministry of Science and Technology under contracts 106-2221-E-194-065-MY2.

REFERENCES

[1] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[2] F. A. A. Cheein and R. Carelli, "Agricultural robotics: Unmanned robotic service units in agricultural tasks," *IEEE Electr. Insul. Mag.*, vol. 7, no. 3, pp. 48–58, Dec. 2013.

[3] N. Wang, J. C. Sun, M. J. Er, and Y. C. Liu, "A novel extreme learning control framework of unmanned surface vehicles," *IEEE Trans. Cybern.*, vol. 46, no. 5, pp. 1106–1117, May 2016.

[4] M. Simsek, A. Aijaz, M. Dohler, J. Sachs, and G. Fettweis, "5G-enabled tactile internet," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 460–473, Mar. 2016.

[5] H. Wang, X. Zhou, and M. C. Reedh, "Coverage and throughput analysis with a non-uniform small cell deployment," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 2047–2059, Apr. 2014.

[6] S.-Y. Lien, S. C. Hung, K. C. Chen, and Y. C. Liang, "Ultra-low-latency ubiquitous connections in heterogeneous cloud radio access networks," *IEEE Wireless Commun. Mag.*, vol. 22, no. 3, pp. 22–31, Jun. 2015.

[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] H. Hsu and K. C. Chen, "A resource allocation perspective on caching to achieve low latency," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 145–148, Jan. 2016.

[9] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, Jan. 2016.

[10] Y. Wang, J. Wu, and M. Xiao, "Hierarchical cooperative caching in mobile opportunistic social networks," in *Proc. of IEEE GLOBECOM*, 2014.

[11] D. J. Daley, J. Gani, and J. M. Gani, *Epidemic modelling: an introduction*. Cambridge University Press, 2001.

[12] Y. Cui, V. K. N. Lau, R. Wang, H. Huang, and S. Zhang, "A survey on delay-aware resource control for wireless systems: Large deviation theory, stochastic lyapunov drift, and distributed stochastic learning," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1677–1701, Mar. 2012.

[13] G. Pallis, "Cloud computing: The new frontier of internet computing," *IEEE Internet Comput.*, vol. 14, no. 5, pp. 70–73, Sep. 2010.

[14] S.-C. Hung, H. Hsu, S.-Y. Lien, and K.-C. Chen, "Architecture harmonization between cloud radio access networks and fog networks," *IEEE Access*, vol. 3, pp. 3019–3034, Dec. 2015.

[15] W. Wang, R. Lan, J. Gu, A. Huang, H. Shan, and Z. Zhang, "Edge caching at base stations with device-to-device offloading," *IEEE Access*, vol. 5, pp. 6399–6410, Mar. 2017.

[16] D. Liu, B. Chen, and C. Y. A. F. Molisch, "Caching at the wireless edge: design aspects, challenges, and future directions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 22–28, Sep. 2016.

[17] D.-J. Deng, S.-Y. Lien, C.-C. Lin, S.-C. Hung, and W.-B. Chen, "Latency control in software-defined mobile-edge vehicular networking," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 87–93, Aug. 2017.

[18] J. Liu, B. Bai, J. Zhang, and K. B. Letaief, "Cache placement in Fog-RANs: From centralized to distributed algorithms," vol. 16, no. 11, pp. 7039–7051, Nov. 2017.

[19] J. Wen, K. Huang, S. Yang, and V. O. K. Li, "Cache-enabled heterogeneous cellular networks: Optimal tier-level content placement," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5939–5952, 2017.

[20] N. Golrezaei, A. G. Dimakis, and A. F. Molisch, "Scaling behavior for device-to-device communications with distributed caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4286–4298, 2014.

[21] B. Chen, C. Yang, and A. F. Molisch, "Cache-enabled device-to-device communications: Offloading gain and energy cost," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4519–4536, 2017.

[22] G. Zhang, J. Liu, J. Ren, L. Wang, and J. Zhang, "Capacity of content-centric hybrid wireless networks," *IEEE Access*, vol. 5, pp. 1449–1459, Feb. 2017.

[23] Y. Zhou, L. Chen, C. Yang, and D. M. Chiu, "Video popularity dynamics and its implication for replication," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1273–1285, Aug. 2015.

[24] S. Meyn, *Control techniques for complex networks*. Cambridge University Press, 2008.

[25] H. Hsu and K.-C. Chen, "A resource allocation perspective on caching to achieve low latency," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 145–148, 2016.

[26] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool Publishers, 2010.

[27] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2012.

[28] C. W. Cobb and P. H. Douglas, "A theory of production," *American Economic Review*, vol. 18, pp. 139–165, 1928.

[29] L. S. Pontryagin, *Mathematical theory of optimal processes*. CRC Press, 1987.

[30] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, Sep. 2016.

[31] W. C. Ao and K. Psounis, "Fast content delivery via distributed caching and small cell cooperation," *IEEE Trans. Mobile Comput.*, vol. 17, no. 5, pp. 1048–1061, May 2018.

[32] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the levy-walk nature of human mobility," *IEEE/ACM Trans. Netw.*, vol. 19, no. 3, pp. 630–643, 2011.

[33] S. Kim, C.-H. Lee, and D. Y. Eun, "Superdiffusive behavior of mobile nodes and its impact on routing protocol performance," *IEEE Trans. Mobile Comput.*, vol. 9, no. 2, pp. 288–304, 2010.