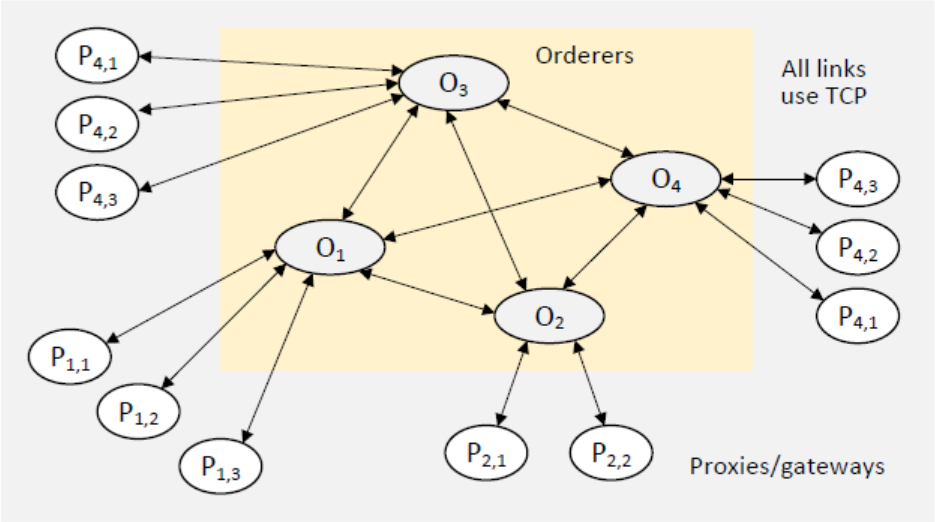
Consensus for large multi-tiered IoT systems

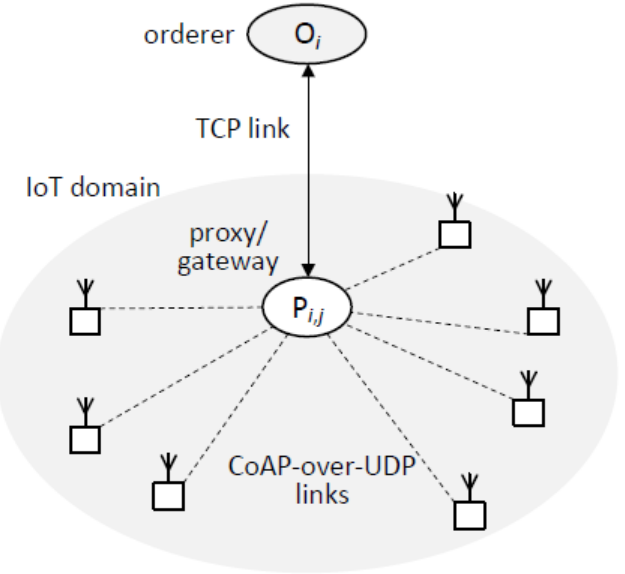
Jelena Misic, Toronto Metropolitan University

Overview

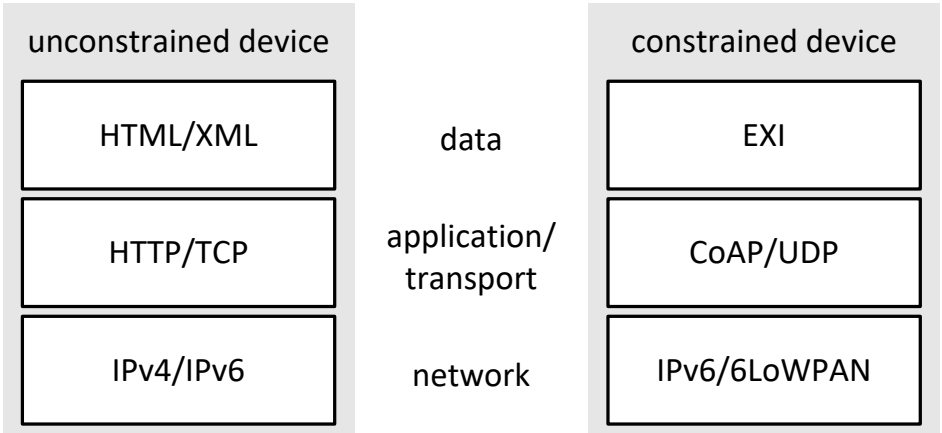
- Motivation: IoT data may need replicated ledger (blockchain) and consensus
- System architecture
- Q: How to cover larger distances ?
- Q: How to achieve multiple entry PBFT?
- Q: how to achieve multi tier, multiple entry PBFT



(a) Network topology for orderers and proxies.



(b) IoT domain architecture.



IoT data may need Consensus in replicated ledger

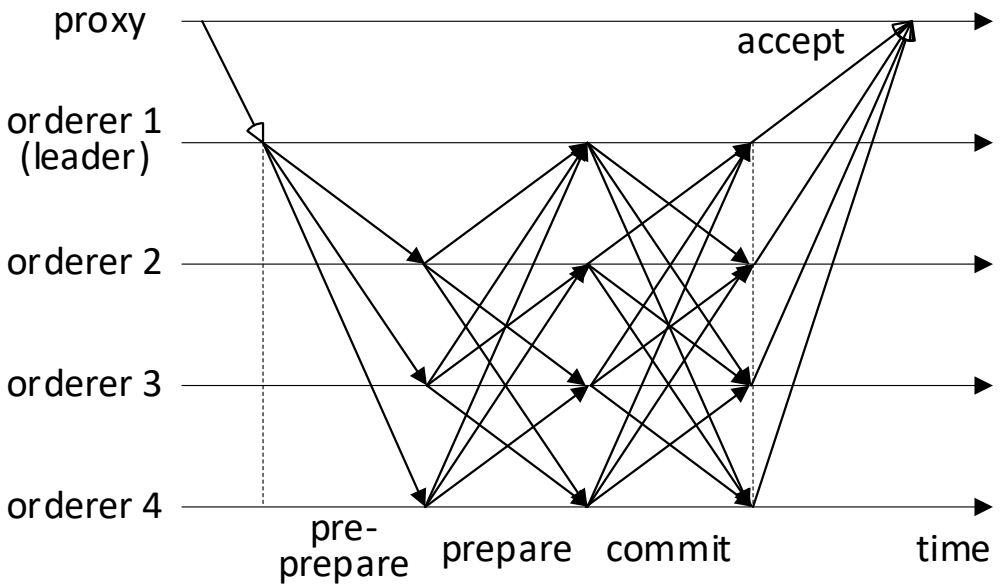
- Blockchain technology is increasingly used in Internet-of-Things (IoT) applications
- POW consensus is unsuitable due to its low throughput, power consumption and heterogeneity of data.
- Practical Byzantine Fault Tolerance (PBFT) is a viable choice, but:
  - It is originally intended for scenarios with physically close ordering nodes – unsuitable for IoT systems that span large geographical areas
  - It is sensitive to the behavior of the leader (primary node)
- To extend PBFT to such cases, it has to be augmented with multiple-entry capability so that any orderer node can lead a consensus round.
  - Contention resolution
  - Hint: CSMA/CA or ordering of proposers by block hashes can help

System architecture

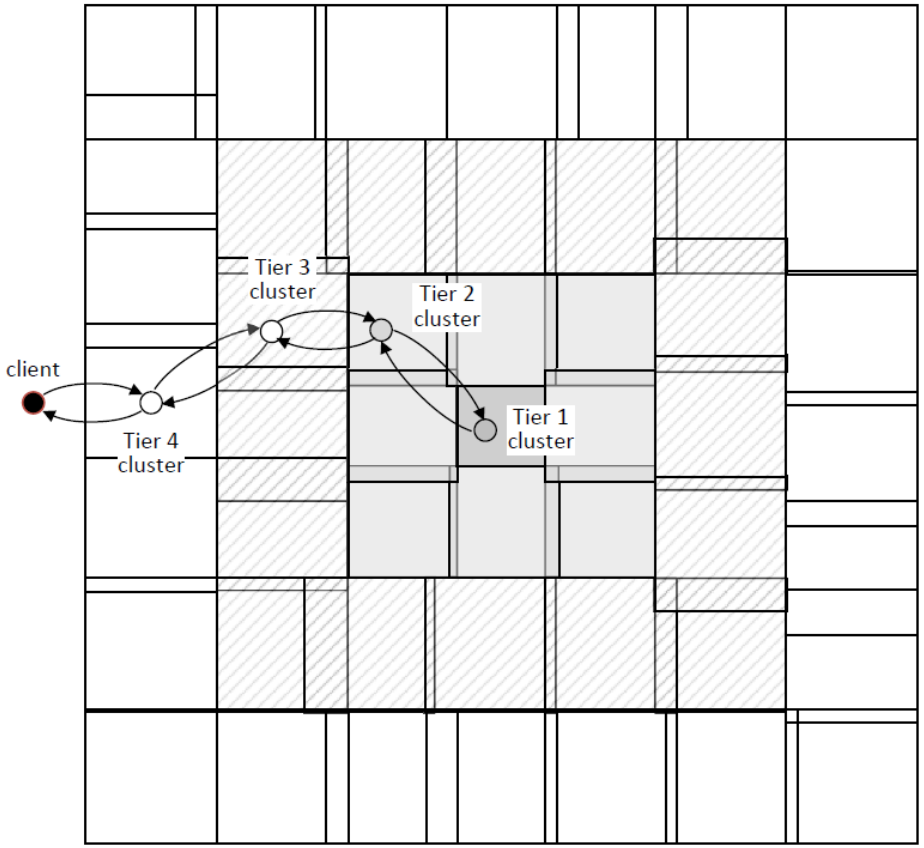
- The IoT system is comprised of IoT domains controlled by proxies
- Proxies collect data from IoT sensors, validate it, possibly apply ML and send to the ordering servers for ordering and inclusion in the replicated ledger
- Both proxy-orderer and inter-orderer connections are implemented using secure connections.
- Orderers form a fully connected overlay network.
- Data is stored until the orderer packages it into a block and submits for approval to other orderers through a PBFT consensus round
- Accepted blocks are inserted into the replicated ledger.
- PBFT is executed over overlay network.
  - A total of n_{ord} =3f+1 orderers allows consensus to be achieved with up to f faulty or misbehaving (Byzantine) nodes
  - Alternatively for a total of n_{ord} orderers majority is $\lceil (2n_{ord} -1)/3 \rceil + 1$

PBFT in a nutshell

- In the pre-prepare phase, the current leader (primary node) announces the next record that the system should agree upon
- On receiving this pre-prepare, every node validates the correctness of the proposal and multicasts a prepare message to the group
- The nodes wait until they collect a quorum of 2f +1 prepare messages and publish this observation with a commit message
- Receipt of 2f +1 commit messages means the proposal is accepted
- *Problem*: several one way delays are needed to complete PBFT voting



Multiple tiers

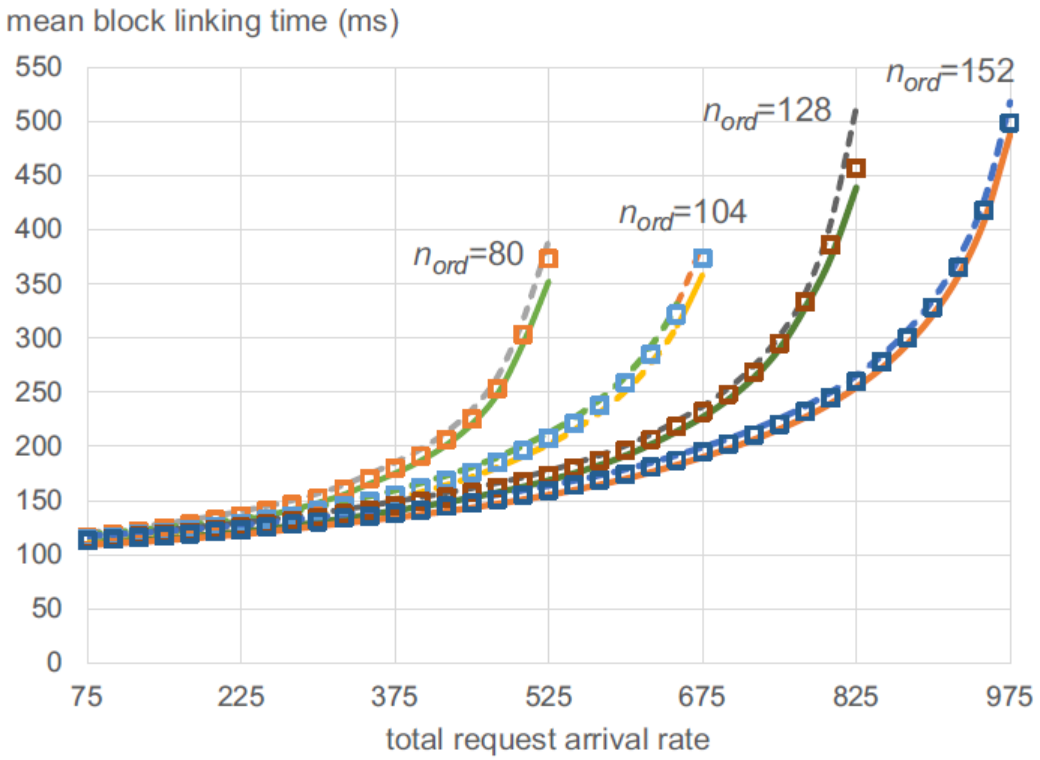


- Single cluster system can not have good performance in large areas with large number of orderers.
- Sometimes territorial organization of service providers demands tiered architecture.
- Clusters may partially overlap since overlay networks do not interfere with each other.
- Sharding architecture can reuse this concept.

Consensus with Multiple tiers

- Each orderer within cluster is connected with one or more clusters in higher tier.
- There are no communications horizontally between clusters in the same tier.
- Request has to be approved with voting majority in each cluster on the path to the top cluster.
- If only ordering of blocks is needed then final order is decided at the top tier committee.

Optimized block linking time under variable total block arrival rate



- Request arrival rate is variable, number of orderers is a parameter, 10ms mean OWD.
- Results for two, three and four tier architectures are shown with solid line, boxes and dashed line respectively.
  - Distance between the curves under moderate offered load is 4-5ms.

Conclusion and directions for future work

- We have described a PBFT ordering service with multiple entry points suitable for blockchain applications in IoT environments that cover large geographical areas
- Multiple entry points bring the problem of contention which can be resolved using a CSMA/CA-inspired resolution scheme
- 2-,3- and 4-tier architectures have similar performance for offered load per cluster lower than 0.8.
- Total number of orderers and system coverage are most important factors of performance.