2013 IEEE 14th International Conference on Mobile Data Management

Mobile Healthcare Systems with Multi-cloud Offloading

Huaming Wu, Qiushi Wang and Katinka Wolter
Department of Mathematics and Computer Science
Free University of Berlin
Takustr.9, 14195 Berlin, Germany
Email: {huaming.wu, qiushi.wang and katinka.wolter}@fu-berlin.de

***Abstract*—The fast growth of cloud computing has attracted more companies to migrate their in-house IT applications into cloud and it also occurs in the medical field. A mobile healthcare system with cloud offloading is considered in this paper and it can be divided into two stages: sensor network and cloud offloading. In the first stage, information collected by body sensors should be transmitted to a remote mobile device. In order to save energy, an energy-efficient transmission scheme called cooperative multi-input multi-output (MIMO) is constructed for the data transfer when allowing individual sensor nodes to cooperate with each other. In the second stage, two offloading schemes called self-reliant multi-cloud offloading system and multi-cloud offloading system are proposed and further analyzed based on serve topology and optimal graph partition. The former provides stability but with high communication cost, while the latter reduces communication cost but is less stable. Both schemes can be applied to other scenarios in which we would like to perform offloading on multiple servers.**

***Index Terms*—mobile healthcare; cloud computing; sensor network; cooperative MIMO; offloading; partition**

I. INTRODUCTION

The emergence of cloud computing [1] promises to solve some of concerns facing mobile computing platforms since the cloud is regarded as an unlimited resource that can be accessed anytime and anywhere in the world.

Mobile healthcare, which is known as the practice of medical and public health supported by mobile devices for delivering medical and healthcare services is one of the applications that can benefit from offloading the computationally intensive operation onto the cloud.

Mobile healthcare systems frequently use body sensors to collect signals on patients and the signal should be transmitted to a remote mobile device. Thus, minimizing the energy consumption needs to be considered and energy-efficient transmission schemes must be deployed. Fortunately, if we allow individual sensor nodes to cooperate with each other, a cooperative MIMO system can be constructed for the data transfer [2].

The mobile healthcare systems are characterized by low coupling and powerful parallel computing capabilities, therefore offloading can be beneficial. Since there are a number of multimedia sensor signals to be processed on different servers, a traditional offloading scheme that from a single mobile device to a single server is not sufficient. Therefore, to find some more suitable offloading schemes is very important for the mobile healthcare systems. One multi-cloud offloading scheme that comes to mind is offloading from a single mobile device to a different server at each time. But the cost spent on communication is huge since the network bandwidth between the mobile device and cloud is small. However, the bandwidth between the clouds is much larger, and thus a multi-cloud offloading scheme is presented in this paper where data is shifted between clouds.

Accordingly, the main contributions of our research work presented in this paper is two-fold. First, we investigate the stage of transmitting information collected by body sensors to a remote mobile device through an equivalent cooperative multi-input single-output (MISO) approach to save energy consumption. Secondly, we study the stage of offloading program from the mobile device to cloud. Two offloading schemes are proposed and compared based on the serve topology and optimal graph partition.

The remainder of this paper is structured as follows. Section II presents the architecture of mobile healthcare system with cloud offloading. Section III compares an equivalent MISO scheme with non-cooperative approach in a sensor network. The self-reliant multi-cloud offloading and multi-cloud offloading schemes are investigated through serve topologies and partition problem in section IV. Section V concludes the paper.

II. MOBILE HEALTHCARE SYSTEM

A. System overview

The architecture of mobile healthcare systems with cloud offloading is shown in Fig.1. It is comprised of three main components: body sensors for collecting physiological signals, mobile devices for joint processing medical information and delivering healthcare services via mobile technology, cloud servers including the database server, data mining server and graphic server for signal processing [3].

978-0-7695-4973-6/13 $26.00 © 2013 IEEE
DOI 10.1109/MDM.2013.92

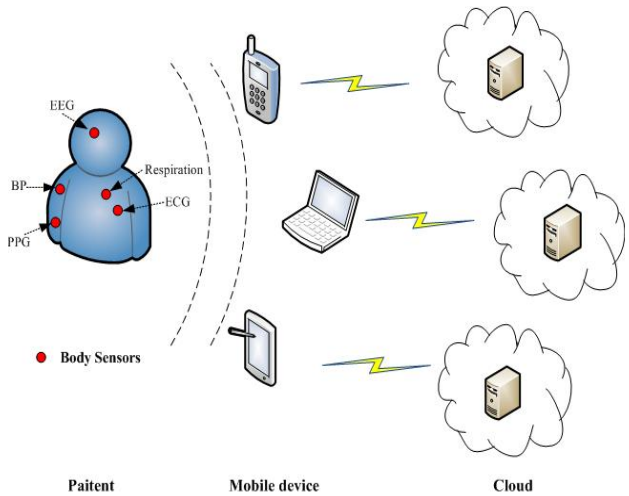


Fig. 1. Architecture of mobile healthcare systems with cloud offloading

1) Body sensor: Many body sensors are used for collecting vital signals such as electrocardiograph (ECG), photoplethysmograph (PPG) and blood pressure (BP) for further analysis [3]. A mobile healthcare system is designed to meet the requirements of different users. For example, the patients with heart diseases who need a long-term monitoring after recovery to prevent its relapse, the hypertension patients who are under the process of medicine adjustment and the sub-healthy people who want to have a good knowledge of and follow up their health conditions to prevent some kinds of chronic diseases.

The collected physiological signals can be transmitted to the mobile device via sensor network.

2) Mobile device: The mobile device such as mobile phone, laptop and PDA, aims to jointly process medical information and deliver healthcare services. Some special software is installed in the mobile device. After collecting data through Bluetooth sent from body sensors, we could get preliminary analysis results such as heart rate, abnormalities of a single test, and etc.

Since mobile devices have limited computational capacity and limited battery life, heavy multimedia and signal processing are unable to run on them. Therefore, to get further analysis, the processing is offloaded to the server in the cloud. [4]

3) Cloud: Recently, there are a large number of cloud platforms appearing in the sky with different cost patterns and conditions, such as Amazon's EC2 [5], Apple's icloud [6], Microsoft's Azure [7], Google's App Engine and so on for data storage and processing. These systems use a proprietary cloud platform to provide a personalized service. The cloud data center specifically designed for healthcare service can provide a platform for large data storage and parallel computing capabilities for data mining.

B. Two stages

The whole mobile healthcare system can be divided into two stages.

1) Sensor network stage: First of all, we need to transmit the information collected by the body sensors to the mobile device. In some cases, the mobile device may be close to the patients, where the Bluetooth can be used, but in most situations, the distance may be a challenge, for example, mobile devices used by the hospital are far from patients at home. In such cases, other transmission methods should be used to save energy consumption and reduce transmit time.

In fact, the body sensors can be seen as sensor networks, and signals collected by multiple body sensors need to be transmitted to a remote mobile device. If the sensors do not cooperate with each other, actually this is the case of multiple SISO (single input single output) transmission from individual sensor to the mobile device. The energy cost would be huge due to the long distance between the local nodes and the remote mobile device. Therefore, a new strategy is needed to minimize the total energy consumption of entire nodes and transmission instead of reducing the energy cost of individual node.

2) Cloud offloading stage: In this stage, the programs in mobile device should be offloaded to the cloud servers for further processing.

The traditional cloud offloading scheme is constrained by offloading computation from a single mobile device to a single server in cloud. However, it can't be applied to full range of scenarios in which we would like to perform offloading. For example, it is not suitable in the above healthcare system since different signal processing such as ECG, PPG, BP and so on, could not be completed only in one server and thus multiple offloading servers should be considered. Therefore, new offloading schemes should be further explored to overcome such complex signal processing.

The above two stages will be focused on in the following sections.

III. Sensor network stage

In this section, we will study the sensor network stage of mobile healthcare systems as illustrated in Fig.1, the transmission of the collected information from local sensors to the remote mobile devices.

A. System model

In the sensor network stage, a sensor network can be abstracted as a mathematical model, which is depicted in Fig.2.

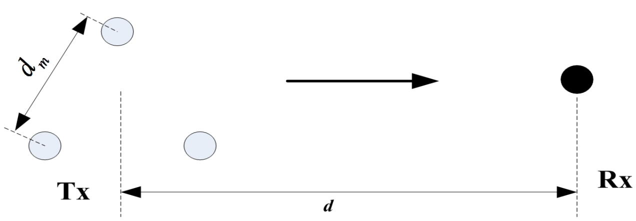


Fig. 2. Architecture of a sensor network

From Fig.2, we assume that there are three body sensors at Tx side which are equal to three nodes, and the mobile device is also abstracted as a node at Rx side. The local distance between each node is d_m, and the distance from the sensors to the mobile device is d, since the mobile device is far away, we usually have $d_m \ll d$.

1) Non-cooperative approach: For the non-cooperative approach, the model in Fig.2 can be treated as M_t mutually independent single-input single-output (SISO) schemes. Each node at Tx side just transmits information to the remote node at Rx side on its own, and the local nodes do not cooperate with each other. The total energy consumption per bit in transmission and circuitry for a single SISO scheme with uncoded MQAM modulation is easily found to be [8]

$$E_{SISO} = E_{tr}\frac{\gamma_0 d^2}{bB} + E_c\frac{2}{bB} \tag{1}$$

where γ_0 denotes the SNR required in the SISO scheme, $b = log_2 M$ is bits per symbol in modulation and M is the constellation size, B is the bandwidth for a fixed transmission, E_{tr} is a constant factor for transmission, and E_c is a constant factor for circuitry.

We assume that there are M_t nodes at the Tx side and each has L_i bits to transmit, where $i = 1, \cdots, M_t$. As a result, the total energy consumption for the non-cooperative approach is given by

$$J_{Non-coop} = \sum_{i=1}^{M_t} L_i E_{SISO} \tag{2}$$

2) MISO approach: As far as we know, MIMO (including MISO, SIMO and MIMO) can save energy in fading channels [2]. Thus, if the multiple local nodes work together in transmitting data to the destination node, they could be treated as multiple antennas and an equivalent MISO system can be constructed.

In order to make the cooperative transmission possible, the M_t nodes at the Tx side will cooperate before the long transmission. Information on each node is broadcasted to all the other local nodes in different time slots. After each node receives all the information from all other nodes, they encode the transmission sequence according to the space-time block code (STBC) [9].

As for MISO strategy, according to the reference [8], the total transmission energy and circuitry energy consumption per bit is

$$E_{MISO} = E_{tr}\frac{d^2}{bB}P_e^{-\frac{1}{M_t}} + E_c\frac{M_t + M_r}{bB} \tag{3}$$

When the system in Fig.2 is treated as cooperative MISO, in addition to the transmission and circuitry energy cost in Equation (3), the energy consumption at the Tx side due to the cooperation overhead needs to be considered. We denote the energy cost per bit for local information flow at the Tx side as E_i, which is expressed as follows

$$E_i = E_{tr}\frac{\gamma_0 d_m^2}{b_i B} + E_c\frac{M_t}{b_i B} \tag{4}$$

where $b_i = log_2 M_i$ is bits per symbol in modulation and M_i is the constellation size during the local transmission at the Tx side.

Therefore, the total energy consumption for the cooperative MISO approach is as follows

$$J_{MISO} = \sum_{i=1}^{M_t} L_i(E_i + E_{MISO}) \tag{5}$$

B. Numerical results

To give numerical examples, we assume that $M_t = 3$, $M_r = 1$, the distance between each sensor is $d_m = 1m$. The information is transmitted from the sensors to the mobile device by using 4QAM modulation, and thus we have $M = 4$. We also set the fixed bandwidth B and bits error rate P_e as $B = 10KHz$ and $P_e = 10^{-3}$, respectively. Besides, the energy constants $E_c = 40uJ$ and $E_t = 10nJ$.

The data to transmit at each node is set as $L_1 = 500Kb$, $L_2 = 1Mb$ and $L_3 = 2Mb$. In the broadcast process of each node, we use 4QAM, 8QAM and 16QAM modulations, respectively. From the BER for SISO scheme $P_e = (1 - \sqrt{\gamma_0/(1+\gamma_0)})/2$ [10], we can calculate the γ_0 for certain P_e. In order to compare the energy consumption in Equation (2) and (5), we obtain the numerical results as shown in Fig.3.

In Fig.3, the total energy cost of the non-cooperative approach and the MISO scheme are plotted over the transmission distance d. From Fig.3, it can be seen that when d is small ($d < 8m$), the non-cooperative transmission approach can still be better energy efficient than the scheme with MISO. But

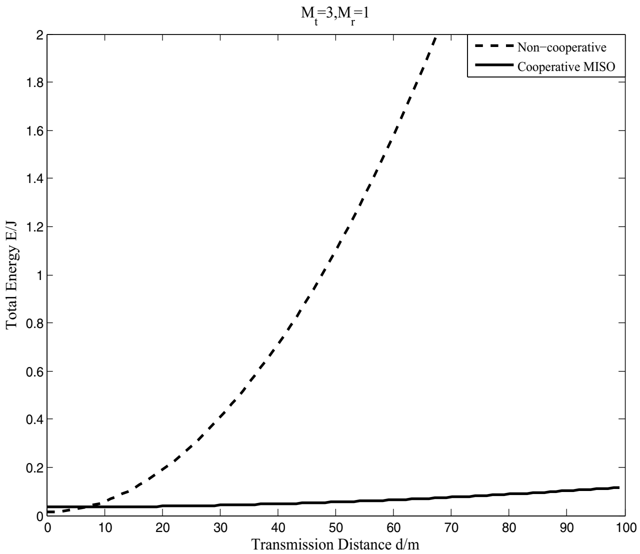


Fig. 3. Total energy consumption over d

when d becomes larger, the transmission energy dominates the energy consumption, and the MISO approach becomes much more energy efficient than the scheme without cooperation.

IV. MULTI-CLOUD OFFLOADING

In this section, we will investigate the second stage of mobile healthcare systems illustrated in Fig.1, that is to offload the program from mobile device to remote cloud for further execution.

The traditional cloud offloading scheme is constrained by offloading computation from a single mobile device to a single server in cloud. However, it can not be applied to full range of scenarios in which we would like to perform offloading. For example, it is not suitable in the above healthcare system since different signal processing such as ECG, PPG and BP could not be finished only in one server and it requires distribute across multiple offloading servers.

Therefore, a new scheme called multi-cloud offloading for the mobile healthcare system is proposed in this paper, which offloads portions of the program to multiple remote clouds.

A. Self-reliant multi-cloud offloading system

A self-reliant multi-cloud offloading system is described as Fig.4. It is actually multiple offloading from the single mobile device to a single server one by one.

We assume that the number of available remote servers is k, $V = \{v_1, v_2, v_3\}$ means that there are three tasks v_1, v_2, v_3 needed to be offloaded, $P = \{p_0, p_1, \ldots, p_k\}$ where p_0 represents the mobile device, and $p_1, \ldots, p_k$ represent the offloading servers. Besides, $w_v^{p_i}$ is the computation cost when

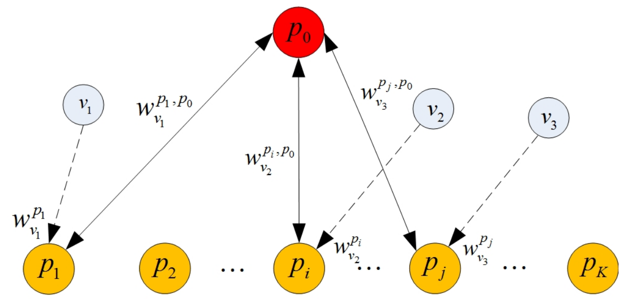


Fig. 4. Self-reliant multi-cloud offloading system

v is assigned to server p_i and $w_v^{p_i,p_0}$ is the communication cost between p_i and p_0 when v is assigned to p_i.

Here, the tasks of v_1, v_2 and v_3 are allocated to p_1, p_i and p_j, the mobile device offloads the computation to each server separately, and after executing the program, the server then sends the processing results back to the mobile device.

The disadvantage of the self-reliant multi-cloud offloading system is that the programs can only be offloaded to different servers one by one separately and such an approach can not support more complex offloads, e.g., parallel offload of different application parts to different servers.

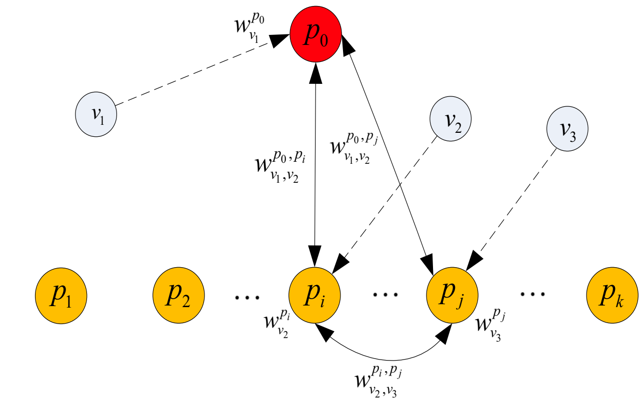
B. Multi-cloud offloading system

In order to overcome the limits of self-reliant multi-cloud offloading system, another multi-cloud offloading system is given as Fig.5, where $w_{v_m,v_n}^{p_i,p_j}$ is the communication cost between p_i and p_j when v_m is assigned to p_i and v_n is allocated to p_j.

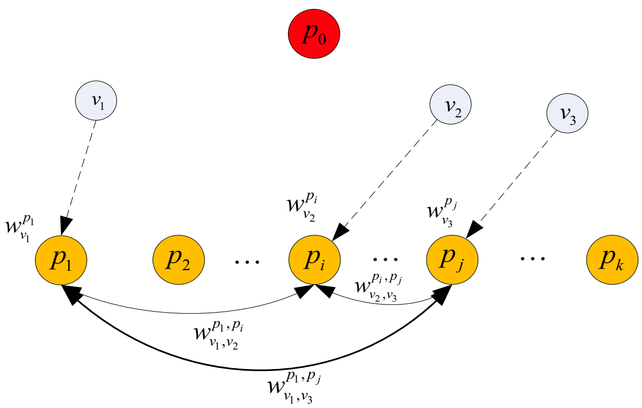
The system depicted in Fig.5a is called partial offloading scheme. Partial programs are offloaded to servers while partial programs are executed locally by the mobile device. It can be seen that the task of v_1 is assigned to the mobile device p_0 while v_2 and v_3 are allocated to p_i and p_j. We compare it with the self-reliant multi-cloud offloading system mentioned in Fig.4. We find that the clouds with the tasks can communicate with each other and also communicate with the mobile device, while the clouds only communicate with the mobile device in Fig.4.

The model shown in Fig.5b is the entire offloading scheme where all the tasks are offloaded to different servers. It can be seen that the tasks of v_1, v_2 and v_3 are allocated to p_1, p_i and p_j. The difference is that the clouds with allocated tasks communicate with each other in Fig.5b while the clouds only communicate with the mobile device in Fig.4.

Since the communication between the two cloud-resident servers such as p_i and p_j may be very fast, while



(a) Partial offloading



(b) Entire offloading

Fig. 5. Multi-cloud offloading systems

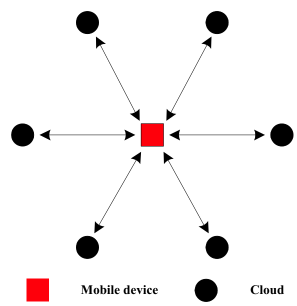
communication between the mobile device and the cloud may be much slower [11], the servers communicate with each other when executing the allocated tasks.

C. Serve topology

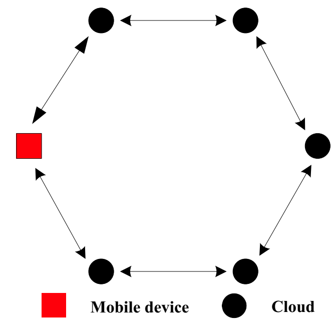
In order to compare the difference between the above two schemes much more vividly, serve topologies of multi-cloud offloading systems are depicted as Fig.6.

It can be found in Fig.6a that the serve topology for self-reliant multi-cloud offloading system is star. The mobile device needs to communicate with different servers in turn. Thus, a lot of time and energy are spent on communication, especially when the network condition is not good enough. However, the offloading system is very stabile due to the separate offloading.

From Fig.6b, it can be seen that the serve topology for multi-cloud offloading system is ring. The mobile device only needs to communicate with two servers. Since the



(a) Self-reliant multi-cloud offloading system



(b) Multi-cloud offloading system

Fig. 6. Serve topology of multi-cloud offloading systems

network bandwidth between two clouds is usually very large, while the communication between the mobile device and cloud may be much slower, this topology is much faster than the star topology. However, the stability is much lower since it depends on every server, and when failure occurs in the middle, the program could not be executed successfully.

D. Partition problem

We can formulate the multi-cloud offloading as graph partition problem with $G = (V, E)$, where V are the vertices and edges $E \in V \times V$. Every cloud has different computational and storage capacities, and hence it requires different weights of nodes, or different network bandwidth that requires different edge weights for communication [12].

We can cast such partition challenge as an optimization problem, and the optimization goal could be to minimize the battery consumption, to minimize the local storage needs while keeping communication cost low, to minimize the computation time and so on.

The optimization problem for given tasks V and servers P is shown as follows [13]

$$\begin{aligned} C \quad = \quad & minimize \sum_{v \in V} \sum_{p_i \in P} w_v^{p_i} \cdot m_v^{p_i} + \\ & \sum_{v_m, v_n \in V} \sum_{p_i, p_j \in P} w_{v_m, v_n}^{p_i, p_j} \cdot m_{v_m, v_n}^{p_i, p_j} \end{aligned} \tag{6}$$

where

$$\sum_{v \in V} \sum_{p_i \in P} w_v^{p_i} \cdot m_v^{p_i}$$

is the total cost of computation and

$$\sum_{v_m, v_n \in V} \sum_{p_i, p_j \in P} w_{v_m, v_n}^{p_i, p_j} \cdot m_{v_m, v_n}^{p_i, p_j}$$

denotes the total cost of communication.

$$m_v^{p_i} = \begin{cases} 1 & \text{if } v \text{ is assigned to } p_i \\ 0 & \text{otherwise} \end{cases}$$

and

$$m_{v_m,v_n}^{p_i,p_j} = \begin{cases} 1 & \text{if } v_m \text{ is assigned to } p_i \text{ and } v_n \text{ to } p_j \\ 0 & \text{otherwise} \end{cases}$$

Further, we have

$$\forall v \in V : \sum_{p_i \in P} m_v^{p_i} = 1 \tag{7}$$

which enforces that each vertex is assigned to exactly one partition.

Each node and edge are assigned with a different cost, respectively, which depends on the partition result of the application graph that the node or the edge belongs to.

The example given in Fig.5b can be further expressed as a cost matrix,

$$\begin{array}{c} \\ p_1 \\ p_2 \\ \vdots \\ p_i \\ \vdots \\ p_j \\ \vdots \\ p_k \end{array} \begin{array}{c} \begin{array}{ccccccccc} p_1 & p_2 & \cdots & p_i & \cdots & p_j & \cdots & p_k \end{array} \\ \begin{bmatrix} 0 & \infty & \cdots & w_{v_1,v_2}^{p_1,p_i} & \cdots & w_{v_1,v_3}^{p_1,p_j} & \cdots & \infty \\ \infty & 0 & \cdots & \infty & \cdots & \infty & \cdots & \infty \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ w_{v_1,v_2}^{p_1,p_i} & \infty & \cdots & 0 & \cdots & w_{v_2,v_3}^{p_i,p_j} & \cdots & \infty \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ w_{v_1,v_3}^{p_1,p_j} & \infty & \cdots & w_{v_2,v_3}^{p_i,p_j} & \cdots & 0 & \cdots & \infty \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \infty & \infty & \cdots & \infty & \cdots & \infty & \cdots & 0 \end{bmatrix} \end{array}$$

where the graph partition is assumed as undirected graph and the edge weight that does not exist is set as ∞.

V. Conclusion

A mobile healthcare system with cloud offloading is investigated and it can be divided into sensor network and cloud offloading stages.

In order to save energy cost on the entire sensor nodes, an energy-efficient transmission scheme is constructed for the data transfer when allowing individual nodes to cooperate with each other. The cooperative MISO approach seems to be much more energy efficient than the scheme without cooperation when the mobile device is far away from the mobile device.

The proposed two new schemes of multi-cloud offloading for mobile healthcare infrastructure are analyzed and compared based on graph partition and service topology. It's actually the tradeoff between stability and communication cost, and both schemes can be applied to other range of scenarios in which we would like to perform offloading on multiple servers.

Through the multi-cloud offloading system, patients can learn about their health information and even the risk factor of some chronic diseases in the future.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica and M. Zaharia, "Above the Clouds: A berkeley view of cloud computing," *Technical Report No. UCB/EECS-2009-28, Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley*, Feb. 10, 2009.

[2] S. Cui, A. J. Goldsmith and A. Bahai, "Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1089-1098, 2004.

[3] F. Miao, X. Miao, W. Shangguan and Y. Li, "MobiHealthcare System: Body Sensor Network Based M-Health System for Healthcare Application," *E-Health Telecommunication Systems and Networks*, vol. 1, no. 1, pp. 12-18, March 2012.

[4] M. T. Nkosi and F. Mekuria, "Cloud Computing for Enhanced Mobile Health Applications," in *Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom'10)*, pp. 629-633, 2010.

[5] http://aws.amazon.com/ec2/.

[6] http://www.apple.com/ icloud/.

[7] https://windows.azure.com/.

[8] W. Liu, X. Li and M. Chen, "Energy efficiency of MIMO transmissions in wireless sensor networks with diversity and multiplexing gains," in *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, vol. 4, pp. 897-900, 2005.

[9] S. Sandhu and A. Paulraj, "Space-time block codes: A capacity perspective," *IEEE Communication Letters*, vol. 12, no. 12, pp. 384-386, 2000.

[10] S. Cui, A. J. Goldsmith and A. Bahai, "Modulation optimization under energy constraints," in *Proceedings of the 2003 IEEE International Conference on Communications (ICC'03)*, vol. 4, pp. 2805-2811, May 2003.

[11] H. Wu, Q. Wang and K. Wolter, "Methods of cloud-path selection for offloading in mobile cloud computing systems," in *Proceedings of the 2012 IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom'12)*, pp. 443-448, Dec. 2012.

[12] K. Sinha and M. Kulkarni, "Techniques for Fine-Grained, Multi-site Computation Offloading," in *Proceedings of the 2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CC-Grid'11)*, pp. 184-194, May 2011.

[13] B. G. Chun and P. Maniatis, "Dynamically partitioning applications between weak devices and clouds," in *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing and Services: Social Networks and Beyond (MCS)*, no. 7, June 2010.