A Study on Latest Techniques Applied in Cloud Computing for the Efficient Data Access and Data Computation

Roslin Dayana.K
Assistant Professor
R.M.D Engineering College,
Gummidipoondi Taluk, Thiruvallur Dist.

Dr. Vigilson Prem.M
Professor
R.M.D Engineering College,
Gummidipoondi Taluk, Thiruvallur Dist.

ABSTRACT-Cloud computing is becoming a well-known catchphrase in recent times. It is an emerging and fast-growing computing technology which uses the internet and remote servers to manage data and applications. It is continuously evolving and showing consistent growth in the field of computing. It is applied in several fields along with big data analytics, internet of things and mighty things are done with these recent technologies. The crucial point of this paper is studying some of the latest techniques in cloud computing which will be useful for the efficient data access and data computation. This paper discusses about the energy efficient Ant Colony System (ACS) algorithm used for Virtual Machine Placement (VMP) in cloud, self- organizing Cloud Radio Access Network (C-RAN) that dynamically adapt to varying network capacity demands, Field Programmable Gate Arrays (FPGA) accelerator scheduling used to minimize the make-span of a given batch of FPGA requests where make-span is the time to complete all job requests in a single batch, and Intelligent Health vessel ABC-DE which is an electro cardiogram computing service where ABC-DE is the abbreviation of Artificial Intelligence, Big Data, Cloud Computing - Detecting ECG (Electro Cardio Gram). These are some of the recent algorithms and techniques used for the efficient data access and data computation.

General Terms-*Cloud Computing, manage data, manage applications, big data analytics, internet of things, data access, data computation*

Keywords-*Ant colony system algorithm, virtual machine placement, cloud radio access network, field programmable gate arrays, artificial intelligence, big data, cloud computing – detecting ECG*

1. INTRODUCTION

Cloud computing is a large-scale distributed computing paradigm, driven by an increasing demand for various levels of pay-per-use computing resources [1]. Cloud facilitates three major types of services to the customer via the Internet. Infrastructure as a service for hardware resources, such as Amazon Elastic Compute Cloud. Platform as a service for a runtime environment, such as Google App Engine. Software as a service, such as Salesforce.com [2]. These services are offered mainly through virtualization [3]. This way, the physical resources are virtualized as uniform resources and therefore are efficient for parallel and distributed computing [4]. Virtual machines (VMs) are created according to the type of operating system and the amount of required resources such as CPU, memory, and storage, specified by the customers and then run on a physical server to host application to meet requirements of customers [5]. On the other hand, virtualization allows multiple VMs to be executed on the same physical server and share hardware resources. This enables VM consolidation, which allocates the maximum number of VMs in the minimum number of physical servers. The unused servers can be switched off to cut the cost for cloud provider and customers. In this paper, we will discuss about some of the recent algorithms and techniques used for the efficient data access and data computation.

2. LITERATURE SURVEY

A selective study on latest cloud computing algorithms has been done to discover the potential benefits of each one.

2.1 An Energy Efficient Ant Colony System for Virtual Machine Placement in Cloud Computing [35]

With a rapid growth in the number and size of cloud datacenters [6], the energy consumption, as well as equipment cooling costs has risen to new highs [10]. The power consumed by an active but idle server is at the ratio between 50% and 70% of a fully utilized server [7]. Therefore, placing the VMs of a lowly utilized server onto other servers and gracefully schedule down the lowly utilized server will efficiently reduce the power consumption. The consolidation of VMs has an implication in energy efficiency. This leads to a VM placement (VMP) problem, a computational problem that seeks to obtain an optimal deployment of VMs onto physical servers [8].

In this paper [35], evolutionary computing is applied to Virtual Machine Placement (VMP) to minimize the number of active physical servers, so as to schedule underutilized servers to save energy. Inspired by the promising performance of the ant colony system (ACS) algorithm for combinatorial problems, an ACS-based approach is developed to achieve the VMP goal. Coupled with order exchange and migration (OEM) local search techniques, the resultant algorithm is termed an OEMACS. It effectively minimizes the number of active servers used for the assignment of virtual machines (VMs) from a global optimization perspective through a novel strategy for pheromone deposition which guides the artificial ants toward promising solutions that group candidate VMs together. The OEMACS is applied to a variety of VMP problems with differing VM sizes in cloud environments of homogenous and heterogeneous servers. The results show that the OEMACS generally outperforms conventional heuristic and other evolutionary-based approaches, especially on VMP with bottleneck resource characteristics, and offers significant savings of energy and more efficient use of different resources.

Related Work

Various methods have been reported in the literature for VMP according to different objectives, such as energy efficiency of the physical servers that are used to host the VMs by optimizing the assignment of VMs [7], maximization of the resource utilization ratio of the physical servers through VM consolidation [9], and load balancing on different physical servers to improve the overall system efficiency. Further, a guideline of VMP mechanisms for backup (snapshots of each VM) and working VMs to support a disaster-resilient cloud has been proposed in [10]. For the energy efficiency objective, the VMP problem is an NP-hard problem. This VMP problem was first solved as a linear programming (LP) problem. For example, stochastic integer programming was used to minimize the cost for hosting VMs in a multiple cloud provider environment. In [11], a server consolidation problem is also formulated as an LP problem, solved with heuristics for a minimized server cost. Using a VM mixed integer LP model, Lawey *et al.* proposed a framework for designing energy efficient cloud computing services over non-bypass IP/wavelength division multiplexing core networks. They adopted an approach slicing the VMs into smaller VMs and placing them in a proximity to their users so as to minimize the total power consumption.

In comparison, heuristic methods have offered higher efficiency in solving the VMP problem. In particular, evolutionary computation (EC) algorithms such as genetic algorithm (GA) have been used to improve resource utilization and reduce energy consumption. A modified GA with fuzzy multi objective evaluation was developed for the VMP in [12]. Wang *et al.* designed an improved GA to maximize resource utilization, balance multidimensional resources, and minimize communication traffic. Wilcox *et al.* modelled the VMP problem as a multi capacity bin packing problem so as to find an optimal assignment homogeneously problem to simplify the VMP with the often-heterogeneous servers in cloud datacenters. Foo *et al.* proposed to use a GA to optimize the neural network, so as to forecast and reduce energy consumption in cloud computing. As the VMP problem can be regarded as a combinatorial optimization problem (COP), many EC algorithms may be applicable. EC algorithms have been successfully applied to many COPs, such as protein structure prediction, music composition, multiple sequence alignment, distribution network restoration, constrained optimization, scheduling problems, and haystack problem, and have shown promising performance. However, among the EC algorithms, the ACO paradigm [13], especially its ant colony system (ACS) variant, fits COPs better and has shown particular strengths in solving real-world COPs. Compared with other EC algorithms, such as a GA and particle swarm optimization (PSO) [14], the adoption of global shared pheromone in ACS allows the experience information to be spread rapidly among the colony and thus help the cooperation among multiple ants. Moreover, the introduction of heuristic information enhances the exploration capacity. The balance of exploration of new solution and exploitation of accumulated experience about the problem ensure fast convergence and good performance of ACS. Therefore, the ACS-based algorithm for VMP optimization is extensively studied in this paper [35]. In cloud computing domain, Feller *et al.* applied an ACO-based approach to minimize the number of cloud servers to support current load. However, this method has a high computing cost and consolidates VMs only on a single resource. In this paper [35], the authors consolidate VMs according to multiple resources (i.e., both CPU and memory), being more applicable in cloud computing, but more challenging.

There also exist some reports on the use of multi objective algorithms to minimize the total resource wastage and power consumption. These algorithms include multi objective ACS, multi objective ACO (MACO), and hybrid ACO with PSO, termed HACOPSO. In [15], an ACS is employed for VMs consolidation in dynamic environment to reduce energy consumption but not directly to reduce the number of servers.

The ACS-based approach is used to allocate the VMs in minimum number of physical servers to reduce energy consumption for cloud computing. To handle both homogeneous and heterogeneous server environment, an order exchange and migration (OEM) mechanism for the ACS, resulting in an OEMACS algorithm is developed. Further, the OEMACS algorithm incorporates a new solution evaluation method with a hierarchical structure.

Evaluation

Energy consumption contributes most to the total cost in a cloud system. For this an energy efficient OEMACS for VMP in cloud computing is developed. The optimal VM deployment has been achieved with the minimum number of active servers and by switching off the idle servers. The VMP problem is a complex NP-hard problem. To solve this problem, OEMACS, an ACS-based approach, has been developed in this paper [35]. The assignment of VMs is constructed by artificial ants based on global search information. OMEACS distributes pheromone between VM pairs, which represents a bond among the VMs on the same server and records good VM groups through learning from historical experience. To revise infeasible solutions, local search is performed, which contributes significantly to improving the solutions and speeding up global convergence of the OEMACS. Moreover, the number of servers provided for placing VMs reduces as the generation number grows, avoiding possible wastes of computation while providing guidance for further advancement of the solutions. These distinct features and the strong global search nature of an ACS make the OEMACS efficient for large-scale problems. It shows a significant advantage compared with other heuristic algorithms, which often encounter difficulties when the problem scale grows with cloud computing. The Experimental results show that OEMACS has achieved the objectives of minimizing the number of active servers, improving the resource utilization, balancing different resources, and reducing power consumption. Moreover, the parameter analysis shows that the performance of OEMACS is not very sensitive to the parameters, and this makes the OEMACS more competitive. In conclusion, the OEMACS is seen an effective and efficient approach to the VMP problem.

2.2 Semistatic Cell Differentiation and Integration with Dynamic BBU-RRH Mapping in Cloud Radio Access Network [36]

In this paper [36], a self-organizing cloud radio access network (C-RAN) is proposed, which dynamically adapt to varying network capacity demands. A load prediction model is considered for provisioning and allocation of base band units (BBUs) and remote radio heads (RRHs). The density of active BBUs and RRHs is scaled based on the concept of cell differentiation and integration (CDI) aiming efficient resource utilization without sacrificing the overall quality of service (QoS). A CDI algorithm is proposed in which a semi-static CDI and dynamic BBU-RRH mapping for load balancing are performed jointly. Network load balance is formulated as a linear integer-based optimization problem with constraints. The semistatic part of CDI algorithm selects proper BBUs and RRHs for activation / deactivation after a fixed CDI cycle, and the dynamic part performs proper BBU to RRH mapping for network load balancing aiming maximum QoS with minimum possible handovers. A discrete particle swarm optimization (DPSO) is developed as an evolutionary algorithm to solve network load balancing optimization problem. The performance of DPSO is tested based on two problem scenarios and compared to genetic algorithm (GA) and the exhaustive search (ES) algorithm. The DPSO is observed to deliver optimum performance for small scale networks and near optimum performance for large-scale networks. The DPSO has less complexity and is much faster than GA and ES algorithms. Computational results of a CDI-enabled C-RAN demonstrate significant throughput improvement compared to a fixed C-RAN, i.e., an average throughput increase of 45.53% and 42.102%, and an average blocked users’ reduction of 23.149%, and 20.903% is experienced for proportional fair and round robin schedulers, respectively.

However, Cloud Radio Access Network (C-RAN) is a promising centralised network architecture that can support super-dense small cells deployment. C-RAN is considered to meet the challenges mentioned above and has attracted considerable attention by both academia and MNOs and is a key enabler of Next Generation Mobile Networks (5G).

The C-RAN architecture consists of the following main parts:

- Several BBUs aggregated into a BBU cloud/pool for centralised management and processing.
- Distributed RRHs in a given geographical area.
- The connection between BBUs and RRHs (also referred to as *front-haul*) via an optical transport network.

Unlike conventional cellular networks, where the base stations are not always in peak time and often work in idle states with their resources not fully utilised, in C-RAN, suitable resource allocation schemes can dynamically adjust the logical connection between BBUs and RRHs. It is necessary for a system to optimise its resources according to varying traffic environments. In C-RAN the problem of resource wastage is overcome by dynamically allocating the shared and centralised BBUs resources to the RRHs. Moreover, significant cost and energy savings can be achieved by dynamically scaling the BBUs concerning varying traffic caused by uneven user distribution in the network [16]. C-RAN is feasible to realise the coordinated control between multiple cells by centralised management.

Although the main features in SONs include self-configuration, self-optimisation, and self-healing. However, this paper [36] emphasises on self-optimisation technique in CRAN concerning network performance improvement. The primary focus is to model a multi-objective optimisation problem along with several other criteria necessary to tailor the optimisation objective according to specific system requirements. C-RAN combined with Self-optimising ability can provide MNOs with a flexible network regarding network dimensioning, adaptation to non-uniform traffic and efficient utilisation of network resources. However, before a full commercial C-RAN deployment, several challenges need to be addressed. Firstly, the front-haul technology used must support enough bandwidth for delivering delay sensitive signals (i.e., the 1 ms physical layer processing requirement of LTE). Secondly, the proper BBU-RRH assignment in C-RAN to not only supports collaboration technology like Cooperative Multipoint Processing (CoMP) but also enabling load balancing in the network. Moreover, significant energy savings can be achieved if the RRHs and BBUs are turned on/off in such a way that the QoS of the network is not degraded.

In this context, a two-stage design is proposed in this paper [36] for efficient resource utilisation in a self-optimised CRAN with real time BBU-RRH mapping. Network resources are utilised based on the concept of Cell Differentiation and Integration (CDI) which allows a cell(s) to split into multiple small cells and vice versa in response to a measured load information in one or more cells in the network. CDI allows C-RAN to adapt to varying capacity demands through resource provisioning and allocation. Resource provisioning not only resizes the number of BBUs in the pool to meet the fluctuating traffic demands but also scales the density of active RRHs required to serve a given geographical area. In the first stage, the optimum number of BBUs is computed to serve the load demand, and the RRHs are activated or deactivated based on the concept of CDI to handle network traffic load. In this paper [36], the number of BBUs required to serve the system load at a given time is computed based on a prediction model called Wiener process [17]. In the second phase, the proper BBU-RRH mapping is identified to avoid unbalanced network scenarios while maintaining high levels of QoS. The second stage in this paper [36] is modelled as an integer based linear optimisation problem with constraints.

Related Work

Numerous studies and methods on self-optimisation have suggested addressing the problem of load balancing in cellular networks via SON. When a traffic imbalance is detected among cells, operation parameters are autonomously adjusted such as antenna angle (Antenna tilt) and/or handover parameters to reduce the coverage area to achieve Mobility Load Balancing (MLB). In MLB, the handover thresholds are adjusted following traffic conditions which result in expansion or contraction of virtual transfer areas among adjacent cells and thereby reducing or increasing users in the cells. However, incorrect handover parameter adjustment can cause additional handovers in the network which often leads to handover ping-pongs / delays and radio link failures. Mobility Robustness Optimisation (MRO) is a SON function which aims to eliminate link failures and reduce unnecessary handovers caused by incorrect handover parameters. Power adaptation for load balancing is another technique to effectively change the cell coverage area which in return changes the association of all users in the coverage area. In LTE, Cell Range Expansion (CRE) is a technique which allows Low Power Nodes (LPN) to expand their coverage area and take in users from the Macro Cell. Usually, users associate to the cell which provides the strongest signal. However, in CRE users connect to the LPNs despite receiving the strongest signal from the Macro cell. A comprehensive survey on self-organisation in future cellular networks, which includes a detailed description of the schemes mentioned above along with hybrid approaches and other existing SON load balancing methods in the literature are provided in [18].

Moreover, the benefits of Artificial Intelligence (AI) techniques while designing load balancing SON algorithms are inevitable. Among numerous AI techniques, the Genetic Algorithm (GA) and Swarm intelligence are the most embraced learning algorithms inspired the process of gene evolution and the natural actions of swarms of ants, a shoal of fish, a flock of birds etc, respectively. Many algorithms have been designed to mimic the behaviour of natural organisms, however, Particle Swarm Optimisation (PSO) [20] remains the backbone of swarm intelligence on which all other algorithms are built. Both GA and PSO are widely discussed in studies related to network planning, interference management, routing and coverage optimisation problems.

On the other hand, a number of research studies on enabling technologies for C-RAN exist. Here, some related studies on BBU-RRH mapping along with RRH-UE association are briefly described. Liu and Yu [19] propose a cross-layer framework for downlink multi-hop C-RAN to improve throughput performance by optimising both physical and network layer resources. Also, RRHs beamforming vectors, user RRH association, and network coding-based routing are optimised in an overall design. Pan *et al.* attempt to solve a joint RRH and precoding optimisation problem which aims to minimise network power consumption in a MIMO based user centric C-RAN. In line with this work, Wang *et al.* propose a weighted minimum mean square error (WMMSE) approach to solving the network-wide beam-forming vector optimisation problem for RRH-UE clusters formation. The BBU scheduling is then formulated as a bin packing problem for energy efficient BBU utilisation in a heterogeneous CRAN environment. A dynamic BBU-RRH mapping scheme is proposed in [20] using a borrow-and-lend approach in C-RAN. Overloaded BBUs switch their supported RRHs to underutilised BBUs for a balanced network load and enhanced throughput. Sundaresan *et al.* proposed a lightweight, scalable framework that utilises optimal transmission strategies via BBU-RRH reconfiguration to cater dynamic user traffic profiles. Reference [21] describes the traffic adaptation and energy saving potential of TDD-based heterogeneous C-RAN by adjusting the logical connections between BBUs and RRHs. Lin *et al.* [31] recently investigated an RRH clustering design and proposed a spectrum allocation genetic algorithm (SAGA) to improve network QoS via efficient resource utilisation. Regarding other related work, research initiatives are taken to develop Network Function Virtualisation (NFV) and Software Defined Network (SDN) solutions for CRAN. NFV is an architectural framework that provides a virtualised network infrastructure, functions and NFV orchestrator for control and management. However, SDN is a concept related to NFV. SDN decouples data and control plane to enable directly programmable control plane while abstracting underlying physical infrastructure from applications and services. Although SDN and NFV are not the prime focus of this paper, they are presented in this section for completeness of the C-RAN introduction. Moreover, [22] and [23] provides a comprehensive survey on C-RAN and highlights the challenges, advantages, and implementation issues regarding different deployment scenarios. Also, an in-depth review of the principles, technologies and applications of C-RAN describing innovative concepts regarding many physical layer, resource allocation, and network challenges together with their potential solutions are highlighted in [24].

The proposed model is suitable for the framework of software defined front-haul with optical switching for C-RAN [25]. However, this paper only focuses on the centralised-SON aspect of the structure.

2.3 Minimize the Make-span of Batched Requests for FPGA Pooling in Cloud Computing [37]

Using Field Programmable Gate Arrays (FPGA) as accelerators is gaining popularity in Cloud computing. Usually, FPGA accelerators in a datacenter are managed as a single resource pool. By issuing a request to this pool, a tenant can transparently access FPGA resources. FPGA requests usually arrive in batches. The objective of scheduling is to minimize the make span of a given batch of requests, which is the completion time of the entire batch of jobs. As a result, either the responsiveness is improved, or the system throughput is maximized. The key technical challenge is the existence of multiple resource bottlenecks. An FPGA job can be bottlenecked by either computation (i.e., computation-intensive) or network (i.e., network-intensive), and sometimes by both. To the best of our knowledge, this is the first work that minimizes the make-span of batched requests for an FPGA accelerator pool in Cloud computing that considers multiple resource bottlenecks. In this paper [37], the authors design several scheduling algorithms to address the challenge. The authors implement their scheduling algorithms in an IBM Cloud system. The authors conduct extensive evaluations on both a small-scale testbed and a large-scale simulator. Compared with the Shortest-Job-First scheduling, these algorithms can reduce the make-span by 36.25% and improve the system throughput by 36.05%.

Motivation: FPGA accelerators have become crucial for Cloud computing. In current Cloud datacenters, CPU resources are no longer adequate for many applications, especially for large-scale machine learning tasks. Leading providers/researchers start to integrate various FPGA/GPU accelerators in their platforms [26]. Compared with CPU, these accelerators can significantly boost the performance of many computation-intensive tasks, such as matrix computation, encryption, and signal processing [27]. For many application scenarios, FPGA is more promising than GPU due to its low cost (i.e., hundreds instead of thousands of dollars per piece), low power footprint (i.e., tens instead of hundreds of Watts per piece) and high-power efficiency (i.e., 2-3x more Gflops than GPU per Watt). FPGA accelerators in a datacenter are usually managed as a single resource pool. In such a datacenter, the operator installs FPGA devices in a portion of the server farm due to cost and deployment constraints. Following the Software-as-a-Service (SaaS) model, tenant programs interact with the Cloud by calling the API functions provided by an FPGA service layer. By issuing a request to this layer, a tenant can transparently access FPGA resources. A centralized scheduler maintains status information of each FPGA node. It assigns job requests to accelerators in the resource pool in an online fashion. Tenants are agnostic to the control and status of FPGA accelerators. An application operation usually triggers a large number of computation requests simultaneously for (FPGA) accelerators. For example, the processing of Online Data Intensive applications (OLDI) and real-time analytics involve a multi-tier split-aggregate workflow, and a single process call triggers a large number of computation tasks. Some streaming data processing systems are even batch-based in nature. For example, Spark Streaming aggregates a batch of requests and submits them together for processing. In this paper [37], the authors study how to schedule FPGA accelerators when job requests come in batches.

The objective of FPGA accelerator scheduling is to minimize the make-span of a given batch of FPGA requests. Make-span is the time to complete all job requests in a single batch. A new batch of requests is considered as completed only after the completion of the last task of this batch. For continuous systems (e.g., Spark with FPGA [28]), minimizing the make-span leads to maximized system throughput. For periodical batch scheduling mode (e.g., Spark Streaming), minimizing the make-span leads to minimized missing ratio of application deadlines.

Challenges: The key technical challenge is to address multiple resource bottlenecks. An FPGA job can still be bottlenecked by computation. For example, 12 MB of photos (2 KB each) need about 1 second to be processed by an FPGA accelerator of deep neural network (DNN). If equipped with a 10/40 Gbps fast network, the communication cost of this job can be negligible. We call such FPGA requests computation-intensive. On the other hand, some jobs can be processed by FPGA at line rate. The I/O bottleneck is usually the network, since PCIE bandwidth is much higher. We call such FPGA requests network-intensive. A typical network-intensive example is AES encryption. Sometimes, both network and computation can be bottlenecks. Stick to the DNN example, if the network provides only 1 Gbps per host (as in some legacy datacenters), the network transfer time cannot be ignored any more. The work closest is done by Julio et al. However, their work performs admission control for an FPGA resource pool, whereas this finding provides services to all requests and minimize the make-span. Contributions: In this paper [37], the authors design several scheduling algorithms to address the challenge. Firstly, they formulate the scheduling of computation-intensive FPGA requests as a parallel machine scheduling problem, which is a well-known NP-hard problem. A polynomial time approximation algorithm is given, with a proven approximation ratio of 2. Secondly, they formulate the scheduling of network-intensive FPGA requests as a mixed integer programming problem, which is also NP-hard. The network intensive case is similar to multiprocessor scheduling. This work is the first one which considers the existence of network bottlenecks at both the sender and the receiver sides and yields a 2-approximation algorithm. At last, the authors extend the algorithms to support cases that both data transmission phase and the computation phase are bottlenecked. They implement their algorithms in an IBM Cloud system. This system achieves full FPGA virtualization and pooling on an OpenStack-based cloud. The authors conducted extensive evaluations on both a small-scale testbed and a large-scale simulator. Compared with the Shortest Job First (SJF) scheduling, these algorithms reduce the make-span by 36.25%, while improve the system throughput by 36.05%.

2.4 Intelligent Health Vessel ABC-DE: An Electrocardiogram Cloud Computing Service [38]

The severe challenges of the fast aging population and the prevalence of cardiovascular diseases highlight the needs for effective solutions supporting more accurate and affordable medical diagnosis and treatment. Recent advances in cloud computing have inspired numerous designs of cloud-based health care services. In this paper [38], the authors developed a cloud-computing platform monitored by physicians, which can receive 12-lead ECG records and send back diagnostic reports to users. Aiming to lessen the physicians' workload, the authors implemented an analysis algorithm that can identify abnormal heart rate, irregular heartbeat, abnormal amplitude, atrial fibrillation and abnormal ECG in it. A large number of testing samples were used to evaluate performance. This algorithm achieved a TPR95 (specificity under the condition of negative predictive value being equal to 95%) of 68.5% and 0.9317 AUC (area under the ROC curve) for classification of normal and abnormal ECG records and a sensitivity of 98.51% and specificity of 98.26% for atrial fibrillation classification, comparable to the state-of-the-art results for each subject. The proposed ECG cloud computing service has been applied in Hunan Jinshengda Aerial Hospital Network and it now can receive and analyze ECG records in real time.

Related Work

The majority of existing works focus on how to design an effective cloud-based telemedicine service, rather than being dedicated to provide accurate diagnostic reports. Even though some cloud platforms have implemented ECG analysis functions, only simple classification algorithms or the intra-patient classification paradigm [29] that is not a realistic measure of performance in real applications are involved. In fact, we can design a secure and economical cloud platform by using the Amazon web/database services as well as programming tools such as HTML, jQuery and Java Server Pages (Other solutions include the Windows Azure platform, .NET programming environment, Socket programming and so on), though a few issues need to be studied further. By contrast, the authors have not been able to develop an algorithm that is qualified for interpreting ECG without physicians' confirmation up to now.

A cloud platform that can pro-vide accurate diagnostic reports for ECG records is of great practical significance to the whole society. It is for this reason that the authors propose a different 12-lead ECG cloud computing service, namely intelligent health vessel. Un-der the proposed architecture, ECG records acquired from a set of ambient / body sensors are sent to the cloud platform monitored by physicians, and then corresponding diagnostic reports are sent back to users. In consideration of the huge possible audience, there are a large number of ECG records needing to be interpreted and the workload of physicians will be very heavy. To improve the diagnostic efficiency, the received ECG record is first routed to an interpretation program to identify any abnormalities automatically. After that, the diagnostic result as well as the ECG record is passed on to physicians for further evaluation [30]. The cloud platform offers 24-hour service; so, if physicians are off duty, another form of di-agnostic reports without physicians' confirmation will be sent back to users. Obviously, the ECG analysis algorithm plays a significant role in the platform and there is a pressing need to devise a high-performance one applied in practical use.

In the recent past, a considerable amount of research has been dedicated to heartbeat classification [31], [32]. Generally, the processing flow of the-se works is such that feature vectors, including physiology characteristics with diagnostic value (such as RR interval and morphology feature of QRS complex) and statistical characteristics (such as principal component analysis and wavelet transform), are extracted from heartbeat segments first, and feature selection is conducted when necessary. Afterwards, a number of machine learning algorithms are employed for classification, such as sup-port vector machine and Gaussian mixture model. Although many of them have achieved excellent results on standard ECG databases, their performance tends to be degraded if we deploy them in clinical setting [33]. The reason behind this lies in the following facts: (1) The standard databases don't involve a wide range of representative ECG records. As a result, the classification models obtained using them tends to be dataset-specific and their performance will be degraded drastically when evaluated on a dataset that has different characteristics from their training sets. (2) The application scenario focuses on the inter-patient classification of ECG records. This paradigm is a hard-artificial intelligence task due to inter-individual variation in ECG characteristics and the complexity of clinical data. There is no doubt that a non-linear model with enough complexity is the only qualified one for this task. However, when going through previous work, the authors found that only linear and/or simple nonlinear transformations were involved. (3) Heartbeat classification involving feature extraction and feature selection is just an intermediate step, and the result of each heartbeat segment extracted from an ECG record needs to be summarized. These mentioned sub-processes all increase the chances of generating incorrect diagnostic reports, since there is no perfect solution for each of them.

Aiming to address the mentioned issues, they utilize the Chinese Cardiovascular Disease Database (CCDD) [34], the deep-learning, rule-inference and ensemble-learning technologies to implement real-time ECG analysis and diagnosis. The proposed approach has the following advantages: (1) The CCDD contains a large number of representative ECG records collected from different people in different places. Actual performance in clinical setting can be effectively reflected if classification models are evaluated on it. (2) The deep-learning technology enables us to construct a complex nonlinear model by increasing the number of hidden layers and nodes in each hidden layer. However, it is not good at solving simple pattern recognition problems, thus the authors employ the rule-inference technology to fill in the gaps. As for the ensemble-learning technology, it offers an alternative solution to construct a nonlinear model and serves as a referee when individual base classifiers have different results. (3) We have no use for segmentation of ECG records and feature extraction except R-peak detection, thus the errors generated by introducing sub-processes can be avoided. In the implementation stage, this approach needs a certain number of ECG samples to train the classification model before deploy it to the cloud platform, and the CCDD provides the initial foundation for the training-and-testing solutions. Once the platform sets to launch, it will receive and store a large number of ECG records. We can use them to train the classification model further so that the performance could be improved. Fig. 1 shows the processing flow of the proposed ECG cloud computing service. As we can see, it not only enables users to enjoy convenient and efficient computing services, but also provides accurate medical diagnosis and treatment.

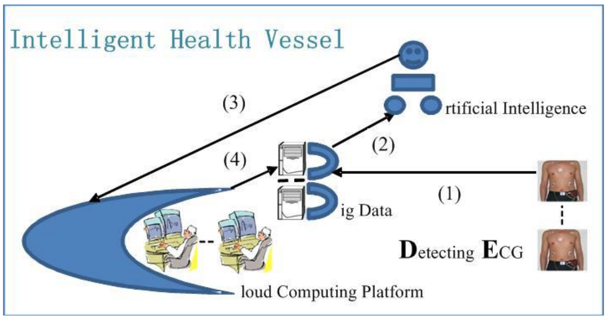

Fig. 1. Intelligent Health Vessel ABC-DE: An Electrocardiogram Cloud Computing Service. Artificial Intelligence (A): analyzes vital signs in real time; Big Data (B): stores data in an open environment; Cloud Computing Platform (C): handles requests and generate diagnostic reports any time any place; (1) acquires data via portable devices; (2) analyses ECG in real time via machines; (3) identifies symptoms via physicians; (4) stores diagnostic data.

3. CONCLUSION

In this paper, we have discussed about the various recently found scheduling algorithms, policies and schemes with their former related algorithms, policies and schemes. It shows that the energy efficient Ant Colony System (ACS) algorithm used for Virtual Machine Placement (VMP) in cloud, self-organizing Cloud Radio Access Network (C-RAN) that dynamically adapt to varying network capacity demands, Field Programmable Gate Arrays (FPGA) accelerator scheduling used to minimize the make-span of a given batch of FPGA requests, and Intelligent Health vessel ABC-DE which is an electro cardiogram computing service where ABC-DE is the abbreviation of Artificial Intelligence, Big Data, Cloud Computing - Detecting ECG (Electro Cardio Gram) are some of the recent algorithms and techniques used for the efficient data access and data computation.

4. REFERENCES

[1] I. Foster, Y. Zhao, I. Raicu, and S. Y. Lu, “Cloud computing and grid computing 360-degree compared,” in Proc. IEEE Grid Comput. Environ. Workshop, Austin, TX, USA, 2008, pp. 1–10.

[2] Z.-G. Chen, K.-J. Du, Z.-H. Zhan, and J. Zhang, “Deadline constrained cloud computing resources scheduling for cost optimization based on dynamic objective genetic algorithm,” in Proc. IEEE Congr. Evol. Comput., Sendai, Japan, 2015, pp. 708–714.

[3] A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, “Distributed energy efficient clouds over core networks,” J. Lightw. Technol., vol. 32, no. 7, pp. 1261–1281, Apr. 1, 2014.

[4] X.-F. Liu, Z.-H. Zhan, J.-H. Lin, and J. Zhang, “Parallel differential evolution based on distributed cloud computing resources for power electronic circuit optimization,” in Proc. Genet. Evol. Comput. Conf., Denver, CO, USA, 2016, pp. 117–118.

[5] Z.-G. Chen et al., “Deadline constrained cloud computing resources scheduling through an ant colony system approach,” in Proc. Int. Conf. Cloud Comput. Res. Innov., Singapore, 2015, pp. 112–119.

[6] Greenpeace. (Apr. 2010). Make It Green: Cloud Computing and Its Contribution to Climate Change. Greenpeace International. [Online]. Available: http://www.thegreenitreview.com/2010/04/greenpeacereports-on-climate-impact-of.html.

[7] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, “The cost of a cloud: Research problems in data center networks,” ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, pp. 68–73, 2009.

[8] W. Vogels, “Beyond server consolidation,” ACM Queue, vol. 6, no. 1, pp. 20–26, Jan./Feb. 2008.

[9] Z. Xiao, Q. Chen, and H. P. Luo, “Automatic scaling of Internet applications for cloud computing services,” IEEE Trans. Comput., vol. 63, no. 5, pp. 1111–1123, May 2014..

[10] R. De S. Couto, S. Secci, M. E. M. Campista, and L. H. M. K. Costa, “Network design requirements for disaster resilience in IaaS clouds,” IEEE Commun. Mag., vol. 52, no. 10, pp. 52–58, Oct. 2014.

[11] S. Chaisiri, B.-S. Lee, and D. Niyato, “Optimal virtual machine placement across multiple cloud providers,” in

Proc. IEEE Asia Pac. Services Comput. Conf., Kuala Lumpur, Malaysia, 2009, pp.103–110.

[12] J. Xu and J. A. B. Fortes, “Multi-objective virtual machine placement in virtualized data center environments,” in Proc. IEEE/ACM Int. Conf. Green Comput. Commun. Int. Conf. Cyber Phys. Soc. Comput., 2010, pp. 179–188.

[13] Q. Yang et al., “Adaptive multimodal continuous ant colony optimization,” IEEE Trans. Evol. Comput., 2016, to be published, doi: 10.1109/TEVC.2016.2591064.

[14] Y. H. Li, Z.-H. Zhan, S. J. Lin, J. Zhang, and X. N. Luo, “Competitive and cooperative particle swarm optimization with information sharing mechanism for global optimization problems,” Inf. Sci., vol. 293, no. 1, pp. 370–382, Feb. 2015.

[15] F. Farahnakian et al., “Using ant colony system to consolidate VMs for green cloud computing,” IEEE Trans. Services Comput., vol. 8, no. 2, pp. 187–198, Mar./Apr. 2015.

[16] M. Khan, R. S. Alhumaima, and H. S. Al-Raweshidy, “Reducing energy consumption by dynamic resource allocation in C-RAN,” in Proc. Eur. Conf. Netw. Commun. (EuCNC), Paris, France, 2015, pp. 169–174.

[17] T. Zhang et al., “Local predictive resource reservation for handoff in multimedia wireless IP networks,” IEEE J. Sel. Areas Commun., vol. 19, no. 10, pp. 1931–1941, Oct. 2001.

[18] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, “A survey of self organisation in future cellular networks,” IEEE Commun. Surveys Tuts., vol. 15, no. 1, pp. 336–361, 1st Quart., 2013.

[19] L. Liu andW. Yu, “Cross-layer design for downlink multihop cloud radio access networks with network coding,” IEEE Trans. Signal Process., vol. 65, no. 7, pp. 1728–1740, Apr. 2017.

[20] Y.-S. Chen, W.-L. Chiang, and M.-C. Shih, “A dynamic BBU–RRH mapping scheme using borrow-and-lend approach in cloud radio access networks,” IEEE Syst. J., to be published, doi: 10.1109/JSYST.2017.2666539.

[21] Z. Yu, K. Wang, H. Ji, X. Li, and H. Zhang, “Dynamic resource allocation in TDD-based heterogeneous cloud radio access networks,” China Commun., vol. 13, no. 6, pp. 1–11, Jun. 2016.

[22] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, “Cloud radio access network (CRAN): A primer,” IEEE Netw., vol. 29, no. 1, pp. 35–41, Jan./Feb. 2015.

[23] M. Peng, Y. Sun, X. Li, Z. Mao, and C. Wang, “Recent advances in cloud radio access networks: System architectures, key techniques, and open issues,” IEEE Commun. Surveys Tuts., vol. 18, no. 3, pp. 2282–2308, 3rd Quart., 2016.

[24] T. Q. Quek, M. Peng, O. Simeone, and W. Yu, Cloud Radio Access Networks: Principles, Technologies, and Applications. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[25] K. Sundaresan, M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy, “FluidNet: A flexible cloud-based radio access network for small cells,” IEEE/ACM Trans. Netw., vol. 24, no. 2, pp. 915–928, Apr. 2016.

[26] B. Li, K. Tan, L. L. Luo, Y. Peng, R. Luo, N. Xu, Y. Xiong, P. Cheng, and E. Chen, “Clicknp: Highly flexible and high performance network processing with reconfigurable hardware,” in Proceedings of the ACM SIGCOMM, 2016, pp. 1–14.

[27] S. Zhou, Y. Zhu, C. Wang, X. Gu, J. Yin, J. Jiang, and G. Rong, “An fpga-assisted cloud framework for massive ecg signal processing,” in IEEE DASC 2014, pp. 208–213.

[28] Y.-T. Chen, J. Cong, Z. Fang, J. Lei, and P. Wei, “When spark meets fpgas: A case study for next-generation dna sequencing acceleration,” in USENIX HotCloud 2016.

[29] C. Ye, B. V. Kumar and M. T. Coimbra, "Heartbeat Classification Using Morphological and Dynamic Features of ECG Signals," IEEE Trans. Bi-omed. Eng., vol. 59, no.10, pp.2930-2941, Oct. 2012.

[30] J. Dong, J. W. Zhang, H. H. Zhu, et al, "Wearable ECG Monitors and Its Remote Diagnosis Service Platform," IEEE Intell. Syst., vol.27, no.6, pp.36-43, Nov. 2012.

[31] L. P. Wang and J. Dong, "The advance research and analysis of ECG pattern classification," Chinese Journal of Biomedical Engineering, vol.29, no.6, pp.916-925, Dec. 2010.

[32] M. Llamedo and J. P. Martinez., "Heartbeat Classification Using Feature Selection Driven by Database Generalization Criteria," IEEE Trans. Bi-omed. Eng., vol.58, no.3, pp.616-625, Mar. 2011.

[33] H. H. Zhu, "Research on ECG Recognition Critical Methods and Development on Remote Multi Body Characteristic Signal Monitoring Sys-tem," Ph.D. dissertation, University of Chinese Academy of Sciences, Beijing, China, 2013.

[34] J. W. Zhang, X. Liu and J. Dong, "CCDD: An Enhanced Standard ECG Database with Its Management & AnnotationTools," Int. J. Artif. Intell. T., vol.21, no.5, pp.1-26, Oct. 2012.

[35] Xiao-Fang Liu, Student Member, IEEE, Zhi-Hui Zhan, Member, IEEE, Jeremiah D.Deng, Member, IEEE, Yun Li, Member, IEEE, Tianlong Gu, and Jun Zhang, Fellow, IEEE “An energy efficient Ant Colony System for Virtual Machine Placement in Cloud Computing”, IEEE Transactions on Evolutionary Computation, Vol.22, No.1, Feb 2018.

[36] M. Khan , Student Member, IEEE, Zainab H. Fakhri, and H. S. Al-Raweshidy, Senior Member, IEEE, "Semistatic Cell Differentiation and Integration with Dynamic BBU-RRH Mapping in Cloud Radio Access Network", IEEE Transactions on Network and Service Management, Vol.15, No.1, March 2018.

[37] M. Khan , Student Member, IEEE, Zainab H. Fakhri, and H. S. Al-Raweshidy, Senior Member, IEEE, "Minimize the make-span of Batched requests for FPGA pooling in Cloud Computing", IEEE Transactions on parallel and distributed systems, DOI 10.1109/TPDS.2018.2829860.

[38] Lin-peng Jin, and Jun Dong, "Intelligent Health vessel ABC-DE: An Electrocardiogram Cloud Computing Service", IEEE Transactions on Cloud Computing, DOI 10.1109/TCC.2018.2825390