Fog-Assisted Operational Cost Reduction for Cloud Data Centers

Liang Yu, Member, IEEE, Tao Jiang, Senior Member, IEEE, and Yulong Zou, Senior Member, IEEE

Abstract—In this paper, we intend to reduce the operational cost of cloud data centers with the help of fog devices, which can avoid the revenue loss due to wide-area network propagation delay and save network bandwidth cost by serving nearby cloud users. Since fog devices may not be owned by a cloud service provider, they should be compensated for serving the requests of cloud users. When taking economical compensation into consideration, the optimal number of requests processed locally by each fog device should be decided. As a result, existing load balancing schemes developed for cloud data centers can not be applied directly and it is very necessary to redesign a cost-ware load balancing algorithm for the fog-cloud system. To achieve the above aim, we first formulate a fog-assisted operational cost minimization problem for the cloud service provider. Then, we design a parallel and distributed load balancing algorithm with low computational complexity based on Proximal Jacobian Alternating Direction Method of Multipliers (PJ-ADMM). Finally, extensive simulation results show the effectiveness of the proposed algorithm.

Index Terms—Cloud computing, fog computing, operational cost, load balancing, parallel and distributed algorithm, Proximal Jacobian ADMM.

I. INTRODUCTION

Cloud computing is envisioned as an effective means of providing worldwide consumers with on-demand computing resources (e.g., networks, servers, storage, applications, and services) in a convenient way. Due to the advantages of high resource utilization, strong computing ability, high reliability, and rapid elasticity of cloud computing, many Internet workloads are processed in cloud data centers. According to a report in Cisco Global Cloud Index (2015-2020), 92 percent of all data center workloads are expected to be processed in cloud data centers by 2020 [1]. Since the operational cost of a cloud data center is very high (e.g., as a significant fraction of the operational cost, the annual energy cost related to many cloud service providers was larger than millions of dollars [2]), it is of great importance to reduce the operational cost for a cloud service provider.

There have been lots of schemes on reducing the operational cost/energy cost of cloud data centers [3]–[12], such as dynamic server provisioning, spatial/temporal load balancing, energy storage, incorporating renewable energies, partial execution, and participating in demand response programs of smart grids. In [3], Rao et al. proposed a geographical load balancing (GLB) scheme to minimize the energy cost of data centers in deregulated electricity markets. In [4], Ren et al. presented a water-constrained GLB scheme to minimize the operational cost of data centers. In [5], Guo et al. proposed an energy cost saving strategy for data centers using energy storage. In [6], Xu et al. investigated the problem of reducing the peak power demand and energy cost of data centers using partial execution. In [7], Xu et al. studied the problem of reducing the operational cost of cloud data centers in smart microgrids with the consideration of electricity selling/buying, energy storage, load balancing, renewable energies, and dynamic server provisioning or partial execution. In addition, the operational cost of cloud data centers could be offset partially by economical compensation obtained from the participation of demand response programs [12].

Different from existing schemes, we intend to reduce the operational cost of cloud data centers with the help of fog devices [13], [14], which are capable of offering certain advantages by serving nearby cloud users, e.g., avoiding wide-area network (WAN) propagation delay and reducing network bandwidth cost. Since fog devices may not be owned by a cloud service provider [13], the cloud service provider should compensate them for their efforts in serving user requests. When taking economical compensation into consideration, the optimal number of requests processed locally by each fog device should be decided. As a result, existing load balancing schemes developed for cloud data centers can not be applied directly and it is necessary to redesign a cost-ware load balancing algorithm for the fog-cloud system. To achieve the above aim, we first formulate a fog-assisted operational cost minimization problem for the cloud service provider, which is a large-scale mixed integer linear programming (MILP).

To solve the formulated problem efficiently, we propose a parallel and distributed load balancing algorithm with low computational complexity based on Proximal Jacobian Alternating Direction Method of Multipliers (PJ-ADMM) [15]. Though ADMM-based algorithm has been developed in [9] for the operational cost reduction of cloud data centers, it could not be directly used in our problem since standard ADMM is applicable to the convex optimization problem with two-block variables, while there are four-block variables in our optimization problem.

The contributions of this paper could be summarized below:

- We formulate a fog-assisted operational cost minimization problem for a cloud service provider, where the cost consists of four parts, namely the energy cost of
cloud data centers, network bandwidth cost, revenue loss due to the WAN propagation delay, and the economic compensation paid to fog devices.

- We propose a parallel and distributed algorithm for the formulated problem based on PJ-ADMM. Note that the proposed algorithm has low computational complexity since all decisions could be made based on close-form expressions or binary search.

- Extensive simulations show that the proposed algorithm could help the cloud to save operational cost effectively in the presence of fog devices.

The rest of this paper is organized as follows. Section II introduces the system model and the problem formulation. Section III proposes a distributed algorithm based on PJ-ADMM. Then, we conduct extensive simulations in Section IV. Finally, conclusions are made in Section V.

II. MODEL AND FORMULATION

The system model studied in this paper is shown in Fig. 1, where the cloud provides service to its cloud users. To improve the user experience and reduce network bandwidth cost, fog devices (e.g., routers, servers, laptops) in the vicinity of cloud users could be selected as helpers of the cloud. In return, fog devices would receive economical compensation for their efforts. Suppose there are \( N \) available fog devices and \( K \) geographically distributed cloud data centers in the considered fog-cloud system. To serve \( J \) types of requests (this paper mainly focuses on delay-sensitive requests) from nearby cloud users, each fog device \( i \) \((1 \leq i \leq N)\) should decide the type and quantity of application requests to be processed. Then, the remaining requests are dispatched to cloud data centers. In the following parts, we would provide the models related to workload allocation, power consumption, operational cost. Then, we formulate a fog-assisted operational cost minimization problem, which is solved periodically at the beginning of each time slot, e.g., every 1 hour.

A. Workload Allocation Model

When the fog device \( i \) receives the application requests from cloud users, it will process part of them locally and dispatch the remaining requests to the remote cloud. Let \( \alpha_{i,j} \) be the request rate of application \( j \) allocated to the fog device \( i \) (in requests/second) and \( v_{i,j} \) be the service rate of application \( j \) supported by fog device \( i \) (in Mbps), we have

\[
\frac{1}{v_{i,j} / s_j - \alpha_{i,j}} \leq \beta_{i,j} \leq \frac{1}{\alpha_{i,j}}, \quad \forall i, j, \tag{1}
\]

\[
\alpha_{i,j} \geq 0, \quad \forall i, k, \tag{2}
\]

where \( s_j \) is the request size of application \( j \) (in Mb/request) and \( t_j^{\text{max}} \) (in seconds) is the maximum tolerant delay of application \( j \).

Let \( \lambda_{i,j} \) \((1 \leq j \leq J)\) and \( \beta_{i,j,k} \) \((1 \leq k \leq K)\) denote the request rate of application \( j \) arriving at fog device \( i \) and the request rate of application \( j \) allocated from fog device \( i \) to data center \( k \) (both in requests/second), respectively. According to workload balance, we obtain

\[
\alpha_{i,j} + \sum_k \beta_{i,j,k} = \lambda_{i,j}, \quad \forall i, j. \quad \tag{3}
\]

Since the ISP link capacities of cloud data centers are limited, we have the following constraints,

\[
\sum_i \sum_j \beta_{i,j,k} s_j \leq A_k^{\text{max}}, \quad \forall k, \quad \tag{4}
\]

\[
\beta_{i,j,k} \geq 0, \quad \forall i, j, \quad \tag{5}
\]

where \( A_k^{\text{max}} \) is the link capacity of data center \( k \) (in Mbps).

In cloud data centers, the number of active servers could be dynamically configured to ensure that the requests of application \( j \) could be finished within the maximum tolerant delay \( t_j^{\text{max}} \). Then, we have

\[
\frac{1}{c_{j,k} \mu_{j,k} - \sum_i \beta_{i,j,k}} + \frac{1}{\mu_{j,k}} \leq t_j^{\text{max}}, \quad \forall k, \quad \tag{6}
\]

\[
0 \leq c_{j,k} \leq C_{j,k}, \quad \forall j, m, \quad \tag{7}
\]

\[
c_{j,k} \in \mathbb{N}^+, \quad \tag{8}
\]

where \( \mu_{j,k} \) denotes the service rate of servers for application \( j \) in data center \( k \) (in requests/second); \( c_{j,k} \) and \( C_{j,k} \) are the number of active servers and the total number of servers for application \( j \) in data center \( k \), respectively.

B. Power Consumption Models Associated with Cloud Data Centers and Fog Devices

Let PUE\(_k\) denote the power usage effectiveness of the data center \( k \). In addition, denote the idle power and peak power of the servers for application \( j \) in data center \( k \) by \( p_{j,k}^{\text{idle}} \) and \( p_{j,k}^{\text{peak}} \) (both in Watts), respectively. Then, the power consumption of data center \( k \) could be estimated by

\[
P_{k}^{\text{cloud}} = \sum_j \left( \mu_{j,k} c_{j,k} a_{j,k} + b_{j,k} \sum_i \beta_{i,j,k} \right), \quad \tag{9}
\]

where \( a_{j,k} = p_{j,k}^{\text{idle}} + (\text{PUE}_k - 1)p_{j,k}^{\text{peak}} \), \( b_{j,k} = p_{j,k}^{\text{peak}} - p_{j,k}^{\text{idle}} \).
For fog devices, their power consumptions \(P_i^\text{log}\) could be calculated according to a linear energy consumption model as in [17]:

\[
P_i^\text{log} = \left( q_i^\text{idle} + (q_i^\text{peak} - q_i^\text{idle}) \sum_j \frac{\alpha_{i,j} s_j}{v_i} \right), \quad \forall \, i,
\]

where \(v_i = \sum_j v_{i,j}\); \(q_i^\text{idle}\) and \(q_i^\text{peak}\) are the idle power and peak power of fog device \(i\), respectively.

C. Operational Cost Model

In this work, we focus on minimizing the operational cost of a cloud service provider in a given time slot with duration \(T\) without sacrificing the interests of fog devices, where the cost consists of four parts, i.e., the energy cost of data centers, network bandwidth cost, revenue loss associated with WAN propagation delay, and the economic compensation paid to the fog devices.

Let \(v_k\) denote the electricity price associated with electric region that data center \(k\) is located. Then, the energy cost of all data centers is given by

\[
\Gamma_1 = \sum_j \sum_k v_k (c_{j,k} a_{j,k} + b_{j,k} \sum_i \beta_{i,j,k} \tau_j) T. \quad (11)
\]

When receiving a user request, cloud data centers should process it and give a response packet to the user. In reality, both request transmitting and response packet could generate the traffic. Since the traffic volume of user requests is usually far smaller than that of responses, we mainly focus on the traffic generated by response and assume that the traffic volume generated by response for a request of application \(j\) is \(\tau_j\). Let \(B_k\) be the bandwidth price of the ISP link connected to data center \(k\) (in $/Mbps/month). Then, the network bandwidth cost is given by [18]:

\[
\Gamma_2 = \sum_i \sum_j \sum_k \beta_{i,j,k} \tau_j B_k. \quad (12)
\]

For delay-sensitive requests, a moderate increase in user-perceived latency would result in substantial revenue loss for the cloud service provider [3]. Compared with the case that all incoming requests are processed at fog devices, transmitting requests from fog devices to data centers would result in extra WAN propagation latency. Denote the propagation latency of the requests associated with fog device \(i\) and data center \(k\) by \(L_{i,k}\) (in ms), which could be measured through empirical approaches [18]. Then, the revenue loss due to the WAN propagation latency is obtained by [11]:

\[
\Gamma_3 = \sum_i \sum_j \sum_k \omega_j L_{i,k} \beta_{i,j,k} T, \quad (13)
\]

where \(\omega_j\) denotes the latency conversion parameter that translates network propagation latency into revenue loss of application \(j\) (in $/ms/req).

Since fog devices could help the cloud to reduce WAN propagation latency and bandwidth cost, the cloud service provider should make a compensation to fog devices for their energy costs incurred by serving requests. We assume that compensations obtained by fog devices are proportional to the served requests. The reason behind this assumption is that fog devices are willing to serve the requests for the cloud service provider on the condition that their interests would not be damaged. According to the above assumption, the compensation paid to the fog devices is given by

\[
\Gamma_4 = \sum_i S_i h_i q_i \sum_j \alpha_{i,j} s_j / v_i T, \quad (14)
\]

where \(h_i \geq 1\) is the compensation factor associated with fog device \(i\) and \(S_i\) is the electricity price associated with electric region that fog device \(i\) is located; \(q_i = q_i^\text{peak} - q_i^\text{idle}\). Note that we implicitly neglects the economical compensation for the fixed energy consumption of fog devices since serving cloud users are not necessarily their sole purpose. In practice, the cloud can just interact with active fog devices.

D. Operational Cost Minimization Problem

With above-mentioned models, a fog-assisted operational cost minimization problem is formulated as follows,

\[
\begin{align}
\text{(P1)} \quad & \min \quad \Gamma_1 + \Gamma_2 + \Gamma_3 + \Gamma_4 \\
\text{s.t.} \quad & \alpha_{i,j} \leq \frac{1}{s_j} (v_{i,j} - s_j / t_j^{\text{max}}), \\
& \sum_i \beta_{i,j,k} \leq c_{j,k} \mu_{j,k} - e_{j,k}, \\
& (2) - (5), (7) - (8),
\end{align}
\]

where \(e_{j,k} = \frac{3}{t_j^{\text{max}} \mu_{j,k}}\); (15b) and (15c) are obtained by adjusting the forms of (1) and (6); the decision variables of P1 are \(\alpha_{i,j}, \beta_{i,j,k},\) and \(c_{j,k}\).

III. ALGORITHM DESIGN

The objective function and constraints of P1 are linear. Moreover, \(c_{j,k} (\forall j, k)\) are integer variables. Thus, P1 is a mixed integer linear programming (MILP). In addition, the number of constraints and variables is \(NJ + 2NJ + JK + 2J + K + NJK + JK + N\), respectively. Since the number of fog devices could be very large, P1 is a large-scale mixed integer linear programming (MILP), e.g., when \(N = 1000, J = 10, K = 20\), the number of constraints and variables would be larger than 200000. When solving such a large-scale MILP in a centralized manner, the corresponding computation time would increase dramatically if the problem size becomes large [19]. Therefore, we are motivated to propose a scalable and distributed algorithm for P1 based on PJ-ADMM, which could solve large-scale convex optimization problems efficiently. To solve P1 using PJ-ADMM, some transformations are needed. Firstly, we should transform P1 into a convex optimization problem by relaxing the constraints related to integer variables. Then, some constraints should be decoupled so that the framework of PJ-ADMM could be used. The transformation detail is given as follows.
A. Problem Transformation

According to (20), (15c) could be transformed into
\[ c_{j,k} \geq \frac{1}{\mu_j}(\sum_i \beta_{i,j,k} + e_{j,k}). \]
Since \( c_{j,k} \in \mathbb{N}^+ \), \( c_{j,k} = \left\lfloor \frac{1}{\mu_j}(\sum_i \beta_{i,j,k} + e_{j,k}) \right\rfloor + 1. \) Thus, \( c_{j,k} \leq \frac{1}{\mu_j}(\sum_i \beta_{i,j,k} + e_{j,k}) + 1. \) Taking constraints (6) and (7) into consideration, we have
\[ \sum_i \beta_{i,j,k} + e_{j,k} \leq \mu_j C_{j,k}. \]
As a result, **P2** is transformed into **P2** by discarding some constant items in the objective function,
\[
\begin{align*}
\text{(P2)} & \quad \min \Gamma \\
& \text{s.t.} \quad (15b), (2) - (5), \\
& \sum_i \beta_{i,j,k} \leq \mu_j C_{j,k} - e_{j,k}, \\
& c_{j,k} = \left\lfloor \frac{1}{\mu_j}(\sum_i \beta_{i,j,k} + e_{j,k}) \right\rfloor, \\
& \beta_{i,j,k} \geq 0, \forall i,j,k.
\end{align*}
\]

where \( \Gamma = \sum_{i,j,k} \frac{b_i s_i}{\alpha_{i,j,k}} \left( T_i \right) + \sum_{i,j,k} (\theta_{i,j,k}^T + \omega_j L_{i,k} T + \omega_j^T (a_{i,j,k} + b_{i,j,k})) \beta_{i,j,k}; \) the decision variables are \( \alpha_{i,j,k} \) and \( \beta_{i,j,k}. \)

To avoid the couplings, we adopt a set of auxiliary variables \( \beta_{i,j,k} = \gamma_{i,j,k} \), and \( \beta_{i,j,k} = l_{i,j,k}. \) Consequently, **P2** could be transformed into **P3** equivalently.

\[
\begin{align*}
\text{(P3)} & \quad \min \Gamma \\
& \text{s.t.} \quad (15b), (16d), (2), (4), (5), \\
& \alpha_{i,j} + \sum_k \gamma_{i,j,k} = \lambda_{i,j}, \forall i,j, \\
& \gamma_{i,j,k} = \beta_{i,j,k}, \forall i,j,k, \\
& \beta_{i,j,k} = l_{i,j,k}, \forall i,j,k, \\
& \sum_i \beta_{i,j,k} \leq \mu_j C_{j,k} - e_{j,k}, \forall j,k, \\
& \gamma_{i,j,k} \geq 0, \forall i,j,k, \\
& l_{i,j,k} \geq 0, \forall i,j,k,
\end{align*}
\]

where the decision variables are \( \alpha_{i,k}, \gamma_{i,j,k}, \beta_{i,j,k}, \) and \( l_{i,j,k}. \)

B. The Proposed Distributed Algorithm

Define \( X \) as the collection of variables \( \alpha_{i,k}, \gamma_{i,j,k}, \beta_{i,j,k}, \) and \( l_{i,j,k} \) for all \( i, j, k. \) Denote the augmented Lagrangian of **P3** by \( \mathcal{L}_p(X; \phi_{i,j,k}, \psi_{i,j,k}, \chi_{i,j,k}), \) which is given in (18), where \( \rho \) is the penalty parameter. \( \phi_{i,j,k} \) and \( \psi_{i,j,k} \) are dual variables associated with (17c)-(17e), respectively.

Following the framework of PJ-ADMM, we design a distributed algorithm for **P3** as follows.

1. **Initialization**: Let all decision variables be zero. For each iteration \( w = 0, 1, 2, \ldots \), the following steps are repeated in parallel until convergence.

2.1 \( \alpha_{i,j}-\text{minimization} \): Each fog device \( i \) solves the following optimization problem to obtain \( \alpha_{i,j}^{w+1} \) in parallel.

\[
\begin{align*}
\text{(P4)} & \quad \min_{\alpha_{i,j}} T_1(\alpha_{i,j}, \alpha_{i,j}^w, \gamma_{i,j,k}^w, \phi_{i,j,k}^w) \\
& \text{s.t.} \quad s \leq \alpha_{i,j} \leq \frac{1}{s_j} (v_j - s_j/t_j^{\max}),
\end{align*}
\]

where \( T_1 = \frac{\phi}{2} (\alpha_{i,j} + \sum_k \gamma_{i,j,k} - \lambda_{i,j})^2 + \frac{\theta}{2} (\alpha_{i,j} - \alpha_{i,j}^w)^2 + \phi (\gamma_{i,j,k} + \gamma_{i,j,k}^w)\alpha_{i,j}, \) where \( \lambda_{i,j} > 0 (\forall i, j) \) are the elements of the diagonal matrix used in the proximal term associated with \( \alpha_{i,j}; \) the solution to **P4** could be found in Appendix A.

2.2 \( \gamma_{i,j,k}-\text{minimization} \): Each fog device \( i \) solves the following optimization problem to obtain \( \gamma_{i,j,k}^{w+1} \) in parallel.

\[
\begin{align*}
\text{(P5)} & \quad \min_{\gamma_{i,j,k}} T_2(\gamma_{i,j,k}, \gamma_{i,j,k}^w, \alpha_{i,j}^w, \beta_{i,j,k}^w, \phi_{i,j,k}^w, \psi_{i,j,k}^w) \\
& \text{s.t.} \quad \gamma_{i,j,k} \geq 0,
\end{align*}
\]

where \( T_2 = \frac{\phi}{2} (\alpha_{i,j} + \sum_k \gamma_{i,j,k} - \lambda_{i,j})^2 + \frac{\theta}{2} (\alpha_{i,j} - \alpha_{i,j}^w)^2 + \psi_{i,j,k} (\gamma_{i,j,k} - \gamma_{i,j,k}^w)\gamma_{i,j,k} + \frac{\phi}{2} (\gamma_{i,j,k} - \gamma_{i,j,k}^w)^2 + \sum_k (\gamma_{i,j,k}^w - \gamma_{i,j,k})^2, \) where \( \gamma_{i,j,k} > 0 (\forall i, j, k) \) are the elements of the diagonal matrix used in the proximal term associated with \( \gamma_{i,j,k}; \) the solution to **P5** could be found in Appendix B.

2.3 \( \beta_{i,j,k}-\text{minimization} \): Each data center \( k \) solves the following optimization problem to obtain \( \beta_{i,j,k}^{w+1} \) in parallel.

\[
\begin{align*}
\text{(P6)} & \quad \min_{\beta_{i,j,k}} T_3(\beta_{i,j,k}, \beta_{i,j,k}^w, \alpha_{i,j}^w, \beta_{i,j,k}^w, \phi_{i,j,k}^w, \psi_{i,j,k}^w) \\
& \text{s.t.} \quad \sum_i \beta_{i,j,k} s_j \leq A_k^{\max},
\end{align*}
\]

where \( T_3 = \sum_i \left( \frac{\theta}{2} (\beta_{i,j,k} - \beta_{i,j,k}^w)^2 + \psi_{i,j,k} (\beta_{i,j,k} - \beta_{i,j,k}^w)^2 + \frac{\phi}{2} (\beta_{i,j,k} - \beta_{i,j,k}^w)^2 + \eta_{i,j,k} \right) \beta_{i,j,k} - \beta_{i,j,k}^w \), where \( \eta_{i,j,k} > 0 (\forall i, j, k) \) are the elements of the diagonal matrix used in the proximal term associated with \( \beta_{i,j,k}; \) the solution to **P6** could be found in Appendix C.

2.4 \( l_{i,j,k}-\text{minimization} \): Each data center \( k \) solves the following optimization problem to obtain \( l_{i,j,k}^{w+1} \) in parallel.

\[
\begin{align*}
\text{(P7)} & \quad \min_{l_{i,j,k}} T_4(l_{i,j,k}, l_{i,j,k}^w, \beta_{i,j,k}^w, \beta_{i,j,k}^w, \phi_{i,j,k}^w, \psi_{i,j,k}^w) \\
& \text{s.t.} \quad l_{i,j,k} \geq 0,
\end{align*}
\]

where \( T_4 = \sum_i \left( \frac{\theta}{2} (l_{i,j,k} - \beta_{i,j,k}^w)^2 + \frac{\phi}{2} (l_{i,j,k} - \beta_{i,j,k}^w)^2 + \kappa_{i,j,k} (l_{i,j,k} - l_{i,j,k}^w)^2 \right) \), where \( \kappa_{i,j,k} > 0 (\forall i, j, k) \) are the elements of the diagonal matrix used in the proximal term.
associated with \( l_{i,j,k} \); the solution to P7 could be found in Appendix D.

### 2.5 Dual update

Dual variables are updated in the following way, i.e.,
\[
\phi_{i,j}^{w+1} = \phi_{i,j}^w + \delta (\alpha_{i,j}^{w+1} + \sum_k \gamma_{i,j,k}^{w+1} - \lambda_{i,j})
\]
(where \( \delta \) is a positive damping parameter); \( \varphi_{i,j,k}^{w+1} = \varphi_{i,j,k}^w + \delta (\beta_{i,j,k}^{w+1} - \beta_{i,j,k}) \), \( \chi_{i,j,k}^{w+1} = \chi_{i,j,k}^w + \delta (\theta_{i,j,k}^{w+1} - \theta_{i,j,k}) \).

### 3. Termination

If the change of the objective function in two consecutive iterations is lower than the chosen threshold \( \varepsilon \) and feasibility violation metric \( \varepsilon_{m} = \sum_{i=1}^{N} \sum_{j=1}^{J} (\alpha_{i,j}^{w+1} + \sum_{k=1}^{K} (\beta_{i,j,k}^{w+1} - \lambda_{i,j})) \) is smaller than \( \zeta \), the proposed algorithm would terminate.

**Theorem 1** If the optimal solution set of P3 is non-empty, the algorithm developed based on PJ-ADMM would converge to an optimal solution of P3 when \( \theta_{i,j} > \zeta, \sigma_{i,j,k} > (K+1)\zeta, \eta_{i,j,k} > 2\zeta, \kappa_{i,j,k} > \zeta \), where \( \zeta = (\frac{1}{2} - \frac{1}{\sigma}) \), \( \sigma > 0, 0 < \delta < 2 \).

**Proof:** In P3, it can be observed that the objective function is separable and continuous over all variables. Moreover, the minimum values of all separable functions are zero and the effective domains of such functions are nonempty/closed. Thus, all separable functions are closed proper convex (i.e., the first assumption for the optimality of PJ-ADMM is satisfied). For the convex optimization problem P4, all inequality constraints are affine. Therefore, the strong duality holds when the optimal solution set is non-empty. Continually, the unaugmented Lagrangian of P3 has a saddle point (i.e., the second assumption for the optimality of PJ-ADMM is satisfied). Since the square of spectral norm of the relation matrix associated with each block of variables (there are totally 4 blocks) is 1, \( K+1, 2 \), and 1, respectively, we can complete the proof according to Lemma 2.2 and Theorem 2.3 in [15].

### C. Algorithmic Complexity

In the proposed distributed algorithm, all decisions could be made based on close-form expressions or binary search (see Appendices A-D). Moreover, all decisions in P4-P7 could be made by each entity (e.g., a fog device or a data center) in parallel. Therefore, we analyze the computation time complexity associated with each entity as follows. Let \( N_{\text{iter}} \) and \( N_b \) be the total number of iterations of the proposed distributed algorithm and the maximum iteration number of binary search (used in solving P6 and P7), respectively. Typically, several tens of iterations are needed for binary search \([23]\), i.e., \( N_b = \mathcal{O}(10) \). The computational time complexity associated for each fog device and each data center is given by \( \mathcal{O}(N_{\text{iter}}JK) \) and \( \mathcal{O}(N_{\text{iter}}NJN_b) \), respectively. Considering the truth that \( N \gg K \), the algorithmic computation time mainly depends on the computation time of data centers.

### D. Algorithmic Implementation

![Fig. 2. An information flow of the proposed distributed algorithm](image)

The information flow of the proposed distributed algorithm for solving P3 could be illustrated by Fig. 2. Firstly, each fog device and each data center make their respective decisions in parallel. Then, fog devices broadcast \( \gamma_{i,j,k}^{w+1} \) to data centers. After receiving \( \gamma_{i,j,k}^{w+1} \), data centers obtain \( \varphi_{i,j,k}^{w+1} \) and broadcast \( \beta_{i,j,k}^{w+1} \) and \( \varphi_{i,j,k}^{w+1} \) to fog devices. In summary, a two-way information broadcast is needed in one iteration. Note that there are communication overheads incurred by transmitting \( \gamma_{i,j,k}^{w+1}, \beta_{i,j,k}^{w+1} \) and \( \varphi_{i,j,k}^{w+1} \) in above-mentioned iteration processes. An alternative way of reducing communication overheads is to implement the proposed distributed algorithm in the cloud, which has powerful computation resources. After obtaining the final results, decision information would be sent back to fog devices. Under this situation, one two-way message interaction is needed.

### IV. PERFORMANCE EVALUATION

#### A. Simulation Setup

| Parameters related to data centers and applications |
| --- |
| \( k \) | \( j \) | \( C_{l,k} \) | \( p_{l,k}^{\text{min}} \) | \( p_{l,k}^{\text{max}} \) | \( p_{l,k} \) | \( \rho_{l,k} \) | \( \rho_{l,k}^{\text{max}} \) |
| --- | --- | --- | --- | --- | --- | --- | --- |
| \( k = 1 \) | \( j = 1 \) | 2000 | 110 | 220 | 3 | 0.5 |
| \( j = 2 \) | 1600 | 100 | 200 | 2.625 | 0.6 |
| \( k = 2 \) | \( j = 1 \) | 2000 | 95 | 190 | 2.7 | 0.5 |
| \( j = 2 \) | 1600 | 90 | 180 | 2.4 | 0.6 |
| \( k = 3 \) | \( j = 1 \) | 2000 | 120 | 240 | 2.85 | 0.5 |
| \( j = 2 \) | 1600 | 100 | 200 | 2.25 | 0.6 |

In this section, simulations are conducted to show the performance of the proposed algorithm. For performance comparison, a baseline is adopted, which intends to minimize
the operational cost (e.g., energy cost, bandwidth cost and revenue loss) of a cloud service provider without considering the help of fog devices, i.e., $\alpha_{i,j} = 0$. The main system parameters are given as follows, $T = 1$ hour, $N = 1000$, $J = 2$, $K = 3$, $s_1 = 0.25$ Mb, $s_2 = 0.5$ Mb, $A_{\text{max}} = 10^5$ Mbps, $A_{\text{peak}} = 0.9 \times 10^5$ Mbps, $A_{\text{max}} = 0.8 \times 10^5$ Mbps, $B_1 = 0.005$ $$/Mbps/hour$ [3], $\text{PUE}_1 = 1.13$, $\text{PUE}_2 = 1.14$, $\text{PUE}_3 = 1.15$ [21]. $T_j = 1$ Mb, $\nu_1 = 30$, $\nu_2 = 35$, $\nu_3 = 40$ (in $$/MWh$). Other parameters related to data centers and applications are provided in Table I (6). Suppose that $v_{i,j}$ follows a uniform distribution with parameters 2.25 and 3 (in requests/second), i.e., $v_{i,j} \sim \mathcal{U}(2.25, 3)$. Since fog devices would consume higher power compared with cloud servers given the same service capability as in [16], we set $q_{i}^{\text{peak}} \sim \mathcal{U}(440, 500)$ (in Watts). In addition, we assume that $S_i \sim \mathcal{U}(30, 60)$, $\lambda_{i,j} \sim \mathcal{U}(\frac{1}{\nu_{i,j}} \sum_k C_{j,k} \mu_{j,k}, \frac{1}{\nu_{i,j}} \sum_k C_{j,k} \mu_{j,k})$ (in requests/second). $L_{i,k} \sim \mathcal{U}(10, 40)$ (in ms). We set $q_{i}^{\text{idle}} = 0.5 q_{i}^{\text{peak}}$ [22], $\delta = 1$.

B. Simulation Results

1) Convergence results of the proposed algorithm: Since $\mathbf{P}_2$ is a relaxed problem of $\mathbf{P}_1$, there may be a performance gap between the optimal total cost generated by the proposed algorithm and the optimal solution of $\mathbf{P}_1$. To obtain the optimal solution of $\mathbf{P}_1$, GAMS commercial solver is adopted. In Fig. [5] it can be observed that the above-mentioned performance gap is negligible, e.g., the relative optimality gap is 0.0094% in this scenario. Although several thousands of iterations are needed to find the optimal solution of $\mathbf{P}_2$, it does not mean that the proposed PJ-ADMM-based algorithm is inefficient. In fact, the proposed algorithm could find a suboptimal solution with very low optimality loss in several hundreds of iterations. For example, running on a single Intel Core i5-2410M 2.3GHz server with 4G RAM, the relative optimality loss achieved by the proposed algorithm at iteration 600 is smaller than 0.00006% and the time consumed is 42 seconds, which would become 14 seconds if a parallel implementation is considered (note that there are at least $K=3$ parallel optimization subproblems in each step of the distributed algorithm).

2) The impacts of $h_i$, $B_k$, and $\omega_j$: Since $h_i$ represents the economical compensation factor for fog devices, changing the value of $h_i$ would affect the extent of their participation. Specifically, fog devices would process less workloads with the increase of $h_i$ as shown in Fig. [6]. According to workload balance, the total workload finished by the cloud would increase with the increase of $h_i$. Under the current simulation scenario, a win-win situation for fog devices and the cloud can be achieved given $1 < h_i < 20$ when $B_k = 0.005$. In addition, given the same $h_i$, the number of requests processed by the cloud is increasing with the decrease of $B_k$, which is obvious since $B_k$ could be regarded as the "weight" of $\beta_{i,j,k}$ in the objective function of $\mathbf{P}_1$. In the future, $B_k$ will continue to decrease, which does not mean that fog computing is unattractive in IoT era. The reason is that some time-critical IoT applications have to be processed locally, e.g., health monitoring. Therefore, more and more requests would be processed by fog devices with the increase of $\omega_j$ as shown in Fig. [5b].

3) Cost components under different algorithms: In Fig. [6] we provide cost components under the proposed algorithm and the baseline. It can be observed that fog devices can help the cloud to reduce the operational cost by processing nearby user requests, resulting in lower energy cost, bandwidth cost and revenue loss, e.g., relative cost reduction (RCR) achieved by the proposed algorithm is 54.22% and 18.08% when $h_i = 1$ and $h_i = 8$, respectively. Though different parameter configurations would affect the value of RCR as shown in next subsection, the performance of the proposed algorithm is not worse than the baseline. The reason is that the proposed algorithm would be equivalent to the baseline if $h_i$ required by each fog device is too high.

4) The impact of workloads: We modify the workload $\lambda_{i,j}$ by enlarging $C_{i,j}$ several times (e.g., 1, 2, 3, 4), considering that $\lambda_{i,j} \sim \mathcal{U}(\frac{1}{\nu_{i,j}} \sum_k C_{j,k} \mu_{j,k}, \frac{1}{\nu_{i,j}} \sum_k C_{j,k} \mu_{j,k})$. Since the total processing capacity of fog devices is limited, more workloads would be dispatched to the cloud for processing with the increase of incoming workloads at fog devices. As a result, the ratio of fog workload to the total workload and RCR would be reduced simultaneously as shown in Fig. [7]. In the future, more and more fog devices will be deployed to support some emerging time-critical IoT applications, resulting in larger processing capacity of fog devices and increasing RCR.

V. CONCLUSIONS

In this paper, we investigated the problem of reducing the operational cost of geo-distributed cloud data centers with the help of fog devices. To be specific, we first formulated a fog-assisted operational cost minimization problem for a cloud service provider with the consideration of economical compensation paid to fog devices. Then, we proposed a parallel and distributed load balancing algorithm to solve the formulated problem based on PJ-ADMM. Note that the proposed algorithm has low computational complexity since all decisions could be obtained based on close-form expressions or binary search. Extensive simulation results showed the effectiveness of the proposed algorithm.

REFERENCES

[1] Cisco Global Cloud Index Projects Cloud Traffic to Nearly Quadruple Representing 92 Percent of Total Data Center Traffic by 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/ cloud/index.html

[2] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, “Cutting the electric bill for internet-scale systems,” Proc. of ACM SIGCOMM, Barcelona, Spain, Aug. 17-21, 2009.

[3] A. Greenberg, J. Hamilton, D.A. Maltz, and P. Patel, “The cost of a cloud: research problems in data center networks,” ACM SIGCOMM Computer Communication Review, vol. 39, no. 1, pp. 68-73, Jan. 2009.

[4] M. Ghamkhari and H. Mohsenian-Rad, “Energy and performance management of green data centers: a profit maximization approach,” IEEE Trans. Smart Grid, vol. 4, no. 2, pp. 1017-1025, June. 2013.

[5] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, “It’s not easy being green,” Proc. of ACM SIGCOMM, Helsinki, Finland, August 13-17, 2012.

[6] L. Rao, X. Liu, L. Xie, and W. Liu, “Coordinated energy cost management of distributed internet data centers in smart grid,” IEEE Trans. Smart Grid, vol. 3, no. 1, pp. 50-58, Sep. 2012.
L. Yu, T. Jiang and Y. Cao, “Energy cost minimization for distributed data centers,” IEEE Trans. Smart Grid, DOI: 10.1109/TSG.2016.2640453, 2016.

N.H. Tran, D.H. Tran, S. Ren, Z. Han, E. Huh, and C. Hong, “How geodistributed data centers do demand response: a game-theoretic approach,” IEEE Trans. Smart Grid, vol. 7, no. 2, pp. 937-947, March 2016.

M. Chiang and T. Zhang, “Fog and IoT: an overview of research opportunities,” IEEE Internet of Things Journal, DOI: 10.1109/JIOT.2016.2584538, 2016.

F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” Proc. of ACM Mobicom, 2012.

W. Deng, M. Lai, Z. Peng, and W. Yin, “Parallel multi-block ADMM with $o(1/k)$ convergence,” http://arxiv.org/abs/1312.3040, 2014.

R. Deng, R. Lu, C. Lai, T. H. Luan, H. Liang, “Optimal workload allocation in fog-cloud computing towards balanced delay and power consumption,” IEEE Internet of Things Journal, vol. 3, no. 6, pp. 1171-1181, Dec. 2016.

F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker, “Fog computing may help to save energy in cloud computing,” IEEE Journal On Selected Areas In Communications, vol. 34, no. 5, pp. 1728-1739, May 2016.

H. Xu and B. Li, “Joint request mapping and response routing for geodistributed cloud services,” Proc. of IEEE INFOCOM, 2013.

B. Chai, A. Costa, S.D. Ahipasaoglu, C. Yu, and Z. Yang, “Optimal meeting scheduling in smart commercial buildings for energy cost reduction,” IEEE Trans. Smart Grid, DOI: 10.1109/TSG.2016.2625313, 2016.

A. Neumaier and O. Shcherbina, Safe bounds in linear and mixed-integer linear programming, Mathematical Programming, vol. 99, pp. 283-296.
Fig. 6. Cost components ($B_k = 0.003$ and $\omega_j = 3 \times 10^{-8}$)

Fig. 7. Relative cost reduction and workload percentage of fog devices

Appendix A

The Solution to P4

Proof: Let the gradient of $Y_1$ with respect to $\alpha_{i,j}$ be zero, we have $\nabla_{\alpha_{i,j}} Y_1 = 0$. Let $\theta_{i,j} = \tilde{\theta}$ for all $i, j$, we have $\alpha_{i,j}^* = \frac{\rho \lambda_{i,j} - \rho \alpha_{i,j}^w - \phi_{i,j}^{w+1} \frac{\phi_{i,j}^{w+1}}{\tilde{\phi}_{i,j}^{w+1}} + \rho \bar{\phi}_{i,j}^{w+1}}{\rho + \tilde{\phi}_{i,j}^{w+1}}$. Note that $\alpha_{i,j}^* \in [0, (v_{i,j} / s_j - 1 / t_j^{\text{max}})]$, we have $\alpha_{i,j}^{w+1} = \min(\max(0, \alpha_{i,j}^*), (v_{i,j} / s_j - 1 / t_j^{\text{max}}))$.

Appendix B

The Solution to P5

Proof: Let the gradient of $Y_2$ with respect to $\gamma_{i,j,k}$ be zero., we have $\nabla_{\gamma_{i,j,k}} Y_2 = 0$. Let $\xi_{i,j,k} = \sigma$ for all $i, j, k$, we have $\gamma_{i,j,k}^{\text{opt}} = \frac{\rho \lambda_{i,j,k} - \rho \gamma_{i,j,k}^{w} - \phi_{i,j,k}^{w+1} \frac{\phi_{i,j,k}^{w+1}}{\tilde{\phi}_{i,j,k}^{w+1}} + \rho \bar{\phi}_{i,j,k}^{w+1}}{\rho + \tilde{\phi}_{i,j,k}^{w+1}}$. Note that $\gamma_{i,j,k}^{\text{opt}} \in [0, (v_{i,j,k} / s_j - 1 / t_j^{\text{max}})]$, we have $\gamma_{i,j,k}^{w+1} = \min(\max(0, \gamma_{i,j,k}^{\text{opt}}), (v_{i,j,k} / s_j - 1 / t_j^{\text{max}}))$.

Appendix C

The Solution to P6

Proof: Let the first derivative of $Y_3$ with respect to $\beta_{i,j,k}$ be zero, we have

$$\beta_{i,j,k}^{\text{opt}} = \frac{g_{i,j,k}}{(\eta_{i,j,k} + 2 \rho)}$$

where $g_{i,j,k} = \rho (\alpha_{i,j,k}^w + \bar{\beta}_{i,j,k}^w)$. Then, $\beta_{i,j,k}^{\text{opt}}$ is gradually reduced to be zero with the increase of $g_k$ according to $\text{(24)}$, we can find the optimal $g_k$ based on binary search.

Appendix D

The Solution to P7

Proof: Let the gradient of $Y_4$ with respect to $l_{i,j,k}$ should be zero, we have

$$l_{i,j,k}^* = \frac{z_{i,j,k}}{(\rho + \kappa)}$$

where $z_{i,j,k} = i_{i,j,k}^w + \bar{\rho}_{i,j,k}^w + \bar{\lambda}_{i,j,k}^w$. Denote the set $\{i | z_{i,j,k} > 0\} = 0$ for all $i, j, k$. Since the first derivative of $Y_4$ with respect to $l_{i,j,k}$ should be zero, we have

$$l_{i,j,k}^* = \frac{z_{i,j,k} - \xi_{i,j,k}}{(\rho + \kappa)}$$

where $\xi_{i,j,k} (\forall j, k)$ are non-negative dual variables related to $\text{(22b)}$; the optimal $\xi_{i,j,k}$ is decided by the equation $\sum_{i \notin K_{i,j,k}} l_{i,j,k} = \mu_{i,j,k} C_{i,j,k} - e_{i,j,k}$. Since $l_{i,j,k}^*$ is gradually reduced with the increase of $\xi_{i,j,k}$, according to $\text{(26)}$, we can find the optimal $\xi_{i,j,k}$ based on binary search.