JCSE Journal of Computing Science and Engineering

Regular Paper

Journal of Computing Science and Engineering,
Vol. 8, No. 1, March 2014, pp. 17-24

An Exponential Smoothing Adaptive Failure Detector in the Dual Model of Heartbeat and Interaction

Zhiyong Yang*, Chunlin Li, Yanpei Liu, Yunchang Liu, and Lijun Xu

School of Computer Science, Wuhan University of Technology, Wuhan, China, and
Key Laboratory of Fiber Optic Sensing Technology and Information Processing, Ministry of Education, China
yzywhlgdx@163.com, chunlin74@aliyun.com, 32933415@qq.com, 1147843397@qq.com, and 77782840@qq.com

Abstract

In this paper, we propose a new implementation of a failure detector. The implementation uses a dual model of heartbeat and interaction. First, the heartbeat model is adopted to shorten the detection time, if the detection process does not receive the heartbeat message in the expected time. The interaction model is then used to check the process further. The expected time is calculated using the exponential smoothing method. Exponential smoothing can be used to estimate the next arrival time not only in the random data, but also in the data of linear trends. It is proven that the new detector in the paper can eventually be a perfect detector.

Category: Smart and intelligent computing

Keywords: Failure detector; Exponential smoothing; Eventually perfect

I. INTRODUCTION

A failure detector is a key building block for fault-tolerant distributed system, which provide a mechanism to collect information of process failure. Chandra and Toueg [1] introduced the concept of an unreliable failure detector and many fault-tolerance algorithms have been proposed based on unreliable failure detectors.

The fixed timeout Δ_{to} is set as a constant value in a conventional failure detector and the monitoring process begins to suspect the monitored process if it does not receive a heartbeat message after time Δ_{to}. The disadvantage is that it is difficult to determine an appropriate Δ_{to}. If Δ_{to} is short, it is easy to make a wrong inference; if Δ_{to} is long, it results in a long time expense of detection time.

Recently, some adaptive failure detectors have been presented [2-6]. Most of them are based on a heartbeat strategy and modify the timeout value dynamically according to the network conditions. Chen et al. [7] estimate (hereinafter, Chen's estimation) the next heartbeat arrival time by computing the average transmission time and a fixed safety margin is added. Bertier et al. [8] combine Chen's estimation and a dynamical safety margin based on Jacobson's [9] estimation of the round-trip time.

In addition, some studies [10-12] introduce Omega failure detectors (type of failure detectors) which judge process failure not according to the check point, but according to an output value ϕ $(0 \le \phi \le 1)$ in order to present the reliability of the process.

The next part of this section will introduce the failure detector properties and the quality of service of the failure detector.

Open Access http://dx.doi.org/10.5626/JCSE.2014.8.1.17 http://jcse.kiise.org

Received 9 January 2014; **Revised** 22 January 2014; **Accepted** 4 February 2014
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

A. Failure Detector Properties

Failure detectors are characterized by two properties: completeness and accuracy. Two kinds of completeness and four kinds of accuracy are defined [1, 13].

Completeness characterizes the failure detector's capability to suspect every incorrect process permanently. Two kinds of completeness are defined.

1) Strong completeness: Eventually, every process that crashes is permanently suspected by every correct process.
2) Weak completeness: Eventually, every process that crashes is permanently suspected by some correct process.

Accuracy is the characterization of the failure detector's capability not to suspect correct processes. Four kinds of accuracy are defined.

1) Strong accuracy: No process is suspected before it crashes.
2) Eventual strong accuracy: There is a time after which correct processes are not suspected by any correct process.
3) Weak accuracy: Some correct process is never suspected.
4) Eventual weak accuracy: There is a time after which some correct process is never suspected by any correct process.

Eight classes of failure detectors are yielded by combining the two kinds of completeness and four kinds of accuracy, named as perfect, P; eventually perfect, $\diamond P$; strong, S; eventually strong, $\diamond S$; Q; $\diamond Q$; weak, W; eventually weak, $\diamond W$.

Generally speaking, a good failure detector should be a *perfect* or *eventually perfect* detector. In this paper we design an eventually perfect detector. The detector meets the requirement of strong completeness and eventual strong accuracy.

B. Quality of Service of Failure Detectors

Some metrics have been proposed to specify the quality of service of a failure detector [14]. The main metrics are as follows:

- Detection time: The time from when the process crashes to the time when it is permanently suspected.
- Mistake recurrence time: The time between two consecutive mistakes.
- Mistake duration: The duration time for the failure detector to correct a mistake.

The rest of this paper is organized as follows: Section II describes our failure detection model; Section III presents the new algorithm in our failure detector; Section IV proves that it is an eventually perfect failure detector; Section V presents the performance evaluation by a series of experiments; and Section VI consists of the conclusion of our research.

II. THE MODEL OF FAILURE DETECTION

A. The Heartbeat Model

The heartbeat model [15] is used in most distributed systems. Every process p periodically sends an "I am alive" heartbeat message to the process q. The period is the heartbeat interval Δ_i.

If q does not receive a heartbeat message from p after a timeout delay Δ_{to}, p is added to the list of suspected processes. If q receives the heartbeat message from p later, then q removes p from its list of suspected processes, as shown in Fig. 1.

The heartbeat interval Δ_i:Δ_i is the time between two emissions of the "I am alive" heartbeat message. The timeout delay Δ_{to}:Δ_{to} is the time between the last reception of the "I am alive" message from p and the time where q starts suspecting p. The transmission delay Δ_{tr}:Δ_{tr} is the time between the emission of the heartbeat message and the reception of the heartbeat message.

B. The Interaction Model

The process q monitors every process p by sending an "Are you alive?" message to p periodically. Once p receives the message from q, it replies with an "I am alive" message to q. If q does not receive the message from p after a timeout delay Δ_{to}, p is added to the list of

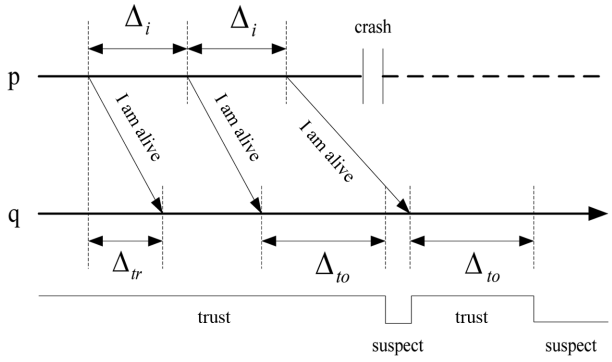

Fig. 1. The heartbeat model.

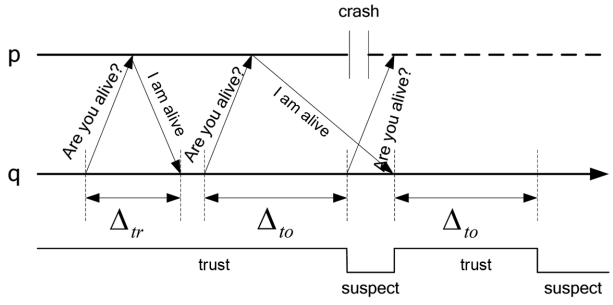

Fig. 2. The interaction model.

suspected processes. If q receives the heartbeat message from p later, then q removes p from its list of suspected processes, as shown in Fig. 2.

C. The Dual Model of Heartbeat and Interaction

The heartbeat failure detector sends half as many messages as the interaction detector for the same detection quality, so the heartbeat failure detector is used in most of the distributed systems. However, it often mistakes actual failure statuses due to loss of packets or long delays in a complicated network. The interaction failure detector needs more detection time and sends more messages than the heartbeat detector, but it can be implemented as a requirement, and the detection result is independent of the message.

In this paper, we will use a dual model of heartbeat and interaction. The dual model is organized in two steps. First, the heartbeat model is used to detect the process failure. If a process p is suspected, then the interaction model is used to check the process p further. The detailed procedure is as shown in Fig. 3.

III. AN EXPONENTIAL SMOOTHING ADAPTIVE FAILURE DETECTOR

Chen's estimation is a classical adaptive failure detector algorithm. It estimates the arrival time of the next heartbeat message according to the historical time sequences of the heartbeat messages. In addition, a constant safety margin is added to raise the detector's accuracy.

The process q receives n heartbeat messages denoted m_1, m_2, m_3, ..., m_n. The receipt time of the messages are presented as A_1, A_2, A_3, ..., A_n and Δ_i is the heartbeat interval. Then, the estimation arrival time of the next message can be expressed as follows:

$$EA_{n+1} = \frac{1}{k}\sum_{n-k}^{n}(A_i - \Delta_i * i) + (k+1)*\Delta_i . \quad (1)$$

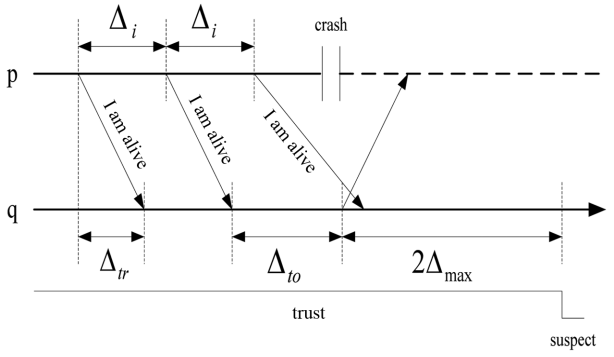

Fig. 3. The dual model of heartbeat and interaction.

The timeout checking point is

$$\tau_{n+1} = EA_{n+1} + \partial , \quad (2)$$

where ∂ is the constant safety margin.

Chen's failure detection algorithm can estimate the arrival time of the next heartbeat message dynamically. However, it uses a constant safety margin, and the suitable constant ∂ is difficult to determine in a condition with a complex network. In addition, the algorithm uses the average transmission delay of the most recent k messages to estimate the next transmission time. And it is only suitable for the fluctuation sequence with small fluctuations. If the delay sequences increase or decrease in a linear manner, Chen's estimation does not correspond to the real delay.

An improved estimation algorithm based on exponential smoothing is proposed to overcome the shortcomings of the base algorithm. Exponential smoothing [16] is a technique that can be applied to a time data sequence. It makes forecasts according to a series of historical data.

The delay data sequence is represented by $\{x_i\}$, $x_i = A_i - \Delta t * i$ and the estimation of the exponential smoothing algorithm is written as $\{s_i'\}$. The simplest form of exponential smoothing is given by the formulas

$$\begin{aligned} s_1' &= x_1 \\ s_i' &= \alpha * x_i + (1-\alpha) * s_{i-1}' \end{aligned} \quad (3)$$

where α is the smoothing factor, and $0 < \alpha < 1$.

The above simple exponential smoothing does not perform well when there is a trend in the data. In such situations, double exponential smoothing is used to estimate the following data in a sequence with a linear trend. Double exponential smoothing works as follows:

$$\begin{aligned} s_1' &= x_1 \\ s_1'' &= x_1 \\ s_i' &= \alpha * x_i + (1-\alpha) * s_{i-1}' \\ s_i'' &= \alpha * s_i' + (1-\alpha) * s_{i-1}'' \\ F_{i+m} &= a_i + m * b_i \end{aligned} \quad (4)$$

where a_i is the estimated level at time i and b_i is the estimated trend at time i.

Then,

$$\begin{aligned} a_i &= 2 * s_i' - s_i'' \\ b_i &= \frac{\alpha}{1-\alpha} * (s_i' - s_i'') \end{aligned} \quad (5)$$

$$\begin{aligned} EA_{i+1} &= (i+1)*\Delta_i + F_{i+1} \\ &= (i+1)*\Delta_i + a_i + b_i \\ &= (i+1)*\Delta_i + \frac{2-\alpha}{1-\alpha} * s_i' - \frac{1}{1-\alpha} s_i'' . \end{aligned} \quad (6)$$

Another improvement in the algorithm is that the dynamic safety margin is calculated using the mean square deviation. As shown in Fig. 1, if the two adjacent heartbeat messages have the same transmission delay, then, $A_{i+1} = A_i + \Delta_t$, so $A_{i+1} - (A_i + \Delta_t)$ can represent the fluctuation in the adjacent transmission delay, and it may be positive or negative.

Finally, we have

$$\partial = \sqrt{\frac{1}{i-1}\sum_{1}^{i-1}(A_{j+1} - A_j - \Delta_t)^2} \text{ , and} \quad (7)$$

$$\tau_{i+1} = EA_{i+1} + \partial . \quad (8)$$

Algorithm 1 is the exponential smoothing adaptive failure detector (ESA_FD) algorithm.

Algorithm 1 EDF_FD algorithm

1: **procedure** EDF_FD (A, Δ_i), where A is the history time series data, and Δ_i is the heartbeat interval.
2: $suspect_q = \Phi$
3: $trust_q = \Phi$
4: **if** time = i * Δ_i
5: p sends a heartbeat message m_i to q
6: Calculate the checking point τ_{i+1}, as detailed in formulas (4), (5), (6), (7), (8).
7: **if** $tr_{i+1} < \tau_{i+1}$
8: {$trust_q = trust_q \cup p$; **return true**;}
9: **else**
10: {
11: q sends heartbeat message to p;
12: **if** p does not receive a response message after a time of 2* $\Delta_{\max}$
13: {$suspect_q = suspect_q \cup p$; **return false**}
14: **else**
15: {$trust_q = trust_q \cup p$; **return true**;}
16: }
17: **end procedure**

IV. PROOF

Our failure detection algorithm can meet the requirements of class $\diamond P$. In other words, it has the properties of strong completeness and eventual strong accuracy.

Theorem 1. Strong completeness.

Every crashed process p is permanently suspected by every correct process.

$\exists t_0 : \forall t \geq t_0$, $\forall q \in correct(t)$,

$\forall q \in crashed$, $p \in suspect_q(t)$.

Before providing proof, we defined several parameters:

$\Delta_{\max}$: The unknown bound time threshold on the message transmission between processes p and q.
m_k: The k^{th} message that process p sends to q.
ts_k : The time when process p sends m_k to q.
tr_k : The time when process q receives m_k.
t_c : The time of a crash of process p.

When the process p crashes at t_c, p stops sending "I am alive" messages. Then $ts_k \leq t_c$, so there is an upper bound $t_c + \Delta_{\max}$ after the time when process q can't receive any more messages from p.

The next thing to be proven is that τ_{i+1} exists as an upper bound.

$$\tau_{i+1} = (i+1)*\Delta_i + F_{i+1} + \partial$$

$$F_{i+1} = a_i + b_i = \frac{2-\alpha}{1-\alpha}s_i' - \frac{1}{1-\alpha}s_i'' < \frac{2-\alpha}{1-\alpha}s_i'$$

$$\begin{aligned} s_i' &= \alpha * x_i + (1-\alpha) * s_{i-1}' \\ &= \alpha * x_i + (1-\alpha) * (\alpha * x_{i-1} + (1-\alpha) * s_{i-2}') \\ &= \alpha * (x_i + (1-\alpha) * x_{i-1} + (1-\alpha)^2 * x_{i-2} + \\ &\quad \ldots\ldots + (1-\alpha)^{i-2} x_2) + (1-\alpha)^i x_1 \\ &< \alpha * x_{\max} * (1 + (1-\alpha) + (1-\alpha)^2 + \\ &\quad \ldots\ldots + (1-\alpha)^{i-2}) + (1-\alpha)^i x_{\max} \\ &< (1/\alpha + 1)\, x_{\max} \end{aligned}$$

where $x_{\max}$ is the maximum value in $\{x_i\}$.

$$\begin{aligned} \partial &= \sqrt{\frac{1}{i-1}\sum_{1}^{i-1}(A_{i+1} - A_t - \Delta_t)^2} \\ &< \sqrt{\frac{1}{i-1}\sum_{1}^{i-1}\Delta_{\max}{}^2} = \Delta_{\max} \end{aligned}$$

So there is an upper bound $(i+1) * \Delta_i + \frac{2-\alpha}{1-\alpha} * (\frac{1}{\alpha}+1) * x_{\max} + 3\Delta_{\max}$ in the worst condition after which process q starts suspecting process p if it does not receive any messages from p.

Theorem 2. Eventual strong accuracy.

Theorem 2 states that there is an upper bound time after which the correct processes are not suspected by any correct process.

$\exists t_{bound}, \forall t \geq t_{bound}, \forall q,$

$p \in correct(t), \ p \notin suspect_q(t)$.

However, since the maximum transmission delay is Δ_{max}, the dual model of heartbeat and interaction is used in this paper. In the worst network conditions, the checkpoint is $\tau_i + 2\Delta_{max}$, so a correct process will not be suspected by any correct process.

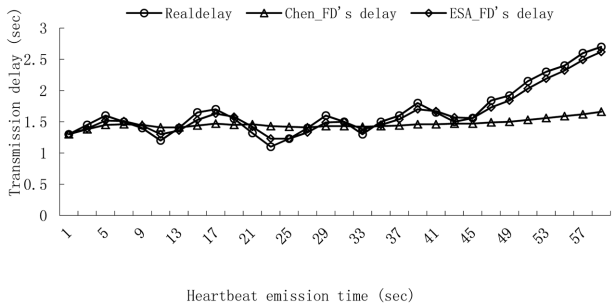

Fig. 5. The transmission delay contrast. ESA_FD: exponential smoothing adaptive failure detector.

V. EXPERIMENT

Two computers are used to build the experimental platform in a local area network. One machine runs a program to send heartbeat messages (like process *p*), and the other machine runs the monitored program to receive heartbeat messages (like process *q*). We simulate complex network conditions by uploading files at a random time, and neither machines fail during the experiments.

In the paper, we compare ESA_FD to conventional FD and Chen's FD. The conventional FD sets a fixed timeout Δ_{to}. If the monitor process does not receive the heartbeat message after Δ_{to}, it begins to suspect the monitored process. A good Δ_{to} is very important to the FD. If Δ_{to} is short, it's easy to make a wrong suspicion. Otherwise, it results in a long time expense in detection time, but this is hard to determine under complex network conditions.

Chen's FD estimates the arrival time of the next heartbeat message dynamically and it uses the average transmission delay of the most recent *k* messages to compute the next transmission time. To improve the accuracy, an additional safety margin ∂ is added. However, ∂ is a constant value in Chen's FD.

Experiment 1. Transmission delay contrast.

We count thirty transmission times of the heartbeat messages and the heartbeat interval is set to 2 seconds. We compute the transmission delay according to the estimation of the arrival time by using Chen's FD and ESA_FD. The result is shown in Fig. 5.

Since the average delay is used to estimate the arrival delay in Chen's FD, Chen's FD is only suitable for a series of random data and there is a big difference between the real transmission delay and Chen's estimation when the data has a linear trend. Whereas the estimation in ESA_FD confirms the real transmission delay very well, exponential smoothing uses a weight value to emphasize the recent data and ESA_FD is suitable for the data not only for random but also for linear trends.

Experiment 2. The average mistake rate contrast.

Two groups of tests are involved in computing the average mistake rate. The heartbeat interval of the first group is set to 2 seconds and the other is set to 5 seconds. The fixed timeout in the conventional FD is set to 1.8 seconds and the safety margin in Chen's FD is set to 0.5 seconds. The result is shown in Fig. 6.

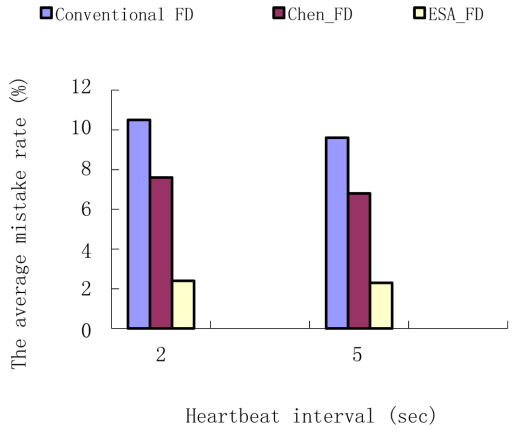

Fig. 6. The average mistake rate contrast. ESA_FD: exponential smoothing adaptive failure detector.

Since the conventional FD uses a fixed timeout to detect a failure, and it is difficult to choose the fixed value to fit complex network conditions, so the fixed timeout of 2 seconds may not be the most suitable value since it results in a high mistake rate of about 10%. Chen's FD is better than the conventional FD, but it is not suitable for data with a linear trend. Therefore, the safety margin is an important adjustment for the result, and in the same manner as the conventional FD, it is hard to determine the value of the safety margin. The ESA_FD in this paper has a very low mistake rate providing accurate estimation to the transmission delay and dual detection starts while a process timeouts first. This further raises the accuracy of detection.

Experiment 3. The average detection time contrast.

The parameters are set as in Experiment 2. In the experiment, we count the detection time and the result is

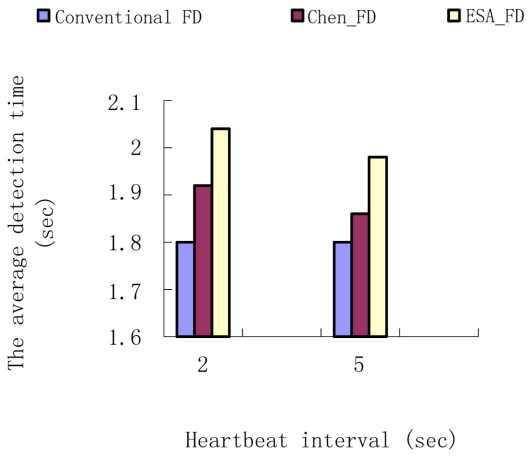

Fig. 7. The average detection time contrast. ESA_FD: exponential smoothing adaptive failure detector.

shown in Fig. 7.

EFA_FD uses dual detection while a process is suspected, so it takes more time than Chen's FD. However, since the estimation to the transmission is more precise, the amount of time for dual detection is shorter, and the difference between the Chen's FD and ESA_FD is very small. So EFA_FD raises the detection accuracy by increasing detection time by a small margin.

VI. CONCLUSION

ESA_FD is an improved adaptive failure detector, which is more suitable for complex network conditions than Chen's FD. On the one hand, ESA_FD uses exponential smoothing to predict the next data and the prediction through exponential smoothing is more accurate than Chen's prediction, which is calculated with the average value, because Chen's FD is only suited for random data with little fluctuation. Hence, ESA_FD can more accurately predict the series of data not only for random, but also in linear trends in applications where the weight value is of exponential smoothing. On the other hand, ESA_FD calculates the safety margin with the mean square deviation and it varies dynamically according the historical transmission delay. ESA_FD meets the requirements of strong completeness and eventually strong accuracy and the experimental results show that ESA_FD can increase the accuracy substantially by increasing the detection time by a small amount.

AKNOWLEDGMENTS

The work was supported by the National Natural Science Foundation (NSF) under grants (No. 61171075); the Special Fund for Fast Sharing of Science Paper in Net Era by CSTD (FSSP) No. 20130143110021; the Program for the High-end Talents of Hubei Province; the Specialized Research Fund for the Doctoral Program of Higher Education under grant (No. 20120143110014); and the Open Fund of the State Key Laboratory of Software Development Environment (SKLSDE-2013KF).

REFERENCES

1. T. D. Chandra and S. Toueg, "Unreliable failure detectors for reliable distributed systems," *Journal of the ACM*, vol. 43, no. 2, pp. 225-267, 1996.
2. S. Bansal, S. Sharma, and I. Trivedi, "Adaptive staircase multiple failure detector for parallel and distributed image processing," in *Proceedings of the 1st International Conference on Recent Advances in Information Technology*, Dhanbad, India, 2012, pp. 91-94.
3. D. Tian, T. Mao, and J. Xie, "Adaptive failure detection algorithm for grid systems," in *Fuzzy Information and Engineering*, Heidelberg, Germany: Springer-Verlag, pp. 288-296, 2009.
4. T. Ma, J. Hillston, and S. Anderson, "On the quality of service of crash-recovery failure detectors," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, no. 3, pp. 271-283, 2010
5. C. Wu, K. Wu, L. Feng, and D. Tian, "NN-SA based dynamic failure detector for services composition in distributed environment," in *Advanced Data Mining and Applications*, Heidelberg, Germany: Springer-Verlag, pp. 443-450, 2010.
6. L. Luan, S. F. Liu, and X. J. Zhang, "Research and improvement of failure detector algorithm based on fresh point," *Journal of Jilin University (Science Edition)*, vol. 46, no. 4, pp. 681-686, 2008.
7. W. Chen, S. Toueg, and M. K. Aguilera, "On the quality of service of failure detectors," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 561-580, 2002.
8. M. Bertier, O. Marin, and P. Sens, "Implementation and performance evaluation of an adaptable failure detector," in *Proceedings of the International Conference on Dependable Systems and Network*, Washington, DC, 2002, pp. 354-363.
9. V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM Computer Communication Review*, vol. 18, no. 4, pp. 314-329, 1988.
10. C. Martin, M. Larrea, and E. Jimenez, "On the implementation of the Omega failure detector in the crash-recovery failure model," in *Proceedings of the 2nd International Conference on Availability, Reliability and Security*, Vienna, Austria, 2007, pp. 975-982.
11. L. Shi and Y. S. Hou, "Adaptive failure detection model based on message delay prediction," *Journal of Computer Applications*, vol. 30, no. 5, pp. 1312-1315, 2010.
12. N. Hayashibara, X. Defago, R. Yared, and T. Katayama, "The φ accrual failure detector," in *Proceedings of the 23rd IEEE International Symposium on Reliable Distributed Systems*, Florianopolis, Brazil, 2004, pp. 66-78.
13. B. Liu, S. Yang, L. Shi, X. Ding, and Q. Zhang, "Modeling

of failure detector based on message delay prediction mechanism," *Journal of Software*, vol. 6, no. 9, pp. 1821-1828, 2011.
14. M. Bertier, O. Marin, and P. Sens, "Performance analysis of a hierarchical failure detector," in *Proceedings of the International Conference on Dependable Systems and Networks*, San Francisco, CA, 2003, pp. 635-644.
15. N. Hayashibara and M. Takizawa, "Performance analysis of the interrogation-based failure detector," in *Proceedings of the 27th International Conference on Distributed Computing Systems Workshops*, Toronto, Canada, 2007, p. 34.
16. Exponential smoothing, http://en.wikipedia.org/wiki/Exponential_smoothing.

Zhiyong Yang

Zhiyong Yang is currently a Ph.D. student in the Department of Computer Science at Wuhan University of Technology. He received his B.S. in 2001 and his M.S. in 2007 from Wuhan University of Technology, China. His research interests are in cloud computing, smart and intelligent computing, and computer networks.

Chunlin Li

Chunlin Li is a Professor of Computer Science at Wuhan University of Technology. She received her M.S. in Computer Science from Wuhan Transportation University in 2000 and her Ph.D. in Computer Software and Theory from Huazhong University of Science and Technology in 2003. Her research interests include cloud computing and distributed computing.

Yanpei Liu

Yanpei Liu is a currently a Ph.D. student in the Department of Computer Science at Wuhan University of Technology. She received her B.S. from Luoyang Normal University, China in 2006 and her M.S. from Nanchang Hangkong University, China in 2009. She has worked at Henan Institute of Science and Technology since 2009. Her research interests are in cloud computing.

Yunchang Liu

Yunchang Liu is currently a Ph.D. student in the Department of Computer Science at Wuhan University of Technology. He received his B.S. from Henan University of Economics and Political Science, Zhengzhou, China, in 1998 and his M.S. from Hubei University of Technology, Wuhan, China, in 2007. He has worked at Pingdingshan Industry School since 1999. His research interests are in cloud computing.

Lijun Xu

Lijun Xu is currently a Ph.D. student in the School of Computer Science and Technology at Wuhan University of Technology and is an associate professor at Xinxiang University. He received his B.S. from Henan Normal University, Xinxiang, Henan, China, in 2001, and his M.S. from the Wuhan University of Technology, Wuhan, Hubei, China, in 2007. He has worked at Xinxiang University since 2001. His research interests include resource management in cloud computing environments and computer networks. He has published about 10 papers.