Cost-Efficient Resource Provisioning in Delay-Sensitive Cooperative Fog Computing

Shuaibing Lu[a,b], Jie Wu[b], Yubin Duan[b], Ning Wang[b], and Zhiyi Fang[a]

[a]College of Computer Science and Tech., Jilin University
[b]Dept. of Computer and Info. Sciences, Temple University

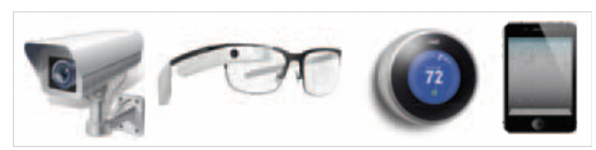
Outline

1. Background and Motivation
2. Model and Formulation
3. RP With Unlimited-Processor Fog Node (UPFN)
4. RP With Limited-Processor Fog Node (LPFN)
5. Evaluations
6. Conclusions

1. Background

- **Cloud Data Center Networks (DCNs)**
  - Supporting cloud-based applications for large enterprises

- **Fog Computing**
  - Providing computation, storage, and networking services between end devices and traditional cloud DCNs.



2. Model and Formulation

Motivation

- IoT devices generate data constantly and the analysis must be very rapid
- Meeting computation and communication demands
- Reduce the transmission latency and decrease monetary cost

Objective

- Find an appropriate scheme for the set of users with minimum cost, and support all users' demands in both resource and deadline constraints.

2. Model and Formulation

Formulation

- Resource provisioning (RP) problem for delay-sensitive users under the capacity constraints by considering the cooperation of fog nodes, while realizing cost efficiency of network operators in fog computing.

minimize	$\sum_{j \in X} \mathbf{1}_{[\sum_{i \in U} \lambda_{ij} w_i > 0]} \tau_j$	(1)
subject to	$D_i \leq T, \forall i \in \mathbf{U}$	(2)
	$D_i = max_{j \in \boldsymbol{X}_i}\{d_{ij} + p_{ij}\}$	(3)
	$0 \leq \lambda_{ij} \leq 1, \sum_{i \in \mathbf{U}, j \in \mathbf{G}} \lambda_{ij} = 1$	(4)
	$\sum_{i \in \mathbf{U}} \lambda_{ij} w_i \leq c_j$	(5)

NP-hard Problem

3. RP With Unlimited-Processor Fog Node (UPFN)

Feasibility Checking

- whether there exists a provision for users that can support users' demands within the constraint of their deadline

Steps

- Construct an auxiliary graph with respect to the connections between users and fog nodes;
- Obtain the maximum flow using Edmonds-Karp algorithm

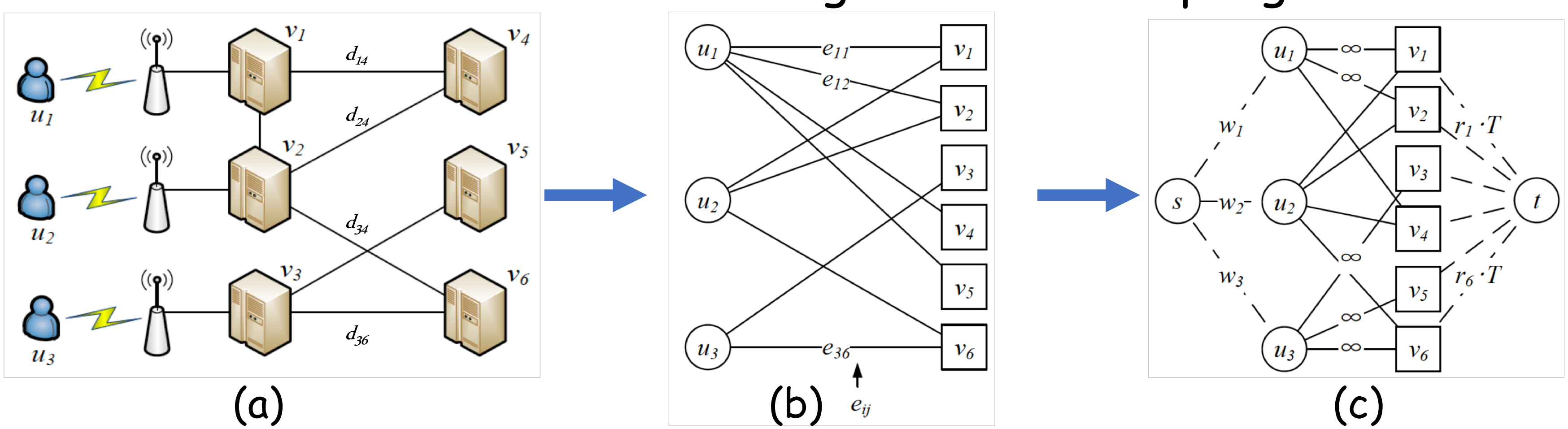


Fig 1. Motivation example

3. RP With Unlimited-Processor Fog Node (UPFN)

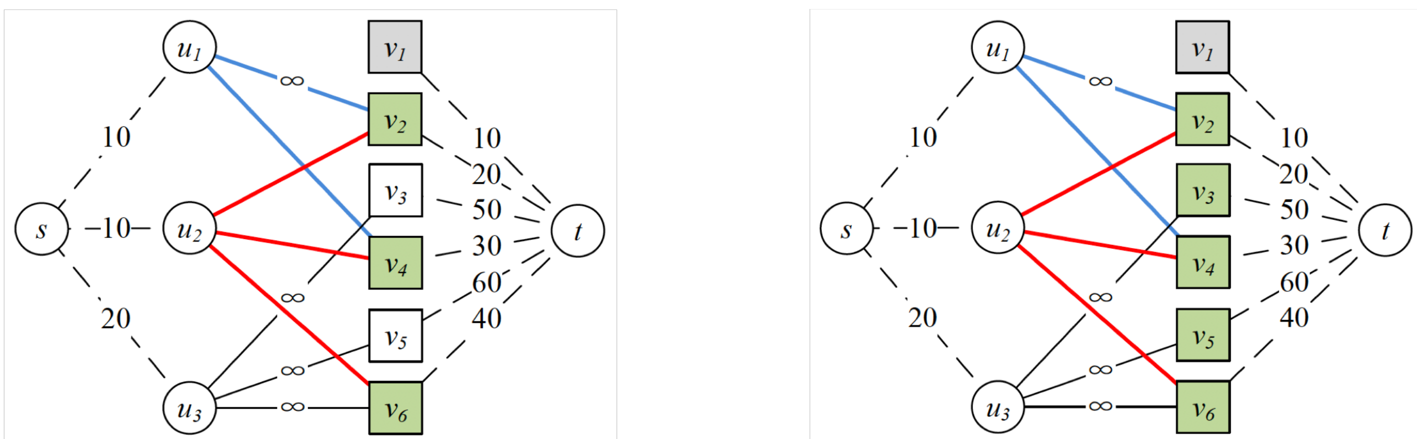
Cooperative Influence

- Local Cooperative Influence

$$\psi_i = \left| \overline{M}_{G^*/v_j} - \overline{M}_{G^*} \right| \text{ where } M_{G^*} = \max_{j \in G^*}\{p_j\}$$

- Global cooperative influence

$$\varphi_i = \left| \bar{R}_{G/v_j} - \bar{R}_G \right| \text{ where } \bar{R}_G = \frac{\sum_{v_j \in G} p_j}{|G|}$$



(a) (b)

Fig 2. Cooperative Influences.

Local Influence Greedy (LIG) Algorithm

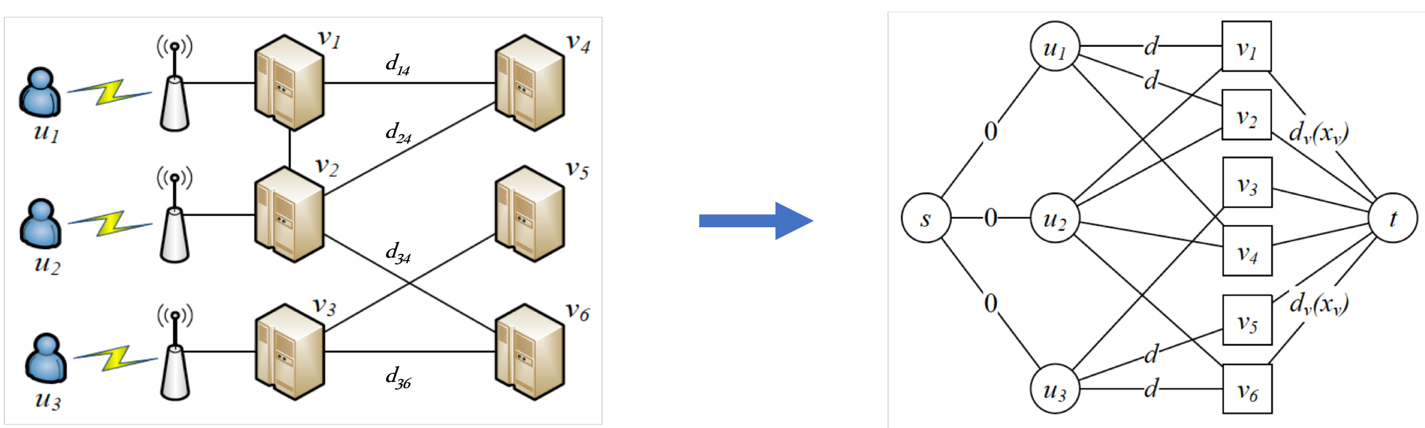
- **Step 1**
  - Find a feasible solution using feasibility checking.
- **Step 2**
  - Calculate the latest finished completion time of fog nodes.
- **Step 3**
  - Calculate φ_i for each fog node and rebuild the set G with fog nodes by an increasing order $j = argmin\{\psi_i/\tau_j\}$;
- **Step 4**
  - Check the feasibility of G by using feasibility checking and remove fog node from set V until there is no feasible solution.

Global Influence Greedy (GIG) Algorithm

- **Step 1, 2, 4, same to LIG**
- **Step 3**
  - Calculate φ_i for each fog node; Rebuild the set G with fog nodes by an increasing order $j = argmin\{\varphi_i/\tau_j\}$;
- **Time Complexity**
  - $\boldsymbol{O}(|\boldsymbol{V}|^5 \cdot (|\boldsymbol{V}| + |\boldsymbol{U}|))$

4. RP With Limited-Processor Fog Node (LPFN)

- **Delay Function:** $d_v(x_v) = \alpha \cdot x_v + b$
- **Optimal Provision Finding (OPF) Problem**
  - Given U, G, and $d_v(x_v)$, an OPF Problem is how to find a provision in G to minimize the delay of users U.
  - Convert the OPF problem into a Continuous Symmetric Network Congestion Game (CSNCG) problem.



(a) (b)

Fig 3. A converted graph based on CSNCG

Local Influence Greedy LPFN (GIG-LPFN)

- **Step 1**
  - Construct graph G'' based on the connections of G and U.
- **Step 2**
  - Replace each edge in G'' with n parallel edges between each node, with weight $d_v(1)$, $d_v(2)$, …, $d_v(n)$.
- **Step 3**
  - Find a minimum delay provision with min-cost flow for each fog node.
- **Step 4**
  - Calculate φ_j for each fog node and rebuild set G with fog nodes by an increasing order $j = argmin\{\varphi_j/\tau_j\}$.
- **Step 5**
  - Remove fog node from set G until there is no feasible solution

Global Influence Greedy LPFN (GIG-LPFN)

- **Step 1, 2, 3, 5 same to LIG-LPFN**
- **Step 4**
  - Calculate ψ_j for each fog node and rebuild set G with fog nodes by an increasing order $j = argmin\{\psi_j / \tau_j\}$;
- **Property**
  - GIG-LPFN and LIG-LPFN are bounded by $\frac{8}{3}opt + \frac{\epsilon^2}{8m\alpha}$.
- **Time Complexity**
  - $\boldsymbol{O}(|\boldsymbol{V}|^5 \cdot (|\boldsymbol{V}| + |\boldsymbol{U}|))$

5. Evaluations

Basic Setting-Synthetic Dataset

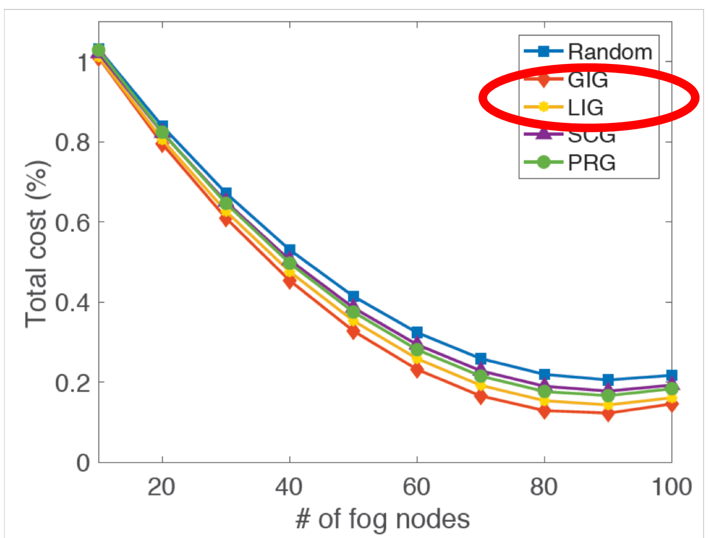
- Unit weight of users' workloads: 1GB
- Sizes of workloads: [0,50] (uniform randomly distribution).

Three Comparison algorithms

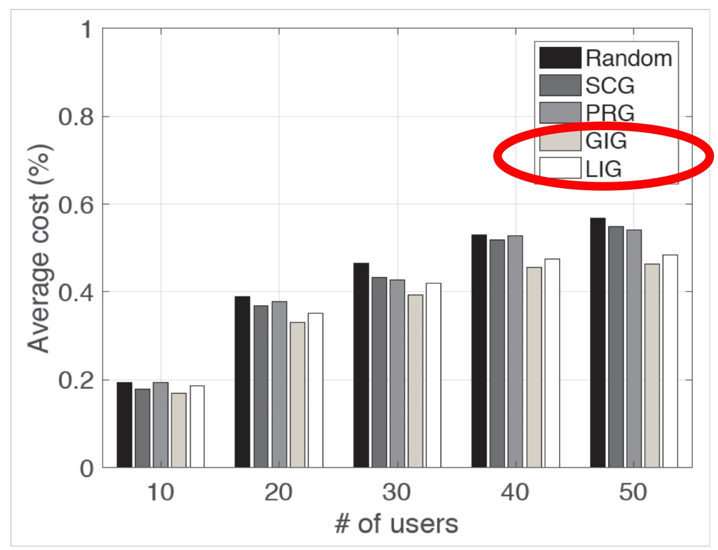
- Random: remove fog nodes iteratively by random order.
- Set-up Cost Greedy Algorithm (SCG): greedy remove fog nodes iteratively by an increasing order of the set-up cost
- Processing Rate Greedy Algorithm (PRG): greedy remove fog nodes iteratively by an increasing order of the maximum processing rate.

Experiment Results (UPFN)-Synthetic Dataset

- **Cost under the UPFN case**
  - Three Comparison algorithms
  - Lower average costs: 15.2% (LIG) and 20.3% (GIG).



(a) The fluctuation of 30 users.



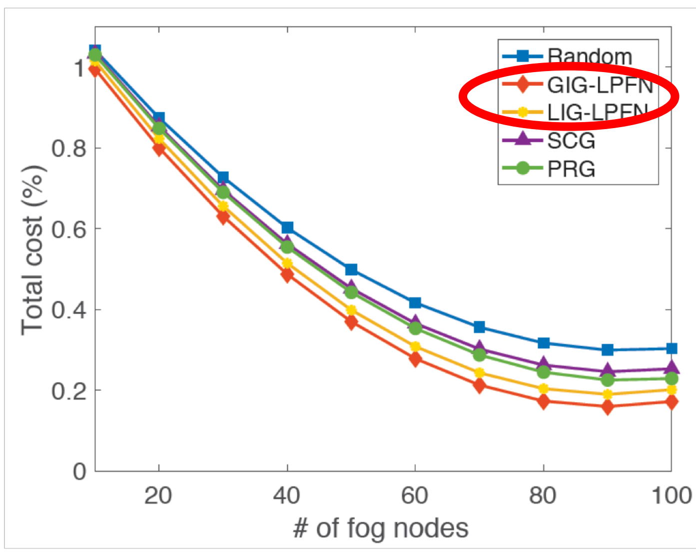
(b) Average cost of users.

Fig 4. The cost under the UPFN case.

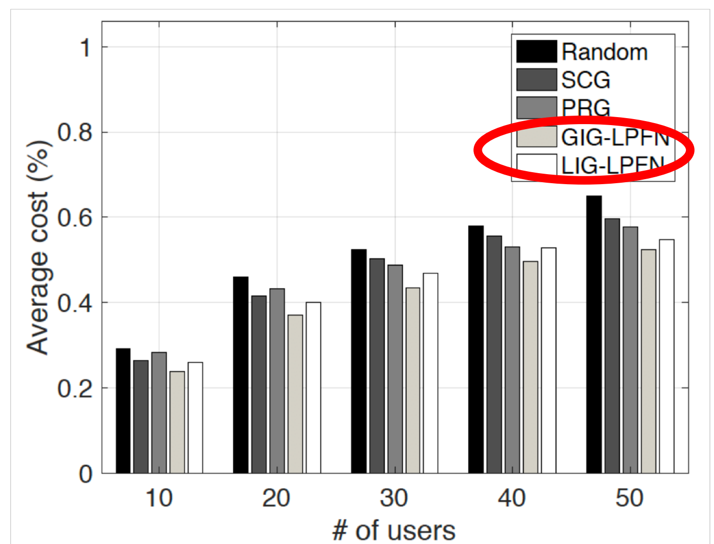
Experiment Results (LPFN)-Synthetic Dataset

Cost under the LPFN case

- Three Comparison algorithms.
- Lower average costs: 10.8% (LIG) and 14.9% (GIG).



(a) The fluctuation of 30 users.



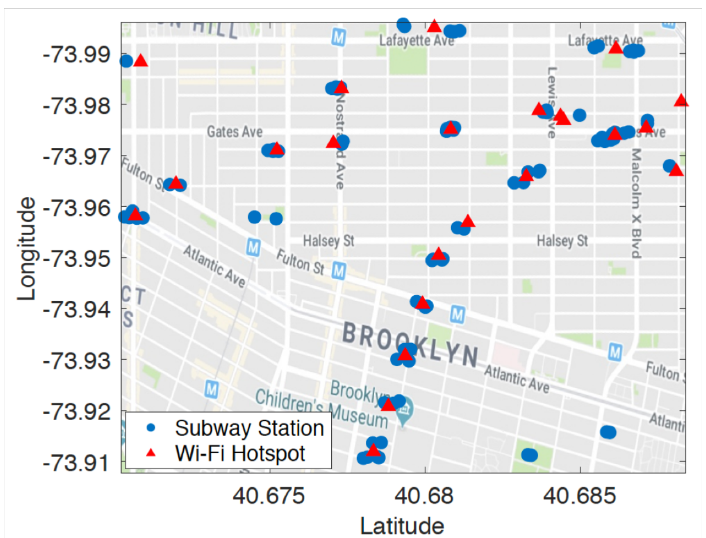
(b) Average cost of users.

Fig 5. The cost under the LPFN case.

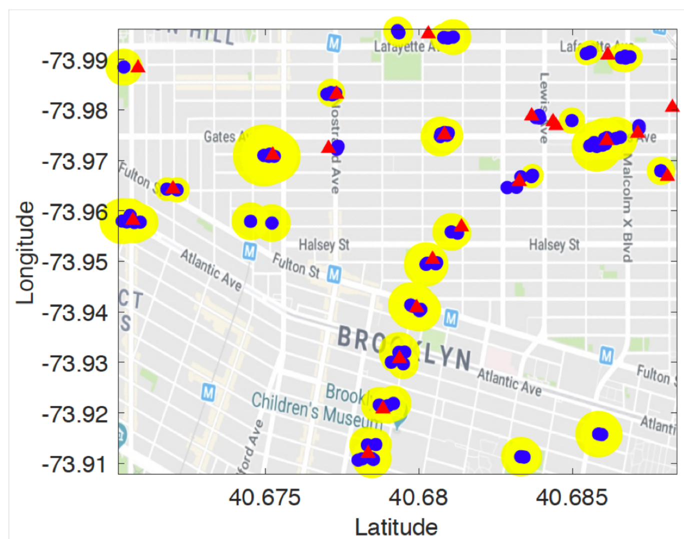
Experiment Results (LPFN)-Real Dataset

❑ Dataset

- Subway locations with users distribution in NYC.
- Combine three datasets: NYC Wi-Fi hotspot locations, entrances of the subway stations, and the transit subway entrance data.



(a) New York City



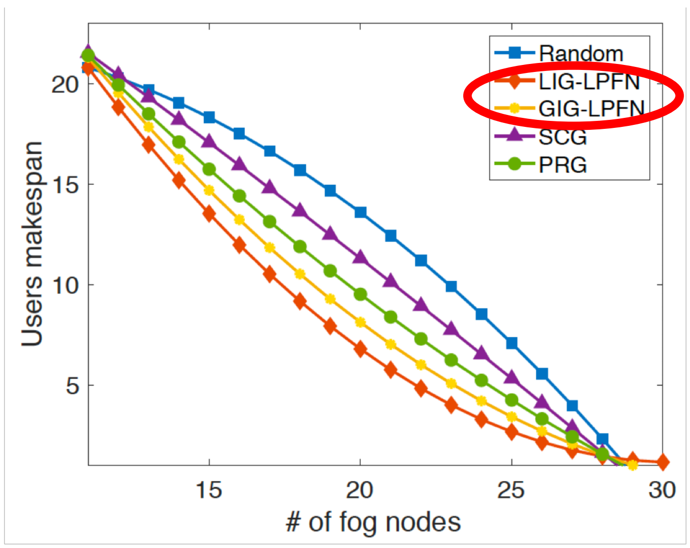
(b) Users Distribution

Fig 6. Subway locations with users distribution in NYC.

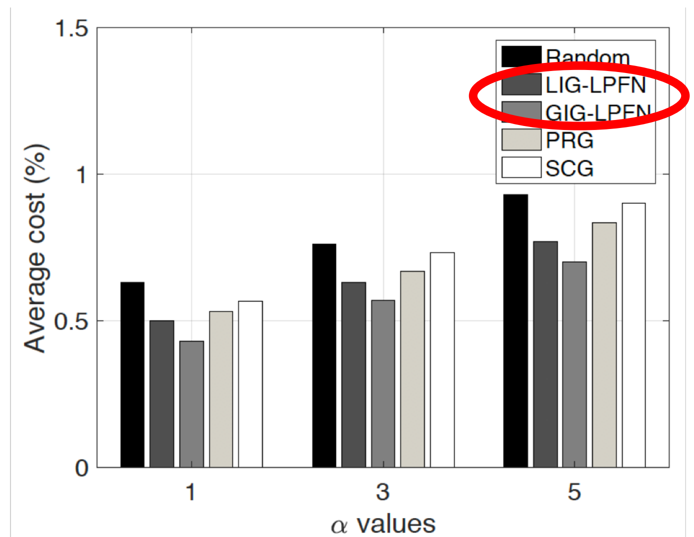
Experiment Results (LPFN)-Real Dataset

Cost under the LPFN case- read dataset

- Three Comparison algorithms
- Lower average costs: 11.4% (LIG) and 12.8% (GIG).



(a) The average makespan of users.



(b) Average cost of users.

Fig 7. The average cost of users.

6. Conclusions

Objective

- Find an appropriate scheme for the set of users with minimum cost, and support all users' demands in both resource and deadline constraints.

RP with two cases

- Unlimited-Processor Fog Node (UPFN)
- Limited-Processor Fog Node (UPFN)

Experiments

- Synthetic Dataset
- Real Dataset

Q&A