Fog Computing: Concepts, Challenges and Research Scope

Chinmaya Dehury

Chinmaya.dehury@ut.ee

07 Dec 2020
Outline

• Fog computing
• Fog computing challenges
• Fog placement strategies
• Future research directions
Layers of Cloud-centric IoT

Remote Cloud-based processing

Connectivity nodes & Embedded processing

Sensing and smart devices
Issues with Cloud-centric IoT

- The cost of a cloud-only solution is too high
  - To run a large scale IoT system with >1000 geo-distributed devices [FogFlow]
- **Latency** issues for applications with sub-second response requirements
  - Health care scenarios
  - Smart cities and tasks such as surveillance need real-time analysis with strict deadlines
  - Many IoT services require <10ms end-to-end latency
- **Network load**
- Certain scenarios do not let the data move to cloud
  - Better security and deeper insights with privacy control
Fog Architecture

Large-scale distributed data analytics
- e.g. MapReduce, Flink, Spark

Edge Analytics
- Filtering, consolidation, error detection, anonymization

Fog Nodes
- Private Clouds, Cisco IOx, Switches, Cloudlets

Gateways
- Access points, Mobile Phones, Single Board Computers, CPE

Edge Nodes
- Sensors, Actuators, Mobile Phones, Smart Vehicles, …
Fog Computing

• Processing across all the layers, including network switches/routers

[Chang et al, AINA 2017; FEC 2019; Mass et al, SCC 2016; Liyanage et al, PDCAT 2016]
Fog applications

- Multi-media applications
  - Finding best data streaming bit rate in different scenarios
    - Such as in video surveillance applications
  - Adjusting video encoding rate (video processing speed) based on network load
- Sensor data filtering and preprocessing
- Sensing frequency calibration
- Sensor data prediction
- Interactive games
  - Electroencephalogram (EEG) Tractor Beam game
  - Real-time EEG signal analysis and brain state (concentration) prediction

[Zao et al, Frontiers in Human Neuroscience 8:370]
Types of fog computing applications

• Top-down applications
  – Applications that are managed by a cloud provider
  – Cloud provider deploys the necessary geo-distributed resources
  – Cloud providers manage the IoT application execution across the fog topology
  – Consider proximity and QoS parameters such as user load, latency, cost models etc.

• Bottom up applications
  – Applications that are managed by individual service providers
  – Fog resources are provided by different vendors
    • E.g. General public, private clouds, network operators
  – Gateways decide how to schedule the components across the fog topology [Dehury et al, iiWas2019; Dehury et al, ccgrid2020]
Advantages of Fog Computing

• **Security**
  – Supports additional security to IoT devices to ensure safety and trustworthiness in transactions

• **Cognition**
  – Enables fog providers the awareness of the objectives of their clients toward supporting autonomous decision-making
    • E.g. where and when to deploy computing, storage, and control functions

• **Agility**
  – Brings the opportunity to individual and small businesses to participate in providing FEC services

• **Latency**
  – Provide rapid responses for the applications that require ultra-low latency

• **Efficiency**
  – Reduces the unnecessary cost of outgoing communication bandwidth
  – Consume minimum power for data offloading and processing when compared to CIoT model

[Chang et al, FEC 2019; Dehury et al, iiWas2019; Dehury et al, ccgrid2020]
Research Challenges in Fog Computing

• Frameworks for establishing fog setups
• Fog resource provisioning
• Dynamic fog computing service discovery and accessing
• Fog execution frameworks
• Fog application placement
Research Challenges in Fog Computing

• Mobility of fog devices
• Multi-Fog environment
• Privacy
• Security
• Authentication and Trust Issues
Quality of Service (QoS) vs Experience (QoE)

• QoS - refers to the overall features of system services which help to meet the stated and implied needs of the end users [ITU]
  – QoS drives through an agreement between user and provider that strongly monitors technical attributes of system services
  – Cost, service delivery deadline, packet loss ratio, jitter, throughput, etc.
  – E.g. Downloading a particular file in max 5 min time

• QoE is the total acceptability of a service that is determined by subjective perception of the end users [ITU]
  – Encapsulates user’s requirement, intentions and perceptions while provisioning system services
  – E.g. 2 users require that file in 3 min and 7 min respectively

• End users perceived QoE can degrade the acceptability of a service greatly even when the proper QoS is maintained

ITU - International Telecommunication Union
IndieFog

• Indie Fog [Chang et al, IEEE Computer 2017]
  – System architecture for enabling fog computing with customer premise equipment

• Proactive fog computing using resource-aware work-stealing [Soo et al, IJMCMC 2017]

• Dynamic fog computing service discovery and accessing
Fog application placement strategies

• Resource intensive tasks of IoT applications can be placed across the Fog topology

• How to decide to which node the task to be offloaded?
  – Based on quality of service (QoS) parameters such as latency, resources, cost etc.
  – Latency and cost-aware [Mahmud et al, JPDC 2020] application module management

• The problem can also be formulated as multi-objective offloading strategy [Adhikari et al, IEEE IoTJ 2020]
  – Latency and resource management
  – Need to consider the types of the jobs and their priority
  – Need to find ideal heuristics, metaheuristics etc.
  – Also have to consider the graph topology of the Fog nodes

QoS – Quality of Service
QoE – Quality of Experience
Quality of Experience (QoE)-aware Placement of Applications in Fog Computing Environments

- Fuzzy logic based approach that prioritizes different application placement requests
  - Based on the user expectations (Rating of Expectation)
- Fuzzy logic based approach that classifies Fog computational instances
  - Based on current status of the instances (Capacity Class Score)
- A linearly optimized mapping of application placement requests to Fog computing instances
  - To ensure maximized QoE-gain of the user

[Mahmud et al, JPDC 2019]
DPTO: A Deadline and Priority-aware Task Offloading in Fog Computing Framework Leveraging Multi-level Feedback Queueing

- **Task Priority 1:**
  - Class that supports the delay-sensitive tasks with hard deadlines
  - No negotiation on deadline

- **Task Priority 2:**
  - Tasks have intermediate priority with soft-deadline
  - Tasks meet their deadline with a negotiation and penalty

- **Task Priority 3:**
  - Lowest priority class that aims to support the resource-intensive tasks
  - No deadline
  - Mostly offloaded to the cloud

**Goal:** To minimize the overall queueing waiting time and offloading time of the real-time tasks while meeting the deadline and resource constraints.

[Adhikari et al, IEEE IoTJ 2020]
Service dispersal mechanism for fog and cloud computing

• Service is delivered by both fog and cloud
• Service Slicing
• What percentage of a service should be delivered by fog and by cloud?
• Parameters
  – Service Latency
  – Resource demand fulfilled by fog
  – Computation latency
  – User’s distance
  – User’s priority

[Dehury et al, iiWas2019; Dehury et al, ccgrid2020]
Service dispersal mechanism for fog and cloud computing

• Multi fog environment
• Use of AI tool to efficiently distribute the slices among fog nodes and cloud
• Challenges:
  – Experimenting in multi-fog environment
  – incorporating serverless environment

[Dehury et al, iiWas2019; Dehury et al, ccgrid2020]
HeRAFC: Heuristic Resource Allocation and Optimization in MultiFog-Cloud Environment

- The infrastructure and applications are modelled using graph theory
- FCI (Fog-Cloud Interface) is introduced, where a fog node can communicate with cloud and other fog nodes as well
- A task can be assigned to other fog nodes at multi-hop distance
- Heuristic approach is followed to achieve the objective of resource utilization maximization

[Dehury and Srirama, TSC 2020 (Under review)]
Fog Computing – Research Challenges - continued

• Process-driven Edge Computing in Mobile IoT
  [Mass et al, IoTJ 2019; CASA 2018; Chang et al, CSUR 2016]
Fog Computing – Research Challenges - continued

- Mobility also becomes critical in Fog computing [Mass et al, IoTJ 2019]

- STEP-ONE : Simulated Testbed for Edge Processes based on the Opportunistic Network Emulator [Mass et al, JSS 2020 (under review)]
  - Extended the ONE simulator to simulate the Fog computing mobility aspects
  - Process execution based on Flowable BPMS
STEP-ONE
Future Research Scope

- Standards-based dynamic deployment and management of fog applications
  - Deployment should consider heterogeneity, mobility aspects, multiple ownership (interoperability) etc.
  - Cloud computing community addressed similar problem through OASIS TOSCA
  - Extending TOSCA to support fog-based applications
  - Develop relevant orchestrators
- Efficient algorithms for self-adaptive resource provisioning and QoS-aware application module scheduling in fog topology
- Reliable and data-locality preserving real-time data processing in fog environments [Dehury et al, FGCS 2020; Dehury et al, ECSA 2020]
  - Controlling the dataflow of distributed data processing applications across the multi-layer fog
  - Restructure the IoT applications into dynamically composable multi-stage data pipelines
  - Actor programming model and serverless functions edge analytics tasks
  - Apache Nifi data pipelines for task composition
- IoT case studies in domains such as smart city, smart healthcare, smart agriculture etc.
References


07/12/2020
Chinmaya Dehury