Interference-Aware Task Assignment in Edge Cloud-Enhanced 5G Fiber-Wireless Access Networks

Pengchao Han[1]**,Yejun Liu**[2]**, Lei Guo**[2]

[1]*School of Computer Science and Engineering, Northeastern University, P. R. China*
[2]*School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, P. R. China*
Corresponding email: yjliu@cqupt.edu.cn (Y. Liu)

Abstract: This paper focuses on the task assignment problem with delay constraints in edge cloud-enhanced 5G fiber-wireless access networks, and proposes an interference-aware algorithm to minimize resource cost. Simulation results verify the effectiveness of proposed algorithm.

OCIS codes: 060.4256 (Networks, network optimization), 060.4258 (Networks, network topology).

1. Introduction

In recent years, mobile edge computing has become a promising technology for task offloading, alleviating computation pressure of mobile devices to facilitate emerging applications of artificial intelligence. Edge-cloud enhanced Fiber-Wireless (FiWi) access networks [1, 2] have gained growing popularity for providing services to end users with low latency leveraging widely deployed infrastructures in 5G. However, newly arrived tasks impact the delays of current tasks on the shared physical links by incurring longer queuing delay with additional load due to the Time-Division Multiple Access (TDMA) strategy in both optical and wireless subnetworks in FiWi. Related task assignment approaches with delay guarantee leave the interference among tasks untouched, especially for the delay characteristics of FiWi. Moreover, the ant colony optimization (ACO) is an effective method to find the near-optimal solutions for task assignment by considering exploiting historic experiences and learning from the environment [3]. However, conventional ACO applies on fixed search graph, requiring a large amount of ants and iterations to find a good solution.

In this paper, we highlight the task assignment problem in edge cloud-enhanced 5G FiWi access networks to guarantee isolation among tasks. We transform the delay constraints of tasks into the Interference-Aware (IA) residual bandwidth capacity of physical paths, based on which, a novel task assignment algorithm based on ACO with dynamic graph pruning is proposed to minimize resource consumption. To the best of our knowledge, this is the first work investigating interference among tasks in edge cloud-enhanced FiWi access networks for 5G.

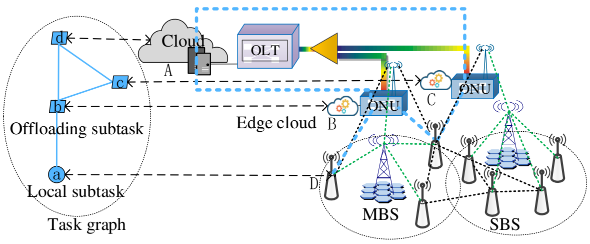

Fig. 1: Network models and task assignment.

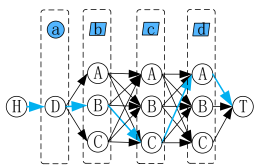

Fig. 2: ACO search graph.

2. Problem Description

2.1. Network Models and Task Assignment

The 5G FiWi access networks G^S consists node set $N^S = \{N^T, N^C\}$ and link set $L^S = \{L^O, L^W\}$ as shown in Fig. 1 (right), where N^T is the set of transmitting nodes consisting of Optical Line Terminal (OLT), Optical Network Units (ONU), Macro cell Base Stations (MBS) and Small cell Base Stations (SBS), N^C is the set of clouds including a central cloud connected to OLT and multiple edge clouds located at ONUs and L^O and L^W are the sets of optical and wireless links respectively. Note that MBSs are responsible for exchanging control signals using LTE frequency, thus we focus on SBSs in this paper. The computation capacity of cloud node $n \in N^C$ is p_n and the bandwidth capacity of link $l \in L^S$ is b_l. Any task to be assigned to G^S contains multiple inter-connected subtasks as shown in Fig. 1 (left). One of the subtasks is specified to be executed locally in the nearest accessible SBS and other offloading subtasks should be offloaded to clouds. The computation resource requirements of subtask i is u_i and the bandwidth requirement and the end-to-end delay constraint of task link j are r_j and d_j, respectively. The load of physical link l, i.e., f_l, is the sum of bandwidth requirements of task links that l serves.

The task assignment problem is to assign offloading subtasks to clouds, mapping task links to loop-free paths between the clouds that their end subtasks are assigned and allocate requested resources with satisfied delay in the objective of minimizing total resource cost, which is the sum of computation resources provided by selected clouds and the bandwidth provided by links along mapped paths. We define the reward for accepting a task as the total amount of required resources (i.e, $\sum_i u_i + \sum_j r_j$). Since reward of a given task is fixed, our objective can also be maximizing the ratio of reward to cost (R/C).

2.2. *Delay Models*

Different physical links have different delays influenced by link types, operating protocols and transmitting power etc. Generally, the packet transmitting process can be modeled as a queuing system with one server, which features packet arrival rate λ, service rate μ and load factor $\rho = \lambda/\mu$. Denote the second moment of service time for each packet by $\overline{X^2}$. For optical subnetwork which is Ethernet Passive Optical Network (EPON) with Multi-Point Control Protocol (MPCP), all the optical links share the same average delay [4]. Thus, for any optical link l, $\lambda_l = \sum_{l \in L^O} f_l/F$ and $\mu_l = \sum_{l \in L_S} b_l/F$, where F is the packet size. Let $\overline{V}$ and σ_V^2 be the mean and variance of the reservation time of MPCP, respectively. The mean queuing time of packets for EOPN with N ONUs is [4]:

$$t_q^O = \frac{\lambda_l \overline{X^2}}{2(1-\rho_l)} + \frac{(3N-\rho_l)\overline{V}}{2(1-\rho_l)} + \frac{\sigma_V^2}{2\overline{V}}. \tag{1}$$

Thus, the total delay of any optical link is $t^O = t_q^O + t_p^O$ where t_p^O is the average propagation time of packets.

For wireless subnetwork, different SBSs work in Frequency Division Multiplexing (FDM) and all tasks assigned to one link share its bandwidth in TDMA. Denote B the link bandwidth in Hz. Let W be the transmitting power of SBSs. The Signal-to-Interference-plus-Noise Ratio (SINR) of link l is $SINR_l = \phi_l W/(I_l N_0 B)$ where ϕ_l is the path loss, $I_l = \sum_{l' \in L^W, l' \neq l} \phi_{l'}$ is the total interference that l receives from other links and N_0 is the power of white noise. According to Shannon's formula, the bandwidth capacity of l is $b_l = B\log_2(1 + SINR_l)$ in bps. Based on above, the delay of each wireless link can be modeled as a M/G/1 queuing model since packets arrive in Poisson process and the service time follows an arbitrary distribution. The arrival and service rates are $\lambda_l = f_l/F$ and $\mu_l = b_l/F$ respectively. Thus, queuing delay of wireless link l according to queuing theory is $t_q^W = \lambda_l \overline{X^2}/(2(1-\rho_l))$. Therefore, the total delay of a wireless link is $t^W = t_q^W + t_p^W$ where t_p^W is the average propagation times of packets. Based on above, the actual delay of any task link j, denoted by τ_j, is the sum of delays of links it passes.

3. Interference-Aware Task Assignment Algorithm based on ACO and Dynamic Pruning

3.1. *Interference-Aware Residual Capacity*

Equation (1) and t_q^W indicate that the link delay is influenced by λ and thus the link load since μ remains unchanged for given bandwidth capacity. In this sense, tasks interfere with each other by changing loads of links. Intuitively, we can transform delay constraint of a task link into the residual capacity of the path it maps. Note that the shortest paths are always used for mapping task links to improve the probability of delay satisfaction. We then formulate the IA residual capacity of physical links and paths for serving future tasks without violating the delay constraints of existing tasks in the system. Assume the set of task links that l accepts is J_l, the delay constraint of l is the maximum delay that l can engender while guaranteeing delay constraints of all task links in J_l, which is given by $\varepsilon_l = \min\{d_j - \tau_j, \forall j \in J_l\} + t_l$ where t_l is either t^O or t^W for link l. By replacing t^O (or t^W) and λ_l in delay expressions with ε_l and ψ_l/F respectively, the IA residual bandwidth capacity of l, can be obtained as (2):

$$\psi_l = \begin{cases} \mu_l F\left(2\varepsilon_l - 2t_p^O - \sigma_V^2/\overline{V} - 3N\overline{V}\right) / \left(\mu_l \overline{X^2} - \overline{V} - \sigma_V^2/\overline{V} + 2\varepsilon_l - 2t_p^O\right), & if\ l \text{ is optical link,} \\ 2\mu_l F\left(\varepsilon_l - t_p^W\right) / \left(\mu_l \overline{X^2} + 2\varepsilon_l - 2t_p^W\right), & if\ l \text{ is wireless link.} \end{cases} \tag{2}$$

The delay constraint of a physical path a, i.e., ε_a, is defined as the minimum delay of task links strictly mapped on the path. Note that task links that pass through partial links in a are not included. Based on ε_a, we can calculate the additional available load of a using binary search method through trying to add possible additional load to all links in a, calculating expected delay and comparing with ε_a in an iterative way. We capture this process using function $\text{binarySearch}(a, \varepsilon_a)$, which returns the additional load value. Overall, the IA residual capacity of path a is the minimum of the additional available load and the minimum (IA) residual capacity of links along the path:

$$\psi_a = \min\{\text{binarySearch}(a, \varepsilon_a), \min\{\min\{\psi_l, b_l - f_l\}, \forall l \in a\}\}. \tag{3}$$

3.2. *ACO with Dynamic Pruning*

The ACO is utilized to assign tasks to minimize resource cost, during which process the IA residual capacity of paths are taking into account to prune search graph. The whole algorithm is as follows.

1) Find the candidate cloud nodes with satisfied computation resources for each offloading subtask;
2) Construct ACO search graph by adding two auxiliary nodes H and T and fully connecting candidate nodes for adjacent subtasks, as shown in Fig. 2 which is the search graph of the task and networks examples in Fig. 1. The weights of links in the graph are set to be the hop lengths of corresponding physical paths and 1 for the links of H and T. The objective of ACO is to find a feasible path from H to T with minimum total weights, such that subtasks are assigned to the nodes on the path;

3) Prune the graph by removing nodes and links that violate task requirements to avoid interfering existing tasks every time an ant moves forward.

Specifically for step 3), let $i.z$ be the candidate node z of subtask i, every time an ant moves a step (e.g., from $b.B$ to $c.C$ in Fig. 2), two conditions are checked on residual graph, i.e., every candidate nodes of d and their links: (1) If the candidate nodes satisfy computation requirements of corresponding subtasks? (2) Are the IA residual capacity of paths large enough for corresponding task links? It should be noted that for the second condition, all paths between assigned nodes of already assigned subtasks and all the candidate nodes of the ongoing to be assigned subtask should be checked (if task link exists). For example, assume an ant is currently locating at $c.C$, all possible mapping paths for links $< b,d >$ and $< c,d >$ should be checked, that is, paths $\{B,A\},\{B,B\},\{B,C\}$ for link $< b,d >$ and paths $\{C,A\},\{C,B\},\{C,C\}$ for $< c,d >$. If IA residual capacity of path $\{B,C\}$ violates the bandwidth requirement of $< b,d >$, then link $< c.C,d.C >$ will be removed. It is meaningful to mention that we allow multiple subtasks in a task to be assigned to the same cloud, and the residual bandwidth capacity within a (edge) cloud is set to be infinite. All ants should move according to the pruned graph and once a feasible path is found, the pheromone factors of links along the path will be updated for further search process according to [3].

4. Performance Evaluation

We evaluate our algorithm on a physical network with 1 OLT, 4 ONUs and 49 SBSs over a $300*300m^2$ area.The computation capacity of central cloud and edge clouds are 1000 and 500 respectively. The total bandwidth capacity of optical links is 1Gbps. Parameters related with wireless subnetwork are $P = 23dBm, N_0 = -174dBm/Hz$ and $B = 200MHz$, the pass loss of wireless links over millimeter wave is $157.4 + 32\log(dis)$ where dis is the link length in km. Parameters of F and $\overline{X^2}$ are 64bytes and $51.468(\mu s)^2$ for optical subnetwork and 20bytes and $1.137(\mu s)^2$ for wireless subnetwork. For MPCP, $\overline{V} = 1.512\mu s$ and $\delta_V^2 = 0$. Subtasks in each task are linked with the probability of 0.5. The computation requirements of offloading subtasks are [5,10] in uniform distribution and the bandwidth requirements and delay constraints of task links are [5,10]Mbps and [0.5,1]ms in uniform distributions. The duration of tasks follows exponential distribution with the mean of 10 time units. We use 2 ants for total 8 iterations for each task assignment which is vary much small and thus terminates fast. Parameters related with ACO is the same as [3]. We compare our algorithm (referred as IA_ACO) with ACO without dynamic pruning and conventional two stage method [5]. Fig. 3 illustrates the performance comparison under different task arrival rates when the average task size (i.e., number of subtasks in a task) is uniformly in [2,6]. Fig. 4 depicts the comparison over varying average size of tasks when the arrival rate is 2 tasks per time unit. Apparently the proposed IA_ACO outperforms Two_stage in both acceptance and R/C ratios. Higher acceptance ratio with comparable R/C ratio is also shown in IA-ACO compared with pure ACO.

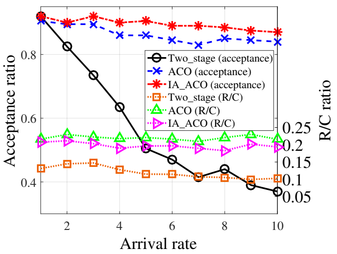

Fig. 3: Performance of different task arrival rates.

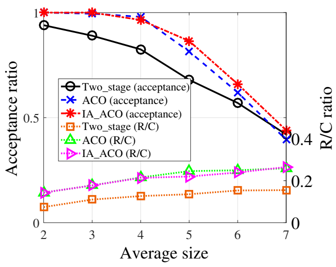

Fig. 4: Performance of different average task sizes.

5. Conclusion

In this paper, we addressed the interference among tasks in edge cloud-enhanced FiWi access network and converted delay requirements of task links into interference-aware residual capacity of physical paths. The delay of already assigned tasks can be never violated through dynamic graph pruning in our proposed algorithm based on ACO. Numerical results confirmed that our proposed method performs better in both acceptance and R/C ratios.

6. Acknowledgement

The works in this paper are supported by the National Natural Science Foundation of China (61775033) and Fundamental Research Funds for the Central Universities (N161608001).

References

1. B. P. Rimal, D. Pham Van, and M. Maier,"Cloudlet enhanced fiber-wireless access networks for mobile-edge computing," IEEE Transactions on Wirel. Commun., 3601-3618 (2017).
2. G. Kalfas, C. Vagionas, A. Antonopoulos, E. Kartsakli, A. Mesodiakaki, S. Papaioannou, P. Maniotis, J. S.Vardakas, C. Verikoukis, and N. Pleros,"Next generation fiber-wireless fronthaul for 5G mmwave net-works," IEEE Commun. Mag., 138-144 (2019).
3. S. Bharati and P. Saengudomler,"Analysis of mean packet delay for dynamic bandwidth allocation algo-rithms in EPONs," J. Light. Technol., 3454-3462 (2010).
4. I. Fajjari, N. Aitsaadi, G. Pujolle, and H. Zimmermann,"VNE-AC: Virtual network embedding algorithmbased on ant colony meta-heuristic," in *IEEE ICC*, (2011), pp. 16.
5. M. Yu, Y. Yi, J. Rexford, and M. Chiang,"Rethinking virtual network embedding: Substrate support for path splitting and migration," SIGCOMM Comput. Commun. Rev., 17-29 (2008).