AoI-Oriented Content Caching and Updating in Maritime Internet of Things

Ruijin Sun*, Yujie Zhang*, Nan Cheng*, Rong Chai†, Tingting Yang‡§, Meng Qin§
* State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China
† Chongqing Key Lab. of Mobile Commun. Technol., Chongqing Univ. of Posts & Telecom., Chongqing 400065, China
‡ Navigation College, Dalian Maritime University, Dalian 116026, China
§ Pengcheng Laboratory, Shenzhen 518055, China
Email: sunruijin@xidian.edu.cn, yjzhang_5@stu.xidian.edu.cn, nancheng@xidian.edu.cn,
chairong@cqupt.edu.cn, yangtingting820523@163.com, qinm01@pcl.ac.cn

***Abstract*—Caching popular contents at the base station (BS) in maritime Internet of Things (IoT) networks makes sensor nodes be free from frequently responding to user requests, which can remarkably save the energy consumption of sensor nodes. However, to ensure the freshness of contents, cached contents need to be updated periodically. Frequent content updating can minimize the age of information (AoI) of contents while increase the energy consumption of sensor nodes. To make a better tradeoff between the AoI and energy consumption, in this paper, both the cache placement and content updating interval are jointly optimized to minimize the weighted sum of AoI of contents and energy consumption of sensor nodes. As the formulated problem is a mixed integer nonlinear programming problem, the cache placement and the content updating interval are alternatively optimized. For the cache placement problem, a local optimal solution is achieved via the binary constraint reformulation and successive convex approximation. For the content updating problem, the optimal solution with semi-closed form is derived. Simulation results show that our proposed algorithm outperforms other benchmarks in terms of the weighted sum of AoI and energy consumption.**

***Index Terms*—AoI, content caching and updating, energy consumption.**

I. Introduction

By enabling massive sensor nodes to sense and interact with physical real world without human intervention, Internet of Things (IoT) has the potential to be widely used in smart cities, such as intelligent traffic surveillance, intelligent vehicular communication networks, smart home systems and so on. Except for the land mobile communications, IoT can also been widely used in maritime communication networks to monitor the marine ecological environment for the human activities and resource explorations in the ocean. Specific applications range from the surveillance of temperature and salinity of the sea to the development surveillance of inshore fishery and aquaculture industries. Along with the widespread proliferation of IoT applications, a large amount of data needs to be transmitted from sensor nodes to users, which significantly increases the energy consumption of low-power sensor nodes and consumes huge wireless communication resources.

To tackle this issue, caching frequently requested contents generated by sensor nodes at the base station (BS) (i.e., IoT gateway) can be regarded as a potential technology [1]–[3]. With the popular contents caching, users can directly fetch their requested contents from the BS, which can avoid frequently awaking sensor nodes and remarkably reduce the energy consumption. Furthermore, as contents are brought closer to users, the content delivery delay can also be reduced [4], [5]. To investigate which contents to be cached at BS, a body of works regarding the IoT caching have been done. In [6], the content popularity in IoT caching networks is predicted via the deep neural networks. Then, with the given content popularity, the cache placement problems are respectively investigated to minimize the power consumption of sensor nodes [7] and reduce the transmission delay [8].

Different from traditional multimedia contents (e.g., video) with static version, contents generated by sensor nodes are generally transient. In specific, the contents usually have limited lifetime and will dynamically change with time and environment. To characterize the freshness of contents, the concept of age of information (AoI) is proposed, which is defined as the time elapsed since the generation of the current version of contents [9], [10]. Since then, IoT caching considering the content freshness has caught significant attention. In [11], the long-term cache placement design in IoT networks is studied via the deep reinforcement learning to minimize the network cost caused by data freshness loss and data communications. Besides the cache placement problem, the optimal content refresh probability is derived in [12] and the content updating scheme is proposed in [13]. In these works about content updating, all contents are assumed to be cached at BS, which is impractical for the scenario with limited cache capacity in BS.

In contrast to previous works separately considering the cache placement and the content updating, this paper investigates the joint cache placement and content updating (JCPCU) problem in maritime IoT caching networks. In particular, in the considered IoT caching networks, user requests are directly served by BS if the corresponding contents are cached in BS. This can reduce the energy consumption of sensor nodes while increase the AoI of cached contents. To make a better tradeoff between the AoI and energy consumption, the cache placement and content updating are jointly designed to minimize the weighted sum of AoI and energy consumption of sensor nodes, considering the content refresh frequency constraint and the

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001473

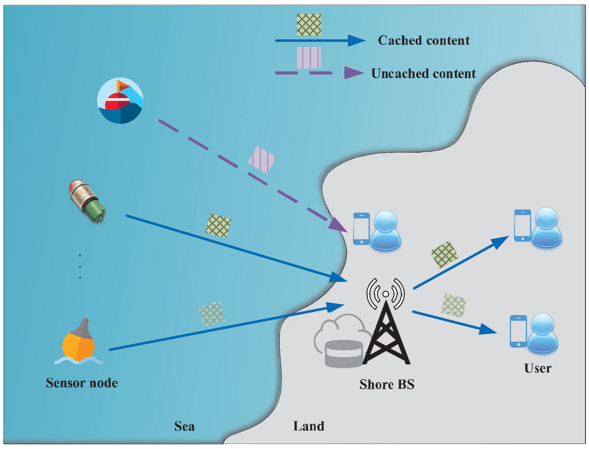

Fig. 1. Maritime IoT caching networks.

cache capacity constraint. To solved this formulated mixed integer nonlinear programming (MINLP) problem, the cache placement and the content updating interval are alternatively optimized. A local optimal cache placement algorithm via the successive convex approximation (SCA) and a semi-closed optimal content updating interval are respectively designed. Simulation results show the effectiveness of our proposed JCPCU algorithm.

The remainder of the paper is organized as follows. Section II provides the system model and problem formulation. Section III proposes the JCPCU algorithm. Section V presents the simulation results and Section VI concludes this paper.

II. System Model and Problem Formulation

In this paper, we consider the IoT caching in inshore communication networks consisting of one shore BS and N different sensor nodes, as illustrated in Fig. 1. These sensor nodes are deployed offshore within the coverage of BS to monitor the Marine ecological environment, such as the temperature and the salinity of the sea, the development of inshore fishery and aquaculture industries and so on. Users interested with monitored contents are submitting their requests randomly. When user request arrives, the contents are sent to the corresponding user immediately. To avoid frequently awaking the sensor nodes and reduce the energy consumption, popular contents can be cached by the BS and directly transmitted from the BS to the requested users. In addition, to guarantee the freshness of the information received by users, contents cached by the BS are updated periodically.

The total user requests in this paper are modeled as Poisson process with intensity Λ. Denote by p_n the user request probability of content n generated by sensor node n, where $\sum_{n=1}^{N} p_n = 1$. Then, the user requests for content n follow Poisson process with intensity $\lambda_n = p_n\Lambda$. For the cache placement, let $\boldsymbol{x} = [x_1, x_2, ..., x_N]$ represent the cache placement vector in BS, where $x_n \in \{0, 1\}, \forall n$ is the binary cache placement indicator of content n. If content n is cached by BS, $x_n = 1$; otherwise, $x_n = 0$. Denote by S the cache capacity of BS and s_n the size of content n, $\sum_{n=1}^{N} s_n x_n \leq S$ should be satisfied due to the limited cache capacity.

A. AoI of Contents

AoI is proposed as a metric to characterize the content freshness and defined as the time elapsed since the generation of the current version of contents. In this paper, the AoI of a content depends on the cache placement at BS. Let A_n^b and A_n^s respectively denote the AoI expressions of cached content n and uncached content n, the average AoI of contents in the considered IoT caching networks is given as

$$A = \sum_{n=1}^{N} p_n \Big(x_n A_n^b + (1 - x_n) A_n^s \Big). \tag{1}$$

1) AoI of Cached Contents: If content n is cached by BS, to ensure the information freshness and reduce the AoI, content n is periodically refreshed with the updating interval T_n. In this case, the AoI of content n consists of both the content caching time T_n^c and content delivery delay $T_{b,n}^d$. The content caching time is the time between the latest content updating time and the arrival time of the user request.

To derive the caching time of content n, we first consider the case that M user requests for content n arrive during the content updating interval T_n. Let the timeline start from the latest content updating time. Denote t_m as the arrival time of m-th user request for content n during T_n. As user requests for content n follow Poisson process with intensity λ_n, t_m follows uniform distribution within $[0, T_n]$. Hence, with the given M user requests, the average conditional caching time of content n is expressed as

$$\mathbb{E}(T_n^c|_{\mathcal{M}=M}) = \mathbb{E}\left(\frac{1}{M} \sum_{m=1}^{M} t_m \middle| \mathcal{M} = M \right) = \frac{T_n}{2}, \tag{2}$$

where $\mathcal{M}$ represents the number of user requests for content n during content updating interval T_n. Then, considering all possible number of user requests, the average caching time of content n is given as

$$T_n^c = \sum_{M=0}^{\infty} P(\mathcal{M} = M)\mathbb{E}(T_n^c|_{\mathcal{M}=M}) = \frac{T_n}{2}. \tag{3}$$

This is due to the fact that $\sum_{M=0}^{\infty} P(\mathcal{M} = M) = 1$.

Then, we analyze the AoI caused by BS's content delivery in response to the user request. Denote by r_b the average transmission rate from the BS to users. Assume that each content size follows Poisson distribution with parameter $\bar{s}$, where $\bar{s}$ is the average size of all contents. Accordingly, the service rate of BS also follows Poisson distribution with parameter $\mu_b = r_b/\bar{s}$. Meanwhile, the user requests for all contents cached at the BS follow Poisson process with intensity λ_b, where $\lambda_b = \sum_{n=1}^{N} \lambda_n x_n$. As a result, the BS's content delivery can be modeled as an M/M/1 queueing model with arrival rate λ_b and service rate μ_b. Hence, the average content delivery delay is expressed as

$$T^d_{b,n} = \frac{1}{\mu_b - \sum_{n=1}^{N} \lambda_n x_n}. \tag{4}$$

Together with the caching time, the average AoI of content n if cached by BS is represented as

$$\begin{aligned} A^b_n &= T^c_n + T^d_{b,n} && (5)\\ &= \frac{T_n}{2} + \frac{1}{\mu_b - \sum_{n=1}^{N} \lambda_n x_n}. && (6)\end{aligned}$$

2) AoI of Uncached Contents: If content n is not cached by BS, the content is directly transmitted from sensor node n to the corresponding requested user. Denote by r^s_n the average transmission rate from sensor node n to users. Different from BS's content delivery process serving user requests for multiple kinds of contents, sensor node n only handles the user requests for content n. As the size of content n, s_n, normally remains unchanged, the service rate of sensor node n can be regarded as deterministic, which is $\mu^s_n = r^s_n / s_n$. Besides, the user requests for content n is Poisson process with intensity λ_n. The delivery model of sensor node n is modeled as an M/D/1 queueing model. As such, the average AoI (i.e., delay) of uncached content n is expressed as

$$A^s_n = \frac{\lambda_n}{2\mu^s_n(\mu^s_n - \lambda_n)} + \frac{1}{\mu^s_n}. \tag{7}$$

B. Energy Consumption of Sensor Nodes

If cached by BS, content n is sent from sensor n to BS at a frequency $1/T_n$ to refresh the cached information. Denote by ϵ_n the energy consumption of sensor node n when transmitting one megabit of content n. Then, the energy consumption for updating content n in unit time is expressed as

$$E^c_n = \frac{1}{T_n} s_n \epsilon_n. \tag{8}$$

For the case that content n is not cached by BS, the energy consumption in unit time is given as

$$E^s_n = \lambda_n s_n \epsilon_n. \tag{9}$$

Combining these two cases, the average energy consumption in the considered scenario is thus expressed as

$$\begin{aligned} E &= \frac{1}{N}\sum_{n=1}^{N} \frac{1}{T_n} s_n \epsilon_n x_n + \frac{1}{N}\sum_{n=1}^{N} \lambda_n s_n \epsilon_n (1 - x_n) \\ &= \frac{1}{N}\sum_{n=1}^{N} s_n \epsilon_n \left(\frac{1}{T_n} x_n + \lambda_n (1 - x_n)\right). && (10)\end{aligned}$$

It can be observed from (10) that, caching content n at BS if $\frac{1}{T_n} < \lambda_n$ can consume less energy. That means, the content updating frequency should be smaller than the user request arrival rate. Otherwise, it is no need to cache the content from the perspective of energy consumption.

C. Problem Formulation

In the considered IoT caching networks, from the users' perspective, the AoI of contents should be as small as possible to ensure the information freshness. This would be achieved by reducing the content updating interval for cached contents. However, the frequent content refreshing would increase the energy consumption of sensor nodes. To find a better tradeoff between the AoI and the energy consumption, in this paper, the cache placement and the content updating interval are jointly optimized to minimize the weighted sum of AoI and energy consumption of sensor nodes. Mathematically, the problem is formulated as ($\mathcal{P}1$)

$$\begin{aligned} \min_{\boldsymbol{x}, \{T_n\}} \; & \sum_{n=1}^{N} p_n \Big(x_n \Big(\frac{T_n}{2} + \frac{1}{\mu_b - \sum_{n=1}^{N} \lambda_n x_n}\Big) + (1 - x_n) A^s_n\Big) \\ & + w\left(\frac{1}{N}\sum_{n=1}^{N} s_n \epsilon_n \Big(\frac{1}{T_n} x_n + \lambda_n (1 - x_n)\Big)\right) && (11a)\\ \text{s. t.} \; & \sum_{n=1}^{N} x_n \frac{1}{T_n} \le F, && (11b)\\ & \sum_{n=1}^{N} x_n s_i \le S, && (11c)\\ & \sum_{n=1}^{N} \lambda_n x_n \le \mu_b, && (11d)\\ & x_n \in \{0, 1\}, \forall n, && (11e)\\ & T_n > 0, \forall n, && (11f)\end{aligned}$$

where $w > 0$ is the weighting factor and F is the maximal allowable content refresh frequency. In problem $\mathcal{P}1$, (11b) is the content updating frequency constraint at BS and (11c) is the cache capacity constraint at BS. Moreover, (11d) is to ensure that the transmission system at BS is stable, i.e., the service rate should be larger than the user request arrival rate. (11e) and (11f) are respectively the binary cache placement constraint and the positive content updating interval constraint.

It can be observed that problem $\mathcal{P}1$ is a MINLP problem, which is non-convex due to the binary cache placement indicator x_n and the coupling of x_n and T_n. In the following section, we propose an alternative method to deal with this non-convex problem.

III. Joint Cache Placement and Content Updating Design

In this section, an alternative algorithm is proposed to solve problem $\mathcal{P}1$. Specifically, the content updating interval T_n and the cache placement x_n are alternatively optimized until the algorithm converges.

A. The Content Updating Design

With the given cache placement indicator $\boldsymbol{x}$, problem $\mathcal{P}1$ is reduced as

$$\begin{aligned} \mathcal{P}2: \quad \min_{\{T_n\}} \quad & \sum_{n=1}^{N} \left(a_n T_n + \frac{b_n}{T_n}\right) + c && (12a)\\ \text{s. t.} \quad & \text{(11b) and (11f).} && (12b)\end{aligned}$$

where $a_n = p_n x_n/2$, $b_n = w s_n \epsilon_n x_n/N$ and $c = \sum_{n=1}^N p_n x_n/(\mu_b - \sum_n \lambda_n x_n) + \sum_{n=1}^N (1 - x_n) p_n A_n^s + \sum_{n=1}^N w\lambda_n(1 - x_n) s_n \epsilon_n/N$. It can be easily checked that $a_n \geq 0$ and $b_n \geq 0$. So problem $\mathcal{P}2$ is a convex problem and can be solved by CVX. To lower the complexity, a semi-closed form optimal solution in this paper is derived with the help of Karush-Kuhn-Tucker (KKT) conditions.

Proposition 1. Let α denote the lagrangian multiplier corresponding to constraint (11b), the optimal solution to problem $\mathcal{P}2$ satisfies one of the following two conditions.

- $\alpha = 0$, $T_n = \sqrt{\frac{b_n}{a_n}}$ and $\sum_{n=1}^N x_n \frac{1}{T_n} \leq F$;
- $\alpha > 0$, $T_n = \sqrt{\frac{b_n + \alpha x_n}{a_n}}$ and $\sum_{n=1}^N x_n \frac{1}{T_n} = F$.

Proof. The lagrangian function of problem $\mathcal{P}2$ is expressed as

$$\mathcal{L}(T_n; \alpha) = \sum_{n=1}^N \left(a_n T_n + \frac{b_n}{T_n} \right) + c + \alpha \left(\sum_{n=1}^N \frac{x_n}{T_n} - F \right). \tag{13}$$

Taking the derivative of $\mathcal{L}(T_n; \alpha)$ with respect to T_n, we have

$$\frac{\partial \mathcal{L}(T_n; \alpha)}{\partial T_n} = a_n - \frac{b_n}{T_n^2} - \frac{\alpha x_n}{T_n^2}, \forall n. \tag{14}$$

By setting $\frac{\partial \mathcal{L}(T_n;\alpha)}{\partial T_n} = 0$, we can derive that

$$T_n = \sqrt{\frac{\alpha x_n + b_n}{a_n}}, \forall n, \tag{15}$$

since $T_n > 0$.

Then, the KKT conditions of problem $\mathcal{P}2$ are listed as

$$\begin{cases} T_n = \sqrt{\dfrac{\alpha x_n + b_n}{a_n}}, \forall n, \\ \displaystyle\sum_{n=1}^N x_n \frac{1}{T_n} \leq F, \\ \alpha \geq 0, \\ \alpha \left(\displaystyle\sum_{n=1}^N x_n \frac{1}{T_n} - F \right) = 0. \end{cases} \tag{16}$$

It can be observed from the KKT conditions that, when $\alpha = 0$, $T_n^0 = \sqrt{\frac{b_n}{a_n}}$. If $\sum_{n=1}^N x_n \frac{1}{T_n^0} \leq F$ holds, all KKT conditions are satisfied. So the optimal solution to problem $\mathcal{P}2$ is $T_n = \sqrt{\frac{b_n}{a_n}}$. Otherwise, the optimal solution is $T_n = \sqrt{\frac{\alpha x_n + b_n}{a_n}}$ with $\alpha > 0$. In this case, to satisfy the complementary slackness condition $\alpha \left(\sum_{n=1}^N x_n \frac{1}{T_n} - F \right) = 0$, we can conclude that $\sum_{n=1}^N x_n \frac{1}{T_n} = F$. This completes the proof. ■

Based on Proposition 1, problem $\mathcal{P}2$ can be solved as follows. Let $T_n = \sqrt{\frac{b_n + \alpha x_n}{a_n}}$ and check that whether constraint (11b) holds with $\alpha = 0$. If the answer is yes, the optimal solution is $T_n = \sqrt{\frac{b_n}{a_n}}$. Otherwise, find the optimal α satisfying $\sum_{n=1}^N x_n \frac{1}{T_n} = F$ via the bisection method.

B. The Cache Placement Design

1) Problem Reformulation: With the given content updating interval T_n, problem $\mathcal{P}1$ is recasted as

$$\mathcal{P}3: \quad \min_{\boldsymbol{x}} \quad \frac{\sum_{n=1}^N p_n x_n}{\mu_b - \sum_{n=1}^N \lambda_n x_n} + \sum_{n=1}^N d_n x_n + e \tag{17a}$$

$$\text{s. t.} \quad (11\text{b}), (11\text{c}), (11\text{d}) \text{ and } (11\text{e}). \tag{17b}$$

where $d_n = p_n(T_n/2 - A_n^s) + w s_n \epsilon_n (1/T_n - \lambda_n)/N$, $e = \sum_{n=1}^N p_n A_n^s + \sum_{n=1}^N w \lambda_n s_n \epsilon_n/N$. Due to the fact that

$$\begin{aligned} \frac{\sum_{n=1}^N p_n x_n}{\mu_b - \sum_{n=1}^N \lambda_n x_n} &= \frac{\sum_{n=1}^N \lambda_n x_n}{\Lambda(\mu_b - \sum_{n=1}^N \lambda_n x_n)} \\ &= \frac{\mu_b}{\Lambda(\mu_b - \sum_{n=1}^N \lambda_n x_n)} - \frac{1}{\Lambda}, \end{aligned} \tag{18}$$

problem $\mathcal{P}3$ can be rewritten as

$$\mathcal{P}4: \min_{\boldsymbol{x}} \frac{\mu_b}{\Lambda(\mu_b - \sum_{n=1}^N \lambda_n x_n)} + \sum_{n=1}^N d_n x_n + e - \frac{1}{\Lambda} \tag{19a}$$

$$\text{s. t.} \ (11\text{b}), (11\text{c}), (11\text{d}) \text{ and } (11\text{e}). \tag{19b}$$

It can be observed that problem $\mathcal{P}4$ is a nonlinear integer programming problem. Although all constraints of problem $\mathcal{P}4$ are integer linear forms, the objective function (19a) is not additive with respect to x_n due to the nonlinear term, $\frac{\mu_b}{\Lambda(\mu_b - \sum_{n=1}^N \lambda_n x_n)}$. Thus, the dynamic programming method can not be adopted to solve this problem. In this paper, we handle this problem from the perspective of convex optimization theory.

Notice that problem $\mathcal{P}4$ is a convex problem except for the binary cache placement constraint. To remove the obstacle of this binary variable, we first relax it as an equivalent continuous form. Specifically, $x_n \in \{0, 1\}$ is equivalent to

$$\begin{cases} 0 \leq \tilde{x}_n \leq 1, \forall n, \\ \displaystyle\sum_{n=1}^N \tilde{x}_n - \sum_{n=1}^N \tilde{x}_n^2 \leq 0. \end{cases} \tag{20}$$

It is worth pointing out that the second expression in (20) would enforce the continuous $\tilde{x}_n$ to be either 0 or 1. Then, by introducing a positive penalty factor ρ and a non-negative slack variable v [14], problem $\mathcal{P}4$ can be solved via the following problem $\mathcal{P}5$.

$$\mathcal{P}5: \min_{\tilde{\boldsymbol{x}}, v} \frac{\mu_b}{\Lambda(\mu_b - \sum_{n=1}^N \lambda_n \tilde{x}_n)} + \sum_{n=1}^N d_n \tilde{x}_n + \rho v \tag{21a}$$

$$\text{s. t.} \quad 0 \leq \tilde{x}_n \leq 1, \forall n, \tag{21b}$$

$$\sum_{n=1}^N \tilde{x}_n - \sum_{n=1}^N \tilde{x}_n^2 \leq v, \tag{21c}$$

$$(11\text{b}), (11\text{c}) \text{ and } (11\text{d}), \tag{21d}$$

where $\tilde{\boldsymbol{x}}$ is the continuous cache placement vector and ρv is the penalty term. If $\tilde{x}_n \notin \{0, 1\}$, $\rho v > 0$. The minimization of (21c) would lead to $\tilde{x}_n \in \{0, 1\}$. Hence, problem $\mathcal{P}5$ is equivalent to problem $\mathcal{P}4$ and problem $\mathcal{P}3$.

2) SCA-Based Cache Placement Design: Note that problem $\mathcal{P}5$ is a convex problem except for constraint (21c). To deal with this challenge, the SCA method is used [15]. In particular, we iteratively replace the non-convex part of a expression by its first-order Tayor expansion. In constraint (21c), $-\sum_{n=1}^{N}\tilde{x}_n^2$ is the non-convex part and its first-order Tayor expansion in $i+1$-th iteration is given as

$$\begin{aligned} -\sum_{n=1}^{N}\tilde{x}_n^2 &\le -\sum_{n=1}^{N}\left(\tilde{x}_n^{(i)}\right)^2 - 2\sum_{n=1}^{N}\tilde{x}_n^{(i)}\left(\tilde{x}_n - \tilde{x}_n^{(i)}\right) \\ &= \sum_{n=1}^{N}\left(\tilde{x}_n^{(i)}\right)^2 - 2\sum_{n=1}^{N}\tilde{x}_n^{(i)}\tilde{x}_n, \end{aligned} \tag{22}$$

where $\tilde{x}_n^{(i)}$ is the given value of $\tilde{x}_n$ obtained from i-th iteration. Substituting (22) into constraint (21c), we can derive the following conservative approximation.

$$\sum_{n=1}^{N}\tilde{x}_n - \sum_{n=1}^{N}\tilde{x}_n \le \sum_{n=1}^{N}\left(1 - 2\tilde{x}_n^{(t)}\right)\tilde{x}_n + \sum_{n=1}^{N}\left(\tilde{x}_n^{(t)}\right)^2 \le v. \tag{23}$$

Consequently, in $i+1$-th iteration, the formulated convex problem is given as

$$\mathcal{P}6: \min_{\tilde{\boldsymbol{x}}, v} \frac{\mu_b}{\Lambda(\mu_b - \sum_{n=1}^{N}\lambda_n\tilde{x}_n)} + \sum_{n=1}^{N} d_n\tilde{x}_n + \rho v \tag{24a}$$

$$\text{s. t.} \quad \sum_{n=1}^{N}\left(1 - 2\tilde{x}_n^{(t)}\right)\tilde{x}_n + \sum_{n=1}^{N}\left(\tilde{x}_n^{(t)}\right)^2 \le v, \tag{24b}$$

$$\text{(11b), (11c), (11d) and (21b),} \tag{24c}$$

which can be effectively solved by the CVX toolbox. By iteratively solving problem $\mathcal{P}6$ until algorithm converges, problem $\mathcal{P}5$ can be handled. Due to the conservative approximation of (23) in each iteration, the achieved solution is feasible to problem $\mathcal{P}5$ and the obtained objective value is an upper bound of problem $\mathcal{P}5$. Furthermore, the achieved solution can also converge to the KKT point of problem $\mathcal{P}5$ according to [15].

C. Joint Cache Placement and Content Updating Algorithm

To summarize, problem $\mathcal{P}1$ is effectively handled by alternatively optimizing the content updating interval and the cache placement. The detailed algorithm is listed as Algorithm 1.

Algorithm 1 Joint Cache Placement and Content Updating Algorithm

Initialize variable $\tilde{\boldsymbol{x}} = \mathbf{1}$;
while The stopping criteria of outer loop is unsatisfied **do**
  Set $T_n = \sqrt{\frac{b_n + \alpha\tilde{x}_n}{a_n}}, \forall n$;
  if $\alpha = 0$ and $\sum_{n=1}^{N}\tilde{x}_n\frac{1}{T_n} \le F$ **then**
    $T_n = \sqrt{\frac{b_n}{a_n}}, \forall n$;
  else
    Find optimal α satisfying $\sum_{n=1}^{N}\tilde{x}_n\frac{1}{T_n} = F$ via the bisection method;
  end if
  Initialize variable $\tilde{\boldsymbol{x}}^{(0)} = \tilde{\boldsymbol{x}}$ and penalty factor ρ;
  while The stopping criteria of inner loop is unsatisfied **do**
    Solve problem $\mathcal{P}6$ via CVX to obtain $\tilde{\boldsymbol{x}}^{(n)}$;
    Set $n := n + 1$;
  end while
end while
return $\tilde{\boldsymbol{x}}$ and $\{T_n\}$.

IV. Simulation Results

In this section, simulation results are shown to verify the effectiveness of the proposed algorithm. In the simulation, we consider scenario where $N = 100$ sensor nodes are deployed within the coverage of BS. As each sensor node generates different content, the number of contents is also $N = 100$. Each content size follows Poisson distribution with parameter $\bar{s} = 0.5$ MB. The cache capacity of BS is $S = 10$ MB, which can store about 20 contents. Furthermore, the file popularity p_n follows Zipf distribution, i.e., $p_n = \frac{1/n^\beta}{\sum_{i=1}^{N} 1/i^\beta}$, where β represents the concentration of the file popularity setting as $\beta = 1.2$ in this paper. The total user requests follow Poisson process and its arrival rate is $\Lambda = 1000$. Besides, the average transmission rate of BS is $r_b = 350$ Mb/s. The transmission rate of sensor node n is generated from Poisson distribution with parameter $r_n^s = 8$ Mb/s. The energy consumption of sensor nodes is set as $\epsilon_n = 0.1$ (unit/Mb) and the weighting factor is $w = 1$. In the simulation, the proposed Algorithm 1 is labeled as 'Proposed JCPCU'. For comparison, three other benchmarks are also simulated, i.e., 'Greedy caching', 'Fixed content updating (CU) caching' and 'Most popular caching'. In the greedy caching, problem $\mathcal{P}3$ is solved by choosing a content providing the minimal objective value each time. In the fixed CU caching, the cache placement problem $\mathcal{P}3$ is directly solved with fixed content updating interval $T_n = \sqrt{b_n/a_n}, \forall n$. In the most popular caching, the content is cached based on the file popularity.

Fig. 2 shows the impact of the maximal content refresh frequency on the weighted sum of AoI and energy consumption. It can be seen from Fig. 2 that, with the increase of the maximal content refresh frequency, the considered weighted sum of AoI and energy consumption decreases and the network performance improves. Furthermore, the proposed JCPCU algorithm outperforms other benchmarks, which verifies the effectiveness of the proposed JCPCU algorithm. Moreover, with the increase of content refresh frequency, the performance gap between JCPCU and fixed CU caching gradually becomes narrow. This is because that, with larger content refresh frequency, the content refresh frequency constraint no longer restricts the network performance and the optimal solution to problem $\mathcal{P}2$ is same with the fixed CU caching scheme.

To offer the tradeoff, Fig. 3 and Fig. 4 respectively show the corresponding AoI and energy consumption. In Fig. 3 and Fig. 4, with the increase of content refresh frequency, the

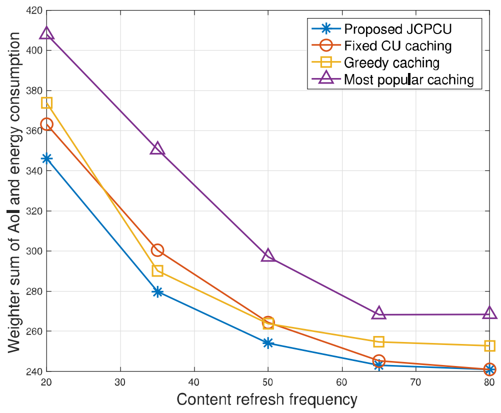

Fig. 2. The impact of F on the network performance.

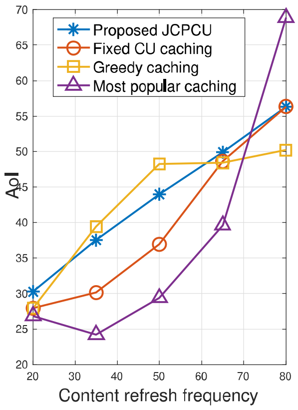

Fig. 3. AoI vs F.

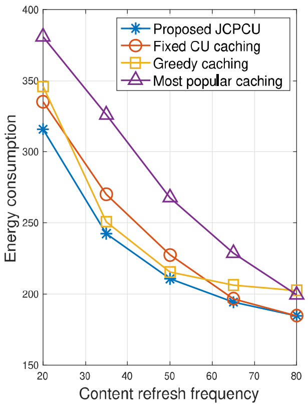

Fig. 4. Energy consumption vs F.

AoI becomes higher while the energy consumption of sensor nodes significantly reduced. The reason is that, larger content refresh frequency at BS allows more contents to be cached in the BS. With more cached contents, both the average content caching time and delivery time become longer, leading to higher AoI. However, the number of contents directly fetched from sensor nodes is greatly reduced, resulting in lower energy consumption of sensor nodes.

V. Conclusion

In this paper, we have jointly designed the cache placement and content updating interval to minimize the weighted AoI of contents and energy consumption of sensor nodes. The formulated MINLP problem has been solved by alternatively optimizing the cache placement and the content updating interval. A local optimal solution to the cache placement problem and a semi-closed form optimal solution to the content updating problem are respectively proposed. Simulation results show that our proposed JCPCU algorithm significantly outperforms other benchmarks. In the future, the cooperative IoT caching between multiple BSs considering the content freshness will be designed.

VI. Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1807700, in part by the National Natural Science Foundation of China (NSFC) under Grant 62071356, and in part by the Chongqing Key Laboratory of Mobile Communications Technology under Grant cqupt-mct-202202.

References

[1] D. Niyato, D. I. Kim, P. Wang, and L. Song, "A novel caching mechanism for Internet of Things (IoT) sensing service with energy harvesting," in *2016 IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.

[2] M. Qin, N. Cheng, Z. Jing, T. Yang, W. Xu, Q. Yang, and R. R. Rao, "Service-oriented energy-latency tradeoff for IoT task partial offloading in mec-enhanced multi-RAT networks," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1896–1907, Feb. 2021.

[3] R. Sun, Y. Wang, N. Cheng, L. Lyu, S. Zhang, H. Zhou, and X. Shen, "QoE-driven transmission-aware cache placement and cooperative beamforming design in cloud-RANs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 636–650, Jan. 2019.

[4] R. Sun, Y. Wang, L. Lyu, N. Cheng, S. Zhang, T. Yang, and X. Shen, "Delay-oriented caching strategies in D2D mobile networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8529–8541, Aug. 2020.

[5] Z. Yin, M. Jia, W. Wang, N. Cheng, F. Lyu, and X. Shen, "Max-min secrecy rate for noma-based uav-assisted communications with protected zone," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[6] B. Chen, L. Liu, M. Sun, and H. Ma, "IoTCache: Toward data-driven network caching for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10 064–10 076, Dec. 2019.

[7] X. Sun and N. Ansari, "Dynamic resource caching in the IoT application layer for smart cities," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 606–613, Apr. 2017.

[8] J. Yao and N. Ansari, "Caching in dynamic IoT networks by deep reinforcement learning," *IEEE Internet Things J.*, early access, 2020.

[9] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *2011 Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Salt Lake City, UT, USA, Jun. 2011, pp. 350–358.

[10] Y. Hui, Z. Su, and T. H. Luan, "Collaborative content delivery in software-defined heterogeneous vehicular networks," *IEEE/ACM Trans. Networking*, vol. 28, no. 2, pp. 575–587, Feb. 2020.

[11] H. Zhu, Y. Cao, X. Wei, W. Wang, T. Jiang, and S. Jin, "Caching transient data for Internet of Things: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2074–2083, Apr. 2019.

[12] S. Zhang, L. Wang, H. Luo, X. Ma, and S. Zhou, "AoI-delay tradeoff in mobile edge caching with freshness-aware content refreshing," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5329–5342, Aug. 2021.

[13] J. Zhong, R. D. Yates, and E. Soljanin, "Two freshness metrics for local cache refresh," in *2018 IEEE International Symposium on Information Theory (ISIT)*, Vail, CO, USA, June 2018, pp. 1924–1928.

[14] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, pp. 5481–5495, Oct. 2014.

[15] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, Oct. 2010.