International Journal of Grid Distribution Computing
Vol.8, No.2 (2015), pp.15-24
http://dx.doi.org/10.14257/ijgdc.2015.8.2.03

A Data Transmission Method for Resource Monitoring under Cloud Computing Environment

Ying Jiang, Heng Sun, Jiaman Ding and Yingli Liu

Yunnan Key Lab of Computer Technology Application, Kunming University of Science and Technology
Faculty of Information Engineering and Automation, Kunming University of Science and Technology
jy_910@163.com,80815893@qq.com, tjom2008@163.com, lyl2002@126.com

Abstract

Currently, the combined push-pull model is usually applied in the resource monitoring under cloud computing environment. However, the quantity of data gathered by the producers is much less than those received by the monitor so that a lot of resource is wasted. In order to maintain the data consistency of real time system, an improved adaptive algorithm based on combined push-pull model is proposed. The adaptive strategy is added on the producers that aims to predict when refreshes data came from the monitor using the history data. Experimental result shows that the algorithm can decrease the resource consumption while exchanging data effectively.

Keywords: *cloud computing, resource monitoring, push-pull model, adaptive*

1. Introduction

Cloud computing is a relatively new business model in the computing world. According to the official NIST definition, cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (*e.g.,* networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. Using cloud computing, the user can apply or release resources quickly according to the business load and pay for the used resources by pay-as-you-need, which can reduce operational cost while improving service quality.

The virtual resources is the foundation to construct the flexible services, which usually running on top of large scale clusters which contain a large number of cheap PC servers. The reliability of service depends on the reliability of virtual resources [2, 3]. Therefore, it is necessary to monitor the virtual resources under cloud computing environment. That is the premise of network analysis, job scheduling, load balancing, event prediction, fault detection and recovery. Because there are plenty of heterogeneous, dynamic and complex resources in clouding computing, the monitoring is more complex than traditional network monitoring [4, 5].

The remainders of this paper are organized as follows. Related works are summarized in Section 2. In Section 3, an adaptive data transmission method based on a hybrid P&P model is presented. In Section 4, monitoring metrics is described to estimate the efficiency of monitoring. The experiments are described in Section 5. In Section 6, the conclusion and future work are proposed.

2. Related Work

For all kinds of monitoring systems, it is essential to ensure certain monitoring capability while the systems being monitored can run normally. So the monitoring system

ISSN: 2005-4262 IJGDC

must not only occupy less system cost but also assure monitoring capability to meet user's monitoring requirements [6]. Therefore, the following two factors need to be considered.

(1) Low interference. The more data to be collected, the greater the system being monitored is interfered. The monitoring capability cannot be ensured while using too many system resources. It is necessary to establish the balance between the monitoring capability and system resources.

(2) High efficiency. Yin pointed that it is necessary to maximize the resources usage efficiency [7]. During the process of monitoring, the resource utilization can be gained through comparing the amount of data produced by monitored object and those received by monitoring system. If the whole or most of collected information are transmit to the monitoring system, the resources that consumed to gather the status can be transformed into usable data. Then the resource utilization can be improved.

At present, the method of data transmission during the resource monitoring is usually push model and pull model [8]. In the push model, the producers push the new status to the consumers under some trigger conditions. In the pull model, the consumers are responsible for pulling information from the producers to inquire status [9]. Zhang pointed that the push model consumes less network resources and its real-time property is good while consuming more computing and storage resources. At the same time, the pull model consumes more network resources and its real-time property is poor while consuming less computing and storage resources [6].

Based on the push model, Chung et al. tried to maintain the consistency between the consumers and producers through reducing the transmission of useless monitoring status, which can improve the performance of the push model at certain extent [10]. Sundaresan, *et al.,* proposed an adaptive model to estimate the time of next update status based on the pull model, which obtain time series information using collector in order to improve the accuracy. However, the monitoring system based on the pull or push model cannot fully avoid the disadvantages of the pull or push model [11, 12]. In order to take the advantages of both push and pull models, Bhide, *et al.,* proposed the hybrid push and pull (P&P) model to meet the user requirements and resource status in the field of network transmission [13]. This approach exchanges data using both push and pull method, and the extent of the pull or push is determined by coordinated parameters. Huang et al. introduced this algorithm into cloud computing [14]. Compared with the single push model or pull model, the hybrid P&P model can reduce the number of updates, system intervention and resource consumption.

However, if the consumers do not know when to request data, the producers usually update collected data in accordance with a fixed time. If the consumers do not know that the producers how to update data, the consumers will inquire the data according to their own policies. Therefore, the number of updated data is less than that of obtained by monitoring system, which will result in the resources waste. Srinivasan, *et al.,* presented an adaptive algorithm based pull model in the field of web data transmission [15]. The experiments indicated that this algorithm is the most effective way to maintain UTD (User Tolerant Degree). UTD shows the user patience for the data consistency, which specifically expresses as the maximum value of difference of the same monitoring data items in two data collection. The smaller value means that the higher consistency for data collection [13].

Currently, there are plenty waste of resources during the data transmission due to the consumers and the producers can not synchronize. Under cloud environment, resource monitoring need meet UTD while reducing resource consumption and improving resource utilization.

Huang, *et al.,* proposed a hybrid P&P model that using the adaptive strategy in the producers [14]. The method dynamically updates the time interval to obtain data according to the changes in two adjacent data collection. When the difference between the current data and previous data is greater than UTD, the time interval is reduced.

Otherwise, the time interval is increased. Then UTD can be meeting while reducing the data transmission. The amount of data communication between the consumers and the producers is reduced. Because the producers use the fixed time interval as an update frequency, the data obtained by the producers cannot guarantee to be transmitting to the consumers. Then the resources would be wasted.

In the stock system, Srinivasan et al. studied selection strategy of stock price TTR (Time To Refresh) while transmitting stock price from trading center to the client [15]. They presented the various selection strategy of update time, including static selection strategy, semi-dynamic selection strategy and dynamic selection strategy of TTR. After comparing the several sets of experiments, a dynamic selection strategy of TTR is the most effective to maintain UTD.

Aimed at the problems under cloud computing environment, we will propose an adaptive data transmission method based on a hybrid P&P model.

3. An adaptive Data Transmission Method based on a Hybrid P&P Model

In order to reduce the resource consumption of the producers, the adaptive strategy needs to add for the producers. The producers will analyze the current data and historical data. Then the next update time can be predicted, which will reduce the amount of update data and resource consumption of the producers.

The prediction need analyze the existing historical data. It will have a greater consumption of resources if the producers analyze the historical data immediately and frequently. Thus, the historical data closed the current time is selected to study. The next predicted value will be obtained through analyzing the historical data and the current data.

Step 1: the data update interval will be calculated to meet UTD during the process of the prediction.

Due to the requirements of UTD, TTR need be adjusted automatically according to the data changes. TTR should be reduced when the changes of data item is much frequent. On the contrary, TTR should be increased.

D_1, D_2, …, $D_{latest-1}$, D_{latest}, D_{next} indicate the values of data item D. $D_{latest-1}$ indicates the previous value. D_{latest} indicates the current value. The time interval to collect these two data is TTR_{latest}. D_{next} indicates the next value that will be obtained. The update interval $TTR_{estimate}$ can be predicted by analyzing the current value D_{latest} and the previous value $D_{latest-1}$. The process is shown in the formula (1) to (4).

During the period of TTR_{latest}, the data change rates $RATE_{latest}$ can be represented by the formula (1).

$$RATE_{latest} = \frac{\left|D_{latest} - D_{latest-1}\right|}{TTR_{latest}} \tag{1}$$

Supposed that next data change rates is as the same as the current $RATE_{latest}$, the next time interval to collect data $TTR_{estimate}$ should satisfy the formula (2).

$$\left|D_{next} - D_{latest}\right| = RATE_{latest} \times TTR_{estimate} \tag{2}$$

In order to ensure the next data change is not more than UTD, the formula (3) should be satisfied at the same time.

$$\left|D_{next} - D_{latest}\right| \le UTD \tag{3}$$

Finally, the formula (4) is produced through brought the critical condition of the formula (1), (3) into the formula (2).

$$TTR_{estimate} = \frac{TTR_{latest}}{\left|D_{latest} - D_{latest-1}\right|} \times UTD \tag{4}$$

If the next data change is as the same as the current data change, the obtained data that using $TTR_{estimate}$ predicted by the formula (4) can meet UTD.

Step 2: the update time interval based on the historical data and current data will be calculated by adjusting the parameters of prediction model.

In order to increase the accuracy of prediction, the final prediction results TTR_{dyn} will be determined through the historical data and the previous update interval TTR_{latest}. The definition is shown in the formula (5).

$$TTR_{dyn} = \omega \times TTR_{estimate} + (1-\omega) \times TTR_{latest} \tag{5}$$

ω ($0.5 \le \omega < 1$) is a weight value, which is used to adjust the weight of the result of Step 1 and the previous update interval TTR_{latest}. The value of ω can be updated dynamically through studying the historical data and current data, just as shown in the formula (6) and (7).

$$\delta = \frac{\left|D_{latest} - D_{latest-1}\right|}{\left|D_{latest-1} - D_{latest-2}\right|} \tag{6}$$

$$\omega = \begin{cases} \dfrac{\delta}{\delta+1}, \delta \ge 1 \\ \dfrac{1}{\delta+1}, \delta < 1 \end{cases} \tag{7}$$

Usually, the initial value of ω is at least 0.5. The influence of the result of Step 1 for TTR_{dyn} is as the same as that of the previous update interval TTR_{latest}. If the value of $\left|D_{latest} - D_{latest-1}\right| / \left|D_{latest-1} - D_{latest-2}\right|$ is close to ∞ or 0, it means that the difference of change between the current data and the previous data is very big. Therefore, the weight of $TTR_{estimate}$ should be increased. Now the value of ω is close to 1.

In order to make the value of TTR_{dyn} has a limit, the final adaptive interval of data update $TTR_{adaptive}$ can be calculated by the formula (8).

$$TTR_{adaptive} = Max\ (\ TTR_{min}, Min(TTR_{max}, \alpha \times TTR_{mr} + (1-\alpha)\ TTR_{dyn})\ ) \tag{8}$$

TTR_{mr} is the smallest update interval, $[TTR_{min}, TTR_{max}]$ is the change boundary of TTR. α ($0 \le \alpha \le 1$) is a parameter. If the value of α is bigger, $TTR_{adaptive}$ can meet UTD better. However, the consumption of the network resource is bigger.

Based on the above, we do not use the fixed data update interval for the producers. The next data update time will be calculated by the formula (8). The algorithm is shown in Figure 1.


```
Input : the value of UTD, ω, α, TTR_mr, D_latest, D_latest-1, D_latest-2 and TTR_latest


WHILE TRUE
    IF there is request of the consumer
        obtain sensor_now of the current sensor;
        // sensor_now indicates the data stored in monitoring sensor
        D_latest-2 ← D_latest-1, D_latest-1 ← D_latest,
        D_latest ← sensor_now,  pull the value of D_latest to the consumer;
    ELSE
        obtain sensor_now of the current sensor;
        IF |sensor_now − D_latest| / (MAX − MIN) ≥ UTD
        //MAX and MIN indicate the upper and lower limit of collected data
            D_latest-2 ← D_latest-1, D_latest-1 ← D_latest,
            D_latest ← sensor_now, push the value of D_latest to the consumer;
        ENDIF
        ENDIF
    calculate TTR_adaptive using the formula (4) to (8);
    IF TTR_adaptive < TTR_mr
        TTR_mr ← TTR_adaptive;
    ENDIF
    Wait TTR_adaptive;
    TTR_latest ← TTR_adaptive;
ENDWHILE
```

Figure 1. The Algorithm of the Producers

The algorithm of the consumers is used the algorithm in [14].

4. Monitoring Metrics

Monitoring metrics can be used to measure the effectiveness of data transmission methods. Aimed at the information system, Aebi, *et al.*, proposed the evaluation of data quality indicators from the consistency, correctness, minimality and completeness [16]. Since the object of the monitoring system and information systems is to obtain, store and deal with data, we define the Quantity of Fresh Data on Producers and Completeness Degree to indicate the amount and integrity of data quality.

1. Quantity of Fresh Data on Producers (QDP)

QDP shows the effect of the producers to the monitored object. It is the amount of data that the producers update within a given period. In addition to reflect the data quality, QDP also reflects amount of the resources consumption. The smaller value of QDP means that the impact on the monitored objects is smaller and the less consumption of resources.

2. Completeness Degree (CD)

CD shows the data amount that collected data distributed at the boundary. Quantity of Overstep Boundary Data (QOBD) is the amount that collected data over a larger value or below a smaller value within a given time. By analyzing the historical collected data, QOBD is the average of maximum and minimum values of daily collected data. The bigger value of QOBD/QDM (Quantity of Received Data on Monitor) indicates the more comprehensive description of the monitored object and the higher data quality. CD is defined in the formula (9).

$$CD = \frac{QOBD}{QDM} \tag{9}$$

Furthermore, we use Collection Efficiency to measure the synchronization and monitoring efficiency between the consumers and producers.

3. Collection Efficiency (CE)

CE is used to measure the synchronization degree of monitoring. Because the resources will be consumed while the producers updating the data, it is a waste if consumed resources do not produce a corresponding monitoring data. Sun pointed that the information collection efficiency can be improved by reducing the information collection frequency [17]. And the data amount obtained by monitoring should also be considered. The value of QDM/QDP in the same period can indicate the collection efficiency. CE is defined as in the formula (10).

$$CE = \frac{QDM}{QDP} \quad (10)$$

4. Quantity of Communication (QC)

QC is used to measure communication cost of communication algorithm [13]. The data under cloud computing environment is also transmitted through the network, so we use QC to evaluate network communication cost. The content of network communications includes the collected data and some control information, such as request data. The value of QC indicates the resources amount consumed by algorithm.

5. Experiment Analysis

In order to simulate the result of data transmission both our method and the method of [14] as much real as possible, the experiments are executed in CloudSim [17, 18]. We use the node logs in PlanetLab project to simulate the resource use under cloud environments. These logs record the CPU operation of each node. Experiments select CPU load rate of part nodes as the monitoring object.

The experimental configuration is shown in Table 1. P&P indicates the method in [14] and MPP indicates our method. Three experiments are executed under these configurations. The value of UTD, MIN and MAX are set to 0.1, 0.1 and 0.6 respectively. The procedures collect CPU load rate and transmit to the consumers.

Table 1. Experimental Configuration

Method	Consumers				Producers			
	PII	PIMAX	PIMIN	PFI	α	ω	TTR_{min}	TTR_{max}
P&P	50s	120s	30s	15s	null	null	null	null
MPP	50s	120s	30s	null	0.5	0.5	30s	100s

PII:PULL_INI_INTERVAL; PIMAX: PULL_INTERVAL_MAX; PIMIN: PULL_INTERVAL_MIN; PFI: PUSH_FIXED_INTERVAL

For the three experiments, 10, 100, 1000 nodes are selected as experimental objects. We use P&P and MPP method to collect data of these nodes during 24 hours. Then these data are analyzed using the monitoring metrics in Section 4. The results are shown in Table 2.

Table 2. Monitoring Metrics of Part Nodes

Experiment No.	Node No.	P&P				MPP			
		QDP	CD	CE	QC	QDP	CD	CE	QC
Ex1	1	5760	0.001	0.231	2017	986	0.000	0.662	1141
	2	5760	0.019	0.298	2501	1108	0.022	0.767	1447
	3	5760	0.200	0.323	2621	1086	0.202	0.838	1528
	4	5760	0.004	0.135	1421	1057	0.002	0.458	944
	5	5760	0.005	0.237	2057	988	0.006	0.691	1178
	6	5760	0.995	0.134	1416	1218	0.994	0.409	971
	7	5760	0.972	0.138	1143	1268	0.980	0.397	989
	8	5760	0.992	0.124	1358	1298	0.994	0.375	969
	9	5760	0.799	0.180	1790	1203	0.836	0.480	1070
	10	5760	0.017	0.145	1479	1109	0.018	0.451	969
Ex2	10	5760	0.017	0.145	1479	1109	0.018	0.451	969
	20	5760	0.926	0.134	1420	1281	0.947	0.394	986
	30	5760	0.882	0.165	1600	1076	0.890	0.490	995
	40	5760	1.000	0.115	1310	1370	1.000	0.360	986
	50	5760	0.838	0.170	1637	1254	0.880	0.451	1058
	60	5760	0.480	0.135	1420	1255	0.496	0.400	984
	70	5760	0.004	0.139	1446	1095	0.002	0.455	962
	80	5760	0.981	0.135	1420	1273	0.986	0.394	985
	90	5760	0.882	0.134	1431	1562	0.932	0.348	1066
	100	5760	0.920	0.132	1418	1489	0.950	0.359	1048
Ex3	100	5760	0.920	0.132	1418	1489	0.950	0.359	1048
	200	5760	0.669	0.174	1656	1082	0.700	0.506	1017
	300	5760	0.840	0.140	1468	1716	0.895	0.340	1120
	400	5760	1.000	0.121	1343	1358	1.000	0.366	992
	500	5760	0.410	0.189	1742	1013	0.434	0.573	1042
	600	5760	0.001	0.159	1516	1016	0.000	0.520	986
	700	5760	0.368	0.295	2456	1065	0.398	0.762	1368
	800	5760	0.864	0.152	1529	1436	0.899	0.393	1064
	900	5760	0.971	0.127	1380	1283	0.982	0.398	999
	1000	5760	0.987	0.249	2145	1021	0.989	0.700	1227

In order to compare the experimental results, the upper limit of CD is 0.6 and the lower of CD limit is 0.1. The average value of the monitoring metrics in three experiments are shown in Figure 2 and 3.

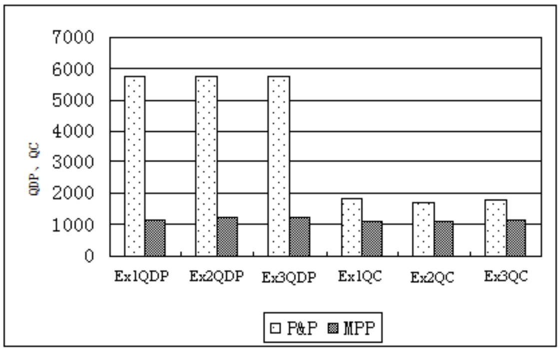

Figure 2. The Average Value of QDP and QC in Three Experiments

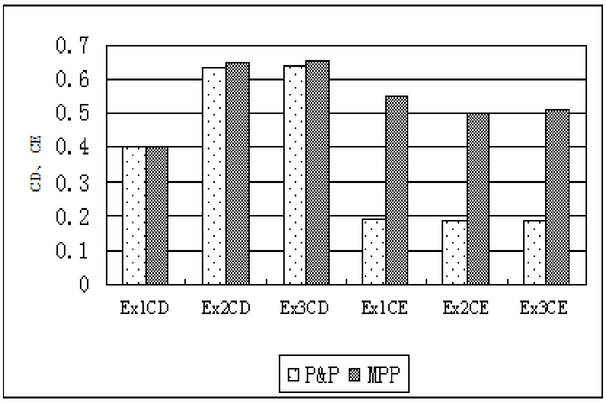

Figure 3. The Average Value of CD and CE in Three Experiments

As shown in Figure 2 and 3, the average value of the monitoring metric in P&P is QDP 5760, QC 1800, CD 0.5, CE 0.18. The average value of the monitoring metric in MPP is QDP 1200, QC 1100, CD 0.5, CE 0.52. So, the average value of our method is better than that of method in [14].

When the value of CD is close to 0, there are some nodes that method in [14] is better than our method. The result is shown in Figure 4. At the horizontal axis, the number before the brackets represents the Experiment No., and the number in the brackets represents the Node No.

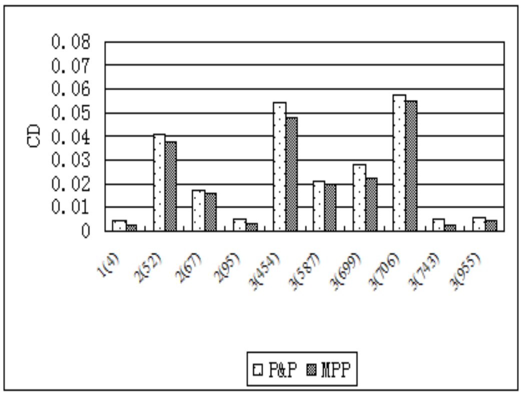

Figure 4. The Value of CD of Part Nodes

In Figure 4, because the update data amount of the producers in our method is much less than that of in [14], the individual data is not sensitive enough.

6. Conclusion and Future Work

In order to reduce resource consumption during the monitoring data transmission, this paper proposes that the producers use adaptive strategies and obtain the next update time through studying the historical data and prediction. Experimental results show that this method can improve the collection efficiency and reduce resource consumption effectively.

Our method is only considering the data update policy for the producers. We will further study fault tolerance, data caching to find more complete, and efficient transmission method under cloud environment.

Acknowledgements

This research is sponsored by the National Science Foundation of China No. 60703116 and 61063006, and the Application Basic Research Plan in Yunnan Province of China No. 2013FZ020.

References

[1] P. Mell and T. Grance, "The NIST definition of cloud computing (draft)", NIST special publication, **(2011)**, pp. 800-145.
[2] K. Chen and W. M. Zheng, "Cloud Computing: System Instances and Current Research", Journal of Software, vol. 20, no. 5, **(2009)**, pp. 1337-1348.
[3] Q. Chen and Q. N. Deng, "Clouding computing and its key techniques", Journal of Computer Applications, vol. 29, no. 9, **(2009)**, pp. 2562-2567.
[4] I. Foster, Y. Zhao, I. Raicu and S. Lu, "Cloud computing and grid computing 360-degree compared", Grid Computing Environments Workshop, **(2008)** November 16, pp. 1-10.
[5] J. Brandt, A. Gentile, J. Mayo and P. Pebay, Resource monitoring and management with OVIS to enable HPC in cloud computing environments", IEEE International Symposium on Parallel & Distributed Processing, **(2009)** May 23-29, pp. 1-8.
[6] Q. S. Zhang, "Research and application of cloud computing platform monitoring system", Beijing Jiaotong University, Beijing, **(2011).**
[7] H. B. Yin, "The design of monitoring platform of distributed application system under clouding computing", Central South University of Forestry and Technology, Changsha, **(2012)**.
[8] S. Zanikolas and R. Sakellariou, "A taxonomy of grid monitoring systems", Future Generation Computer Systems, vol. 21, no. 1, **(2005)**, pp. 163-188.
[9] J. W. Ge, B. Zhang and Y. Q. Fang, "Study on Resource Monitoring Model in Cloud Computing Environment", Computer Engineering, vol. 37, no. 11, **(2011)**, pp. 31-33.
[10] W. C. Chung and R. S. Chang, "A new mechanism for resource monitoring in Grid computing", Future Generation Computer Systems, vol. 25, no. 1, **(2009)**, pp. 1-7.
[11] R. Sundaresan, T. Kurc, M. Lauria, S. Parthasarathy and J. Saltz, "A slacker coherence protocol for pull-based monitoring of on-line data sources", 3rd IEEE International Symposium on Cluster Computing and the Grid, **(2003)** May 12-15, pp. 250-257.
[12] R. Sundaresan, M. Lauria, T. Kurc, S. Parthasarathy and J. Saltz, "Adaptive polling of grid resource monitors using a slacker coherence model", 12th IEEE International Symposium on High Performance Distributed Computing, **(2003)** June 22-24, pp. 260-269.
[13] M. Bhide, P. Deolasee, A. Katkar, A. Panchbudhe, K. Ramamritham and P. Shenoy, "Adaptive push-pull: Disseminating dynamic web data", IEEE Transactions on Computers, vol.51, no. 6, **(2002)**, pp. 652-668.
[14] H. Huang and L. Wang, "P&P: A combined push-pull model for resource monitoring in cloud computing environment", IEEE 3rd International Conference on Cloud Computing, **(2010)** July 5-10, pp. 260-267.
[15] R. Srinivasan, C. Liang and K. Ramamritham, Maintaining temporal coherency of virtual data warehouses, The 19th IEEE Real-Time Systems Symposium, **(1998)** December 2-4, pp. 60-70.
[16] D. Aebi and L. Perrochon, "Towards Improving Data Quality", Proceedings of the International Conference on Information Systems and Management of Data, **(1993)** October 6-8, pp. 273-281.
[17] A. T. Sun, "Research and implementation of key technology in cluster monitoring system", Liaoning Normal University, Dalian, **(2010).**
[18] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. DeRose and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", Software: Practice and Experience, vol. 41, no. 1, **(2011)**, pp. 23-50.

Authors

Ying Jiang, Ph. D. and professor. Her research interests include software engineering, software testing and service computing.

Heng Sun, Master candidate. His research interests include software testing technology and service computing.

Jiaman Ding, Master and lecturer. His research interests include software engineering and cloud computing.

Yingli Liu, Master and lecturer. Her research interests include software engineering and cloud computing.

