Fog: An Emerging Computing Platform for Modern Applications

Prasanta K. Jana, IEEE Senior Member

Department of Computer Science and Engineering
Indian Institute of Technology (ISM), India
E-mail: prasantajana@iitism.ac.in

Organization of the Talk

- Background
- Fog Architecture
- Resource Management and Task Offloading
- Fog Applications
- Open Research Issues

Background

- The world has witnessed a tremendous growth of IoT devices.
- Millions of devices are now connected through Internet.
- They provide services in various application domains like
  - Autonomous vehicles
  - Smart home
  - Smart city
  - Smart agriculture
  - Healthcare.
- The services should be given instantly, securely and reliably.

Some Facts:

- Approximately 60-70 billion devices will be connected to the Internet by 2022 .
- The number of smart devices per person would be 9 on an average by 2025.
- Current smart phones contain, on average, more than a dozen of sensors.
- These smart, connected devices generate data that IoT applications use to aggregate, analyze, and deliver insight.

Implication: So, we require huge storage and huge processing power.

Cloud as Supporting Technology

Cloud Computing provides

- Massive data storage
- Massive processing power
- Cost-effective solution with security and reliability.
- And Service time 24 X 7 hours

Thus Cloud Computing becomes an attractive choice for massive processing and storing a high volume of data .

Consequence: Therefore, 90% of global Internet users are now dependent on cloud-based services.

Limitations of Traditional Cloud Computing

- End-to-End delay: Distance between the cloud and the end devices causes latency issue.
- This creates hindrance to sensitive applications such as disaster management, healthcare etc.
- Traffic congestion: Due to large no. of applications.
- Energy consumption: It increases carbon footprint in the atmosphere.

Need of FOG Computing

$$Latency = T_{from\ device\ to\ cloud} + T_{data\ analysis} + T_{from\ cloud\ to\ device} \tag{1}$$

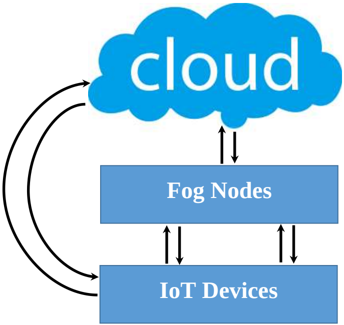


Fig 1: Fog as a middle layer

Thus we require a middle layer.

Fog as an Intermediate Layer

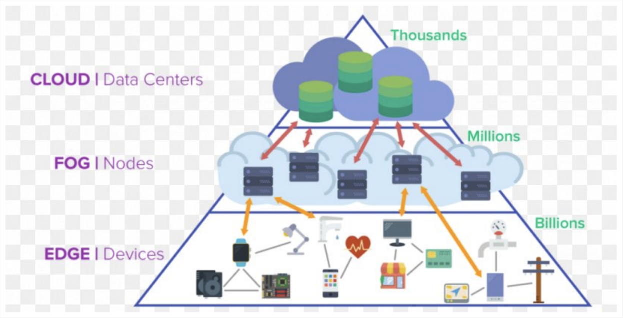


Fig. 2: Fog as an intermediate layer

- In the IoT environment, we require both cloud & fog.
- FOG layer is used for intermediate processing and transient storage.
- Sensor data is processed in the fog before it is sent to the cloud.

Aim of FOG Computing

- The idea of fog computing is to extend the cloud nearer to the IoT devices.
- The term 'fog' is used simply because "fog is a cloud close to ground", i.e., From cOre to edGe computing.
- Fog is not a substitute to cloud but it is complementary.

Primary Aim: To solve problems faced by cloud computing during IoT processing.

What is Fog Computing?

Fog computing is a highly virtualized platform that provides computing, storage, and networking services between End User(EU) and Data Centre of the traditional cloud computing.

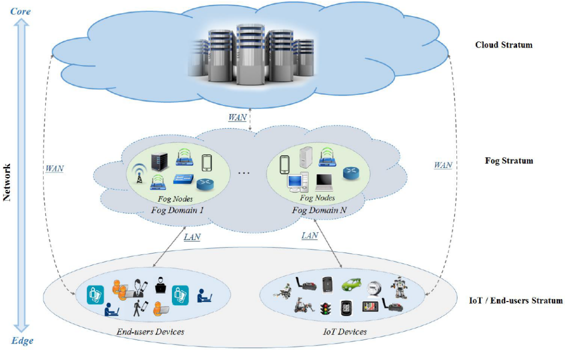


Fig. 3: The Fog System

[Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J., Polakos, P. A. (2018). A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. IEEE Communications Surveys & Tutorials, 20(1), 416–464.]

How Fog Works?

- A very large number of geo-distributed devices forms a 'mini-cloud' at the edge of the network.
- Instead of always upload/download data to/from core network, the edge devices which are in proximity, can obtain data from other users through direct link and adjacent Small Cell networks.
- In addition, the edge devices in fog network release some of their resource like computing and storage capacity to support the demands of their neighbors.
- Only the task that is not well handled by the edge devices are sent to the core cloud part for further processing.

[Mukherjee, M., Shu, L., & Wang, D. (2018). Survey of fog computing: Fundamental, network applications, and research challenges. IEEE Communications Surveys & Tutorials, 20(3), 1826-1857.]

Difference b/w Cloud & Fog Computing

Table 1: Difference b/w Cloud computing & Fog computing [1]

Features	Traditional cloud computing	Fog computing
Computing model	Centralized	Distributed fog nodes are controlled in both distributed and centralized manner
Deployment cost	High due to sophisticated planning	Low, fog enables ad-hoc deployment with or without planning
Resource optimization	Global	Local
Size	Cloud data centers are very large in size	Smaller, however, a large number of small fog nodes form a large fog system.
Mobility management	Easy	Hard
Latency	High	Very low
Operation	Operated by large companies	Often operated by small companies, however large companies can operate depending on the size
Reliability	High	Low
Maintenance	Operated and maintained by technical experts	Generally requires no or little human involvement
Applications	Cyber-domain applications	Support both cyber-domain and cyber-physical applications, most importantly time-critical applications

Characteristics of Fog Nodes

- **Storage:** Transit
- **Computing facility:** To process data before it sends to cloud.
  - To take quick decision.
- **Network connectivity:** To connect end devices and fog nodes
- **Examples of fog devices:** Servers, Routers, Switches and Access points.

Types of Data Handled

- **Very-time-sensitive data:** Data which should be analysed within a fraction of second, e.g., Surveillance system, Fire detection etc.
- **Less-time-sensitive data:** Data which should be analysed after second or minutes, e.g., Non-critical healthcare data.
- **Non-time-sensitive data:** Data which can wait for hours, day or a week, e.g., prediction of air pollution, Customers' purchase behaviour.

Features of Fog Computing

- Low latency and location awareness
- Supports geographic distribution
- End device mobility
- Capacity of processing high number of nodes
- Wireless access
- Real-time applications
- Heterogeneity

Difference b/w Related Technologies

BASIC DIFFERENCE BETWEEN EDGE COMPUTING TECHNOLOGIES

Features	Cloudlets	MEC	Fog computing
Applications	Mobile offloading	Focus on the applications that are better suited for both mobile and non-mobile network edge	Support a wider range of latency-sensitive applications for resource-constraint end devices
Virtualization	Only depends on VMs	Can use other technologies apart from VMs	Other virtualization technologies can be used
Operational Mode	Can work in standalone mode	Can work in standalone and can connect to the cloud	Cannot work in standalone, need the support of cloud

Fig. 5: Difference B/W Related Technologies

[Mukherjee, M., Shu, L., & Wang, D. (2018). Survey of fog computing: Fundamental, network applications, and research challenges. IEEE Communications Surveys & Tutorials, 20(3), 1826-1857.]

Fog Computing Use Cases

A Fog System Use Case for CDN

- A Content Delivery Networks (CDN) consists of origin servers, surrogate servers (also known as replica servers), and a controller.
- The origin servers store the original content and this content is replicated on the surrogate server(s).
- For each end-user request, the controller selects the most appropriate surrogate server which provides the service to the requester.

Use Case 1: Video Streaming in CDN

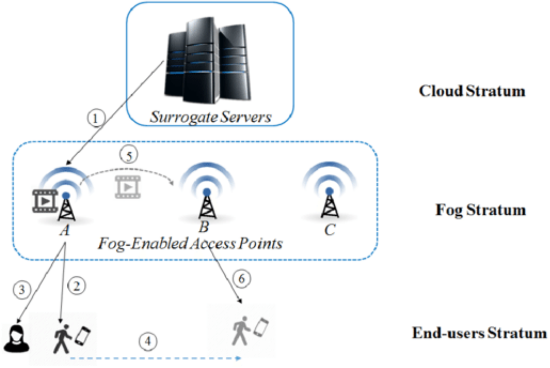


Fig. 6: A Fog System Use Case for CDN

[Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J., Polakos, P. A. (2018). A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. IEEE Communications Surveys & Tutorials, 20(1), 416–464.]

Use Case 2: Fire Monitoring & Detection

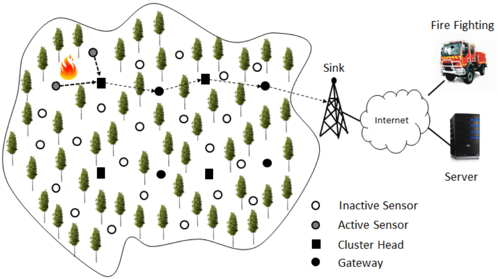


Fig. 7: WSN architecture for fire Monitoring and Detection

Use Case 2: Fire Detection and Fighting

A Fog System Use Case for Fire Detection and Fighting

- Fire detection monitors geographic areas (e.g., city, forest) and dispatches fleets of robots when fire is detected.
- The monitoring is done through the gathering of information such as wind speed, moisture, and temperature.
- When fire is detected, the application evaluates its intensity and contour and dispatches the most appropriate robots to extinguish it.

Use Case 2: Fog System

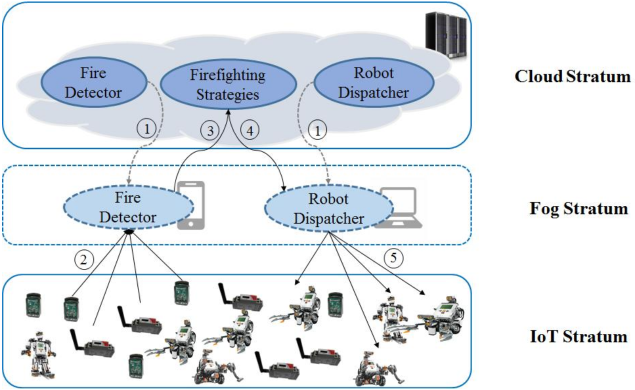


Fig. 8: A Fog System Use Case for Fire Detection and Fighting

[Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J., Polakos, P. A. (2018). A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. IEEE Communications Surveys & Tutorials, 20(1), 416–464.]

Fog Computing-based Architecture

Three-tier Architecture

- **Tier-1 Things/End Devices:** This layer contains IoT devices, End User smart devices (e.g., smartphones, tablets, smart cards, smart vehicles, and smartwatch), also known as terminal nodes.
- **Tier-2 Fog:** This is fog computing layer comprised of network devices such as a router, gateway, switch, and Access Points. These fog nodes can collaboratively share storage and computing facilities.
- **Tier-3 Cloud:** Traditional cloud servers and cloud Data Centre reside in the top-most tier. This tier has sufficient storage and computing resources.

[Mukherjee, M., Shu, L., Wang, D. (2018). Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges. IEEE Communications Surveys & Tutorials, 1–1.]

Three-tier Architecture

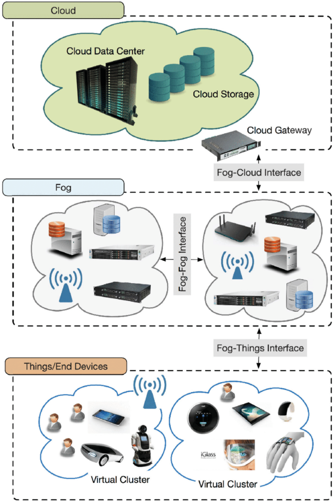


Fig. 9: Three-tier Architecture [1]

Layered Architecture

Layered architecture for fog computing

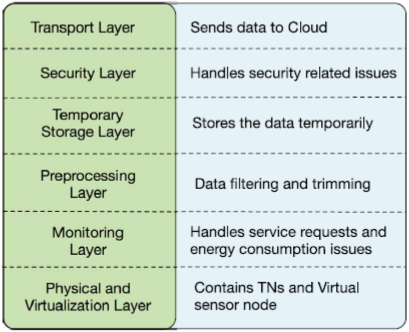


Fig. 10: Layered Architecture [1]

An Architecture of Combined Fog-Cloud (CFC)

- The *Bottom layer* mainly consists of Terminal nodes.
- Fog type 1: Composed of low capacity fog servers with an aim to provide delay-constraint services requests.
- Fog type 2: Fixed nodes for collaborative sharing in neighbourhood area. This layer benefits a medium no. of service request with low delay when there are insufficient resources in a lower 1st layer with 1-hop connection.
- The Upper layer: Contains the traditional cloud servers with enough computing resources, however, at the cost of higher latency to the terminal nodes.

Combined Fog-Cloud (CFC) Architecture

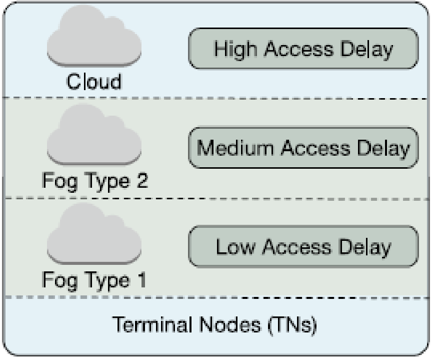


Fig. 11: An architecture of Combined Fog-Cloud [1]

Fog System Research

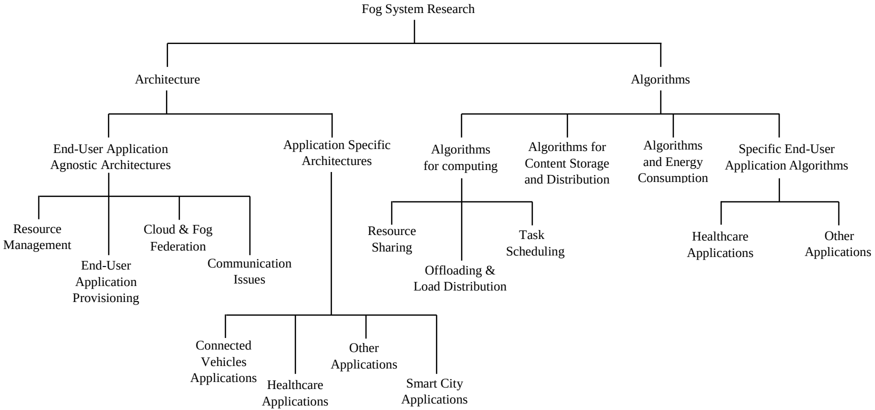


Fig. 4: Flow diagram of fog system research

[Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J., Polakos, P. A. (2018). A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. IEEE Communications Surveys & Tutorials, 20(1), 416–464.]

Task Offloading

What is Task Offloading in Fog computing?

- It refers to transferring part of the TNs' task to other entities such as fog nodes or cloud server so that they can process and send back the results quickly to the TNs.

What is the main goal?

- To reduce the burden of the cloud servers.

Why task offloading?

- Because the tasks may be very large and computation intensive and the TNs may have limited processing power.

Offloading Destinations

Task offloaded from a TN to an FN:

- Here, the task requires less storage, less computation and is time-sensitive.

Task offloaded from one FN to another FN:

- Due to less resource availability, an FN can choose to offload its task partially or fully to its nearby resource-rich FN.

Task offloaded from an FN to the cloud:

- When the execution of a task demands large storage and computation of data at an FN.

FAQs Continued...

What QOS parameters to be fulfilled?

- Minimum delay.
- Minimum energy consumption.
- Minimum cost.
- Maximum reliability.

What are the Advantages?

- Proper utilization of idle resources,i.e., load balancing.
- Proper energy management.
- Proper latency management.

Service Allocation & Resource Management

- So it becomes an important issue how to decide which task or service should be processed in the fog layer or be forwarded to the cloud.
- In the fog computing literature, this is also referred as Service Allocation & Resource Management

Resource Management Issues

So how to find out the target fog node?

Following policies are suggested [17]:

- **Random Policy:** A fog node is randomly and uniformly selected without considering any other factors.
- **Lowest Latency Policy:** A fog node that provides minimum total latency at the current state of the system is selected for the given workload.
- **Maximum Available Capacity Policy:** This policy selects a fog node which has maximum remaining resource.

A. Energy Consumption

- The impact of CO_2 emission by cloud servers is always a vital issue.
- Fog computing is always a viable solution towards reduced power consumption.
- This leads the issue of the selection of Energy Efficient Applications
- However, there must be a tradeoff between power consumption and transmission delay

Other Resource Management Issues

B. Resource Sharing

- To provide services to the EUs with benifits, all the resources need to be shared.

C. Proactive Caching

- The content of the upcoming tasks are predicted and prefetched during the current task computation.
- It reduces the burden on access link by the computing results in advance without any prefetched content of the tasks.

Requirement while Designing Task Offloading Algorithm

A. Model Descriptions

- Fog model
- Energy Model
- Latency Model

B. Problem Formulation.

- Possibly Constrained optimization.
- LPP, ILP, Non-LPP.

C. Solution and Performance Measurement

- Proposed Method
- Simulation Results
- Comparison with state-of-arts Algorithms

A Fair and Energy-minimized Task Offloading (FEMTO) Algorithm

Objective: To minimize the overall energy consumption during task offloading services in a fog-enabled IOT networks while taking care of transmission delay.

[Zhang, G., Shen, F., Liu, Z., Yang, Y., Wang, K., Zhou, M.-T. (2018). FEMTO: Fair and Energy-Minimized Task Offloading for Fog-Enabled IoT Networks. IEEE Internet of Things Journal, 1–1.]

FOG Model

A fog cluster consisting of N heterogeneous FNs

A virtual controller providing task offloading service to TNs. The active FNs are cable-powered. The passive FNs are battery powered and so sensitive to energy consumption.

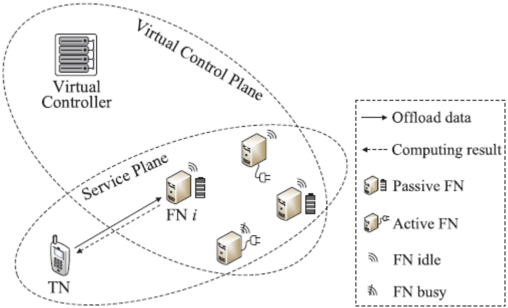


Fig. 12: Fog Model with heterogeneous FNs, TNs and a controler

Task Offloading Problem

- An l-bit task with delay bound d_{max} is generated at the TN.
- The TN may partly process it and may call for task offloading service from the virtual controller.
- The controller of this fog cluster assigns an idle FN say FN i.
- Then, the task is divided into two subtasks with size of l_T and l_i.
- l_T bits are processed locally and l_i bits are offloaded to FN i.
- Finally, the processing result is transmitted back from FN to the TN.

Task Offloading

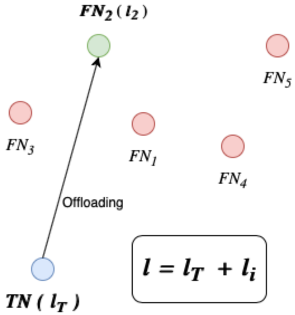


Fig. 13 :Problem Scenario

Summary of Key Notations

Table 2: SUMMARY OF KEY NOTATIONS

Nota.	Unit	Description
l	bit	Overall task size of the TN.
$d_{\max}$	s	Upper bound of the task delay.
$p_{\max}$	J/s	Upper bound of the transmission power of the TN.
l_{T}	bit	Subtask size processed locally at the TN.
l_i	bit	Subtask size offloaded to FN i.
η_{T}	cycle/bit	CPU cycles for processing 1 bit data at the TN.
η_i	cycle/bit	CPU cycles for processing 1 bit data at FN i.
f_{T}	cycle/s	CPU frequency of the TN.
f_i	cycle/s	CPU frequency of FN i.
θ_{T}	J/cycle	Energy consumption per CPU cycle of the TN.
θ_i	J/cycle	Energy consumption per CPU cycle of FN i.
ρ_i	—	The probability that FN i is idle.
W	Hz	Spectrum bandwidth for task offloading.
p_i	J/s	Transmission power of the TN to FN i.
γ_i	—	Path loss factor between the TN and FN i.
β_i	—	Shadowing factor between the TN and FN i.
m_i	—	Scheduling metric of FN i.

Latency Model

Task delay includes two parts:

1. Local processing delay of the subtask with l_T bits (d_T)
2. The offloading delay of the subtask with l_i bits (d_i)

Overall task delay:

$$d = max(d_T, d_i) \tag{11}$$

where d_T is the local processing delay:

$$d_T = \frac{l_T \eta_T}{f_t} \tag{12}$$

and d_i is the offloading delay: d_i

$$d_i = \frac{l_i}{WB_i} + \frac{l_i \eta_i}{f_i} = l_i \left(\frac{1}{WB_i} + \frac{\eta_i}{f_i} \right) \tag{13}$$

Energy Consumption:

$E =$ Energy consumed by TN in processing l_T bits and transmission of l_i bits to FN$_i$ $\quad + \quad$ Energy consumed by FN$_i$ in processing l_i

$$E = E_T + E_i = l_T \eta_t \theta_T + \frac{l_i p_i}{W B_i} + l_i \eta_i \theta_i \tag{14}$$

where

$$B_i = \log_2 \left(1 + \frac{(p_i \gamma_i \beta_i)}{I_i + W N_o} \right)$$

is the spectral efficiency of the wireless link.

$$l = l_t + l_i \tag{10}$$

Overall task delay

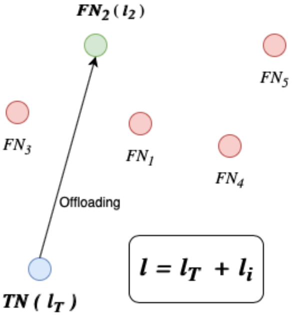
$$d = max\left[\frac{(l - l_i)\,\eta_T}{f_T}, l_i\left(\frac{1}{W\log_2\left(1 + \frac{p_i \Upsilon_i \beta_i}{I_i + W N_0}\right)}\right) + \frac{\eta_i}{f_i}\right] \tag{16}$$

Overall energy consumption

$$E = l\eta_T\theta_T + l_i\left[\frac{p_i}{W\log_2\left(1 + \frac{p_i \gamma_i \beta_i}{l_i + W N_0}\right)} + (\eta_i\theta_i - \eta_T\theta_T)\right] \tag{17}$$

FN Selection

- We have to offload the subtask l_t to a single FN and when it is idle.
- However, l_i and p_i vary from FN to FN.



Problem Formulation

We are supposed to select that FN i which minimizes total energy consumption, i.e.,

$$\min_{l_i, p_i} \left[l\eta_T\theta_T + l_i \left(\frac{p_i}{W \log_2(1 + \frac{p_i \gamma_i \beta_i}{I_i + WN_o})} + (\eta_i\theta_i - \eta_T\theta_T) \right) \right],$$

$$\text{s.t. } d \leq d_{max}$$

$$0 \leq p_i \leq p_{max}$$

$$0 \leq l_i \leq l$$

Task Offloading based on Clustering

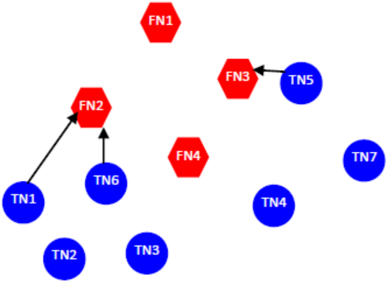


Fig. 13: Problem Scenario

Task Offloading based on Clustering

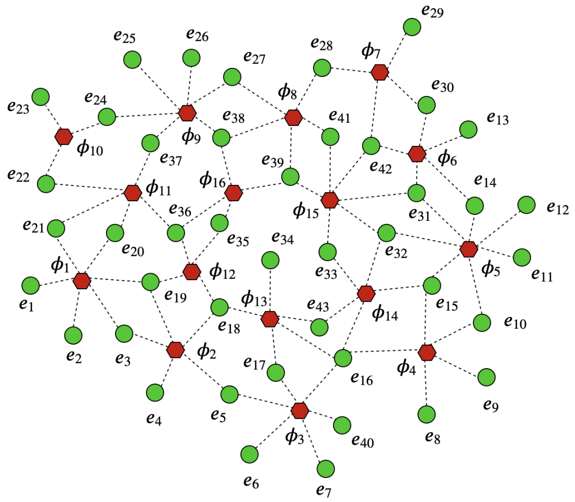


Fig. 15: Scenario Before Clustering

Task Offloading based on Clustering

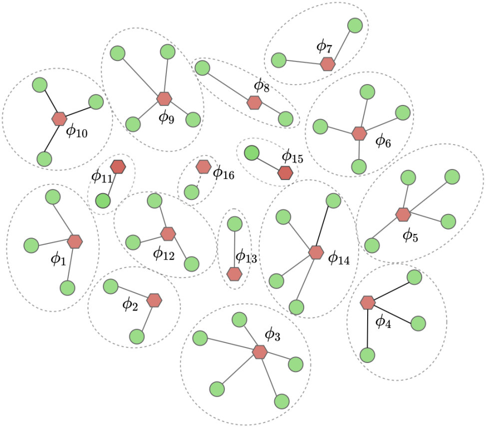


Fig. 16: Scenario After Clustering

IOT Applications of FOG Computing

Smart Cities:

Fog computing can be widely adopted for:

- Smart-transportation
- Waste management
- Water management
- Greenhouse gas control

Smart Cities

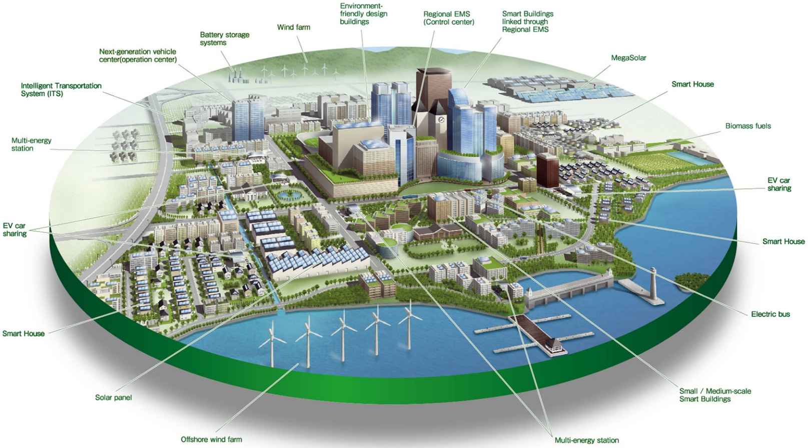


Fig. 17:Smart City

Applications Continued...

Intelligent Transportation System:

- Urban environments are now overcrowded with vehicles, traffic congestions, accidents, and pollution.
- These need to control, which depend on real-time feedback from road-side infrastructure, traffic flow, and passengers' activities.
- Fog computing is well suited in Vehicular Ad-hoc Networks (VANET) for local decision-making mechanism.

Intelligent Transportation System

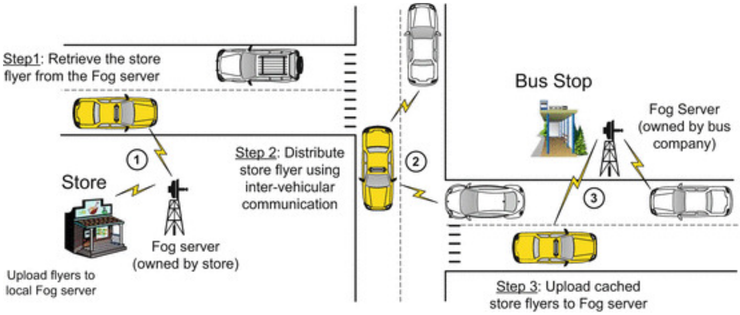


Fig. 18: Intelligent Transportation System

Applications Continued...

Healthcare:

- To take care of health of old people. EEG data is preprocessed in the smartphones.
- Monitoring of patients and activities of nurses in a hospital.
- A smartphone-based service, called an Emergency Help Alert Mobile Cloud (EHAMC) has been introduced.

Healthcare

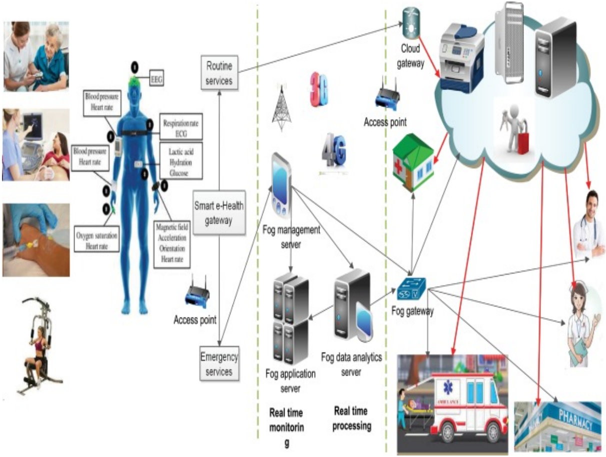


Fig. 19: Intelligent Health Care System

Open Research Issues I

1. **Application Offloading**

   Application offloading is not always efficient due to lack of available resources in fog layer.

2. **Resource Management**

   More efficient resource management as the fog computing does not enough computing and storage resources.

3. **Heterogeneity**

   Bottom-most layer of FG consists of various end devices with heterogeneity of data collection, data format, and data processing capability.

4. **Security**

   Standardization is required so that various IoT systems can securely interact each other and cloud server

Open Research Issues II

5. **SDN-Based Fog Computing**

   SDN-based fog computing, several research issues arise as follows:

   Local Coordinator: A local coordinator required to handle the dynamic service requirement.

   Controller Design: The fog-SDN controllers need to cooperate among themselves to efficiently handle the limited fog resources.

   Implementation of Test-bed: Due to centralized nature, it is challenging to implement at the edge switches.

6. **Machine Learning for FOG:**

   Hybrid Deep Learning Framework for smart manufacturing industrial cyber-physical system for Industry 4.0 applications
   Machine learning-Based traffic offloading in fog networks

Open Research Issues III

7. **Other Issues:**

   Reliability
   Real time analysis
   Mobility of fog nodes/ IoT devices

References

1. Mukherjee, M., Shu, L., Wang, D. (2018). Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges. IEEE Communications Surveys & Tutorials, 1–1.
2. Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R. H., Morrow, M. J., Polakos, P. A. (2018). A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. IEEE Communications Surveys & Tutorials, 20(1), 416–464.
3. Cisco Delivers Vision of Fog Computing to Accelerate Value From Billions of Connected Devices, Cisco, San Jose, CA, USA, Jan. 2014.
4. X. Zhao, L. Zhao, K. Liang, An energy consumption oriented offloading algorithm for fog computing, in: International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Springer, 2016, pp. 293–301.
5. K. Liang, L. Zhao, X. Chu, H.-H. Chen, An integrated architecture for software defined and virtualized radio access networks with fog computing, IEEE Netw. 31 (1) (2017) 80–87.
6. C. Fricker, F. Guillemin, P. Robert, G. Thompson, Analysis of an offloading scheme for data centers in the framework of fog computing,ACMTrans. Model. Perform. Eval. Comput. Syst. 1 (4) (2016) 16.
7. R. Hasan, M. Hossain, R. Khan, Aura: An incentive-driven ad-hoc IoT cloud framework for proximal mobile computation offloading, Future Gener. Comput. Syst. (2017).
8. G. Orsini, D. Bade, W. Lamersdorf, Computing at the mobile edge: Designing elastic android applications for computation offloading, in: IFIP Wireless and Mobile Networking Conference, WMNC, 2015 8th, IEEE, 2015, pp. 112–119.
9. L. Pu, X. Chen, J. Xu, X. Fu, D2d fogging: An energy-efficient and incentiveaware task offloading framework via network-assisted D2D collaboration, IEEE J. Sel. Areas Commun. 34 (12) (2016) 3887–3901.
10. K. Habak, M. Ammar, K.A. Harras, E. Zegura, Femto clouds: Leveraging mobile devices to provide cloud service at the edge, in: Cloud Computing, CLOUD, 2015 IEEE 8th International Conference on, IEEE, 2015, pp. 9–16.
11. Y. Wang, M. Sheng, X. Wang, L. Wang, J. Li, Mobile-edge computing: Partial computation offloading using dynamic voltage scaling, IEEE Trans. Commun. 64 (10) (2016) 4268–4282.
12. R. Craciunescu, A. Mihovska, M. Mihaylov, S. Kyriazakos, R. Prasad, S. Halunga, Implementation of fog computing for reliable e-health applications, in: Signals, Systems and Computers, 2015 49th Asilomar Conference on, IEEE, 2015, pp. 459–463.
13. Mohammad Aazam, Sherali Zeadally, Khaled A. Harras, Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities, Future Generation Computer Systems 87 (2018): 278-289. Web. 28 Dec. 2019.
14. Zhang, G., Shen, F., Liu, Z., Yang, Y., Wang, K., Zhou, M.-T. (2018). FEMTO: Fair and Energy-Minimized Task Offloading for Fog-Enabled IoT Networks. IEEE Internet of Things Journal, 1–1.
15. S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of Internet of things," IEEE Trans. Cloud Comput., vol. 6, no. 1, pp. 46–59, Jan./Mar. 2018.
16. Gao, L., Luan, T. H., Liu, B., Zhou, W., & Yu, S. (2017). Fog computing and its applications in 5G. In 5G Mobile Communications (pp. 571-593). Springer, Cham.
17. Intharawijitr et al. , "Analysis of fog model considering computing and communication latency in 5G cellular networks," Proc. IEEE Int. PerCom Workshops, Mar. 2016, pp. 1–4.

Questions


Thank you