ISSN 2348 - 9928
Doi:10.46532/ijaict-2020022 Published on 05 (07) 2020

A Survey on Service Level Components in Big-Cloud-IoT Systems with Hybrid Meta-heuristic Techniques

[1]Xueqiang Yin and [2]Athreya Tao Chen
[1,2] School of Computer Science, Northwestern University, China.
[1]xueqiangyin@yahoo.com

ArticleInfo
International Journal of Advanced Information and Communication Technology
(https://www.ijaict.com/journals/ijaict/ijaict_home.html)
https://doi.org/10.46532/ijaict-2020022
Received 10 March 2020; Revised form 15 April 2020; Accepted 30 May 2020; Available online 05 July 2020.

Abstract - Big data is one such technology. When we receive huge volume of data, there will be high demand in processing the huge data. It can also be said challenging task in big data processing. The increases in IoT devices in the network system collect more data to be processed in centralized devices called cloud storage. Every big data is processed and stored in the cloud. To overcome the performance and latency issues in large data computation, big cloud processing system uses edge computing in it. One of the key components of IoT is edge computing. We combine big data with cloud and edge computing in this paper as hybrid edge computing system. In the edge computing system, huge number of IoT devices computes services in its nearby network edge. Data sharing and transmission between the various service components may affect performance of the system. The main aim of this research article is to reduce the delay in data transfer between the components. This optimization goal is achieved by new Hybrid Meta-heuristic optimization (HMeO) algorithm. New HMeO algorithm designed for IoT devices to deploy the service components. MHO model is design to optimize the process by selecting the edge computing with minimum latency. Our proposed HMeO algorithm is compared with existing genetic algorithm and ant colony algorithm. The result shows HMeO algorithm provides more performance and efficient in in-depth data analysing and locating the component in big data-based cloud environment.

Keyword - Bigdata; Cloud services; Data-intensive computing; Edge computing; Hybrid meta-heuristic optimization.

1. Introduction

At present mobile devices such as smart phones, laptops, mini tablets, smart data storage devices become significant thing in every human hand. Development of these devices makes tremendous development in data over the network and service computing. These devices are sensitively live with us daily life. Growing information in industries and academic needs more research on efficient data computing. The big data processing becomes complex in handling large data sets. These are considered in research for computing improved solutions. One such solution is cloud-based architecture explained in paper [1] for storing the huge data and this is accepted as next generation data processing solution widely.

Data intensive computing is process of parallel computing for huge volume of data in terabytes or petabyte in the big data. This data-intensive computing with the cloud technology bring the challenging growth in the service computing technologies [2]. The service deployment over the cloud-based environment is much focused by service providers rather than edge computing [3]. However, the data finally used to store at edge of network were users become closer to service deployment, which seems difficult. In our research article data intensive computing become key role for IoT –edge computing for big data processing in the cloud network. Traditional service computing shows transmission time is very smaller than execution time between the IoT service components. Due to this problem, previous research fully focused on improving execution time, bandwidth etc in data intensive computing platform. They do not concentrate much on service deployment. Nowadays, growth in big data is due to growth of information in world and social media etc. big data need a greater number of service components in its process. Due to individuality in the service component, its deployment takes much time.

In paper [4], intelligent solutions for mobile edge computing in industries digital platform is discussed. The main requirement of industry is optimization, intelligent applications, privacy, and security in computing. The numerous edge tiers are added between the big data and cloud system. This helps to enable the service from IT and cloud technology at the edge computing via mobile edge (ME) devices. The speedy development in downloading different files, data, services, and application in the cloud network is done by boosting ME environment with ultra-high bandwidth and low latency in the network. By this process users can get highly qualified services. The mobile edge computing executes the services in the cloud servers which are nearer edge server for the current services by caching cloud services in edge servers. This process brings the users nearer to cloud. Edge computing improved its nature by working as mobile cloud computing. It can also reduce the time by transmitting services between edge devices and mobile devices instead of working through internet. This can reduce the response time of the services invocations in the network.

Internet of things (IoT) defined as object connected via internet such as web services devices, scanner, web cam, smart devices, smart city buildings with sensors etc can able to transmit and receive numerous data at a time. The edge computing plays main role in developing IoT applications. our proposed hybrid big – cloud, IoT is a key device by low cost processing the information. In edge computing [5] numerous data processing and interactions takes place between digital devices of the network. It's necessary for providing intelligence to the edge computing technology for performing better communication between the human and its environment. Through this operational efficiency, less time transmission, better reliability, and resource consumption is achieved. This proves that technology development needs more IoT processing which requires big-cloud –IoT hybrid architecture.

Advancement in the technology leads this research field remarkably interesting in IoT development. Data 's are directly processed in the cloud storage with a help of IoT devices [6]. This process increases the integrity of the data which leads efficient data mining in the IoT. Recently many research on AWS based cloud computing for IoT services draws interesting attention. Some advantages of edge computing as follows:

- Distributed computing: it provides low latency in real time data processing. at present intelligent processing in edge computing provides short time data analysis for better performance.
- Efficiency: cloud servers are nearer to the user so data analysis at edge gets faster execution, it results in high efficiency.
- Traffic relieve edge nodes process the data nearby; it reduces traffic in the data transmission from edge to cloud server.
- Security: whenever there is attack in the network, normal cloud network affects lot. But in our architecture edge component is near to the user and reduces the risk of attack. Also edge computing handles data processing locally unlike cloud it does not affect the whole network.

Traditional service deployment has some limitation in edge computing are, balancing the data load, capabilities of computing power, efficiency of IO devices. Some key factor which affects service deployment has overcome by our proposed system. we assure following limitation in our paper to solve the deployment problem [7].

Huge data transfer: In the edge-cloud computing is based on data intensive process. It transfers large volume of data at a time in various data centres. To handle this situation most favorable solution must be designed using service deployment strategy so that bandwidth supports large data transmission.

Dependencies: There are two types of service dependencies, 1. Data dependency 2. Logical dependency.

Data dependency is based on input and output processing. Logical is based on service execution. There is lot of logical computations like 'OR', 'NOT', 'AND', is processed in the service execution. Dependencies in the service components are deployed highly in data transfer. Challenges to service provider: In the cloud computing multiple services computes an application with available service provider [8].

These services are fixed in the network based on providers need. Not everyone can access the service. Some services placed in edge system so that cloud computing plays major challenge in service deployment. Every data centre must have enough storage capacity. For executing the service with input in the data centre requires the storage to process and store the input and its result. In this case data must not exceed data centre storage capacity.

In our proposed work, above challenges are targeted and combine IoT in data –intensive parallel processing for better optimization solutions. Also new Hybrid Meta-heuristic optimization (HMeO) algorithm makes service deployments optimal. By combining the big data with edge computing and cloud computing efficiency of the proposed system is improved [9].

The contributions are:

- Service deployment architecture is designed such that huge data are processed parallel in hybrid big-cloud-edge computing architecture. This architecture is explained with various key components with service deployment execution.
- Proposed HMeO algorithm goal is to reduce the data transfer between the devices in big cloud edge computing networks. The parameters are executed for better results. The existing genetic algorithm, ant colony optimization algorithm is compared with HMeO in some experiments for better results. The results of proposed system proved its efficiency. We also consider some limiting factors which can be solved in future work.

The paper is organized with architecture overview in section 2. In section 3 optimization process is discussed. The algorithm is discussed in section. Conclusion of the present work is discussed in section 5.

2. Literature Survey

In cloud computing platform deployment of service is difficult task. There are more studies about deployment issues. The many public cloud providers combined to meet present cloud market for the huge users. Clouds computing in big data are divided in terms of different schemes, interfaces, and machine processing techniques. The multiple clouds across the network create the virtual infrastructure to the users for service deployment. At present cloud computing networks lags in advanced services like

Corresponding author at: Xueqiang Yin, Northwestern University, China.

ISSN 2348 - 9928
Doi:10.46532/ijaict-2020022 Published on 05 (07) 2020

automated process, intelligent computing, etc. it is somewhat difficult for selecting the optimal storage in the cloud computing for service deployment and service distribution. Cloud has more service components among its networks. In [10], proposed a new architecture for the cloud computing based on service scheduling in the multiple cloud virtual environment.

The [11], clouds latency is discussed. When the bandwidth of the data centre is limited, it results in high latency during the data transmission. During the less data movement between various data centres in network makes correlation in them. Their proposed work is based on clustering of data in place of storage. Nowadays mobile devices are very portable and data intensive computing. Map reduce in [12] is based on multiple edge computing in cloud environment. Big data oriented IoT is executed with Local Weighted Linear Regression (LWLR) methodology. Further IoT DeM implementation in the Hadoop 2 with less than 10% error.

Existing research works by overcoming dynamic principle in deployment of services at infrastructure. In [13], introduced game theory model in its dynamic framework. Resources are allocated dynamically by considering multiple service providers in the network. The major challenge in the cloud computing is providing service deployment based on diversity in user. The [14] classification method is proposed in selecting service deployment based on the user preferences and their optimization goal in cloud. Cost and performance-based optimization used to solve the cloud trafficking. Various components in the network focused optimally in scheduling the services. The service deployment has been peak due to development edge computing in research field to reduce the delay, storage process. The [15] introduced the algorithm for optimizing the interactions between edge components in big data management and cloud. By combining the edge computing with cloud computing, became new technology for networks and communication. In this paper literature survey explains very innovative techniques and efficient IoT algorithm are designed. They are mostly used in IoV's application for scheme deployment in the multimedia optimization in the cloud computing. In this paper they introduced unified model for QoS and QaMeC problem solved. Further NSGA-II algorithm is applied to search for the optimal deployment problems. QaMeC method proved for various applications to meet QoS requirements.

3. Problem formulation and Design

Here we design a big-cloud –IoT hybrid service deployment architecture. Then the service deployment problem for parallel processing of huge data is addressed in big-cloud –IoT hybrid system. Important definitions are presented for main concepts in this experiment and problem is implemented finally in Fig 1.

3.1 Service Deployment Of Components In Data Intensive Computing

The big-cloud –IoT hybrid service deployment architecture is designed in figure 1.The architecture is composed of four layers. They are 1. Cloud layer 2. Internet layer 3. Edge services 4. User IoT layer.

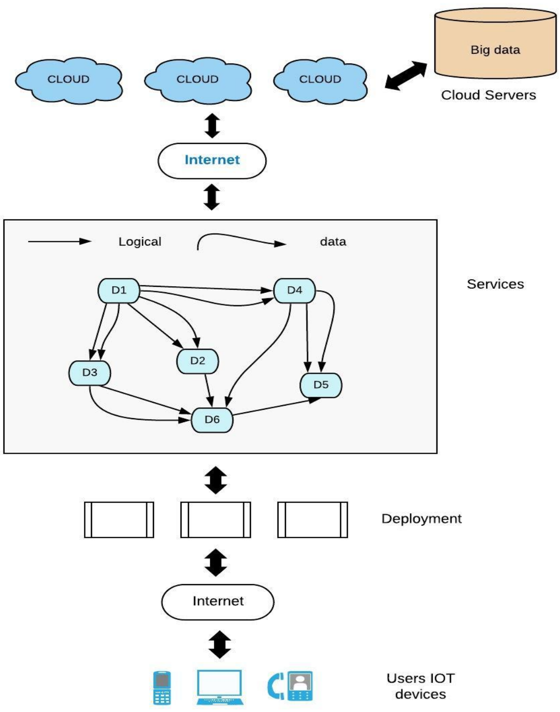

Fig 1. Big-Cloud –IoT hybrid service deployment architecture.

Cloud layer: It is the first layer were more cloud servers combined to provide big data processing services. Numerous cloud service providers are available in this layer to manage the incoming big data. This cloud server can produce vast storage capability and speedy computations in its server. Location of this server is very far from the location of the user.

Internet layer: This is network computation layer. It connects various networks in big data and cloud computing through the communication protocols. It helps IoT devices to transfer the files communicate, and store in the network storage. Servers in the various places communicate via this internet protocols. In our architecture, internet connects user IoT devices to edge devices, secondly edge devices to cloud servers in the big data environment.

Edge computing layer: The main components of the service deployment are edge servers in the network. This layer is remarkably close to the users IoT devices. So it provides high quality in services also low latency in data transfer. Its computation capability is limited when comparing with cloud computing servers. Data intensive computing have complex architecture which cannot be computed by single edge server. So, in our architecture we have computed six services from DS1 to DS6 with data centers. Data transmission will be huge between the service components in the edge network. Also, different data is computed in between the devices. In the service dependency diagram, it clearly shows how DS1 has to process two data to be depended on DS2 and DS3. Edge service provider make

Corresponding author at: Xueqiang Yin, Northwestern University, China.

decision about which server will process the data. Here cost of service and execution, processing logic is considered.

User Layer: User layer has many smart devices in it. Users uses many smart devices in them day to day life such as phones, PC, laptops, tablets, IoT devices etc. They were connected to edge servers with few local networks. Some limitations like computing, storage which cannot process locally. In our Big-Cloud –IoT hybrid system user can freely choose were to process the application. They can select local edge server or remote cloud server for processing the data. Big- Cloud –IoT hybrid system, the edge devices reduce the traffic in the network. The security level of the system is improved more efficiently. Service computing became research highlight in our paper. So few components for the service deployment in the cloud system for considering the servers. Edge services can be affected by complex IoT services dependencies between its components. This is one of challenge in research field.

3.2 Big - Cloud –IoT hybrid system

In Big - Cloud –IoT hybrid system, data intensive computing is very essential in IoT devices. Some Key concepts are required for service deployments. This definition are referred from various sources of references with some inclusions.

Problem definition 1: *Data-Intensive computation: Data services are represented in directed graph as DS(),* where, DS is services in from various components. From above figure

- DS is collection of six services as DS1 to DS6.
- Ld = {depl ij , | dsi, dsj ϵ dsiu} is a logical dependency between the services in the network system. This states that different multi logic executes between the various services in the network. The paths of the execution of each service are affected. Execution path of the service components generates different time in data transmission.
- Dd = {depd ij , | dsi, dsj ϵ dsiu} defines the data dependencies between the services. There is more number of data transmissions between the service components. This represents the dependencies between the data.

Input data required are stored in the data center for performing the services in the network. In Fig 1, curve connecter represents the data dependencies between the service components. Also, some data's are generated by the service components during the service execution in the network.

Problem definition 2*: Logical data Dependency:* it represents how the data executes between the service components. Logical dependencies are formulated as follows:

- They are executed one after another as sequence like DS3 executes after DS2 in order.
- Split AND: One process divided into multiple processes in parallel and node performs computation simultaneously. Example: d5 and d7 computes simultaneously in Fig 2.
- Join AND: Different process converged computes in parallel manner.
- Split XOR: One process divided into multiple processes in parallel and node performs computation only one at a time. Logic is similar, only differences is processes chooses ranges from 0 to number of total processes.
- Join XOR: optionally two or more branches synchronized to perform the process.
- Loop: services are executed multiple number of times. In our scenario fixed number computations are performed.

In the Fig 2 shows the example of IoT number of computations. Logical data dependencies between the various service components are represented as follows:

- Arrow mark represents the logical relationship and its computation.
- Dashed line represents the data dependency relationship.

Data d5 depends on ds6 and ds7. For processing ds5, the output ds4 and ds7 is necessary.

Problem definition 3: *Data Dependency Process*:
depdij = < dsi, dsj, dataij > dependency between the two service components defined such that input of dsj depends on output dsi . the transfer of data defined as dataij. One of the basic features of data intensive computing is data dependency.

Problem definition 4: *Data storage center*: it is used to deploy the various service components
which can be represented as follows:

$$DSC= \{dscij \mid i = 1, 2\ldots\ldots\} \quad (1)$$

Each data center is represented as dsci = <sti>, sti , is represented as capacity of data storage of dsci. Data bandwidth (BW) between the various centers are represented in the matrix i j in the network.

Problem definition 5: Network Bandwidth: In the hybrid big-cloud-IoT system, bandwidth between the network in edge computing servers and cloud servers where represented in arrays called Network Bandwidth (NB). The NB elements communicated between the two data centres is represented as NBi. Different data centres in the cloud network works in different bandwidth. Some have data transfer rate amazingly fast and some have very moderate. So, timing for transferring data between the servers is unpredictable.

Problem definition 6: service components in the cloud: service components are fixed in the cloud environment for our experimental purpose. For complete execution of the

ISSN 2348 - 9928
Doi:10.46532/ijaict-2020022 Published on 05 (07) 2020

problem, components provided by cloud server and some free deployable service components in the edge servers are used. service components which are fixed in the cloud are not used. Only the process we need to do is service components which are freely deployed in edge centres are managed. Anyway in cloud storage capacity is always unlimited.The user cannot add any extra memory to cloud storage except the service provider. In the cloud computing security is authorized only to service provider Shown in Fig2.

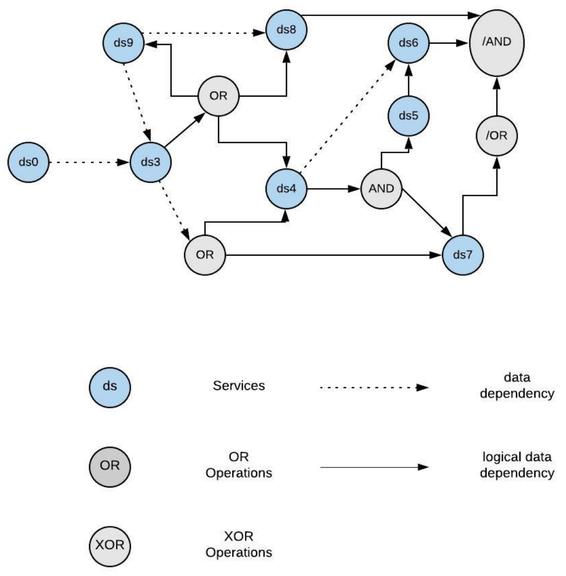

Fig 2: Service dependency model

Problem definition 7: relations of edge and cloud: Here we describe how the interactions between the edge components and cloud service components works. Several cloud computing servers communicate with the edge computing components in transferring the data. The main problem in previous edge-cloud hybrid server is communication bandwidth in the network between the cloud computing servers and edge computing servers increases the complexity of the system [15]. To overcome this in our architecture we have only one cloud servers and few edge devices.

Problem definition 8: service deployment: consider the given set data intensive services DIS and set of data storage server set as DST for the deployment of service process. Service Deployment process is represented as:

$$dys = U\ I = \{ 1,2,\ldots, |\ dis.S|\}\ \{sei\ |\ dysj\ |\ dysj\ \}. \qquad (2)$$

Apart from fixed service devices in the cloud storage, others are in DIS set. To store the data for deployment, DST must be identified. In the same data enter it may have different components of the deployment. The main criteria followed are input cannot exceed the storage canter of the data. Spare data canters are allowed in exceptional case. In the deployment method we can believe that some vacant data centers are always available which is having with lot of storage space and highly scalable.

Problem definition 9. Latency in transfer: it is defined as, the total time taken by service computing from accepting the service to processing and returning the result.

$$Ldata = Lexe + Lother \qquad (3)$$

Ldata is the time taken for transmit the data between the service components. Services based on data dependency which may take place in the same server in the cloud or data center is ignored. Lexe is refer for time taken for execution of the service. Lother is time taken by others request time, response time , transfer time , storage identification time, network connection time. In our experimental scenario, transmission of data between the different components of the service is large . so Lexe, Lother are not considered.

$$latency \approx Ldata \qquad (4)$$

Problem definition 10: Few DIS, data intensive service computing and data storage center DST were fixed in our cloud computing platform and the remaining components are assigned to edge data storage based on the strategy of deployment. In respective datacentre the input needed for the service component is stored and deployed before the service execution in the system. In our research goal, cost efficient transmission and deployment strategy between the services is main process. This can be considered as a goal optimization process and in our optimization strategy, deployment of service with low latency will be main aim.

There exists a more execution path based on our complex structure of logical dependencies between the various service components. Among these all path, we must choose data transmission with less latency. In between the service components large data transmission takes place in the single data server in the network. Though theoretically service components are closely deployed, there is storage limitation for all data centers. If the storage level is high than the data server, then it is considered as a most scalable.

4. HMeO algorithm:

In this chapter the hybrid meta-heuristic algorithm is designed for the perfect optimization solution. Many algorithms in swarm intelligence are analysed. The famous ant colony algorithm is combined with annealing simulation algorithm in our proposed work. HMeO algorithm is combination of the above two algorithms for best optimization solution in the big data based cloud processing.

4.1 Working of ACO and SA algorithm:

The ant colony optimization (ACO) algorithm works based on the ants behavior during the search of food. This ant behavior creates the path for their colony for food source. How the numerous ant follows same path? This is where the ant colony innovatively performs. The ant,

ISSN 2348 - 9928
Doi:10.46532/ijaict-2020022 Published on 05 (07) 2020

which going to seek food and found the food source releases the pheromones on the way. These pheromones are sensed by other ants and path is marked with help of pheromones. Ants are finally following the shortest path for reaching the food source. This optimization algorithm works better but only drawback is sometimes it consumes more time.

In other hand hill climbing or simulated annealing algorithm is selected for better deployment of services. This annealing algorithm works based on metallurgy annealing. It finds the best universal optima from large number of optimum (best results). Time problem in ACO algorithm is mainly due to considering the solution of each iteration. Finally, ant with good solution is accepted with shortest pheromones level. Larger values are eliminated. Efficiency is high when the best result is easily trapped.

HMeO algorithm combines the advantage of both ACO algorithm and simulated annealing (SA) algorithm for the better results. Once the solution obtained is best, then it is accepted as best solution. If it not best, algorithm takes the decision based on the probability. By this process efficiency of the algorithm is improved. The local best solutions are avoided and globally best solutions are chosen for better service deployment in the network.

- Start the algorithm and path of the ant is monitored.
- Best among them are identified in each iteration and updated
- Matrix is initialized with pheromones.
- Temperature value is initialized as maximum T-MAX and minimum T-MIN.
- The temperature is set as T = T-MAX. iterations is started.
- In matrix, pheromones generates ants path from its home.
- Algorithms says select the best path among all generated paths.
- Matrix is updated with new iterations solutions.
- In annealing metallurgy process, each time lower the temperature and stops when
- achieved minimum temperature. Else goto second step.

HMeO algorithm applied in the deployment of the more services in network. Finding the best path from ant home to food location applies to deployment of services . here the latency is corresponds to length of the path for all services. If latency is high the pheromones released is
very less.

4.1 Initialization of services

HMeO is based on ACO approach where pheromones act as a core algorithm. service path is initialized with matrix S, as pheromones are represented. Matrix contains x rows and y columns. ‘x’ is represented as number of service components and y represents number of data servers in the network. The Sij represents pheromones concentration in the path for the service DSi and data storage server dsti. Initially matrix value is set to 0. when algorithm starts its computation, value in the matrix S is updated. Positive correlation between each element is updated. Finally, maximum value in the row is selected as optimal solution for path.

4.2 Strategy formation

Starting iteration, path is generated randomly. In the next iteration ant follows the two-step process. First probability method finds best pheromones for optimal solution. Secondly, chose the path randomly to avoid local best solution.

In our article, path is represented as deployment strategy *dps*. Total number of services denoted as *ts* for the generated iteration. In the matrix *S,* algorithm calculates number of service strategy deployed by service components *C* in the network by probability method. Number of deployment strategies num(c) for the service component

4.3 Others are deployed randomly in network

The algorithm never converges by using probability method. So it is applied with low probability strategy z. In the matrix S, the larger value computed in the row d, then it is high performance strategy in that service component d. now num(c) is largest and strategies are chosen using probability method.

5. Conclusion and future work

In this paper Hybrid big-cloud-IoT based service deployment problem is studied. The data transfer between the networks are minimized using the edge servers which is our goal of optimization. The methods show HMeO algorithm works better than upcoming algorithms and HMeO algorithm provides better optimal result in minimum time.

References

[1]. S. K. B., “A 3-tier Cloud-Cloudlet-Device Hierarchical Trust-based Service Management for Mobile Cloud IoT Systems,” Journal of Advanced Research in Dynamical and Control Systems, vol. 12, no. 1, pp. 249–253, Feb. 2020.

[2]. L. Xing, G. Levitin, and Y. Xiang, “Defending N-version Programming Service Components against Co-resident Attacks in IoT Cloud Systems,” IEEE Transactions on Services Computing, pp. 1–1, 2019.

[3]. C. Liu, C. Yang, X. Zhang, and J. Chen, “External integrity verification for outsourced big data in cloud and IoT: A big picture,” Future Generation Computer Systems, vol. 49, pp. 58–67, Aug. 2015.

[4]. D. M. Vistro, “IoT based Big Data Analytics for Cloud Storage Using Edge Computing,” Journal of Advanced Research in Dynamical and Control Systems, vol. 12, no. SP7, pp. 1594–1598, Jul. 2020.

[5]. M. Razian, M. Fathian, and R. Buyya, “ARC: Anomaly-aware Robust Cloud-integrated IoT service composition based on uncertainty in advertised quality of service values,” Journal of Systems and Software, vol. 164, p. 110557, Jun. 2020.

[6]. C. Stergiou, K. E. Psannis, B. B. Gupta, and Y. Ishibashi, “Security, privacy & efficiency of sustainable Cloud Computing for Big Data & IoT,” Sustainable Computing: Informatics and Systems, vol. 19, pp. 174–184, Sep. 2018.

[7]. Yassine, S. Singh, M. S. Hossain, and G. Muhammad, “IoT big data analytics for smart homes with fog and cloud computing,” Future Generation Computer Systems, vol. 91, pp. 563–573, Feb. 2019.

Corresponding author at: Xueqiang Yin, Northwestern University, China.

[8]. M. Wazid, A. K. Das, R. Hussain, G. Succi, and J. J. P. C. Rodrigues, “Authentication in cloud-driven IoT-based big data environment: Survey and outlook,” Journal of Systems Architecture, vol. 97, pp. 185–196, Aug. 2019.

[9]. K. Chau, “Use of Meta-Heuristic Techniques in Rainfall-Runoff Modelling,” Water, vol. 9, no. 3, p. 186, Mar. 2017.

[10]. E. K. Teh, M. A. Md Zawawi, M. F. Packeer Mohamed, and N. A. M. Isa, “Practical System-on-Chip Repeater Design With Hybrid Meta-Heuristic Techniques,” IEEE Access, vol. 6, pp. 46334–46345, 2018.

[11]. E. Azhar, Z. Iqbal, and Z. Mehmood, “Application of neural network for computing heat performance in axisymmetric viscoelastic transport: Hybrid meta heuristic techniques,” Results in Physics, vol. 8, pp. 1076–1085, Mar. 2018.

[12]. J. Jallad, S. Mekhilef, H. Mokhlis, J. Laghari, and O. Badran, “Application of Hybrid Meta-Heuristic Techniques for Optimal Load Shedding Planning and Operation in an Islanded Distribution Network Integrated with Distributed Generation,” Energies, vol. 11, no. 5, p. 1134, May 2018.

[13]. M. Jaeyalakshmi, “Task Scheduling Using Meta-Heuristic Optimization Techniques in Cloud Environment,” International Journal Of Engineering And Computer Science, Nov. 2016.

[14]. G. Dhiman and M. Garg, “MoSSE: a novel hybrid multi-objective meta-heuristic algorithm for engineering design problems,” Soft Computing, Jun. 2020.

[15].]S. Arivalagan and K. Venkatachalapathy, “Face Recognition based on a Hybrid Meta-heuristic Feature Selection Algorithm,” International Journal of Computer Applications, vol. 55, no. 17, pp. 18–22, Oct. 2012.