

Integrated IoT-Fog-Cloud Systems

Subjects: **Computer Science, Artificial Intelligence**

Contributor: Jameela Al-Jaroodi

Integrated IoT-fog-cloud system (iIFC) offers the opportunity to create suitable platforms to develop and operate important smart city applications. These applications can utilize services provided by IoT devices, fog nodes, and cloud services.

smart cities

smart city applications

Internet of Things

fog computing

cloud computing

optimization

networking

security

1. Introduction

Progress in communication technologies, IoT, cloud and fog computing, and software technologies strongly contributed to the evolving concept of smart cities. Many unique opportunities can be offered by these technologies to create a wide array of applications that optimize smart cities' services. IoT enables the interaction of different physical objects, including sensors and actuators, and integrates them with a smart city network. Cloud computing offers powerful, scalable, and on-demand computation, data storage, and innovative software services to enable different smart city applications. Fog computing provides real-time services, better location and context awareness processing, mobility supports, and advanced communication services, such as streaming support for smart city applications ^[1]. The combination of these technologies allows applications to utilize the best features of each one to optimize operations and enhance the quality of service. As the cloud provides virtually limitless storage and processing capabilities, fog nodes supplement those with locality and mobility tuned functions with accurate location awareness, real-time responses, caching services, and localized critical processing.

When the IoT, fog computing, and cloud computing are integrated into one system, this integrated system can offer an advanced and comprehensive platform for developing and operating many smart city applications ^{[2][3][4]}. In such a platform, all entities of smart cities like cars, individuals, buildings, roads, hospitals, and power and water plants are equipped with IoT components, which are connected together and linked with fog computing and cloud computing platforms executing smart programs to improve the capabilities and features of smart cities.

2. iIFCs Utilization Issues

Although iIFCs can provide several benefits for smart city applications, their effective utilization involves several issues such as heterogeneity, integration, reliability, effectiveness, etc., which need to be addressed.

One of the major advantages of fog nodes is their closeness to the IoT components and ability to provide real-time support. Using fog nodes can provide localized system resources to support the real-time capabilities of IoT-based applications, in a smart city [1]. Fog nodes in a smart city can be close to different IoT devices where data is generated and where most actions are needed. However, fog nodes may have limited resources and with increasing load, they will not be able to provide the necessary real-time support, thus lowering the value of using them. One approach to address this is to enable active collaboration among multiple fog nodes. Collaboration between fog nodes in close proximity to each other will enable better utilization of their resources and an enhanced ability to achieve real-time performance. With fog node collaboration, better load balancing across these nodes is achieved and situations where some nodes become overloaded while there are other nodes with low loads can be eliminated [5][6]. However, adding features to enable fog nodes collaboration is also a complex task, as several issues must be addressed. These include allowing fog nodes to monitor each other's loads, creating efficient replication and migration techniques, managing collective resources, and optimizing services and resources allocation to achieve load balancing which will satisfy the real-time requirements.

Furthermore, it is important to utilize different advanced approaches, such as software-defined-networking [7] and mobility-aware application scheduling [8], to effectively support mobility. These issues will further complicate the necessary integration requirements and increase the overall complexity of developing these features for smart city applications with mobile components.

Developing smart city applications that run on iIFCs and ensuring that they meet several functional and nonfunctional requirements is challenging. There are some development environments designed to enable the development and operations of smart city applications on iIFCs. Examples are SmartCityWare [2] and FogFlow [3]. These development environments provide a base upon which to build services for such applications. However, these environments are limited and do not provide comprehensive solutions for developing, operating, and testing smart city applications and services. There is still a need for more innovative development environments that can deal with both functional and non-functional requirements for iIFCs for smart city applications.

3. Optimization with iIFCs

Many real-world problems in smart cities are considered optimization problems. Examples include finding the optimal traffic-light configuration or creating optimal process schedules to improve some predefined performance metrics. Therefore, the use of optimization methods can provide advantageous features to applications for smart cities. Optimization methods can benefit significantly from the availability of iIFCs, which enables the collection and analysis of relevant data. The final outcome of these optimization methods can be decision support or even fully automated actions. As thousands (possibly millions) of IoT devices collect smart city data, fog nodes can become the organization and filtering point before shipping data to the cloud. Integrated fog nodes can also do some of the local processing and monitoring using the collected data from the IoT devices directly connected to them. They can in real-time make the necessary decisions and take the appropriate actions. All data eventually will reach the cloud for more in-depth analysis and optimizations on a larger scale. The challenge for the use of such optimization methods is defining performance measures and objectives using formal specification, analysis, and measurement

models. Typically, smart city problems are multidimensional, yielding multi-objective optimization problems. This implies that different performance measures target different areas, and all of them need to be quantified and weighted with respect to their importance. With the support of iIFCs that can accurately provide the necessary data and measurements, the use of optimization methods will result in more efficient operations and better utilization of smart city resources.

Different intelligent algorithms and ML techniques can be used to support optimizations for iIFCs based on the smart city applications. Alli and Alam [9] developed a ML-based approach to secure computation offloading schemes in iIFCs for smart city applications. This approach supports employing the fog nodes for processing large amounts of data generated by many IoT devices, accelerates processing of large tasks, and conserves energy in the IoT devices. The authors used the Neuro-Fuzzy model and Particle Swarm Optimization (PSO) via the smart gateway to select an optimal fog node for computation offloading. If the fog node does not have enough resources for the workload, it sends it to the cloud.

Efficient optimization is also needed for effective service placement in the fog environment to support smart city applications that run on iIFCs. To achieve good support for these applications, it is important to manage and optimize service placement methods [10]. Canali and Lancellotti [11] proposed a heuristic solution based on genetic algorithms for placing smart city services on fog nodes. Both the communication delay and the processing time on the fog nodes are considered in this optimization. Choi and Ahn [12] proposed another service placement approach based on the logical fog network and the pre-obtained service demand as defined by previous usage. This approach considers the nature of such large-scale smart city applications and the need for a scalable service placement approach. The authors found that service placement based on pre-obtained demand can improve the utilization of the resources available to fog nodes; however, a service migration capability is needed for this approach. Zhu et al. [13] proposed an optimization technique for service allocations on vehicular fog computing. In this model, a trade-off is preserved between the service delay and quality level.

Data collection is at the core of the emergence of smart services, since efforts to optimize services, learn patterns, and predict trends that are mainly based on how systems behave and behaved historically. The ease of collection and availability of data is the key to enabling and enhancing most of these smart city applications. Optimization techniques, in particular, require extensive knowledge of past processes and their performance. In addition, they require detailed information on the resources, capabilities, availability, and overall conditions of the running systems. The data must also be accessible to various processes at different levels, whether in IoT locations, on the edge, or in the cloud. The iIFCs provide a suitable architecture to support data collection and analysis at all levels. The inherent characteristic of IoT devices is that they collect data. Fog nodes can, to some extent, store, and process that data, while the cloud offers flexible and large-scale resources for storage and processing, making it possible to use high-performance methods for optimization. Collectively, and with the correct allocation of necessary tasks, the iIFC is an ideal infrastructure for the optimization processes of smart city applications.

4. Smart City Networking Architectures and Communication Requirements

In the following discussion, we focus on the link between the IoT smart city devices and fog platforms. Applications with short-range communication, such as smart buildings and smart water networks, can utilize protocols from the personal area network (PAN) classes, such as IEEE 802.15.4 (Zigbee) and 802.15.1 (Bluetooth) [14][15]. These protocols are usually distinguished by lower bandwidth, low energy consumption, and short-range communication. Applications and services demanding longer ranges, like intelligent transportation and manufacturing and control, can use protocols in the local area network (LAN) class, such as IEEE 802.11 [16]. Applications involving wide-range communication, such as UAVs and smart grids, can use wide area network (WAN) protocols, such as IEEE 802.16 [17], cellular, and satellite communication. All of these protocols can support asynchronous and synchronous data connections. Asynchronous data connections can be used for smart city applications with best-effort traffic, which can tolerate delay. These applications include structure monitoring and data-gathering services. Such services can be a part of smart city applications such as smart structural health management, smart water networks, smart sewage and waste management, smart environmental monitoring, and smart energy. Synchronous data connections can be used for smart city applications with data traffic requiring more stringent QoS requirements, such as bandwidth and delay [18] such as real-time and multimedia communications. Services that generate such traffic can be a part of smart city applications, such as smart transportation, smart public safety and security, and smart healthcare and wellness. These applications may be using cameras to monitor and transfer captured videos of different situations to determine the appropriate actions, which must be done in real-time. Additionally, these networking protocols provide security and reliability functions. Though, majority of these security functions need extra processing and can create additional delay and consume a significant energy amount. Consequently, these considerations should be considered before enabling these features.

Additionally, some smart city applications require very low end-to-end communication delays. Examples of such smart city applications are smart transportation and smart grids. Delays are not tolerated, since the control information being communicated needs to reach its destination within a few micro- or milli-seconds to support reliable control systems and quick response times so as to avoid the possibility of physical damages or collisions. In contrast, other applications have a higher tolerance for communication delay, such as UAV applications that collect sensor data for future analysis.

Energy efficiency is also a critical necessity for smart city applications, though various applications that have in-house power sources, like smart grid systems, can work with high energy consumption protocols [18]. Other smart city applications have very limited energy sources and require protocols that consume less power. Some examples include UAV-based smart city applications and applications for remote pipeline monitoring [19]. It is also imperative to consider that some applications such UAV-based smart city applications and remote pipeline monitoring use wireless communication [20]. Others, such as smart grids and smart buildings, use both wired and wireless communication networks. In addition, some applications need mobility support while others do not. These are important requirements that need to be considered in connecting with other systems such as fog and cloud computing.

References

1. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing; ACM: New York, NY, USA, 2012; pp. 13–16.
2. Mohamed, N.; Al-Jaroodi, J.; Jawhar, I.; Lazarova-Molnar, S.; Mahmoud, S. SmartCityWare: A service-oriented middleware for cloud and fog enabled smart city services. *IEEE Access* 2017, 5, 17576–17588.
3. Cheng, B.; Solmaz, G.; Cirillo, F.; Kovacs, E.; Terasawa, K.; Kitazawa, A. Fogflow: Easy programming of iot services over cloud and edges for smart cities. *IEEE Internet Things J.* 2017, 5, 696–707.
4. Dutta, J.; Roy, S. IoT-fog-cloud based architecture for smart city: Prototype of a smart building. In Proceedings of the 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence, Noida, India, 12–13 January 2017; IEEE: New York, NY, USA, 2017; pp. 237–242.
5. Qayyum, T.; Trabelsi, Z.; Malik, A.W.; Hayawi, K. Multi-level resource sharing framework using collaborative fog environment for smart cities. *IEEE Access* 2021, 9, 21859–21869.
6. Verma, M.; Bhardwaj, N.; Yadav, A.K. Real time efficient scheduling algorithm for load balancing in fog computing environment. *Int. J. Inf. Technol. Comput. Sci.* 2016, 8, 1–10.
7. Bi, Y.; Han, G.; Lin, C.; Deng, Q.; Guo, L.; Li, F. Mobility support for fog computing: An SDN approach. *IEEE Commun. Mag.* 2018, 56, 53–59.
8. Bittencourt, L.F.; Diaz-Montes, J.; Buyya, R.; Rana, O.F.; Parashar, M. Mobility-aware application scheduling in fog computing. *IEEE Cloud Comput.* 2017, 4, 26–35.
9. Alli, A.A.; Alam, M.M. SecOFF-FCIoT: Machine learning based secure offloading in Fog-Cloud of things for smart city applications. *Internet Things* 2019, 7, 100070.
10. Manogaran, G.; Rawal, B.S. An Efficient Resource Allocation Scheme with Optimal Node Placement in IoT-Fog-Cloud Architecture. *IEEE Sens. J.* 2021, 21, 25106–25113.
11. Canali, C.; Lancellotti, R. A Fog Computing Service Placement for Smart Cities based on Genetic Algorithms. In Proceedings of the CLOSER 2019, Heraklion, Greece, 2–4 May 2019; pp. 81–89.
12. Choi, J.; Ahn, S. Scalable service placement in the fog computing environment for the IoT-based smart city. *J. Inf. Process. Syst.* 2019, 15, 440–448.
13. Zhu, C.; Tao, J.; Pastor, G.; Xiao, Y.; Ji, Y.; Zhou, Q.; Li, Y.; Ylä-Jääski, A. Folo: Latency and quality optimized task allocation in vehicular fog computing. *IEEE Internet Things J.* 2018, 6, 4150–4161.

14. Olenewa, J.L. Guide to Wireless Communications; Cengage Learning: Boston, MA, USA, 2014.
15. Stallings, W. Wireless Communications and Networks. Prentice Hall; Pearson Education, Inc.: Upper Saddle River, NJ, USA, 2005.
16. IEEE 802.11. Available online: https://en.wikipedia.org/wiki/IEEE_802.11 (accessed on 14 July 2021).
17. IEEE 802.16. Available online: https://en.wikipedia.org/wiki/IEEE_802.16 (accessed on 14 July 2021).
18. Jawhar, I.; Mohamed, N.; Al-Jaroodi, J. Networking and Communication for Smart City Systems. In Proceedings of the IEEE International Conference on Smart City Innovations (IEEE SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1145–1151.
19. Jawhar, I.; Mohamed, N.; Al-Jaroodi, J.; Zhang, S. A framework for using unmanned aerial vehicles for data collection in linear wireless sensor networks. *J. Intell. Robot. Syst.* 2014, 74, 437.
20. Jawhar, I.; Mohamed, N.; Agrawal, D.P. Linear wireless sensor networks: Classification and applications. *J. Netw. Comput. Appl. JNCA* 2011, 34, 1671–1682.

Retrieved from <https://encyclopedia.pub/entry/history/show/39982>