http://www.icact.org/journal.asp

ICACT-TACT JOURNAL

Transactions on Advanced Communications Technology

Volume 5 Issue 4, Jul. 2016, ISSN: 2288-0003

Editor-in-Chief
Prof. Thomas Byeongnam YOON, PhD.

GIRI Global IT Research Institute

Journal Editorial Board

■ Editor-in-Chief

■ Editors

Dr. Youssef SAID, Tunisie Telecom, Tunisia
Dr. Noor Zaman, King Faisal University, Al Ahsa Hofuf, Saudi Arabia
Dr. Srinivas Mantha, SASTRA University, Thanjavur, India
Dr. Shahriar Mohammadi, KNTU University, Iran
Prof. Beonsku An, Hongik University, korea
Dr. Guanbo Zheng, University of Houston, USA
Prof. Sangho Choe, The Catholic University of Korea, korea
Dr. Gyanendra Prasad Joshi, Yeungnam University, korea
Dr. Tae-Gyu Lee, Korea Institue of Industrial Technology(KITECH), korea
Prof. Ilkyeun Ra, University of Colorado Denver, USA
Dr. Yong Sun, Beijing University of Posts and Telecommunications, China
Dr. Yulei Wu, Chinese Academy of Sciences, China
Mr. Anup Thapa, Chosun University, korea
Dr. Vo Nguyen Quoc Bao, Posts and Telecommunications Institute of Technology, Vietnam
Dr. Harish Kumar, Bhagwant institute of technology, India
Dr. Jin REN, North china university of technology, China
Dr. Joseph Kandath, Electronics & Commn Engg, India
Dr. Mohamed M. A. Moustafa, Arab Information Union (AIU), Egypt
Dr. Mostafa Zaman Chowdhury, Kookmin University, Korea
Prof. Francis C.M. Lau, Hong Kong Polytechnic University, Hong Kong
Prof. Ju Bin Song, Kyung Hee University, korea
Prof. KyungHi Chang, Inha University, Korea
Prof. Sherif Welsen Shaker, Kuang-Chi Institute of Advanced Technology, China
Prof. Seung-Hoon Hwang, Dongguk University, Korea
Prof. Dal-Hwan Yoon, Semyung University, korea
Prof. Chongyang ZHANG, Shanghai Jiao Tong University, China
Dr. H K Lau, The Open University of Hong Kong, Honh Kong
Prof. Ying-Ren Chien, Department of Electrical Engineering, National Ilan University, Taiwan
Prof. Mai Yi-Ting, Hsiuping University of Science and Technology, Taiwan
Dr. Sang-Hwan Ryu, Korea Railroad Research Institute, Korea
Dr. Yung-Chien Shih, MediaTek Inc., Taiwan
Dr. Kuan Hoong Poo, Multimedia University, Malaysia
Dr. Michael Leung, CEng MIET SMIEEE, Hong Kong
Dr. Abu sahman Bin mohd Supa'at, Universiti Teknologi Malaysia, Malaysia
Prof. Amit Kumar Garg, Deenbandhu Chhotu Ram University of Science & Technology, India
Dr. Jens Myrup Pedersen, Aalborg University, Denmark
Dr. Augustine Ikechi Ukaegbu, KAIST, Korea
Dr. Jamshid Sangirov, KAIST, Korea
Prof. Ahmed Dooguy KORA, Ecole Sup. Multinationale des Telecommunications, Senegal
Dr. Se-Jin Oh, Korea Astronomy & Space Science Institute, Korea
Dr. Rajendra Prasad Mahajan, RGPV Bhopal, India
Dr. Woo-Jin Byun, ETRI, Korea
Dr. Mohammed M. Kadhum, School of Computing, Goodwin Hall, Queen's University , Canada
Prof. Seong Gon Choi, Chungbuk National University, Korea
Prof. Yao-Chung Chang, National Taitung University, Taiwan
Dr. Abdallah Handoura, Engineering school of Gabes - Tunisia, Tunisia
Dr. Gopal Chandra Manna, BSNL, India

Dr. Il Kwon Cho, National Information Society Agency, Korea
Prof. Jiann-Liang Chen, National Taiwan University of Science and Technology, Taiwan
Prof. Ruay-Shiung Chang, National Dong Hwa University, Taiwan
Dr. Vasaka Visoottiviseth, Mahidol University, Thailand
Prof. Dae-Ki Kang, Dongseo University, Korea
Dr. Yong-Sik Choi, Research Institute, IDLE co., ltd, Korea
Dr. Xuena Peng, Northeastern University, China
Dr. Ming-Shen Jian, National Formosa University, Taiwan
Dr. Soobin Lee, KAIST Institute for IT Convergence, Korea
Prof. Yongpan Liu, Tsinghua University, China
Prof. Chih-Lin HU, National Central University, Taiwan
Prof. Chen-Shie Ho, Oriental Institute of Technology, Taiwan
Dr. Hyoung-Jun Kim, ETRI, Korea
Prof. Bernard Cousin, IRISA/Universite de Rennes 1, France
Prof. Eun-young Lee, Dongduk Woman s University, Korea
Dr. Porkumaran K, NGP institute of technology India, India
Dr. Feng CHENG, Hasso Plattner Institute at University of Potsdam, Germany
Prof. El-Sayed M. El-Alfy, King Fahd University of Petroleum and Minerals, Saudi Arabia
Prof. Lin You, Hangzhou Dianzi Univ, China
Mr. Nicolai Kuntze, Fraunhofer Institute for Secure Information Technology, Germany
Dr. Min-Hong Yun, ETRI, Korea
Dr. Seong Joon Lee, Korea Electrotechnology Research Institute, korea
Dr. Kwihoon Kim, ETRI, Korea
Dr. Jin Woo HONG, Electronics and Telecommunications Research Inst., Korea
Dr. Heeseok Choi, KISTI(Korea Institute of Science and Technology Information), korea
Dr. Somkiat Kitjongthawonkul, Australian Catholic University, St Patrick's Campus, Australia
Dr. Dae Won Kim, ETRI, Korea
Dr. Ho-Jin CHOI, KAIST(Univ), Korea
Dr. Su-Cheng HAW, Multimedia University, Faculty of Information Technology, Malaysia
Dr. Myoung-Jin Kim, Soongsil University, Korea
Dr. Gyu Myoung Lee, Institut Mines-Telecom, Telecom SudParis, France
Dr. Dongkyun Kim, KISTI(Korea Institute of Science and Technology Information), Korea
Prof. Yoonhee Kim, Sookmyung Women s University, Korea
Prof. Li-Der Chou, National Central University, Taiwan
Prof. Young Woong Ko, Hallym University, Korea
Prof. Dimiter G. Velev, UNWE(University of National and World Economy), Bulgaria
Dr. Tadasuke Minagawa, Meiji University, Japan
Prof. Jun-Kyun Choi, KAIST (Univ.), Korea
Dr. Brownson ObaridoaObele, Hyundai Mobis Multimedia R&D Lab , Korea
Prof. Anisha Lal, VIT university, India
Dr. kyeong kang, University of technology sydney, faculty of engineering and IT , Australia
Prof. Chwen-Yea Lin, Tatung Institute of Commerce and Technology, Taiwan
Dr. Ting Peng, Chang'an University, China
Prof. ChaeSoo Kim, Donga University in Korea, Korea
Prof. kirankumar M. joshi, m.s.uni.of baroda, India
Dr. Chin-Feng Lin, National Taiwan Ocean University, Taiwan
Dr. Chang-shin Chung, TTA(Telecommunications Technology Association), Korea

Editor Guide

■ Introduction for Editor or Reviewer

All the editor group members are to be assigned as a evaluator(editor or reviewer) to submitted journal papers at the discretion of the Editor-in-Chief. It will be informed by eMail with a Member Login ID and Password.

Once logined the Website via the Member Login menu in left as a evaluator, you can find out the paper assigned to you. You can evaluate it there. All the results of the evaluation are supposed to be shown in the Author Homepage in the real time manner. You can also enter the Author Homepage assigned to you by the Paper ID and the author's eMail address shown in your Evaluation Webpage. In the Author Homepage, you can communicate each other efficiently under the peer review policy. Please don't miss it!

All the editor group members are supposed to be candidates of a part of the editorial board, depending on their contribution which comes from history of ICACT TACT as an active evaluator. Because the main contribution comes from sincere paper reviewing role.

■ Role of the Editor

The editor's primary responsibilities are to conduct the peer review process, and check the final camera-ready manuscripts for any technical, grammatical or typographical errors.

As a member of the editorial board of the publication, the editor is responsible for ensuring that the publication maintains the highest quality while adhering to the publication policies and procedures of the ICACT TACT(Transactions on the Advanced Communications Technology).

For each paper that the editor-in-chief gets assigned, the Secretariat of ICACT Journal will send the editor an eMail requesting the review process of the paper.

The editor is responsible to make a decision on an "accept", "reject", or "revision" to the Editor-in-Chief via the Evaluation Webpage that can be shown in the Author Homepage also.

■ Deadlines for Regular Review

Editor-in-Chief will assign a evalaution group(a Editor and 2 reviewers) in a week upon receiving a completed Journal paper submission. Evaluators are given 2 weeks to review the paper. Editors are given a week to submit a recommendation to the Editor-in-Chief via the evaluation Webpage, once all or enough of the reviews have come in. In revision case, authors have a maximum of a month to submit their revised manuscripts. The deadlines for the regular review process are as follows:

Evalution Procedure	Deadline
Selection of Evaluation Group	1 week
Review processing	2 weeks
Editor's recommendation	1 week
Final Decision Noticing	1 week

■ Making Decisions on Manuscript

Editor will make a decision on the disposition of the manuscript, based on remarks of the reviewers. The editor's recommendation must be well justified and explained in detail. In cases where the revision is requested, these should be clearly indicated and explained. The editor must then promptly convey this decision to the author. The author may contact the editor if instructions regarding amendments to the manuscript are unclear. All these actions could be done via the evaluation system in this Website. The guidelines of decisions for publication are as follows:

Decision	Description
Accept	An accept decision means that an editor is accepting the paper with no further modifications. The paper will not be seen again by the editor or by the reviewers.
Reject	The manuscript is not suitable for the ICACT TACT publication.
Revision	The paper is conditionally accepted with some requirements. A revision means that the paper should go back to the original reviewers for a second round of reviews. We strongly discourage editors from making a decision based on their own review of the manuscript if a revision had been previously required.

■ Role of the Reviewer

Reviewer Webpage:

Once logined the Member Login menu in left, you can find out papers assigned to you. You can also login the Author Homepage assigned to you with the paper ID and author's eMail address. In there you can communicate each other via a Communication Channel Box.

Quick Review Required:

You are given 2 weeks for the first round of review and 1 week for the second round of review. You must agree that time is so important for the rapidly changing IT technologies and applications trend. Please respect the deadline. Authors undoubtedly appreciate your quick review.

Anonymity:

Do not identify yourself or your organization within the review text.

Review:

Reviewer will perform the paper review based on the main criteria provided below. Please provide detailed public comments for each criterion, also available to the author.

- How this manuscript advances this field of research and/or contributes something new to the literature?
- Relevance of this manuscript to the readers of TACT?
- Is the manuscript technically sound?
- Is the paper clearly written and well organized?
- Are all figures and tables appropriately provided and are their resolution good quality?
- Does the introduction state the objectives of the manuscript encouraging the reader to read on?
- Are the references relevant and complete?

Supply missing references:

Please supply any information that you think will be useful to the author in revision for enhancing quality of the paperor for convincing him/her of the mistakes.

Review Comments:

If you find any already known results related to the manuscript, please give references to earlier papers which contain these or similar results. If the reasoning is incorrect or ambiguous, please indicate specifically where and why. If you would like to suggest that the paper be rewritten, give specific suggestions regarding which parts of the paper should be deleted, added or modified, and please indicate how.

Journal Procedure

Dear Author,

- **You can see all your paper information & progress.**

- **Step 1. Journal Full Paper Submission**

Using the Submit button, submit your journal paper through ICACT Website, then you will get new paper ID of your journal, and send your journal Paper ID to the Secretariat@icact.org for the review and editorial processing. Once you got your Journal paper ID, never submit again! Journal Paper/CRF Template

- **Step 2. Full Paper Review**

Using the evaluation system in the ICACT Website, the editor, reviewer and author can communicate each other for the good quality publication. It may take about 1 month.

- **Step 3. Acceptance Notification**

It officially informs acceptance, revision, or reject of submitted full paper after the full paper review process.

Status	Action
Acceptance	Go to next Step.
Revision	Re-submit Full Paper within 1 month after Revision Notification.
Reject	Drop everything.

- **Step 4. Payment Registration**

So far it's free of charge in case of the journal promotion paper from the registered ICACT conference paper! But you have to regist it, because you need your Journal Paper Registration ID for submission of the final CRF manuscripts in the next step's process. Once you get your Registration ID, send it to Secretariat@icact.org for further process.

- **Step 5. Camera Ready Form (CRF) Manuscripts Submission**

After you have received the confirmation notice from secretariat of ICACT, and then you are allowed to submit the final CRF manuscripts in PDF file form, the full paper and the Copyright Transfer Agreement. Journal Paper Template, Copyright Form Template, BioAbstract Template,

Journal Submission Guide

All the Out-Standing ICACT conference papers have been invited to this "ICACT Transactions on the Advanced Communications Technology" Journal, and also welcome all the authors whose conference paper has been accepted by the ICACT Technical Program Committee, if you could extend new contents at least 30% more than pure content of your conference paper. Journal paper must be followed to ensure full compliance with the IEEE Journal Template Form attached on this page.

➢ How to submit your Journal paper and check the progress?

Step 1. Submit	Using the Submit button, submit your journal paper through ICACT Website, then you will get new paper ID of your journal, and send your journal Paper ID to the Secretariat@icact.org for the review and editorial processing. Once you got your Journal paper ID, never submit again! Using the Update button, you can change any information of journal paper related or upload new full journal paper.
Step 2. Confirm	Secretariat is supposed to confirm all the necessary conditions of your journal paper to make it ready to review. In case of promotion from the conference paper to Journal paper, send us all the .DOC(or Latex) files of your ICACT conference paper and journal paper to evaluate the difference of the pure contents in between at least 30% more to avoid the self replication violation under scrutiny. The pure content does not include any reference list, acknowledgement, Appendix and author biography information.
Step 3. Review	Upon completing the confirmation, it gets started the review process thru the Editor & Reviewer Guideline. Whenever you visit the Author Homepage, you can check the progress status of your paper there from start to end like this, " Confirm OK! -> Gets started the review process -> ...", in the Review Status column. Please don't miss it!

Volume. 5 Issue. 4

An Adaptive Compression Algorithm for Energy-Efficient Wireless Sensor Networks

Beihua Ying*

* *School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, Ningbo, Zhejiang, 315100, China*

yingbh@nit.zju.edu.cn

***Abstract*—Energy efficiency is one of the most important design metrics for wireless sensor networks. As sensor data always have redundancies, compression is introduces for energy savings. However, different emphases on algorithm design influence the operation effect of compression under various applications and network environments. In order to improve the energy utilization efficiency for the whole network, an adaptive data compression is proposed in this paper, which realizes a real-time adjustment of compression strategy. By prediction and feature extraction of several relevant parameters, the algorithm provides optimal execution strategies for each sensor node in the network. The simulation results show that, the proposed compression scheme enables all nodes to complete data communication with near optimal energy consumptions, and the maximum deviation against the ideal condition is no more than 5%. Moreover, the algorithm can effectively act on different data precision, transmit power and retransmission rate to meet the dynamic requirements of the network with only a few costs introduced.**

***Keyword*—wireless sensor networks, data compression, energy efficiency, adaptive mechanism**

I. INTRODUCTION

WIRELESS sensor network (WSN) is an emerging and promising networking technique that has attracted more and more attention in recent years. It facilitates humans to sense and monitor the region of interest, and is widely used in many application fields such as agricultural planting, medical care, smart homes, ecological monitoring and so on [1]–[3].

Since the power of sensor nodes supplied by batteries is high-limited and not easy to complement commonly, the most important issue in WSNs is prolonging network lifetime by energy-efficient strategy. Data compression is introduced into WSNs due to its ability to reduce the data amount by exploiting the redundancy resided in sensing data [4]. Meeting the application requirement as a prerequisite condition, such kind of in-network information processing is strongly recommended to decrease energy consumptions in data communication and prolong the lifetime of WSNs.

However, from the implementation perspective and reevaluation, data compression may not achieve total energy savings in any case [5]. On one hand, sensor nodes are densely deployed which permits a low transmit power during data collection. On the other hand, if high-resolution data are required by applications, it is hard to get a satisfied compression ratio. Thus, in-network data compression may increase rather than decrease the total energy consumptions when the savings in communication cannot compensate the additional costs in data processing. Thus, a compression arbitration system has been proposed in [6] by which compression algorithm in time domain is examined carefully to assess its energy efficiency before compression. The case in which compression is unnecessary will be avoided and sensor node will transmit raw data directly instead of the compressed ones.

Based on our previous works, it is reasonable to believe that data compression in WSNs should be carefully carried out. The variety of application scenarios will affect its efficiency profoundly. In order to obtain more energy savings for whole networks, more researches need to be done on data compression. Among them, the adaptability of algorithm is one of the most important design considerations, because it is an effective way to deal with the variability of different situations. In this paper, we pay greater attention to this adaptability and propose an adaptive algorithm that enables a real-time adjustment of compression strategy to increase the energy efficiency of the whole network.

The remainder of this paper is structured as follows: in Section 2, we discuss the related work on the adaptability of compression algorithms and the motivation for our adaptive data compression. The proposed system and its mathematical analysis are presented in Section 3. The process of building prediction models is detailed described in Section 4, and subsequently, Section 5 presents the results of algorithm simulation and implementation using an environmental application. Finally, Section 6 concludes the paper.

II. RELATED WORK AND MOTIVATION

Most existing works enhance the adaptability of compression algorithm in WSNs by using tunable data processing methods. In these researches, data precision is

Manuscript received December 31, 2015. This work was sponsored by the Nature Science Foundation of Zhejiang Province (Project No. LQ13F010005) and Ningbo Natural Science Foundation (Project No. 2011A610185). The research was supported by Ningbo Sc. & Tech. plan project (2014B82015). The author would like to appreciate the valuable datasets from the TAO Project Office of NOAA/PMEL, the ShakeMap Working Group, and the NWISWeb Support Team.

Beihua Ying is with the School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, Ningbo, Zhejiang, 315100, China (corresponding author to provide phone: 86-574-88130020; fax: 86-574-88229505; e-mail: yingbh@ nit.zju.edu.cn).

restricted as an error tolerance, which is specified by each application. The compression can be lossless or lossy for different needs, such as wavelet transformation [7], predictive compression [8], data fitting [9], discrete cosine transform (DCT) [10], compressive sensing [11]–[12], fuzzy transform [13], and so on. Moreover, some lossless compression methods are improved to adapt to WSN applications. For example, an adaptive Huffman code is proposed in [14] that can achieve tunable compression with required accuracy.

Another meaningful work is about the energy consumption in compression that has received more and more attentions instead of compression ratio. Different from other applications, energy saving is the ultimate design purpose of WSNs in most cases. As a result, compression cannot merely pursue high compression ratio, but take energy costs into full consideration [15]–[16]. It also reflects the adapt degree to WSN applications. Several studies have begun to focus on this. In order to reduce the total energy consumptions, a trade-off between computation during data compression and communication energy is made carefully. [15] constructs several energy models of computation and communication for mathematical analysis, but gzip is adopted which cannot be implemented in the resource constrained sensor nodes. In [17], error radius of prediction algorithm are tuned for optimizing the desired tradeoffs between data quality and energy saved, and a similar work is proposed in [18] based on the discrete cosine transform.

Once the energy consumption of compression is considered, the practicality of several algorithms will be greatly reduced, especially for the one with high computational complexity. What is more, compression itself cannot get energy saving in some situations. According to this point, we introduce a novel pre-judgment mechanism to data compression. Once system estimates that compression cannot save energy, sensor node will send raw data directly. A similar work is presented in [19], which decides whether compressing or not based on time delay.

As shown in our previous works, using "compression pre-judgment" can remarkably make sensor nodes more energy efficiency in most cases. It also makes our compression system more suitable for WSN applications [20]. Nevertheless, this system is still insufficient in two aspects. Firstly, the compression arbitration in [20] focuses on a single node, but not a whole network. Decisions about whether or not to compress data are made at the node-level, which are not involved in the ubiquitous multi-hop communication mode. As a result, data receiving is neglected when the total energy costs are calculated and subsequently compared. Since data reduction not only affects the energy dissipation in sending, but also in receiving, it is not comprehensive to estimate the energy-saving benefits of compression algorithms without data reception at the network-level.

More importantly, the compression arbitration is proposed under the assumption that only one compression algorithm is adopted in the whole network, which would be unable to make the best use of data compression. According to the evaluation results of compression algorithms presented in [5], different algorithms show different properties though they are all aimed to be used in WSNs. Take the existing algorithms for example: LAA [21] and predictive coding [22] represent one kind of algorithms that have low complexity, whereas their compression effects are not prominent. Conversely, PMC-MR [23] and LTC [9] tend to achieve better compression ratio at the costs of more energy losses in calculation. It is clear that, in those algorithms, compression effect and algorithm complexity are two major considerations, but have different emphasis during the design. This is probably the main reason causing the differences between compression algorithms. As a result, when various settings of applications are selected such as data type, precision requirement, communication quality, node location, and so on, energy-saving benefits obtained by compression algorithms are distinct.

In summary, from the point of view of the entire network, it is reasonable to believe that single compression algorithm cannot achieve better energy efficiency. An adaptive data compression with tunable precision is urgently needed which is able to determine the optimal execution strategy according to different application requirements and network settings. Benefit from the adaptability, compression algorithms with distinct characteristics will be adopted to maximize energy savings at the network-level.

III. Adaptive Compression System

A. System Description

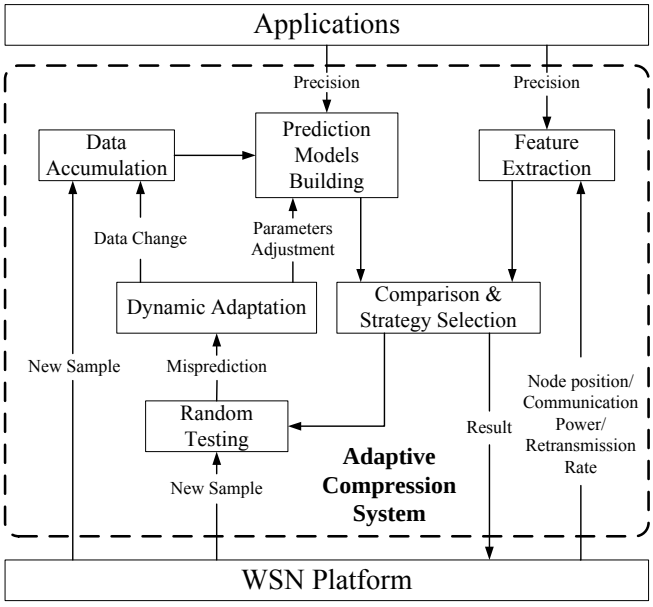


Fig. 1. Framework of the adaptive compression system

In order to make the best use of different compression algorithms for the total energy savings, a lightweight strategy selection mechanism is proposed in this paper. Before compressing, an optimal executive strategy for each sensor node is derived by predicting and feature extracting of all related parameters, which makes the whole network achieve energy efficiency to the greatest degree. The adaptive compression system framework is shown in Fig. 1. According to the function of each module, the whole procedure can be divided into four steps:

1. Prediction modeling

Before deciding on the optimal strategy, two models are established to predict the compression ratio and compression time on-line. Data compression under various algorithms is performed during this initial stage. Compression ratio and execution time for each algorithm based on different datasets and application requirements are recorded. These data are the basis for the prediction model building. Since this is done on-line, only a few samples are used to save energy.

2. Feature extraction

When starting the decision-making process, three types of feature need to be extracted, including raw data, algorithm and network. Among them, data type, error tolerance and algorithm type, which represent the features of raw data and algorithm respectively, are used to calculate the compression ratio and the required time. Moreover, the useful information extracted from the network include node position, communication power and retransmission rate.

3. Compression evaluation & strategy selection

Once the required parameters are ready, system can evaluate all kinds of alternative strategies by calculating the total energy consumptions including data compression and data transmission from each source node to sink. The optimal executive strategy can be obtained based on the comparison results. The selected strategy leading to the lowest energy loss may be an alternative compression algorithm, or may not perform any compression.

4. Model modification

Since compression ratio and execution time will greatly influence the energy consumptions in communication and calculation respectively, ensuring the predict precision can effectively improve the accuracy of the strategy selection. In view of this, model modification is used to guarantee the accuracy of the two predictions. Raw datasets are randomly selected to verify the predicted results. The model parameters will be modified if the prediction accuracy is not satisfied.

B. Mathematical Analysis

In our system, the optimal strategy choosing is based on the total energy costs of each sensor node in data communication. Correct energy calculation is very important, because it will directly affect the judgment of the strategy. Thus, the energy losses in both communication and calculation are considered here.

$$E_{comp} = P_{MCU} * L * T_{MCU}(e) + P_{TX}(d) * L * CR(e) * T_{tran} * (1+\gamma), h = 1 \quad (1)$$

$$E_{comp} = P_{MCU} * L * T_{MCU}(e) + \sum_{i=1}^{h} P_{TX}(d_i) * L * CR(e) * T_{tran} * (1+\gamma_i) + \sum_{i=1}^{h-1} P_{RX} * L * CR(e) * T_{tran} * (1+\gamma_i), h > 1 \quad (2)$$

$$E_{uncomp} = P_{TX}(d) * L * T_{tran} * (1+\gamma), h = 1 \quad (3)$$

$$E_{uncomp} = \sum_{i=1}^{h} P_{TX}(d_i) * L * T_{tran} * (1+\gamma_i) + \sum_{i=1}^{h-1} P_{RX} * L * T_{tran} * (1+\gamma_i), h > 1 \quad (4)$$

The proposed system is analyzed where E_{uncomp} denotes the energy consumption for transmitting the raw data directly and E_{comp} denotes the total energy costs for compressing the same data and then transmitting the compressed one. It is clear that E_{uncomp} only involve the communication costs, whereas E_{comp} is related to the compression costs in MCU and the communication costs in RF module. Considering that the cost in RF wake-up is the same, regardless of whether compression is executed, it will not affect the final decision results. On the other hand, since the length of the frame head is much smaller than the data part, it also can be ignored. As a result, E_{comp} and E_{uncomp} can be simplified as (1) to (4).

L denotes the original data size and CR describes the compression ratio which is a function of the error bound e. T_{tran} is the time cost for transmitting one byte of data and T_{MCU} is the time overhead for compressing one byte, which is strongly dependent on e. The transmit power P_{TX} is closely related to the communication distance d, while the MCU power P_{MCU} and the received power P_{RX} are approximately constant when the related modules work in the active mode. It is clear that CR (e) and T_{MCU} (e) are obtained by the two prediction models. As the sink is a super node with unlimited energy, there is no need to consider receiving energy consumption for its neighbors whose hop count h is equal to 1. The data retransmission rate γ_i reflects the quality of communication channel in the multi-hop routing. The worse the communication channel becomes, the higher the data retransmission rate is.

(2) shows that for different compression methods, the characteristics of each algorithm determine the energy saving benefits of sensor nodes. For the compression with low complexity, T_{MCU} will be very small, which leads to a lower computational energy costs. If sensor nodes are close to the sink (h is small), computational costs will account for a large proportion, especially when P_{TX} is turned down for the short-range communication. In this regard, algorithm with low complexity will bring a huge advantage. With the increase of h, the proportion of computational costs will be decreased, while the corresponding communication costs is obviously increased. At this point, the influence of compression ratio on communication energy consumptions will directly act on the total energy costs. Therefore, compression algorithm, which focuses on the compression effect gradually, shows its energy saving advantages with the increase of node hops.

IV. Adaptive Mechanism in Prediction Modeling

Accurate prediction of compression ratio and execution time is an important part of ensuring the correct selection of strategy. Thus, an adaptive mechanism is introduced into our proposed system. This mechanism requires fewer samples to build the prediction models, which makes on-line modeling possible. Even if the initial models are somewhat inaccurate, they will be adjusted adaptively to the best results. The adaptive mechanism is illustrated in Table I.

The beginning of the adaptive mechanism uses an initial sample step with a given range. Once a new sample is ready for verification, the compression ratio and execution time are both measured. The comparison then determines whether the difference between the predicted value and the real one exceeds a preset error bound. If the prediction error is large, the information including the compression ratio and time overhead are recorded for the new model. Two prediction

models are rebuilt when the sample step (*step*) reaches the minimum value or the sample step is reduced continuously. Alternatively, the sample step is increased until the maximum value.

TABLE I
PROCESS OF THE ADAPTIVE MECHANISM

Algorithm Adaptive mechanism for prediction modeling

```
1:  set an initial sample step (step)
2:  set an allowable step range [step_min, step_max]
3:  if (sample is awaiting verification) then
4:    measure CR and T_MCU
5:    if ((|prediction error of CR|>=error bound)
          or (|prediction error of T_MCU |>=error bound)) then
6:      record compression ratio and time overhead for the sample
7:      if (step == step_min) then
8:        rebuild the models with the recorded samples
9:        reset step
10:     else
11:       decrease step
12:     end if
13:   else
14:     increase step until step_max
15:   end if
16: end if
```

V. PERFORMANCE EVALUATION

A. Experimental setting

In order to verify the energy saving effect of the proposed system, the experimental settings are confirmed, such as original datasets, network topology, alternative strategies and experimental platform.

1. Original datasets

Datasets from the Tropical Atmosphere Ocean Project (TAO) [24], which collects real-time oceanographic and meteorological data in the Pacific Ocean, are selected for the experiments. Among the various types of datasets, air temperature, sea level pressure and relative humidity are chosen because of their very different data characteristics.

2. Network topology

Grid-based network topology is adopted with nodes even distribution, as show in Fig. 2.

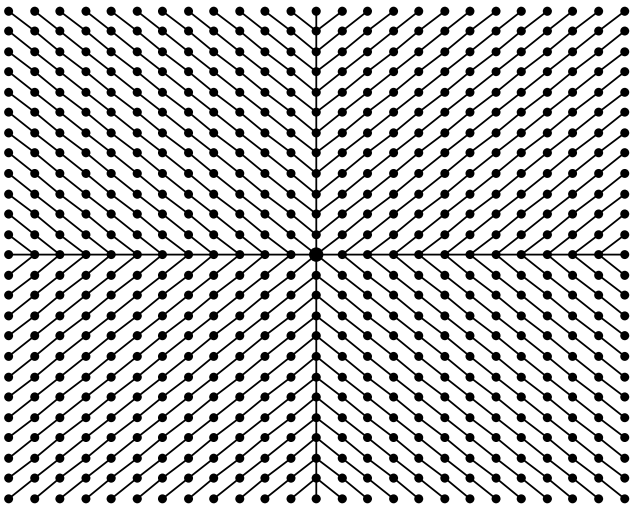


Fig. 2. Network topology in the test

The network size is 25*25, and the total number of sensor nodes is 625. In these nodes, sink node is located in the center of the whole network, which is denoted by a large solid circle. The rest of the nodes are homogeneous. Each source node is routed to the sink using SPT (Shortest Path Tree) [25]. Because of the even distribution, sensor nodes are assumed to transmit data with a uniform RF power level and retransmission rate.

3. Alternative strategies

Based on the evaluation results of data compression in [5], the tests select four different algorithms with better performances, as shown in Table II. Combined with the implementation of not compressed, sensor nodes can choose the best on from these five alternative strategies.

TABLE II
COMPRESSION ALGORITHM IN THE TEST

Algorithm	Type	Remark
Single Moving Average	Predictive compression	N=3
LAA	Linear regression	——
PMC-MR	Linear regression	——
LTC	Linear regression	——

4. Experimental platform

We choose MicaZ nodes as the test platform for our experiments. They are commonly used in WSNs. The processor is an 8-bit Atmel ATmega128L microcontroller, and the processor speed is fixed at 8MHz. As the results shown in [26], supply current of processor is nearly constant in active mode. Therefore, we consider P_{MCU} as a fixed value in the test. T_{MCU} is obtained by ATMEL AVR Studio [27]. In MicaZ node, a CC2420 unit is responsible for communicating with other nodes. It is a single-chip RF transceiver that operates at 2.4 GHz. According to [28], the data transmission rate of a MicaZ node is up to 250 kbps. Besides, transmit power is configurable; in that case, CC2420 can be powered down by setting control register when communication distance is short. In the test, we assume that source nodes send information up to 100 m when the transmit power level is set to 31. Experimental parameters mentioned above are listed in Table III.

TABLE III
VALUES OF THE EXPERIMENTAL PARAMETERS

Symbol	Value	Unit	Remark
d	5-100	m	Outdoor monitoring
P_{MCU}	26.4	mW	8mA current draw
P_{TX}	57.42	mW	PA_Level=31
P_{RX}	62.04	mW	18.8mA current draw
T_{tran}	32	μs	250kbps data rate

5. Relevant assumption

Several reasonable hypotheses are given in our experiments. First, when source nodes, which are homogeneous, establish a communication route with the sink using SPT, they are able to get the communication hops h and the transmit power P_{TX}. Second, accuracy requirement of the data is given by WSN applications, and this information is sent to the whole network by the sink. Third, the test network is uniform distribution, so the node density determines the RF

transmit power required by the single hop communication. With the failure of the nodes, the density will be reduced, and the RF power and data retransmission rate will be increased accordingly.

B. Impact of the characteristics in data and network

In order to explore the impact of the characteristics in original data and network on energy saving strategy selection, we choose data accuracy, transmit power level and data retransmission rate as test variables for the deeply analysis.

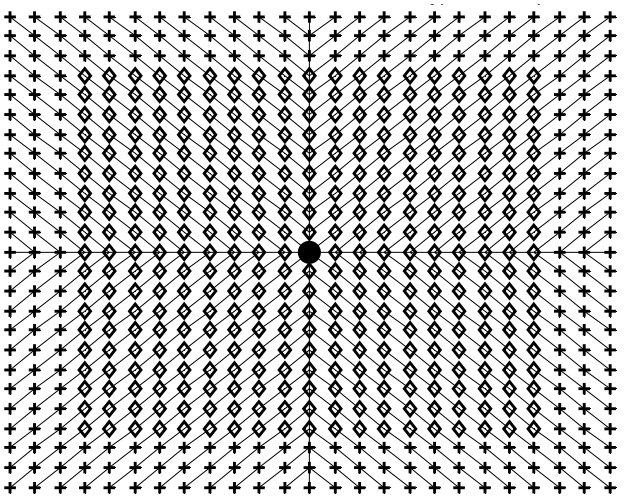
Taking air temperature as an example, Fig. 3 shows the results of selecting the best compression implementation strategy by our adaptive compression method, which are set by the different data accuracy requirements, the transmit power level and the data retransmission rate. Among them, taking into account the actual situations of WSN applications, the variation grades of data accuracy, transmission power and retransmission rate are set to 1-12, 3-31, 10%-150%, respectively.

Comparing Fig. 3(a) and Fig. 3(b) reflects the impact of the data accuracy on strategy selection. It can be seen that there are three alternative strategies in the results: PMC-MR, LTC and no compression. When the accuracy level is set higher (the numerical value is small), source nodes near the sink are more inclined to choose no compression, which is in good agreement with the conclusion that compression may not achieve total energy savings in any case. At that time, due to the compression algorithms cannot achieve satisfactory compression effect, the energy savings in communication may not compensate for costs in calculation. Therefore, from the point of view of total energy costs in nodes, transmitting raw data without compression will obtain lower energy losses. With the increase of the distance between source nodes and the sink, the benefit of compression is also increasing. As a result, the nodes tend to select the algorithm (LTC in Fig. 3) which has a better compression effect.

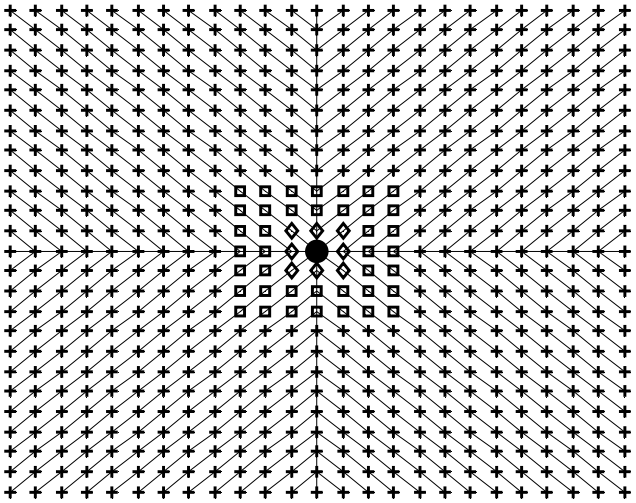
Reducing data accuracy requirements, the effect obtained by data compression is gradually clear. Sensor nodes mostly choose to compress at first, and send the compressed data instead of the original ones. Furthermore, during the selection of different algorithms, nodes near the sink will make a trade-off between computational complexity and compression effect, whereas nodes away from the sink will still consider the compression effect as a primary goal. Therefore, from the results of Fig. 3(b), although LTC can achieve the minimum compression ratio among the four compression algorithms, source nodes near the sink still select PMC-MR as the best energy saving strategy in virtue of its high computational costs.

Comparing Fig. 3(a) and Fig. 3(c) reflects the impact of the transmission power level on strategy selection. The lower the transmission power level is, the higher the probability that compression wastes the total energy costs. As a result, there are over 50% of the nodes do not compress the raw data in Fig. 3(a). With the increase of the transmission power, the advantage of compression in energy savings becomes more obvious. Thus, more nodes choose to perform the compression operation, as shown in Fig. 3(c).

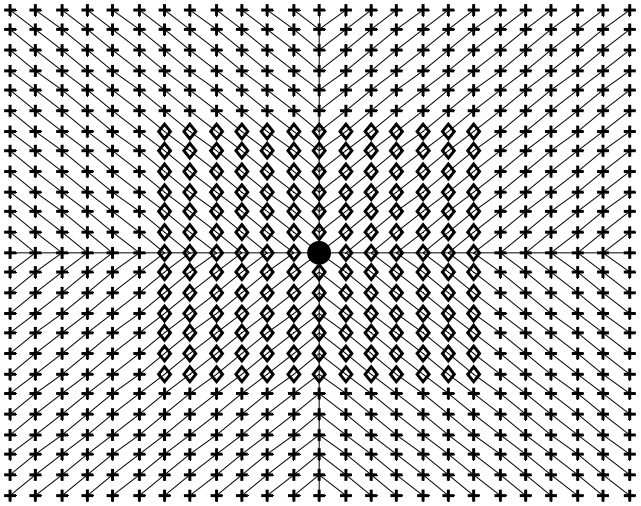
The impact of the retransmission rate on strategy selection can be seen in Fig. 3(b) and Fig. 3(d). The lower the data retransmission rate, the more likely that node that is close to the sink selects no compression. With the deterioration of channel quality, retransmission rate and communication energy costs increase correspondingly, and nodes will gradually tend to select the algorithm which has good compression effect.



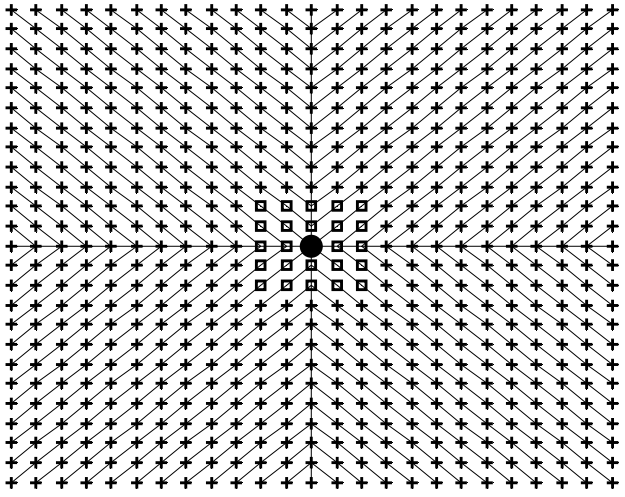
(a)



(b)



(c)



(d)

□: PMC-MR +: LTC ◊: No compression

Fig. 3. Best compression strategy in different situations: (a) data accuracy: level 1; transmission power: level 3; retransmission rate: 10%; (b) data accuracy: level 12; transmission power: level 3; retransmission rate: 10%; (c) data accuracy: level 1; transmission power: level 23; retransmission rate: 10%; (d) data accuracy: level 1; transmission power: level 3; retransmission rate: 100%.

C. Energy-saving effect of the adaptive compression

In order to evaluate the energy-saving effect of the proposed compression algorithm quantitatively, the total energy consumptions of the sensor nodes are measured in the following two cases. One is that all nodes use a single data compression algorithm or do not perform any compression, and the other is that each node adopts the adaptive compression to choose the best one from the five alternative strategies.

In fact, if the nodes use the optimal strategy in each data transfer, it can be considered as the best situation of energy costs. Taking this ideal case as a reference, the statistical results of the total energy costs in different cases are compared in the form of maximum deviation (expressed as a percentage), as shown in Table IV. In this table, E, RF and γ represent the level of data accuracy, transmission power and retransmission rate, respectively. That is to say, E2_RF7_γ0.1 denotes that the data accuracy and the transmission power are in level 2 and level 7, and the retransmission rate is 10%.

From the results of Table IV, it is clear that the proposed compression method can make the whole sensor nodes complete data transfer tasks with nearly optimal energy consumptions, no matter how the precision requirement, transmit power and retransmission rate change. It should mainly owe to the adaptive mechanism introduced into the data compression that uses strategy selection for energy efficiency. When the accuracy requirement is high, like E2, data compression cannot obtain a satisfied compression ratio. Instead of saving energy costs, compression increases the additional computational losses. Thus, the total energy consumption has a greater deviation from the optimal once complex compression is executed. At that time, no compression or compression algorithm with lower complexity will make the energy consumption approximate to the optimal. With the decrease of the accuracy requirement, the compression method gradually reveals the advantages of energy savings. More and more sensor nodes adopt compression methods, especially the one with good compression effect and low implementation complexity, and no compression gradually increases the degree of deviation from the ideal situation.

Due to the limitation of prediction accuracy in compression ratio and execution time, erroneous judgment will inevitably occur during the process of compression evaluation and the following strategy selection. It also probably leads to that a single compression algorithm is more close to the optimal energy consumption than the proposed one in some cases, for example E10_RF7_γ0.1. But in general, the adaptive compression method can provide a relatively accurate energy efficient strategy, so that sensor nodes can complete data transmission under an approximate optimal energy costs, and the highest degree of deviation is no more than 5%.

Certainly, the introduction of adaptive mechanism will also bring two aspects of the costs, namely the overhead of computation and storage. By simulation, the time cost of executing the proposed method once is about 0.4ms (about 3000 clock cycles), which is approximately equal to the time overhead of using the LAA algorithm to compress 50 bytes

TABLE IV
MAXIMUM DEVIATION OF THE TOTAL ENERGY COSTS IN DIFFERENT SITUATIONS

Parameters	Forecast	LAA	PMC-MR	LTC	No comp.	Adaptive comp.
E2_RF7_γ0.1	59.84%	31.99%	36.17%	43.17%	12.99%	4.77%
E2_RF7_γ1.0	32.07%	17.66%	15.69%	19.32%	13.89%	3.07%
E2_RF15_γ0.1	46.78%	25.17%	26.51%	31.94%	13.17%	4.00%
E2_RF15_γ1.0	25.69%	14.64%	11.08%	13.86%	13.99%	3.08%
E2_RF23_γ0.1	38.07%	20.66%	20.07%	24.45%	13.32%	3.54%
E2_RF23_γ1.0	21.57%	13.68%	8.15%	10.36%	14.07%	3.08%
E5_RF7_γ0.1	52.98%	40.97%	15.70%	32.13%	48.00%	4.96%
E5_RF7_γ1.0	34.34%	41.80%	5.89%	12.47%	49.56%	3.83%
E5_RF15_γ0.1	43.44%	41.13%	9.29%	22.33%	48.30%	3.87%
E5_RF15_γ1.0	32.76%	41.89%	5.86%	9.84%	49.73%	3.76%
E5_RF23_γ0.1	37.61%	41.27%	5.69%	16.16%	48.56%	3.64%
E5_RF23_γ1.0	32.77%	41.97%	5.88%	8.41%	49.88%	3.78%
E10_RF7_γ0.1	64.93%	97.17%	3.37%	40.58%	133.31%	3.55%
E10_RF7_γ1.0	65.65%	99.70%	3.79%	24.36%	137.68%	3.79%
E10_RF15_γ0.1	65.04%	97.57%	3.43%	36.29%	134.01%	3.43%
E10_RF15_γ1.0	65.59%	99.50%	3.75%	26.82%	137.34%	3.75%
E10_RF23_γ0.1	65.14%	97.93%	3.49%	32.92%	134.62%	3.49%
E10_RF23_γ1.0	65.78%	99.85%	3.81%	24.44%	137.77%	3.81%

of raw data. Since the execution frequency of the method depends on the change frequency of error tolerance, transmit power and retransmit rate, it can be considered that the computational energy consumptions of the adaptive mechanism is acceptable.

On the other hand, the implementation of the adaptive mechanism is based on a set of alternative strategies. Therefore, it needs to embed all kinds of alternative strategies into sensor nodes. Through the realization of each compression algorithm, the storage overhead is obtained, as shown in Table V. Compared to the overhead of data acquisition and communication (about 12540 bytes), the storage cost of the adaptive mechanism is also acceptable.

TABLE V
STORAGE OVERHEAD OF COMPRESSION ALGORITHMS

Algorithm	Storage cost	Unit
Single Moving Average	902	byte
LAA	356	byte
PMC-MR	690	byte
LTC	2490	byte

VI. CONCLUSION

Different emphases on algorithm design will influence the energy efficiency of data compression under various applications and network environments. In order to raise the energy efficiency of data compression for the whole network, this paper presents an adaptive data compression with tunable precision, which enables a real-time adjustment of compression strategy.

By the prediction and feature extraction of the relevant parameters, the method can provide the optimal energy saving strategy for different sensor nodes. Experimental results show that, by means of the network level energy-saving improvement, all nodes can complete the data transfer task with near optimal energy consumptions, and the deviation degree is no more than 5%. Furthermore, the method only introduces low costs in computation and storage, and can effectively act on different error tolerance, transmit power and retransmission rate to meet the dynamic requirements of the network.

ACKNOWLEDGMENT

In this paper, the research is sponsored by the Nature Science Foundation of Zhejiang Province (Project No. LQ13F010005) and Ningbo Natural Science Foundation (Project No. 2011A610185). The work is supported by Ningbo Sc. & Tech. plan project (2014B82015). The author would like to appreciate the valuable datasets from the TAO Project Office of NOAA/PMEL, the ShakeMap Working Group, and the NWISWeb Support Team.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol.38, pp.393–422, 2002.

[2] K. Romer, F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp.54 – 61, 2004.

[3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks the International Journal of Computer & Telecommunications Networking,* vol. 52, no. 12, pp. 2292-2330, 2008.

[4] A. Mainwaring, D.Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the First Acm International Workshop on Wireless Sensor Networks & Applications*, Atlanta, Georgia, USA, 2002, pp. 88-97.

[5] B. Ying, Y. Liu, H. Yang, and H. Wang, "Evaluation of tunable data compression in energy-aware wireless sensor networks," *Sensors,* vol. 10, no. 4, pp. 3195-217, 2010.

[6] B. Ying, W. Liu, Y. Liu, H. Yang, and H. Wang, "Energy-efficient node-level compression arbitration for wireless sensor networks," in *Proceedings of the 11th International Conference on Advanced Communication Technology*, Phoenix Park, Korea, 2009, pp. 564-568.

[7] Z. J. Ming, L. Y. Ping, Z. S. Wang, "Haar wavelet data compression algorithm with error bound for wireless sensor networks," *Journal of Software,* vol. 21, no. 6, pp. 1364-1377.

[8] Y. Liang, "Efficient temporal compression in wireless sensor networks," in *IEEE the 36th Conference on Local Computer Networks (LCN*), Bonn, Germany, 2011, pp.466-474.

[9] T. Schoellhammer, E. Osterweil, B. Greenstein, M. Wimbrow, D. Estrin, "Lightweight Temporal Compression of Microclimate Datasets," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks,* Tampa, Florida, USA, 2004, pp.516-524.

[10] Y. I. Kefu, J. Wan, T. Bao, L. Yao, "A DCT regularized matrix completion algorithm for energy efficient data gathering in wireless sensor networks," *IEEE Communications Letter*, vol. 19, pp. 54-57, 2015.

[11] R. Hussein, A. Mohamed, M. Alghoniemy, "Adaptive compression and optimization for real-time energy-efficient wireless EEG monitoring systems," in *The 2013 Biomedical Engineering International Conference (BMEiCON-2013),* Krabi, Thailand, 2013, pp.1-5.

[12] X. Xing, D. Xie, G. Wang, "Data Gathering and Processing for Large-Scale Wireless Sensor Networks," in *IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks,* Dalian, China, 2013, pp.354-358.

[13] M. Abdelaal, O. Theel, "An Efficient and Adaptive Data Compression Technique for Energy Conservation in Wireless Sensor Networks," in *IEEE Conference on Wireless Sensor (ICWISE),* Kuching, Sarawak, Malaysia, 2013, pp.124-129.

[14] Y. Mo, Y. Qiu, J. Liu, Y. Ling, "A Data Compression Algorithm Based on Adaptive Huffman Code for Wireless Sensor Networks," in *The 4th International Conference on Intelligent Computation Technology and Automation,* Shenzhen, China, 2011, pp.3-6.

[15] N. Arastouie, M. Sabaei, H. S. Shahreza, "Adaptive Approach for Reducing Energy Consumption, Considering Nodes' Mission in Wireless Sensor Networks," in *The 2nd International Conference on Communication Software and Networks,* Singapore, 2010, pp.152 - 156.

[16] S. Eswaran, J. Edwards, A. Misra, T. Porta, "Adaptive In-Network Processing for Bandwidth and Energy Constrained Mission-Oriented Multihop Wireless Networks," *IEEE Transaction on Mobile Computing*, vol. 11, no. 9, pp. 1484-1498, 2012.

[17] M. I. Mohamed, W. Y. Wu, M. Moniri, "Adaptive data compression for energy harvesting wireless sensor nodes," in *The 10th IEEE International Conference on Networking, Sensing and Control (ICNSC),* Evry, France, 2013, pp.633-638.

[18] K. Hua, H. Wang, W. Wang, S. Wu, "Adaptive Data Compression in Wireless Body Sensor Networks," in *IEEE the 13th International Conference on Computational Science and Engineering,* Hong Kong, China, 2010, pp.1-5.

[19] X. Deng, Y. Yang, "Online adaptive compression in delay sensitive wireless sensor networks," *IEEE Transactions on Computer,* vol. 61, no. 10, pp.1429-1442, 2012.

[20] B. Ying, Y. Liu, H. Wang, "Improved adaptive compression arbitration system for wireless sensor networks," *Tsinghua Science and Technology,* vol. 15, no. 2, pp. 202-208, 2010.

[21] L. Ying, S.W. Loke, M.V. Ramakrishna, "Energy-saving data approximation for data and queries in sensor networks," in *The 6th International Conference on ITS Telecommunications,* Chengdu, China, 2006, pp. 782-785.

[22] C. Hui, L. Cui, "Forecast-based temporal data aggregation in wireless sensor networks," *Computer Engineering and Applications*, vol. 43, no. 21, pp. 121-125, 2007.

[23] I. Lazaridis, S. Mehrotra, "Capturing sensor-generated time series with quality guarantees," in *IEEE International Conference on Data Engineering*, Bangalore, India, 2003, pp. 429-440.

[24] Tropical Atmosphere Ocean Project [Online]. Available: http://www.pmel.noaa.gov/tao/index/shtml
[25] T. S. Moh, M. Dumont, T. S. Moh, "Evaluation of dynamic tree-based data gathering algorithms for wireless sensor networks," in *IEEE International Symposium on Signal Processing and Information Technology,* Athens, Greece, 2005, pp.170-175.
[26] M. Kramer, A. Geraldy, "Energy Measurements for Micaz Node," *Technical Report*, Technical University Kaisers Lautem, GI/ITG KuVS, 2006, pp. 1-7.
[27] ATMEL AVR Studio [Online]. Available: http://www.atmel.com/dyn/Products/tools_card.asp?tool_id=2725
[28] CC2420 Data Sheet [Online]. Available: http://inst.eecs.berkeley.edu/~cs150/Documents/CC2420.pdf

Beihua Ying received her Ph.D. degree from Tsinghua University, Beijing, China, in 2010 and the BS degree from Xidian University, Xi'an, China, in 2004, both in Electronic Engineering. She is currently a lecturer with the school of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University. Her research interests include design and analysis of energy-efficient data process for wireless sensor networks, data gathering and routing protocol in wireless ad hoc and sensor networks, and optimization problems.

Automatic Zooming Mechanism for Capturing Clear Moving Object Image Using High Definition Fixed Camera

Hsien-Chou Liao*, Po-Yueh Chen**, Zi-Jun Lin**, Zi-Yi Lim*

*Department of Computer Science and Information Engineering, Chaoyang University of Technology, 168, Jifeng E. Rd., Wufeng District, Taichung, 41349 Taiwan, R. O. C.

**Department of Computer Science and Information Engineering, National Changhua University of Education, No.1, Jinde Rd., Changhua City, Changhua County, 500 Taiwan, R. O. C.

seafoodliao@gmail.com, pychen@cc.ncue.edu.tw, beiwue2006@gmail.com, joslimzy@gmail.com

***Abstract*—High definition (HD) camera is widely used in surveillance systems. An HD camera with optical zoom is useful for monitoring a large area. However, it is inconvenient for a user to manually control the optical zoom for a long time. To exploit the functionality and extend the application domains of a HD camera, the zooming should be controlled automatically. Therefore, an automatic zooming mechanism is proposed in this paper. When the number of an object is small in the field of view (FOV) of the camera and an object is moving through the FOV, the zoom is controlled for capturing the object as clear as possible. A clear object image is useful for related image-based services, such as face recognition. In order to achieve the above goal, a Gaussian Mixture Model (GMM), temporal image differencing, a CamShift tracking method, and a Kalman filter are utilized for object detection and tracking. Then, an adaptive neuro-fuzzy inference system (ANFIS) is used to learn and determine a suitable value for adjusting the zoom. According to the experimental study of the prototype, the results show that the proposed mechanism is useful to capture the clear images of moving objects in a practical environment. A face detection algorithm is also used to demonstrate the feasibility of the captured clear images.**

***Keywords*—Object tracking, Surveillance system, Intelligent video surveillance, Neural network**

I. INTRODUCTION

Surveillance systems are widely adopted for security and safety consideration. In most of cases, surveillance cameras are connected to the internet and called IP (Internet Protocol), or network cameras. The development of hardware technology also increases the image quality of a camera to high definition (HD). An HD quality image is useful for many related image-based functions, such as face recognition. There are mainly two types of camera commonly used in a surveillance system, fixed cameras and active cameras. A fixed camera can only be used to monitor a fixed area. On the other hand, an active camera is equipped for pan/tilt/zoom operations. Such a camera can be used to monitor a large area. However, an active camera is inconvenient to be manually controlled for a long time period. These two types of cameras are shown in Fig. 1. A control mechanism must be designed for specific applications, e.g., moving object tracking, smoke/flame detection.

(a) **(b)**

Fig. 1. Two types of cameras (a) fixed camera (b) active camera

In this study, an automatic zooming mechanism is proposed for tracking a moving person and capturing clear images, using a HD fixed camera with an optical zooming lens. The mechanism enables the facility to capture a clear image of a moving person during the tracking period. An example shown in Fig. 2 is used to illustrate the implemented system. A fixed HD camera is used to monitor a square at a distance. The distance is about 100 meters in the experimental site. When a person is moving through the square, the person is blurred and difficult to identify from his face if the zoom is also fixed at wide-angle. If the zooming function can be controlled to track the person as shown in Fig. 2(a), a series of images as shown in Fig. 2(b) can be easily captured. For these six captured images, the leftmost and rightmost images are still blurred. However, two clear images can be captured as shown in the series of the images. The clear images are useful for the related image-based functions or services, such as face recognition. Basically, the HD camera is controlled to zoom in when a person is moving toward to the center of the square. Oppositely, the camera is controlled to zoom out when the person is moving away from the center of the square.

In order to achieve the above purpose, an adaptive neuro-fuzzy inference system (ANFIS) is used to design the zoom controller in the zooming mechanism. The zoom value and location of the moving person in a real-time image are input to the ANFIS. Accordingly, a suitable value for adjusting the zoom is derived and submitted to the camera. The zoom is controlled to be as large as possible

Manuscript received March 7, 2016. This work is supported by Ministry of Science and Technology, Taiwan, R.O.C., under Grant MOST 103-2221-E-324-016.

Hsien-Chou Liao is with the Department and Graduate Institute of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung, Taiwan, R.O.C. Post code: 413. (Corresponding author, phone: +886-4-2332-3000~4211; fax: +886-4-2374-2375; e-mail: seafoodliao@gmail.com).

Zi-Yi Lim is with Chaoyang University of Technology, Taichung, Taiwan, R.O.C. Post code: 413. (phone: +886-4-2332-3000~7713; fax: +886-4-2374-2375; e-mail: joslimzy@gmail.com).

Po-Yueh Chen and Zi-Jun Lin are with the Department and Graduate Institute of Computer Science and Information Engineering, National Changhua University of Education, Taiwan, R.O.C. Post code: 500. (e-mail: pychen@cc.ncue.edu.tw, beiwue2006@gmail.com).

and to capture the image as clear as possible.

Fig. 2. The demonstration of the automatic zooming mechanism (a) moving person tracking (b) captured images

The remainder of this paper has been organized as follows: Section 2 reviews related work; Section 3 presents the method of the proposed zooming mechanism; Section 4 describes the experimental study; and Section 5 presents the conclusion and the need for future work as indicated by the results of this study.

II. Related Works

Several previous studies related to HD cameras are discussed here. Methods or systems were proposed for various requirements under some specific environments. These studies are helpful to increase the functionalities of HD cameras. For example: S. C. Chan et al. proposed an intelligent video surveillance (IVS) system in a multi-HD camera environment [1]. Every HD camera was connected to a proposed FPGA (Field Programmable Gate Array) board. This board was designed to handle some basic functions of pattern recognition, such as background model establishment, foreground object detection and tracking. The results were then transmitted to a server for handling advanced analysis functions with high-computing needs, such as consistent labeling of objects cameras or computing of image depth information.

G. Scotti et al. proposed a pedestrian classification system, called panoramic scene analysis (PSA) system, by integrating omni-directional cameras and HD PTZ (Pan-Tilt-Zoom) cameras [2]. The PTZ camera can monitor a large area by using its rotation and zooming functions. It can also provide HD images of pedestrians. Firstly, PSA system transformed the polar image to a wide-view panoramic image. Then, the system classified the moving objects into pedestrians or vehicles. The PSA system controlled the PTZ camera to track all the pedestrians

Besides, M. S. Sayed and J. G. R. Delva proposed an efficient intensity correction algorithm for HD images [3]. The trend of surveillance systems is toward low cost, high efficiency, and high resolution. Many intelligent surveillance functions, such as intrusion detection, should be built in the HD camera for providing efficient response or alarm. However, most of the function can only perform well under fixed light condition. Therefore, the proposed intensity correction algorithm is useful to increase the robustness of the intelligent functions. The correction can be either global or local. An apparent gain factor is defined for intensity correction. The algorithm is also realized in Xilinx Spartan3A digital signal processor (DSP) XC3SD3400A device. The results showed that the algorithm can process 1080p (1920×1080) images at 30 frames per second (FPS).

According to the above studies related to HD cameras, the method or system design must take into account the high computing load on processing HD image. In this paper, the design of zooming mechanism not only speeds up the processing of HD images, but also integrates the tracking technique and fuzzy controller for an HD camera with optical zoom in order to increase the feasibility of HD camera in the practical environment.

III. Method

In order to design an automatic zooming mechanism for a HD fixed camera, two basic steps are performed. One is foreground object detection and tracking, the other is zoom control. After several attempts to achieve the desired results, the final process is depicted in Fig. 3. Firstly, the foreground moving objects are detected. When an object is detected, its template is extracted for tracking. CamShift tracker is used in this process. Then, the location and moving direction of the object is estimated and input to the ANFIS zoom controller. A suitable delta zoom value for zoom-in or zoom-out is submitted to the HD camera. These main steps are presented in the following subsections.

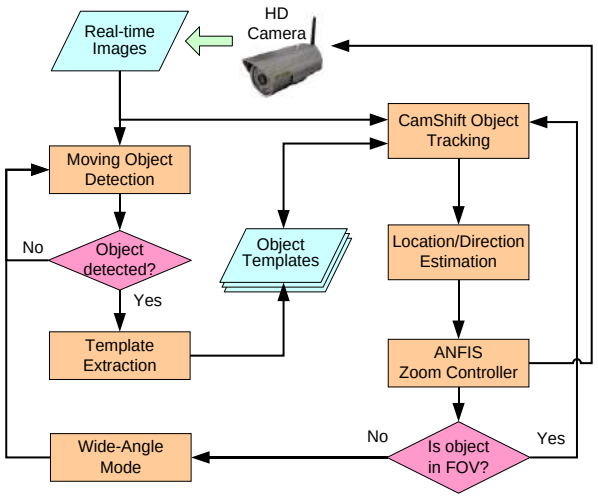


Fig. 3. The process of the proposed automatic zooming mechanism

A. Object detection phase

Foreground object detection is the first step for zooming control. Two popular methods are used in this step, background subtraction and temporal differencing. In this paper, the object detection is performed only in the wide-angle mode. In order to avoid the influence of the environmental brightness, the Gaussian Mixture Model (GMM) is used to construct the background model. Then, the real-time image is subtracted with the background image to detect the foreground objects. Several morphological operations are also used to generate clear foreground objects.

In GMM, a set of Gaussian distributions is used to represent the distribution of pixel values [4]. The model can be represented using Eq. (1).

$$p(x_N|\lambda) = \sum_{i=1}^{M} w_i g_i(x_N) \quad (1)$$

Here, $p(x_N|\lambda)$ is formed by M probability density functions of GMM. Notation x represents the input data and N is the number of input data. λ is formed by $\{w_i, \mu_i, \Sigma_i\}, i = 1,2,\dots,M$, where w_i is mixture weighted value, μ_i is the average vector, Σ_i is covariance matrix. $g_i(x)$ represents the i-th density function of the Gaussian distribution. The initial values of M, μ, Σ, and w are determined by using the K-means algorithm. Then, Maximum likelihood estimation (MLE) method is used for updating the values to obtain the best results. The density function of GMM is represented in Eq. (2).

$$E(\lambda) = \ln(\prod_{i=1}^{n} P(x_i)) = \sum_{i=1}^{n} \ln P(x_i) \quad (2)$$

Then, expectation maximization (EM) algorithm is used to find the maximal value iteratively, as shown in Eq. (3).

$$\beta_m(x) = p(m|x) = \frac{w_m g(x|\mu_m, \Sigma_m)}{\sum_{j=1}^{M} w_j g(x|\mu_j, \Sigma_i)} \quad (3)$$

$\beta_m(x)$ is the generating function of m-th Gaussian distribution. w_m, μ_m, and Σ_m are the parameters to be estimated and can be obtained by Eq. (4)-(6):

$$\mu_m = \frac{\sum_{i=1}^{n} \beta_m(x_i) x_i}{\sum_{i=1}^{n} \beta_m(x_i)} \quad (4)$$

$$\Sigma_m = \frac{\sum_{i=1}^{n} \beta_m(x_i)(x_i - \mu_m)(x_i - \mu_m)^T}{\sum_{i=1}^{n} \beta_m(x_i)} \quad (5)$$

$$w_m = \frac{1}{n} \sum_{i=1}^{n} \beta_m(x_i) \quad (6)$$

Finally, the results of the above three parameters are applied to Eq. (3) to check whether or not the convergence condition is satisfied. If it is not satisfied, the iteration of EM algorithm is repeated.

An example of foreground object detection is shown in Fig. 4. The background image constructed by using GMM is shown in Fig. 4(a). The real-time image shown in Fig. 4(b) is subtracted by the background image and the result is shown in Fig. 4(c), which contains some noise or fragments. After the processing of morphological operations, including erosion and dilation, the final result is shown in Fig. 4(e), which is a clear detection.

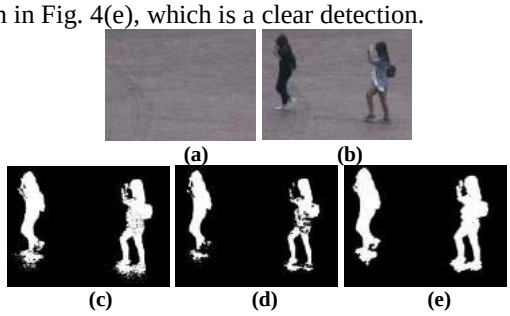


Fig. 4. An example of foreground object detection (a) GMM background image (b) real-time image (c) background subtraction (d) erosion (e) dilation

B. Object tracking phase

After an object is detected, the next step is object tracking. CamShift tracking algorithm is used in this step. CamShift algorithm combines basic mean-shift algorithm with an adaptive region sizing step [5]. In general, the size of a moving object is varied during tracking. CamShift algorithm can reduces the error caused by the change of object size in the mean-shift algorithm. It searches the most likely region, which is most similar to the template, in an area. Assume the template image at time t is denoted as F_t. The probability distribution of F_t is denoted as $P(t)$. The centroid of F_{t-1} and $P(t)$ is used to search the new location of the template in the CamShift algorithm. In order to achieve the above goal, the zero order moments (M_{00}) and the first order moments (M_{01} and M_{10}) are estimated by using Eq. (7)-(10):

$$M_{00} = \sum_{y}^{h} \sum_{x}^{w} I(x, y) \quad (7)$$

$$M_{10} = \sum_{y}^{h} \sum_{x}^{w} xI(x, y) \quad (8)$$

$$M_{01} = \sum_{y}^{h} \sum_{x}^{w} yI(x, y) \quad (9)$$

$$X_c = \frac{M_{10}}{M_{00}}, \; Y_c = \frac{M_{01}}{M_{00}} \quad (10)$$

$I(x, y)$ is the pixel intensity at the location (x, y) of F_t. h and w represent the height and width of the search area, respectively. X_c and Y_c represent the moving distance in the X and Y direction, respectively. The above process is repeated until the convergence of the location (X_c, Y_c) is achieved. An example of such a convergence is shown in Fig. 5.

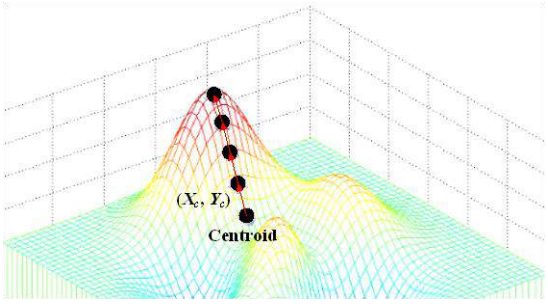


Fig. 5. Convergence of the location (X_c, Y_c) in the CamShift algorithm

A series of (X_c, Y_c) can be obtained in the above CamShift algorithm. However, the location is unstable due to the influence of noise or environment brightness. So, a Kalman is used to smooth the tracking results. The filter is described in the following equations [6].

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (11)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (12)$$

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (13)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (14)$$

$$P_k = (I - K_k H)P_k^- \quad (15)$$

where $\hat{x}_k^-$ is a priori state estimate at step k given knowledge of the process prior to step k, $\hat{x}_k$ is a posteriori state estimate at step k given measurement. z_k The matrix A relates the state at the previous time step k-1 to the state at the current step k, and matrix B relates the optional control input u_k to the state x. P_k is estimate error covariance, K_k is Kalman gain, Q is process noise covariance, and R is the

measurement noise covariance. Eqs. (11) and (12) are time update equations. Eqs. (13)-(15) are measurement update equations. An application of Kalman filter is shown in Fig. 6. A series of y coordinates is processed by the Kalman filter. The original coordinates in blue are smoothed into the curve in red.

Fig. 6. An application of Kalman filter (parameters Q=0.001 and R=0.0001)

C. Zooming mechanism

For the zoom control of an HD fixed camera, the camera should be zoomed in when the object is moving toward the center of FOV. Oppositely, the camera should be zoomed out when the object is moving away from the center of FOV. Therefore, a fuzzy controller is used for the zooming mechanism. In advance, the settings of fuzzy rules and membership functions are the main challenge on designing the fuzzy controller. The adaption of a fuzzy controller from one environment to another is also an issue. According to the above considerations, an adaptive neuro-fuzzy inference system (ANFIS) is used on the design of automatic zooming mechanism [7]. The first-order Sugeno fuzzy model is used in ANFIS. The structure of ANFIS is shown in Fig. 7.

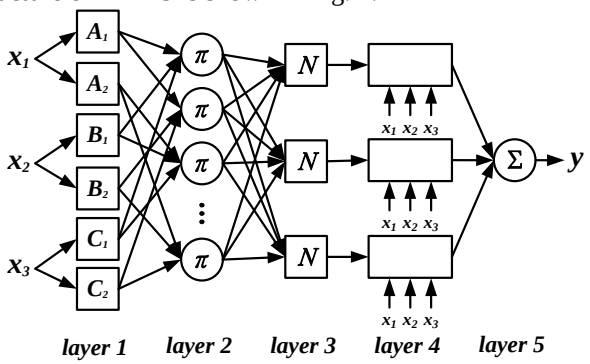


Fig. 7. The structure of ANFIS

ANFIS is a five-layered structure. Every layer is described as follows:

1) The first layer is input layer. Every node in this layer is an adaptive node with a node function represented in Eq. (16).

$$O_{1,i} = \mu A_i(x), \text{for } i = 1,2, \text{or}$$
$$O_{1,i} = \mu B_{i-2}(y), \text{for } i = 3,4 \quad (16)$$

where $O_{1,i}$ is the membership value of the input x or y corresponding to the fuzzy set A_i or B_i. The parameters in this layer are called premise parameters.

2) The second layer is rules. All the nodes are fixed node, denoted as π. Its output equals the multiplication of all the input as represented in Eq. (17). The output represents the strength of the rule.

$$O_{2,i} = w_i = \mu A_i(x)\mu B_i(y), i = 1,2 \quad (17)$$

3) The third layer is the normalization layer. The nodes are denoted as N. In a node i, the ratio of the strength of i-th rule to the strength of all the rules are computed using Eq. (18). The output is the normalized firing strength.

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1+w_2}, i = 1,2 \quad (18)$$

4) The forth layer is inference layer of consequence. The node i in this layer is also an adaptive node with the node function represented in Eq. (19). $\overline{w_i}$ is the output and $\{p_i, q_i, r_i\}$ is the parameter set. All the parameters are called consequence parameters.

$$O_{4,i} = \overline{w_i} f_i = w_i(p_i x + q_i y + r_i), i = 1,2 \quad (19)$$

5) The fifth layer is output layer. There is only one node in this layer denoted as Σ. The output is the summation of all the outputs in the previous layer as represented in Eq. (20).

$$O_{5,1} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (20)$$

Although ANFIS is more complicated than the conventional fuzzy inference system, it cannot be applied to any fuzzy inference applications unless the following criteria are satisfied:

1) The ANFIS structures must be the first-order or zero-order Sugeno models.
2) Singleton output: A weighted average method is used for defuzzification.
3) Different rules cannot be applied to the same membership function.
4) The weight of all the rules is the same.

Sugeno models apply polynomial functions to construct the consequent values. A weighted average method defuzzification method is used to generate the final result.

The construction of the fuzzy inference system (FIS) is based on the subtractive clustering method. The parameters, including range of influence, squash factor, accept ratio, and reject ratio are set to 0.2, 1.25, 0.5, and 0.15, respectively. When the FIS is initialized, a set of a series of zoom values is collected. During the period of an object moving through the monitoring area of HD camera, the zoom values of manual control are recorded. Then, a fuzzy error backpropagation algorithm is used to learn the fuzzy control rules membership functions. The errors are propagated backward from layer to layer through the node links. A steepest descent method is used to decrease the error by updating the weights in the neurons of the different layers. The parameters, error tolerance and epochs, are set to 0.001 and 10000, respectively. According to the moving directions of foreground objects, four fuzzy controllers are learning separately. The inputs of the controller include the current zoom value and location. The training results are shown in Fig. 8. The blue circles represent the training data

and the red circles represent the training results. There are 600 training data, including 200, 200, 100, and 100 data for left, right, up and down directions separately. When an HD camera is installed at a different environment, a fuzzy controller can still be trained based on the above process.

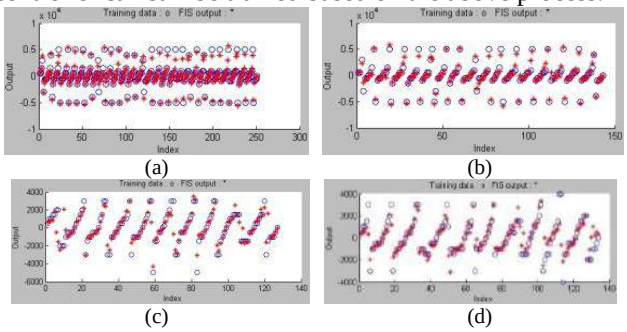


Fig. 8. The training data and results based on zoom value and location (a) up (b) down (c) left (d) right

Another set of fuzzy controllers are also trained using the zoom value and moving speed of an object as inputs. The training data and results are shown in Fig. 9. There are four controllers for different directions. However, the zoom value is not close related to the moving speed. Therefore, the errors of these controllers are larger than the previous controllers shown in Fig. 8. The maximal zoom value can be achieved during the moving period is smaller than the previous controllers. It causes the captured image is also less clear than that captured by using the previous controllers. So, the previous controllers are chosen in the design of zooming mechanism.

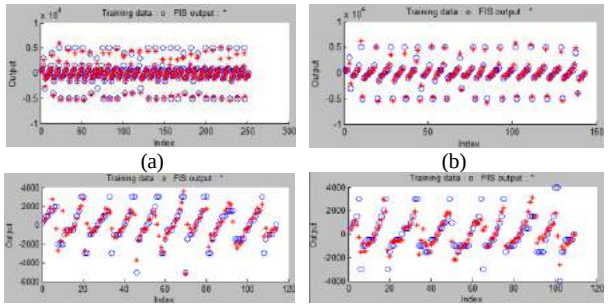


Fig. 9. The training data and results based on zoom value and moving speed (a) up (b) down (c) left (d) right

IV. Experimental Study

An experiment is designed to evaluate the performance of the automatic zooming mechanism. The experiment design and results are presented in the following subsections.

A. Experiment Design

The experiment is conducted on a commercial HD camera (AXIS Q1755) and a prototype implemented using Visual C# 2012, MATLAB R2012a and the EmguCV. The screenshot of the prototype is shown in Fig. 10. The prototype is executed on a PC with i3-550 3.2GHz CPU and 6GB RAM.

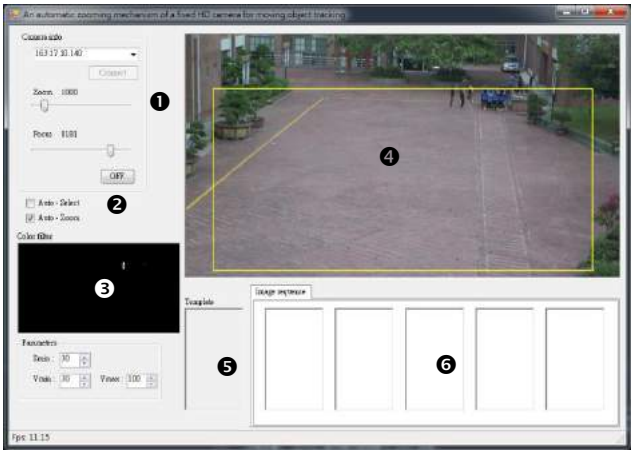


Fig. 10. The screenshot of the prototype

The area marked with ❶ is the basic control of the HD camera. The area marked with ❷ is the functions of the automatic zooming mechanism. The checkbox "Auto-Select" enables the system to select a moving object automatically. The checkbox "Auto-Zoom" enables the automatic zooming mechanism. The area marked with ❸ is the result of the foreground object detection and the setting of parameters. The area marked with ❹ is the real-time image of HD camera. The area marked with ❺ is the tracking template of an object. The area marked with ❻ is the captured images of a moving object under different zooms.

There are totally 70 tests in the experiment. 50 tests are objects moving in horizontal direction and 20 tests are objects moving in vertical direction. The tests of horizontal moving are larger than those of vertical moving because most of the objects are moving in horizontal direction. A moving object is selected randomly in a square and the automatic zooming mechanism is enabled. The zooming mechanism adjusts the zoom every 600 and 800ms for horizontal and vertical moving, respectively. For the vertical moving of objects, the frequency of zoom adjustment is lower because the object size and location is more stable, compared with the horizontal moving. For further analysis, the images of the tracking object are captured before the zoom adjustment as well.

B. Experiment Results

When the automatic zooming mechanism of the prototype is enabled, the zoom of the HD camera is controlled to capture the object image as clear as possible. Two examples of the captured images with two different moving directions are shown in Fig. 11 and Fig. 12. The image resolutions are increasing when the object is moving toward the center and then decreasing when leaving the center.

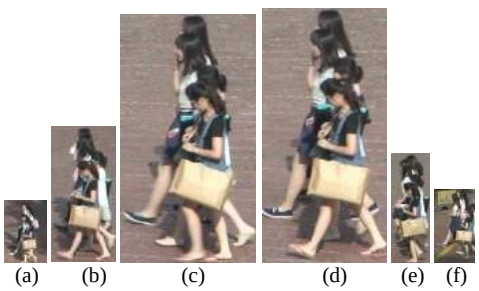


Fig. 11. A series of images captured when the object is moving left (a) 0s (b) 3.8s (c) 5.2s (d) 5.3s (e) 8.9s (f) 11.8s

Fig. 12. A series of images captured when the object is moving down (a)0s (b)1s (c)4s (d)5.1s (e)6.6s (f)7.7s (g)9.7s

The results are analyzed based on the captured images collected in the experiment. There are three results described below:

1) The processing time: The average processing time of all steps are computed. An HD image is usually resized to shorten the processing time.
2) The maximal zoom value: For every tracking, the maximal zoom value is recorded and counted to realize the performance of the automatic zooming mechanism. A higher zoom value means that the zooming mechanism is effective to control the optical zoom of an HD camera.
3) The maximal height of an object: The larger the height of an object is, the clearer is the object quality. Therefore, the maximal height of an object is recorded and counted.

Firstly, the processing time of an image in each step is recorded. In order to reduce the computing load of an HD image, the image is resized to 640×480. The average time of the foreground object detection, CamShift tracking, and zoom control is 7.96 ms, 3.11 ms, and 2.65 ms, respectively. It shows that the proposed zooming mechanism is efficient.

The largest zoom value of the HD camera is 10,000. The maximal zoom values of object tracking are recorded for horizontal and vertical moving separately. These values are summarized by counting the ratio of the maximal values larger than a specific value. The results are depicted in Fig. 13. For the curve of the horizontal moving, it shows that all the maximal zoom values are larger than 3,000 (100%). The ratio remains high (92%) for the zoom value 6,000. For the zoom value 7,000, the ratio is decreased to 80%. For the vertical moving, the object in the image is stable. The zooming mechanism is able to achieve higher zoom than the horizontal moving. So, the ratio of vertical moving is always higher than the ratio of horizontal moving. The results show that the zooming mechanism can achieve a quite high zoom value for capturing clear images.

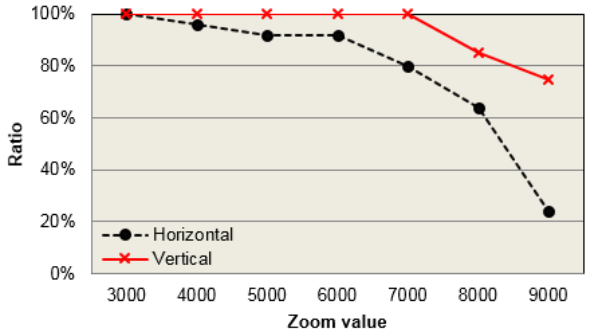


Fig. 13. The percentage of the maximal zoom values

The maximal height of an object is also recorded. The distributions of the maximal heights of two kinds of moving directions are shown in Fig. 14. For the horizontal moving, most of the heights are within 350 to 500 pixels. Oppositely, most of the heights are within 500 to 600 pixels for the vertical moving. It is caused by the same reason that vertical moving is more stable.

Some examples of the captured images with the maximal height are shown in Fig. 15(a) and (b). They illustrate that the object images can be identified clearly.

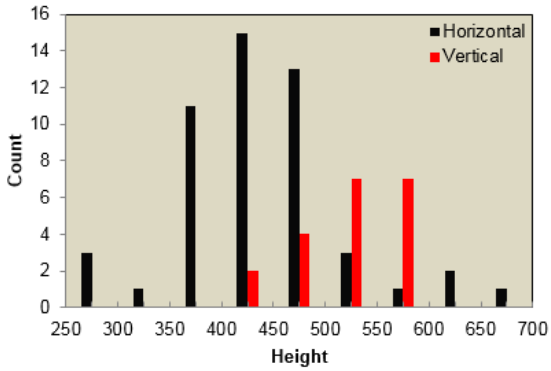


Fig. 14. The distributions of the maximal heights

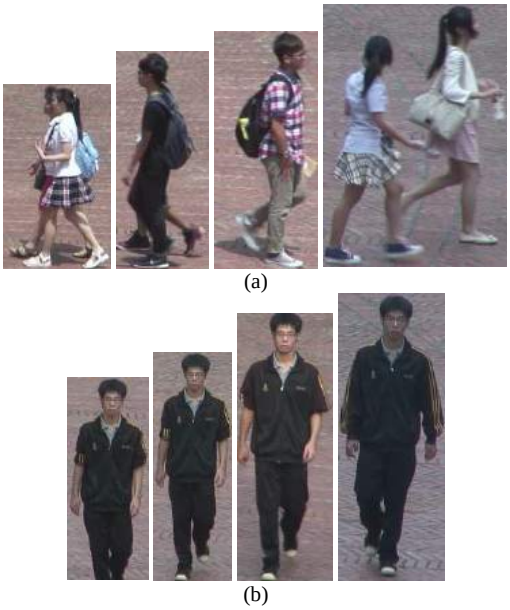


Fig. 15. The captured images with various heights (a) horizontal (b) vertical

Besides, 20 captured images with maximal zoom value of the vertical moving are processed for face detection algorithm. The detection algorithm is based on the skin regions and the geometric constraints, such as ratio. The detection algorithm is implemented by using OpenCV

library. All of them can be detected successfully as shown in Fig. 16 since their zoom value is larger than 6,000. Several examples are also shown in Fig. 17. When the face is detected, the detection result is marked by a red circle. These results demonstrate that the proposed automatic zooming mechanism is useful to capture clear images for related image-based services, such as face recognition.

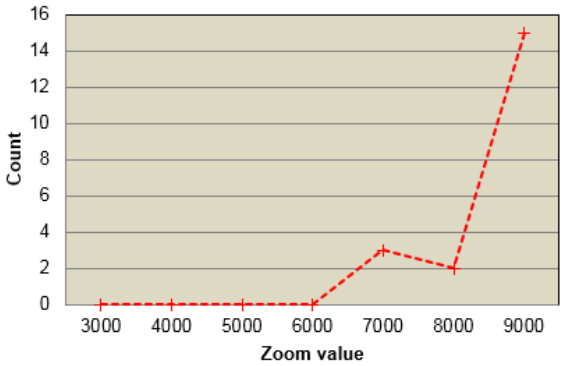


Fig. 16. The zoom values of successful face detection

Fig. 17. Examples of the captured images applied to face detection

Besides, the time spans between two successive zoom adjustments are also recorded. The distributions of time spans for horizontal and vertical directions are shown in Figure 18. The x-axis represents the time span in milliseconds and the y-axis represents the counts of adjustments. The time spans of horizontal direction are about 1,400 to 1,600 milliseconds. The time spans of vertical directions are about 1,700 to 1,900 milliseconds. The moving of an object in the vertical directions is stable and causes the time spans can be longer than that of horizontal direction. The time span is also related to the training data of ANFIS. If the time spans of manual zoom adjustments can be shorten, the time spans of the automatic zooming mechanism can be shorten, too. The maximal zoom values will be possibly higher than the current mechanism.

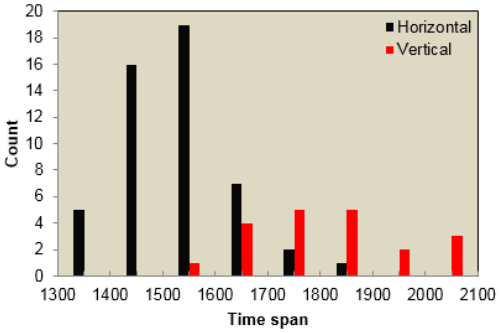


Fig. 18. The distribution of time span between zoom adjustments

V. Conclusion

HD cameras are widely used in a surveillance system. In this paper, an automatic mechanism is proposed for the HD fixed camera with zooming capability. When an object is moving through the FOV of the camera, a clear image can be captured and preserved. The mechanism consists of the object detection, CamShift tracking, ANFIS algorithm, and fuzzy controller. ANFIS is used to help the fuzzy controller learn the fuzzy rules and membership functions from the manual control data. The mechanism can be used in a new environment based on the same learning process.

According to the experiment results, the maximal zoom values of 80% tests can reach 7,000. It ensures that a high resolution image of the moving object can be captured. Furthermore, the average processing time of the mechanism is efficient, about 13.7ms. The face can also be detected successfully in the captured images. It demonstrates that the images captured by the automatic zooming mechanism are feasible for image-related services. Many related services, such as license plate recognition (LPR), can be applied to such HD fixed cameras.

In the experimental study, it is found that the zooming mechanism is directly influenced by the object tracking. The tracking accuracy will be enhanced in the future to increase the performance of the mechanism.

References

[1] S. C. Chan, S. Zhang, J. F. Wu, H. J. Tan, J. Q. Ni, and Y. S. Hung, "On the Hardware/Software Design and Implementation of a High Definition Multiview Video Surveillance System," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, Vol. 3, No. 2, June 2013, pp. 248-262.

[2] G. Scotti, A. Cuocolo, C. Coelho, and L. Marchesotti, "A Novel Pedestrian Classification Algorithm for a High Definition Dual Camera 360 Degrees Surveillance System," *The IEEE International Conference on Image Processing* (*ICIP 2005*), Genoa, Italy, 11-14 Sept. 2005, Vol. III, pp. 880-883.

[3] M. S. Sayed and J. G. R. Delva, "An Efficient Intensity Correction Algorithm for High Definition Video Surveillance Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 21, No. 11, Nov. 2011, pp. 1622-1630.

[4] S. C. Jeng, "A GMM-based Method for Dynamic Background Image Model Construction with Shadow Removal," *Master Thesis, National Chiao-Tung University, ECE*, June, 2005, pp. 83.

[5] G. R. Bradski, "Real Time Face and Object Tracking as a Component of a Perceptual User Interface," *The Fourth IEEE Workshop on Applications of Computer Vision* (*WACV'98*), 1998, pp. 214-219.

[6] G. Welch and G. Bishop. "An Introduction to the Kalman Filter," *Technical Report, TR95-041, University of North Carolina, Chapel Hill*, 2004, 16 pages.

[7] J. S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 23, 1993, pp. 665-685.

Hsien-Chou Liao received the B.S. and Ph.D. degrees in Computer Science and Information Engineering from National ChiaoTung University in 1991 and 1998, respectively. He is a professor, Department of Computer Science and Information Engineering, at Chaoyang University of Technology, Taiwan, R.O.C. He is also a senior member of IEEE. His research interests include location-based service, smart technology, mobile computing, computer vision, and pattern recognition.

Po-Yueh Chen received his B.S, M.S., and Ph.D. degrees in EE from National Taiwan University in 1988, National Chiao Tung University in 1990 and University of Maryland at College Park in 1997 respectively. Currently, Dr. Chen is a faculty

member in Department of Computer Science and Information Engineering, National Changhua University of Education, Changhua, Taiwan. His research interests include image processing, FPGA implementations, VLSI architecture design and digital IC design.

Zi-Jun Lin received the B.S. and M.S. in Computer Science and Information Engineering from the Chaoyang University of Technology in 2014 and National Changhua University of Education in 2016, respectively. His current research interests are in the computer vision.

Zi-Yi Lim received the B.S. and M.S. in Computer Science and Information Engineering from the Chaoyang University of Technology, Taiwan, Republic of China, in 2014 and 2016. His current research interests are in the area of computer vision.

Analysis of Very High Throughput (VHT) at MAC and PHY Layers under MIMO Channel in IEEE 802.11ac WLAN

Gul Zameen Khan*, Ruben Gonzalez*, Eun-Chan Park**, Xin-Wen Wu*
*School of Information and Communication Technology, Griffith University, Australia
**Dept. of Information and Communication Engineering, Dongguk University, South Korea
gz.khan@griffithuni.edu.au, r.gonzalez@griffith.edu.au, ecpark@dongguk.edu, x.wu@griffith.edu.au

***Abstract*—This paper analyses the very high system throughput of IEEE 802.11ac by taking into consideration the key features of MAC and PHY layers under a Multiple In Multiple Out (MIMO) channel. Throughput at the MAC layer is calculated from the transmission probability, contention window and transmission stage. Likewise, the new critical attributes of 802.11ac PHY (i.e. modulation and coding schemes, spatial streams, and channel bandwidth) are used to determine the throughput at the PHY layer. To this end, a theoretical model is formulated at the MAC and PHY layers followed by a system model of MIMO multipath fading channel for 802.11ac. The system model is verified by simulation analysis. The results compare theoretical and simulation findings for different sets of parameters. Furthermore, important trends and trade-offs are identified between system throughput and (MAC + PHY) features as a function of number of contending stations and payload size. The system throughput of 802.11ac networks is significantly improved due to the addition of new PHY features. However, the system may degrade upto 50% in terms of symbol reception in case of a high error-prone MIMO channel. The performance of 802.11ac systems is also analyzed under different MIMO TGn channel models in terms of Packet Error Rate (PER). Thus based on our simulation results, an appropriate channel model can be chosen for 802.11ac network under a given configuration to achieve a better performance.**

***Keywords*—Performance, analysis, throughput, MAC, physical, MIMO, multipath, fading, transmission probability, contention window, modulation, coding, spatial streams, channel bandwidth, channel model**

I. Introduction

Gigabit Wireless Local Area Networks (WLAN) is a state-of-the-art technology based on the emergence of new IEEE standard i.e., 802.11ac [1]. The standard achieves Very High Throughput (VHT) with the aid of efficient Modulation and Coding Schemes (MCS) such as 256-Quadrature Amplitude Modulation (QAM), explicit transmit beamforming, enhanced Multiple Input Multiple Output (MIMO) technology, and large bandwidth. Most of the leading vendors and manufacturers have already implemented 802.11ac in their Wi-Fi chipsets [2]. 802.11ac operates in 5 GHz band and can support a high data rate up to 6.933 Gb/s.

Currently, most of the literature available on 802.11ac focuses on improving the underlying technology of 802.11ac. This includes but is not limited to exploring Single User (SU) and Multi User (MU) MIMO, adding new Spatial Streams (SS), improving frame aggregation techniques, channel bonding, and incorporating advanced modulation and coding techniques in 802.11ac. In [3], a review of PHY layer features of 802.11ac is presented. The performance of MU-MIMO is analysed via a testbed. However, the chips used in the experiment are based on 802.11n [4]. In [5], the authors present a performance analysis of energy efficiency and interference in 802.11ac. It is shown that larger channels consume more power while the addition of more SS is energy efficient. However, the performance measurements are not verified by any theoretical model. Although a comparison of 802.11ac and 802.11n is drawn in [6] in terms of different frame aggregation techniques, other key features of 802.11ac have not been explored. Similarly, [7] discusses the requirement in MAC modifications and enhancements for downlink MU-MIMO transmission. In particular, it introduces the technique of enhancing Transmit Opportunity (TXOP) and the revised backoff procedures. In the same manner, the authors of [8] present a review of how the capacity can be estimated and optimized using Ekahau Site Survey tool in 802.11ac. Similarly, [9] presents a performance comparison between 802.11n and 802.11ac in terms of throughput for three MAC frame aggregation techniques under constant PHY conditions. For the most part, 802.11ac outperforms 802.11n due to its larger frames in error-free channel. On the contrary, the optimal frame size is determined by bit error in error-prone channel conditions. The paper however, considers only the frame size instead of other key factors of MAC and PHY layers. Similarly, an overview of static and dynamic channel selection methods is demonstrated in [10] with the help of simulations. It is shown that dynamic selection of a primary channel achieves high throughput for 802.11ac when other stations operating in 802.11a/n occupy

Manuscript received on March 10, 2016. This work is a follow-up of the invited journal to the accepted out-standing conference paper of the 18th International Conference on Advanced Communication Technology (ICACT2016).

Gul Zameen Khan is with the School of ICT, Griffith University, Australia. (corresponding author, phone: +61450039010, email: gz.khan@griffithuni.edu.au)

Dr. Ruben Gonzalez is a senior faculty member, head of Intelligent Wireless Technology and Applications research group, and director of the bachelor of computer science program at School of ICT, Griffith University, Australia. (email: r.gonzalez@griffith.edu.au).

Prof. Eun-Chan Park is with the Dept. of Information and Communication Engineering, Dongguk University, South Korea. (email: ecpark@dongguk.edu)

Dr. Xin-Wen Wu is with the School of ICT, Griffith University, Australia. (email: x.wu@griffith.edu.au)

the secondary channel. However, the simulation is performed in an ideal environment without taking into consideration the effects introduced by PHY and channel impairments.

Previous work addresses the individual features of 802.11ac. However, a comprehensive performance analysis of 802.11ac is important in order to estimate the achievable throughput. This paper extends our previous work [11] and considers several new features of 802.11ac (i.e., MCS, channel bandwidth, spatial stream) which can be a baseline to determine the system level throughput of an 802.11ac wireless network. To this end, a theoretical analysis based on both MAC and PHY layers is presented under a MIMO channel which is followed by simulation results. The paper thoroughly investigates the impacts of different features of 802.11ac on system throughput under different set of parameters. In addition, it provides a baseline model to measure the MAC and PHY layers performance of 802.11ac in terms of aggregate throughput. The impact of TGn channel models on the performance of 802.11ac is also presented for a complete networked system analysis of VHT.

The remainder of this paper is organized as follows: A brief overview of 802.11ac is discussed in section II. Section III describes a theoretical system model, MIMO channel model of 802.11ac and simulation set up. In Section IV, results and discussions are presented. Lastly, the paper is concluded in Section V.

II. Overview of IEEE 802.11ac

There are three key features that lead to VHT in 802.11ac. Each of them is described as follows:

A. More Spatial Streams(SS)

802.11ac has increased the number of SS from 4 in 802.11n up to 8 at PHY layer in MIMO Orthogonal Frequency Division Multiplexing (OFDM). There are three main advantages of using more MIMO in 802.11ac namely:

i. Extending the range
ii. Improving reliability
iii. Achieving higher throughput

In addition to SU-MIMO, 802.11 WAVE-2 also supports down link MU-MIMO in which an Access Point (AP) can send multiple data frames in the form of Aggregated MAC Protocol Data Unit (A-MPDU) to multiple receivers at the same time. In practice, the current 802.11ac devices support 3 SS and 4 SS in 802.11ac WAVE-1 and WAVE-2, respectively. However, a total of 8 SS can be supported in the upcoming products [1], [12].

B. Modulation and Coding

A more advanced modulation i.e. 256 QAM has been added to 802.11ac standard. This increases the number of bits per sub-carrier of OFDM compared to 64 QAM. As a result, the PHY data rate raises up to 33% as compared to previous 802.11 standards. On one hand it significantly increases data rate; on the other hand, however, it requires higher Signal to Noise Ratio (SNR) for receivers to correctly demodulate the symbols. Similarly, 802.11ac supports various coding rate of $\frac{1}{2}, \frac{2}{3}, \frac{3}{4}$, and $\frac{5}{6}$ [1].

C. Wider Channel Bandwidth

With 5 GHz band, the bandwidth of 802.11ac has been increased. In addition to 20 MHz and 40 MHs channels that were already available in 802.11n, wider channels of 80 MHz and 160 MHz have been added in 802.11ac. Furthermore the 160 MHz channel can be established with two contiguous or non-contiguous 80 MHz channels [1].

III. Simulation and Numerical Analysis

In this section we describe our system model for MAC and PHY layers and derive a theoretical formulation to calculate the performance of MAC and PHY layers numerically. We also formulate a MIMO channel model for 802.11ac and describe our simulation environment.

A. MAC Layer Theoretical Analysis

We consider a single hop fully connected Single Basic Service Set (BSS) with n Stations(STAs) and one AP. It is assumed that each STA can sense transmission from every other STA in the same BSS. All STAs operate in uplink saturated mode i.e., they always have data to send to the AP. A data transmission is considered successful if it is followed by an Acknowledgement (ACK) from the AP, otherwise it is retransmitted. The wireless channel is assumed ideal i.e., a frame is failed only due to collision that occurs when two or more STAs access the shared channel simultaneously.

We consider the Markov model of [13] that represents the Distributed Coordination Function (DCF) as two stochastic processes namely: $b(t)$ and $s(t)$ to model the backoff time counter and the backoff stage, respectively. Without the loss of generality, let CW_{min} and CW_{max} represent minimum and maximum contention windows, respectively. Let m indicates maximum backoff stage such that $CW_{max} = 2^m CW_{min}$. Let CW_i represents the contention window of a STA at ith backoff stage i.e., $CW_i = 2^i CW_{min}$ where $i \in [0, m]$. Thus $b(t)$ takes any random value from $[0, CW_i]$ where i is modelled by $s(t)$. For the sake of simplicity, we represent $CW_{min} = W$. The key approximation is that collision probability is constant regardless of retransmission stage. This is a reasonable approximation as long as W and n get larger.

During a randomly selected slot, a STA senses the channel in one of the three states namely: idle state (no transmission activities), busy due to successful transmission, or busy due to collision. Suppose a STA attempts to transmit a frame in a randomly chosen slot with probability τ. The system considers a Binary Exponential Backoff (BEB) mechanism which doubles the contention window on collision. If p represents the collision probability; then τ is given by [13]

$$\tau = \frac{2(1-p)}{(1-2p)(W-1) + pW(1-(2p)^m)} \quad (1)$$

The probability of collision p can be calculated from the fact that collision occurs if at least one of the remaining $n-1$ STAs starts transmission. Therefore, if $1-\tau$ is the probability that exactly one STA is idle then $(1-\tau)^{n-1}$ is the probability

that $n-1$ STAs are idle. It follows that the probability that at least one of $n-1$ STAs transmits is given by [13]

$$p = 1 - (1-\tau)^{n-1} \tag{2}$$

Transmission probability τ and collision probability p can be calculated numerically by solving Eq. 1 and Eq. 2 using some numerical method (e.g., fixed point iteration). In addition, it can be proved that this system of non linear equations has a unique solution [13]. We have used Maple 15 [20] to solve Eq. 1 and Eq. 2.

We are interested in calculating throughput S of the system which is expressed as a ratio of average payload information transmitted in a slot per average duration of a slot.

$$S = \frac{E[D]}{E[T]} \tag{3}$$

where $E[D]$ is the expected value of data transmitted successfully in a randomly selected slot while $E[T]$ is the average length of a time slot;

$$E[D] = P_{tr} P_s E[L] \tag{4}$$

where P_{tr} denotes the probability that there is at least one transmission in the considered time slot. On the other hand, P_s is the probability that the given transmission is successful and $E[L]$ denotes the average length of payload data. Consequently, P_{tr} can be calculated for n contending stations as

$$P_{tr} = 1 - (1-\tau)^n \tag{5}$$

Similarly, P_s can be calculated from the fact that a transmission is successful if and only if exactly one STA transmits given that at least one STA transmits among n STAs, i.e.,

$$P_s = \frac{n\tau(1-\tau)^{n-1}}{1-(1-\tau)^n} \tag{6}$$

Accordingly, $E[D]$ is calculated from Eq. 5 and Eq. 6. In order to calculate $E[T]$, let us denote T as a random variable that indicates a randomly selected time slot. Moreover T takes any of the following three values:

$$T = \begin{cases} \sigma & \text{if the medium is idle} \\ T_s & \text{if successful transmission} \\ T_c & \text{if there is collision} \end{cases} \tag{7}$$

where σ is the duration of an empty slot while T_s and T_c are the average time when the channel is busy due to successful transmission and collision, respectively. The corresponding probability for these three cases can be calculated as

$$f_T(t) = \begin{cases} 1 - P_{tr} & \text{if } T = \sigma \\ P_{tr} P_s & \text{if } T = T_s \\ P_{tr}(1-P_s) & \text{if } T = T_c \end{cases} \tag{8}$$

Using Eq. 7 and Eq. 8, $E[T]$ can be calculated as

$$E[T] = \sum_{\forall t,T} T f_T(t) \tag{9}$$

Finally, the normalized throughput S is calculated by using Eq. 4 and Eq. 9 in Eq. 3. As far as T_s and T_c are concerned, they are calculated using PHY system model discussed in the following sub section.

TABLE I
802.11AC PHY FRAME

L-STF	L-LTF	L-SIG	VHT-SIG-A	VHT-STF	VHT-LTFs	VHT-SIG-B	Data

B. PHY Layer Theoretical Analysis

As discussed in Section II, the IEEE 802.11ac standard can increase throughput with the help of wider Radio Frequency (RF) channel bandwidth, more spatial streams, MU-MIMO, and advanced MCSs. The performance of PHY can be modelled by taking into account the aforementioned features. TABLE I shows a general format of a PHY layer frame [1].

Let T_x be the transmission time of a STA. We assume that the frame aggregation is not employed. Then, T_x can be derived as follows [1].

$$T_x = T_{LEG-PREAMBLE} + T_{L-SIG} + T_{VHT-SIG-A} + T_{VHT-PREAMBLE} + T_{VHT-SIG-B} + T_{DATA} \tag{10}$$

$$T_{LEG-PREAMBLE} = T_{L-STF} + T_{L-LTF} \tag{11}$$

$$T_{VHT-PREAMBLE} = T_{VHT-STF} + N_{VHTLTF} \times T_{VHT-LTF} \tag{12}$$

where T_{L-SIG}, $T_{VHT-SIG-A}$, $T_{VHT-SIG-B}$, T_{LTF}, T_{STF}, $T_{VHT-STF}$, $T_{VHT-LTF}$ are fields of PHY frame shown in TABLE II. N_{VHTLTF} shows the number of long training symbols which is determined from the number of space-time streams [14]. Similarly,

$$T_{DATA} = \begin{cases} N_{SYM} \times T_{SYM} & \text{for long GI} \\ T_{SYML} \lceil \frac{T_{SYMS} \times N_{SYM}}{T_{SYML}} \rceil & \text{for short GI} \end{cases} \tag{13}$$

where GI, T_{SYM}, T_{SYMS} and T_{SYML} indicate Guard Interval, symbol interval, short GI symbol interval, and long GI symbol interval, respectively. Their values are listed in TABLE II.

$$T_{SYM} = \begin{cases} T_{SYML} & \text{for long GI} \\ T_{SYMS} & \text{for short GI} \end{cases} \tag{14}$$

$$N_{SYM} = m_{STBC} \times \lceil \frac{M}{m_{STBC} \times N_{DBPSL}} \rceil \tag{15}$$

where $\lceil x \rceil$ = smallest integer $\geq x$ and N_{DBMS} indicates the number of data bits per symbol.

$$M = 8 \times APEP_{LENGTH} + N_{Service} + N_{tail} \times N_{ES} \tag{16}$$

$$m_{STBC} = \begin{cases} 2 & \text{if STBC is used} \\ 1 & \text{otherwise} \end{cases} \tag{17}$$

where $APEP_{LENGTH}$ indicates the final value of A-MPDU i.e., payload size and N_{ES} represents the number of Binary Convolution Code (BCC) encoders. The value of N_{ES} depends on the MCS and channel bandwidth and they are listed in TABLEs IV-VII. Space-Time Block Coding (STBC) is an encoding technique that greatly improves the reliability of communication in 802.11ac. T_x is calculated by using Eq. 11-14 in Eq. 10.

TABLE II
MAC AND PHY PARAMETERS

Parameter	Value	Para	Value	Para	Value
Slot time (σ)	9 μs	CW_{min}	16	T_{DIFS}	34 μs
L_{macH}	34 bits	CW_{max}	1024	T_{SIFS}	16 μs
$T_{VHT-STF}$	4 μs	T_{SYMS}	3.6 μs	T_{STF}	8 μs
$T_{VHT-SIG-A}$	8 μs	T_{L-SIG}	4 μs	T_{LTF}	8 μs
$T_{VHT-SIG-B}$	4 μs	T_{SYML}	4 μs	N_{tail}	6 bits
$T_{VHT-LTF}$	4 μs	$N_{service}$	16 bits	ρ	1 μs

Now, we calculate successful transmission time (T_s) and collision time (T_c) for basic DCF i.e., without Ready To Send (RTS)/Clear To Send (CTS).

$$T_s = T_{DIFS} + T_x + \rho + T_{SIFS} + T_{ACK} + \rho$$

$$T_c = T_{DIFS} + T_x + \rho + T_{ACK-TOut}$$

where T_{DIFS}, T_{SIFS}, T_{ACK}, and ρ indicate DCF Inter-Frame Spacing time, Short Inter-Frame Spacing time, transmission time of ACK frame and propagation delay of 802.11ac frame, respectively. Their values are listed in TABLE II. Similarly, $T_{ACK-TOut}$ represents the time out for ACK frame and is calculated as

$$T_{ACK-TOut} = T_{ACK} + T_{SIFS} + \rho$$

The T_s and T_c for DCF with RTS/CTS can be determined in a similar way.

C. Channel Model for 802.11ac

In order to evaluate the impact of the channel on MAC and PHY layers of 802.11ac, we consider a set of channel models that are designed for IEEE 802.11 WLAN [15]. Each model is applicable to a specific environment with a set of 6 profiles, labelled A to F. All these profiles cover different scenarios as listed in TABLE III. Each channel model has a path loss model including shadowing, and a MIMO multipath fading model, which describes the multipath delay profile, the spatial properties, the K-factor distribution shown as Rician K-factor in TABLE III, and the Doppler spectrum.

Each channel model has a certain number of taps which are associated with specific delays. Furthermore, each channel model is comprised of a certain number of clusters. A cluster is made up of a set of taps. The number of taps, Root Mean Square (RMS) delay (σ_{RMS}), maximum delay (σ_{Max}), the number of clusters and standard deviation of shadow fading both in case of Line of Sight (LOS) and Non Line of Sight (NLOS) for each model are listed in TABLE III. The LOS K-factor is applicable only to the first tap while all the other taps K-factor remain at $-\infty$ dB.

A set of spatial properties are defined for each cluster:

i. Mean Angle of Arrival (AoA)
ii. Mean Angle of Departure (AoD)
iii. Angular Spread (AS) at transmitter
iv. AS at receiver

These parameters determine the transmit and receive correlation matrices associated with each tap delay. The LOS component can only be present on the 1st tap. If the distance between the transmitter and the receiver is greater than d_{BP} then LOS component is not present. Note that d_{BP} is the break-point distance or distance of first wall (i.e., distance of transmitter from the first reflector). The d_{BP} for all channel models (A-F) are listed in 5th column of TABLE III.

We consider a path loss model that takes into account the free space loss L_{FS} (log-distance model with the path-loss exponent of 2) up to d_{BP} and log-distance model with the path-loss exponent of 3.5 after d_{BP} [16]. For each of the models, a different break-point distance d_{BP} was chosen.

$$L(d) = \begin{cases} L_{FS}(d) & \text{if } d \leq d_{BP} \\ L_{FS}(d_{BP}) + 35log10(d/d_{BP}) & \text{if } d > d_{BP} \end{cases} \quad (18)$$

where d is the distance between transmitter and receiver. The other parameters of the path loss model are listed of TABLE III. The standard deviations of log-normal (Gaussian in dB) shadow fading are also included in 10th column in TABLE III. The values were found to be in the range between 3 and 14 dB [17].

Similarly, the zero-mean Gaussian probability distribution is given by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{[\frac{x^2}{2\sigma^2}]} \quad (19)$$

We are interested in modelling the MIMO channel of 802.11ac. Thus the correlation between transmit and receive antenna is an important aspect of the MIMO channel. To this end, we follow a procedure based on the transmitter and receiver correlation matrices [18] to calculate the MIMO channel matrix H for each tap, at one instance of time, in the A-F delay profile models. The channel matrix H is derived as a sum of two matrices namely: a fixed LOS matrix with constant entries, and a Ryleigh NLOS matrix with variable entries as follows

$$H = \sqrt{P}\left(\sqrt{\frac{K}{K+1}}H_F + \frac{1}{K+1}H_V\right) \quad (20)$$

where P shows the power of each tap which is obtained by summing all the power of LOS and NLOS powers, and K is the Ricean K-factor. Eq. 20 can be expressed for any number of transmitter and receiver for MIMO. If there are T input antennas at the transmitter and R output antennas at the receiver then H_F and H_V in Eq. 20 can be represented as

$$H_F = \begin{bmatrix} e^{j\phi_{11}} & e^{j\phi_{12}} & \dots & e^{j\phi_{1T}} \\ e^{j\phi_{21}} & e^{j\phi_{22}} & \dots & e^{j\phi_{2T}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j\phi_{R1}} & e^{j\phi_{R2}} & \dots & e^{j\phi_{RT}} \end{bmatrix}$$

$$H_V = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1T} \\ X_{21} & X_{22} & \dots & X_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ X_{R1} & X_{R2} & \dots & X_{RT} \end{bmatrix}$$

where $e^{j\phi_{ij}}$ shows the constant elements of LOS matrix H_F and X_{ij} represents the element of variable NLOS Rayleigh matrix H_V between i^{th} receiving and j^{th} transmitting antenna.

TABLE III
PARAMETERS OF A-F PATHLOSS CHANNEL MODELS 802.11AC

Channel name	User case scenario	Conditions	K (dB)	D_{BP}(m)	No of taps	σ_{RMS}	σ_{Max}	No of clusters	Shadow fading std. dev. (dB) before/after d_{BP} (LOS/NLOS)
A	Flat Fading	LOS	0	5	1	0	0	1	3
		NLOS	$-\infty$						4
B	Residential	LOS	0	5	9	15	80	2	3
		NLOS	$-\infty$						4
C	Residential/ small office	LOS	0	5	14	30	200	2	3
		NLOS	$-\infty$						4
D	Typical office	LOS	3	10	18	50	390	3	3
		NLOS	$-\infty$						6
E	Large office	LOS	6	20	18	100	730	4	3
		NLOS	$-\infty$						6
F	Large space indoors and outdoors	LOS	6	30	18	150	1050	6	3
		NLOS	$-\infty$						6

It is assumed that X_{ij} is a complex Gaussian random variable with zero mean and unit variance.

In order to correlate the X_{ij} elements of the matrix H_V, the following method is used

$$[X] = [R_{rx}]^{\frac{1}{2}}[H_{iid}][R_{tx}]^{\frac{1}{2}} \tag{21}$$

where R_{rx} and R_{tx} are the receive and transmit correlation matrices, respectively, and H_{iid} is a complex Gaussian random variable. All these Gaussian random variables are supposed to be identically independent with zero mean and unit variance. In addition, R_{tx} and R_{rx} are given by

$$[R_{tx}] = [\rho_{txij}] \tag{22}$$

$$[R_{rx}] = [\rho_{rxij}] \tag{23}$$

where ρ_{txij} are the complex correlation coefficients between i^{th} and j^{th} transmitting antennas, and ρ_{rxij} are the complex correlation coefficients between i^{th} and j^{th} receiving antennas.

Alternatively, we use another approach i.e., Kronecker product of the transmit and receive correlation matrices to calculate X [18].

$$[X] = ([R_{tx}] \otimes [R_{rx}])^{\frac{1}{2}}[H_{iid}] \tag{24}$$

It can be seen that H_{iid} is an array in this case instead of matrix. The R_{tx} and R_{rx} matrices are given as:

$$R_{tx} = \begin{bmatrix} 1 & \rho_{tx12}{}^* & \cdots & \rho_{tx1T}{}^* \\ \rho_{tx21} & 1 & \cdots & \rho_{tx2T}{}^* \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{txR1} & \rho_{txR2} & \cdots & 1 \end{bmatrix}$$

$$R_{rx} = \begin{bmatrix} 1 & \rho_{rx12}{}^* & \cdots & \rho_{rx1T}{}^* \\ \rho_{rx21} & 1 & \cdots & \rho_{rx2T}{}^* \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{rxR1} & \rho_{rxR2} & \cdots & 1 \end{bmatrix}$$

The values of complex correlation coefficient ρ are calculated from power angular spectrum (PAS), AS, mean AoA, mean AoD and individual tap powers [15], [19]. Consequently, for the Uniform Linear Array (ULA), the complex correlation coefficient at the linear antenna array is expressed as

$$\rho = R_{XX}(D) + jR_{XY}(D) \tag{25}$$

where $D = 2\pi d/\lambda$ (λ shows the wavelength in metre) , and R_{XX} and R_{XY} are the cross-correlation functions between the real parts and between the real part and imaginary part, respectively, with

$$R_{XX}(D) = \int_{-\pi}^{\pi} \cos(D\sin\phi)PAS(\phi)d\phi \tag{26}$$

$$R_{XY}(D) = \int_{-\pi}^{\pi} \sin(D\cos\phi)PAS(\phi)d\phi \tag{27}$$

We calculate the correlation coefficients matrices in three ways namely: uniform, truncated Gaussian, and truncated Laplacian PAS shapes [15].

D. Simulation Environment

We have implemented MAC and PHY layers of 802.11ac in matlab with given parameters as defined in [1]. TABLE II-VII list the parameters that are used in our theoretical analysis as well as simulation setup. We do not consider the frame aggregation scheme for the sake of simplicity. To simulate different features of PHY, we summarize the MCSs into four tables i.e., TABLES IV-VII for 20 MHz, 40 MHz, 80 MHz, and 160 MHz channels, respectively. The values of MCSs are chosen such that we cover maximum modulation and coding schemes.

We consider various modulation schemes namely: Quadrature Phase Shift Keying (QPSK), 16-QAM (Quadrature Amplitude Modulation), 64-QAM, and 256-QAM. The coding rate (R) is chosen as $\frac{1}{2}$, $\frac{2}{3}$, $\frac{3}{4}$, and $\frac{5}{6}$. In the same way, the number of SS (N_{SS}) is selected to be 1, 2, 4, and 8. The data rate is calculated from Eq. 13 whereas ACK rate is fixed at basic rate. We have run each simulation 20 times and have calculated average values to show stable results.

In order to implement a TGn channel in matlab, we use the parameters from TABLE III. In addition, we use OFDM with MIMO channel at our transmitter and receiver for our PHY and channel analysis of 802.11ac.

TABLE IV
MCS FOR 20 MHZ CHANNEL

MCS	Modulation	R	N_{ss}	N_{DBPS}	N_{ES}	Data rate (Mbps) 800 ns GI	400 ns GI
1	QPSK	1/2	1	52	1	13	14.4
			4	208	1	52	57.8
			8	416	1	104	115.6
3	16-QAM	1/2	1	104	1	26	28.9
			4	416	1	104	115.6
			8	832	1	208	231.1
5	64-QAM	2/3	1	208	1	52	57.8
6	64-QAM	3/4	1	234	1	58.5	65
7	64-QAM	5/6	1	260	1	65	72
8	256-QAM	3/4	1	312	1	78	86.7

TABLE V
MCS FOR 40 MHZ CHANNEL

MCS	Modulation	R	N_{ss}	N_{DBPS}	N_{ES}	Data rate (Mbps) 800 ns GI	400 ns GI
1	QPSK	1/2	1	108	1	27	30
			2	216	1	54	60
			4	432	1	108	120
			8	864	1	216	240
3	16-QAM	1/2	1	216	1	54	60
			2	432	1	108	120
			4	864	1	216	240
			8	1728	1	432	480
5	64-QAM	2/3	1	432	1	108	120
9	256-QAM	5/6	1	720	1	180	200

TABLE VI
MCS FOR 80 MHZ CHANNEL

MCS	Modulation	R	N_{ss}	N_{DBPS}	N_{ES}	Data rate (Mbps) 800 ns GI	400 ns GI
1	QPSK	1/2	1	234	1	58.5	65
			4	936	1	234	260
			8	1872	1	468	520
3	16-QAM	1/2	1	468	1	117	130
			4	1872	1	468	520
			8	3744	2	936	1040
5	64-QAM	2/3	1	936	1	234	260
8	256-QAM	3/4	1	1404	1	351	390
			4	5616	3	1404	1560
			8	11232	6	2808	3120

TABLE VII
MCS FOR 160 MHZ CHANNEL

MCS	Modulation	R	N_{ss}	N_{DBPS}	N_{ES}	Data rate (Mbps) 800 ns GI	400 ns GI
1	QPSK	1/2	1	468	1	117	130
			4	1872	1	468	520
			8	3744	2	936	1040
3	16-QAM	1/2	1	936	1	234	260
4	16-QAM	3/4	1	1404	1	351	390
5	64-QAM	2/3	1	1872	1	568	520
			4	7488	4	1872	2080
			8	14976	8	3744	4160
8	256-QAM	3/4	1	2808	2	702	780
			4	11232	6	2808	3120
			8	22464	12	5616	6240

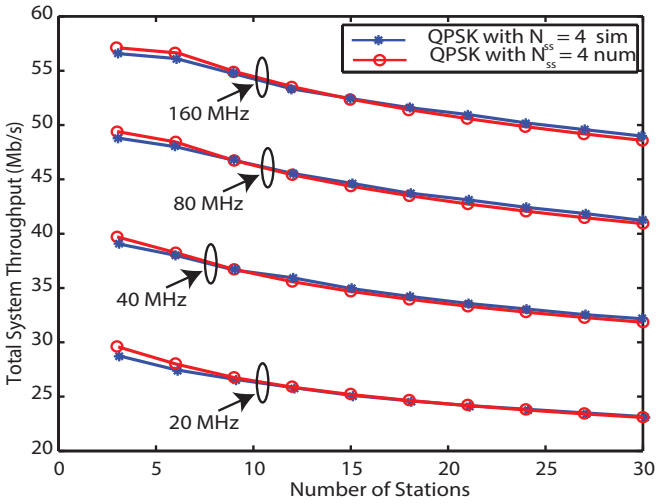


Fig. 1. Throughput for different channels as a function of number of STAs

IV. RESULTS AND DISCUSSIONS

We calculate aggregate throughput for different number of STAs and payload size.

A. Wider Channels

In order to evaluate the effect of wider channels on system throughput, we calculate total throughput for 20 MHz, 40 MHz, 80 MHz, and 160 MHz channels. We simulate a case that uses QPSK with $N_{ss} = 4$, and $GI = 400\ ns$. The payload size is fixed at 1500 bytes. As shown in Fig. 1, the total throughput decreases for the all cases of channel bandwidths as the number of STAs increases. However, the total throughput increases with respect to the increase of channel bandwidth. Compared to the case of 20 MHz channel bandwidth, the total throughput is increased by more than 10, 20, and 30 Mb/s in the cases of 40, 80, and 160 MHz channel bandwidth, respectively. It is important to note that the increase of total throughput is not proportional to the increase of channel bandwidth, i.e., less than one quarter increase of channel bandwidth. The theoretical (num) results also show a similar trend.

In a similar way, we observe the effects of available channel bandwidth on throughput as a function of payload size. For this purpose, we use a modulation of 16 QAM with one SS i.e., $N_{ss} = 1$, and $GI = 800\ ns$. In the case of 20/40/80/160 MHz channel, the coding rate of 16QAM can be 1/2 or 3/4 (see Table IV-VII). The number of STAs is 20 (i.e., $n = 20$) in this set up. Fig. 2 illustrates that total throughput increases by nearly 10 Mbps as the channel width is doubled.

B. Modulation

The choice of a particular modulation scheme can greatly affect the system throughput. To observe this result, we fix the number of SS to 1 in a 20 MHz channel with $GI = 800\ ns$ and payload size of 1500 bytes. We calculate throughput for different modulation schemes. As illustrated in Fig. 3, QPSK gives the lowest throughput while 256 QAM produces the

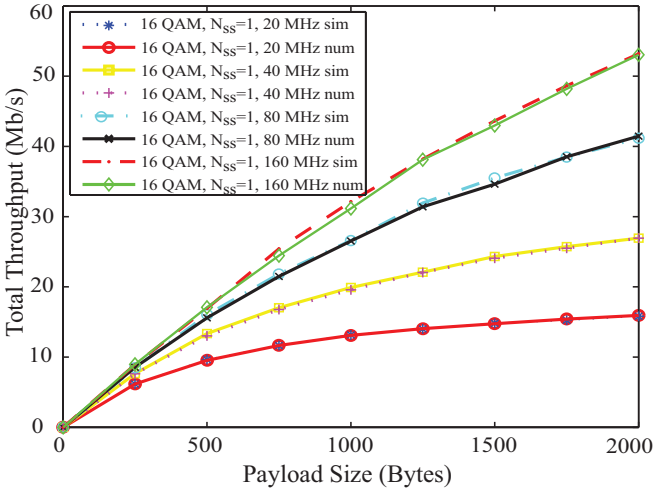


Fig. 2. The effects of wider channels on throughput for different payload size

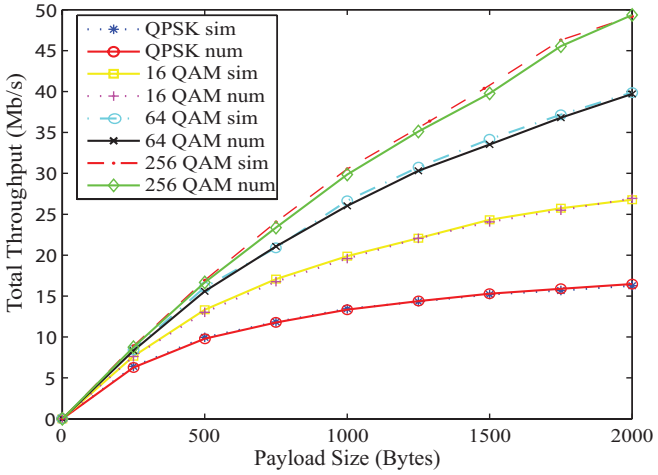


Fig. 4. Throughput for different modulations as a function of payload size

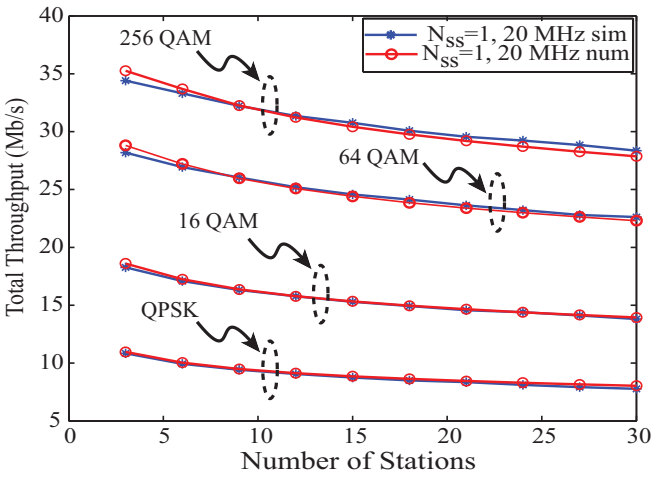


Fig. 3. Throughput of different modulation schemes for different STAs

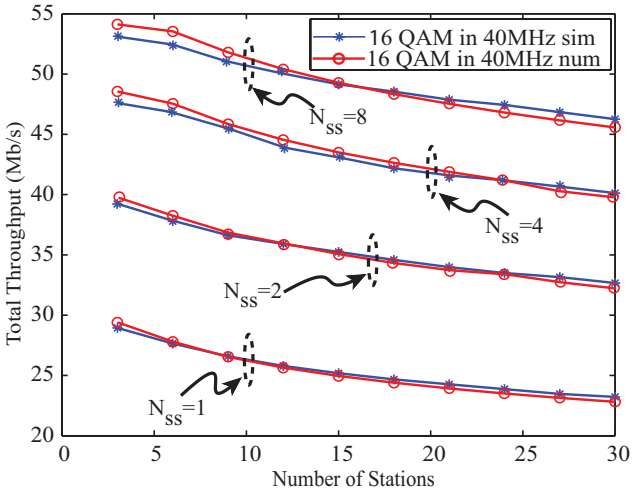


Fig. 5. The effects of SS on throughput for different number of STAs

highest total throughput. The improvement is 20 Mpbs from QPSK to 256 QAM.

It is also worth noting that total throughput is greatly affected by different modulation techniques under different payload sizes. To notice this fact, the number of STAs is fixed to 20 while $N_{ss} = 1$ with QPSK in a 40 MHz channel and $GI = 800\ ns$. Fig. 4 represents that the total throughput increases as the payload size is increased. In addition, the higher order modulation schemes produce higher throughput. The difference of throughput between different modulation schemes is increased with increase in payload size. For instance, when payload size reaches 2000 bytes, there is an elevation of 10 Mbps for each modulation scheme (i.e., QPSK, 16 QAM, 64 QAM and 256 QAM).

C. Multiple Spatial Streams

Now, we focus on the effects of different number of transmitting antennas on total throughput. We consider a case of 16 QAM with 40 MHz channel, $GI = 800\ ns$ and fixed payload size of 1500 bytes. Although 802.11ac may support 1 to 8 SSs, however, for the sake of simplicity and comprehension, we consider $N_{ss} = \{1, 2, 4, 8\}$. The results are depicted in Fig. 5. It is clear that the total throughput decreases as we increase the number of STAs for all SSs. Nonetheless, the overall throughput for a particular number of SS increases by 10 Mbps as the number of SS is doubled. In the same manner, Fig. 6 represents the total throughput for different antenna streams. The modulation used is QPSK with $GI = 800\ ns$ in 40 MHz channel. The number of STAs is once again fixed at 20. As can be seen in Fig. 6, the total throughput increases as we increase the payload size. In particular, the total throughput increases by almost 10 Mbps for every $2N_{ss}$.

D. Coding Rate

The relationship between total throughput and different number of STAs is examined under various coding rate i.e., $R = \{\frac{2}{3}, \frac{3}{4}, \frac{5}{6}\}$. The other parameters are set as $N_{ss} = 1$, modulation = 64 QAM, and $GI = 400\ ns$ in 20 MHz channel. Fig. 7 shows that the total system throughput increases by 1

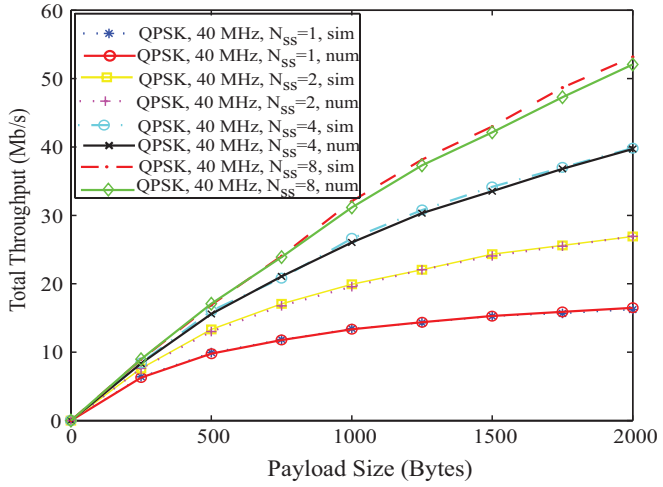


Fig. 6. Throughput for different antenna streams with variable payload size

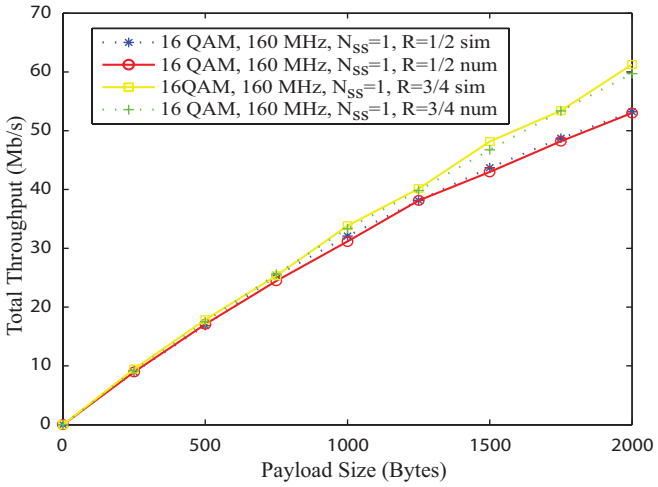


Fig. 8. Throughput of different coding rate for different payload

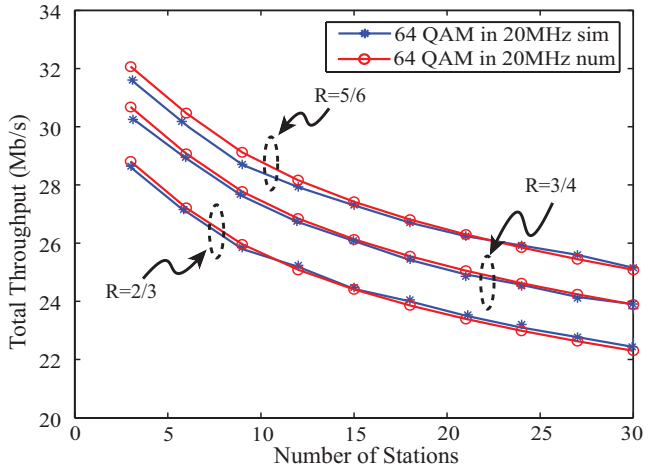


Fig. 7. Throughput in different coding rate (R) for different STAs

Mbps as we move to the next available coding rate under the aforementioned setup.

The influence of coding rate on total throughput as a function of payload size is illustrated in Fig. 8. We choose 16 QAM in 160 MHz channel with $N_{ss} = 1$. The number of STAs is 20 while $GI = 800ns$. The total throughput increase by 4 Mbps for $R = \frac{3}{4}$ than $R = \frac{1}{2}$ when the payload size is 1500. However, the throughput rise reaches upto 9 Mbps for $R = \frac{3}{4}$ than $R = \frac{1}{2}$ as payload size is increased to 2000 bytes. Fig. 8 indicates that coding rate can tremendously affect total throughput under variable frame size.

E. Analysis under TGn Channel

As illustrated in TABLE III, each TGn channel has a different profile. In order to investigate the impact of a channel on MCS, we simulate 802.11ac under channel D and see the performance in terms of Symbol Error Rate (SER). For this purpose, we use an OFDM system under a 40 MHz MIMO multipath Rayleigh fading channel. Accordingly, a 128 point Fast Fourier Transform (FFT) is used with 128 subcarriers for 40 MHz.The bandwidth of each sub carrier is 312.5 KHz for OFDM in 802.11ac. Out of 124 subcarriers, the 114 subcarriers (-58 to 2 and 2 to 58) are used for data and pilot (timing and synchronization). The indices used for pilot subcarriers are -53, -25, -11, 11, 25, and 53. Subcarriers -1, 0, -1 are used as DC while the remaining 11 subcarriers are used as left (6) and right (5) guard bands.

The SER of the system is calculated for BPSK, QPSK, 16-QAM, 64-QAM and 256-QAM under different Energy per Symbol (E_s/N_o) for OFDM transceiver under 40 MHz 2x2 MIMO TGn D channel as shown in Fig. 9. The SER decreases as E_s/N_o increases for all modulation schemes. However, in lower E_s/N_o, the SER of BPSK is much lower as compared to other modulation schemes. Similarly, QPSK shows lower SER at higher noise as compared to other high order modulation schemes. A similar trend can be seen for 16-QAM and 64-QAM as compared to 256-QAM. Thus in an error-prone channel conditions, 802.11ac suffers from almost 50% SER for 256-QAM as compared to BPSK.

Next, we consider the impact of all TGn channels i.e., (A to F) TABLE III on the performance of 802.11ac network. For this purpose, we fix all other parameters except the number of of transmit and receive antennas. This will give us an idea of how 1x1, 2x1, and 4x4 MIMO changes the performance of 802.11ac network under different TGn channels. We use the MIMO OFDM system with 64 point FFT, 20 MHz channel, 64-QAM, coding rate of 1/2 (i.e., MCS = 3) as illustrated in TABLE IV. The packet length is set to 1500 bytes. We calculate the Packet Error Rate (PER) to estimate the performance of MIMO OFDM 802.11ac under different channels. We sent 1000 packets for each Signal to Noise Ratio (SNR) point that ranges from 0 dB to 50 dB. The distance between transmitter and receiver is set to 10 m. We also add Additive White Gussian Noise (AWGN) in order to make a realistic estimation of wireless environment.

We observe that the PER and SNR can be divided into three different levels when the SNR of channel is changed from 0 dB to 50 dB for all the 6 channel models in 1x1, 2x2 and 4x4 MIMOs. This trend can be seen in different configurations of transmit and receive antennas as shown in Fig. 10-15. Let us

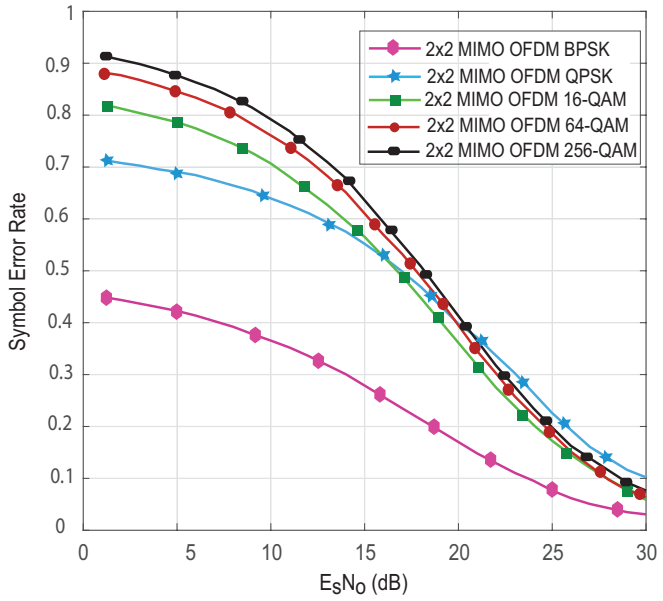


Fig. 9. Symbol Error Rate vs. EsNo for different Modulation schemes

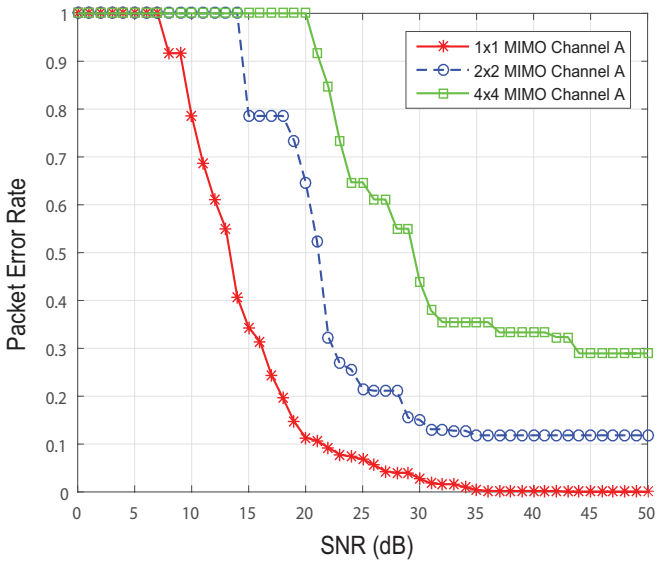


Fig. 10. Packet Error Rate vs. SNR of Channel A under different MIMO configurations

call these three levels as *initial level*, *intermediate level*, and *final level*. The PER is explained with the help of Fig. 10-15 and TABLE VIII-X. As shown in Fig. 10-15 and TABLE VIII-X, the PER remains constant at *initial* and *final levels* while it drops down over a range of SNR values at *intermediate level* for all the channel models under 1x1, 2x2 and 4x4 MIMOs. The range of SNR values for the three levels changes differently for the 6 channel models under different MIMO configurations.

As illustrated in TABLE VIII-X, the PER is 100% at *initial level* for all the channels and all cases. Thus the system performance is worst at *initial level*. However, the range of SNR points at *initial value* increases as we increase the number of transmit and receive antennas. For example: the average range of SNR (average of A-F) is 0 to 9.6 dB (TABLE VIII) , 0 to 14.3 dB (TABLE IX), and 0 to 20.1 dB (TABLE X) for 1x1, 2x2, and 3x3 MIMOs, respectively. It shows that the performance of 1x1 MIMO is better than that of 2x2 MIMO and 4x4 MIMO. Similarly, the performance of 2x2 MIMO is better than that of 4x4 MIMO in terms of packet reception at lower SNR values. In the same way, the PER remains constant at *final level* for all channel models and all MIMO configuration. However, the values of PER and the range of SNR are different for different channels under different MIMOs. For example at *final level*, the PER and SNR of channel model A remains 0%, 11.82%, and 28.94% at 35-50 dB, 35-50 dB, and 44-50 dB for 1x1 MIMO, 2x2 MIMO, and 4x4 MIMO configurations, respectively. A similar trend can be found for other channels. On the other hand for *intermediate level*, the PER drops down from higher values to lower values over a range of SNR points. The slop and trend of each drop is different for different channel models and different MIMO configurations.

Based on our simulation results, we find interesting performance patterns for 802.11ac under different channel models. As shown in Fig. 10-15 and TABLE VIII, the PER of 802.11ac is 100% at *initial level* and 0% *final level* for channel models A, B, and C under 1x1 MIMO. The PER is 2% and 6% for channel models D and E, respectively. However, the PER is 11.70% in case of channel model F which is the worst performance for 1x1 MIMO for a given configurations.

Similarly, under 2x2 MIMO, the system performs best for channel models B, C and D where the PER remains at 2% at *final level* for these three channel models as shown in Fig. 10-15 and TABLE IX. However, the system performance degrades to some extent for channel models A, E, and F where the PER is 11.82%, 6.1%, and 18.03%, respectively. The PER for channel model F is the maximum (18.03%) as compared to other models. Channel model B relatively provides the best results under 2x2 MIMO as shown in Fig. 10-15.

The PER of 802.11ac network remains lower for channel models B, C, and D under 4x4 MIMO as illustrated in Fig. 10-15 and TABLE X. In this case, the performance is worst for channel model A where PER remains 28.94% at *final level*. The system achieves relatively the best performance for channel model D as compared to the other channel models under 4x4 MIMO configuration. This can be seen in TABLE X where the PER is 0%, 84.61%-0.26%, and 0.20% for *initial*, *intermediate*, and *final levels*, respectively.

V. Conclusion

This paper investigated the performance of 802.11ac in terms of system throughput under its several new key features. A theoretical model was presented that is based on MAC and PHY layer parameters. It was shown through simulation and theoretical analysis that the choice of a particular modulation and coding scheme, number of spatial streams, and channel size can greatly affect the total throughput of system. A MIMO multipath fading channel model was formulated to further investigate the effects of new features of 802.11ac. Although 802.11ac increases throughput many fold due to high-order

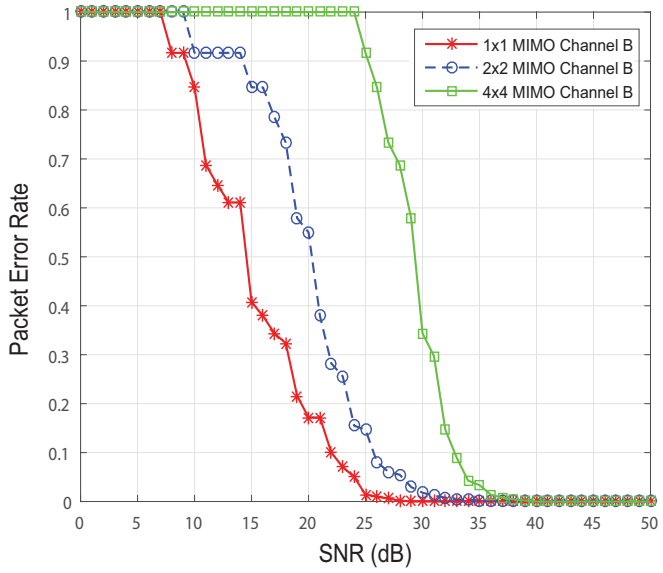


Fig. 11. Packet Error Rate vs. SNR of Channel B under different MIMO configurations

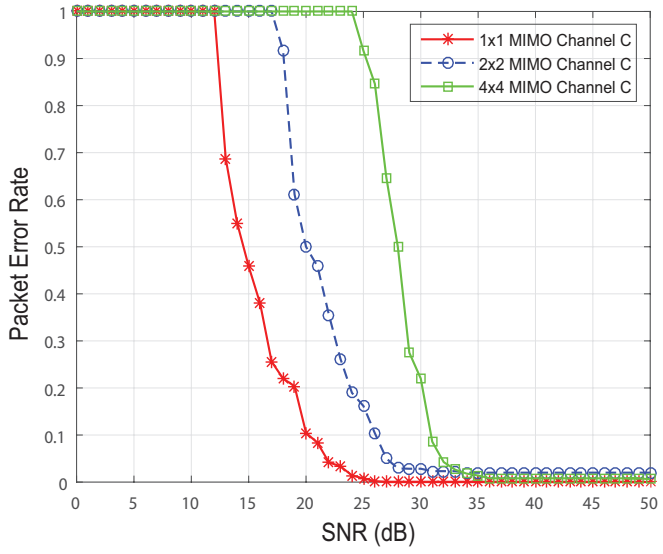


Fig. 12. Packet Error Rate vs. SNR of Channel C under different MIMO configurations

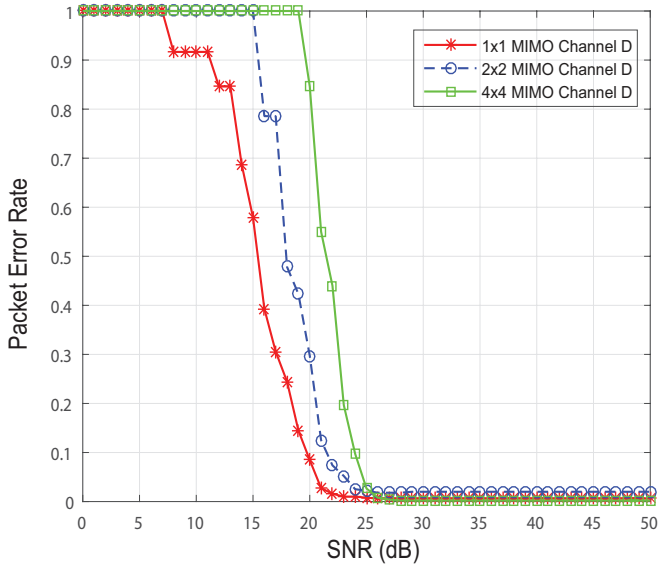


Fig. 13. Packet Error Rate vs. SNR of Channel D under different MIMO configurations

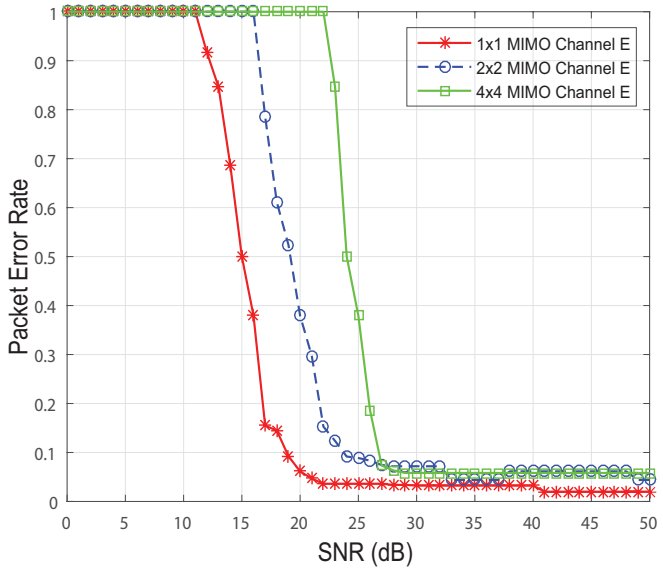


Fig. 14. Packet Error Rate vs. SNR of Channel E under different MIMO configurations

TABLE VIII
PER LEVELS FOR A RANGE OF SNR OF 1X1 MIMO-OFDM 802.11AC WLAN SYSTEM

Channel model	Parameter name	Initial level	Intermediate level	Final level
A	SNR (dB)	0-7	8-34	35-50
	PER (%)	100	91.66-2	0
B	SNR (dB)	0-7	8-25	26-50
	PER (%)	100	91.66-2	0
C	SNR (dB)	0-13	14-26	27-50
	PER (%)	100	68.75-2	0
D	SNR (dB)	0-7	8-25	26-50
	PER (%)	100	91.66-8	6
E	SNR (dB)	0-11	12-40	41-50
	PER (%)	100	91.66-3.2	2
F	SNR (dB)	0-13	14-20	21-50
	PER (%)	100	68.75-14.10	11.70

TABLE IX
PER LEVELS FOR A RANGE OF SNR OF 2X2 MIMO-OFDM 802.11AC WLAN SYSTEM

Channel model	Parameter name	Initial level	Intermediate level	Final level
A	SNR (dB)	0-14	15-34	35-50
	PER (%)	100	78.57-12.79	11.82
B	SNR (dB)	0-9	10-34	35-50
	PER (%)	100	91.66-4	2
C	SNR (dB)	0-17	18-33	34-50
	PER (%)	100	91.66-2.32	2
D	SNR (dB)	0-15	16-28	29-50
	PER (%)	100	78.57-1.8	2
E	SNR (dB)	0-16	17-36	37-50
	PER (%)	100	78.57-3.4	6.1
F	SNR (dB)	0-15	16-33	34-50
	PER (%)	100	91.66-19.29	18.03

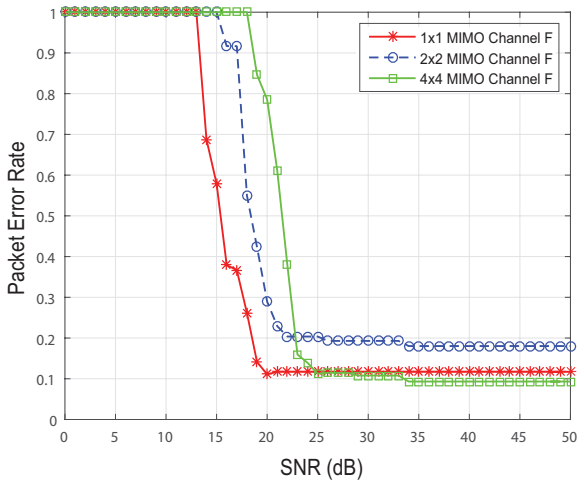


Fig. 15. Packet Error Rate vs. SNR of Channel F under different MIMO configurations

TABLE X
PER LEVELS FOR A RANGE OF SNR OF 4X4 MIMO-OFDM 802.11AC WLAN SYSTEM

Channel model	Parameter name	Initial level	Intermediate level	Final level
A	SNR (dB)	0-20	21-43	44-50
	PER (%)	100	91.66-32.35	28.94
B	SNR (dB)	0-24	25-35	36-50
	PER (%)	100	91.66-4.2	0
C	SNR (dB)	0-24	25-35	36-50
	PER (%)	100	91.66-1	0.8
D	SNR (dB)	0-19	20-27	28-50
	PER (%)	100	84.61-0.26	0.20
E	SNR (dB)	0-16	17-28	29-50
	PER (%)	100	84.61-6.3	5.8
F	SNR (dB)	0-18	19-33	34-50
	PER (%)	100	84.61-10.78	9.32

modulation scheme, more bandwidth and spatial streams, the performance can also be degraded drastically in an error-prone channel. We also investigated the performance of 802.11ac under different TGn channel models to find the performance patterns of different channel models.

REFERENCES

[1] *IEEE 802.11ac-Enhancements for Very High Throughput for operation in bands below 6 GHz*, IEEE P802.11ac/D5.0, 2013.

[2] Wi-Fi Alliance, *Wi-Fi certified products finder.*[Online]. Available:https://www.wi-fi.org/product-finder

[3] V. Jones, H. Sampath, "Emerging technologies for WLAN," *IEEE Communications Magazine*, vol. 53, no.3, pp. 141–149, March 2015.

[4] *IEEE 802.11n-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Enhancements for Higher Throughput,* IEEE 802.11n-2009.

[5] Z. Yunze, P.H. Pathak, and P. Mohapatra, "A first look at 802.11ac in action: Energy efficiency and interference characterization," *IFIP Networking 2014 Conf.*, 2-4 June 2014, pp. 1–9.

[6] E. H. Ong et al., "IEEE 802.11ac: Enhancements for very high throughput WLANs," *Proc. IEEE PIMRC*, Sep. 2011, pp. 849-853.

[7] C. Zhu et al., "Mac enhancements for downlink multi-user mimo transmission in next generation wlan," *IEEE Consumer Communications and Networking Conf. (CCNC),* Jan. 2012, pp. 832–837.

[8] Timo Vanhatupa, *Wi-Fi Capacity Analysis for 802.11 ac and 802.11 n: Theory and Practice,* Ekahau Inc., 2013.

[9] Oran Sharon, Yaron Alpert, "MAC level Throughput comparison: 802.11ac vs. 802.11n," *Physical Communication,* vol. 12, September 2014, pp. 33-49.

[10] Minyoung Park, "IEEE 802.11ac: Dynamic Bandwidth Channel Access,"in *IEEE International Conference on Communications (ICC) 2011*, pp.1-5, 5-9 June 2011.

[11] Gul Zameen Khan, Ruben Gonzalez, Eun-Chan Park, "A performance analysis of MAC and PHY layers in IEEE 802.11ac wireless network," in *18th International Conference on Advanced Communication Technology (ICACT)*, Pyeongchang Kwangwoon Do, South Korea, pp.20-25, 31 Jan - 3 Feb 2016.

[12] Aruba Networks, *802.11ac In-Depth.*[Online]. Available: http://www.arubanetworks.com/pdf/technology/whitepapers/WP_80211acInDepth.pdf

[13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function,"*IEEE J. Sel. Areas Commun.,* vol. 18, no.3, pp. 535–547, March 2000.

[14] Yong Soo Cho, et al., *MIMO-OFDM wireless communications with MATLAB,* John Wiley and Sons, 2010.

[15] G. Bianchi, *IEEE P802.11 Wireless LANs, TGn Channel Models,* IEEE 802.11-03/940r4, 2004-05-10.

[16] V.J. Rhodes, *Path loss proposal for the IEEE 802.11 HTSG channel model Ad Hoc group,* April 22, 2003.

[17] J.B. Andersen, T.S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communication channels," *IEEE Commun. Mag.,* Jan. 1995, pp. 42-49.

[18] L. Schumacher, K. I. Pedersen, and P.E. Mogensen, "From antenna spacings to theoretical capacities guidelines for simulating MIMO systems," in *Proc. PIMRC Conf.*, vol. 2, Sept. 2002, pp. 587-592.

[19] A. Perahia and R. Stacey, *Next Generation Wireless LANs - Throughput, Robustness, and Reliability in 802.11n*, Cambridge, 2008.

[20] A. Heck, *Introduction to Maple,* 3rd ed., New York: Springer-Verlag, 2003.

Gul Zameen Khan received a Bachelor degree in Computer Systems Engineering from University of Engineering and Technology Peshawar Pakistan in 2007 and a Master degree in Computer Engineering from Hanyang University South Korea in 2011. He is currently working as a researcher with the Intelligent Wireless Technology and Applications research group at Griffith University Australia..
He has worked as Lab Engineer in Ghulam Ishaq Khan Institute of Engineering Sciences and Technology Pakistan. In addition, he has worked as lecturer in Sarhad University of IT Peshawar Pakistan and COMSATS Institute Abbottabad Pakistan. He has also worked as a Research Assistant at Wireless Networks Lab, Dongguk University, South Korea. His areas of interest are MAC and PHY layers analysis of 802.11, multicast in Wi-Fi Direct, and wireless sensor networks.
Mr. Gul Zameen Khan is a member of IEEE and Institute for Integrated and Intelligent Systems, Griffith University Australia. He is also a professional member of Pakistan Engineering Council.

Dr. Ruben Gonzalez received a B.E. and PhD from the University of Technology, Sydney (UTS) and is currently a senior Lecturer in the School of ICT, at Griffith University Australia. He is also the director of the bachelor of computer science at Griffith University.
Previously he was founder and CTO of ActiveSky Inc, a wireless media technology company and has also held research positions at Wollongong University, OTC Ltd Research Labs and UTS. He has also held software engineering positions at various private enterprises.
Dr. Ruben Gonzalez is a member of the Institute for Integrated and Intelligent Systems, Griffith University Australia. He has over 80 refereed publications and a number of patents.

Prof. Eun-Chan Park received the B.S., M.S., and Ph.D degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea, in 1999, 2001, and 2006, respectively.
He worked for Samsung Electronics, Korea, as a senior engineer from 2006 to 2008. He is currently an Associate Professor in the Department of Information and Communication Engineering, Dongguk University-Seoul, Korea. He is also the head of the Wireless Networks Lab, at Dongguk University South Korea. His research interests include performance analysis, resource allocation, quality of service, congestion control, and cross-layer optimization in wired and wireless networks.
Dr. Eun-Chan Park is a member of IEEE Communications Society.

Dr. Xin-Wen Wu received the Ph.D. degree from the Chinese Academy of Sciences, Beijing, China.
He was with the Chinese Academy of Sciences; the University of California at San Diego, La Jolla, CA, USA, as a Post-Doctoral Researcher; and the University of Melbourne, Parkville, VIC, Australia, as a Research Fellow. He was with the faculty of School of Information Technology and Mathematical Science, University of Ballarat, VIC, Australia. He is currently with the faculty of School of Information and Communication Technology, Griffith University, Gold Coast, QLD, Australia.
Mr. Xin-Wen Wu is a senior member of IEEE. His research interests include cyber and data security, coding and information theory and their applications, communications and networks. He has co-authored over 70 research papers, book chapters and two books in the above-mentioned areas.

A Speculative Execution Strategy Based on Node Classification and Hierarchy Index Mechanism for Heterogeneous Hadoop Systems

Qi Liu*, Weidong Cai*, Jian Shen*, Zhangjie Fu**, Xiaodong Liu***, Nigel Linge****

*Nanjing University of Information Science and Technology, 219 Ningliu Road, Nanjing, Jiangsu, 210044, China

** Jiangsu Engineering Centre of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, Jiangsu, China

***School of Computing, Edinburgh Napier University, 10 Colinton Road, Edinburgh EH10 5DT, UK

****The University of Salford, Salford, Greater Manchester, M5 4WT, UK

qi.liu@nuist.edu.cn, caiweidongsuzhou@163.com, s_shenjian@126.com, wwwfzj@126.com, x.liu@napier.ac.uk,n.linge@salford.ac.uk

***Abstract*—MapReduce (MR) has been widely used to process distributed large data sets. MRV2 working on Yarn, as a more advanced programing model, has gained lots of concerns. Meanwhile, speculative execution is known as an approach for dealing with same problems by backing up those tasks running on a low performance machine to a higher one. In this paper, we have modified some pitfalls and taken heterogeneous environment into consideration. Besides, Node classification is used and a novel hierarchy index mechanism is created. We also have implemented it in Hadoop-2.6 and the strategy above is called Speculation-NC while optimized Hadoop is called Hadoop-NC. Experiment results show that our method can correctly backup a task, improve the performance of MRV2 and decrease the execution time and resource consumption compared with traditional strategies.**

***Keyword*—MapReduce; Speculative Execution; Time Prediction; Node Classification; Hierarchy Index Mechanism**

Manuscript received March 23, 2016. This work is a follow up of the invited journal of the accepted conference paper for the 18th International Conference on Advanced Communication Technology. This work is supported by the NSFC (61300238, 61300237, 61232016, U1405254, 61373133), Marie Curie Fellowship (701697-CAR-MSCA-IFEF-ST), Basic Research Programs (Natural Science Foundation) of Jiangsu Province (BK20131004), Scientific Support Program of Jiangsu Province (BE2012473) and the PAPD fund.

Q. Liu is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: qi.liu@nuist.edu.cn).

W. Cai is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (corresponding author to provide phone: +8615251708925; fax: +86-15251708925; e-mail: caiweidongsuzhou@163.com).

S. Jian is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: s_shenjian@126.com).

Z. Fu is with the Jiangsu Engineering Centre of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: wwwfzj@126.com).

X. Liu is with School of Computing, Edinburgh Napier University, 10 Colinton Road, Edinburgh EH10 5DT, UK (E-mail: x.liu@napier.ac.uk).

N. Linge is with the School of Computing, Science and Engineering, University of Salford, Salford, M5 4WT, UK (E-mail: n.linge@salford.ac.uk).

I. INTRODUCTION

THE time that data grow bigger and bigger is coming. The fast development of Cloud Computing makes that big data can be faster and more easily processed [1]. Nimbus provides a high quality of extensible architecture, which supports Xen and KVM virtual machine, and can be used as a PBS resource scheduler. OpenStack is an open source cloud computing platform designed by Rackspace and National Aeronautics and Space Administration (NASA) for public and private cloud. It is compatible with Amazon's EC2 and S3, and can provide the same service as the Amazon [2]. Sector and Spheres [3] is another open source platform designed implemented by Gu, which is used to process geographically dispersed large data sets in parallel. Hadoop cloud computing is one of the most popular open source cloud platforms, it is mainly supported by Yahoo, and applied to the Facebook and Amazon, Twitter, Baidu and other companies like IBM and the New York times. At present, in the latest Release version is 2.7.1. In 2010, Facebook announced that it has the largest Hadoop cluster. In 2011, its data storage space has reached 30 PB.

MapReduce, proposed by Google, which is one of the most important parts in Hadoop, has been the most popular distribute programming model in a cloud environment. With map and reduce procedures used in the cloud infrastructure transparently, those datasets can be processed easily and efficiently.

Many enterprises and companies obtain a large amount of business profits through analyzing and dealing with a lot of new data in time [4]. Data analysis application may have different complexity, resource requirement and data delivery deadlines. As a result, this diversity creates new requirement of job scheduling, workload management and program design. Although different users have different goals, a similar goal is to improve the performance of the framework. Several projects have been launched to relieve the difficulty in writing complex data analysis or data mining programs, e.g., Pig [5] and Hive [6] built above the MapReduce engines in the

Hadoop environment.

However, cloud systems suffer from poor load-scheduling strategies, which can consume much more execution time and temporary space than expected. While theoretically infinite computing resources can be provided in a cloud, unreasonable increment of mappers/reducers cannot achieve process efficiency, and may waste more storage to complete. Many optimization schemes have been proposed [7-13].

II. Related Work

General solutions on performance evaluation and load efficiency in a cloud system have been examined and presented. PQR2, an approach to accurate performance evaluation of distributed application in a cloud was presented [8]. Jing et al. presented a model that can predict resource consumption of MapReduce processes based on a classification and regression tree [9].

Mars were developed to run on NVIDIA GPUs, AMD GPUs as well as multicore CPUs and implemented in Hadoop. Mars hides the programming complexity of GPUs behind the simple and familiar MapReduce interface, and automatically manages task partitioning, data distribution, and parallelization on the processors. Six representative applications also have been implemented on Mars The experimental results show that integrating Mars into Hadoop enabled GPU acceleration for a network of PCs [10]. However, the implementation is much more complex while in a cluster, the performance of GPUs and CPUs.

PrIter, a distributed computing framework was developed. The prioritized execution of iterative computations is supported in it. PrIter either stores intermediate data in memory for fast convergence or stores intermediate data in files for scaling to larger data sets. PrIter achieves up to 50 × speedup over Hadoop for a series of iterative algorithms. In addition, PrIter is shown better performance for iterative computations than other state-of-the-art distributed frameworks such as Spark and Piccolo [11]. But, the biggest pitfall is that it only adapts to iterative algorithms and cannot be applied to all algorithms.

Besides the above methods, some researchers are studying optimizing the speculative execution strategy in MapReduce. Zaharia et al. [12] proposed a modified version of speculative execution called Longest Approximate Time to End (LATE) algorithm that uses a different metric to schedule tasks for speculative execution. Instead of considering the progress made by a task so far, they compute the estimated time remaining, which gives a more clear assessment of a straggling tasks' impact on the overall job response time. But the time every stage occupies is not stable and the std representing standard deviation used in LATE cannot represent all cases. To solve the disadvantages in LATE, MCP [13] was proposed by QI CHEN et al. MCP uses average progress rate to identify slow tasks while in reality the progress rate can be unstable and misleading. Straggles can be appropriately judged when there is data skew among the tasks. However, there are still a lot of pitfalls in MCP. Moreover, it can only be used in MR, not including MRV2.

All research works above have devoted themselves to particular and/or comprehensive load and usage efficiency in a distributed environment. However, in a Heterogeneous environment, reasonable scheduling is still a hard problem. In LATE, MCP or the newest speculative execution strategy in Hadoop-2.6, the biggest pitfall is that situation of heterogeneous environment is not well concerned.

III. Our Strategy

A. Remaining Time Estimation of Current Task

Remaining time of the current task can be calculated by adding the remaining time of current phase and following phases. Detailed method is shown in (1) (2) and (3). $factor_d$ represents the volume of current input data and average value of historical tasks. where T_{rem} represents the remaining time of current task, which consists of the remaining time of current phase and following phases, marked as T_{cp} and T_{fp} . Avg_p includes the average consumption of shuffle phase, sort phase and reduce phase on some node. While the type of current task is map task, the remaining time only includes map phase.

$$T_{rem} = T_{cp} + T_{fp} \tag{1}$$

$$T_{rem} = \frac{rem\_data_{cp}}{bandwidth_{cp}} + \sum_{p}^{in}{}_{fp} Avg_p * factor_p \tag{2}$$

$$factor_d = \frac{data_{input}}{data_{avg}} \tag{3}$$

While there is no historical information on the node, global historical data is used to calculate the average finishing time of current phase

B. Time Prediction of a Backup Task

When a task finishes, our strategy automatically stores task running time on some node. If current task is a map task, stored data would be hostname and average duration. Otherwise, while it is a reduce task, duration of per-phase would be written down and the duration of whole phase would be recorded at the same time. We have established a storage mechanism to help us store information while having a low space occupation. The detailed storage method is shown in Fig.1.

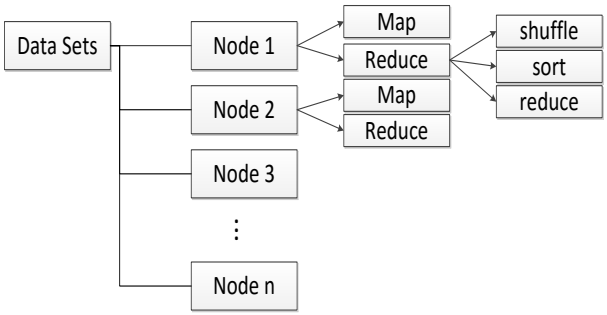


Fig.1 Hierarchy Index Mechanism

As show in Fig.1, a data set is found according to hostname at the beginning of storage procedure. In a same data set, data are collected from the task running on the same host. Every storage area is further divided in two sub storage areas called Map and Reduce. If the task finished just now is map type, the running time is directly recorded in Map. Moreover, while it is a reduce task, detailed running information, containing shuffle time, sort time and reduce time, is respectively written

in 3 sub blocks. Finally, summary information of last level (shuffle sort or reduce) is stored in its up level: Reduce. We need to search some data, a binary search is started based on index mechanism.

While detailed average running time of a task on some node needs to be known (such as $Avg_{shuffle}$, Avg_{sort} or Avg_{reduce}), hostname is first gotten. If a data set hostname mapping to is empty, overall running time of different hosts would replace the detailed one. Otherwise, average time of some phase is directly obtained. Detailed algorithm is shown in Algorithm 1.

Algorithm 1 Get the estimated running time of a backup task

Input: TaskId Tid **,NodeId** Nid **,TaskType** $Type$

Output: Running time of a backup task T_{back}

1 Get hostname according to Tid of current task
2 Get the data set that hostname mapping to
3 If the data set is empty
4 If $Type$ equals Map
5 Get the average running time of different hosts, Recorded as Avg_{map}, $T_{back} = Avg_{map}$
6 Else if $Type$ equals Reduce
7 Get the per phase average running time of different hosts, Recorded as $Avg_{shuffle}$ Avg_{sort} and Avg_{reduce}
8 $T_{back} = Avg_{shuffle} + Avg_{sort} + Avg_{reduce}$
9 End If
10 Else
11 If $Type$ equals Map
Get the average running time of the same host as Avg_{map} and $T_{back} = Avg_{map}$
13 Else
14 Get sum of per phase average running time of same host, Recorded as T_{back}
15 End If
16 End If
17 Return T_{back}

C. Node Selection and Task selection

Node selection is import for that data-local would help framework execute job faster. Besides, node to be selected should have relatively good performance at that time. The object of task selection is judging whether a task runs on a poor performance node. Also, in our strategy, the condition of start a back task is continuously changing according to (4). That is to say, if current progress is 50% and T_{rem} equals 15 seconds, T_{back} must smaller than 10, then the backup task would start. Once a backup task starts, we kill the native task immediately.

$$\frac{T_{rem}}{T_{back}} > Condition = 1 + TaskProgress \qquad (4)$$

IV. Experiment And Analysis

In order to test the performance and benefits of our load balancing strategy, we simulate a real environment. Test-bed is made up of a personal computers and a server. The server is equipped with 288 GB of memory, a 10 TB SATA driver. The personal computers is equipped with 12GB of memory, a single 500GB disk and four Intel(R) Core(TM) 2.4GHz two-core processors. On the server, we run eight virtual machines and each virtual machine is given different amounts of memory and different number of processors. The detailed information is shown in Table I in Section IV.

Then, the virtual machines run as the data nodes and the server run as the name node. To evaluate the performance of load strategy, we use WordCount, Sort and Grep. They are common applications used in MRV2 to evaluate strategy. The Purdue Benchmarks Suite provides us with this application workload and we use the free datasets [14] as the workloads input. Detailed inputs of these applications are shown in Table II.

TABLE I
THE DETAILED INFORMATION OF EACH VIRTUAL MACHINE

NodeId	*Memory(GB)*	*Core processors*
Node1	10	8
Node2	8	4
Node3	8	1
Node4	8	8
Node5	4	8
Node6	4	4
Node7	18	4
Node8	12	8

TABLE II
INPUT OF THESE APPLICATIONS

	WordCount	**Sort**	**Grep**
Input(GB)	50	30	30
Number of Mappers	200	200	200
Number of Reducers	16	15	15

The evaluation method is this paper can be expressed as the Eq. (5), which represents the different between another strategy and our Hadoop-NC divided by another strategy.

$$Improvement = \frac{OtherStrategy - NC}{OtherStrategy} \qquad (5)$$

A. Evaluation of Node Classification

To prove Node of classification is reasonable, we first ran WordCount for 5 times under the circumstance that speculative execution is disable. Then we collected the running information of map phase, and average consuming time was calculated.

Fig.2 shows average consuming time of each node to finish a task. Difference is obvious and tasks running on node3 consume much longer time than others. This proves that it is unreasonable to use mean consuming time to calculate.

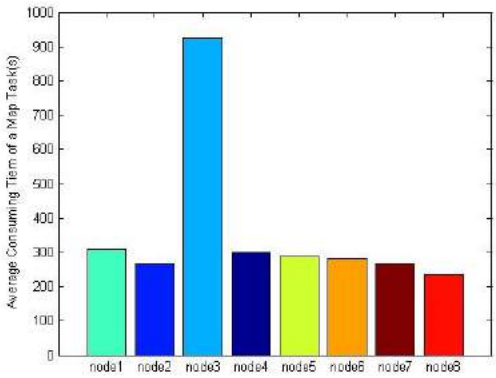


Fig.2 Average Consuming Time of Each Node

Fig.3 shows detailed running time of tasks on node1 and node8. This picture is depicted by 24 groups of sample s from all samples. Although lots of peaks in the figure, if we replace these data with mean value, these 2 lines would be relatively smooth. Those points have been smoothed can be seen as tasks generated by soft straggler, as shown in Fig.4.

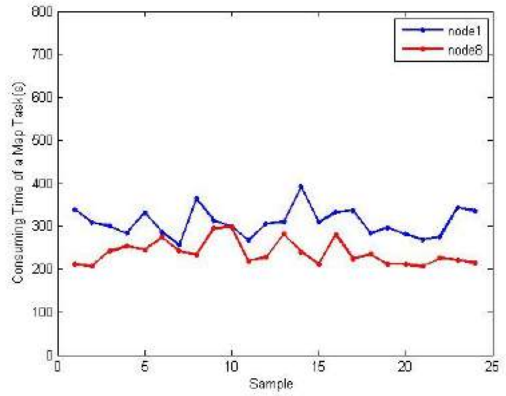


Fig.3 Detailed Running Time of Tasks on node1 and node8

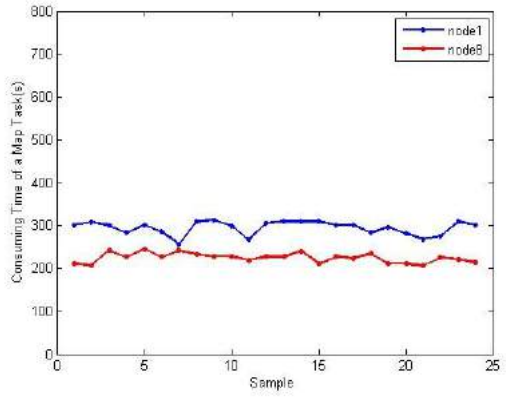


Fig.4 Data After Smoothing

Concluded from Fig.3 and Fig.4, peaks are in Fig.3 can been seen as software straggles. And in figure.4, we assume that this the straggles have been processed and these tasks have been transferred in other nodes. Then, the tendency of time that tasks running on the same node is relative stable. So, we our method classify the time according to node is reasonable. Based on these, further experiments were operated, we used WordCount and Sort to evaluate our performance, which can respectively represent 2 types of application that one consumes CPU and the other is more sensitive to Memory.

B. Evaluation of Hierarchy Index Mechanism

In this part, we will evaluate it from the prosperity of time complexity.

If we put all running information into a same list, then, when we search the list for a piece of data, the time complexity of search would be O(k) (k represents the number of all pieces of data). However, in our storage mechanism, if we successfully find data, average search time can be can simplified as $\log_2(n)+k/n$ (n is the number of nodes). So the average time complexity can be simplified as $O(\log_2(n))$.

Obviously, our hierarchy index mechanism can get a better performance, which has a smaller time complexity.

C. Overall Evaluation of Performance

Fig.5 shows for WordCount case, NC finishes jobs 12% faster than Hadoop-Original and 20% faster than Hadoop-None on average.

Fig.6 shows that on average, NC finishes jobs 3% faster than Hadoop-Original and 19% faster than Hadoop-None. It is very terrible that Original speculative execution strategy has poor performance, but our Hadoop-NC still can reduce the execution time.

Fig.7 shows for Grep case, NC finishes jobs 18% faster than Hadoop-Original and 10% faster than Hadoop-None on average.

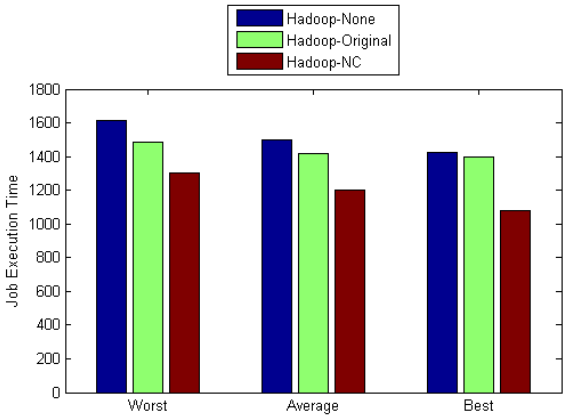


Fig.5 Job Execution Time of WordCount

To analyze the reason why Original strategy has worse performance, we calculated the running information of these two applications under different strategies and drew a table.

As shown in Table III. For WordCount, the backup success rate of Hadoop-NC is 20% and 16.7% higher than Hadoop-Original for map and reduce tasks. For Sort, the backup success rate of Hadoop-NC is 24.3% and 30.7% higher than Hadoop-Original. For Grep, the backup success rate of Hadoop-NC is 30% higher than Hadoop-Original.

Also, resource consumption in cloud environment is also an important indicator to evaluate the performance of PaaS platform. Traditional speculative strategies only evaluate the performance from the prosperity of job execution time. However, resource consumption is usually ignored. In this paper, resource consumption is represented the number of containers and the time of the container occupied. So an Eq. is gotten as shown in (6).

$$Consumption = Containers * Seconds \quad (6)$$

TABLE III
DETAILED COMPARISON

Workloads	Strategy	Sum of Backed-up Map Tasks	Sum of Successful Backed-up Map Tasks	Backup success rate (%)
WordCount	Hadoop-Original	85	40	47.1
	Hadoop-NC	55	38	69.1
Sort	Hadoop-Original	84	24	28.6
	Hadoop-NC	70	37	52.9
Grep	Hadoop-Original	68	32	47.1
	Hadoop-NC	35	27	77.1

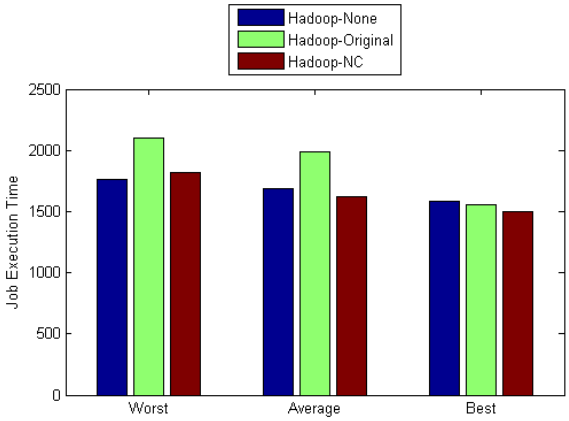


Fig.6 Job Execution Time of Sort

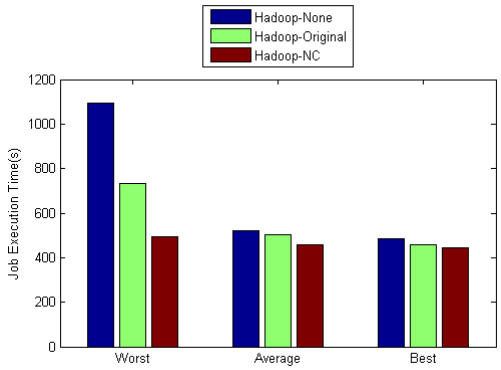


Fig.7 Job Execution Time of Grep

When it comes to resource consumption, shown in Fig.8, our strategy has an obvious improvement for WordCount samples, it save about 10% and 13% than Hadoop-None and Hadoop-Original. However, for Sort samples, although our strategy has an improvement over Hadoop-Original, more resource than Hadoop-None is consumed. It is very normal that if the speculative strategy cannot find straggle on time. For Gerp, because it consumes little time and the saved resource is not very obvious, but our strategy still has an improvement compared with Hadoop-None and Hadoop-Original.

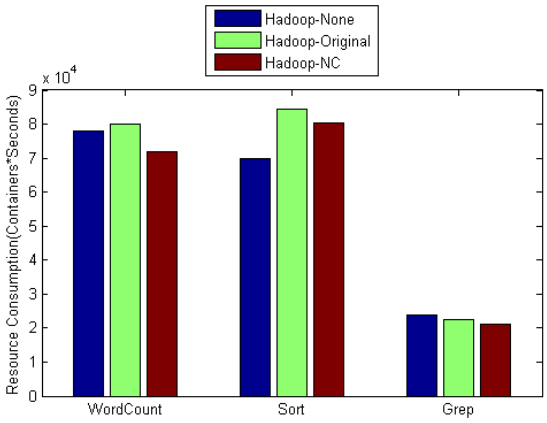


Fig.8 Resource consumption

I. CONCLUSION

In this paper, a new strategy called Speculation-NC is introduced and implemented in Hadoop-2.6. Experiment results have shown that our method can relatively save time and resource for WordCount sample. However, there is still much work can be done to improve the performance of MRV2 for Sort sample further, both in execution and resource consumption.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski and M. Zaharia, " A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50-58, 2010.

[2] X. Wen, G. Gu, Q. Li, Y. Gao and X. Zhang, "Comparison of open-source cloud management platforms: OpenStack and OpenNebula," *In Proceedings of the 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pp. 2457-2461, 2012.

[3] Y. Gu and L. R. Grossman, "Sector and Sphere: the design and implementation of a high-performance data cloud," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1897, pp. 2429-2445, 2009.

[4] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," *Communications of the ACM*, vol. 51, no.1, pp. 107-113, 2008.

[5] Apache pig. <http://pig.apache.org/> [accessed on 14.09.15].

[6] Apache hive. <https://hive.apache.org/> [accessed on 14.09.15].

[7] Z. Fu, X. Sun, Q. Liu, L. Zhou and J. Shu, "Achieving Efficient Cloud Search Services: Multi-keyword Ranked Search over Encrypted Cloud Data Supporting Parallel Computing," *IEICE Transactions on Communications*, vol. 98, no. 1, pp. 190-200, 2015.

[8] A. Matsunaga and J. Fortes, "On the use of machine learning to predict the time and resources consumed by applications," *Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, pp. 495-504, 2010.

[9] W. Fang, B. He, Q. Luo and N. K. Govindaraju, "Mars: accelerating mapreduce with graphics processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 4, pp. 608-620, 2010.
[10] Y. Zhang, Q. Gao, L. Gao and C. Wang, "Priter: a distributed framework for prioritizing iterative computations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1884-1893, 2014.
[11] Piao, J. Tai and J. Yan, "Computing resource prediction for mapreduce applications using decision tree," *In Web Technologies and Applications*, pp. 570-577, 2012.
[12] M. Zaharia, A. Konwinski, A. Joseph, R. Katz and I. Stoica, "Improving Mapreduce Performance in Heterogeneous Environments," *OSDI*, vol. 8, no. 4, 2008.
[13] C. Qi, C. Liu and Z. Xiao, "Improving MapReduce performance using smart speculative execution strategy," *IEEE Transactions on Computers*, vol. 63, no. 4, pp. 954-967, 2014.
[14] F. Ahmad, S. Chakradhar, A. Raghunathan and T. Vijaykumar, "Tarazu: optimizing MapReduce on heterogeneous clusters," *ACM SIGARCH Computer Architecture News*, pp.61-74, 2012.

Qi Liu (M'11) received his BSc degree in Computer Science and Technology from Zhuzhou Institute of Technology, China in 2003, and his MSc and PhD in Data Telecommunications and Networks from the University of Salford, UK in 2006 and 2010. His research interests include context awareness, data communication in MANET and WSN, and smart grid. His recent research work focuses on intelligent agriculture and meteorological observation systems based on WSN.

Weidong Cai received her bachelor's degree in Software Engineering from Nanjing University of Information Science and Technology in 2014, and he is pursuing a master's degree in software engineering at the Nanjing University of Information Science and Technology. Hi research interests include Cloud Computing, Distributed Computing and Data Mining.

Jian Shen received his bachelor's degree in Electronic Science and Technology Specialty from Nanjing University of Information, Science and Technology in 2007, and he received his masters and PhD in Information and communication from CHOSUN University, South Korean in 2009 and 2012. His research interests includes Computer network security, information security, mobile computing and network, wireless ad hoc network.

Zhangjie Fu received his BS in education technology from Xinyang Normal University, China, in 2006; received his MS in education technology from the College of Physics and Microelectronics Science, Hunan University, China, in 2008; obtained his PhD in computer science from the College of Computer, Hunan University, China, in 2012. Currently, he works as an assistant professor in College of Computer and Software, Nanjing University of Information Science and Technology, China. His research interests include cloud computing, digital forensics, network and information security.

Xiaodong Liu received his PhD in Computer Science from De Montfort University and joined Napier in 1999. He is a Reader and is currently leading the Software Systems research group in the IIDI, Edinburgh Napier University. He was the director of Centre for Information & Software Systems. He is an active researcher in software engineering with internationally excellent reputation and leading expertise in context-aware adaptive services, service evolution, mobile clouds, pervasive computing, software reuse, and green software engineering. He has meanwhile a successful track record of teaching in a number of software engineering courses which are widely informed by his research activities.

Nigel Linge received his BSc degree in Electronics from the University of Salford, UK in 1983, and his PhD in Computer Networks from the University of Salford, UK, in 1987. He was promoted to Professor of Telecommunications at the University of Salford, UK in 1997. His research interests include location based and context aware information systems, protocols, mobile systems and applications of networking technology in areas such as energy and building monitoring.

A Comparison Study of the Quick Removing (QR) Approach (HSR Mode X) under Cut-Through and Store and Forward Switching Modes

Saad Allawi NSAIF *, Semog KIM*, Jong Myung RHEE*

*Dept. of Information and Communications Engineering, Myongji University, Yongin-si, South Korea

saad.allawi1@gmail.com, kimsemog@empas.com, jmr77@mju.ac.kr

***Abstract*—We previously introduced a quick removing (QR) approach (currently part of the IEC 62439-3 standard under the terminology of "HSR Mode X") to improve the high-availability seamless redundancy (HSR) protocol's traffic performance. The idea of the QR approach is to remove the duplicated frame copies from the network when all nodes have received one copy of the sent frame and begin to receive the second copy. Therefore, forwarding the frame copies until they reach the source node, as occurs in a standard HSR, is not needed in a QR. In our previous paper, we applied the store and forward switching mode, whereas in this paper, we present the performance of the QR approach using a cut-through switching mode. The performance analysis showed a reduction percentage in frame latency that reaches about 49% compared to the store and forward switching mode. Consequently, this will improve network performance, free more bandwidth, and quickly deliver sent frames to their required destinations, which is a firm condition in many industrial and automation applications.**

***Keyword*— HSR protocol, QR approach, HSR Mode X, HSR traffic performance, cut-through, seamless redundancy, IEC 62439-3.**

I. INTRODUCTION

MODERN control and time-critical systems typically require seamless communication networks. This means that some form of fault tolerance in the communications network is required to achieve satisfactory system availability. High availability, which can be achieved through redundancy and fault tolerance, can prevent a system from momentary interruptions that may be considered inconvenient and may cause critical stoppage in some industrial applications.

Generally, seamless communication with fault tolerance is one of the key requirements for Ethernet-based mission-critical systems, such as substation automation system (SAS) networks. A fault-tolerant Ethernet (FTE) eliminates the single point of failure and therefore improves the overall system availability [1]. Since the standard Ethernet does not provide a fault-tolerance capability, various FTE protocols for power applications have been developed [2]. Among these, only two protocols, the parallel redundancy protocol (PRP) and the high-availability seamless redundancy (HSR) protocol, which is standardized as IEC 62439-3, are suitable for seamless communication [2-4]. Both HSR and PRP are based on the same principle of active redundancy by duplicating the information frames, therefore resulting in a zero-delay reconfiguration in the event of a switch or link failure; however, only the HSR protocol is addressed in this paper because it is generally accepted as the developed version of PRP. The HSR protocol is a redundancy protocol for Ethernet-based networks, and it provides duplicated frame copies for each frame sent. In other words, the HSR protocol provides two frame copies for the destination node, one from each side, enabling zero-fault recovery time in case one of the frame copies is lost. This means that even in the case of a node failure or a link failure, there is no stoppage of network operations. If both sent copies reach the destination node, the node will take the fastest copy and discard the other copy. This feature of the HSR protocol makes it very useful for time-critical and mission-critical systems.

Generally, the HSR-based network has four types of nodes [5]:

- A doubly-attached node for HSR (DANH), which has two HSR ports sharing the same medium access control (MAC) and Internet protocol (IP) set addresses. This allows the address management protocols, such as the address resolution protocol (ARP), to operate as usual without modification, which simplifies network engineering. Each DANH node will duplicate a non-HSR frame that is generated at the upper layer into two frame copies. It will then append an

Manuscript received April 28, 2016. This work was supported by the Technology Innovation Program (No.10058109) and was funded by the Ministry of Trade, Industry, and Energy (MI, Korea). It was also supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) and was funded by the Ministry of Education (No. NRF-2015R1D1A1A02059506).

Saad Allawi NSAIF is with the Information and Communications Engineering department, Myongji University, yongin, Gyeonggi-do 449-728, South Korea. (Phone/fax: +820-31-330-6824; e-mail: saad.allawi1@gmail.com)

Semog KIM is with the Information and Communications Engineering department, Myongji University, yongin, Gyeonggi-do 449-728, South Korea. (Phone/fax: +820-31-330-6824; e-mail: kimsemog@empas.com)

Jong Myung RHEE is with the Information and Communications Engineering department, Myongji University, yongin, Gyeonggi-do 449-728, South Korea. (Corresponding author, Phone:+820-31-330-6886; fax: +820-31-330-6824 ;e-mail: jmr77@mju.ac.kr)

HSR tag to each copy and send two copies out through DANH ports, one in a clockwise direction and the other in a counter-clockwise direction;

- A single-attached node (SAN), which is a non-HSR device, such as a commercial printer, server, or laptop. It cannot be directly inserted into the HSR networks because it lacks the forwarding capability of an HSR node and does not support the HSR tag. They must be attached through a redundancy box (RedBox) node type;
- A RedBox node, which has three ports. Two ports are HSR ports, and the third one is an Ethernet port that any SAN device, such as a PC, can be plugged into to be engaged with the HSR network. The RedBox node forwards the frames over the HSR network like any other HSR node and acts as a proxy for all SANs that access it. Thus, it must keep track of all traffic on behalf of the SANs. The RedBox can also act as a switch for the SANs. Therefore, it is somewhat more complex than HSR nodes;
- A quadruple port device (QuadBox node), which has four HSR ports. Each pair shares the same MAC and IP addresses. This type of node is used to connect two rings or networks. The QuadBox node removes duplication and performs additional tasks, such as multicast and virtual local area network (VLAN) filtering.

The primary drawback of the HSR protocol is the unnecessary traffic due to the duplicated frames that are generated and circulated inside the network. If a typical HSR is applied, traffic will be doubled at each QuadBox node type. The situation becomes worse when the HSR is used for a generic object-oriented substation event (GOOSE) and sampled values (SV) messages' transmissions, which generate large amounts of traffic. This downside will degrade network performance and may cause network congestion or even stoppage. Therefore, data traffic control is essential for maximizing network operational performance [6]; however, due to the potential advantages of HSR, several works have also focused on implementing the HSR [7-9] and improving its performance by reducing its unnecessary redundant traffic [10-15]. To address this issue, we have proposed an HSR approach called quick removing (QR) [10] to remove the unnecessary redundant traffic from the network instead of allowing it to circulate inside the network and consume its bandwidth. The idea of the QR approach was to remove the duplicated frame copies from the network when all nodes have received one copy of the sent frame and began to receive the second copy. Due to the high efficiency of our QR approach, other papers and research studies have referred and reported on the QR approach, such as [16,17]. Consequently, the International Electrotechnical Commission (IEC) has included the QR approach in the newest edition of the IEC 62439-3 standard, Ed. 03- 2016, under the terminology of HSR mode X.

Nevertheless, in [10], we only considered the store and forward switching mode in which the entire frame needs to be stored in the ingress of the node before the node can forward it. Obviously, the node will require more time to forward the frame because the CRC code must be calculated and compared with the code appended to the frame to ensure that the received frame is error-free before sending it to the upper layers or forwarding it to the next nodes. In this case, as soon as the receiving node realized that it is the destination node of the received frame, the receiving node will wait to receive and read the source MAC address of the received frame. Then, the node will check its received frames table to see whether it has already received a copy from that frame. If the node does not find the source MAC address in its table, then it will buffer all received bytes until the reception of all is complete, and then, calculate the CRC code and compare it to the frame's CRC. If they are equal, the node will send that frame to the upper layers and update its received frames table; otherwise, it will delete it.

If the node does find the received source MAC address in its received frames table, then it will clearly, under this type of switching node, the node will require less time to forward the frame compared to the time used for the store and forward type. In other words, the cut-through mode will be much faster than the store and forward mode [18].

In this paper, we discuss the HSR node's behavior under the QR approach using the cut-through switching mode. Previously, we introduced this in [19], but in this paper, we add more details to the performance analysis of the QR approach under this type of switching mode.

The remainder of the paper is organized as follows. In Section II, we introduce our QR approach behavior under the cut-through switching mode. In Section III, we describe the analytical performance analysis of the QR approach under the cut-through switching mode. Section IV shows the results of the simulation analysis. Finally, in Section V, we summarize our work and present our conclusions.

II. QR Approach under The Cut-through Switching Mode

QR is an approach used to improve the traffic performance of the standard HSR protocol. It can be applied to any closed-loop network topology for all traffic types, especially the multi/broadcast type. The QR approach aims to remove the redundant frame copies from the network when all nodes have received one copy of the sent frame and begin to receive the second copy; however, the QR approach will only be applied to the data HSR frame types. Exempt from that the HSR supervisory frame type to allow the network nodes to check the network status of redundancy. The HSR node will recognize whether the received frame is a supervisory or a data frame through the bit pattern of the 4-bit path field of the received frame. The bit pattern of the supervisory frame type is 1111, whereas the data HSR frames have a bit pattern in the range of 0001-1001.

The QR approach does not need to use any special control frames; however, in [10], we discussed our QR approach using the store and forward switching mode only, which showed that the QR approach can reduce the unnecessary redundant network traffic in HSR networks up to 50%, as illustrated in Fig. 1.

In this paper, the application of the cut-through switching mode in the QR approach is described, which provides faster transmission and frees more bandwidth. Under the store and forward switching mode, the QR nodes wait to receive all of the frame's bytes and to calculate the CRC code to ensure that the received frame is error-free before deciding to send it to

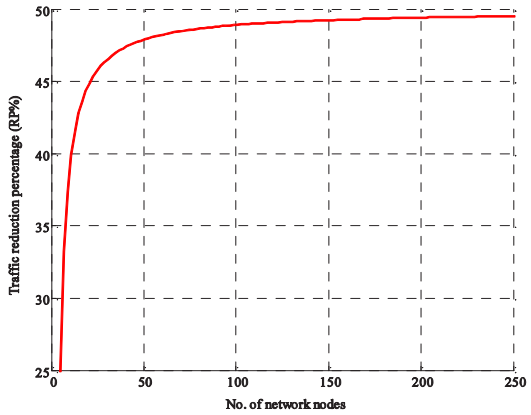


Fig. 1 Traffic reduction percentage (RP) of QR to the standard HSR protocol with respect to the number of nodes in a single ring.

the upper layers or forward it to the next nodes; however, under the cut-through switching mode, when a node receives a frame, it reads the first 6 bytes, or the destination MAC address. Thereafter, the node takes the following actions according to the type of destination MAC address.

A. Unicast MAC Type (The Receiving Node is the Destination Node for The Received Frame)

In this case, as soon as the receiving node realized that it is the destination node of the received frame, the receiving node will wait to receive and read the source MAC address of the received frame. Then, the node will check its received frames table to see whether it has already received a copy from that frame. If the node does not find the source MAC address in its table, then it will:

1) Buffer all received bytes until the reception of all is complete, and then,
2) Calculate the CRC code and compare it to the frame's CRC. If they are equal, the node will send that frame to the upper layers and update its received frames table; otherwise, it will delete it.

If the node does find the received source MAC address in its received frames table, then it will:

3) Wait to receive and read the HSR frame's header to check the sequence number if it is in the received frames table. If it is, then the node will delete the bytes that have been received thus far and all remaining bytes that will be received sequentially; otherwise, it will:
4) Buffer all received bytes until the reception of all is complete, and then,
5) Calculate the CRC code and compare it to the frame's CRC. If they are equal, the node will send that frame to the upper layers and update its received frames table; otherwise, it will delete it

B. Unicast MAC Type (The Received Frame is Heading to other Node)

If the received frame is a unicast type and is heading to a node other than the receiving node, or in other words, the frame is passing through the receiving node and going to another node, the receiving node will:

1) Look for that destination MAC address of the received frame in its forwarded frames table. If the address is not in the table, then the node will sequentially forward all received bytes into the other node's ports and then update its forwarded frames table.
2) If the destination MAC address is in the table, the node will wait to receive and read the source MAC address of the received frame and look up the table again for that pair of addresses. If the address pair does not exist, then the node will sequentially forward all received bytes into the other node's ports and then update its forwarded frames table; however, if that address pair is in the table, the node will wait to receive and read the HSR header of the frame to see whether it has already forwarded the same frame. If the node finds that it has forwarded a frame with the same destination, source MAC address, and HSR header, then the node will delete the received bytes and all remaining bytes of that frame as soon as they reach the node. Otherwise, the node will sequentially forward all received bytes into the other node's ports and then update its forwarded frames table.

C. Multi/broadcast MAC Type

The receiving node will wait to receive and read the source MAC address to determine whether the node has already received a copy of that frame. If the source node of that frame is not in the received frames table, then the node will:

1) Make a copy of each received byte, buffer it, and then forward the original bytes into the other ports.
2) When all bytes of that frame are received, the node will calculate the CRC code and compare it to the frame's CRC code. If they are equal, the node will send that frame to the upper layers and update its received frames table. Otherwise, it will delete it.

If the source MAC address is shown in the received frames table, the node will wait to receive and read the HSR frame's header. If the node shows that it has already received a copy of that frame, then the node will delete the received bytes and stop making a copy of each received byte of that frame; but it will only forward these bytes into the other ports if the node has not did that for that received frame.

If the source MAC address is not shown in the received frames table, then the node will:

1) Make a copy of each received remaining byte of that received frame, buffer it, and then forward the original bytes into the other ports.
2) When all bytes of that frame are received, the node will calculate the CRC code and compare it to the frame's CRC code. If they are equal, the node will send that frame to the upper layers and update its received frames table. Otherwise, it will delete it.

Fig. 2 includes a flow chart that summarizes the QR steps under the cut-through switching mode.

III. Analytical Performance Analysis

In this section, we will demonstrate the performance of the QR approach under the cut-through switching mode compared to the store and forward mode from the point of view of the frame latency using the analytical approach. Let us consider the network shown in Fig. 3 as an example for our performance analysis; however, assume that node C has sent a unicast frame to node G. The following expression represents the latency of the frame from the time at which the sending

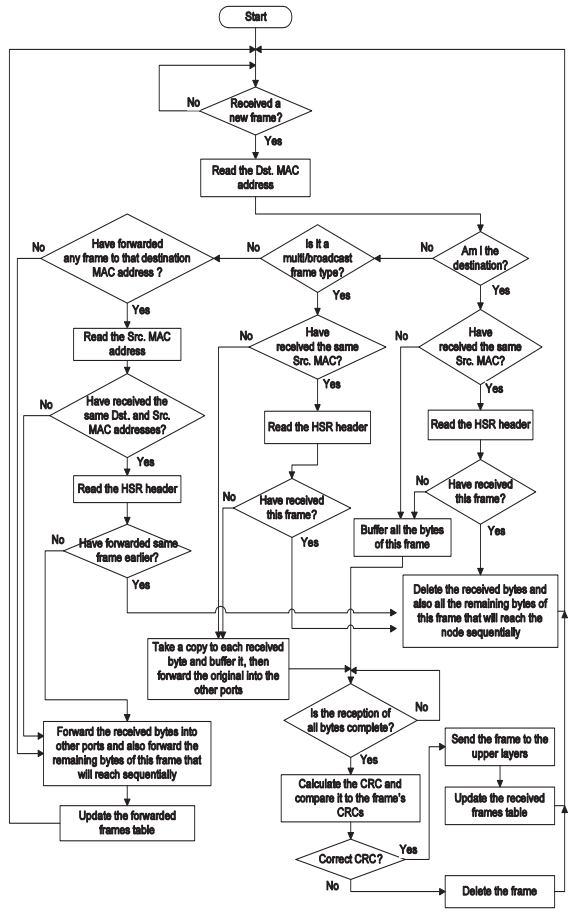


Fig. 2 Flow chart of the HSR node under the operation of the QR approach using the cut-through switching mode.

node sent its bits until the destination node has received all of the bits:

$$t_c = \tau_r + \tau_p + \tau_n$$

where t_c is the frame latency under the cut-through switching mode, τ_r is the transmission delay time, τ_p is the propagation delay time, which is assumed to be zero because the network example has short link lengths, and τ_n is the node processing rate. Assume the queuing delay time is zero.

The above expression can be written as:

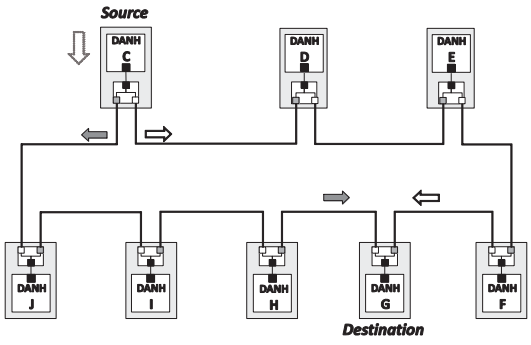


Fig. 3 An HSR example network for analytical analysis.

$$t_c = l\left(\frac{f}{\beta}\right) + \left[(n-2)\left(\frac{h}{\rho}\right) + \left(\frac{f}{\rho}\right)\right] \quad (1)$$

where l is the number of network links, f is the frame size in bits, β is link capacity in bits per second, n is the number of nodes in the path of the sent frame, h is the frame header size in bits, and ρ is the node processing rate. h might be 48 bits when the received node only reads the destination MAC address of the received frame, or it might be 96 bits or even 144 bits if the destination MAC, the source MAC, and the HSR frame header are read.

The term $(n-2)$ represents the exclusion of the source and destination nodes from the calculation of the delay time on the path of the sent frame, whereas $\left(\frac{f}{\rho}\right)$ represents the delay time of the destination node.

On the other hand, the latency under the store and forward switching mode can be determined as follows:

$$t_{sf} = l\left(\frac{f}{\beta}\right) + \left[(n-1)\left(\frac{f}{\rho}\right)\right] \quad (2)$$

$(n-2)$ represents the exclusion of the source node from the calculation of the delay time on the path of the sent frame.

Assume that the frame size that node C has sent is 1522 bytes, each of the network links' capacity is 100Mb/s, and the node processing rate is 100Mb/s. Thereafter, the latency under both the cut-through and store and forward switching modes will be determined using (1) and (2) as follows:

$$t_c = 4\left(\frac{1522\times 8}{100\times 10^6}\right) + \left[(5-2)\left(\frac{48}{100\times 10^6}\right) + \left(\frac{1522\times 8}{100\times 10^6}\right)\right]$$

$$= 0.610 \ ms$$

Note that $h = 48$ bits, which is the size of the destination MAC address of the sent frame, because each node can recognize that the frame is a unicast frame and is sent to a unique address. Therefore, the nodes do not need to read additional bits from the frame.

$$t_{sf} = 4\left(\frac{1522\times 8}{100\times 10^6}\right) + \left[(5-1)\left(\frac{1522\times 8}{100\times 10^6}\right)\right] = 0.974\, ms$$

Let us find the latency reduction percentage with respect to the network size, or in other words, with respect to the number of network nodes. This can be represented in the following expression:

$$\lim_{n\to\infty} \varepsilon = \lim_{n\to\infty}\left(1 - \frac{t_c}{t_{sf}}\right)\times 100\%$$

where ε is the latency reduction percentage of the cut-through mode compared to the store and forward mode.

$$\lim_{n\to\infty} \varepsilon = \lim_{n\to\infty}\left(1 - \frac{l\left(\frac{f}{\beta}\right) + \left[(n-2)\left(\frac{h}{\rho}\right) + \left(\frac{f}{\rho}\right)\right]}{l\left(\frac{f}{\beta}\right) + \left[(n-1)\left(\frac{f}{\rho}\right)\right]}\right)\times 100\%$$

Assume that $\beta = \rho$. In Fig. 3, the path from nodes C to G has a number of links equal to the number of nodes –1, or it can be said that $l = n-1$.

Thus, we can rewrite the above expression as follows:

$$\lim_{n\to\infty} \varepsilon = \lim_{n\to\infty} \left(1 - \frac{n\left(\frac{f}{\rho}\right) - \left(\frac{f}{\rho}\right) + \left[(n-2)\left(\frac{h}{\rho}\right) + \left(\frac{f}{\rho}\right) \right]}{n\left(\frac{f}{\rho}\right) - \left(\frac{f}{\rho}\right) + \left[(n-1)\left(\frac{f}{\rho}\right) \right]} \right) \times 100\%$$

$$\lim_{n\to\infty} \varepsilon = \lim_{n\to\infty} \left(1 - \frac{nf + nh - 2h}{2nf - 2f} \right) \times 100\%$$

$$\lim_{n\to\infty} \varepsilon = \left(\lim_{n\to\infty} 1 - \lim_{n\to\infty} \left(\frac{n\left(f + h - 2\frac{h}{n} \right)}{2n\left(f - \frac{f}{n} \right)} \right) \right) \times 100\%$$

$$\lim_{n\to\infty} \varepsilon = \lim_{n\to\infty} \left(1 - \frac{f+h}{2f} \right) \times 100\% \quad (3)$$

It is obvious from (3) that ε is not affected by the number of nodes when this number becomes large; on the contrary, it varies inversely with respect to the frame size. Fig. 4 shows that ε increases when the frame size increases, and the maximum reduction percentage that can be achieved is 49.4% when $h = 144$ and $f = 1522$ bytes; however, the lowest ε value is about 35% when the frame size is 64 bytes. It is also shown in Fig. 5 that the cut-through mode offers less latency than the store and forward mode, which in turn improves network performance and delivers the sent frames quickly.

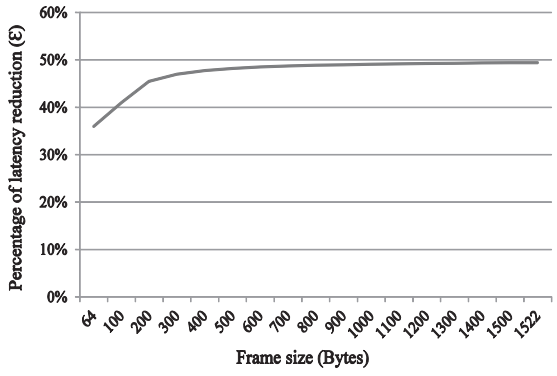


Fig. 4 Percentage of latency reduction versus frame size.

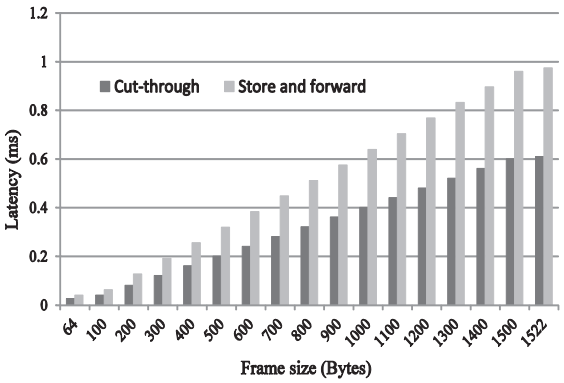


Fig. 5 Frame latency under the cut-through and store and forward switching modes using the QR approach.

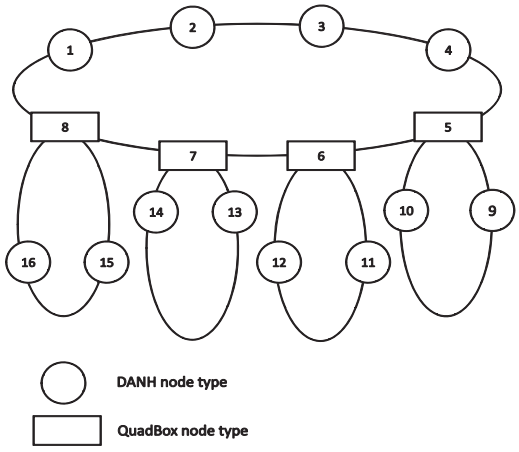


Fig. 6 An example network for simulation analysis.

IV. Simulation Analysis

To illustrate the performance of the QR approach in a wider network, the network shown in Fig. 6 was selected as a network example for simulating the QR operation under both the cut-through switching and the store and forward modes. Then, the results were compared. For this purpose, the OMNeT++ Simulator version 4.6 [20] was used in two scenarios. In these scenarios, the frame latency of all travelling frames in the network was accumulated and then compared under both switching modes. Assume that the frame size is 1522 bytes, all channels have a capacity of 100Mb/s, and all nodes have a processing rate of 100 Mb/s.

A. First scenario

In this scenario, each source node of the connection pairs listed in Table I sent 10 frames to each corresponding node in its pair (source-destination). In other words, this scenario was used for testing the QR approach with the cut-through switching mode under a unicast traffic type.

B. Second scenario

Table II shows the details of the connection pairs for this scenario in which each source node sent 10 multi/broadcast frame types.

For both scenarios, the simulation tests were conducted

TABLE I
Details of The Connection Pairs For The First Scenario

#	Source node	Destination node
1	Node 1	Node 14
2	Node 2	Node 10
3	Node 11	Node 3
4	Node 9	Node 13
5	Node 15	Node 12
6	Node 16	Node 4
7	Node 3	Node 15
8	Node 4	Node 9
9	Node 13	Node 10
10	Node 14	Node 2
11	Node 10	Node 1
12	Node 12	Node 16

TABLE II
DETAILS OF THE CONNECTION PAIRS FOR THE SECOND SCENARIO

Group number	Source node	Destination node
Group 1	Node 1	Nodes 3, 4 and 16
Group 2	Node 10	Nodes 11, 13 and 12
Group 3	Node 13	Nodes 3, 9, and 14
Group 4	Node 2	Nodes 10, 15, 16 and 2
Group 5	Node 15	Broadcast

under the cut-through and store and forward switching modes. The results are shown in Figures 7 and 8. The results illustrate that the QR approach has a better performance under the cut-through switching mode with respect to the frame latency parameter.

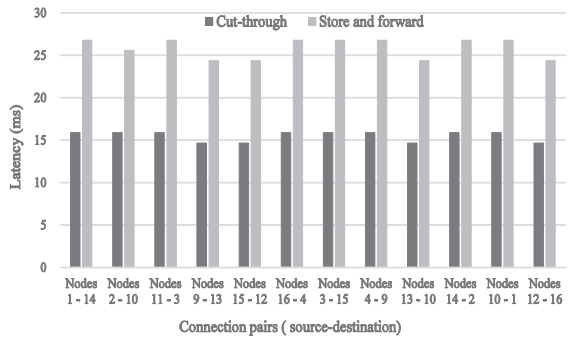


Fig. 7 Frame latency under the cut-through and store and forward switching modes for the first scenario.

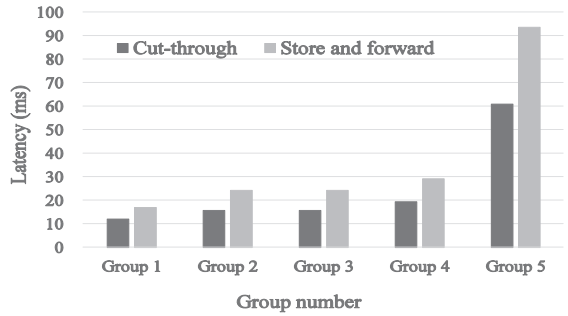


Fig. 8 Frame latency under cut-through and store and forward switching modes for the second scenario.

V. CONCLUSIONS

In this paper, we describe HSR node behavior when it receives a frame under the QR approach (HSR mode X) using the cut-through switching mode as well as which actions it will take to receive or forward the received frame depending on the destination MAC, source MAC, and HSR header. We also simulated the QR performance from the point of view of the frame latency because the cut-through and the store and forward switching modes will not differ from the traffic reduction aspect. In a single ring network topology, the QR approach under the cut-through switching mode was shown to have reached a reduction percentage in the frame latency of more than 49% compared to the store and forward mode when $h = 144$ and $f = 1522$ bytes. In general, a reduction percentage ranges from 35% to 49% with respect to the frame size. The simulation analysis also showed a superior performance of the QR approach under the cut-through mode for both of the proposed scenarios, or in other words, under any traffic type.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education (No. NRF-2015R1D1A1A02059506).

REFERENCES

[1] D. H. Lee, H. A. Pham, and J. M. Rhee, "SAFE: A Scalable Autonomous Fault-Tolerant Ethernet Sch eme for Large-Scale Star Networks," *Proc. IEICE Transactions on Communications*, vol. 95, no.10, pp.3158-3167, Oct. 2012.

[2] IEC 62439-1 Ed. 1.1, "IEC Industrial communication networks – High availability automation networks – Part 1: General concepts and calculation methods," *The International Electrotechnical Commission*: Geneva, Switzerland, 2013.

[3] IEC 62439-3 Ed.03, "IEC Standard. Industrial Communication Networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)," *The International Electrotechnical Commission*: Geneva, Switzerland, 2016.

[4] IEC 61850-90-4, "IEC Standard. Network Engineering Guideline for Communication Networks and Systems in Substations," *The International Electrotechnical Commission*: Geneva, Switzerland, 2013.

[5] S. A. Nsaif and J. M. Rhee, "DVP: A Novel High-Availability Seamless Redundancy (HSR) Protocol Traffic-Reduction Algorithm for a Substation Automation System Network," *Proc. Energies*, vol. 7, no. 3, pp.1792-1810, 2014

[6] J.C. Tan and W. Luan, "IEC 61850 based substation automation system architecture design," *Proc. IEEE Power Energy Soc. General Meeting*, pp.1-6, San Diego, CA, USA, July 2011.

[7] J. A. Araujo, J. Lázaro, A. Astarloa, A. Zuloaga, and J. I. Garate, "PRP and HSR for High Availability Networks in Power Utility Automation: A Method for Redundant Frames Discarding," *Proc. IEEE Transaction on Smart Grid*, vol. 6, no. 5, pp.2325-2332, Jan. 2015

[8] A. Astarloa, J. Lazaro, U. Bidarte, J. A. Araujo, and N. Moreira, "FPGA implemented cut-through vs store-and-forward switches for reliable Ethernet networks," *Proc. IEEE conference on design of circuits and integrated circuits (DCIS)*, pp. 1-6, Madrid, Spain, Nov. 2014.

[9] J. A. Araujo, J. Lazaro, A. Astarloa, A. Zuloaga, and N. Moreira, "Duplicate and circulating frames discard methods for PRP and HSR (IEC62439-3)," *Proc. IECON 2013-39th Annual Conference of the IEEE in Industrial Electronics Society*, pp.4451-4456, Vienna, Austria, Nov. 2013.

[10] S. A. Nsaif and J. M. Rhee, "Improvement of high-availability seamless redundancy (HSR) traffic performance for smart grid communications," *Proc. Journal of Communications and Networks*, vol.14, no.6, pp.653-661, Dec. 2012.

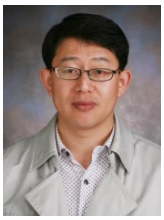
[11] S. A. Nsaif and J. M. Rhee, "RURT: A novel approach for removing the unnecessary redundant traffic in any HSR closed-loop network type," *Proc. International Conference on ICT Convergence (ICTC)*, pp.1003-1008, Jeju, South Korea, Oct.2013.

[12] S. A. Nsaif and J. M. Rhee, "RMT: A novel algorithm for reducing multicast traffic in HSR protocol networks," *Proc. Journal of Communications and Networks,* vol.18, no.1, pp.123-131, Feb. 2016.

[13] N. X. Tien, S. A. Nsaif, and J. M. Rhee, "High-availability Seamless Redundancy (HSR) Traffic Reduction Using Optimal Dual Paths (ODP)," *Proc. International Conference on Green and Human Information Technology (ICGHIT)*, pp.37-40, Da Nang, Vietnam, Feb. 2015.

[14] N. X. Tien and J. M. Rhee, "FHT: A Novel Approach for Filtering High-Availability Seamless Redundancy (HSR) Traffic," *Proc. Energies 8*, no. 7, pp.6249-6274, 2015.
[15] I. R. Altaha, J. M. Rhee, and H. A. Pham, "Improvement of High-Availability Seamless Redundancy (HSR) Unicast Traffic Performance Using Enhanced Port Locking (EPL) Approach," *Proc. IEICE Transactions on Information and Systems*, vol. 98, no. 9, pp.1646-1656, 2015.
[16] H. D. Ngo and H.S. Yang. "Latency and Traffic Reduction for Process-Level Network in Smart Substation based on High-availability Seamless Redundancy", *Proc. IEEE Transaction on Industrial Electronics Society*, pp. 2181 - 2189, 2015.
[17] J. A. Araujo, J. Lazaro, A. Astarloa, A. Zuloaga, and A. Garcia. "PRP and HSR version 1 (IEC 62439-2), improvements and a prototype implementation", In *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*, pp. 4410-4415. IEEE, 2013.
[18] Cisco white paper, "Understanding Switch Latency," June 2012. Available: http://www.cisco.com/c/en/us/products/collateral/switches/nexus-3000-series-switches/white_paper_c11-661939.html
[19] S.A. Nsaif, S. Kim, and J. M. Rhee. "Quick removing (QR) approach using cut-through switching mode." In *2016 18th International Conference on Advanced Communication Technology (ICACT)*, pp. 170-174. IEEE, South Korea, 2016.
[20] http://omnetsimulator.com/

Saad Allawi NSAIF received his B.Sc. degree in Electrical Engineering from University of Baghdad, Iraq, in 1999 and his M.Sc. degree in Computer and Control Systems from the same university in 2002. After graduation, he joined the University of Baghdad as an Assistant Lecturer. Later, he joined the Iraqi Ministry of Defense in 2004. He was the Director of the Command and Control Systems (C2) for 7 years. His contribution in designing, developing, and establishing the Iraqi C4I systems is well-known, especially with the Iraqi Defense Network (IDN). In Feb. 2015, he received his Ph.D. in Information and Communications Engineering from Myongji University, South Korea. Currently, he is a Research Assistant Professor at Myongji University. His current research interests are in ubiquitous networks, layer 2 switching, industrial Ethernet, ad hoc networks, and smart grid communications. He is also a member of IEEE-SA and Myongji working group of IEEE and IEC standardization project.



Semog KIM received his B.Sc. degree in Electrical Engineering from Pukyong National University, South Korea, in 1998. In 1999, he joined Dist. of Motorola Broadband tech part as an HFC Network Engineer until 2010. He received his Ph.D. in Information and Communications Engineering from Myongji University in 2013. Currently, he is a researcher at Myongji University. His interests are in ubiquitous networks, smart grid communications, and HFC networks.

Jong Myung RHEE received his PhD from North Carolina State University, USA, in 1987. After 20 years at the Agency for Defense Development in Korea, where he made noteworthy contributions to C4I and military satellite communications, he joined DACOM and Hanaro Telecom in 1997 and 1999, respectively. At Hanaro Telecom, which was the second largest local carrier in Korea, he served as Chief Technology Officer (CTO) with a senior executive vice-president position. His main duty at Hanaro Telecom was a combination of management and new technology development for high-speed Internet, VoIP, and IPTV. In 2006, he joined Myongji University, and currently, he is the Vice President of Myongji University as well as a full professor in the Information and Communications Engineering Department. His recent research interests are centered on military communications and smart grids, including ad-hoc and fault-tolerant networks. He is a member of IEEE and IEICE.

Volume 5 Issue 4, Jul. 2016, ISSN: 2288-0003

ICACT-TACT JOURNAL

Global IT Research Institute

1713 Obelisk, 216 Seohyunno, Bundang-gu, Sungnam Kyunggi-do, Republic of Korea 13591

Business Licence Number : 220-82-07506, Contact: secretariat@icact.org Tel: +82-70-4146-4991