Edge-cloud Collaborative Heterogeneous Task Scheduling in Multilayer Elastic Optical Networks

Zeyuan Yang*, Rentao Gu*, Zuqing Zhu†, Yuefeng Ji*
*Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, China
†School of Information Science and Technology, University of Science and Technology of China, Hefei, China
Email: *{buptyzy, rentaogu, jyf}@bupt.edu.cn, †zqzhu@ieee.org

***Abstract*—With the explosive growth of edge applications in the 5G/B5G era, edge-cloud collaboration (ECC) is playing a prominent role in edge service provisioning. For highly diversified edge-cloud collaborative services (ECSs), the joint allocation of heterogeneous computing resources in heterogeneous servers and multi-dimensional underlying optical network resources should be conducted. In this paper, we investigate the heterogeneous task scheduling for ECSs over multilayer elastic optical network (ML-EON), which involves the joint allocation of heterogeneous computing resources in edge and cloud servers and high-dimensional network resources. We propose a Task-Node Matching Score (TNMS) based method, which evaluates the fitness for each mapping tuple between each task in ECS and each substrate node in ML-EON, and adaptively generates a specific matching score for each task-node pair. Furthermore, TNMS is extended with a pre-allocation mechanism (TNMS-Pre) to estimate the costs of multi-dimensional resources in ML-EON for virtual link (VL) mapping. The estimated VL mapping costs are integrated into the matching scores to guide the task placement to be cost-efficient. To guarantee the feasibility, a maximal weight matching (MWM) based method is presented to determine the task placement schemes. Simulation results demonstrate the effectiveness of the adaptive scoring for heterogeneous task placement and the pre-allocation mechanism for reducing the ML-EON costs.**

***Index Terms*—Multi-access Edge Computing, Elastic Optical Network, Multilayer Network, Heterogeneous Task Placement.**

I. Introduction

Faced with the proliferation of edge applications, including industrial IoT, smart cities, and VR/AR, multi-access edge computing (MEC) is becoming a promising service paradigm to ease the computational intensity at end devices and reduce data transmission burden to the cloud [1]. It is reported that by 2023, 50% of the newly deployed on-premises infrastructures will be in edge locations instead of corporate datacenters [2].

Considering the capital expenditures (CAPEX) in edge infrastructure construction, edge servers often own fewer computing capacities than cloud servers. To cope with such computing capacity limitation at the edge, edge-cloud collaboration (ECC) becomes a necessity to better handle the services [3]: Edge-cloud collaborative services (ECSs) are split into several heterogeneous computing tasks, where the latency-sensitive computing tasks are offloaded to edge servers, while the computing tasks that are latency-insensitive and computational-intensive are served by cloud datacenters (DC). The heterogeneities of the tasks in ECS are reflected in the requirements for heterogeneous computing resources (CPU, GPU, FPGA, storage, etc.) [4] and heterogeneous servers (edge or cloud server) [3].

Recent works for task placement in ECC scenarios have focused on *i)* computing job partition [5], *ii)* hierarchical functional division [6] [7], and *iii)* resource procurement and auction [8]. In the aspect of resource allocation for data exchanges among the tasks in ECS, which are represented as virtual links (VLs) in services, these works only consider the bandwidth-level allocation but neglect the resource granularity and physical transmission effects in the underlying optical network. However, the underlying optical network, especially multilayer elastic optical network (ML-EON) with traffic aggregation, fine-granular bandwidth, sliceable spectrum, and tunable modulation format, is of high-dimensional resources to be allocated [9]–[11], e.g., frequency slots (FSs), modulation format, and sliceable bandwidth variable transponders (SBVTs) [12] [13]. It is still a challenge to perform joint allocation of heterogeneous computing resources in heterogeneous servers and high-dimensional ML-EON resources.

Virtual optical network embedding (VONE) has been investigated to jointly allocate the optical network and computing resources [14]–[17]. Most of the existing works take the residual/required resource capacities of the substrate/virtual network as criteria to rank the substrate nodes/virtual nodes for embedding. However, two drawbacks exist in these methods for ECS provisioning in ML-EON: *i)* Faced with task heterogeneities, the static ranking for substrate nodes only considers the residual resources but does not integrate the differences among tasks. Thus, the adaptiveness to different ECS is insufficient. *ii)* The costs for high-dimensional optical network resources in ML-EON for VL mapping are neglected in the virtual node mapping process in these methods, which will cause the ECS to be blocked due to VL mapping failings.

From the above analyses, to better handle the heterogeneous task scheduling in ML-EON, the following challenges are of the essence but have not been fully investigated: *i)* for each specific ECS, the substrate nodes should be dynamically evaluated rather than statically evaluated, and *ii)* the potential cost of high-dimensional optical network resources for VL mapping should be considered in the task placement process. To tackle the above challenges, we propose a Task-Node Matching Score (TNMS) based method for heterogeneous computing task placements in the ECC environment over ML-EON. The main contributions of this paper are as follows:

- The proposed TNMS-based method computes a specific matching score that is tightly coupled with ECS for each task-node pair to achieve adaptiveness to different ECSs. In addition to the residual resources, the heterogeneous task adjacency relations in the ECS are integrated into the computation of the matching scores.
- We further extend the TNMS with a VL pre-allocation mechanism (TNMS-Pre) to integrate the mapping costs of VLs over ML-EON into the matching scores. TNMS-Pre estimate the fine-grained costs of VL mapping in ML-EON, including the costs of bandwidth (BW), FSs, and sub-transponders (STPs) in SBVTs, and then integrate them into the matching score to guide the task placements towards lower operational expenditures (OPEX).
- As the matching scores represent tendencies of task placements, we present a method based on maximal weight bipartite graph matching to determine the final task placement, where the matching scores are taken as the weights of the bipartite graph. The method can balance tendencies among tasks and enforce task placement to conform to basic constraints.

The rest of the paper is organized as follows. Section II describes the system model. TNMS and TNMS-Pre, are presented in Section III. The performance evaluations are performed in Section IV, and Section V summarizes the paper.

II. System Model

A. Substrate Network

The underlying ML-EON is modeled as a directed graph $\mathcal{G}^s = \{V, E\}$, with V and E represent the substrate node set and fiber link set, respectively. V is composed of an edge computing node set V_E, a cloud computing node set V_C, and a transmission-only node set V_T. The computing node set $V_c = V_E \cup V_C$, where nodes in V_E and V_C are edge servers and cloud servers, and $\mathcal{N}_s = |V_c|$ is the node number in V_c. Heterogeneous computing resources are equipped in each computing node $n^s \in V_c$, with $\mathcal{T}_c$ type of computing resources. The initial and residual capacity of k-th type of computing resource in node n^s are I_k^s and R_k^s, respectively. Location label $L^s \in \{edge, cloud\}$ is assigned to each computing node to indicate the role of this node in ML-EON. Each fiber link $e_{sr} \in E$ between node n^s and node n^r consists of two opposite directional fibers, each of which consists of S FSs, and the granularity of each FS is set as B GHz. The lightpath between source node n^r and end node n^s is denoted as P_{rs}.

As is depicted in Fig. 1, each node in ML-EON maintains an IP/OTN-over-EON architecture. The IP layer provides class-specific traffic engineering with multiprotocol label switching (MPLS) [9], while OTN layer supports large-granular broadband service transmissions with efficient multiplexing and switching [18]. We abstract the IP and OTN layers as the upper layer, which acts as the role of aggregating multiple low-bitrate VLs into high-speed traffic flows, for underlying EON. SBVTs with sub-transponders are equipped between IP/OTN router and reconfigurable optical add/drop multiplexer (ROADM) for electrical-to-optical conversion and multiple optical flows grooming [13]. After converting signals from electrical domain to optical domain with SBVT, ROADM is used for switching in FS granularity [19]. As is illustrated in Fig. 1, the VLs with the same source node and destination node are groomed together and transmit in the optical domain. The traffic will only go up to the upper layer at the destination node, while bypass the intermediate routers.

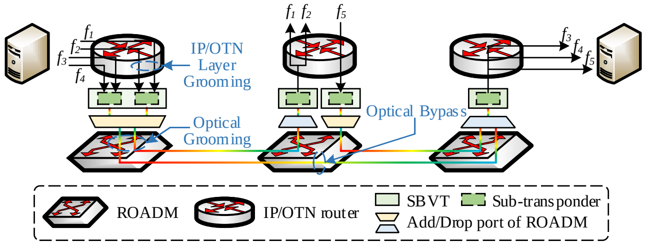

Fig. 1. Illustrations on traffic scheduling in multilayer EON.

B. Edge-cloud Collaborative Service

Given an ECS $\mathcal{G}^v = \{T, L\}$, $t^v \in T$ denotes the computing task and $l_{uv} \in L$ denotes the VL for data transmission from task t^u to task t^v. $\mathcal{N}_v = |T|$ is the task number in $\mathcal{G}^v$. C_k^v is the required capacity for k-th computing resource by task t^v. Each task also requires a specific location $L^v \in \{edge, cloud\}$. Each computing task t^v can be placed onto only one substrate computing node n^s which must satisfy the constraints of $L^s = L^v$ and $R_k^s \geq C_k^v$. The bandwidth of VL l_{uv} is denoted as b_{uv}. The dynamic ECS arrival follows the Poisson process with an average arrival rate of λ requests per time unit, and the lifetime of each ECS follows the negative exponential distribution with an average of $1/\mu$ time units.

III. Task-Node Matching Score-based Heterogeneous Task Placement

In polynomial time, the task placement with VL mapping in ML-EON can be reduced to the virtual network embedding (VNE) problem, which has been proved NP-hard [20]. Node ranking (NR) based methods, with moderate computational complexity, are typical methods for jointly allocating the computing and network resources for network-form requests [14]–[17] [21]. However, the NR-based methods face two drawbacks in heterogeneous task scheduling in ML-EON: *i)* **Lack of adaptiveness to different ECSs:** the ranks are statically evaluated, which are only associated with the status of substrate nodes, instead of being adaptively evaluated according to different ECS requirements. *ii)* **Unaware of VL mapping cost in VN mapping:** the VL mapping costs of high-dimensional optical network resources in ML-EON are not fully considered in the task mapping process. Such inherent drawbacks will mislead the heterogeneous task placement to sub-optimality and constraint violation. The misleading effects are illustrated in Fig. 2. The assumption is held that all substrate links are with the same and sufficient residual bandwidth. For **ECS 1**, if only check the residual computing

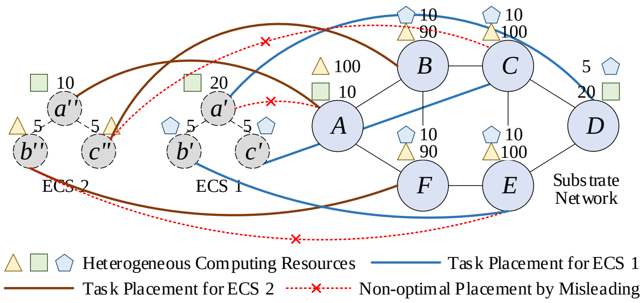

Fig. 2. Illustrations of *Misleading* effects when do not consider task heterogeneity and VL mapping cost in task placement.

resources in the substrate network, node A will rank higher than node D. However, node A is incapable of handling the task a' due to insufficient capacity of the second type computing resource. For **ECS 2**, if do not consider the VL mapping cost, the rank for node C and E are higher than node B and F. However, place tasks b'' and c'' onto nodes C and E will cause more bandwidth consumption. Therefore, it is proper to place tasks b'' and c'' onto the nodes B and F because shorter paths will be used for VL $a''b''$ and $a''c''$.

Thus, we propose the Task-Node Matching Score (TNMS) method for task placement to achieve high adaptiveness to ECSs, and then extend TNMS with pre-allocation (TNMS-Pre) to perform VL mapping cost-aware task placement.

A. Task-Node Matching Score (TNMS)

In TNMS, a specific matching score ρ_{sv} for each task-node pair (t^v, n^s) is adaptively computed according to the topology of ECS and the adjacency relations of the tasks. The following criteria lead to a higher ρ_{sv} include: *i) there are more computing resources in n^s that can be given to t^v; ii) ρ_{ru} is higher, where n^r is another substrate node and t^u is a neighbor of t^v; and iii) there are more bandwidth between n^s and n^r*. With the above criteria, the matching score ρ_{sv} for each task-node (t^v, n^s) pair is adaptively computed and tightly coupled with ECS. Since ρ_{sv} is associated with all other ρ_{ru}, it is formulated in a recursive form:

$$\rho_{sv} = \alpha \sum_k R_k^s / C_k^v + (1-\alpha) \sum_{r \neq s, u \in N(v)} m_{ru}^{sv} \cdot \rho_{ru}, \tag{1}$$

where R_k^s / C_k^v stands for computing ability of the k-th type of computing resource in n^s for t^v, $N(v)$ is the set of neighbor tasks of t^v, and $\alpha \in (0,1)$ is the balance coefficient. The

$$m_{ru}^{sv} = \begin{cases} b(s,r), & L^{s(r)} = L^{v(u)}, R_k^{s(r)} \geq C_k^{v(u)} \\ 0, & otherwise \end{cases} \tag{2}$$

is the residual bandwidth metric where $b(s,r)$ is the minimum bandwidth of links in the shortest path between n^s and n^r.

The parameters in (1) are vectorized as follows: $\boldsymbol{\rho} = (\rho_{11}, \cdots, \rho_{1\mathcal{N}_v}, \rho_{21}, \cdots, \rho_{\mathcal{N}_s\mathcal{N}_v})^T \in \mathbb{R}^{\mathcal{N}_s\mathcal{N}_v \times 1}$ represents the task-node matching score vector and $\mathbf{C} = \sum_k \left(\frac{R_k^1}{C_k^1}, \cdots, \frac{R_k^1}{C_k^{\mathcal{N}_v}}, \frac{R_k^2}{C_k^1}, \cdots, \frac{R_k^{\mathcal{N}_s}}{C_k^{\mathcal{N}_v}} \right) \in \mathbb{R}^{\mathcal{N}_s\mathcal{N}_v \times 1}$ represents the computing ability vector. To facilitate the convergence of (1), m_{ru}^{sv} is normalized as:

$$\bar{m}_{ru}^{sv} = \frac{m_{ru}^{sv}}{\sum_{r=1}^{\mathcal{N}_s} \sum_{u=1}^{\mathcal{N}_v} m_{ru}^{sv}}. \tag{3}$$

And $\bar{m}$ is vectorized to the residual bandwidth metric matrix $\bar{\mathbf{M}} \in \mathbb{R}^{\mathcal{N}_s\mathcal{N}_v \times \mathcal{N}_s\mathcal{N}_v}$, which is defined as

$$\bar{\mathbf{M}} = \begin{pmatrix} \bar{m}_{11}^{11} & \cdots & \bar{m}_{1\mathcal{N}_v}^{11} & \bar{m}_{21}^{11} & \cdots & \bar{m}_{\mathcal{N}_s\mathcal{N}_v}^{11} \\ \vdots & & \vdots & \vdots & & \vdots \\ \bar{m}_{11}^{1\mathcal{N}_v} & \cdots & \bar{m}_{1\mathcal{N}_v}^{1\mathcal{N}_v} & \bar{m}_{21}^{1\mathcal{N}_v} & \cdots & \bar{m}_{\mathcal{N}_s\mathcal{N}_v}^{1\mathcal{N}_v} \\ \bar{m}_{11}^{21} & \cdots & \bar{m}_{1\mathcal{N}_v}^{21} & \bar{m}_{21}^{21} & \cdots & \bar{m}_{\mathcal{N}_s\mathcal{N}_v}^{21} \\ \vdots & & \vdots & \vdots & & \vdots \\ \bar{m}_{11}^{\mathcal{N}_s\mathcal{N}_v} & \cdots & \bar{m}_{1\mathcal{N}_v}^{\mathcal{N}_s\mathcal{N}_v} & \bar{m}_{21}^{\mathcal{N}_s\mathcal{N}_v} & \cdots & \bar{m}_{\mathcal{N}_s\mathcal{N}_v}^{\mathcal{N}_s\mathcal{N}_v} \end{pmatrix}. \tag{4}$$

Then, the vectorization form of (1) can be rewritten as:

$$\boldsymbol{\rho} = \alpha \mathbf{C} + (1-\alpha) \bar{\mathbf{M}} \cdot \boldsymbol{\rho}. \tag{5}$$

The existence of a unique solution for $\boldsymbol{\rho}$ is proved as follows.

Theorem 1. *The infinite-norm of $\bar{\mathbf{M}}$ satisfies $\left\|\bar{\mathbf{M}}\right\|_\infty = 1$.*

Proof. For any vector $\mathbf{x} \in \mathbb{R}^{\mathcal{N}_s\mathcal{N}_v \times 1}$ whose infinite-norm satisfies $\|\mathbf{x}\|_\infty = \max_i |x_i| = 1$, $\bar{\mathbf{M}} \cdot \mathbf{x} \in \mathbb{R}^{\mathcal{N}_s\mathcal{N}_v \times 1}$ and $\left\|\bar{\mathbf{M}} \cdot \mathbf{x}\right\|_\infty = \max_i \left| \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} \bar{m}_{ij} x_j \right| \leq \max_i \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij}| |x_j| \leq \max_i \sum_{j=1}^{\mathcal{N}_s N_v} |\bar{m}_{ij}|$. According to the definition of infinite-norm of matrix, we can conclude that $\left\|\bar{\mathbf{M}}\right\|_\infty \leq \max_i \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij}|$. Select a column k in $\bar{\mathbf{M}}$ that satisfies $\sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{kj}| = \max_i \sum_{j=1}^{\mathcal{N}_s N_v} |\bar{m}_{ij}|$ and set an auxiliary vector $\mathbf{y} = (y_1, \cdots, y_{\mathcal{N}_v})$, with

$$y_j = \begin{cases} 1, & \bar{m}_{kj} = 0 \\ |\bar{m}_{kj}| / \bar{m}_{kj}, & \bar{m}_{kj} \neq 0 \end{cases}. \tag{6}$$

Obviously, $\|\mathbf{y}\|_\infty = 1$ and $\bar{\mathbf{M}} \cdot \mathbf{y} = \left(\cdots, \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{kj}|, \cdots \right)$. Also, $\left\|\bar{\mathbf{M}}\right\|_\infty \geq \left\|\bar{\mathbf{M}} \cdot \mathbf{y}\right\|_\infty = \max_i \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij} \cdot y_j| \geq \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{kj}| = \max_i \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij}|$. From the above steps, $\left\|\bar{\mathbf{M}}\right\|_\infty = \max_i \sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij}|$. According to (3), $\sum_{j=1}^{\mathcal{N}_s\mathcal{N}_v} |\bar{m}_{ij}| \leq 1$, and thus, $\left\|\bar{\mathbf{M}}\right\|_\infty = 1$ is proved. ■

Theorem 2. $\mathbf{I} - (1-\alpha) \bar{\mathbf{M}}$ *is reversible.*

Proof. Suppose the matrix $\mathbf{I} - (1-\alpha) \bar{\mathbf{M}}$ is irreversible. Then $\det \left[\mathbf{I} - (1-\alpha) \bar{\mathbf{M}} \right] = 0$, and the linear homogeneous equation set $\left[\mathbf{I} - (1-\alpha) \bar{\mathbf{M}} \right] \cdot \mathbf{x} = 0$ has a non-zero solution $\mathbf{x}_0$. Therefore, there exists an $\mathbf{x}_0$, where $\|\mathbf{x}_0\|_\infty = \left\|(1-\alpha) \bar{\mathbf{M}} \cdot \mathbf{x}_0\right\|_\infty$. According to the Compatibility of Matrix, $\left\|(1-\alpha) \bar{\mathbf{M}} \cdot \mathbf{x}_0\right\|_\infty \leq |1-\alpha| \left\|\bar{\mathbf{M}}\right\|_\infty \|\mathbf{x}_0\|_\infty$. Because $1 - \alpha < 1$ and $\left\|\bar{\mathbf{M}}\right\|_\infty = 1$ (**Theorem 1**), we can conclude that $\|\mathbf{x}_0\|_\infty > \left\|(1-\alpha) \bar{\mathbf{M}} \cdot \mathbf{x}_0\right\|_\infty$, which will lead to a contradiction. Therefore, the matrix $\mathbf{I} - (1-\alpha) \bar{\mathbf{M}}$ is proved to be reversible. ■

With **Theorem 2**, the unique solution of $\boldsymbol{\rho}$ can be given by $\boldsymbol{\rho} = \alpha \left[\mathbf{I} - (1-\alpha) \bar{\mathbf{M}} \right]^{-1} \cdot \mathbf{C}$. When the network scale is large, the direct computation is infeasible. An iterative algorithm, which calculates the $\boldsymbol{\rho}$ with $\boldsymbol{\rho}_{k+1} = \alpha \mathbf{C} + (1-\alpha) \bar{\mathbf{M}} \cdot \boldsymbol{\rho}_k$ until $\|\boldsymbol{\rho}_{k+1} - \boldsymbol{\rho}_k\| < \sigma$ ($\sigma \ll 1$), are used to ease the computational complexity [21].

B. TNMS with Pre-allocation Mechanism (TNMS-Pre)

TNSM holds the idea of "*how to successfully allocate the ECS*". The paths that have more residual bandwidth are more likely to be used. However, to minimize the cost of ECS provisioning, the cost of VL mapping should be integrated into matching score computation. Nevertheless, the accurate and actual VL mapping cost cannot be explicitly evaluated for matching score before task placement because VL mapping can be conducted only after the source and end node are confirmed, which is associated with the task placement.

To address this dilemma, we estimate the transmission cost of VLs under each task placement scheme with a pre-allocation mechanism and integrate the estimated VL costs into matching scores computation. For each VL l_{uv}, we pre-allocate the l_{uv} with all possible task placement schemes. Denoted ζ_{ru}^{sv} as the cost of l_{uv} when task t^v and t^u are respectively placed onto the node n^s and n^r. Then, the cost corresponding to a specific scheme is embedded into (2) as:

$$m_{ru}^{sv} = \begin{cases} b\left(s,r\right) - \beta\zeta_{ru}^{sv}, & L^{s(r)} = L^{v(u)}, R_k^{s(r)} \geq C_k^{v(u)} \\ 0, & otherwise \end{cases}, \tag{7}$$

where β balances between the residual bandwidth and mapping cost. Eq. (7) indicates that *the task placement schemes* (t^v, n^s) *and* (t^u, n^r) *are preferred if the transmission cost between* n^s *and* n^r *for VL* l_{uv} *is low*. The cost ζ_{ru}^{sv} is computed as $\zeta_{ru}^{sv} = \mathbf{w} \cdot \mathbf{c}_{ru}^{sv}$, where $\mathbf{w} = \left(w^{BW}, w^{FS}, w^{STP}\right)$ and $\mathbf{c}_{ru}^{sv} = \left(c_{uv,rs}^{BW}, c_{uv,rs}^{FS}, c_{uv,rs}^{STP}\right)^T$ are the unit cost and usage of BW, FS, and STP when l_{uv} is mapped between n^s and n^r, respectively. The usages of BW, FS, and STP are estimated as follows: The VL l_{uv} firstly tries to aggregate with the existing lightpaths between n^s and n^r. If there exist a lightpath with sufficient bandwidth, $c_{uv,rs}^{FS} = c_{uv,rs}^{STP} = 0$, as no FS and STP are newly used. If the aggragation fails, a new lightpath should be established. The routing, modulation format, spectrum, and transponder assignment (RMSTA) algorithm, which hold the ideas of K-shortest path (KSP) for path selection and first-fit (FF) for FS, SBVT, and STP assignments, will be executed. $c_{uv,rs}^{FS}$ equals to the occupied FS number, and $c_{uv,rs}^{STP} = 2$ for STPs used in source node and end node. $c_{uv,rs}^{BW}$ always equals to the product of the bandwidth of l_{uv} and the number of links in the mapped lightpath. Noted that, such process is only the pre-allocation for l_{uv} to estimate the mapping cost, and the final VL mapping will be conducted after the task placement. The computing of $\boldsymbol{\rho}$ in TNMS-Pre is the same as III. A.

C. Task Placement with Auxiliary Bipartite Graph

For the task placement, we set the placement sequence as $\mathcal{S} = (s_1, \ldots, s_v, \ldots, s_{\mathcal{N}_v})$, with $s_v \in [1, \mathcal{N}_s]$ indicates that the computing task t^v is placed onto the substrate computing node n^{s_v}. Each s_v is selected with the maximum matching score $s_v = \arg_s \max \rho_{sv}$. However, multiple tasks may tend to be placed onto the same substrate node which will violate the basic constraint. To seek balance among these tasks, we set the objective as maximizing the sum of matching scores.

Algorithm 1: Task Placement with ABG.

```
Input: $\boldsymbol{\rho}$, computing node set $V_c$, task set $T$
Output: Placement sequence $\mathcal{S}$
Initialization: Construct a complete bipartite graph $\tilde{\mathcal{G}}$,
  with the computing node set $V_c$ and the task set $T$ as
  two node sets, and weight set $\tilde{\mathbf{w}} \leftarrow 0$;
for each $n^s \in V_c$ do
    for each $t^v \in T$ do
        if $L^s = L^v$ and $R_k^s \geq C_k^v$ then
            $\rho_{sv}^M \leftarrow 1$
        else
            $\rho_{sv}^M \leftarrow 0$
        $\rho'_{sv} = \rho_{sv} \cdot \rho_{sv}^M$
        $\tilde{w}_{sv} \leftarrow \rho'_{sv}$
$\mathcal{M} \leftarrow \mathrm{KM}\left(\tilde{\mathbf{w}}\right)$
Extracting $\mathcal{S}$ from $\mathcal{M}$
```

A method based on maximal weight bipartite graph matching is presented to obtain the placement sequence. Algorithm 1 shows the procedure of the task placement with bipartite graph matching. An Auxiliary Bipartite Graph (ABG) $\tilde{\mathcal{G}} = \left\{V_c, T, \tilde{E}\right\}$ is constructed, with computing node set V_c and task set T as two separate node sets of the graph. $\tilde{\mathcal{G}}$ is a complete bipartite graph, in which all the (t^v, n^s) pairs have a link $\tilde{e}_{sv} \in \tilde{E}$ between n^s and t^v. To satisfy the computing capacity constraint and location constraint, a mask matrix $\boldsymbol{\rho}^M$ is constructed to remove the infeasible task placement schemes, with each element

$$\rho_{sv}^M = \begin{cases} 1, & L^s = L^v, R_k^s \geq C_k^v \\ 0, & otherwise \end{cases}. \tag{8}$$

The weight vector for $\tilde{\mathcal{G}}$ is computed as $\boldsymbol{\rho'} = \boldsymbol{\rho} \odot \boldsymbol{\rho}^M$, where $\boldsymbol{\rho'} \in \mathbb{R}^{\mathcal{N}_s \mathcal{N}_v \times 1}$, and $\odot$ is the Hadamard Product of matrix. Each element ${\rho'}_{sv} \in \boldsymbol{\rho'}$ is the weight of the $\tilde{e}_{sv}$. With above settings, the objective of maximizing the sum of matching scores is equivalent to finding the maximal weight matching (MWM) of $\tilde{\mathcal{G}}$. It is intuitive that when finding an MWM $\mathcal{M} = \left\{\left(n^{\tilde{s}_1}, 1\right), \ldots, \left(n^{\tilde{s}_{\mathcal{N}_v}}, \mathcal{N}_v\right)\right\}$ satisfies that the sum of weights is maximized, we can let $\mathcal{S}$ equal to the substrate node sequence in $\mathcal{M}$ as $\mathcal{S} = (\tilde{s}_1, \ldots, \tilde{s}_v, \ldots, \tilde{s}_{\mathcal{N}_v})$, which maximize the sum of matching scores and satisfies

$$\mathcal{S} = \arg\max_{\mathcal{S}} \left\{{\rho'}_{s_1 1} + \ldots + {\rho'}_{s_v v} + \ldots + {\rho'}_{s_{\mathcal{N}_v} \mathcal{N}_v}\right\}. \tag{9}$$

To this end, the Kuhn-Munkres (KM) algorithm is adopted to compute the MWM in $\tilde{\mathcal{G}}$ and the placement sequence $\mathcal{S}$. After the task placement, the VL mapping follows the process for VL mapping in ζ_{ru}^{sv} computing in V. B. The total VL mapping cost ζ_{ECS} is computed as

$$\zeta_{ECS} = \sum_{u,v} \left(w^{BW} b_{uv} \left|P_{rs}\right| + w^{FS} c_{uv}^{FS} + w^{STP} c_{uv}^{STP}\right). \tag{10}$$

TABLE I
SIMULATION CONFIGURATIONS OF SUBSTRATE NETWORKS AND ECSS.

Parameters	NSFNET	CORONET
Number of substrate nodes/substrate links	14/22	75/99
Number of total computing nodes	14	45
Number of edge/cloud nodes	10/4	30/15
Types of computing resources	CPU, GPU, FPGA, storage	
Computing capacity in edge/cloud nodes	$[50, 100]$ / $[500, 1000]$ units	
Number of edge tasks in ECS	$[2, 5]$	$[4, 10]$
Number of cloud tasks in ECS	$[1, 3]$	$[2, 6]$
Computing requirement of edge tasks	$[1, 5]$ units	
Computing requirement of cloud tasks	$[10, 50]$ units	
Bandwidth requirement of VLs	$[10, 100]$ Gbps	

IV. PERFORMANCE EVALUATION

In this section, we perform experiments to demonstrate the effectiveness of the proposed methods in the aspects of blocking ratio, ECS capacity, and VL mapping costs.

A. Experimental Settings

The 14-node NSFNET and 75-node CORONET [22], are adopted as substrate networks to evaluate the methods. The configurations of substrate network are listed in Line 2-6 in Table I. The computing SNs are randomly appointed to guarantee the generality. In ML-EON, each fiber contains FSs with the number of $S = 200$, and the granularity of each FS is 12.5 GHz. The guard band is set as 1 FS [23]. The number of SBVTs in each node, the number of STPs in each SBVT, and the maximum FSs that an STP can carry are 20, 10, and 10, respectively. The modulation format of a lightpath, including BPSK, QPSK, 8-QAM, and 16-QAM, is determined in terms of the transparent transmission distance of the lightpath [23]. The configurations of tasks and VLs in ECS are listed in Line 7-11 in Table I. When generating ECS, a spanning tree is randomly constructed to guarantee the connectivity of ECS, and other edges are connected with probability of $\tau = 0.2$. Parameters in TNMS and TNMS-Pre are determined by grid search and are fixed as $\{\alpha, \beta\} = \{0.1, 0.2\}$. $w^{BW} = 0.4$ [24], and w^{FS} and w^{STP} are set as 1. All simulations are performed with 2.60 GHz Intel Xeon-6240 CPU and 16 GB RAM.

B. Performance Comparisons

We adopted two heuristics as benchmarks: GRC [21] and TMR [15]. Since the benchmarks do not consider the task heterogeneities, they are modified as follows: after getting the node ranks, for each task t^v, sort the nodes that have the same type of location and sufficient computing resources with t^v in a descending order. Then place the t^v onto the first node, and delete the node to avoid overlapping tasks placement.

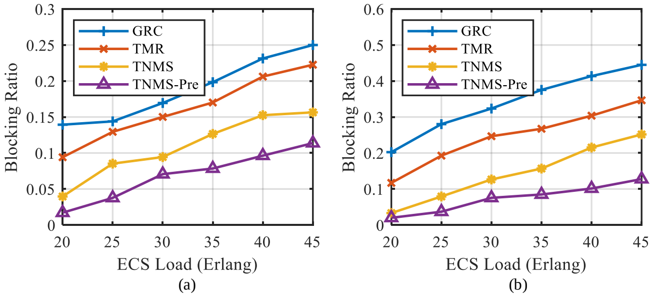

Fig. 3. The blocking ratios of TNMS, TNMS-Pre and benchmarks under (a) NSFNET topology and (b) CORONET topology.

TABLE II
ECS CAPACITY UNDER DIFFERENT NETWORK TOPOLOGIES.

	NSFNET ($\mathcal{C}_{\text{ECS}} \pm std$)	**CORONET** ($\mathcal{C}_{\text{ECS}} \pm std$)
GRC	75.2 ± 8.5	48.9 ± 4.2
TMR	82.7 ± 8.7	51.7 ± 3.6
TNMS	116.6 ± 8.2	60.9 ± 4.7
TNMS-Pre	142.3 ± 11.8	72.8 ± 5.6

The blocking ratios (BRs) of the proposed methods and benchmarks are evaluated in Fig. 3. BR measures the ratio between the number of blocked ECSs to the total number of ECSs. Under the light traffic load scenario, the BRs are similar among the four methods. However, with the increase of ECS load, the BRs of the proposed methods, TNMS and TNMS-Pre, outperform the benchmarks in both of the topologies. Furthermore, the TNMS-Pre which integrates the ML-EON transmission costs in to task placements is superior to other methods in reducing the number of the blocked ECS requests.

We further evaluate the maximum ECS number (ECS capacity $\mathcal{C}_{\text{ECS}}$) that the substrate networks can support. With a higher ECS capacity, the network resource utilization will be higher and the BRs will be reduced. We repeat the experiments ten times and compute the average $\mathcal{C}_{\text{ECS}}$ and the standard deviation (std). As is listed in Table II, in both topologies, $\mathcal{C}_{\text{ECS}}$ under the GRC and TMR are smaller compared with the proposed methods. It is mainly because the benchmarks do not adaptively evaluate the task placement schemes according to the requirements of the heterogeneous tasks and do not consider the VL mapping cost in the task placement process. Furthermore, the ECS capacity of TNMS-Pre is larger than that of TNMS because it tends to map the VLs with fewer costs and thus, occupies fewer network resources.

It is also worthy to examine the cost of ECS with different task placement methods. The average costs of VL mapping in each ECS over ML-EON are depicted in Fig. 4. With the increase of ECS load, decreasing trends exist in all conditions. This is mainly because that with more ECSs remain in the network, the services are more compact in the substrate network. Among four methods, the decreasing trend of TNMS-Pre is flatter because no matter what the underlying ML-EON status is, it always tends to minimize the VL mapping cost. It can also be observed that TNMS-Pre obtains the least cost among the methods, which demonstrates the effectiveness of the pre-allocation mechanism and integration of the VL mapping cost into task placement in TNMS-Pre.

C. Computation Complexity Evaluations

The time-efficiencies and scalabilities of TNMS and TNMS-Pre in different topologies are evaluated in Fig. 5. The TNMS achieves average computing times less than 18ms and 65ms for an ECS in NSFNET and CORONET topologies. For TNMS-Pre, although a longer computing time is caused by the pre-

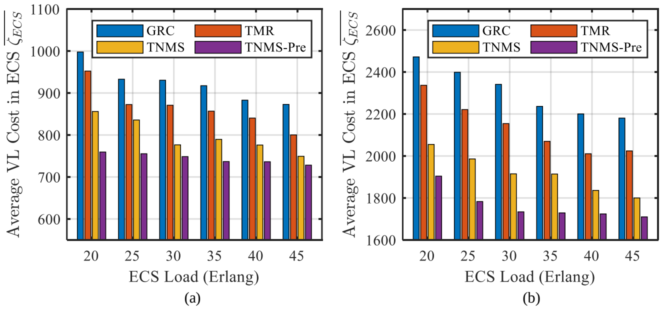

Fig. 4. The average cost of a single ECS with TNMS, TNMS-Pre and benchmarks under (a) NSFNET topology and (b) CORONET topology.

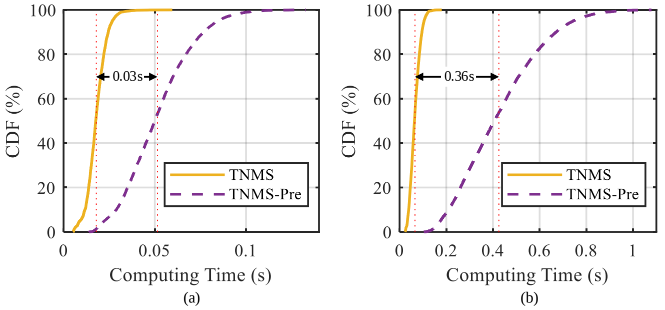

Fig. 5. The cumulative distribution function (CDF) of the computing time for a single ECS under (a) NSFNET topology and (b) CORONET topology.

allocation process of VLs, the average increased computing times in the two networks are within 0.03s and 0.36s, respectively.

V. Conclusion

In this paper, we investigate the heterogeneous task placement in the edge-cloud collaboration scenario over ML-EON. TNMS is proposed to adaptively evaluate the task placement scheme for each specific ECS request. To further reduce the ECS cost of VL mapping in ML-EON, TNMS-Pre is advocated with a pre-allocation mechanism to integrate the cost of VL mapping into the task placement process. The proposed methods outperform benchmarks in blocking ratio, ECS capacity, and the average cost of ECSs. The computational efficiencies of the proposed methods are also demonstrated.

Acknowledgment

This work is supported by the National Key Research and Development Program (2018YFB1800504) and National Natural Science Foundation of China (61971070), P. R. China, and BUPT Excellent Ph.D. Students Foundation (CX2020103).

References

[1] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, “On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration,” *IEEE Communications Surveys and Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.

[2] The Importance of Effective Operations in Unlocking Edge IT Value, IDC White Paper, 2020. Available Online: https://www.ibm.com/downloads/cas/0EM5DJZM.

[3] Y. Wu, “Cloud-edge orchestration for the Internet of Things: Architecture and AI-powered data processing,” *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12 792–12 805, 2021.

[4] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, “Edge intelligence: The confluence of edge computing and artificial intelligence,” *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020.

[5] X. Zhang, M. Qiao, L. Liu, Y. Xu, and W. Shi, “Collaborative cloud-edge computation for personalized driving behavior modeling,” in *Proceedings of ACM/IEEE Symposium on Edge Computing (SEC)*, 2019.

[6] L. Ruan, Y. Yan, S. Guo, F. Wen, and X. Qiu, “Priority-based residential energy management with collaborative edge and cloud computing,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1848–1857, 2020.

[7] R. Li, Z. Zhou, X. Chen, and Q. Ling, “Resource price-aware offloading for edge-cloud collaboration: A two-timescale online control approach,” *IEEE Transactions on Cloud Computing*, in press, 2019.

[8] T. Q. Dinh, B. Liang, T. Q. S. Quek, and H. Shin, “Online resource procurement and allocation in a hybrid edge-cloud computing system,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2137–2149, 2020.

[9] Q. Zhu, X. Yu, Y. Zhao, A. Nag, and J. Zhang, “Auxiliary-graph-based energy-efficient traffic grooming in IP-over-fixed/flex-grid optical networks,” *Journal of Lightwave Technology*, vol. 39, no. 10, pp. 3011–3024, 2021.

[10] R. Li, R. Gu, W. Jin, and Y. Ji, “Learning-based cognitive hitless spectrum defragmentation for dynamic provisioning in elastic optical networks,” *IEEE Communications Letters*, vol. 25, no. 5, pp. 1600–1604, 2021.

[11] R. Gu, Z. Yang, and Y. Ji, “Machine learning for intelligent optical networks: A comprehensive survey,” *Journal of Network and Computer Applications*, vol. 157, 2020.

[12] P. Lu, L. Zhang, X. Liu, J. Yao, and Z. Zhu, “Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks,” *IEEE Network*, vol. 29, no. 5, pp. 36–42, 2015.

[13] N. Sambo, P. Castoldi, A. D’Errico, E. Riccardi, A. Pagano, M. S. Moreolo, J. M. Fàbrega, D. Rafique, A. Napoli, S. Frigerio, E. H. Salas, G. Zervas, M. Nolle, J. K. Fischer, A. Lord, and J. P. F.-P. Giménez, “Next generation sliceable bandwidth variable transponders,” *IEEE Communications Magazine*, vol. 53, no. 2, pp. 163–171, 2015.

[14] L. Gong and Z. Zhu, “Virtual optical network embedding (VONE) over elastic optical networks,” *Journal of Lightwave Technology*, vol. 32, no. 3, pp. 450–460, 2014.

[15] W. Wei, H. Gu, K. Wang, X. Yu, and X. Liu, “Improving cloud-based IoT services through virtual network embedding in elastic optical inter-DC networks,” *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 986–996, 2019.

[16] M. Zhu, S. Zhang, Q. Sun, G. Li, B. Chen, and J. Gu, “Fragmentation-aware VONE in elastic optical networks,” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 9, pp. 809–822, 2018.

[17] Y. Zong, Y. Ou, A. Hammad, K. Kondepu, R. Nejabati, D. Simeonidou, Y. Liu, and L. Guo, “Location-aware energy efficient virtual network embedding in software-defined optical data center networks,” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 7, pp. 58–70, 2018.

[18] M. Song, J. Zhu, F. Zhou, and Z. Zhu, “On security-aware multilayer planning for IP-over-optical networks with OTN encryption,” in *IEEE International Conference on Communications (ICC)*, 2020.

[19] M. Jinno, “Elastic optical networking: Roles and benefits in beyond 100-Gb/s era,” *Journal of Lightwave Technology*, vol. 35, no. 5, pp. 1116–1124, 2017.

[20] A. Fischer, J. F. Botero, M. T. Beck, H. de Meer, and X. Hesselbach, “Virtual network embedding: A survey,” *IEEE Communications Surveys Tutorials*, vol. 15, no. 4, pp. 1888–1906, 2013.

[21] L. Gong, Y. Wen, Z. Zhu, and T. Lee, “Toward profit-seeking virtual network embedding algorithm via global resource capacity,” in *IEEE Conference on Computer Communications (INFOCOM)*, 2014.

[22] R. Gour, G. Ishigaki, J. Kong, and J. P. Jue, “Availability-guaranteed slice composition for service function chains in 5G transport networks,” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 13, no. 3, pp. 14–24, 2021.

[23] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, “Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing,” *Journal of Lightwave Technology*, vol. 31, no. 1, pp. 15–22, 2013.

[24] Amazon web services. Available Online: https://aws.amazon.com/cn/ec2/pricing/reserved-instances/pricing/.