Nageswara S.V. Rao · Richard R. Brooks
Chase Q. Wu *Editors*

Proceedings of International Symposium on Sensor Networks, Systems and Security

Advances in Computing and Networking with Applications

In honor of Dr. S.S. Iyengar's 70th Birthday

Springer

Proceedings of International Symposium on Sensor Networks, Systems and Security

Nageswara S.V. Rao • Richard R. Brooks
Chase Q. Wu
Editors

Proceedings of International Symposium on Sensor Networks, Systems and Security

Advances in Computing and Networking with Applications

In honor of Dr. S.S. Iyengar's 70th Birthday

Springer

Editors
Nageswara S.V. Rao
Computer Science & Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, TN, USA

Richard R. Brooks
Clemson University
Clemson, SC, USA

Chase Q. Wu
Department of Computer Science
New Jersey Institute of Technology
Newark, NJ, USA

ISBN 978-3-319-75682-0 ISBN 978-3-319-75683-7 (eBook)
https://doi.org/10.1007/978-3-319-75683-7

Library of Congress Control Number: 2018941098

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The 25 years since the first introduction of distributed sensor networks as a special issue edited by S.S. Iyengar, R.L. Kashyap, and Dr. Madan in *IEEE-Transactions on Systems Man and Cybernetics* (Sept/Oct 1991 vol. 21, 5) appeared have seen the field of sensor networks ripen. This research area has exploded becoming one of the central branches of computer science and engineering. The current state of the research in this area could hardly have been predicted when the special issue was published in 1991.

This state of the art book discusses current trends in sensor networks, systems, and security. This proceedings contains a collection of articles authored by outstanding researchers in the field, providing an excellent reference for graduate students and researchers in this area.

This proceedings is also in celebration of Dr. Iyengar in honor of his 70th birthday. It is a monumental task to bring together some of the novel work that is going on in this area. This book will essentially provide an interface to the body of research in the area of algorithms/architectures and applications to the real world.

This book presents current research results in advanced computing and networking within the broader context of sensors, physical and biological systems, while concentrating on the topics that have the most impact on society:

- **Privacy**—The public has become aware of the lack of privacy provided by most current applications. Public debate of privacy may be leading toward increased regulation.
- **High-performance computing in the cloud**—The cloud has changed the current business model. New advances are possible, but it is unclear how these new advances will be managed.
- **Networking and the Internet of Things**—Computing is now part of the physical environment. It is both omnipresent and part of the physical environment.
- **Bioinformatics**—These new technologies are increasing our ability to understand and manipulate not only our environment but also our physical selves.

All of these themes are making important changes in society. We note that the full impact of our evolving technologies is only evident when these themes are put

together. This book brings together these themes in a way that provides the reader with a fuller understanding of the current state of technology and society.

This book provides the reader with a view of current trends that are dominating technology and society. Networked devices are observing our physical environment, which allows high performance computing in the cloud to extract more detailed insights into our lives. These insights include the possible benefits from an increased understanding of our health and biology. They also include risks due to a loss of privacy.

This proceedings comprises the following parts:

Part I: Privacy and Game Theory

1. A Novel Perfect Privacy PIR Scheme for Privacy Critical Applications authored by Radhakrishna Bhat and N.R. Sunitha proposes a generic protocol for privacy preservation.
2. Intelligent Access Control: A Self-Adaptable Trust-Based Access Control Framework (SATBAC) Using Game Theory Strategy authored by Thejas G.S. et al. proposes adaptive framework to addresses security issues in case of experienced user threatening to be malicious.
3. Public Key Cryptosystem for Privacy Sensitive Location-Based Services authored by Mahesh Kumar et al. provides modified solution by using existing approaches to overcome the false negative in proximity testing.
4. Stochastic Tools for Network Intrusion Detection authored by Li Yu et al., focuses on rapid application development, models security issues as stochastic systems, and proposes hybrid network security scheme.

Part II: High-Performance Computing and the Cloud

5. Accelerating the Big Data Analytics by GPU-based Machine Learning: A Survey authored by K. Bhargavi et al. discusses the several GPU-based machine learning algorithms.
6. Multimedia Data Management for Disaster Situation Awareness authored by Maria et al. describes how multimedia data management plays a prominent role in improving the capabilities to readily manage disaster situations.
7. On the Need of Security Standards in Big Data Information Flow authored by Christopher Harrison et al. points out the need for practical security standards in big data information flow.
8. Using Markov Models and Statistics to Learn, Extract, Fuse, and Detect Patterns in Raw Data authored by R.R. Brooks et al. describes their data-driven approach for extracting stochastic state machine directly from the observed data and demonstrates with numerous practical data.

9. Dynamic Firewall Policy Management Framework for Private Cloud authored by Mahesh et al. proposes a dynamic firewall policy management scheme that uses a centralized controller to manage all firewalls in distributed data centers of a private cloud.
10. Security Threats and Solutions for Virtualization and Migration in Virtual Machines authored by Ravi N. et al. discusses the various threats and associated attacks pertaining to virtualization and migration in virtual machine and also proposes a new framework.
11. Techniques to Certify Integrity and Proof of Existence for a Periodical Re-encryption-Based Long-Term Archival Systems authored by Shantakumara et al. proposes techniques to evaluate periodical re-encryption-based archival system and storage systems for long-term point of view.

Part III: Networking and the Internet of Things (IoT)

12. A Study of Contact Durations for Vehicle to Vehicle Communications authored by Quynh et al. discusses the various aspects in contact durations for vehicular communication.
13. Evolution of Sensors Leading to Smart Objects and Security Issues in IoT authored by Sanjeev Kaushik Raman et al. highlights the importance of sensors and the way they have evolved to make IoT possible and the various security issues that exist.
14. Flexible Bandwidth Scheduling for Streaming Data Movement Over Dedicated Networks authored by Liudong et al. proposes a flexible bandwidth reservation algorithm for streamed data movement over dedicated networks.
15. Internet of Things: Current Trends and Emerging Prospects authored by Amartya Sen et al. provides a high level discussion of the overall impact of the IoT paradigm and future research directions.
16. Measurements and Analytics of Data Transport Over Dedicated Connections authored by Nageswara Rao presents a historic account of early works in robot navigation in unknown terrains.
17. Trends and Future Directions of Research for Smart Grid IoT Sensor Networks authored by Arif I. Sarwat et al. provides a high level description and summary of a smart grid IoT and potential challenges to future research.
18. SCADA: Analysis of Attacks on Communication Protocols authored by Pamod T.C. et al. discusses the attacks from the year 1982 to 2017 and possible analytical solutions using SCADA.

Part IV: Bioinformatics

19. TASB-AC: Term Annotated Sliding-Window-Based Boosting Associative Classifier for DNA Repair Gene Categorization authored by Vidya et al. proposes a data mining technique to relate association of DNA repair genes with the aging process of the organism.
20. Temporal Analysis of Stress Classification Using QRS Complex of ECG Signals authored by Neha et al. demonstrates that ECG signals can be used to detect and classify stress in a person using as low as 5 s data stream.
21. Fuzzy Logic Approach for Adaptive Health Interventions by Juan M. Calderon et al. demonstrates their proposed system that is able to relieve patient from stress, while minimizing the intensity of the interventions.

Oak Ridge, TN, USA — Nageswara S.V. Rao
Clemson, SC, USA — Richard R. Brooks
Newark, NJ, USA — Chase Q. Wu

Contents

1 On the Need of Security Standards in Big Data Information Flow ... 1
Christopher Harrison, Makala Quinn, Jacob Livingston, and Karim O. Elish

2 On Data Transfers Over Wide-Area Dedicated Connections 13
Nageswara S. V. Rao and Qiang Liu

3 Temporal Analysis of Stress Classification Using QRS Complex of ECG Signals 35
Neha Keshan, Isabelle Bichindaritz, Patanjali V. Parimi, and Vir V. Phoha

4 Trends and Future Directions of Research for Smart Grid IoT Sensor Networks 45
Arif I. Sarwat, Aditya Sundararajan, and Imtiaz Parvez

5 Accelerating the Big Data Analytics by GPU-Based Machine Learning: A Survey 63
K. Bhargavi and B. Sathish Babu

6 A Novel Perfect Privacy PIR Scheme for Privacy Critical Applications 85
Radhakrishna Bhat and N. R. Sunitha

7 Intelligent Access Control: A Self-Adaptable Trust-Based Access Control (SATBAC) Framework Using Game Theory Strategy 97
G. S. Thejas, T. C. Pramod, S. S. Iyengar, and N. R. Sunitha

8 Dynamic Firewall Policy Management Framework for Private Cloud 113
Mahesh Nath Maddumala and Vijay Kumar

9 Evolution of Sensors Leading to Smart Objects and Security Issues in IoT 125
Sanjeev Kaushik Ramani and S. S. Iyengar

10 Multimedia Data Management for Disaster Situation Awareness 137
Maria E. Presa Reyes, Samira Pouyanfar, Hector Cen Zheng, Hsin-Yu Ha, and Shu-Ching Chen

11 Internet of Things: Current Trends and Emerging Prospects 147
Amartya Sen and Sanjay Madria

12 Public Key Cryptosystem for Privacy Sensitive Location-Based Services 163
K. M. Mahesh Kumar and N. R. Sunitha

13 A Study of Contact Durations for Vehicle to Vehicle Communications 173
Quynh Nguyen and Bhaskar Krishnamachari

14 Flexible Bandwidth Scheduling for Streaming Data Movement Over Dedicated Networks 185
Liudong Zuo, Michelle Zhu, Chase Wu, Nageswara S. V. Rao, Min Han, and Anyi Wang

15 Stochastic Tools for Network Intrusion Detection 197
Lu Yu and Richard R. Brooks

16 Techniques to Certify Integrity and Proof of Existence for a Periodical Re-encryption-Based Long-Term Archival Systems 207
A. H. Shanthakumara and N. R. Sunitha

17 SCADA: Analysis of Attacks on Communication Protocols 219
T. C. Pramod and N. R. Sunitha

18 Security Threats and Solutions for Virtualization and Migration in Virtual Machines 235
N. Ravi and N. R. Sunitha

19 TASB-AC: Term Annotated Sliding-Window-Based Boosting Associative Classifier for DNA Repair Gene Categorization 245
A. Vidya, Santosh Pattar, M. S. Roopa, K. R. Venugopal, and L. M. Patnaik

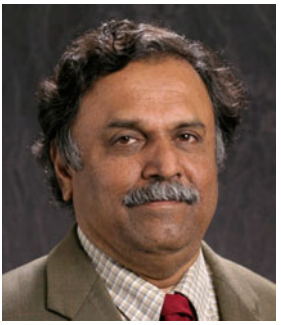
20 Using Markov Models and Statistics to Learn, Extract, Fuse, and Detect Patterns in Raw Data 265
R. R. Brooks, Lu Yu, Yu Fu, Guthrie Cordone, Jon Oakley, and Xingsi Zhong

21 A Control-Based Modeling Approach for Simulating Reaction to Stress Interventions 285
Juan M. Calderon and Luis G. Jaimes

Author Index 297

Subject Index 299

A Brief Biography of Dr. S.S. Iyengar



S.S. Iyengar is a Distinguished Ryder Professor and Director of the School of Computing and Information Sciences at Florida International University, Miami. Dr. Iyengar is a pioneer in the field of distributed sensor networks/sensor fusion, computational aspects of robotics, and high-performance computing. He has published over 600 research papers and has authored/edited 22 books published by MIT Press, John Wiley & Sons, Prentice Hall, CRC Press, Springer Verlag, etc. These publications have been used in major universities all over the world. He has many patents and some patents are featured in the World's Best Technology Forum in Dallas, Texas. His research publications are on the design and analysis of efficient algorithms, parallel computing, sensor networks, and robotics. During the last four decades, he has supervised over 55 Ph.D. students, 100 master's students, and many undergraduate students who are now faculty at major universities worldwide or scientists or engineers at national labs/industries around the world. He has also had many undergraduate students working on his research projects. Recently, Dr. Iyengar received the Times Network 2017 Nonresident Indian of the Year Award—a prestigious award for global Indian leaders.

Dr. Iyengar is a member of the European Academy of Sciences, a Fellow of IEEE, a Fellow of ACM, a Fellow of AAAS, a Fellow of the National Academy of Inventors (NAI), a Fellow of Society of Design and Process Program (SPDS), a Fellow of Institution of Engineers (FIE), and a Fellow of the American Institute for Medical and Biological Engineering (AIMBE). He was awarded a Distinguished Alumnus Award of the Indian Institute of Science, Bangalore, and the IEEE Computer Society Technical Achievement for the contributions to sensor fusion algorithms and parallel algorithms. He also received the IBM Distinguished Faculty Award and NASA Fellowship Summer Award at Jet Propulsion Laboratory. He is a Village Fellow of the Academy of Transdisciplinary Learning and Advanced Studies in Austin, Texas, 2010.

He has received various national and international awards including the Times Network NRI (Nonresident Indian) of the Year Award for 2017, the National Academy of Inventors Fellow Award in 2013, the NRI Mahatma Gandhi Pradvasi Medal at the House of Lords in London in 2013, and a Lifetime Achievement Award conferred by the International Society of Agile Manufacturing (ISAM) in recognition of his illustrious career in teaching, research, and administration and a lifelong contribution to the fields of Engineering and Computer Science at Indian Institute of Technology (BHU). In 2012, Iyengar and Nulogix were awarded the 2012 Innovation-2-Industry (i2i) Florida Award. Iyengar received a Distinguished Research Award from Xiamen University, China, for his research in sensor networks, computer vision, and image processing. Iyengar's landmark contributions with his research group include the development of grid coverage for surveillance and target location in distributed sensor networks and the Brooks Iyengar fusion algorithm. He has also been awarded Honorary and Doctorate of Science and Engineering Degree. He serves on the advisory board of many corporations and universities around the world. He has served on many national science boards such as NIH—National Library of Medicine in Bioinformatics, National Science Foundation review panel, NASA Space Science, Department of Homeland Security, Office of Naval Security, and many others. His contribution to the US Naval Research Laboratory was a centerpiece of a pioneering effort to develop image analysis for science and technology and to expand the goals of the US Naval Research Laboratory.

The impact of his research contributions can be seen in companies and national labs like Raytheon, Telecordia, Motorola, the United States Navy, DARPA, and other US agencies. His contribution in DARPA's program demonstration with BBN, Cambridge, Massachusetts, MURI, researchers from PSU/ARL, Duke, University of Wisconsin, UCLA, Cornell university, and LSU has been significant.

He is also the founding editor of the *International Journal of Distributed Sensor Networks*. He has been on the editorial board of many journals and is also a PhD committee member at various universities, including CMU, Duke University, and many others throughout the world. He is presently the editor of *ACM Computing Surveys* and other journals. He is also the founding director of the FIU's Discovery Laboratory. His research work has been cited extensively. His fundamental work has been transitioned into unique technologies. All through his four-decade long professional career, Dr. Iyengar has devoted and employed mathematical morphology in a unique way for quantitative understanding of computational processes for many applications.

The Editors

Nageswara S.V. Rao —Dr. Rao received his B.Tech in Electronics and Communications Engineering from Regional Engineering College, Warangal, India. He received his ME in Computer Science and Automation from Indian Institute of Science in 1984. He completed his PhD under the supervision of Dr. Iyengar from Louisiana State University. He served as an assistant professor in the Department of Computer Science, Old Dominion University, from 1988 to 1993. He was the UT-Battelle Corporate Fellow (2001), Distinguished Staff Member (2001), Senior Staff Member (1997), and Staff Member (1993) in Oak Ridge National Laboratory. His research interests include sensor networks, robotics, and algorithms.

Richard R. Brooks —Dr. Brooks has in the past been PI on research programs funded by the Air Force Office of Scientific Research, National Science Foundation, Department of Energy, National Institute of Standards, Army Research Office, Office of Naval Research, and BMW Corporation. These research projects include coordination of combat missions among autonomous combat vehicles (ARO), situation and threat assessment for combat command and control (ONR), detection of protocol tunneling through encrypted channels (AFOSR), security of intelligent building technologies (NIST), experimental analysis of Denial of Service vulnerabilities (NSF), mobile code security (ONR), and security analysis of cellular networks used for vehicle remote diagnostics (BMW). Dr. Brooks' current research interests include game theory, strategic reasoning, and information assurance. He was PI of the Mobile Ubiquitous Security Environment (MUSE) Project sponsored by ONR as a Critical Infrastructure Protection University Research Initiative (CIP/URI). It concentrated on creating distributed countermeasures to overcome large-scale network attacks like distributed denial of service and worms. Dr. Brooks was co-PI of a NIST project defining the security standards and protection profiles for the ISO BACNET networked building control systems standard. Dr. Brooks was co-PI of a DARPA ISO program coordinating air campaign command and control and PI of the Reactive Sensor Networks (RSN) Project sponsored by DARPA IXO. RSN explored collaborative signal processing to aggregate information moving

through the network, and the use of mobile code for coordination among intelligent sensor nodes. He has received DURIP awards from ONR and ARO that support the study of networked systems interacting with the real world. Current projects include authentication and authorization of exascale computing systems and establishing Internet freedom in West Africa.

Chase Q. Wu —Dr. Qishi Wu is currently an associate professor in the Department of Computer Science and the director of the Center for Big Data at New Jersey Institute of Technology (NJIT). He joined NJIT in fall 2015 from the University of Memphis, where he is an associate professor in the Department of Computer Science. His research interests include big data, high-performance networking, parallel and distributed computing, sensor networks, scientific visualization, and cyber security. His research in networking develops fast and reliable data transfer solutions to help users in a wide spectrum of scientific domains move big data over long distances for collaborative data analytics. His research in computing develops high-performance workflow solutions to manage the execution of and optimize the performance of large-scale scientific workflows in heterogeneous computing environments. Dr. Wu's work has been supported by various funding agencies, including the National Science Foundation, the U.S. Department of Energy, the U.S. Department of Homeland Security, and Oak Ridge National Laboratory, where he is a research staff and works on a number of high-performance networking projects and big data computational science projects. He has published over 190 research articles in highly reputed conference proceedings, journals, and books and won best paper awards at many conferences.

Chapter 1
On the Need of Security Standards in Big Data Information Flow

Christopher Harrison, Makala Quinn, Jacob Livingston, and Karim O. Elish

Abstract Big Data has become increasingly popular due to its ability to deliver information and optimize current business, health, economic, and research processes. Even though the use of Big Data spans over multiple industries, there is still a lack of security standards and regulations surrounding what data can be captured and how it can be used. This absence of laws and regulations surrounding Big Data security has potentially led to many data breaches. Since there are many participants within the Big Data information flow, each one needs to be further inspected to determine what security measures should be implemented at each stage to prevent these data breaches from occurring. The objective of this position paper is to point out the need for practical security standards in Big Data information flow. In particular, we identify which security standards should be applied at each stage of the Big Data information flow to ensure the privacy and security of valuable and sensitive data.

Keywords Big data · Security standards · Security and privacy · Information flow

1.1 Introduction

In this data-driven society, Big Data has become a key factor in decision making in many different industries. The US National Science Foundation defines Big Data as data sets generated from different data sources, e.g., Internet transactions, sensor data, email, and videos, which are usually by nature large, diverse, and/or distributed [4]. Many industries that have access to these massive data sets are able to incorporate large amounts of structured and unstructured data to analyze, forecast, and monitor trends unlike ever before.

C. Harrison (✉) · M. Quinn · J. Livingston · K. O. Elish
Florida Polytechnic University, Lakeland, FL, USA
e-mail: charrison3465@floridapoly.edu; makalaquinn2245@floridapoly.edu; jacoblivingston0427@floridapoly.edu; kelish@floridapoly.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_1

These industries have a broad range of their application of Big Data. Some industries are using Big Data analysis techniques to monitor economic activities such as retail activity or monitoring payments for their financial institution. Others are using Big Data analytics for the purpose of discovering new findings in healthcare or other scientific research.

Although Big Data provides many benefits, there is a lack of security measures to secure and protect the valuable and sensitive information that is collected. There are limited laws and regulations on what data can be collected, analyzed, and sold to third parties. Thus, there is a need for more rigorous standards on data security to ensure the sensitive and personal data, which is processed by the Big Data industry, is protected. In this paper, we aim to point out the need for practical security standards in Big Data information flow and what security standards should be enforced at each stage to ensure that data is kept secure throughout the entire process.

The rest of this paper is organized as follows. We survey the existing security and privacy standards of Big Data in Sect. 1.2. Section 1.3 reviews the Big Data industries. In Sect. 1.4, we review the Big Data breaches and summarize the Big Data information flow in Sect. 1.5. We present and discuss the applications of security standards to Big Data in Sect. 1.6. Section 1.7 concludes the paper.

1.2 Related Work

This section briefly highlights the existing work related to Big Data ethics, Big Data security and privacy in healthcare, and challenges within Big Data security.

1.2.1 Big Data Ethics

Richards and King [6] assessed the state of Big Data ethics and recognized four principles of Big Data ethics as follows:

- Data privacy should be seen as information rules
- Shared private data can be kept confidential
- Big Data requires transparency to prevent abuses
- Big Data can compromise identities

It is pointed out, throughout Richards and King examination, that there are many innovative possibilities when taking advantage of what Big Data has to offer [6]. From a security perspective, we have to realize and address the lack of data protection laws and regulations because by the time these are implemented, the technology has already evolved to where the legislation no longer applies. In order to take advantage of the Big Data benefits, there needs to be a conscious effort surrounding the creation of privacy, confidentiality, and transparency regulations to prevent potential harms from occurring.

Rubinstein [8] discussed the reform efforts implemented with the European Union Data Protection Directive and how they have fallen short since the laws and regulations simply cannot keep up with the technological advances. These advances are allowing industry participants to reidentify the data provider, which removes all data privacy. To reduce the unethical use of the data provider's information, the data provider should be given access to their data or the ability to use Personal Data Services to securely store and de-identify their data [8]. These concerns will continue to exist until there is a mind shift in Big Data ethics, laws, and regulations.

1.2.2 Big Data Security and Privacy in Healthcare

During this time of increased data collection in the medical field, Big Data became vital to healthcare. Patil and Seshadri [5] explored the emerging vulnerabilities that have and will occur as Big Data plays a bigger role in healthcare. They also stated that taking advantage of being able to process and integrate large amounts of data will be crucial in improving patient care. Due to the criticality of patient data, the authors provided a few procedures that should be implemented to help preventing data breaches from occurring in healthcare [5]:

- Data governance
- Real-time security analytics
- Privacy-preserving analytics

1.2.3 Security in Big Data: Concepts, Approaches, and Challenges

Big Data is changing the world with discoveries and innovation. This leads to new developments of data analytic software that can be used to increase understanding of human and social processes and interactions [3]. These insights and breakthroughs can help industries to improve their productivity and efficiency by accessing the benefits that Big Data can offer. However, in today's society there are many security risks that must be accounted for and protected against. The main threats to security can be categorized into three main topics:

- Unauthorized data observation
- Incorrect data modification
- Data unavailability

These topics cover the security breaches found in Big Data systems and deal with the majority of issues that can be resolved using a DBMS or database management system [1]. A DBMS can be used for unauthorized data observation by using an access control mechanism. Incorrect data modification and data integrity can be

controlled using a semantic correctness tool; this ensures that the entered data is compatible with the field. The history of these issues and solutions are discussed in [3].

Liang et al. [3] discovered a new secure transmission that is based upon nested sparse sampling and coprime sampling. This could be used in Big Data to reject interference while still having good transmission performance.

1.3 Big Data in Industries

1.3.1 Big Data in Healthcare

Big Data can be a valuable tool in the healthcare field. It is used when analyzing patient data, satisfaction metrics, and medical device data sets. The government has created initiatives to have all patient records electronic through the Affordable Care Act. These medical records hold great opportunity for Big Data advantages. Health informatics is a new field that involves the design, development, adoption, and adaptation of IT solutions in healthcare, and Big Data is an integral part of these solutions. The knowledge available in the data can be harnessed and help improve the understanding of disease and pinpoint innovative therapies that are more efficient than current solutions [2].

Big Data is also a crucial component when making important decisions about patient care. Big Data has been used to identify issues in medications that cause life-threatening illnesses that in turn have led to policy change [9]. It has been used in process improvement techniques that increase the quality of care that patients receive, such as handwashing monitoring systems. Big Data can be integrated into expert systems that are able to form clinical support systems and increase the accuracy of patient diagnoses. The tools available when using Big Data techniques are also able to help with medical staff fatigue. This may free up physicians and nursing staff to have more time with the patient.

1.3.2 Big Data in Finances

Big Data is used in the financial industry for a wide range of purposes from protecting personal finances to predicting stock market trends. The financial industry can leverage Big Data systems to detect fraudulent payments. This can be used when identifying stolen credit card information for a credit card provider or payment fraud for merchants [9]. The stock market has become more accurately predictable, and the success of today's stock brokers can be improved through to the use of Big Data.

1.3.3 Big Data in Fitness

Big Data has driven a new era of fitness tools that are used in today's industry. Many body vital monitoring devices are available today with mobile applications to pair with the data collected. These devices are all collecting data, and when these data points are collected, organized, and displayed, they offer a wide range of health and fitness benefits. The following are just a few features that are enabled by the use of Big Data in fitness:

- Tracking exercise
- Creating a health portfolio
- Personalized meal plans
- Personalized fitness plans

These tools can help create a more informed and balanced fitness lifestyle for the international community [9].

1.3.4 Big Data in Retail

Retail is an industry driven by Big Data, which has made it possible for vendors to know the exact number of products bought and sold in real time. This accurate sales and inventory data can be used to restock shelves in real time; when one item is bought and sold, it can be requested to be restocked. Supermarkets can also identify popular items and have data to reinforce item purchase for future stocking and replenishment. Big Data shows customer interest's models that are used in features such as amazon's "customers who bought this also bought" feature [9]. It is used in personalized advertisements, designed to target users' interests and purchasing patterns.

1.4 Overview of Major Data Breaches

Data breaches are becoming increasingly popular throughout the world. Most people's information is being collected without people's knowledge and being shared. The data loss during data breaches can include an individual's name, social security number, driver's license number, medical record, or financial record. These information breaches can be intentional or unintentional information that is released to an untrusted source. These data breaches are not only limited to stealing people's information digitally but could consist of accessing an individual's personal record without consent.

Not only are the number of data breaches increasing, but their impact is as well. For one breach, there could be a substantial impact for over 145 million individuals [7]. This ends up costing companies a lot of time and money in order to repair what was lost or stolen from the people that were affected. Data breaches occur every day throughout the world. Personal information is constantly threatened by these breaches. Some examples of top data breaches are presented below:

- Anthem: In 2015, over 80 million individuals were exposed by having their names, social security numbers, birthdates, email addresses, employment information, and income data shared. This data breach could cost Anthem up to $16 billion to fix [7].
- eBay: Over 145 million customers were affected in 2014 when their personal information was shared. This will end up costing the company up to $200 million to fix [7].
- Heartland: In 2009, over 130 million individuals had their financial information shared. This costed the company $2.8 billion to fix. In 2015, the company suffered another small breach that affected another 2000 individuals [7].
- Tricare: Approximately 5 million Tricare military beneficiaries were affected in 2011 when computer tapes with unencrypted personal data about the military service members were taken from a car [7].

1.5 Overview of Big Data Information Flow

The Big Data information flow typically describes the extraction/collection, transmission, and analysis of large amounts of data, which at a high level are the processes completed during Big Data initiatives [10]. Though this information flow process can be complex, it can be broken down into four major stages: data provider, data collector, data miner, and decision maker [10]. These major actors within the Big Data information flow, which are shown in Fig. 1.1, are discussed below.

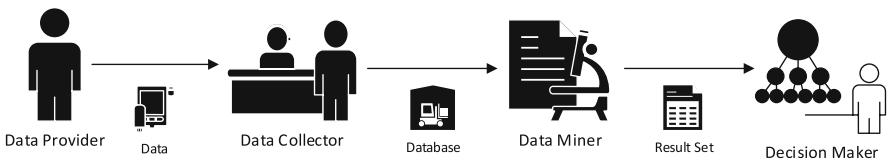


Fig. 1.1 Actors in Big Data information flow

1.5.1 Data Provider

The data provider, the first actor in Fig. 1.1, is an individual who possesses the source data, which is seen as valuable by another person or organization. This data can range from personal information such as date of birth, address, telephone number, personal interests, and shopping habits to scientific testing results or even traffic patterns. This only being a very small subset of the data that can be collected shows that even information that might seem trivial can be used to identify a pattern or trend about the data provider.

Since the data provider owns data that is valuable to the decision maker, the individual has the right to disclose or refuse to provide their data. In the case that the data is disclosed, the data provider needs to consider what data is private or sensitive and how it could be used and identify what damage could be done, if this data is stolen during a data breach.

1.5.2 Data Collector

The data collector, the second actor in Fig. 1.1, is the entity that collects the data about and/or from the data providers. It intakes all the data deemed necessary in many different formats such as a data stream, data mart, or data warehouse. This information is then commonly transferred and stored within its own distributed database. This data source is then published to the data miner.

Since the data collector intakes the raw data from the data provider, it contains private or sensitive information. It is the data collector's responsibility to transform the data in such a way that still provides value to the data miner but de-identifies the data providers in the process. There is a fine line when de-identifying the data because it could cause it to lose its usefulness, making the data collection unproductive.

1.5.3 Data Miner

The data miner, the third participant in Fig. 1.1, is the individual who executes the data mining techniques against the data that is published by the data collector. The data miner uses tools to run complex algorithms against large amounts of this data. Their main goal is to detect any potential relationships or common patterns about the data provider, which earlier may have been undiscovered, that could provide beneficial insight to the decision maker.

Since these in-depth data mining techniques could possibly expose sensitive information about the data provider, the data miner needs to also take precautionary measures to protect this sensitive data from being exposed in the result set provided to the decision maker. This means that the data miner, like the data collector, must also alter the data without reducing its value.

1.5.4 Decision Maker

The decision maker, the last participant in Fig. 1.1, is the end user who uses the data that is collected and analyzed to make a decision to achieve some end goal. This information could be used to identify which products to sell at what price or to move forward with some breakthrough research based on what is presented in the data results.

Since the data has been transformed potentially multiple times by the data collector(s) and miner(s), the decision maker needs to ensure that the data still provides credible information that can be used to achieve their end goal. Without credible information the decision maker could make detrimental business, financial, or research decisions which could lead to irreversible outcomes.

1.6 Application of Security Standards to Big Data Information Flow

After exploring the four major actors in the Big Data information flow in the Sect. 1.5, we discuss the application of security standards next. Each actor within the Big Data information flow must comply with their specific security standards to ensure that data is kept secure throughout the entire process.

1.6.1 Data Provider

Since the data provider's information is being captured, it is their responsibility to be the first line of defense to keeping their data safe. Ultimately, the data provider has a choice on what they want to share with the data collector, even though it is sometimes taken for granted, which leads to personal data being provided without data provider even realizing it.

At least from an online perspective, to protect themselves and their valuable data, the date provider should consider using VPNs, coming up with complex passwords, installing antivirus software, and keeping their software patched to the latest version to reduce security vulnerabilities.

1.6.2 Data Collector

The data collector is equally as important when securing the data provider's information due to the fact that it receives the raw data from the provider. In order to keep the data miner from receiving data that can be linked directly back to identify the data provider, the data collector must take the necessary steps to de-identify the data.

To de-identify the data, the data collector must use some type of data scrambling techniques, which will obfuscate it in such a way that it still can be used for its desired purpose but will not allow the data miner to link it directly back to the provider. As a part of the data masking process, the collector may want to implement tokenization to provide a level of encryption.

The last security standard that should be implemented by the data collector would be to set up a secure access mechanism to ensure that only the data miner and any other acceptable parties can gain access to the data.

1.6.3 Data Miner

The data miner has the ability to discover hidden information within the data, which makes it important that they also modify the data in such a way that prevent the identifiable information from being present in the data mining result set. To accomplish this, the data miner could implement many different transformation techniques to maintain the data provider privacy. Some examples of these techniques would be randomization of certain parts of the data and/or reconstructing the data by applying filters to obfuscate the data [10]. This approach will allow the data miner to protect the privacy of the data while providing the decision maker with effective generalized results.

1.6.4 Decision Maker

The decision maker is the end user and would not typically be seen as one of the parties that need to be concerned with keeping the data secure. It is key that the decision maker is diligent in protecting their results, so that some outside source cannot gain access to, which could lead to a loss in competitive advantage.

To keep their data secure, the decision maker could implement an audit system to know who has accessed or updated the information. Another important thing to remember is that the data is only as good as its accuracy and reliability. One way to ensure the data is reliable is to enforce the data collector(s) and miner(s) to deliver a data provenance, or a tracking mechanism, to provide clear picture of how the data was transformed and manipulated.

1.7 Discussion and Conclusions

Big Data has become a tool used by many companies to analyze large amounts of data to monitor, forecast, and analyze trends. It has become increasingly popular for industries to use the information they have been collecting for numerous years and analyze it to be able to visualize trends and patterns. As data flow increases, the challenges in Big Data security increases. There are numerous benefits that the use of Big Data offers, but this is often canceled out with the risk of security issues.

Industries benefit from the use of Big Data every day. Areas such as healthcare, finances, fitness, retail, and more have been utilizing the benefits of Big Data to increase revenue and productivity.

With the increased use of Big Data, the data breaches have become more impactful. Data breaches occur when someone's information has been accessed by an unauthorized source. This could be someone intentionally or unintentionally receiving this information. Multiple companies have been victims of large data breaches involving their customer's and employees' data. Due to the increase of occurrences and impact of data breaches, we must find a way to protect personal information from untrusted sources.

The Big Data information flow is used to show the process of collecting, transmitting, and analyzing large amounts of data. This process is broken down into four major steps: data provider, data collector, data miner, and decision maker [10]. Data providers process valuable data including personal information such as date of birth, address, telephone number, personal interests, and shopping habits to scientific testing results or even traffic patterns. The data collector is responsible for collecting the data from the data providers. It takes data in different formats and converts them into a data warehouse. The data miner uses tools to run algorithms on large amounts of data and finds relationships and patterns that can be useful to the decision maker. The decision maker takes the information from the data miner and makes decisions to achieve a goal. This process shows the steps Big Data takes to become useful and successful.

Unfortunately, through the Big Data information flow, there are problems with security. When this information is passed from data provider to decision maker, it is often times shared with untrusted sources. In order to make this collection secure, each actor in the process needs to take different measures. The data provider is the person who owns the data, and it is their responsibility to decide what will be shared to the data collector and what personal information is useless. Keeping as much of individual's personal data out of the process as possible will save some information from being affected. The data collector's job is to de-identify the data, which includes a scrambling technique. This technique is used so that the data miner cannot trace any of the data back to an individual. The data collector must also make sure there is a secure way to transfer the data. The data miner is responsible for finding the trends in the data but also must make sure that there is no information that

traces back to an individual. By the time this data gets to the decision maker, there should be no trace of personal information in the data. Using these steps will help protect the individual's personal information and keep it from being compromised.

References

1. Bertino, E., Sandhu, R.: Database security-concepts, approaches, and challenges. IEEE Trans. Dependable Secure Comput. **2**(1), 2–19 (2005)
2. Koumpouros, Y.: Big data in healthcare. In: Healthcare Administration: Concepts, Methodologies, Tools, and Applications, p. 23. IGI Global, Hershey (2014)
3. Liang, Q., Ren, J., Liang, J., Zhang, B., Pi, Y., Zhao, C.: Security in big data. Secur. Commun. Netw. **8**(14), 2383–2385 (2015)
4. NSF-NIH Interagency Initiative: Core techniques and technologies for advancing big data science and engineering (BIGDATA) (2012)
5. Patil, H.K., Seshadri, R.: Big data security and privacy issues in healthcare. In: IEEE International Congress on Big Data (2014)
6. Richards, N., King, J.H.: Big data ethics. Wake Forest Law Rev. **49**, 393–432 (2014)
7. Ross, A.: 11 Major US data breaches. http://www.bankrate.com/finance/banking/us-data-breaches-1.aspx. Retrieved November 2016
8. Rubinstein, I.: Big data: the end of privacy or a new beginning? International Data Privacy Law, 12–56 (2012)
9. Tene, O., Polonetsky, J.: Big data for all: privacy and user control in the age of analytics. Northwest. J. Technol. Intellect. Prop. **11**, 239 (2013)
10. Xu, L., Jiang, C., Wang, J., Yuan, J., Ren, Y.: Information security in big data: privacy and data mining. IEEE Access **2**, 1149–1176 (2014)

Chapter 2
On Data Transfers Over Wide-Area Dedicated Connections

Nageswara S. V. Rao and Qiang Liu

Abstract Dedicated wide-area network connections are employed in big data and high-performance computing scenarios, since the absence of cross-traffic promises to make it easier to analyze and optimize data transfers over them. However, nonlinear transport dynamics and end-system complexity due to multi-core hosts and distributed filesystems make these tasks surprisingly challenging. We present an overview of methods to analyze memory and disk file transfers using extensive measurements over 10 Gbps physical and emulated connections with 0–366 ms round-trip times (RTTs). For memory transfers, we derive performance profiles of TCP and UDT throughput as a function of RTT, which show concave regions in contrast to entirely convex regions predicted by previous models. These highly desirable concave regions can be expanded by utilizing large buffers and more parallel flows. We also present Poincaré maps and Lyapunov exponents of TCP and UDT throughput traces that indicate complex throughput dynamics. For disk file transfers, we show that throughput can be optimized using a combination of parallel I/O and network threads under direct I/O mode. Our initial throughput measurements of Lustre filesystems mounted over long-haul connections using LNet routers show convex profiles indicative of I/O limits.

Keywords Wide-area transport · Dedicated connections · TCP · UDT · Throughput and file I/O profiles · Throughput dynamics · Poincaré map · Lyapunov exponent

N. S. V. Rao (✉) · Q. Liu
Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA
e-mail: raons@ornl.gov; liuq1@ornl.gov

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_2

2.1 Introduction

In big data and high-performance computing (HPC) scenarios, there has been an unprecedented increase in the volume of datasets transferred over long distance network connections. For example, effective memory transfers are critical for coordinated computations over geographically dispersed cloud server sites. Unpredictable cross-traffic flows in shared networks are not well-suited for these specialized scenarios, which has led to the use of dedicated connections in both scientific and commercial environments, for example, by Department of Energy's (DOE) ESnet [4] and Google's software-defined network (SDN) [9]. It is generally expected that the performance dynamics of dedicated connections should be simpler than those of shared connections and that the underlying data flows should achieve high throughput for large data transfers and stable dynamics for control and steering operations. However, experimental and analytical studies of such flows are quite limited, since most research has focused on shared network environments [26]. Our work has shown that while parameter optimizations specific to dedicated connections are indeed somewhat simpler than those required for shared connections, they are not simple extensions of the well-studied shared connection solutions [11, 19]. This observation led to the establishment of a testbed at Oak Ridge National Laboratory (ORNL) and the development of measurement and analysis methods for these scenarios.

We conduct structured experiments of both memory-to-memory and disk-to-disk transfers using the testbed. For memory-to-memory transfers, we consider the UDP-based Data Transfer Protocol (UDT) [6] and four TCP variants, namely, CUBIC [22], Hamilton TCP (HTCP) [25], BBR [3], and Scalable TCP (STCP) [10], which are among those considered suitable for high-bandwidth connections. For disk-to-disk transfers, we study XDD [28] and also mount the Lustre filesystem over long-haul connections. We use dedicated physical connections and a suite of hardware-emulated 10 Gbps connections with 0–366 ms round-trip times (RTTs).

For a given configuration of end systems, hosts, transport method, and connection parameters, we denote the throughput at time t over a connection of RTT τ as $\theta(\tau, t)$. Its average over an observation period T_O is called the *throughput profile*:

$$\Theta_O(\tau) = \frac{1}{T_O} \int_0^{T_O} \theta(\tau, t)\, dt$$

Figure 2.1a shows a representative plot of mean throughput profiles for a single STCP stream [20]. It is *concave* for lower RTTs and *convex* for higher RTTs. These dual-mode profiles are observed for both TCP [18] and UDT [11]. The concave regions are highly desirable since the throughput decreases slowly as RTT increases and in particular is higher than linear interpolation of measurements. Interestingly, such profiles are in stark contrast both to those predicted by analytical models and

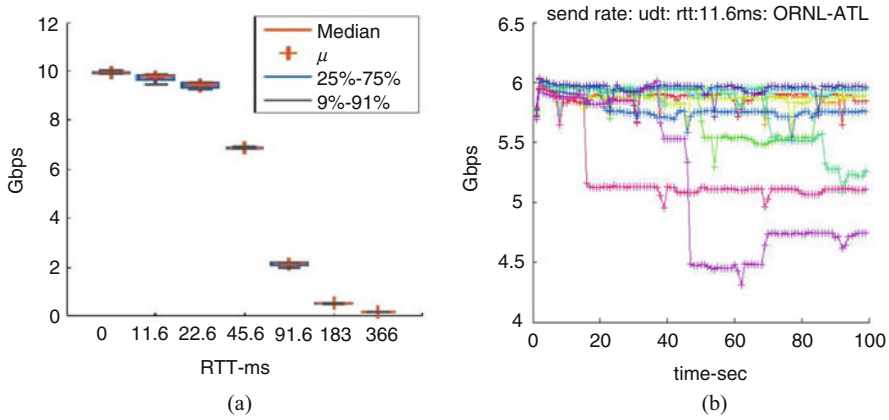


Fig. 2.1 Throughput profile of Scalable TCP and time traces of UDT. (**a**) Throughput profile $\Theta_O(\tau)$ for single STCP flow [20]. (**b**) Time trace $\theta(\tau, t)$ of UDT memory-to-memory transfer [11]

to experimental measurements collected for various TCP variants [7, 17, 29] and UDT [6]. The conventional TCP models lead to entirely convex profiles where the throughput decreases faster with RTT [13, 15, 26] and do not explain this dual-regime profile. Our measurements demonstrate that both large host buffers and an increasing number of parallel streams expand the concave regions, in addition to improving throughput.

Throughput time traces also exhibit rich dynamics, as illustrated in Fig. 2.1b for memory-to-memory transfers using UDT. These dynamics impact the throughput profiles in subtle ways, as revealed by our application of Poincaré map and Lyapunov exponent methods from chaos theory [20]. For TCP, at lower RTTs, higher throughput and smaller variations result in concave regions, and at higher RTTs, lower throughput and larger variations lead to convex regions.

This chapter presents a comprehensive view of measurements and analyses of systematic experiments conducted using our testbed. It unifies the analysis of throughput profiles of TCP [20], UDT [11], XDD [19], and BBR [21], which we have previously treated separately but in more detail. Additionally, we include some preliminary throughput profiles of the Lustre filesystem mounted over long-haul Ethernet connections and initial GridFTP throughput measurements over dedicated connections with 0–366 ms RTT.

The organization of this chapter is as follows. A brief description of our testbed is provided in Sect. 2.2. The concave and convex regions of TCP and UDT throughput profiles are discussed in Sect. 2.3. Analysis of transport dynamics using Poincaré map and Lyapunov exponents is briefly presented in Sect. 2.4. File transfer measurements are described in Sect. 2.5. Conclusions and open areas are briefly described in Sect. 2.6.

2.2 Network Testbed with FileSystems

We performed experiments over a testbed consisting of 32/48-core Linux workstations, 100G Ethernet local and long-haul switches, QDR InfiniBand (IB) switches, 10 Gbps hardware connection emulators, and disk controllers. The testbed is located at ORNL and is also connected to a 500 mile 100 Gbps physical loop-back connection to Atlanta. For various network connections, hosts with identical configurations are connected in pairs over a back-to-back fiber connection with a negligible 0.01 ms RTT and a physical 10GigE connection with a 11.6 ms RTT via Cisco and Ciena devices, as shown in Fig. 2.2a. The 10GigE and SONET/OC192 connections represent different physical modalities; we present results with the latter.

We use ANUE devices to emulate 10GigE and SONET/OC192 connections with RTTs $\tau \in \{0.4, 11.8, 22.6, 45.6, 91.6, 183, 366\}$ ms. These RTT values are chosen to represent three basic scenarios: lower values represent cross-country connections, for example, between facilities across the USA; 93.6 and 183 ms represent intercontinental connections, for example, between the USA, Europe, and Asia; and 366 ms represents a connection spanning the globe and is mainly used as a limiting case.

Memory-to-memory throughput measurements for TCP are collected using *iperf* and for UDT using its custom codes. Typically, 1–10 parallel streams are used for each configuration, and throughput measurements are repeated ten times. Three different TCP buffer sizes are used: default, normal (recommended values for 200 ms RTT), and large (the largest size allowed by the host kernel); and the socket buffer parameter for iperf is 2 GB. These settings result in the allocation of 250 KB, 250 MB, and 1 GB socket buffer sizes, respectively.

The Lustre filesystem employs multiple Object Storage Targets (OSTs) to manage collections of disks, multiple Object Storage Servers (OSSes) to stripe file contents, and distributed MetaData Servers (MDSes) to provide site-wide file naming and access. Our testbed consists of a distributed Lustre filesystem supported by eight OSTs connected over the QDR IB switch as shown in Fig. 2.2b. Host systems are connected to the IB switch via Host Channel Adapters (HCA), and Lustre over IB clients are used to mount the filesystem on them. In addition, our solid-state storage device (SSD) drives are connected over PCI buses on the hosts, which mount local XFS filesystems. Data transfers between filesystems locally mounted at sites and connected over long-haul links are carried out using XDD [28] whose codes are used for throughput measurements. We also consider configurations wherein Lustre is mounted over long-haul connections using LNet routers, and in this case we utilize IOzone for throughput measurements for both site and remote filesystems.

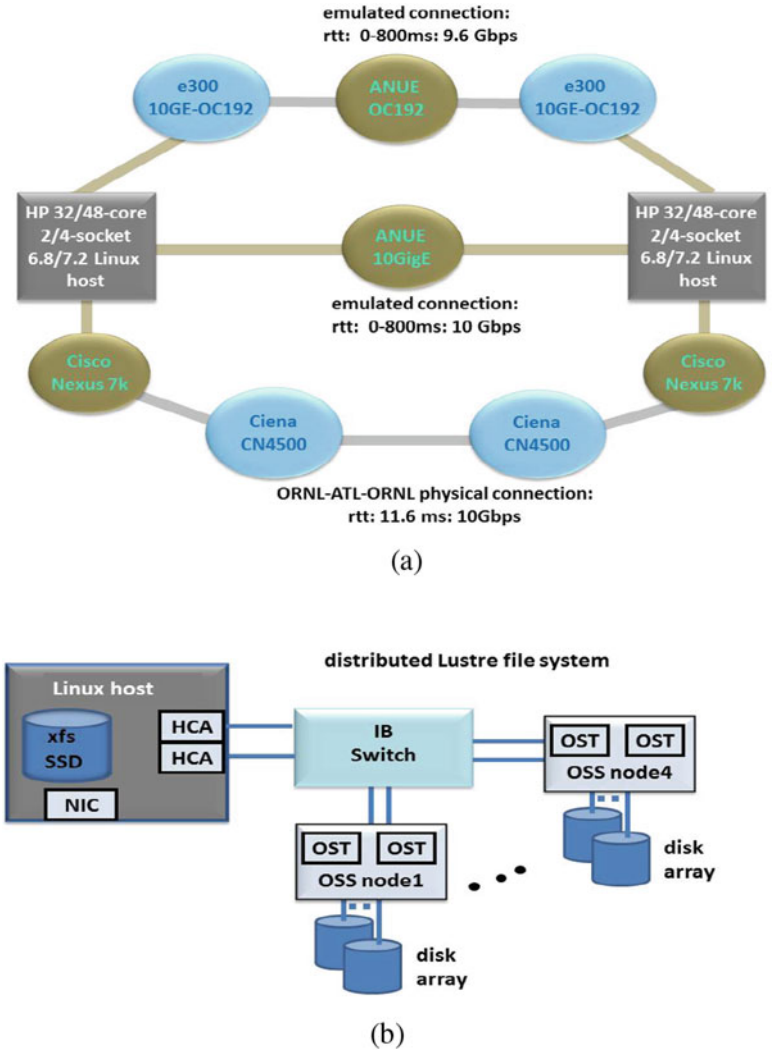


Fig. 2.2 Testbed network connections and filesystems. (**a**) Physical and emulated connections between hosts. (**b**) Lustre and XFS filesystems

2.3 Throughput Profiles

We have collected extensive UDT and TCP throughput measurements over connections of seven different lengths, using either UDT or TCP variants (BBR, CUBIC, HTCP, STCP) and, for the latter, three buffer sizes and 1–10 parallel streams. The connection throughput profiles share certain common qualitative properties in terms

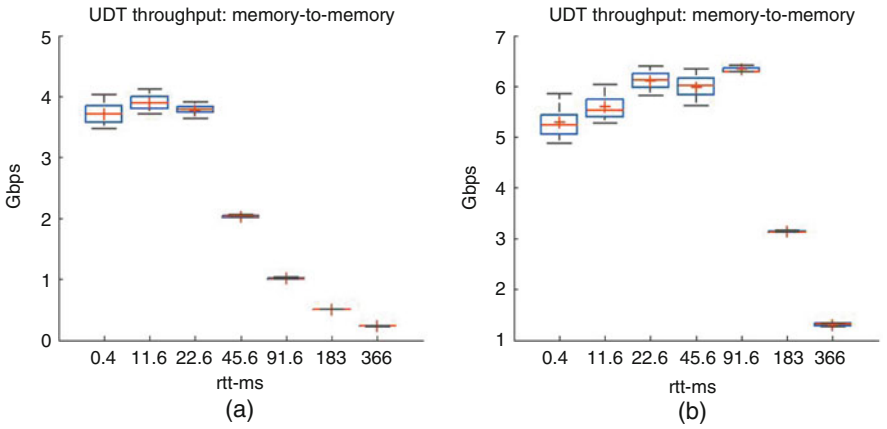


Fig. 2.3 Aggregate throughput profiles for UDT. (**a**) Default. (**b**) Jumbogram

of monotonicity and concave/convex regions, which we summarize in this section; more detailed descriptions and analyses of these large data sets are provided in [11] and [20].

2.3.1 UDT and TCP Throughput Measurement Profiles

We compute the mean throughput profile by taking the average throughput rates from repeated transfer experiments conducted at specific RTT (τ) values for each TCP variant and number of parallel streams. For each time trace, we sample throughput for a total duration of 100 s at 1 s intervals and repeat this process 10 or 100 times for each RTT. We estimate the average throughput for each sampled trace that provides multiple throughput estimates at each RTT, which together constitute the aggregate throughput profile.

The aggregate UDT throughput profile in Fig. 2.3 shows the throughput mean (denoted as "+"), quartiles, minima, and maxima. We examine both the standard packet size of 1500 bytes (as specified by the current protocol) and for 9000 byte jumbograms. Such 9000 byte frames have been shown to reduce overheads and CPU cycles and thereby improve the communication efficiency over 10 Gbps connections. The profile box plots indeed show that compared to the default packet size, jumbograms effectively lead to higher throughputs that are sustained for connections with RTTs above 100 ms.

In Fig. 2.4, mean throughput rates of BBR are plotted for three buffer sizes, namely, default, normal, and large. A large buffer size significantly improves the mean throughput, especially for longer connections; for instance, the throughput of 10 streams for 366 ms RTT improves from under 100 Mbps to over 8 Gbps as the buffer size increases. Unless otherwise specified, subsequent discussions primarily

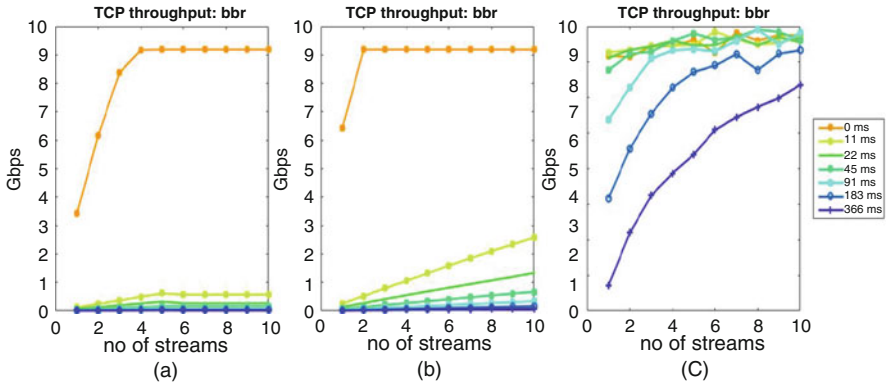


Fig. 2.4 Throughput with variable RTTs, number of streams, and buffer sizes for BBR. (**a**) Default. (**b**) Normal. (**c**) Large

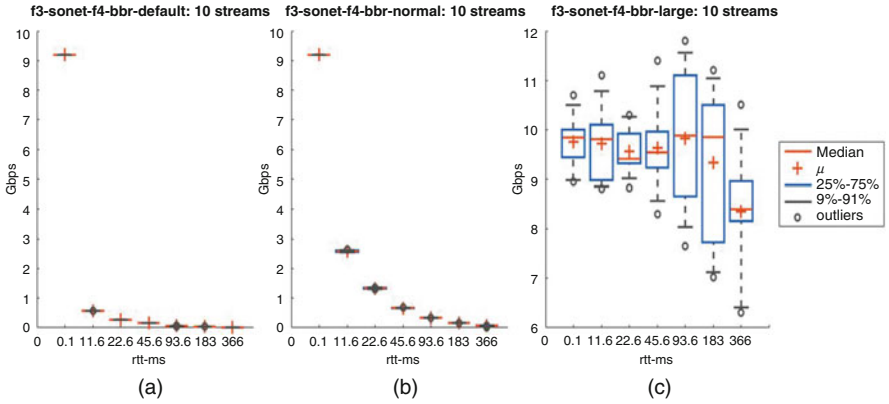


Fig. 2.5 Throughput box plots with variable RTTs and buffer sizes for BBR with 10 streams. (**a**) Default. (**b**) Normal. (**c**) Large

address performance with large buffers. As seen from Fig. 2.5, for BBR with large buffers and 10 parallel streams, using the default and normal buffer sizes results in entirely convex profiles; with the large buffer size, a concave profile is observed for the mean throughput, as shown in Fig. 2.5c for the 10 stream case and in Fig. 2.6 for the single stream case. Comparing these BBR profiles with the corresponding profiles for CUBIC, HTCP, and STCP in [20], we observe that BBR generally requires much larger buffers; namely, large buffers are required for BBR to achieve similar throughput rates under "default" or "normal" buffer sizes in the other TCP variants. In addition, Fig. 2.5c shows the mean throughput values of BBR with ten parallel streams, and large buffers are higher than those achieved using other TCP variants, albeit with much larger variations.

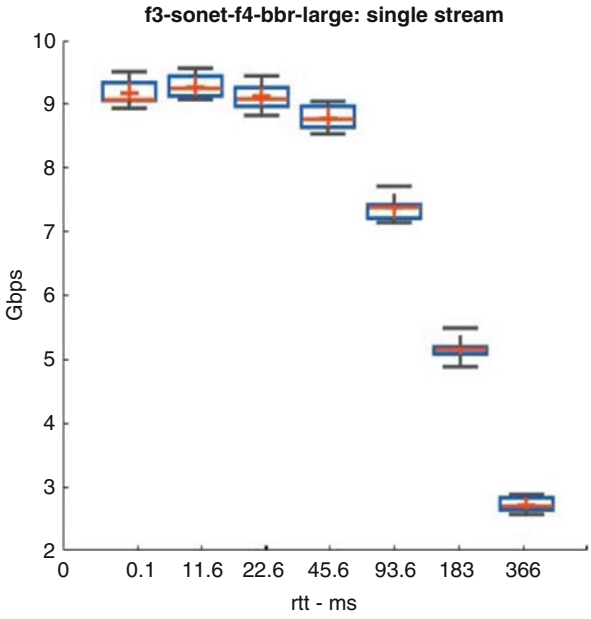


Fig. 2.6 Throughput box plots with variable RTTs for single-stream BBR with large buffers

2.3.2 Throughput Model

To explain the overall qualitative behavior observed in measurements as discussed in the previous subsection, a generic and coarse throughput model is proposed for TCP [18, 20] and UDP [11]. The throughput dynamics with fixed parameters over a connection with RTT τ and capacity C are characterized by two phases:

(a) *Ramp-Up Phase:* In the ramp-up phase, the throughput increases for a duration of T_R until it reaches a peak and then switches to a sustained throughput phase. The average throughput in this phase is

$$\bar{\theta}_R(\tau) = \frac{1}{T_R} \int_0^{T_R} \theta(\tau, t)\, dt\,.$$

(b) *Sustained Throughput Phase:* Once the throughput reaches the peak, it is "sustained" using a mechanism that processes the acknowledgments and infers and responds to losses. The average throughput in this phase is

$$\bar{\theta}_S(\tau) = \frac{1}{T_S} \int_{T_R}^{T_R+T_S} \theta(\tau, t)\, dt\,.$$

In general, the average $\bar{\theta}_S$ lies below the link capacity due to variations in throughput as shown in Fig. 2.1b.

Using $\bar{\theta}_R$ and $\bar{\theta}_S$, the average throughput is given as

$$\Theta_O(\tau) = \frac{T_R}{T_O}\bar{\theta}_R(\tau) + \frac{T_S}{T_O}\bar{\theta}_S(\tau) = \bar{\theta}_S(\tau) - f_R\left(\bar{\theta}_S(\tau) - \bar{\theta}_R(\tau)\right),$$

where $T_O = T_R + T_S$ and $f_R = T_R/T_O$. For a large observation period T_O with a fast ramp-up, typical in small RTT (τ) settings, the qualitative properties of $\theta_S(\tau)$ directly carry over to $\Theta_O(\tau)$. On the other hand, for large RTT, the ramp-up period can be significantly longer, for example, 10 s for 366 ms RTT, and therefore the difference term $\bar{\theta}_S - \bar{\theta}_R$ modulates the behavior of $\theta_S(\tau)$ in determining that of $\Theta_O(\tau)$. If $\theta_S(t) \approx C$, then $\Theta_O(\tau)$ decreases with τ since $\bar{\theta}_R \leq C$ and $\bar{\theta}_S - \bar{\theta}_R > 0$. However, if throughput falls much below C after the ramp-up and has significant random variations when repeated, it is quite possible for $\Theta_O(\tau)$ to increase with respect to τ in certain albeit small regions, for example, lower RTT regions in the UDT profile of Fig. 2.3.

Now, a concave throughput profile with respect to RTT is desirable since throughput is sustained as τ increases. The concave-to-convex transition point for the profiles is associated with an exponential increase in ramp-up followed by sustained throughput, $\theta_S(t) \approx C$; either a slower ramp-up or an unsustained peak throughput can lead to non-concave profiles. In most of the measured throughput profiles shown in the previous subsection, the profile is concave when RTT is small, and at a certain transition point, it becomes and continues to be convex as RTT increases. This behavior is in part a result of various host buffers being sufficiently large to fill up the connection to the near capacity C combined with the fast response of TCP congestion control at lower RTT. Furthermore, it can also be shown qualitatively that adopting larger buffer sizes and/or more parallel streams leads to expanded concave regions and higher overall throughput [20]. In contrast, traditional TCP models, driven primarily by losses, result in throughput profiles having the generic form $\hat{\mathscr{T}}(\tau) = a + b/\tau^c$, for suitable constants a, b, and $c \geq 1$ [26]. These convex profiles (since $\frac{d\hat{\mathscr{T}}}{d\tau} = -b/\tau^2$ increases with τ) are typical of large transfers and longer RTTs and do not adequately account for transfers that lead to concave portions in the measured profiles.

2.4 Dynamics of Throughput Measurements

The throughput traces provide more detailed information about the transfer processes than the mean throughput profiles. As before, we sample the transfer rates at 1-s intervals for a total duration of 100 s. However, in contrast to the experiments described in the previous section, the transfer size here is not fixed, and a higher average throughput indicates a larger transfer size.

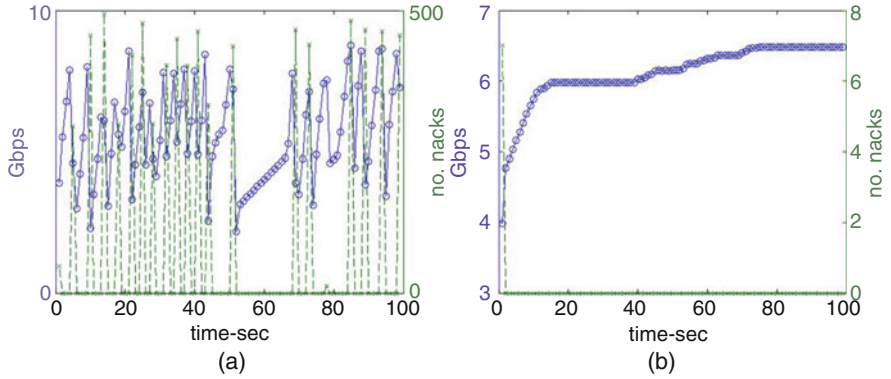


Fig. 2.7 Sample send rate and NACK traces for UDT. (**a**) 22.6 ms RTT, trace 1. (**b**) 22.6 ms RTT, trace 2

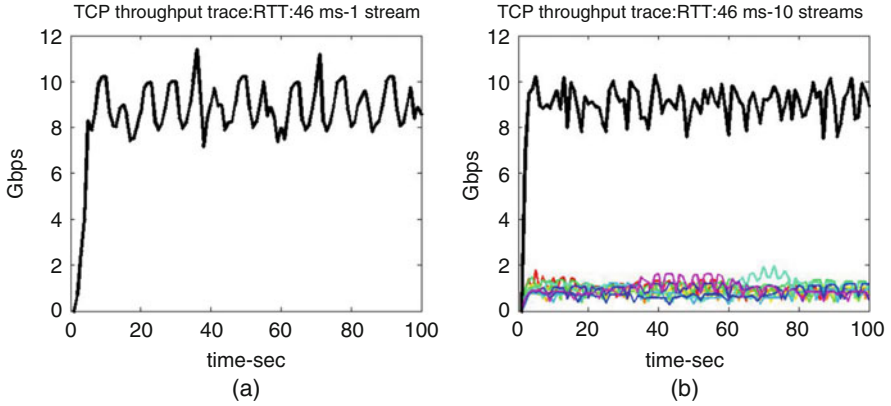


Fig. 2.8 CUBIC throughput traces with large buffers, 45.6 ms RTT, and variable numbers of streams. (**a**) 1 stream. (**b**) 10 streams

Figure 2.7 shows two sample time traces of UDT throughput measurement (along with the number of NACKs) for a 22.6 ms RTT connection. Large NACK counts, often in the hundreds, accompany each drop in the send rate; and such drops can occur frequently, as shown in Fig. 2.7a. In fact, most traces at lower RTTs exhibit such oscillations over time. Nevertheless, there also exist traces, such as that shown in Fig. 2.7b, where losses are few during a 100 s run. With increasing RTTs, such steady behavior becomes more common.

Figure 2.8 shows time traces for CUBIC throughput measurements from [20] over a 45.6 ms RTT connection, with 1 and 10 streams and large buffers. In these plots, the thick black curves describe the aggregate transfer rates, whereas different colored curves correspond to rates for individual streams. As can be seen from these plots, while the per stream transfer rate decreases with more streams, the aggregate rates appear to hover around 9 Gbps, and the transfer size is around 100 GB for most cases.

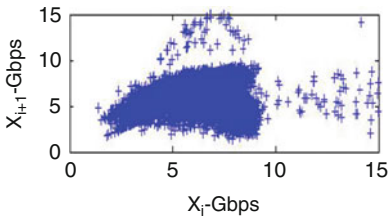


Fig. 2.9 Poincaré map for UDT with 22.6 ms RTT

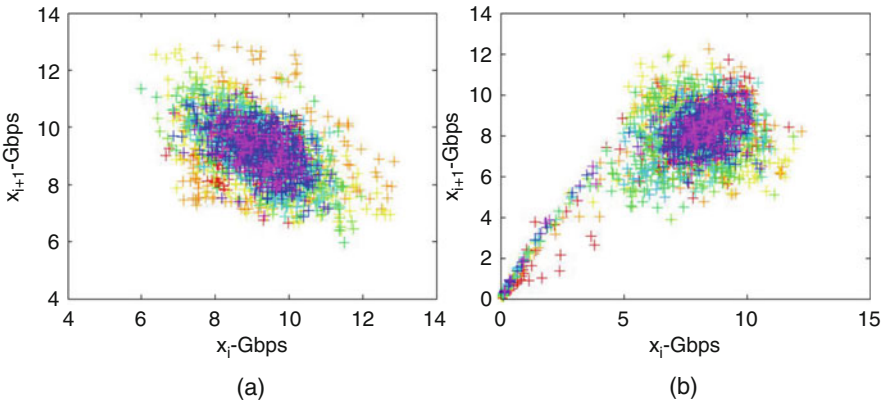


Fig. 2.10 Aggregate Poincaré maps for CUBIC with large buffers and 11.6 or 183 ms RTTs. (**a**) 11.6 ms. (**b**) 183 ms

We can visualize the stability profile of these throughput traces via Poincaré maps. A *Poincaré map* $M : \Re_d \longmapsto \Re_d$ specifies a sequence, called the trajectory, of a real vector state $X_i \in \Re_d$ that is updated at each time step i such that $X_{i+1} = M(X_i)$ [1]. The trajectories generated by a Poincaré map M are characterized by the *Lyapunov exponent*, defined as $\mathscr{L}_M = \ln \left| \frac{dM}{dX} \right|$, which describes the separation of the trajectories that originate from the nearby states. In particular, negative Lyapunov exponents correspond to stable system dynamics [1].

The theoretical Poincaré map of the UDT analytical model is a simple monotonically increasing function that flattens out at peak throughput values [11]. Figure 2.9 shows the Poincaré map for UDT estimated from traces for 22.6 ms RTT. It is evident that the sample points have covered most of the bottom left area here (relative to the (10,10) point), indicating large variations in time traces, as seen in Fig. 2.7a. Nevertheless, the Poincaré map estimated using measurements gradually becomes thinner with increasing RTTs and draws closer to the theoretical shape described above.

In terms of the aggregate transfer rate, in Poincaré maps for CUBIC in Fig. 2.10, each color represents a different stream count, and the points are superimposed on top of one another with varying flow counts, forming a cluster that describe the sustainment stage. In particular, the 183 ms RTT case demonstrates the effect of a longer ramp-up stage by the points from the origin leading up to the cluster, which

is absent in the 11.6 ms RTT case. Interestingly, the "tilts" of the two clusters also appear different: the 183 ms RTT cluster in Fig. 2.10b aligns more with the ideal 45° line, whereas the 11.6 ms RTT cluster in Fig. 2.10a tilts to the left, indicating a less stable profile of the corresponding time traces (even with overall higher mean throughput rates). This analysis is further confirmed by the Lyapunov exponents [20], whose values in the 183 ms RTT case form a more compact cluster closer to the zero line as opposed to the 11.6 ms RTT case. In addition, it was shown that more streams and larger buffers reduce the instability in aggregate transfer rates by pulling the Lyapunov exponents closer to zero (details can be found in [20]).

2.5 File Transfer Measurements

A wide-area disk-to-disk file transfer encompasses storage devices, data transfer hosts, and local- and wide-area connections as illustrated in Fig. 2.11. Major sites often use dedicated data transfer hosts, such as DTNs, with high-performance network interface cards (NICs) to access network connections and Host Channel Adapters (HCAs) to access network storage systems. Transfers also involve a range of software components, including filesystem I/O modules for disk access and the TCP/IP stack for network transport. When filesystems are mounted at individual sites, file transfer software such as GridFTP [5] and XDD [28] running on the hosts is used for transport, and detailed measurements and analyses of XDD transfers using our testbed are presented in [19]. Another method is to mount filesystems over wide-area networks [16, 27], wherein file transfers are handled by the underlying filesystem and are transparent to the user.

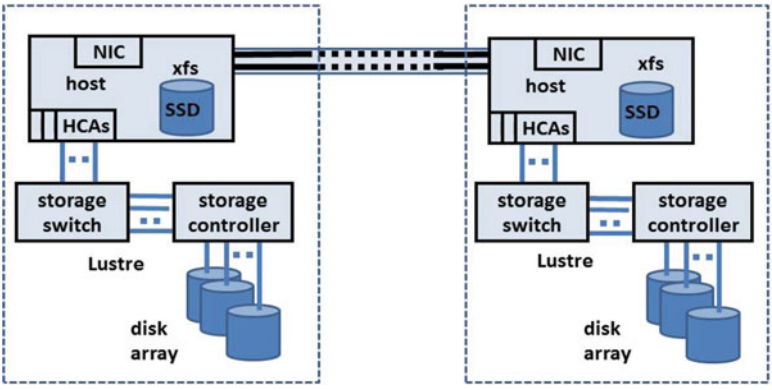


Fig. 2.11 File transfers over long-haul connections

2.5.1 XDD File Transfer Measurements

High-performance disk-to-disk transfers between filesystems at different sites require the composition of complex file I/O and network subsystems and host orchestration. For example, as mentioned earlier, the Lustre filesystem employs multiple OSTs to manage collections of disks, multiple OSSes to stripe file contents, and distributed MDSes to provide site-wide file naming and access. However, sustaining high file transfer rates requires *joint* optimization of subsystem parameters to account for the impedance mismatches among them [24]. For Lustre filesystems, important parameters are the stripe size and number of stripes for the files, and these are typically specified at the creation time; the number of parallel I/O threads for read/write operations is specified at the transfer time. To sustain high throughput, I/O buffer size and the number of parallel threads are chosen to be sufficiently large, and as we will illustrate, this simple heuristic is not always optimal. For instance, wide-area file transfers over 10 Gbps connections between two Lustre filesystems achieve transfer rates of only 1.5 Gbps, when striped across 8 storage servers, accessed with 8 MB buffers, and with 8 I/O and TCP threads [19], even though peak network memory-transfer rate and local file throughput are each close to 10 Gbps.

We measured file I/O and network throughput and file transfer rates over Lustre and XFS filesystems for a suite of seven emulated connections in the 0–366 ms RTT range, which are used for memory-transfer measurements in Sect. 2.3. We collected two sets of XDD disk-to-disk file transfer measurements, one from XFS to XFS and one from Lustre to Lustre, and each experiment was repeated ten times. We considered both buffered I/O (the Linux default) and direct I/O options for Lustre. In the latter, XDD avoids the local copies of files on hosts by directly reading and writing into its buffers, which significantly improves the transfer rates. Results based on these measurements are summarized in [19]: (a) strategies of large buffers and higher parallelism do not always translate into higher transfer rates; (b) direct I/O methods that avoid file buffers at the hosts provide higher wide-area transfer rates, and (c) significant statistical variations in measurements, due to complex interactions of nonlinear TCP dynamics with parallel file I/O streams, necessitate repeated measurements to ensure confidence in inferences based on them.

We first consider XFS-to-XFS transfers. From the aggregate mean throughput line plot in Fig. 2.12a, we can see that throughput increases with the number of flows. For instance, whereas the mean throughput peaks at 5 Gbps with 1 flow, the peak (occurring with 0.4 ms RTT) rapidly jumps to above 9 Gbps with 4 flows, even closely approaching 10 Gbps with 7 flows. Mean throughput generally decreases with RTT, consistent with most data transfer protocols. The surface plots in Fig. 2.12b indicate a *monotonically increasing* trend.

In the default I/O Lustre setup, the number of flows varies from 1 to 8, and the number of stripes is either 2 or 8. Figure 2.13 shows the default I/O Lustre mean throughputs. Compared to XFS, the overall throughput here is much lower, especially for lower RTTs (such differences become less pronounced as RTT

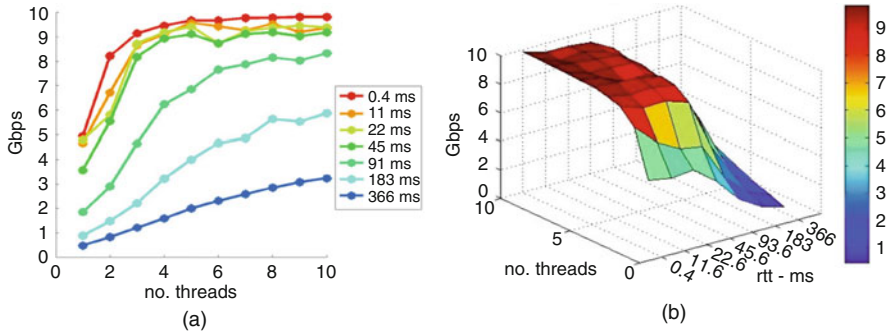


Fig. 2.12 Mean XFS file write rates. (**a**) Write rates: line plot. (**b**) Write rates: surface plot

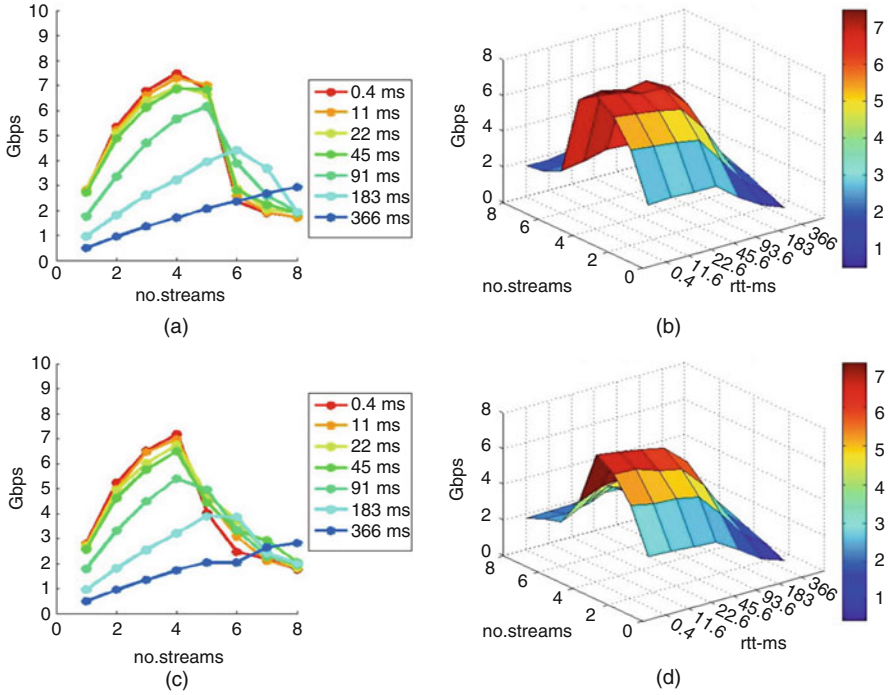


Fig. 2.13 Mean default I/O Lustre file write rates. (**a**) Two stripes: line plot. (**b**) Two stripes: surface plot. (**c**) Eight stripes: line plot. (**d**) Eight stripes: surface plot

increases). For example, at lower RTTs, the mean throughput peaks at four flows, starts to decrease with five flows, and takes a nosedive at six flows. The sharp drop is delayed at higher RTTs, with throughput peaking at five flows for 91 ms RTT and at six flows for 183 ms RTT and increasing all the way through eight flows for 366 ms RTT. The line plots in Fig. 2.13a, c show that the sharp drop in throughput, if any,

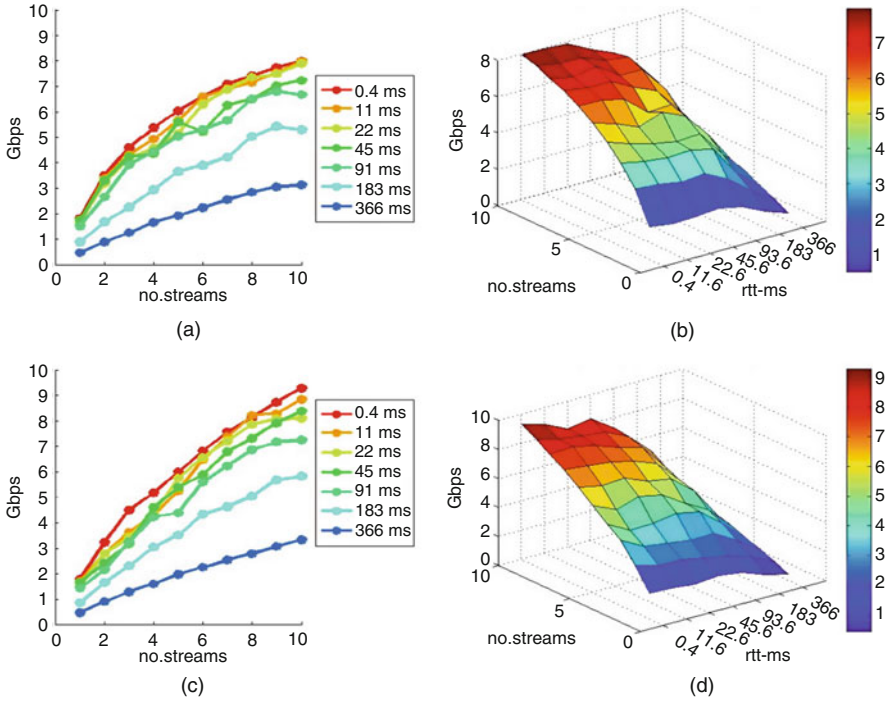


Fig. 2.14 Mean direct I/O Lustre file write rates. (**a**) Two stripes—line plot. (**b**) Two stripes—surface plot. (**c**) Eight stripes—line plot. (**d**) Eight stripes—surface plot

occurs earlier, at five flows, when eight stripes are used instead of two stripes. This result is also confirmed by the surface plots: the "dome" in Fig. 2.13d is narrower than that in Fig. 2.13b.

We repeated the above experiments using direct I/O Lustre. The mean throughput plots in Fig. 2.14 show the overall monotonic trend as observed in the XFS results. In particular, single-flow throughput is lower than that of default I/O Lustre, but the throughput steadily increases with the number of flows. Using two stripes yields somewhat higher transfer rates compared to eight stripes for lower flow counts. With more flows, overall throughput is higher, and eight stripes is the better option. Although the peak rates with ten flows and eight stripes are lower compared to XFS, they are significantly higher than that with default I/O. In addition, the XFS surface in Fig. 2.12b is higher and increases faster at lower RTTs compared to those of the direct I/O Lustre in Fig. 2.14b, d, as a result of rates starting higher with one flow and further improving and approaching the peak with additional flows.

In summary, file transfer rates for both XFS and Lustre are affected by the number of flows. The mean transfer rate increases monotonically with the number of flows for both XFS and direct I/O Lustre but decreases beyond a certain point for default I/O Lustre. The number of stripes in Lustre seems to have less impact on the

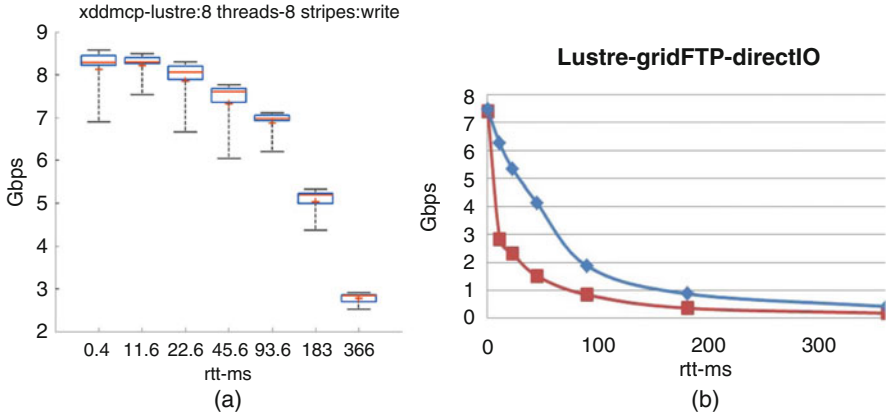


Fig. 2.15 Throughput profiles of XDD and GridFTP with direct I/O. (**a**) XDD. (**b**) GridFTP

transfer rate than does the number of flows. Additional measurements with request sizes of 65 and 145 MB show that the default 8 MB selection used here consistently yields the best performance. This result is expected, as the smaller request size provides finer-resolution data chunks to TCP streams.

In addition, we collected some preliminary measurements using GridFTP [5] under the same configurations used for XDD measurements. While more comprehensive measurements are being collected, our initial results show that the overall throughput profiles of GridFTP are somewhat lower compared to XDD and, in contrast to xdd, are convex under all configurations studied so far. A typical cease is illustrated in Fig. 2.15 for 10 GB file transfers using direct I/O options for both XDD and GridFTP. The profiles of GridFTP in Fig. 2.15b are higher with 8 parallel streams (shown in blue) and somewhat "less" convex, and the peak throughput of 7.5 Gbps is achieved in both 8 streams and single stream (shown in red) cases. Comparatively, xdd profiles are concave and can achieve the peak throughput above 8 Gbps as shown in Fig. 2.15a.

2.5.2 Lustre Over Wide-Area Networks

There have been several works that extend Lustre filesystems over wide-area networks providing a number of desired features: (1) file transfers are natively supported by the copy operation, which obviates the need for file transfer tools, such as XDD, GridFTP, and Aspera [2] and (2) applications involving file operations can be supported transparently over wide-area networks. Typical site Lustre filesystems are mounted over IB networks, which are subject to 2.5 ms latency limitation of the IB flow control method; hardware-based IB range extenders can be used to overcome this limitation, although such a solution is not scalable due to the high

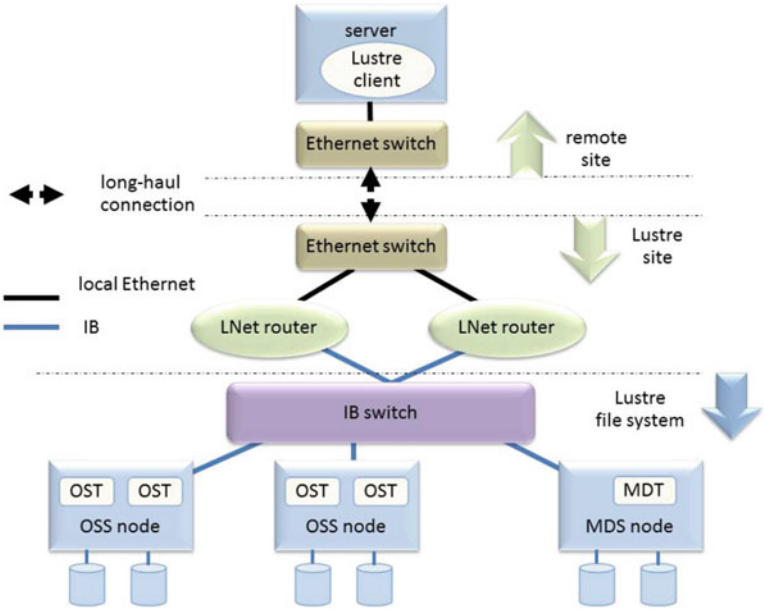


Fig. 2.16 Lustre over long-haul connections using LNet routers between local IB and Ethernet

cost and need to bookend each long-haul connection with a pair of such extenders. In this subsection, we present initial results that utilize LNet routers (implemented on Linux hosts) to extend Lustre over long-haul links with 0–366 ms RTT, without requiring any additional hardware. Previous results are limited to relatively shorter distances and somewhat limited range of latencies [8, 14].

We utilized Ethernet clients on remote servers to mount Lustre and used LNet routers at Lustre site to route between site IB network and Ethernet LAN connected to WAN, as shown in Fig. 2.16. Under this scheme, no changes to the site Lustre system over IB network are needed, and hosts connected to both IB network and site Ethernet LAN are configured as LNnet routers. Initial proof-of-principle throughput measurements using IOzone writes are shown in Fig. 2.17 for 10GigE emulated connections between two pairs of hosts, namely, 48-core stand-alone systems used for data transfers and 32-core hosts which are part of a compute cluster. The hosts are configured to use HTCP, and their buffer sizes are set at largest allowable values. These throughput profiles are lower than the corresponding iperf profiles as expected, and the differences between the two types of hosts are within 10%. It is striking that the profiles are convex, indicative of buffer limits. Further investigations, including additional test configuration and examination of LNet routers configuration, are needed to improve the throughput performance.

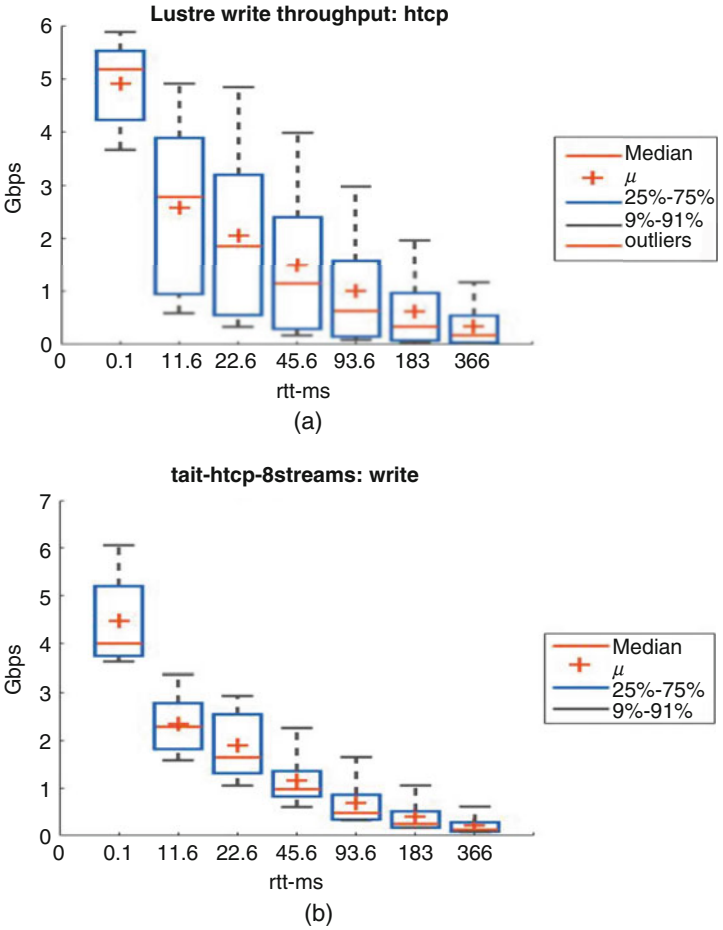


Fig. 2.17 Throughput profiles of wide-area Lustre. (**a**) 48-core stand-alone hosts. (**b**) Physical and emulated connections between hosts

2.6 Conclusions and Open Areas

In HPC and cloud computing scenarios, wide-area data transfers have been increasingly performed over dedicated network connections, driven in part by the promise of high throughput and stable transport dynamics. To study the performance of UDT and TCP variants and the impact of their parameters for such transfers, we established a testbed and systematically collected measurements using physical and emulated dedicated connections. In this chapter, we have provided a consolidated view of the results of these studies, some of which have been described separately in detail in previous publications (throughput profile [18], XDD measurements [19], UDT dynamics [11], TCP dynamics [20], LNet Lustre extension [21]), and also

reported on initial results on GridFTP measurements on dedicated connections with 0–366 ms RTT. Our results reveal important concave-convex throughput profiles and rich transport dynamics as indicated by the Poincaré map and Lyapunov exponents. These measurements and analyses enable the selection of an ideal high throughput transport method and parameters for a specific RTT range.

Further research advances are required before we can fully optimize data transfers over dedicated connections. We require (1) detailed analytical models that closely match the measurements of file and disk I/O throughput profiles and (2) enhanced first-principle UDT and TCP models to explain the dual-mode throughput profiles by integrating dynamics parameters, such as the Lyapunov exponents. Further measurements and analyses can provide more insights into the performance of BBR transport over dedicated connections, particularly when end systems are dissimilar in their computing, network, and I/O capacities. It is also of future interest to pursue learning-based approaches [12, 23] to quantify the effect of various factors on throughput performance and develop SDN methods that use throughput profiles to select and set up suitable paths to match the transport protocols. Detailed experimentation with the LNet-based wide-area Lustre filesystem and investigation of its performance optimization are also of future interest. In general, the joint optimization of I/O, host, and network parameters to achieve peak and stable wide-area data transfers continues to be a challenge.

Acknowledgements This work is supported in part by the RAMSES and Net2013 projects, Office of Advanced Computing Research, US Department of Energy, and by the Extreme Scale Systems Center, sponsored by US Department of Defense, and performed at Oak Ridge National Laboratory managed by UT-Battelle, LLC for US Department of Energy under Contract DE-AC05-00OR22725.

References

1. Alligood, K.T., Sauer, T.D., Yorke, J.A.: Chaos: An Introduction to Dynamical Systems. Springer, Reading, MA (1996)
2. Aspera high-speed file transfer. http://www-03.ibm.com/software/products/en/high-speed-file-transfer
3. Cardwell, N., Cheng, Y., Gunn, C.S., Yeganeh, S.H., Jacobson, V.: BBR: congestion based congestion control. ACM Queue **14**(5), 1–34 (2016)
4. Energy Sciences Network. http://www.es.net
5. GT 4.0 GridFTP. http://www.globus.org
6. Gu, Y., Grossman, R.L.: UDT: UDP-based data transfer for high-speed wide area networks. Comput. Netw. **51**(7) (2007)
7. Hassan, M., Jain, R.: High Performance TCP/IP Networking: Concepts, Issues, and Solutions. Prentice Hall, Upper Saddle River, NJ (2004)
8. Henschel, R., Simms, S., Hancock, D., Michael, S., Johnson, T., Heald, N., William, T., et al.: Demonstrating lustre over a 100 Gbps wide area network of 3,500 km. In: High Performance Computing, Networking, Storage and Analysis (SC), 2012 International Conference on, pp. 1–8 (2012)

9. Jain, S., Kumar, A., Mandal, S., et al.: B4: experience with a globally-deployed software defined Wan. SIGCOMM Comput. Commun. Rev. **43**(4), 3–14 (2013)
10. Kelly, T.: Scalable TCP: Improving performance in high speed wide area networks. Comput. Commun. Rev. **33**(2), 83–91 (2003)
11. Liu, Q., Rao, N.S.V., Wu, C.Q., Yun, D., Kettimuthu, R., Foster, I.: Measurement-based performance profiles and dynamics of udt over dedicated connections. In: International Conference on Network Protocols, Singapore (2016)
12. Liu, Z., Balaprakash, P., Kettimuthu, R., Foster, I.T.: Explaining wide area data transfer performance. In: ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC), Washington, DC (2017)
13. Mathis, M., Semke, J., Mahdavi, J., Ott, T.: The mascroscopic behavior of the TCP congestion avoidance algorithm. Comput. Commun. Rev. **27**(3), 62–82 (1997)
14. Michael, S., Zhen, L., Henschel, R., Simms, S., Barton, E., Link, M.: A study of Lustre networking over a 100 gigabit wide area network with 50 milliseconds of latency. In: Proceedings of the fifth international workshop on Data-Intensive Distributed Computing, pp. 43–52 (2012)
15. Padhye, J., Firoiu, V., Towsley, D.F., Kurose, J.F.: Modeling TCP Reno performance: a simple model and its empirical validation. IEEE/ACM Trans. Networking **8**(2), 133–145 (2000)
16. Palencia, J., Budden, R., Sullivan, K.: Kerberized Lustre 2.0 over the WAN. In: Proceedings of the 2010 TeraGrid Conference (2010)
17. Rao, N.S.V., Gao, J., Chua, L.O.: On dynamics of transport protocols in wide-area internet connections. In: Complex Dynamics in Communication Networks. Springer, Berlin (2005)
18. Rao, N.S.V., Towsley, D., Vardoyan, G., Settlemyer, B.W., Foster, I.T., Kettimuthu, R.: Sustained wide-area TCP memory transfers over dedicated connections. In: IEEE International Conference on High Performance and Smart Computing, New York, NY (2015)
19. Rao, N.S.V., Liu, Q., Sen, S., Hinkel, G., Imam, N., Settlemyer, B.W., Foster, I.T., et al.: Experimental analysis of file transfer rates over wide-area dedicated connections. In: 18th IEEE International Conference on High Performance Computing and Communications (HPCC), Sydney, Australia, pp. 198–205 (2016)
20. Rao, N.S.V., Liu, Q., Sen, S., Henley, J., Foster, I.T., Kettimuthu, R., Towsley, D., Vardoyan, G.: TCP throughput profiles using measurements over dedicated connections. In: ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC), Washington, DC (2017)
21. Rao, N.S.V., Liu, Q., Sen, S., Towsley, D., Vardoyan, G., Foster, I.T., Kettimuthu, R.: Experiments and analyses of data transfers over wide-area dedicated connections. In: The 26th International Conference on Computer Communications and Network(ICCCN 2017), Vancouver, Canada (2017)
22. Rhee, I., Xu, L.: CUBIC: a new TCP-friendly high-speed TCP variant. In: Proceedings of the Third International Workshop on Protocols for Fast Long-Distance Networks (2005)
23. Sen, S., Rao, N.S.V., Liu, Q., Imam, N., Foster, I.T., Kettimuthu, R.: On analytics of file transfer rates over dedicated wide-area connections. In: First International Workshop on Workflow Science (WOWS) in conjunction with 13th IEEE International Conference on e-Science, Auckland, New Zealand, (2017)
24. Settlemyer, B.W., Dobson, J.D., Hodson, S.W., Kuehn, J.A., Poole, S.W., Ruwart, T.M.: A technique for moving large data sets over high-performance long distance networks. In: IEEE 27th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1–6 (2011)
25. Shorten, R.N., Leith, D.J.: H-TCP: TCP for high-speed and long-distance networks. In: Proceedings of the Third International Workshop on Protocols for Fast Long-Distance Networks (2004)

26. Srikant, Y., Ying, L.: Communication Networks: An Optimization, Control, and Stochastic Networks Perspective. Cambridge University Press, Cambridge (2014)
27. Walgenbach, J., Simms, S.C., Westneat, K., Miller, J.P.: Enabling Lustre WAN for production use on the TeraGrid: a lightweight UID mapping scheme. In: Proceedings of the 2010 TeraGrid Conference (2010)
28. XDD - The eXtreme dd toolset. https://github.com/bws/xdd
29. Yee, T., Leith, D., Shorten, R.: Experimental evaluation of high-speed congestion control protocols. Trans. Netw. **15**(5), 1109–1122 (2007)

Chapter 3
Temporal Analysis of Stress Classification Using QRS Complex of ECG Signals

Neha Keshan, Isabelle Bichindaritz, Patanjali V. Parimi, and Vir V. Phoha

Abstract This paper demonstrates that electrocardiogram (ECG) signals can be used to detect and classify stress in a person using as low as 5 s data stream. The analysis focuses on determining the minimum ECG data required to classify stress levels. Time taken to detect level of stress can be crucial to prevent cardiac arrest and other abnormalities. The ECG data of 10 drivers driving through different traffic conditions is segmented into 60, 30, 20, 15, 10, and 5 s instances. Two levels of stress, low and high, and features from Q-, R-, and S-fiducial points and classifiers such as Naïve Bayes, logistic, multilayer perceptron, SMO (sequential minimal optimization), IB1 (nearest neighbor), J48 (decision tree), and random forest are used for experiments. The results show that stress can be identified with high accuracy. It is found that even a 5 s data stream provides an 87.98% accuracy using random forest twofold cross validation test, opening the door for rapid stress detection.

Keywords Electrocardiogram signals · Temporal analysis · Stress classification · Naive Bayes classifier · Multilayer perceptron · Decision tree · Random forest · Cross validation

N. Keshan (✉)
Electrical Engineering and Computer Science Department, Syracuse University, New York, NY, USA

Advanced Wireless Systems Research Center, State University of New York at Oswego, New York, NY, USA
e-mail: nkeshan@syr.edu

I. Bichindaritz
Computer Science Department, State University of New York at Oswego, New York, NY, USA

P. V. Parimi
Advanced Wireless Systems Research Center, State University of New York at Oswego, New York, NY, USA

V. V. Phoha
Electrical Engineering and Computer Science Department, Syracuse University, New York, NY, USA
e-mail: vvphoha@syr.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_3

3.1 Introduction

With the proliferation of wearable devices that can interact with the human body and monitor physiological signals, physicians and companies alike are looking for ways to take advantage of these devices for providing better health care and diagnosis. Stress in the human body is key factor of many abnormalities and is related to multiple health conditions. According to Sudden Cardiac Arrest Foundation, chronic job stress sharply increases the likelihood of having a "heart attack." Every minute generates some amount of stress, either eustress or distress, in an individual. If the individuals are not alerted in time, this stress may lead to cardiac arrests.

Studies have been carried out that show electrocardiogram (ECG) can be used as one of the physiological signals to detect stress [1–5, 9–11]. For ECG signals (Fig. 3.1), stress classification is carried out based on either fiducial (information local to individual beat) or non-fiducial (information on the whole ECG trace) points or both.

Healey and Picard [8] have carried out multisensor stress analysis employing data from EMG, respiration, instantaneous heart rate (extracted from ECG), and galvanic skin response (GSR) as well as additional features. These authors have done intra-level stress detection and found a predictive accuracy of 97.4% with leave-one-out cross validation for high stress and 94.7% for moderate stress.

Another study by Akbas [1] calculated the differences between the three stress levels of averaged feature values extracted and average number of contractions per minute for all the signals.

Deng et al. [3, 4] have investigated stress using principal component analysis and found that foot GSR duration, hand GSR duration, hand GSR area, foot GSR area, and foot GSR frequency were key features of this dataset. Later in 2013 [5], they have pursued their research by combining feature selection with signal selection. With tenfold CV (averaged 6 times), they obtained accuracy of 74.5% with SVM using all features and 85.46% with C4.5 using 5 features described above. However, they have used only 65 data samples.

Singh and Queyam [10, 11] have used signal fusion of multiple sensor data for stress classification. They have reported accuracy of 80% in detecting stress of 6 out of 10 drivers using neural networks.

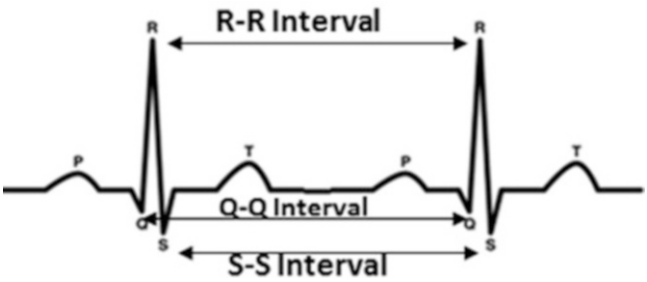


Fig. 3.1 ECG signal depicting different fiducial points and RR, QQ, and SS intervals

Recently, Keshan et al. [9] have used the QRS complex of ECG to classify between 3 levels of stress and report a predictive accuracy of 88.24% for 3 stress levels.

3.2 Significance of the Work

In all the aforementioned studies, time period of the ECG data required for classification analysis was not a factor and longtime period data of 1 min to 15 min was used [1–5, 9–11], which highly limits the applicability of ECG-based device to detect and prevent cardiac and other stress-related health conditions. Also, previous investigations focused on the derivative of the ECG, which is instantaneous heart rate, for the stress classification analysis. Today with commercially available devices, 24/7 ECG data recording is possible. These data if used for identification of the level of stress would provide new pathways in health monitoring and diagnosis. Previously, a patient or user was needed to go to a clinic and subjected to a workout on a treadmill or such device to get the stress tested under the supervision of a trained clinician. If the level of stress is identified quickly with wearable devices, many of the problems associated with clinical test can be avoided, and 24/7 real-time stress level data can be provided to the physicians. Also, using multiple physiologic signals [1–5, 10, 11] for stress classification is a cumbersome and time-taking task. In most situations, specifically wearable devices may not have all the different sensors required for such multisensor stress detection. The use of only QRS complex of ECG for stress classification also decreases the overall computation time while increasing the quick classification process.

Table 3.1 compares our work with previous studies done on stress detection using the same dataset (discussed in Sect. 3.3).

Table 3.1 Comparison of our work with previous works. ECG, EMG, and GSR stands for electrocardiogram, electromyography and galvanic skin response, respectively

	Signals used	Fiducial points	Accuracy
Healey and Picard [7]	ECG, EMG, respiration, GSR	Instantaneous heart rate	97.4% for high stress
Akbas [1]	ECG, EMG, respiration, GSR	Instantaneous heart rate	No classification performed
Deng et al. [3, 4]	GSR	NA	78.46%
Deng et al. [5]	GSR	NA	74.5–85.46%
Singh and Queyam [10, 11]	ECG, EMG, respiration, GSR	Instantaneous heart rate	Overall 80% on 6 out of 10 drivers
Keshan et al. [9]	ECG	QRS complex	88.24% full period instances
Our	ECG	QRS complex	87.98–100% using 5 s instances

3.3 ECG Data and QRS Wave Complex

ECG signal consists of at least five waves in a normal person, P-, Q-, R-, S- and T waves. QRS complex results from the depolarization of the ventricles and is usually the strongest part of the ECG signal. The Q wave corresponds to the depolarization of interventricular septum and is the first initial downward or "negative" deflection. The R wave corresponds to the depolarization of the main mass of the ventricles and is the next upward deflection (provided it crosses the isoelectric line and becomes "positive"). The S wave corresponds to the last phase of the ventricular depolarization at the base of the heart and is then the next deflection downward, provided it crosses the isoelectric line to become briefly negative before returning to the isoelectric baseline.

The data used for present study is obtained from PhysioNet [6] in section "Stress Recognition in Automobile Drivers." This dataset consists of multisensor signals (ECG, EMG, foot GSR, hand GSR, respiration, and heart rate) of 17 drivers. Based on our observation of previous studies [1–5, 9–11] using this dataset, we have chosen stress data of five drivers (Table 3.2). The data of these five drivers seem to be well recorded and free from noise.

Since we need to compare results to investigate the minimum time interval needed for detection of stress, we have considered 5 min intervals from each driving section. These are shown in Table 3.2. Here, the assumption is that the stress induced in an individual is due to the traffic conditions only.

For our study, we considered two cases—low stress and high stress. Based on the experimental setup explained by Healy [8], we consider the initial rest and final rest data as low stress and city1, city2, and city3 (Table 3.2) as high stress data. Data are annotated using Cygwin and predefined commands in PhysioNet [6]. A customized Java program is run on the annotated files to extract features described in Table 3.3 and create Attribute-Relation File Format (ARFF) file. This ARFF file is then used in Weka [7] for classification between low and high stress.

Table 3.2 Time interval in minutes of the five driving segments corresponding to five drivers obtained from available multiparameter dataset

Drivers	IR	C1	C2	C3	FR
DRIVE06	5–10	19–24	38–43	55–60	70–75
DRIVE08	5–10	19–24	39–44	55–60	70–75
DRIVE10	5–10	19–24	40–45	55–60	70–75
DRIVE11	5–10	19–24	37–42	55–60	70–75
DRIVE12	5–10	19–24	37–42	55–60	70–75

Here IR and FR represent the initial rest and the final rest and correspond to low stress data. C1, C2, and C3 stand for city1, city2, and city3, respectively, and correspond to the high stress data

3.4 Stress Classification Analysis

Through our discussion with a physician, we have learnt that R-R interval is an important feature in the ECG data for stress detection. With an increase in stress, the frequency of the pulse consisting PQRST waves also increases. This corresponds to shortening of the fiducial point distances, prominently the R-R interval. We have applied this information as the base for our study to investigate QRS complex alone.

For our experiment, we have employed seven classifiers—Naïve Bayes (NB), logistic (LO), neural network or multilayer perceptron (MP), sequential minimal optimization (SMO), nearest neighbor (IB1), decision tree (J48), and random forest (RF). We report the results for each classifier for four different tests: training set, onefold cross validation, twofold cross validation, and tenfold cross validation. To determine the smallest time interval required for various levels of stress detection, we looked at 5 sets of 5 min intervals for each driver (Table 3.2). These data were then segmented into 60 s, 30 s, 20 s, 15 s, 10 s, and 5 s instances for analysis. In other words, for driver 6 the 5 min data for initial rest, city1, city2, city3, and final rest is segmented in to 1 min instances for the first experiment, 30 s instances for the second experiment, and 5 s instances for the sixth and final experiment. This was repeated for the other four drivers. Thus, our dataset for the first experiment consists of 125 one-minute instances of 5 drivers and 2 classes—low and high stress. All the 14 extracted features (Table 3.3) were used. Table 3.4 shows the results for 1 min instances. It is seen that level of stress can be detected with 84.8% accuracy

Table 3.3 Extracted features where first seven features are extracted directly from the annotated QRS while the rest seven features are extracted by taking the difference between an individual's high stress and low stress periods

Features	
• Average QRS interval	• Average difference beats
• Average RR interval	• Average difference QRS
• Average QQ interval	• Average difference RR
• Average SS interval	• Average difference QQ
• Average QR interval	• Average difference SS
• Average RS interval	• Average difference QR
• Average beats	• Average difference RS

Table 3.4 ROC area (ROC) and accuracy percentage (ACC) for a dataset of 25 one-minute low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.86	77.6	0.83	73.6	0.84	71.2	0.83	74.4
LO	0.91	82.4	0.81	74.4	0.74	72.8	0.82	74.4
MLP	0.97	94.4	0.84	82.4	0.81	76.8	0.84	81.6
SMO	0.79	78.4	0.79	77.6	0.77	76	0.79	77.6
IB1	1	100	0.79	80.8	0.75	75.2	0.78	80
J48	0.92	92	0.75	84.8	0.79	82.4	0.81	83.2
RF	1	100	0.92	84.8	0.88	79.2	0.91	84

Table 3.5 ROC area and accuracy percentage for a dataset of 50 thirty seconds low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.86	77.2	0.85	76	0.84	75.2	0.85	76.4
LO	0.90	83.6	0.86	81.2	0.85	80.4	0.86	81.2
MLP	0.95	91.6	0.88	85.2	0.88	83.2	0.89	85.2
SMO	0.81	81.2	0.81	80.4	0.82	81.6	0.81	80.8
IB1	1	100	0.81	82	0.83	83.6	0.83	83.2
J48	0.97	94.8	0.88	82	0.85	84.4	0.82	84.4
RF	1	100	0.93	85.6	0.94	87.2	0.93	86.4

Table 3.6 ROC area and accuracy percentage for a dataset of 75 twenty seconds low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.84	75.2	0.83	74.13	0.83	74.4	0.84	74.4
LO	0.88	84.27	0.85	81.6	0.85	80.8	0.86	81.6
MLP	0.95	90.1	0.88	83.2	0.87	82.7	0.89	84
SMO	0.81	81.1	0.80	80.5	0.80	80.8	0.80	80
IB1	1	100	0.84	85.1	0.79	80.3	0.83	84.5
J48	0.91	91.5	0.91	87.2	0.76	81.1	0.83	85.9
RF	1	100	0.90	84.53	0.88	81.8	0.90	84.8

with onefold cross validation using J48 and RF. On training set both IB1 and RF gives 100% accuracy as expected. The Receiver Operating Characteristic (ROC) area ranges from 0.74 to 1.

A physician may be able to detect the stress level in an individual with 1 min ECG data. To find the minimum duration of the ECG signal required for accurate stress analysis, we carry out 5 more experiments with 30 s, 20 s, 15 s, 10 s, and 5 s instances each in the similar fashion.

The 30 s dataset consists of 250 instances and all the extracted features. Here, the results (Table 3.5) show a slight improvement in overall accuracy percentage. Two levels of stress were identified with 87.2% accuracy when RF twofold cross validation is used. The ROC area range has also improved to 0.81–1.

The 20 s dataset consists of 375 instances and all the extracted features. Here, the results (Table 3.6) echo the previous one for accuracy percentage. Two levels of stress could be identified with 87.2% accuracy. This is possible if J48 onefold cross validation is used. The ROC area ranges between 0.76 and 1.

The 15 s dataset consists of 500 instances and all the extracted features and resulted in accuracy percentage of 86.6% for RF onefold and twofold cross validation as shown in Table 3.7. The ROC area range has increased to 0.81–1.

Table 3.7 ROC area and accuracy percentage for a dataset of 100 fifteen seconds low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.82	71.1	0.81	70.1	0.80	71.1	0.81	70.7
LO	0.86	81	0.84	79	0.84	80	0.84	78.8
MLP	0.96	88	0.88	85.8	0.86	83.2	0.85	85.2
SMO	0.80	80.2	0.80	79.4	0.79	79	0.80	79.4
IB1	1	100	0.83	83.2	0.82	83	0.83	83.4
J48	0.95	91.6	0.81	81.8	0.86	84.8	0.84	84.4
RF	1	100	0.93	86.57	0.92	86.57	0.93	86.17

Table 3.8 ROC area and accuracy percentage for a dataset of 150 ten seconds low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.79	72.13	0.79	71.33	0.79	71.2	0.79	71.07
LO	0.87	81.47	0.85	80.4	0.85	79.47	0.85	80.53
MLP	0.93	89.2	0.85	84.5	0.85	80.8	0.87	83.6
SMO	0.78	78.27	0.78	78.13	0.78	77.6	0.78	75.13
IB1	1	99.87	0.82	82.93	0.81	81.33	0.82	82.4
J48	0.96	90.13	0.89	85.87	0.87	84.67	0.87	84
RF	1	99.87	0.92	85.2	0.91	84.67	0.91	85.33

Table 3.9 ROC area and accuracy percentage for a dataset of 300 five seconds low and high stress instances for each of the 5 drivers

	Training		Onefold		Twofold		Tenfold	
	ROC	ACC	ROC	ACC	ROC	ACC	ROC	ACC
NB	0.84	73.8	0.84	73.59	0.84	73.59	0.84	73.73
LO	0.91	83.46	0.9	82.77	0.90	83.25	0.90	83.46
MLP	0.94	90.69	0.92	86.13	0.92	86.17	0.92	85.34
SMO	0.83	81.51	0.82	80.96	0.82	81.03	0.82	81.17
IB1	1	99.86	0.82	82.90	0.82	82.56	0.82	82.84
J48	0.97	91.8	0.90	87.07	0.88	85.89	0.91	87.21
RF	1	99.86	0.94	87.35	0.94	87.98	0.94	87.14

For 10 s datasets with 750 instances, overall accuracy decreases compared to the previous results. Highest accuracy acquired is 85.87% with J48 onefold cross validation (Table 3.8). ROC area ranges between 0.78 and 1.

Finally, for 5 s instances with 1500 instances, the accuracy percentage obtained is 88% with RF twofold cross validation. ROC area goes from 0.82 to 1 (Table 3.9).

Less than 5 s datasets were not considered as short-time duration datasets do not contain reasonable number of pulses required for the analysis. For example, a normal human being would have 4–7 ECG pulses or QRS peaks in 5 s data. For such

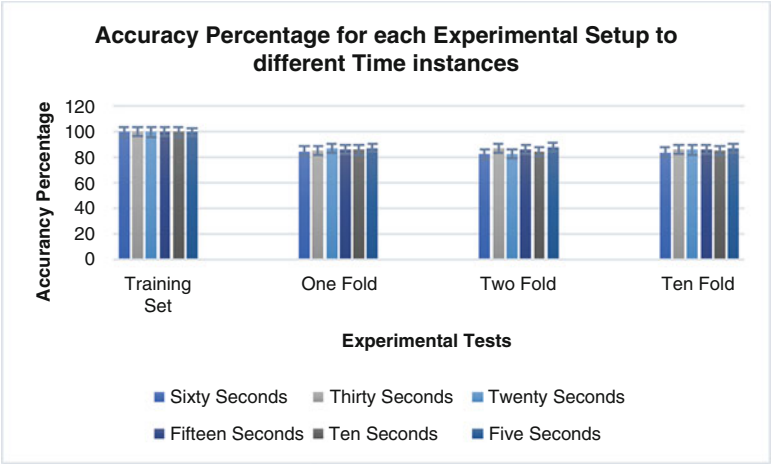


Fig. 3.2 The accuracy percentage (rounded off to two decimal places) and error bars for each of the training set, onefold cross validation (CV), twofold CV, and tenfold CV. This is plotted for experiments on 5, 10, 15, 20, 30, and 60 s data. The graph shows that accuracy percentage is best obtained with 5 s interval when taken cross validation experiments into consideration

small number pulses, training the system is difficult. Secondly, because of noise and other external factors, it becomes difficult to record the signal accurately.

The best obtained accuracy percentage for each of the four experimental setups and all six different time instances are plotted (Fig. 3.2). In cross validation the complete set is divided into n-parts. Leaving one part, the rest $(n - 1)$ parts are considered as training set. The part that is left is then used as the test set. This is carried out for all the parts, and the result is averaged, which corresponds to the real-world scenario where a new data is added to the trained dataset to detect stress level. This makes the cross validation results more important than the results on the training set itself.

From Fig. 3.2, for every cross validation setup, 5 s instance dataset gives the best results. For every cross validation setup, there is an increase in accuracy percentage by 2–4% from 60 s to 5 s dataset. Maximum accuracy percentage for training set for each setup is approximately 100% which is expected.

3.5 Discussion and Conclusions

We have used 15 min of high stress (city1, city2, and city3) and 10 min of low stress (initial rest and final rest) data. When these are divided into small time instances, the classes became imbalanced. The results might be improved if the analysis is done on balanced data. We could also carry out some feature selection algorithms on the extracted features to improve the results.

Through our analysis we can conclude that the 5 s QRS complex data of the ECG signal is sufficient to detect various levels of stress. In this way both the computation time and cost of computation decrease. The other waves of the ECG signal, P and T, seem to not be needed for stress detection. Using only the 5 s annotated data of QRS complex, our system can detect stress with 88% accuracy. A reason for getting better results with QRS alone is that the QRS complex corresponds to the depolarization of the right and left ventricles of the human heart and hence has the highest amplitude and normally lasts 60–90 ms. Mean P and T wave durations are >150 ms, and their amplitudes are small resulting in low signal to noise ratio. Note that long duration signals with small amplitude are prone to stochastic noise. In addition, P and T waves tend to overlap leading to offset of T wave of a pulse mixing up with the onset of P wave of the successive pulse. The QRS complex due to its short duration and significantly high amplitude has high SNR ratio and thereby results in accurate detection of stress.

Therefore, we conclude that multisensor data fusion and stress analysis is not required. This reduces the number of sensors an individual needs to wear and cost of such devices. The level of stress in an individual can be obtained with only 5 s data with the accuracy percentage being 87.98%. This is first time an efficient 5 s ECG-based stress detection is thoroughly addressed. Thus, the present research opens the door for miniature, low-cost ECG-based wearable devices for rapid health monitoring systems and finds applications in smart watches and other portable devices.

Acknowledgments This research was supported in part by National Science Foundation Award SaTC Number: 1527795.

References

1. Akbas, A.: Evaluation of the physiological data indicating the dynamic stress level of drivers. Sci. Res. Essays. **6**(2), 430–439 (2011)
2. Begum, S.: Intelligent driver monitoring systems based on physiological sensor signals: a review. In: 2013 16th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE, pp. 282–289 (2013)
3. Deng, Y., Wu, Z., Chu, C.-H., et al.: Evaluating feature selection for stress identification. In: 2012 IEEE 13th International Conference on Information Reuse and Integration (IRI), IEEE, pp. 584–591 (2012)
4. Deng, Y., Wu, Z., Chu, C.-H., et al.: An investigation of decision analytic methodologies for stress identification, The International Journal on Smart Sensing and Intelligent Systems (ISSN: 1178-5608) (2012)
5. Deng, Y., Wu, Z., Chu, C.-H., et al.: Sensor feature selection and combination for stress identification using combinatorial fusion. Int. J. Adv. Robot. Syst. **10**, 306–313 (2013). Jofish Kaye and Paul Dourish. 2014. Special issue on science fiction and ubiquitous computing. Personal Ubiquitous Comput. 18, 4 (April 2014)
6. Goldberger, A.I., Amaral, L.A.N., Glass, I., Hausdorff, J.M., Ivanov, P.C., Mark, R.G., Mietus, J.E., Moody, G.B., Peng, C.K., Stanley, H.E.: PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation. **101**(23),

e215–e220 (2000). https://doi.org/10.1161/01.CIR.101.23.e215. [Circulation Electronic Pages. http://circ.ahajournals.org/cgi/content/full/101/23/e215]; PMID: 10851218
7. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explor. **11**(1), 10–18 (2009)
8. Healey, J., Picard, R.W.: Detecting stress during real-world driving tasks using physiological sensors. IEEE Trans. Intell. Transp. Syst. **6**(2), 156–166 (2005)
9. Keshan, N., Parimi, P.V., Bichindaritz, I.: Machine learning for stress detection from ECG signals in automobile drivers. In: 2015 IEEE International Conference on big data (Big Data), IEEE, pp. 2661–2669 (2015)
10. Singh, M., Queyam, A.B.: A novel method of stress detection using physiological measurements of automobile drivers. Int. J. Electron. Eng. **5**(2), 13–20 (2013)
11. Singh, M., Queyam, A.B.: Correlation between physiological parameters of automobile drivers and traffic condition. Int. J. Electron. Eng. **5**(2), 6–12 (2013)

Chapter 4
Trends and Future Directions of Research for Smart Grid IoT Sensor Networks

Arif I. Sarwat, Aditya Sundararajan, and Imtiaz Parvez

Abstract This chapter provides a high-level description and summary of smart grid IoT entity's three logical modules responsible for its intelligence: wide, medium, and local area sensor networks. For each module, the architecture, key applications, and potential challenges to future research are discussed using a widely recognized system as an example. In particular, the interdependency between these IoT entity modules, underlying power system and overlying operational entity, is presented, with a special emphasis on big data, communication, and security challenges. This chapter will serve as a starting point for researchers entering the field of smart grid IoT big data analytics, communication, and security.

Keywords Smart grid · IoT · Synchrophasors · AMI · Smart meters · Transactive energy · Situation awareness · Prosumers · Future directions

4.1 Introduction

The smart grid is one of the most complex, dynamic, and geographically widespread Internet of Things (IoT) infrastructure today. With countless sensors deployed across the transmission, sub-transmission, distribution and customer networks for real-time situation awareness, dynamic visualization, reliability estimation, customer billing and revenue, direct load control, interoperable energy exchange between distributed generations, and much more, the grid has become a live, breathing body of intelligent devices that constantly sense, communicate, and act. Due to an increased availability of high-dimensional IoT data, operators are now capable of keeping the grids behavior in track down to every second. This, made possible by the presence of distributed sensor networks, has greatly improved accuracy in measurement and processing, which in-turn has led to an enhanced grid performance [50].

A. I. Sarwat (✉) · A. Sundararajan · I. Parvez
Florida International University, Miami, FL, USA
e-mail: asarwat@fiu.edu; asund005@fiu.edu; iparv001@fiu.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_4

Intelligent electronic devices (IEDs) at the substations are now being replaced by smart meters that make use of the powerful advanced metering infrastructure (AMI), by virtue of which the quality of data measurement has significantly advanced. As a ripple effect, installation of these meters has prompted the execution of power system applications such as load estimation to undergo a paradigm shift from manual to automated algorithmic processes. While the conventional grid supported a unidirectional power flow from generation, transmission, sub-transmission, distribution, and finally down to the customer homes, smart grid has revolutionized the energy generation and delivery. Consumers have now been transformed to prosumers, wherein they may install renewable energy sources (RESs) like solar photovoltaic (PV) or wind turbines, augment them with energy storage technologies such as batteries and supercapacitors, and even increase demand dynamism by deploying smart loads such as electric vehicles (EVs).

Functionally, the grid is divided into generation, transmission, distribution, customer, market, and service providers, where each module has a specific functional requirement it must fulfill at all times. From an operational standpoint, it can be viewed as a system of interconnected modules. Smart grid can be thus logically divided into three entities as depicted by Fig. 4.1. Functional entity (FE) comprises generation, transmission, sub-transmission, distribution, customer, and market. Operational entity (OE), located at distribution and transmission control centers, consists of energy management system (EMS), distribution management system (DMS), outage management system (OMS), and meter data management (MDM). The EMS, DMS, and OMS form an integrated distribution management system (IDMS) that offers a unified computation and visualization interface for the three modules under one roof [2, 4].

The smart grid is not intelligent unless FE and OE engage in a robust, bidirectional communication bolstered by accurate sensing, measurement, and actuation. This is encapsulated by the IoT entity (IE). For every module at FE, there exists a corresponding module at IE to which is physically tethered to, but logically distinct from. These IE modules constantly sense measurements from their FE counterparts and report them to OE modules through the enterprise information system (EIS) using flexible and scalable communication protocols. EIS serves as the corporate information gateway for IE to interact with OE. For instance, MDM and OMS interact with AMI smart meters via customer and geographical information systems (CIS and GIS, respectively) to perform billing operations and map system-level interruptions for reliability analyses. EIS is equipped with powerful visualization and computation models which help operators and dispatchers maintain operational visibility of the grid. It also has multiple applications such as workforce management (WFM), asset management (AM), interactive voice response (IVR), and enterprise-level IT-cybersecurity governance policies. IE in turn can be categorized into three modules: wide area sensor network (WASN) for generation and transmission, medium area sensor network (MASN) for sub-transmission and distribution, and local area sensor network (LASN) for customer.

Synchronized phasor measurement units (PMUs) and phasor data concentrators (PDCs) that provide real-time situation awareness and power quality measures are

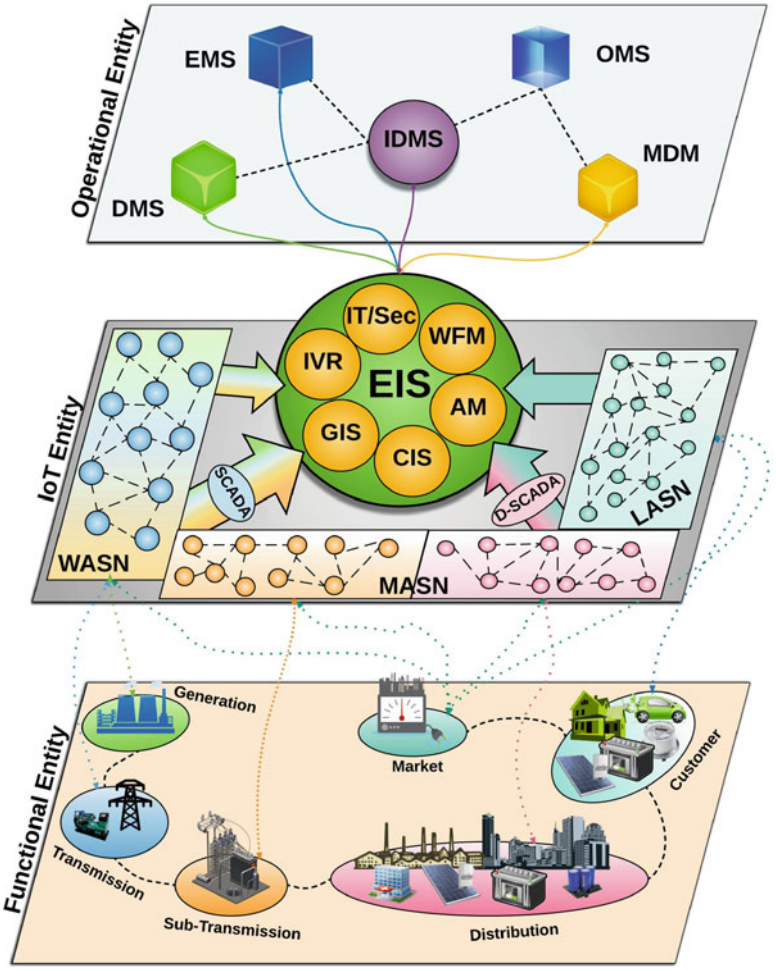


Fig. 4.1 Smart grid layout showing Operational, IoT and Functional Entities

part of the wide area measurement system (WAMS) which is used interchangeably with WASN in this chapter. Real-time market-based economic exchange of distributed energy generation between producers and prosumers, called "transactive energy," falls under MASN. Load tap changers (LTCs), circuit breakers, and voltage regulators at the distribution level and the residential/commercial AMI smart meters at the customer level constitute the LASN.

Traditionally, EIS has been a centralized cloud-powered framework, but recently, edge and/or fog computing frameworks have emerged due to issues of big IoT data such as latency, computation complexity, and network congestion. A robust, flexible, and scalable communication infrastructure is the bond that holds all these interdependent entities together, letting them behave harmoniously to create a

cohesive grid. The smart grid IoT can be analogized to a human body where FE is represented by the organs while IE is represented by sensory organs, bones, and muscles (sensors and actuators). IDMS and EIS are the brain. The cardiovascular and central nervous systems are the communication infrastructure that binds these entities in a cohesive manner.

The primary contribution of this chapter is to not only signify the interdependency between FE, IE, and OE of the smart grid but also to highlight key advancements, potential applications, and key challenges to the widespread adoption of the technologies they encompass. From these observations, future directions describing open research topics are briefly elucidated. The first three sections are dedicated to WASN, MASN, and LASN, respectively, while the fourth section summarizes the observations to draw a brief conclusion.

4.2 Wide Area Sensor Network (WASN)

WASN, also known as WAMS, is an interconnected network of strategically placed devices capable of sensing outgoing measurements from underlying FE and incoming control signals from overlying OE, programmed to ensure real-time cognizance of grid stability and visibility at all times. Synchrophasor devices like PMUs and PDCs are one of the most prominent WASN devices. PMUs can be deployed either as standalone systems or as part of a larger system like IEDs at substations [46]. It can be deduced that while measurement units (like PMUs) and concentrator units (like PDCs) occupy IE, the analytical and visualization models are part of the EIS realm. A high-level layout of WASN showing PMUs, PDCs, and the more recent frequency disturbance recorders (FDRs) is shown in Fig. 4.2.

While it is beyond the scope of discussion of this chapter, it is worth mentioning that PMUs that have traditionally been applied across long-range transmission lines have now found use even in medium-range distribution networks, owing to the increasing penetration levels of RESs and smart loads. To this effect, the Oak Ridge National Laboratory (ORNL) has developed and deployed FNET/GridEye project, active since 2004, as part of which frequency disturbance recorders (FDRs) have been installed and managed in many countries to capture dynamic behaviors of the grid [6]. These IoT devices communicate directly using the internet to the information systems secured by Firewall. They are also capable of measuring 1440 samples per second, nearly more than ten times that of PMU sampling rate. PDCs are software applications that conduct various data quality checks and set necessary flags according to the issues encountered, perform performance logging and alarming, data validation, transformation, scaling, normalization, protocol conversion, and much more. PDCs usually have an associated host computer which helps with local processing, descriptive analytics, and data storage. There is typically a direct interface between the PDC and the utilitys SCADA. PDCs, like PMUs, can also be integrated with other systems in the grid. Sometimes, multiple PMUs, especially at a substation, might report their data to a local PDC, which does

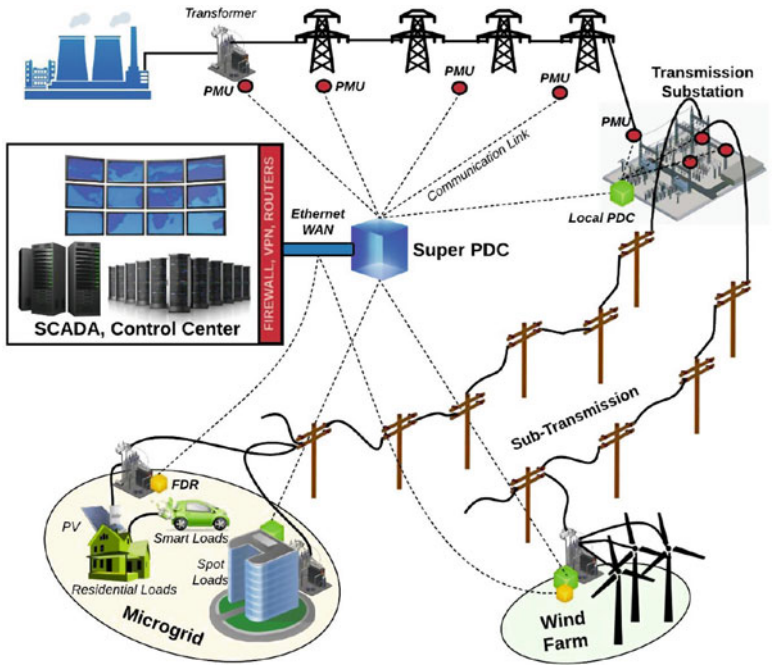


Fig. 4.2 Layout of smart grid WASN comprising Synchrophasors

local quality checking and processing before forwarding the data to a SuperPDC located at the control center. In some cases, local PDCs interact among one another over a peer-to-peer network. At times, different utilities, control areas, or reliability coordinators also establish such networks to get respective PDCs communicating.

4.2.1 Architecture of WASN for Synchrophasors

A PMU comprises multiple modules, each dedicated to a specific purpose. Current and potential transformers (CTs, PTs) are respectively used to measure three-phase values of current and voltage with sampling rates between 10 and 120 samples per second, which are then digitized using an analog-to-digital converter (ADC). A microprocessor module compiles these values with timestamp, computes phasors, and synchronizes them with Coordinated Universal Time (UTC) standard reference used by the Global Positioning System (GPS) receivers with an accuracy of 1 µs. PMUs also measure local frequency and its rate of change and with further customization can record individual phase voltage-current and harmonics. This information generated helps paint a dynamic picture of the grid at any given time. The protocol specified by IEEE Standard C37.118.1/2-2011 transmits PMU

Table 4.1 The various standards for Synchrophasors

Body	Standard	Core contribution
IEEE	1344-1995	Original parameter definitions for synchrophasors
	C37.118-2005	Improved message formats, inclusion of time quality, Total Vector Error
	C37.239-2010	PMU/PDC event logging
	1711-2010	Serial SCADA Protection Protocol (SSPP) for integrity, confidentiality for substation serial link cybersecurity
	C37.118.1-2011	PMU measurement provisions, performance requirements
	C37.118.2-2011	Synchrophasor data transfer requirements
	C37.238-2011	Common profile for applying Precision Time Protocol (PTP) to A-P-C[a] applications using Ethernet
	C37.242-2013	Synchronization, calibration, testing and installation of PMUs for P-C
	C37.244-2013	PDC functions and requirements for PC and monitoring
	C37.111-2013	PMU/PDC data logging using COMFEDE
	1686-2013	Procuring, installing and commissioning IED cybersecurity
	C37.240-2014	Sound engineering practices for high cybersecurity of substation A-P-C
IEC	61850	Interoperable and adaptable architectures to substation automation
	61850-90-5	Requirements for data exchange between PMUs, PDCs, control center
	62351-1,2	Security threats and vulnerabilities in smart grid devices
	62351-6	Prescribes digital signature using asymmetric cryptography for sending PMU data
NERC	CIP 002-009	Series of standards to ensure enterprise, field and personnel security
NIST	NISTIR 7628	Provides guidelines for smart grid cybersecurity (including WAMS)

[a] A: Automation, P: Protection, C: Control

data as frames [19]. Irrespective of the communication medium, which could be TCP/IP, Fieldbus Message Specification (FMS), Ethernet, or RS-232, PMUs usually transmit data and the 16-bit cyclic redundancy check (CRC-CCITT) that ensures data integrity. Multiple standards currently exist for data measurement, transfer, and communication among PMUs and PDCs, proposed by IEEE, the National Institute of Standards and Technology (NIST), the North American Electric Reliability Commission (NERC), and the International Electrotechnical Commission (IEC), summarized in Table 4.1 [11, 20–25, 32].

4.2.2 Key Applications

PMUs are a better option for streamlining security, reliability, and stability of power systems. Applications can be categorized as real-time and offline, described below.

- **Real-Time Applications:** PMUs enhance real-time wide area situation awareness (WASA) which is one of the key milestones recognized by the North American Electric Reliability Commission (NERC), the National Institute of Standards and Technology (NIST), and the North American SynchroPhasor Initiative (NASPI) [1]. Other real-time applications include timely detection and isolation of faults at transmission level, improving power quality by addressing harmonic distortions, event detection, and classification for further analysis and improving the accuracy and computational time of state estimation.
- **Offline Applications:** Offline applications include event analysis (post classification), SCADA model validation, direct load control and demand response, evoking special protection schemes and islanding, operation planning, and benchmarking power system performance [51].

4.2.3 Challenges and Future Research Directions

The applications and challenges of PMUs are a well-researched area, with multiple survey papers available in the literature [30, 31, 33, 43, 45, 47]. However, those which tackle challenges specifically related to data quality and cybersecurity are more recent. These challenges are summarized below.

- **Placement:** NASPI Research Initiative Task Force (RITF) emphasizes that the optimal placement of PMUs is a significant challenge but also one that is dependent on the nature of application requirements. The literature has multiple models like genetic algorithm, simulated annealing, Tabu search, Madtharad's method, particle swarm optimization, artificial neural networks, and binary search dedicated to addressing this challenge [7, 34, 49]. However, lack of a standardized approach is still a challenge to widespread implementation.
- **Analytics:** Undoubtedly, managing and analyzing large volumes of data generated by PMUs is a challenge that gets worse with time. Lack of standardized data management solutions has only made this problem harder to resolve [28, 35].
- **Data Quality:** Data analytics is powerful only when the data from underlying devices is of good quality. The quality of data is defined by its completeness, consistency, accuracy, timeliness, and plausibility. Many works in current research identify the data quality challenges, but are quite introductory in nature. Missing industry-wide standards is a key hurdle to achieve a common solution.

- **Communication:** PMUs require flexible, scalable, and resilient communication to reduce latency and improve throughput. To realize WASA, NASPI conceptualized the NASPI network (NASPInet) with the US Department of Energy (DOE) to develop standardized two-way communication [3]. However, significant challenges like meeting strict quality of service (QoS) requirements exist, to address which, a two-tier fault-tolerant hub-spoke architecture was proposed.
- **Cybersecurity:** Synchrophasors maintain WASA through data measurement, processing, and visualization. These processes increase the attack surface, exposing vulnerabilities that could be potentially exploited by attackers. There are introductory works which examine cybersecurity taxonomy and identify requirements, but diligent research to properly address this challenge is lacking. Further, the research on impacts of cybersecurity of synchrophasors has largely been conducted in isolation from other related challenges such as data quality. For instance, false data injection (FDI) attacks aim to compromise the integrity of sensitive phasor data that important power system applications depend on. Compromised information could result in erroneous predictions of the grid conditions that could, in severe cases, trigger cascading failures or even blackouts. This example illustrates the complex interplay between data quality and cybersecurity, which is not emphasized enough by the existing literature.

4.3 Medium Area Sensor Network (MASN)

Sandwiched between WASN and LASN is the MASN for the sub-transmission and distribution networks. One of the key systems that effectively leverage the smart grid's IoT nature is transactive energy. Grid modernization and the move toward a decentralized electricity infrastructure with an increased penetration level of RESs have fostered the need for developing a transitive energy market.

4.3.1 Architecture of MASN for Transactive Energy

Transactive energy (TE) is a term coined first by DOE's GridWise Architecture Council (GWAC) in its Transactive Energy Framework and is currently adopted by NIST [12]. While this concept is not new for bulk transmission systems where operators use markets to manage supply-demand balance and ensure grid reliability, its application is nascent for sub-transmission and distribution networks where the integration of rooftop and commercial solar, large-scale wind farms, and smart loads such as EVs are on the rise. This paradigm shift makes TE a lucrative subject of discussion for MASN in this chapter. While traditional markets used seasonal average rates for pricing electricity, that model fails when generation and loads change by the hour. Further, today's economic model gives importance

to interoperability, transparency, distributed intelligence, and the power to control energy usage to end customers. With the deployment of IEDs smart meters and protection coordination devices, there are even more opportunities to invest in TE.

This need has been the impetus to initiating an organized research in developing TE systems. Some of the key requirements of such systems include coordinated control and optimization, allowing equal participation in the market by qualified stakeholders; ensuring visibility of the system, scalability, and interoperability across FE module interfaces; and correctly abstracting the cyber-physical model of the grid. A TE system involves multiple stakeholders: bulk generator, end customer, energy provider, energy financier, energy servicer, distribution operator, and energy asset owner. These stakeholders bear different objectives, perceptions, willingness and ability to comply, and constraints. A well-rounded TE system must ensure the enforcement of policies that fulfill these stakeholder interests without jeopardizing system reliability. GWAC presents four areas of concern that can be directly mapped to the smart grid three-entity framework proposed in this chapter. Corresponding to FE is the cyber-physical infrastructure, followed by information interoperability and exchange similar to IE and business models, value addition, policymaking, and regulation at the highest level (similar to OE). Accordingly, the level of abstraction also increases as one goes from lower to higher levels. While the cyber-physical infrastructure deals with both the physical power infrastructure and the cyber communication networks, the information interoperability and exchange addresses the impacts of TE system on the grid and how it affects the stakeholder interests. The top layer involving business, market, policy, and regulation identifies the stakeholders, their powers and limitations, jurisdictional constraints, and reward mechanisms for improving system and stakeholder behaviors.

Transactive energy is a concept that spans multiple applications across all three entities and exploits the hallmark characteristics of IoT. At the FE, it establishes real-time communication across transmission, sub-transmission, and distribution modules with the OE modules of EMS, DMS, OMS, and MDM through EIS.

In the recent years, the focus has been on the sellers and buyer transactions within wholesale (transmission level) market structures. A well-developed transactive energy market will open the door for transactions to flow between wholesale and retail (distribution level) parties [5]. Only with a well-developed translation and communications capabilities (with new/updated sensors and meters installed at the transmission/distribution/ and customer levels), operators at the OE will be able to send/receive signals to/from different service providers all the way to customer side where he/she will have the choice whether to respond or not to the signals. As more decentralized electricity system and DERs come into the picture with increasing prosumers, the more the stress will be on MASN and LASN IoT parts to communicate with the top and same entity parts. Researchers, legislative, and regulatory stakeholders have been working on designing transactive energy market framework architectures and solutions [12, 17].

Two alternative TE markets at the distribution level (retail) have been proposed. The first structure is based on enabling the sell/buy energy transactions by retail

energy parties through the retail-wholesale interface in response to wholesale forward/spot prices as well as forward/spot distribution prices. The second structure suggests changes to the design of the existing retail rates, specifically their distribution service price components. TE has also been recently applied to microgrids [36]. A decentralized multi-agent control approach has been proposed to lower the energy imbalances in microgrids and reduce the reliance on energy from the grid.

4.3.2 Key Applications

The concept of TE has been recently imbibed into different applications, one of them being, advocated by NIST, "PowerMatcherSuite transactive smart energy" by Flexible Power where different energy demanding applications bid by clearly specifying their requirements, subject to changes due to alterations in its state, based on which the market's equilibrium can be identified at the shortest numbers of iterations. Some of the key applications highlighted by the SGIP Transactive Energy Coordination Group are briefed below [18].

- **Load shedding during peak demands:** During peak hours, TE system can be used to balance the supply and demand by spooling RESs, peak load shaving, or customer demand response operations. This application would involve the operators, generators, customers, financiers, and aggregators.
- **Energy response services:** With distributed RESs becoming prevalent, impacts due to their intermittent generation can be adverse on the grid. TE systems can accommodate for these fluctuations or ramps by engaging timely participation from generators, operators, customers, aggregators, and owners to absorb or inject required power in order to maintain the net generation within limits of demand.
- **Managing an islanded microgrid:** Dynamic islanding is a protective feature used by microgrids to shield themselves from the cascading failures in the main grid. However, to ensure continued power availability, TE systems can perform optimized energy balancing functions by negotiating interests of stakeholders like customers and owners, load shedding, or kick-starting reserved RESs.

4.3.3 Challenges and Future Research Directions

Despite designing architectures for TE systems, more work must be done to establish a robust MASN. The NIST TE Challenge has outlined six key working areas pertaining to: (a) developing tools and platforms for TE research and advance interoperability and application standards; (b) understanding the interlinking between TE and some of the grid's current challenges in reliability and quality; (c) developing use cases which can provide the baseline for modeling and simulation; (d) creating

and maintaining a positive environment for sharing resources, ideas, and data among the community; (e) engaging in tethering the simulations toward deploying pilot ventures by communicating with utilities, regulators, and policymakers; and (f) developing a healthy and robust medium for disseminating the key research observations made. From these primary working areas, different underlying challenges exist, described below:

- **Policy revision:** The policies, tariffs, system, and market architectures today are designed for the centralized frameworks of energy generation, dispatch, monitoring, and control. With imminent changes to this model like consumers being prosumers, distributed generation, and transition from cloud to edge/fog computing, there is a growing challenge for the policies to be updated accordingly.
- **Roadmap to implementation:** While many regulatory bodies have begun experimenting with the concept of TE in coordination with their jurisdictional legislative authorities, increasing challenges in developing feasible and specific implementation plans have been identified. The requirement, thus, is for roadmaps that effectively and seamlessly help transitioning the market economy from its existing state to the one advocated by TE.
- **Demonstrating viability and providing incentives:** As with any transformative concept, TE needs heavy lifting from the legislative and regulatory bodies in terms of incentivizing the adoption of its principles, first on a smaller community scale and then on a larger city scale. Modular adoption and scalable integration in the form of micro- and nanogrids will be instrumental in creating scenarios for implementing TE and will also make the process much simpler. With its feasibility, economical advantage and broader impacts demonstrated, it can be expanded across other communities with much less hassle.
- **Stakeholder behavior:** As described earlier, a TE system constantly interacts with multiple stakeholders, juggling their respective interests in an optimal manner to ensure continued grid operation at its best quality and reliability. However, this raises potential conflicts in matters like controlling assets that span multiple jurisdictions, customer choice, and privacy and bias in participation. This requires business models that carefully balance interests and the "global good."
- **Information exchange**: Interoperability between devices, networks, applications, or systems separated physically and/or logically is one of the key enablers to a successful TE system. However, achieving seamless interoperability for information exchange and putting that information to proper use is still a challenge. GWAC proposed a framework to provide the context for stakeholders to discuss, debate, and resolve the technical challenges of interoperability and information exchange, like shared content management, time synchronization and sequencing, QoS, transaction and state management, security and privacy, and much more.

4.4 Local Area Sensor Network (LASN)

LASN includes AMI that engages in energy consumption data collection, net metering, market analysis and transactive energy at the customer levels, building automation, control monitoring, and demand-side management [8–10, 14].

4.4.1 Architecture of LASN for AMI Smart Meters

AMI is a mesh, hierarchical, or hybrid IoT network of smart meters at FE with bidirectional communication for transferring consumption data to MDM at OE. Both wired and wireless communications exist, with power-line communication (PLC) being the standard for wired and cellular technologies, WiFi, or ZigBee [15] for wireless [38, 39, 42]. AMI thus invalidates the need for monthly manual consumption data collection. A typical AMI network shown in Fig. 4.3 has the following components:

- **Smart meter:** Smart meter is the primary IoT sensor in an AMI. It is a solid state device which collects and stores consumption data. Based on its requirement, a smart meter can send data in bursts from every 15 min to 1 h. Generally, smart meters use ZigBee, WiFi, or cellular networks to transmit data.
- **Home area network (HAN):** Smart appliances like TVs, refrigerators, pool heaters, and air conditioners at each home are connected to a smart meter over a wireless network known as HAN, which uses open protocols like IEEE Standard 802.15.4 ZigBee, WirelessHART, ISA100.11a, and proprietary protocols such as Z-Wave, INSTEON, and Wavenis for communication [26, 38].
- **Neighborhood area network (NAN):** Meters are connected to each other over a mesh, hierarchical, or hybrid network known as NAN. It can utilize short-range communication protocols like ZigBee or WiFi or long-range ones like

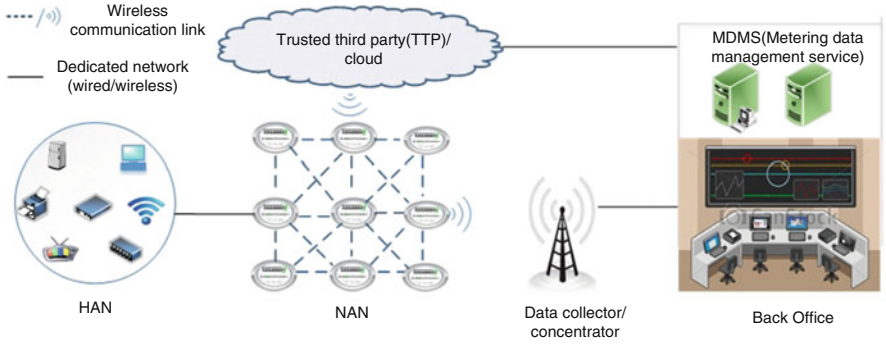


Fig. 4.3 A typical AMI network comprised of HAN, NAN, data collector and MDM

cellular (4G/LTE/5G) [15]. Some of the notable routing protocols for NAN include distributed autonomous depth-first routing, hybrid routing protocol, IEEE 802.11s, timer-based multipath diversity routing, and on-demand distance vector [15, 44].

- **Data collector/concentrator:** At the head of a NAN is a concentrator or gateway which forwards aggregated data to MDM. It can collect data from up to 5000 m and generally uses dedicated wired communication like optic fiber or long-range wireless communication like cellular networks.
- **MDM:** The MDM receives consumption data from meters and performs a wide range of operations such as billing and revenue, network and asset management, and deployment and planning. Hence, MDM constantly interacts with EIS modules such as WFM, AM, CIS, and GIS.
- **Trusted third party (TTP):** Since smart meters are connected over a wireless network, they are vulnerable to various cyber-physical attacks. The threats might range from identity and electricity thefts to cascading blackouts. Hence, data security is a growing concern, to address which TTP is enforced in some cases. It manages security schemes, distributes keys, and monitors the AMI.

AMI systems comprise different subsystems which can be elaborated as follow:

- **Field devices:** Field devices can be the smart meters, APs, routers, or switches, and other associated physical devices spread across the grid. These devices are embedded with multiple IoT sensors that are calibrated to record interval energy consumption, aggregate the information at the neighborhood level, and apply data quality measures to ensure integrity, accuracy, and consistency. These devices can be considered to be the intelligence at the grid's edge.
- **Back-end applications:** As described earlier, MDM collects, processes, manages, and maintains the wealth of AMI data. The applications the MDM is tethered to are present at the command and control center.
- **Communication channel:** A myriad of communication protocols make up the channels that different field devices use to communicate either among themselves or with the central back-end applications. These protocols have also been identified and briefly described earlier in the section.

4.4.2 Key Applications

Due to their widespread adoption, AMI has numerous applications as listed below:

- **Net metering:** RESs like rooftop solar are increasingly being adopted by customers [48]. The surplus power generated from the solar system can either be stored for later use or sold back to the grid. In the latter case, customers truly become prosumers, in that they sell their generated energy to the utility to offset the cost of power they purchase.

- **Residential energy management:** LASN applications can monitor real-time power consumption, allowing customers to schedule and shift high-utility applications to nonpeak demand hours [27]. For example, utilities supply AMI meters with their dynamic Time of Use (ToU) pricing according to which it schedules pool heating and laundry for late night hours when electricity is cheaper. Not only does it help the utility manage loads better, but also is economical for customers.
- **Building automation:** An automated building allows communication between appliances such as lights and heating, ventilation and air conditioning (HVAC), to optimize their running time, thereby reducing energy wastage [13]. AMI meters play a crucial role in realizing this, helping save up to 30% of energy.
- **Power grid equipment monitoring and control:** Sensors in smart meters and customer-level transformers report power quality parameters like over- and under-voltage, sag, swell, transients, harmonics, flickers, phase shifts, and reactive power, to the control center [16]. Based on the power quality report, control centers can take steps like direct load control or real power injection into the grid.

4.4.3 Challenges and Future Research Directions

There are many challenges to large-scale AMI deployment which are listed below:

- **Topology design:** AMI covers huge of number of nodes and area. In such cases, network topology design could become challenging since many factors like coverage, management, and security must be considered. While the most widely applied topology involves the use of AMI smart meters tethered in the form of a mesh network, their periodic measurements collected by aggregators like APs, which are then fed to the command and control center where the data is processed, stored, and managed for an extended period of time. Just as the topologies are varied, so are the solutions available to meet those needs. Thus, a careful consideration of optimal technology-to-topology combination must be envisaged.
- **Security:** Security is a vital issue in AMI [37, 40, 41]. By learning energy consumption patterns from data, customer behavior can be determined. Greedy customers might engage in electricity thefts to get cheaper bills. Adversaries can perpetrate denial of service, Man-in-the-Middle, and replay attacks to steal, spy on, or modify sensitive energy consumption data. At a higher level, security for communication interfaces both across and within domains must be taken into account.
- **Heterogeneous network:** AMI uses WiFi, ZigBee, or Bluetooth which must coexist with other application networks [29]. Interferences with other networks in the vicinity could cause transmission delays, packet losses, or data corruption. The heterogeneity also causes interoperability concerns as further explained below.

- **Interoperability:** It is common knowledge that there exist multiple standards that independently ensure robust, resilient, and efficient communication network solutions across the smart grid. However, the breadth of these standards is huge and raises evident interoperability concerns. Different sensors spread over different geographical or jurisdictional areas might use components obtained from different vendors which adhere to different acceptable standards. However, to establish and maintain a secure, seamless, and ubiquitous communication between heterogeneous sensors and devices, interoperability requirements must be clearly specified.
- **Interference:** Since AMI has a large number of nodes, unlicensed bands like 900 MHz, 2.4 GHz, and 5 GHz are preferred [38]. However, they can be used by anyone. Further, the electromagnetic interference (EMI) under high-voltage scenarios causes propagation delays or even losses to the data being transmitted. Radio frequency (RF) signals could interfere with other wireless equipment in the vicinity and cause either malfunctioning of the equipment or disarray of the communication medium. Hence, network design considering interference and security is challenging. One of the solutions that can be explored, but is difficult to implement due to an increased cost, is the creation of a dedicated unlicensed band for supporting smart grid communication infrastructure.
- **QoS:** Due to the use of public frequency bands and huge number of nodes, QoS parameters such as bit error rate might be adversely impacted. Decentralization of intelligence from centralized client-server architecture to distributed peer-to-peer architecture will redefine the existing QoS requirements as well. Increasing data volume, slow response time requirements, and low latency needs will require an overhaul of QoS specifications.

Acknowledgements The work for this chapter is supported by the NSF Grant 1553494.

References

1. Alcaraz, C., Lopez, J.: Wide area situational awareness for critical infrastructure protection. IEEE Comput. **46**, 30–37 (2013)
2. Areva T&D: The Integrated Distribution Management (IDMS). Technical report (2009)
3. Bobba, R., Heine, E., Khurana, H., Yardley, T.: Exploring a tiered architecture for NASPInet. In: Innovative Smart Grid Technologies (ISGT) (2014). https://doi.org/10.1109/ISGT.2010.5434730
4. Boroojeni, K.G., Amini, M.H., Bahrami, S., Iyengar, S.S., Sarwat, A.I., Karabasoglu, O.: A novel multi-time-scale modeling for electric power demand forecasting: from short-term to medium-term horizon. Electr. Power Syst. Res. **142**, 58–73 (2017)
5. Cazalet, E., De Martini, P., Woychik, E., et al.: Transactive energy models, NIST transactive energy challenge: business and regulatory models working group (2016)
6. Chai, J., et al.: Wide-area measurement data analytics using FNET/GridEye: a review. In: Power Systems Computation Conference (2016). https://doi.org/10.1109/PSCC.2016.7540946
7. Chow, J.H., et al.: Guidelines for siting phasor measurement units. North American SynchroPhasor Initiative Research Initiative Task team (NASPI-RITT) report (2015)

8. Erol-Kantarci, M., Mouftah, H.T.: Wireless Sensor Networks for smart grid applications. In: 2011 Saudi International Electronics, Communications and Photonics Conference (SIECPC), Riyadh, pp. 1–6 (2011)
9. Erol-Kantarci, M., Mouftah, H.T.: Wireless multimedia sensor and actor networks for the next-generation power grid. Ad Hoc Netw. **9**(4), 542–551 (2011)
10. Fadel, E., Gungor, V.C., Nassef, L., Akkari, N., Malik, M.G.A., Almasri, S., Akyildiz, I.F.: A survey on wireless sensor networks for smart grid. Comput. Commun. **71**, 22–33 (2015), ISSN 0140-3664, http://dx.doi.org/10.1016/j.comcom.2015.09.006. (http://www.sciencedirect.com/science/article/pii/S0140366415003400)
11. Firouzi, S.R., Hooshyar, H., Mahmood, F., Vanfretti, L.: An IEC 61850-90-5 gateway for IEEE C37.118.2 synchrophasor data transfer. In: NASPI-ISGAN International Synchrophasor Symposium (2016)
12. GridWise Architecture Council (GWAC): GridWise transactive energy framework version 1.0. technical report (2015)
13. Guan, X., et al.: Energy-efficient buildings facilitated by microgrid. IEEE Trans. Smart Grid **1**, 243–252 (2010)
14. Gungor, V.C., Lu, B., Hancke, G.P.: Opportunities and challenges of wireless sensor networks in smart grid. IEEE Trans. Ind. Electron. **57**, 3557–3564 (2010)
15. Gungor, V.C., et al.: Smart grid technologies: communication technologies and standards. IEEE Trans. Ind. Inf. **7**(4), 529–539 (2011). https://doi.org/10.1109/TII.2011.2166794
16. Guo, Z., et al.: A wireless sensor network for monitoring smart grid transmission lines. In: 23rd International Conference on Computer Communication and Networks (ICCCN) Shanghai, 2014, pp. 1–6 (2014). https://doi.org/10.1109/ICCCN.2014.6911790
17. GWAC: GridWise interoperability contextsetting framework. Technical report (2008)
18. Holmerg, D., Hardin, D., Cunningham, R., Melton, R., Widergren, S.: Transactive energy application landscape scenarios. Transactive energy coordination group white paper (2016)
19. Huang, Z., Dagle, J.: SynchroPhasor measurements: System architecture and performance evaluation in supporting wide-area applications. Emerging technologies in support of smart grids, Pacific Northwest National Laboratory (PNNL) report (2008)
20. IEEE Power & Energy Society: IEEE standard for synchrophasor measurements for power systems (2011)
21. IEEE Power and Energy Society: IEEE trial-use standard for a cryptographic protocol for cyber security for substation serial links (2011)
22. IEEE Power and Energy Society: IEEE standard for common format for event data exchange (COMFEDE) for power systems (2011)
23. IEEE Power and Energy Society: IEEE standard for intelligent electronic devices cyber security capabilities (2013)
24. IEEE: Common format for transient data exchange (COMTRADE) for power systems: IEC 60255-24 Part 24 (2013)
25. IEEE Power and Energy Society: IEEE standard cybersecurity requirements for substation automation, protection, and control systems (2014)
26. Jamei, M., Sarwat, A.I., Iyengar, S.S., Kaleem, F.: Security breach possibility with RSS-based localization of smart meters incorporating maximum likelihood estimator. In: International Conference on Systems Engineering (2015)
27. Kantarci, M.E., Mouftah, H.T.: Wireless sensor networks for domestic energy management in smart grids. In: Proceedings of 25th Biennial Symposium on Communications (2010)
28. Khan, M., Li, M., Ashton, P., Taylor, G., Liu, J.: Big data analytics on pmu measurements. In: 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), China (2014). https://doi.org/10.1109/FSKD.2014.6980923
29. Khan, F., Rehman, A.u., Arif, M., Aftab, M., Jadoon, B.K.: A survey of communication technologies for smart grid connectivity. In: 2016 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), Quetta, pp. 256–261 (2016). https://doi.org/10.1109/ICECUBE.2016.7495234

30. Lauby, M.: Real-time application of PMUs to improve reliability task force (RAPIR TF). NERC technical presentation (2010)
31. Lee, H., Tushar, Cui, B., et al.: A review of synchrophasor applications in smart grid. WIREs Energy Environ. **6**, e223 (2017)
32. Mackiewicz, R.: Technical overview and benefits of the IEC 61850 standard for substation automation (2006)
33. Mohanta, D.K., Murthy, C., Roy, D.S.: A brief review of phasor measurement units as sensors for smart grid. Electr. Power Compon. Syst. **44**, 411–425 (2016)
34. Negash, K., Khan, B., Yohannes, E.: Artificial intelligence versus conventional mathematical techniques: a review for optimal placement for phasor measurement units. Technol. Econ. Smart Grids Sustain. Energy. **1**, 10 (2016)
35. NERC: FNet and PMU data overview. Cause analysis and data workshop (2012)
36. Nunna, H.S.V.S.K., Srinivasan. D.: Multi-Agent based transactive energy framework for distribution systems in smart microgrids. IEEE Trans. Ind. Inf. **13**, 1–1 (2017)
37. Parvez, I., Islam, A., Kaleem, F.: A key management-based two-level encryption method for AMI. In: IEEE PES General Meeting Conference Exposition, pp. 1–5 (2014)
38. Parvez, I., Sundararajan, A., Sarwat, A.I.: Frequency band for HAN and NAN communication in Smart Grid. In: IEEE Symposium on Computational Intelligence, Orlando (2014)
39. Parvez, I., Jamei, M., Sundararajan, A., Sarwat, A.I.: RSS based loop-free compass routing protocol for data communication in advanced metering infrastructure (AMI) of Smart Grid. In: 2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG), Orlando, FL, pp. 1–6 (2014)
40. Parvez, I., Abdul, F., Sarwat, A.I.: A location based key management system for advanced metering infrastructure of smart grid. In: 2016 IEEE Green Technologies Conference (GreenTech), Kansas City, MO, pp. 62–67 (2016)
41. Parvez, I., Sarwat, A.I., Wei, L., Sundararajan, A.: Securing metering infrastructure of smart grid: a machine learning and localization based key management approach. Energies **9**(9), 691 (2016). https://doi.org/10.3390/en9090691
42. Parvez, I., Islam, N., Rupasinghe, N., Sarwat, A.I., Gven, İ: LAA-based LTE and ZigBee coexistence for unlicensed-band smart grid communications. In: SoutheastCon 2016, Norfolk, VA, pp. 1–6 (2016)
43. Sanchez-Ayala, G., Aguerc, J.R., Elizondo, D., et al.: Current trends on applications of PMUs in distribution systems. In: Proceedings of IEEE PES Innovative Smart Grid Technologies (ISGT) (2013)
44. Saputro, N., Akkaya, K., Uludag, S.: A survey of routing protocols for smart grid communications. Comput. Netw. **56**(11), 2742–2771 (2012)
45. Sexauer, J., Javanbakht, P., Mohaghehi, S.: Phasor measurement units for the distribution grid: necessity and benefits. In: Proceedings of IEEE PES Innovative Smart Grid Technologies (2013)
46. Shahraeini, M., Javidi, M.H.: Wide Area Measurement Systems. Communication infrastructure planning for wide area measurement systems in power systems (2012). https://doi.org/10.1504/IJCNDS.2013.054229
47. Singh, B., Sharma, N.K., Tiwari, A.N., et al.: Applications of Phasor Measurement Units (PMUs) in electric power system networks incorporated with FACTS controllers. Int. J. Eng. Sci. Technol. **3**, 64–82 (2011)
48. Smart Sensor Networks: Technologies and applications for green growth. http://www.oecd.org/dataoecd/39/62/44379113.pdf (2009)
49. Sreenivasareddy, P.S., Chowdhury, S.P., Chowdhury, S.: PMU placement- a comparative survey and review. In: IET International Conference on Developments in Power System Protection (2010). https://doi.org/10.1049/cp.2010.0326
50. The Department of Energy (DOE): The smart grid: an introduction. A DOE technical report (2009)
51. Zhang, P.: Phasor Measurement Unit (PMU) implementation and applications. Electric Power Research Institute (EPRI) technical report (2007)

Chapter 5
Accelerating the Big Data Analytics by GPU-Based Machine Learning: A Survey

K. Bhargavi and B. Sathish Babu

Abstract Today a large volume of structured and unstructured data is being generated online; the main sources for big data are social media profiles, MOOC (massive open online courses) log, social influencer, Internet of Things (IoT) data, the web, transactional applications, stream monitoring technologies, NoSQL (not only structured query language) stored data, log files, legacy document, and so on. There is a need to analyze such huge volume of data at a faster rate by uncovering the hidden patterns and correlation between the data to provide intelligent business decisions with high accuracy. The GPU (graphics processing unit)-enabled machine learning-based techniques are the strongest solution being used to perform big data analytics operation at an accelerated speed. This paper discusses selective GPU-based machine learning algorithms like decision tree, neural network, random forest, Q-learning, SARSA learning, K-means, NB (naive Bayes), AdaBoost, deep learning, support vector machine (SVM), linear regression, logistic regression, Apriori, and HMM (hidden Markov model) being used for big data analysis.

Keywords Machine learning · Big data · GPU · Acceleration · Analytics

5.1 Introduction

The term big data refers to the data made up of three Vs, i.e., high volume, high velocity, and high variety, which cannot be handled by traditional relational database tool. It covers a large volume of data including structured, unstructured, batch processed, or live streaming whose size varies from terabytes to petabytes. Big data analytics is the process of analyzing the huge volume of data to draw

K. Bhargavi (✉)
Department of CSE, Siddaganga Institute of Technology, Tumkur, India

B. S. Babu
Department of CSE, RV College of Engineering, Bengaluru, India
e-mail: bsbabu@rvce.edu.in

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_5

meaningful inferences by identifying hidden patterns in the existing data. The big data analysis operation involves several implementation level and processing level challenges like increased noise levels in high-dimensional data, computational cost, incomplete data with high degree of uncertainty, heterogeneity in data, improper extraction of hidden patterns from outliers of big data, experimental variation due to statistical inclination exhibited by unstructured data, and so on. The GPUs consisting of thousands of cores with massively parallel architecture are becoming a key tool for accelerating big data analytics to get a proper insight into the existing data to draw meaningful inferences. Today there are GPUs with around 5000 cores which are 32 times faster than conventional CPUs (central processing units), and a single core can have a memory of 192 GB which provides significant support for analysis of huge datasets. MapD is the leading service provider of GPUs for big data analytics [1] (https://www.rtinsights.com/gpu-computing-big-data-visualization-mapd-nvidia/; http://blogs.parc.com/2015/11/the-new-kid-on-the-block-gpu-accelerated-big-data-analytics/).

Many advanced big data analysis techniques like predictive analytics, machine learning, data mining, natural language processing, and so on are being used. Out of all advanced data analysis techniques, machine learning is one of the oldest and popular methods for big data analysis that continuously learn from the past data by recursively building analytical data models. Although the machine learning techniques for data analysis are being used for decades, the availability of a large amount of the social media data and the use of GPU with several thousands of cores for parallel processing changed the conventional look of machine learning. Many machine learning libraries are released to support big data analysis on GPU, which includes BIDMach, Capio, Deepgram, PolyAnalyst, UETorch, Theano, TensorFlow, Meson, Keras, H2O, Dextro, Caffe, nvGRAPH, and so on [2, 3] (https://www.ngdata.com/machine-learning-and-big-data-analytics-the-perfect-marriage/).

5.2 Machine Learning Algorithms

The machine learning algorithms have evolved from the concept of computational learning theory and have the capability to learn from the data without explicit programming by constructing learning models with frequent interaction with the environment. Based on the methodology of learning, the machine learning techniques are grouped into various categories like supervised learning, reinforcement learning, and unsupervised learning. The supervised learning algorithms learn from labeled pair of input-output examples; the reinforcement learning algorithms learn from the sequence of reward and punishment sequences, and the unsupervised learning algorithms learn from an unlabeled pair of input-output examples. The taxonomy representing the classification of machine learning techniques and distribution of algorithms among these categories is depicted in Fig. 5.1.

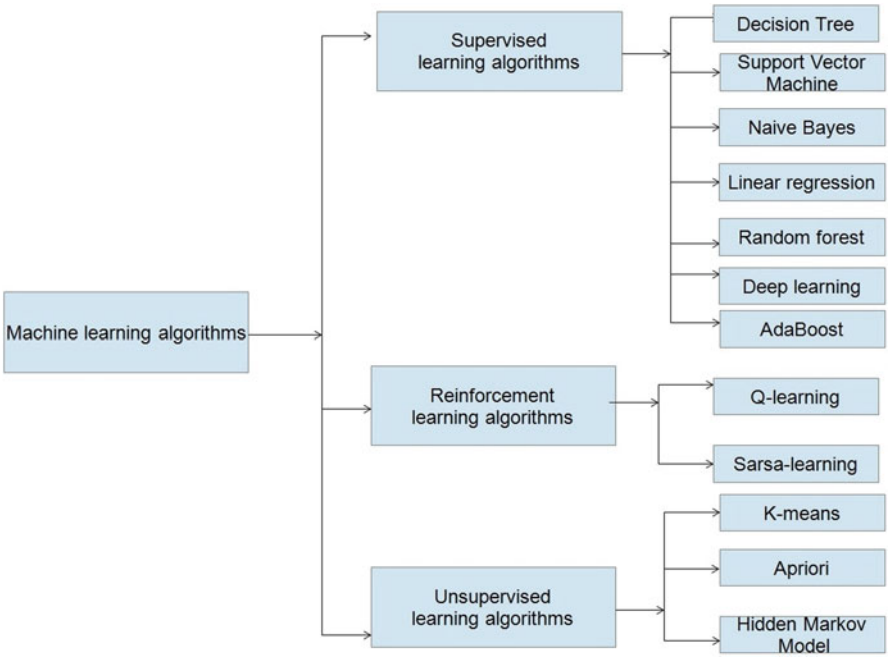


Fig. 5.1 Taxonomy of machine learning algorithms

5.3 GPU Empowering Machine Learning

The machine learning algorithms consist of complex mathematical operations involving a lot of vectors and matrices of very large size. The efficiency of matrix operations on GPU is more when compared to CPU as the GPUs are made up of several cores and each core is capable of spooling multiple threads in parallel. The efficiency of the learning algorithms improves over time by fine-tuning the input parameters; this feature of machine learning algorithm suites well with the parallel architecture of GPU made up of several computational units which are capable enough to perform several floating point operations per second. The GPUs are now being used to train machine learning algorithms with large labeled or unlabeled data samples in multiple levels to do data analysis at high speed with fewer infrastructures. Some of the parallel implementation of machine learning algorithms on GPU are discussed below.

5.3.1 Parallel Decision Tree

It is the supervised algorithm based on tree form of a graph which is used for classification purpose in data mining and analysis applications. The CPU-based implementation of the decision tree algorithm is found to be more time-consuming in nature; hence parallel initiatives of decision tree algorithms have been started. Some of the parallel implementation of decision tree algorithms are SPRINT and CUDAT. The SPRINT basically splits the dataset into several subsets and constructs the decision tree in parallel; the speed of splitting and sequencing the dataset is very high in SPRINT which in turn reduces the memory consumption rate also. The prefix-sum and CUDT stands for (CUDA based Decision Tree) library functions are used in CUDAT to support the parallel implementation of decision tree algorithms. The CUDAT consists of two functions, one is finding the split point in the input dataset, and another is splitting the identified input attribute list. After splitting the identified input attributes, decision trees are constructed and stored in host memory. By storing the tree in host memory, it can be scaled to larger data easily. Compared to SPRINT, CUDAT performance is found to be better as it achieves optimal trade-offs between CPU and GPU computation [4]. The illustration of decision tree construction process by iteratively splitting the dataset is shown in Fig. 5.2. The merits and demerits of parallel implementation of decision tree algorithm are listed below.

Merits

- It works well when the data items exhibit a high degree of correlation.
- The interpretation of tree results is easy.
- The accuracy of the results generated using parallel decision trees is high compared to the sequential ones.

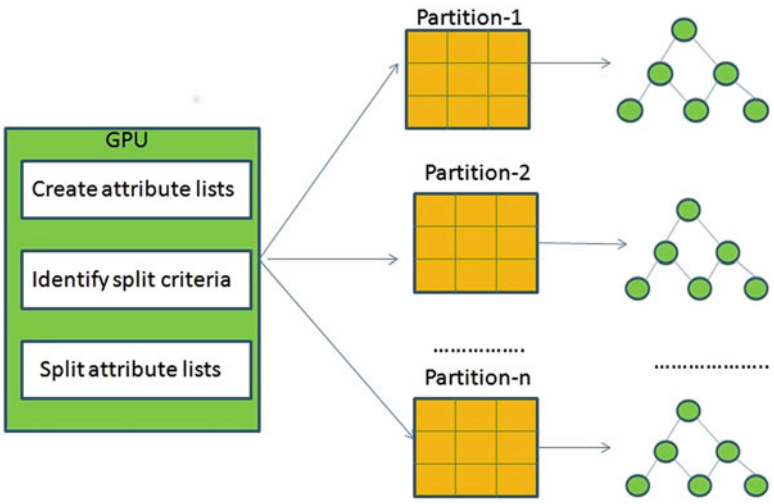


Fig. 5.2 Parallel decision tree

Demerits

- The technique is unstable with respect to the noisy dataset as the synchronization between the trees is less.
- It does linear classification of the dataset; the performance decreases when there is a need to draw soft margin.
- On creation of more bushy trees in parallel, the load imbalance ratio is high.

5.3.2 Parallel Neural Network

It is one of the fastest adaptive learning models with the ability to remember the patterns by adjusting the weight of input variables through block mode backpropagation learning mechanism. Here the parallelism can be achieved in two ways; one is by parallelizing the input data with the help of POSIX threads, and other is parallelizing the data nodes with the help of SIMD (single instruction, multiple data) combined with CUDA (Compute Unified Device Architecture). The GotoBLAS linear algebra library function is used to accelerate the data partitioning and summing up activities.

In input data parallelization method, the input is partitioned into several threads, and the weight synchronization is being carried out periodically for every thread and summed up at last. The GotoBLAS linear algebra library function is used to accelerate the data partitioning and summing up activities. In the case of node parallelization method, the network layers are divided into disjoint partitions of neurons; those neuron outputs are combined to provide output. The CUDA and CUBLAS framework along with CULA library function is used to accelerate the linear and nonlinear operations inside the kernel to achieve node-level parallelism. Figure 5.3 shows the parallel implementation of a neural network using multiple POSIX threads and multiple cores of GPU [5]. The merits and demerits of parallel implementation of parallel neural network are listed below.

Merits

- The neural networks achieve a high degree of parallelism by doing a proper trade-off between CPU and GPU performances.
- The parallel implementation of artificial neural networks with multiple hidden layers makes the analysis process of complex problems easier.

Demerits

- The activation function of the parallel neural network is harder to implement.
- The performance of the parallel neural network is worse compared to its nonparallel implementation when the size of the input is large.

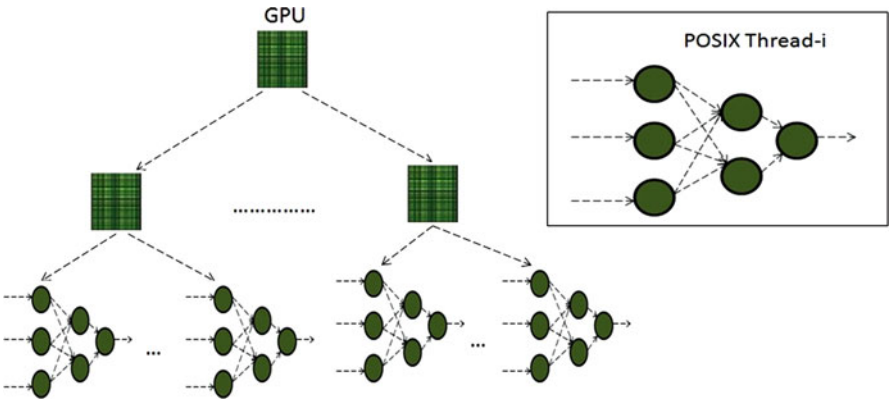


Fig. 5.3 Parallel neural network

5.3.3 Optimized Random Forest

Random forest is a supervised machine learning-based classifier; it is a collection of decision trees arranged in the random order used to classify big datasets. The parallelization of individual decision trees is usually carried out in two ways; one is achieving data parallelism through depth first search mechanism, and another is task parallelism through breadth first search mechanism [6].

CudaTree is GPU implementation of random forest which speeds up the decision tree creation process by parallelizing the decision tree creation process on multiple cores of GPU and prevents frequent switching between coarse-grained and fine-grained tasks. BIDMachRF is a GPU-CPU framework of random forest with high scalability, and it is ten times faster than CudaTree. The BIDMachRF works on the principle of achieving maximum parallelism at the microlevel and also removes unnecessary data transfer operation between cores. CURFIL is GPU based random forest used for image labeling which basically predicts the RGB colors in the image and does hyper parameter optimization to achieve higher throughput. Figure 5.4 shows ensemble learning process of a set of decision trees in every core of GPU. The merits and demerits of random forest algorithm are listed below.

Merits

- The over-fitting problem occurrence is less as the summation of several tree outputs are taken into consideration.
- The accuracy of the output achieved is high with the ensemble of many decision trees.
- Practical feasibility is more as it can be used for both classification and regression related jobs.

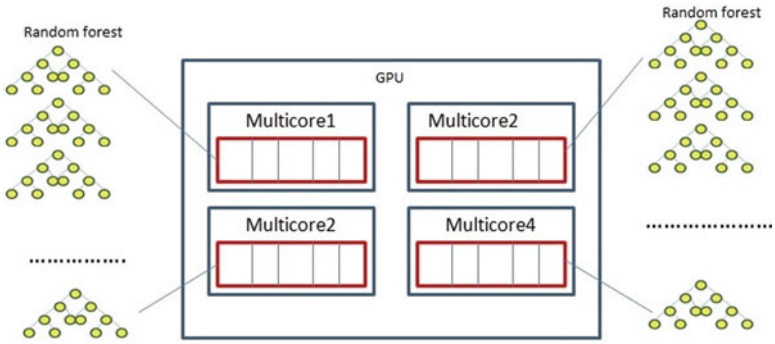


Fig. 5.4 GPU based random forest

Demerits

- Its parallel implementation is tedious as there exists; it requires a lot of synchronization between individual decision trees.
- The training time of the algorithm is high, and the accurate predictions are drawn very slowly.

5.3.4 Deep SARSA Learning

Deep SARSA learner is a State-Action-Reward-State-Action form of reinforcement learning algorithm. It overcomes the Q-learning mechanism as Q-learning always tries to find local optimal solution, whereas SARSA learning takes each and every action of the agent into control and keeps updating its actions to find a globally optimal solution. Julia is a dynamic programming tool which supports SARSA learning through CPU and GPU parallelization with the help of packages like OpenCL.jl, CUDAnative.jl, CUDAdrv.jl, CUDArt.jl, and GPUArrays.jl. To achieve distributed parallelism for SARSA learning among multiple TensorFlow GPUs, OpenAI Gym packages are used [7]. A multiple GPU-based deep SARSA parallel learning environment with stages from Q1 to Qn is depicted in Fig. 5.5. The merits and demerits of SARSA algorithm are listed below.

Merits

- Achieves high speed by employing exploratory behavior policy during the learning stage of the agents.
- Supports asynchronous learning process by doing n step learning activities.
- The usage of locality of state space while computation accelerates the convergence rate of the algorithm.

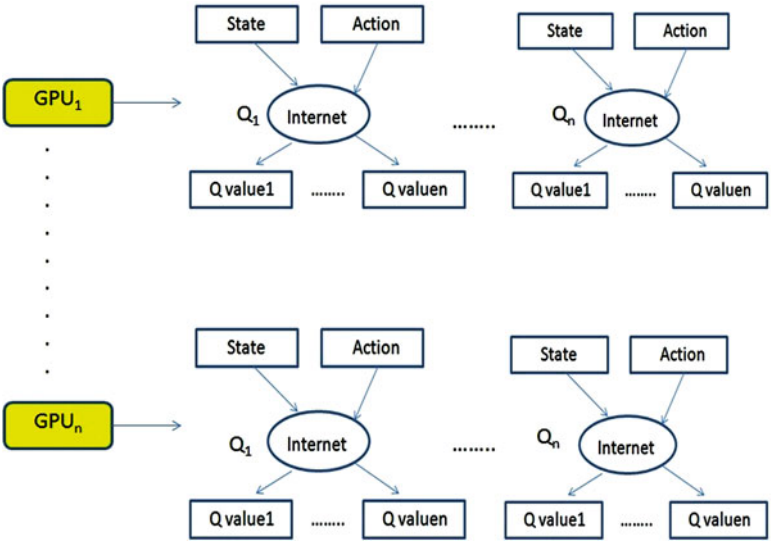


Fig. 5.5 Deep SARSA learning

Demerits

- The communication cost is high due to the exchange of large volume of information between the agents.
- The frequent updating of agent learning state in several cores leads to a deadlock situation.

5.3.5 Deep Q-Learning

The deep Q-learning is also another reinforcement learning algorithm used by the agents to perform optimal actions using Markov decision process. The Q-learning process in depth is supported by Deeplearning4j machine learning library, which is popularly referred as RL4J (reinforcement learning 4J) version library. The procedure to train a robot to play Atari game using deep Q-learning was first introduced by Google in 2013 with the help of GPU-based machine learning packages like Keras and Gym [8]. The pictorial representation of a deep Q-learning network with one input layer, two hidden layers, and one output layer is shown in Fig. 5.6; the nodes in the output layer are enabled with Q-learning algorithm. The merits and demerits of Q-learning algorithm are listed below.

Merits

- Accurate decision-making ability as it takes soft actions by exploring best policies.

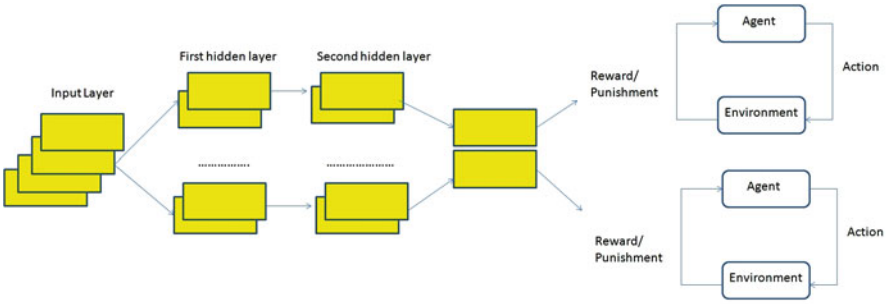


Fig. 5.6 Deep Q-learning

- The learning ability of the agent is flexible as it always tries to find a best optimal solution.
- The parallel implementation of Q-learning agents is successful on continuous and discontinuous input datasets.

Demerits

- Chances of getting trapped into local minimal are more.
- The policies made by the agents are fast but in terms of stability, it is weak.

5.3.6 Parallel K-Means

It is an unsupervised clustering algorithm used to cluster high-dimensional vectors nearly of size 128 dimensions by achieving parallelism at the task level. The K-means implementation on GPU uses master-slave architecture, the master node sends the incoming data to worker nodes by computing centroid, and the worker nodes in parallel compute new centroids to perform data-intensive operation. Here the job of worker nodes is equally distributed among n CPUs in a sequential manner, and their mean is calculated by the master node in a GPU. It also provides differentiated Quality of Service (QoS), by classifying the input dataset as low-dimensional and high-dimensional dataset based on their service requirements. For low-dimensional dataset, the processing is carried out in CPU registers, whereas for high-dimensional dataset both GPU and CPU registers are used. NVIDIA G80 PCI board with CUDA framework supports the parallel implementation of K-means on GPU. Many varieties of K-means algorithms are implemented on CUDA which includes GPU Miner, UVKMeans, and HPKMeans [9]. The implementation of K-means algorithm using master-slave architecture in shared memory of GPU is given in Fig. 5.7. The merits and demerits of parallel implementation of K-means algorithm are listed below.

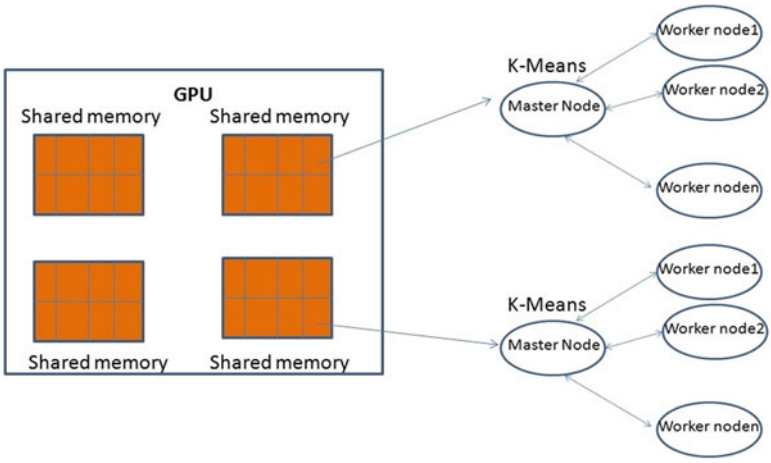


Fig. 5.7 Parallel K-means

Merits

- The multi-core implementation of K-means is simple, fast, and easy to implement.
- Creates the clusters with uniform variance, as a result, the cluster result interpretation is easy.
- The speed of clustering is faster as it recursively divides the input space into disjoint subspaces.

Demerits

- The wrong selection of cluster centers leads to inappropriate results.
- The performance is weak on the nonlinear and noisy dataset.
- Uneven size input partitions lead to the formation of global clusters and predicting K-value for such clusters is difficult.

5.3.7 GPU-NB

GPU-NB is most widely used parallel version of naive Bayes classifier for classifying the documents. It decides the probability of assigning a document to a particular category based on the probabilistic model. Both training and testing phases of the algorithm are parallelized; first, the training phase is parallelized by using GPU kernels which improve the training speed by reducing the data dependency between the training phases. The dense probability matrix is formed by using the learning GPU kernel; at last, the documents are classified to a particular class which exhibited the highest probability. The CUDA kernels are more commonly used for training and classification purpose along with the NVIDIA GeForce GTX 460 GPU cards [10, 11]. The GPU kernel learning and testing phases using NB algorithm over

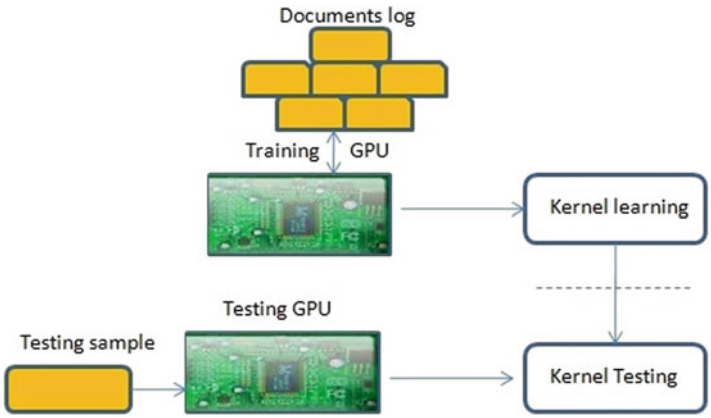


Fig. 5.8 GPU-NB

huge document log are shown in Fig. 5.8. The merits and demerits of parallel implementation of NB algorithm are listed below.

Merits

- It is the simplest form of classifier.
- The performance of the algorithm is good with respect to multi-class problems.
- The classifier robustly handles the bell shaped or spherical shaped input dataset.
- The parallel version of NB is widely accepted classifier for classification of text documents.

Demerits

- The prediction output of NB is weak when there is scarcity of available dataset.
- The practical feasibility of the approach is less as it assumes that the input predictors are totally independent of each other.
- The process of mining continuous features from the dataset for likelihood estimation is not clear.

5.3.8 Accelerated AdaBoost

AdaBoost is used to recognize objects (car, face, pedestrian, iris, fingerprint, and so on) precisely even when the input data sample is huge, highly volatile, and uncertain in nature. The object recognition and classification is carried out at high speed in parallel with the help of GPUs. The training of the algorithm is carried out without any dependencies by modifying the weights of the input samples, and the optimal sample is identified by spooling multiple GPU streams without any overlapping situations [12]. Figure 5.9 illustrates a sample image recognition scenario by mining every pixel of the target object using the AdaBoost algorithm. The merits and demerits of AdaBoost algorithm are listed below.

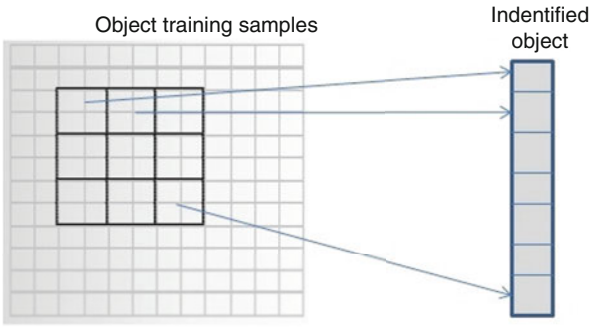


Fig. 5.9 Image recognition using AdaBoost

Merits

- The parallel implementation of the classifier is fast due to its self-adaptive nature.
- It is computationally efficient against illumination changes of the input data.
- The classifier by default uses hierarchical models which increases its computational efficiency.
- The accuracy of the result is high due to fairly good generalization mechanism.

Demerits

- It is cost expensive whenever subjected to large instances of input data samples.
- The parallel implementation is sensitive to outliers and noisy input data.
- The algorithm often gets stuck in suboptimal solutions due to interpolation problems.

5.3.9 Deep Belief Network

It is the process of training a computer in depth with hierarchical models by dividing the original dataset into multiple subsets based on different characteristics exhibited by the subsets. The deep algorithms usually contain many nonlinear operations which are arranged in multiple levels to form deep belief networks. Some of the recent deep learning techniques used to learn complex decision-making tasks are Adadelta, Adagrad, Adam, affine neural network, Alexnet, backpropagation with respect to time, bidirectional recurrent neural network, Deep Dream, GloVe (Global Vectors for Word Representation), Keras, negative log likelihood, and so on. The AlchemyAPI is an IBM-based company, which is the leading provider of deep learning oriented cloud services with many APIs like AlchemyData News API, AlchemyLanguage, AlchemyVision, and so on http://timdettmers.com/2017/04/09/which-gpu-for-deep-learning/. The concept of deep learning in multiple layers with increasing depth levels ranging from one to n in every layer is shown in Fig. 5.10. The merits and demerits of deep belief network are listed below.

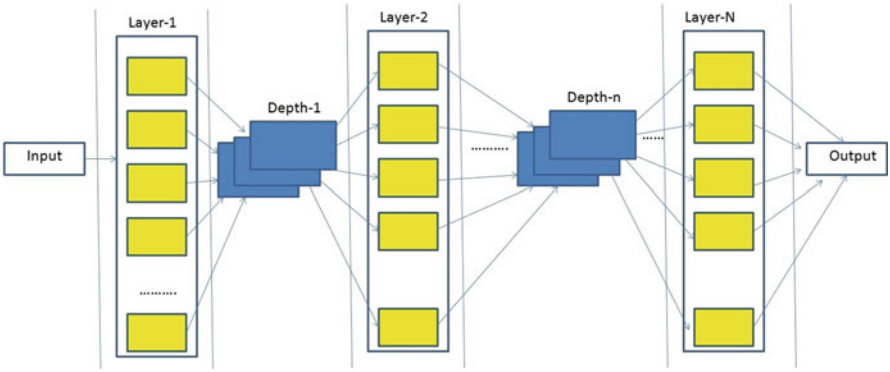


Fig. 5.10 Deep belief network

Merits

- The deep belief networks are structured uniformly with many layers of depth which make it highly suitable for computation-intensive jobs, and they are able to perform billions of computation per unit of time
- The architecture of deep networks suites well with the GPU; the computation speed offered by the combination of GPU and deep network is extremely high.

Demerits

- For accurate predictions, the algorithm needs to be trained with more number of training examples which results in more training time.
- The process of determining the exact values for hyper parameters to train the deep belief network is not transparent.
- Parallelizing deep networks over multiple GPUs is a difficult operation.

5.3.10 Parallel Support Vector Machine

It is one of the powerful supervised machine learning-based classifiers to perform high computation involved jobs. The SVMs are classified into two types: one is binary SVM and multi-class SVM. The binary SVM is used mainly to solve a convex optimization problem using SMO (sequential minimal optimization) algorithm using GPU devices. The SMO algorithms inherently make use of second order WSS (working set selection) logic to do classification at a faster rate. For multi-class classification using SVM, multi-threaded approach with GPU at task level but the memory allocation and launching functions are assigned for CPUs only [13, 14]. The trade-off use of CPU and GPU allows the multiple SVM based classifier to scale at a higher rate. Figure 5.11 depicts the training and testing phases of multiple SVM-based classifiers in a parallel programming environment. The merits and demerits of parallel SVM is listed below.

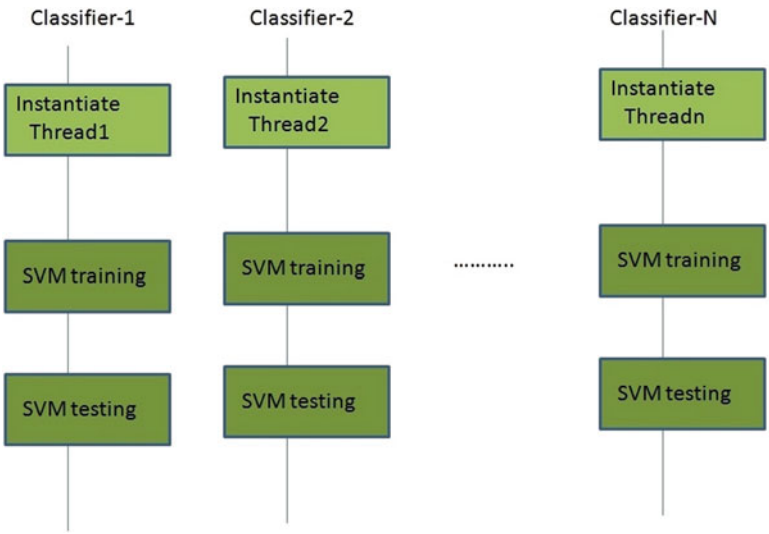


Fig. 5.11 Parallel SVM

Merits

- Delivers accurate novel solution for convex optimal problems.
- With the Gaussian kernel, the simplicity of the algorithm increases, and it evenly separates the input data samples.
- It enforces efficient capacity control scheme among multiple support vector machines by optimizing the margin and reduces the chances of a locally optimal solution.

Demerits

- The interpretation of the result is difficult due to the lack of transparency.
- The learning time of support vector machine is in polynomial times.
- It is suitable for binary classification problems as the approach fails on multi-class problems.
- The misclassification rate is high; it does not consider the history of domain knowledge.

5.3.11 Large Scale Linear Regression

The least median of square model is one of the most commonly used linear regression models for multi-threaded GPU programming. The parallel implementation of linear regression models does mining and analysis operations on the large dataset at a faster rate and achieves high forecasting accuracy also. The GPGPU (general

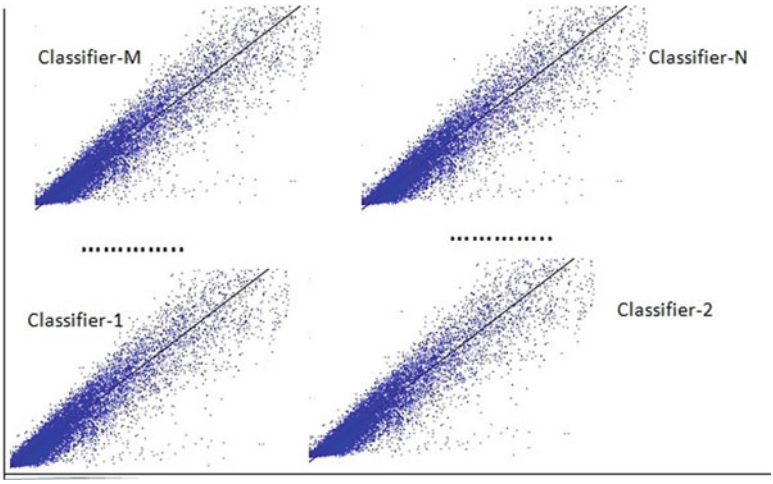


Fig. 5.12 Parallel linear regression

purpose GPU) kernels are most commonly used to handle big data and to solve several floating point arithmetic operations with more precision using machine learning algorithms, and by statistically modeling the power between GPU kernels, the regression operations are carried out robustly even in the presence of several outliers to arrive at timely solutions [15]. The simultaneous operation of multiple linear regression model-based classifiers is illustrated in Fig. 5.12. The merits and demerits of linear regression are listed below.

Merits

- The linear model is simple and easy to implement.
- The forecasting accuracy is high as it efficiently mines the hidden patterns in the dataset.
- Most commonly used modeling tool which can automatically map the N-dimensional datasets.

Demerits

- Its scope is limited to datasets exhibiting a linear relationship as the approach fails on the dependent dataset.
- The accuracy of the prediction goes down with the presence of outliers.
- It can only map the N-dimensional input space to one-dimensional output space, which makes it suitable only for linear separable data items.

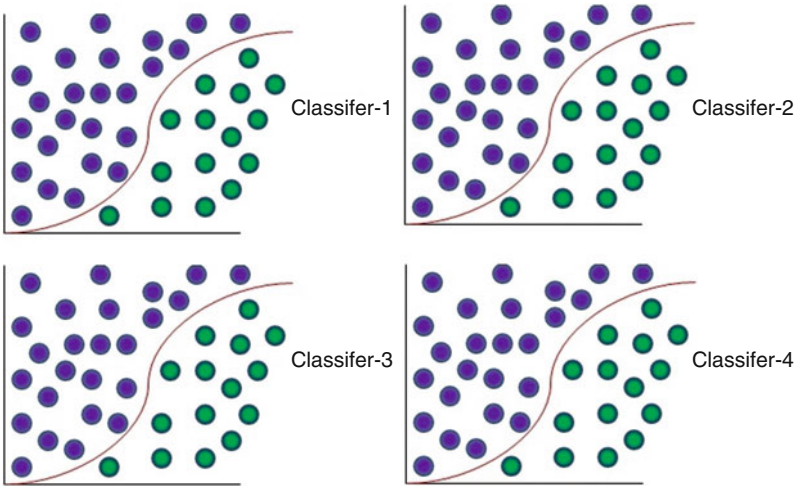


Fig. 5.13 Parallel logistic regression

5.3.12 Large Scale Logistic Regression

Logistic regression is one of the oldest binary classification algorithm used to forecast the chances of occurrence of an event using logit function. The sequential implementation of logistic regression is slow with respect to high-dimensional problems, but the parallel implementation of logistic regression model using GPU exhibits a high degree of suitability towards high-dimensional problems. The CUDNN.torch is the popular GPU-based library which facilitates the parallel implementation of logistic regression using torch learning framework [16, 17]. The simultaneous operation of multiple logistic regression classifiers is depicted in Fig. 5.13. The merits and demerits of logistic regression are listed below.

Merits

- It is capable enough to represent nonlinear relationships between the data items.
- Robust enough to handle the noise and floating point variance in the input as the independent data items are equally distributed among the groups.

Demerits

- The training time of the algorithm is high as it needs to be trained with more examples in order to attain the stable state.
- The prediction accuracy is less as it achieves performance at a lower bound.

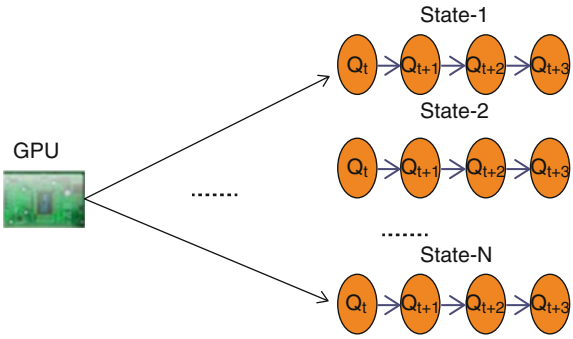


Fig. 5.14 Parallel HMM

5.3.13 Massively Parallel Hidden Markov Model (HMM)

The HMM is usually used to perform sequential classification operations based on Markov property using a set of hidden states, but nowadays the parallel implementation of HMM is used to perform computation-intensive jobs like handwriting recognition, face recognition, signal detection, pattern mining, gene analysis, and so on. The parallel version of HMM-based algorithms is Baum-Welch, Viterbi, and forward, in which the performance of Viterbi and Baum-Welch are found to be good. The CUDA and OpenCL offer general purpose frameworks for HMM applications and for parallel implementation of HMM; the thread blocks are considered where the Markov model is evaluated by considering the forward probability of every thread in each thread block. The HMMER, HMMoC, and GPU-HMMER are the popular tool to create GPU-based hidden Markov models [18]. A working of HMM with N states on every core of a GPU is shown in Fig. 5.14. The merits and demerits of parallel HMM are listed below.

Merits

- The simultaneous execution of multiple Markov models in several cores of GPU increases the rate of completion of jobs.
- The data parallelism is achieved at a higher rate by exploiting the data independence feature of the Markov chain.
- The learning model is most accurate as it can learn from a raw sequence of dataset and exhibit efficient generalization capability even in the presence of variable length input dataset.

Demerits

- It is basically a sequential classifier, parallelizing it takes time.
- The performance is weak with a discrete set of data samples.
- The model fails to capture higher order correlation among the items datasets.
- The computation power required to evaluate the success or failure of the model is too expensive.

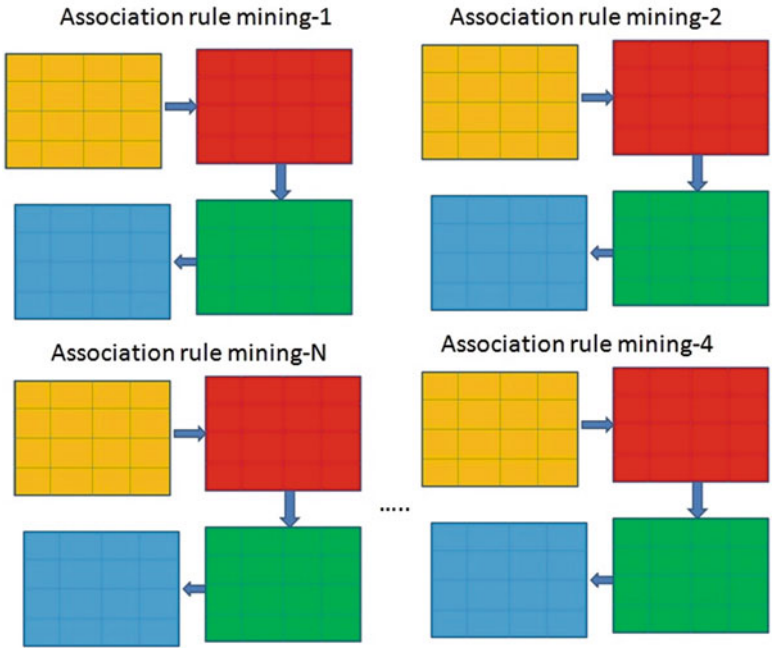


Fig. 5.15 Parallel Apriori

5.3.14 Parallel Apriori

Apriori algorithm is used to determine associated data items in transaction database by mining the hidden patterns among the data items using association rules. The parallel implementation of Apriori algorithm is in CUDA architecture; the algorithm speed increases with the increase in dataset size as it exhibits superior floating point computation capability. The GPApriori is a GPU-based implementation of Apriori on NVIDIA Tesla T10 GPU card, whose speed is 100 times better than sequential Apriori; the high speed is achieved by representing the candidate dataset in the vertical bit set form. OpenCL (Open Compute Language) is the most commonly used Java based GPGPU framework for implementation of Apriori algorithm [19–21]. A simple example showing the parallel association rule mining process of Apriori algorithm is given in Fig. 5.15. The merits and demerits of parallel Apriori algorithm are listed below.

Merits

- The classifier is easily parallelizable.
- The implementation of it is easy.
- Exhibits a high degree of scalability by exploiting spatial locality property of the data items.
- Communication cost is less due to the fewer interaction between processors.

Demerits

- Due to the construction of entire hash tree in every processor, the memory utilization is not optimal.
- Processing overhead increases with the increase in dataset size.
- The tendency of extracting not so formal association rules in two-dimensional databases is more, which decreases the performance of the classifier.

5.4 Accelerated Machine Learning Based Big Data Analytic Applications

Some of the modern big data analytics applications, which use GPU-based machine learning algorithms for acceleration and accuracy are given below.

Modern Facebook Faces Recognition with Deep Learning Earlier to recognize friends in the Facebook, the friend name was supposed to be tagged, but now it is able to automatically recognize the friends and pop up their name by just seeing the photograph. This is achieved by using a nine-layered neural network with around 120 million synaptic weights which are trained with four million images of the Facebook users.

Emerging Twitter Hashtag Prediction The machine learning classifier algorithms like a decision tree, SVM, NB, k-nearest neighbors, and logistic regression are used to predict newly emerging hashtags based on the content features of hashtag string and contextual features of users.

Twitter Users Latent Attributes Prediction The Twitter usually doesn't provide personal information while creating the user account, but with the help of more user information, it is possible to recommend relevant content to users. Supervised machine learning technique based on user LinkedIn profile or social networking website and unsupervised learning technique based on Markov Chain Monte Carlo algorithm are used to predict attributes related to users.

Smartphones Deloitte Global predicts that around 300 million smartphones provide machine learning services by understanding the user of the phone during 2017. The phone uses chips on which low-power machine learning processors run to provide artificial intelligence-based services like profiling the driver behavior to prevent accidents, providing health predictions to the owner, dynamic navigation guidelines while driving, etc.

Driverless Cars Reinforcement learning algorithms are used in autonomous car based on the data collected from the IoT devices to provide services like automatic navigation on road, redirecting the car to hospital if the driver's condition is bad, applying brakes during emergency situations, estimating the cars approaching quickly from behind, playing songs according to the mood of the owner, and so on.

Precision Medicine Unsupervised machine learning algorithms K-means and supervised algorithm like the random forest are most widely used to identify multi-factorial disease and provide treatment by keeping track of the individual lifestyle and gene characteristics. The services offered by precision medicine include finding suitable blood group while donating blood, identifying the type of cancer based on genetic profiles, finding the root cause of disease via diagnostic analysis, etc.

E-Commerce E-commerce is made up of lots of data; supervised machine learning algorithms like decision tree, naive Bayes, and support vector machine are executed on the numerous business-related data to provide smart services like identifying the likelihood of purchasing the items, clustering the customers based on their interest, content-based filtering to offer personalized products, detecting anomaly in the shopping data, and so on.

Natural Language Processing Convoluted or recurrent neural networks are being used to automatically process natural language and have yielded successful output in academic and industrial point of view.

Computer Vision Computer vision enabled with machine learning operations are used to solve image processing problems in an orthogonal manner. Some of the image processing problems solved are recognizing the objects from the noisy text, scaling the image, removal of background clutter, carrying out intra-class variations in the image, identifying analogy of documents, etc.

5.5 Conclusion

The paper provides a brief discussion on the basics of big data analytics, highlights issues while analyzing the big data, and provides a comprehensive discussion on some of the popular GPU-based machine learning algorithms used for accelerated big data analytics applications.

References

1. Hua, F., Zhaoyang, Z., Chanpaul, J.W., Mahmoud, D., Chonggang, W., Honggang, W.: A survey of big data research. IEEE Netw. **29**(5), 6–9 (2015)
2. Acharjya, D.P., Kauser, A.P.: Survey on big data analytics: challenges, open research issues and tools. Int. J. Adv. Comput. Sci. Appl. 7(2), 1–11 (2016)
3. Win-Tsung, L., Yue-Shan, C., Ruey-Kai, S., Chun-Chieh, C., Shyan-Ming, Y.: CUDT: a CUDA based decision tree algorithm. Sci. World J. **2014**, 745640 (2014)
4. Toby, S.: Implementing decision trees and forests on a GPU. In: Computer Vision-ECCV 2008. Lecture Notes in Computer Science, vol. 5305, pp. 595–608. Springer, Berlin (2008)
5. Raghavendra, D.P.: GNeuron: parallel neural networks with GPU. In: International Conference on High Performance Computing, posters (2007)

6. Mitchell, L., Sloan, T.M., Mewissen, M., Ghazal, P., Forster, T., Ptotwski, M., Andtrew, A.S.: A parallel random forest classifier for R. In: Proceedings of the Second International Workshop on Emerging Computational Methods for the Life Sciences (2011)
7. Dongbin, Z., Haitao, W., Shao, K., Yuanheng, Z.: Deep reinforcement learning with experience replay based on SARSA. In: IEEE Symposium Series on Computational Intelligence (SSCI). This work was supported in part by National Natural Science Foundation of China, IEEE (2016)
8. Iuri, F., Stephen, T., Jason, C., Jan, K.: GA3C: GPU-based A3C for deep reinforcement learning. In: 30th Conference on Neural Information Processing Systems (NIPS 2016)
9. Mario, Z., Michael, G.: Accelerating K-means on the graphics processor via CUDA. In: First International Conference on Intensive Applications and Services, IEEE (2009)
10. Lei, Z., Hai, J., Ran, Z., Xiaowen, F.: Effective naive bayes nearest based image classification on GPU. J. Supercomput. **68**(2), 820–848 (2014)
11. Felipe, V., Guilherme, A., Jussara, A., Gabriel, R., Leonardo, R.: GPU-NB: a fast CUDA-based implementation of naive bayes. In: International Symposium on Computer Architecture and High Performance Computing (2013)
12. Pin, Y.T., Yarsun, H., Ching-Te, C., Tsai-Te, C.: Accelerating AdaBoost algorithm using GPU for multi-object recognition. In: IEEE International Symposium on Circuits and Systems (ISCAS) (2015)
13. Bryan, C., Narayanan, S., Kurt, K.: Fast support vector machine training and classification on graphics processors. In: 25th international Conference on Machine Learning. ACM (2008)
14. Quan, L., Jibo, W., Yue, W., Watson, I.A.: GPU accelerated support vector machines for mining high-throughput screening data. J. Chem. Inf. Model. **49**(12), 2718–2725 (2009)
15. Vaibhav, M., Mayank, G: Data regression with normal equation on GPU using CUDA. Int. J. Comput. Sci. Inf. Technol. Secur. **2**(2), 418–422 (2012)
16. John, C.: Extreme machine learning with GPUs. Computer Science Division, University of California, Berkeley (2014)
17. Larsen, A.B.L.: CUDArray: CUDA-based NumPy. DTU Compute Technical Report (2014)
18. Chuan, L.: cuHMM: a CUDA implementation of hidden Markov model training and classication. The Chronicle of Higher Education (2009)
19. Spandana, K., Sirisha, D., Shahida, S.: Parallelizing Apriori algorithm on GPU. Int. J. Comput. Appl. **155**(10), 22–27 (2016)
20. Fan, Z., Yan, Z., Jason, B.: GPApriori: GPU-accelerated frequent itemset mining. In: IEEE International Conference on Cluster Computing (2011)
21. William, A., Fayaz, K., Veerabhadra, B.: HSApriori: high speed association rule mining using apriori based algorithm for GPU. Int. J. Multidiscip. Curr. Res. **2**, 759–763 (2014)

Chapter 6
A Novel Perfect Privacy PIR Scheme for Privacy Critical Applications

Radhakrishna Bhat and N. R. Sunitha

Abstract Majority of the business vendors have incorporated the policy-driven privacy setting that greatly disappoints the end user or customer even though the protocol-based privacy assurance is strongly expected. The reason behind this major disagreement between the vendor and the customer is due to the business survival necessity or the business expansion for the vendor and the mandatory technology adoption or the business monopoly creation for the customer. In order to cope up with the exponentially growing business needs, both vendor and customer have to agree upon the protocol-based privacy setting. Therefore, we have proposed a new generic (i.e., applicable for both client-server and peer-to-peer) perfect privacy preserving information retrieval protocol using the concept of Private Information Retrieval (PIR).

More interestingly, we have overcome the trivial solution of downloading the entire database by achieving $o(n)$ communication cost by introducing a new perfect privacy preserving single database private information retrieval for privacy critical applications using quadratic residuosity as the underlying *data privacy* primitive. Finally, we have concluded by claiming a generic scheme suitable for privacy critical applications.

Keywords Perfect privacy · Private information retrieval · PIR · Quadratic residuosity · Privacy critical applications

6.1 Introduction

The drawback of most of the privacy-enabled retrieval techniques of the server is that they either adopt information theory-based information-theoretic or cryptographic assumption-based computationally bounded privacy techniques which

R. Bhat (✉) · N. R. Sunitha
Siddaganga Institute of Technology, Tumakuru, Karnataka, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_6

are assuring only partial privacy. But the user assumes that his privacy is always guaranteed by the other party. This serious monopoly move by the server leads to several problems.

Scenario-1: Let us consider that all the patients pertaining to a disease of some region have uploaded all their health cards in a plain or encoded format on a server by considering policy-driven privacy assurance.

The server has several possible moves. What if the server shares the stored patient information with other healthcare industries that are eager to know the disease count there by producing more products? What if the server shares the information with other regional bodies which are already tied up for food and nutrition exchange policies?

What is the mere negative consequence? The healthcare industries may pressurize the government body to exchange such food that creates sufficient malnutrition or set a customized health boundary point so that more patients should be included in that boundary. In turn, analytics-enabled pharmaceutical industries may advertise their products so that the physicians should refer to the same products. If this business cycle continues, then at some point in time, all people will become patients.

Scenario-2: Let the public database maintain all the information particular to its domain (e.g., search engines, social media, multimedia, patent, etc.). Let the authenticated user search or retrieve the subset of information from the database to which he is subscribed to. What happens when the analytics-enabled server tracks all the search or retrieval sequences of a particular user or a group of users related to a particular domain? What if the server shares its analytical results with user's business opponent?

Scenario-3: Let us consider that a severe war is happening between two rivals on the war field. What if the global positioning system (GPS) server tracks and shares one of the opponent's livestream information to others?

Scenario-4: Suppose if any two peer devices like two military commanders want to share secret information securely and privately through insecure communication channel or through a mediator? Also, if any two end devices want to communicate with end-to-end encryption enabled like "private chat"? What if the mediator or the third party reveals the communication information?

Perfect Privacy Solution: To overcome the above problems, the only solution is to shift from policy-driven privacy architecture to protocol-driven privacy architecture. Therefore, we have introduced a new protocol-driven (i.e., scheme level privacy support) perfect privacy-preserving information retrieval scheme using a concept called private information retrieval (PIR).

Private information retrieval [7] is one of the ways of reading the bit information from the other party privately, and private block retrieval (PBR), a realistic extension of PIR, is the way of reading single block information from the database privately. We have successfully constructed a new PBR scheme in a single database setting

which neither belongs to information-theoretic (i.e., no replicated database) nor belongs to computationally bounded (i.e., no privacy assumptions). The proposed scheme fully supports "perfect privacy," i.e., all the queries are mutually exclusive and give no information (not even partial information) about the user privacy. Note that the proposed scheme conventionally uses the term "PIR" but PBR by default until and unless externally stated. Note that the *privacy* refers to the user privacy, and *perfect privacy* refers to zero percent privacy leak until and unless externally stated.

The construction uses "quadratic residuosity" as the underlying data privacy operation. Note that the quadratic residuosity property is only used for preserving data from the intermediate adversaries. This property is not related to hide the privacy of the user, i.e., even on identifying the quadratic residuosity property of the numbers sent in the query, server gains no information about the user's interest. By this, we claim that the construction supports perfect privacy, and the success probability of identifying user's interest is equal to "random guessing."

Finally, we have achieved the following results.

- We have successfully overcome the trivial database download requirement claimed by [7] by achieving the overall communication cost as $o(n)$ where n is the database size which is surely a non-trivial communication as claimed by [9].
- The protocol is generic in nature and can be adopted by both client-server and peer-to-peer privacy critical applications.

Related work: Extensive work has been carried out on PIR by various researchers to fulfill the trade-off between communication and computation overheads, to preserve the user as well as server privacy, handling fault tolerance and integrity. The PIR is mainly classified into two categories in which one relay on non-colluding server replication and the other relay on single database with limited computation power.

Information-Theoretic PIR (itPIR) In order to provide protocol-driven privacy, Chor et al. [7] introduced the concept of private reading from k replicated databases and further improved the communication cost in [8] using XOR operations. There are several other improvements introduced by [1, 2] over communication and computation overheads in itPIR setting. Gertner et al. [13] highlight on the data privacy of the server along with the user privacy using the concept of conditional disclosure of secrets.

Computationally Bounded PIR (cPIR) The first quadratic residuosity assumption-based privacy-preserving PIR scheme was introduced by [16] in a single database setting with sub-polynomial communication cost. Chor and Gilboa [6] also presented a one-way function-based PIR scheme with the minimal database replication to achieve only computationally bounded privacy. Cachin et al. [4] presented ϕ-hiding-based scheme with polylogarithmic communication cost. Ishai et al. [15] introduced an efficient cPIR scheme using anonymity techniques. Aguilar-Melchor and Gaborit [19] introduced fast cPIR scheme based on coding theory

and lattice assumptions. Groth et al. [14] proposed multi query cPIR with constant communication rate. Jonathan and Andy [24] improve computational complexity of existing cPIR using trapdoor groups. Kushilevitz and Ostrovsky [17] presented a computationally intractable cPIR using one-way trapdoor permutations. Chang [5] presented a computationally bounded PIR with logarithmic communication using Paillier cryptosystem as the underlying intractability assumption. Gentry and Ramzan [11] presented a PBR scheme with log-squared communication using a decision subgroup problem called ϕ-hiding assumption. In order to protect both user and server privacy, several *oblivious transfer* (OT) schemes [9, 18, 20, 21] have also been introduced in a single database setting. The first keyword-based PIR search [3] has been introduced to apply PIR on the existing server data structure.

Perfect Privacy The term "perfect privacy" as defined in [7] strongly suggests the requirement of the uniformly distributed probability for any two random variables (PIR queries are treated as the random variables). The first information-theoretic single-database PIR scheme was introduced by [12] and recently by [22]. In order to preserve the user privacy in multiuser setting, input anonymity by secret sharing technique is presented by Toledo et al. [23].

Organization: The rest of the paper is organized as follows. The required notations and preliminaries are described in Sect. 6.2; the preliminary modules, the proposed PIR scheme, and the performance analysis along with the required security proofs are all described in Sect. 6.3; and, finally, the open problems are listed along with the conclusion in Sect. 6.4.

6.2 Notations and Preliminaries

Let $[u]$=$\{1, 2, \cdots, u\}$ and $[1, u]$ be the method of selecting all the integers from 1 to u; $\mathcal{DB}_u^{b_v}$ is a set of u number of v bit matrix. Let N=pq (where $N \xleftarrow{R} \{0,1\}^k$ with the security parameter k) be the RSA composite modulus, and $\mathcal{S}_{QR}$, $\mathcal{S}_{QNR} \subseteq \mathbb{Z}_N^{+1}$ are the quadratic residue and non-residue subsets respectively. Let $\mathcal{JS}$ and $\mathcal{LS}$ be the *Jacobi* and *Legendre* symbols respectively. Let c be the total number of l-bit groups of a database block. Let $\mathcal{U}$ be the end user or the client or the intended service seeker and $\mathcal{S}$ be the server or the intended service provider.

Quadratic residuosity: $\forall x, y \in \mathbb{Z}_N^{+1}$, if $x \equiv y^2 \pmod N$ then $x \in \mathbb{Z}_N^{+1} \backslash \mathcal{S}_{QNR}$, i.e., $x \in \mathcal{QR}$; otherwise $x \in \mathbb{Z}_N^{+1} \backslash \mathcal{S}_{QR}$, i.e., $x \in \mathcal{S}_{QNR}$.

Definition 1 (Trapdoor Function of [10]) : $\forall x \in \mathbb{Z}_N^*$, $r \in \mathbb{Z}_N^{-1}$, $s \in \mathcal{S}_{QNR}$, $\forall jx, hx \in \{0,1\}$, the function $\mathcal{T}(x,r,s) = (x)^2 \cdot r^{jx} \cdot s^{hx}$=$z$ such that jx=1 if $\mathcal{JS}_N(x)$=-1 otherwise $jx = 0$ and hx=1 if $x > \frac{N}{2}$ otherwise hx=0. The inverse function $\mathcal{T}^{-1}$ is defined as $\mathcal{T}^{-1}(z) = \sqrt{(z) \cdot r^{-jx} \cdot s^{-hx}}$=$x$. The generalized formula for l number of inputs is given as $\mathcal{T} : (x_1, \cdots, x_l, r, s) \rightarrow z_1, \cdots, z_l$ where $\mathcal{T}(x_1, \cdots, x_l, r, s)$=$\mathcal{T}_1, \cdots, \mathcal{T}_l$ and $\mathcal{T}_i(x_i, r, s)$=$(x_i)^2 \cdot r^{jx_i} \cdot s^{hx_i}$=$z_i$, $i \in [1, l]$.

We have used a slightly modified version of the above trapdoor function for our proposed PIR scheme and is given as $\mathcal{MT} : x \rightarrow (z, t), t \in \{0, 1\}$ where $\mathcal{MT}=x^2=z$ and t is assigned with the "hx" value of the input x. The generalized formula for l number of inputs is given as $\mathcal{MT} : (x_1, \cdots, x_l) \rightarrow ((z_1, \cdots, z_l), (t_1, \cdots, t_l))$ where $\mathcal{MT}(x_1, \cdots, x_l)=\mathcal{MT}_1, \cdots, \mathcal{MT}_l$ and $\mathcal{MT}_i(x_i)=x_i^2=(z_i,t_i), i \in [1, l]$.

Definition 2 (Perfect Privacy PIR Query) : If any two randomly selected PIR queries are independent of block reference (or block index), i.e., $Pr[Q_i \overset{R}{\leftarrow} \mathcal{QF}(1^k) : A(n, Q_i, 1^k) = 1]$ is equal to $Pr[Q_j \overset{R}{\leftarrow} \mathcal{QF}(1^k) : A(n, Q_j, 1^k) = 1]$ where A is a distinguishing server, $\mathcal{QF}$ is the query generating function, and Pr is the probability distribution function then the mutual information between them is $I(Q_i, Q_j)=0$. This implies that the queries are independent of privacy or the PIR queries are exhibiting perfect privacy.

Definition 3 (Perfect Privacy Single Database PIR (perfectPIR)) It is a 5-tuple ($\mathcal{U}$, $\mathcal{S}$, $\mathcal{QF}$, $\mathcal{RC}$, $\mathcal{IE}$) protocol where $\mathcal{U}$ is the customer, $\mathcal{S}$ is the service provider, $\mathcal{QF}$ is the query formulation algorithm run by $\mathcal{U}$, $\mathcal{RC}$ is the response creation algorithm run by $\mathcal{S}$, and $\mathcal{IE}$ is the interest extraction algorithm run by $\mathcal{U}$. Let n bit database $\mathcal{DB}_u^{b_v}$ be a two-dimensional matrix of u rows and v columns. For any interested block $\mathcal{DB}_i$, $i \in [u]$, of the database $\mathcal{DB}_u^{b_v}$, the user $\mathcal{U}$ generates PIR query described in Definition 2 to achieve the user privacy and sends to the database server $\mathcal{S}$ where all the queries sent over the insecure communication channel are coved under "quadratic residuosity assumption" (QRA) to achieve "data privacy." The database server $\mathcal{S}$ replies by generating block-specific response ciphertext set R_j and trapdoor bit set as communication bits for all the blocks $\mathcal{DB}_j$, $j \in [1, u]$ where all the generated ciphertexts from $\mathcal{S}$ are coved under QRA to achieve "data privacy." In turn, user $\mathcal{U}$ retrieves or reads required block $\mathcal{DB}_i$ using the block-specific response ciphertext set R_i and its corresponding trapdoor bit set.

6.3 Perfect Privacy PIR Scheme

Let the database $\mathcal{DB}_u^{b_v}$ be viewed as a two-dimensional matrix of u rows and v columns where $n=uv$. The database $\mathcal{DB}_u^{b_v}$ of size $n=uv$ is constituted by individual matrix or a block $\mathcal{DB}_i=b_1, b_2, \cdots, b_v, i \in [u]$, each of size v where b is the bit of $\mathcal{DB}_i$. Let us consider sufficiently large RSA composite modulus $N=pq$ where $p \equiv q \equiv 3 \pmod 4$. Assume that both the parties (user and server) have exchanged some prior information like the database size n, c, and l where each database block is divided into c number of l-bit groups and public key combination table (as described in Table 6.1).

At the high level design, the proposed 5-tuple protocol of Definition 3 is viewed as a way of retrieving or reading information from the service provider privately. The intended service seeker or the customer $\mathcal{U}$ wishes to retrieve some information from the intended service provider $\mathcal{S}$ privately using user-centric "public key

Table 6.1 Public key combinations for 2-Bit and 3-bit encoding where Q$\in \mathbb{Z}_N^{+1} \setminus \mathcal{S}_{QNR}$, N$\in \mathbb{Z}_N^{+1} \setminus \mathcal{S}_{QR}$

	f_1f_2	f_1f_2	f_1f_2	f_1f_2	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$	$f_1f_2f_3$
Input	0 0	0 1	1 0	1 1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
Public key combinations	2 2	2 1	1 2	1 1	2 2 2	2 2 1	2 1 2	2 1 1	1 2 2	1 2 1	1 1 2	1 1 1
Output property	QQ	QN	NQ	NN	QQQ	QQN	QNQ	QNN	NQQ	NQN	NNQ	NNN

Conventionally, in the second row, $2 \Leftrightarrow \mathcal{PK}_2$ and $1 \Leftrightarrow \mathcal{PK}_1$

cryptography." In order to achieve private retrieval or private reading to read the block from the server, $\mathcal{U}$ generates perfect privacy supported PIR query Q (i.e., query is selected as described in Definition 2) using the *initialization* or query formulation algorithm $\mathcal{QF}$ and sends to $\mathcal{S}$. The service provider $\mathcal{S}$ generates and sends back the response by involving all the database blocks using *reply* or response creation algorithm $\mathcal{RC}$. Finally, the customer $\mathcal{U}$ retrieves the required block privately using *reading* or interest extraction algorithm $\mathcal{IE}$.

***l*-Bit Input v/s *l*-Output Property Combination:** The public key combination selection for a particular bit or a group of bits is described as follows. Let us consider an encoding function $f : (b \xleftarrow{R} \{0, 1\}, x \xleftarrow{R} \mathbb{Z}_N^*, \mathcal{PK} \xleftarrow{R} \mathbb{Z}_N^{+1}) \rightarrow (z \in \mathbb{Z}_N^{+1})$ using the encoding bit b, random input x, and public key $\mathcal{PK}$ as

$$f(b, x, \mathcal{PK}) = \begin{cases} (x^2 \cdot \mathcal{PK} \mid \mathcal{PK} \xleftarrow{R} \mathcal{S}_{QNR}) \equiv (z \in \mathcal{S}_{QNR}) \pmod N & \text{ifb} = 1 \\ (x^2 \cdot \mathcal{PK} \mid \mathcal{PK} \xleftarrow{R} \mathcal{S}_{QR}) \equiv (z \in \mathcal{S}_{QR}) \pmod N & \text{ifb} = 0 \end{cases} \tag{6.1}$$

If the encoding bit b=1, then the public key $\mathcal{PK}$ should always be selected from $\mathcal{S}_{QNR}$ so that the output ciphertext z always resides in $\mathcal{S}_{QNR}$. Similarly, if the encoding bit b=0, then the public key $\mathcal{PK}$ should always be selected from $\mathcal{S}_{QR}$ so that the output ciphertext z always resides in $\mathcal{S}_{QR}$. If there are l-bit input functions $f_1, \cdots, f_l$ producing l output ciphertexts and each function drawn from (6.1) encodes one bit, then l public key combinations are to be used to encode l-bit input. For instance, for 2-bit input, there are two encoding functions f_1,f_2, two public keys $\mathcal{PK}_1 \xleftarrow{R} \mathcal{S}_{QNR}$,$\mathcal{PK}_2 \xleftarrow{R} \mathcal{S}_{QR}$, and four public key combinations, namely, (($\mathcal{PK}_1$,$\mathcal{PK}_1$),($\mathcal{PK}_1$,$\mathcal{PK}_2$),($\mathcal{PK}_2$,$\mathcal{PK}_1$),($\mathcal{PK}_2$,$\mathcal{PK}_2$)) as shown in Table 6.1. Similarly, for 3-bit input, there are three encoding functions f_1,f_2,f_2, two public keys $\mathcal{PK}_1$,$\mathcal{PK}_2$, and eight public key combinations, namely, (($\mathcal{PK}_1$,$\mathcal{PK}_1$,$\mathcal{PK}_1$),($\mathcal{PK}_1$, $\mathcal{PK}_1$, $\mathcal{PK}_2$),($\mathcal{PK}_1$,$\mathcal{PK}_2$, $\mathcal{PK}_1$),($\mathcal{PK}_1$,$\mathcal{PK}_2$,$\mathcal{PK}_2$), ($\mathcal{PK}_2$, $\mathcal{PK}_1$,$\mathcal{PK}_1$), ($\mathcal{PK}_2$,$\mathcal{PK}_1$,$\mathcal{PK}_2$), ($\mathcal{PK}_2$,$\mathcal{PK}_2$,$\mathcal{PK}_1$),($\mathcal{PK}_2$, $\mathcal{PK}_2$, and $\mathcal{PK}_2$)) as shown in Table 6.1. If the input bit is 0, then always $\mathcal{PK}_1$ is selected; otherwise $\mathcal{PK}_2$ is selected in the encoding function. Clearly, in order to get unique l ciphertext output quadratic residuosity property combinations (as shown in Table 6.1), public key $\mathcal{PK}_1$ is selected if the input bit is

1 otherwise public key $\mathcal{PK}_2$ is selected during encoding process using the encoding function f.

Bit Group Encoding: Let us view the database block $\mathcal{DB}_i$=$b_1, b_2, \cdots, b_v, i \in [u]$ as a set of l-bit groups $\{G_1 = (b_1, \cdots, b_l), G_2 = (b_{l+1}, \cdots, b_{2l}), G_3 = (b_{2l+1}, \cdots, b_{3l}), \cdots, G_\sigma = (b_{v-l+1}, \cdots, b_v)\}$ for some v=lc where $c > 0$ is an integer constant and $\sigma \in [1, c]$. In order to accomplish PIR operation on a block $\mathcal{DB}_i$, the PIR encoding function for $l \in \mathbb{N}$ bit group G_σ, $(y_1,\cdots,y_l) \xleftarrow{R} \mathbb{Z}_N^*$ bit input and public keys $\mathcal{PK}_1 \xleftarrow{R} \mathcal{S}_{QNR}$,$\mathcal{PK}_2 \xleftarrow{R} \mathcal{S}_{QR}$ is $\mathcal{E} : (G_\sigma, N, y_1, \cdots, y_l, \mathcal{PK}_1, \mathcal{PK}_2) \rightarrow (\alpha_1, \cdots, \alpha_l)$ where $\alpha_1, \cdots, \alpha_l$ are the corresponding ciphertext outputs. The detailed description of the encoding function $\mathcal{E}$ is as follows.

$$\mathcal{E}_\sigma(G_\sigma, N, y_1, \cdots, y_l, \mathcal{PK}_1, \mathcal{PK}_2) = \begin{cases} f_1 = [(y_1)^2 \cdot \mathcal{PK}_j \equiv \alpha_1 \pmod N)] \\ \cdot = [\ \cdot \quad \cdot \quad \equiv \quad \cdot \quad] \\ \cdot = [\ \cdot \quad \cdot \quad \equiv \quad \cdot \quad] \\ f_l = [(y_l)^2 \cdot \mathcal{PK}_{j'} \equiv \alpha_l \pmod N)] \end{cases} \tag{6.2}$$

where $j, j' \in [2]$ and each f which encodes one bit of G_σ in (6.2) is drawn from (6.1).

Connecting Two Encoding Functions Using [10]: For any two consecutive PIR encoding functions $\mathcal{E}_\sigma$ and $\mathcal{E}_{\sigma+1}$ of (6.2) where $1 \leq \sigma \leq (c-1)$, the connecting function $\mathcal{C}$ is described as follows. Let us consider $\mathcal{E}_\sigma : (G_\sigma, N, y_1,\cdots, y_l, \mathcal{PK}_1, \mathcal{PK}_2) \rightarrow \{\alpha_1, \alpha_2,\cdots, \alpha_l\}$ and $\mathcal{E}_{\sigma+1}$:$(G_{\sigma+1}, N, \alpha_1,\cdots, \alpha_l, \mathcal{PK}_1, \mathcal{PK}_2) \rightarrow \{\alpha_1^{'}, \alpha_2^{'}, \cdots, \alpha_l^{'}\}$ then the connecting function $\mathcal{C} : (\mathcal{E}_\sigma, \mathcal{E}_{\sigma+1}) \rightarrow (\{\alpha_1^{'}, \alpha_2^{'}, \cdots, \alpha_l^{'}\}, \{t_1, t_2, \cdots, t_l\})$ where each *trapdoor bit* t_i, $i \in [l]$ generated from the modified trapdoor function $\mathcal{MT}$ of (6.3) and is equivalent to "hx" value of the trapdoor function $\mathcal{T}$ described in Definition 1. Each connecting function $\mathcal{C}$ in turn connects to the next connecting function.

$$\mathcal{C}(\mathcal{E}_\sigma, \mathcal{E}_{\sigma+1}) = \begin{cases} \text{Apply } \mathcal{E}_\sigma \text{ first} \\ \text{then} \\ \mathcal{E}_{\sigma+1} = \begin{cases} (\mathcal{MT}(\alpha_1))^2 \cdot \mathcal{PK}_j \equiv \alpha_1^{'} \pmod N) \\ \cdot \quad \cdot \quad \equiv \quad \cdot \\ \cdot \quad \cdot \quad \equiv \quad \cdot \\ (\mathcal{MT}(\alpha_l))^2 \cdot \mathcal{PK}_{j'} \equiv \alpha_l^{'} \pmod N) \end{cases} \end{cases} \tag{6.3}$$

Note that only $\mathcal{E}_1$ selects the input $y_1,\cdots,y_l$ from $\mathbb{Z}_N^*$, and all other $\mathcal{E}_i$, $i \in [2, c]$, select $\alpha_1,\cdots,\alpha_l$ from $\mathbb{Z}_N^{+1}$.

6.3.1 Generic l-Bit Perfect Privacy PIR Scheme

By combining above modules, we have finally constructed a perfect privacy preserving single database PIR as follows. All the below described PIR algorithms are taken from Definition 3.

- **Initializing** ($\mathcal{QF}$): User $\mathcal{U}$ sends a block independent single PIR query Q=(N, $\mathcal{PK}_1$, $\mathcal{PK}_2$, y_1,$\cdots$, y_l) to the server where $\mathcal{PK}_1 \overset{R}{\leftarrow} \mathcal{S}_{QR}$, $\mathcal{PK}_2 \overset{R}{\leftarrow} \mathcal{S}_{QNR}$ and y_1,$\cdots$, $y_l \overset{R}{\leftarrow} \mathbb{Z}_N^*$.
- **Reply** ($\mathcal{RC}$): Server generates the block-specific response R_i, $i \in [1, u]$, as follows. As a result of response, each block $\mathcal{DB}_i$ generates two ciphertexts and trapdoor bit set as

$$\begin{aligned} cl\text{Block PIR encryption} &= \mathcal{E}_i(G_i, N, \mathcal{MT}(\mathcal{E}_{i-1}), \mathcal{PK}_1, \mathcal{PK}_2) \\ &= ((\beta_i^{\alpha_l} = (\alpha_1, \cdots, \alpha_l)), (\rho_i^{t_{l(c-1)}} = (t_1, \cdots, t_{l(c-1)}))) \\ &= R_i \end{aligned} \tag{6.4}$$

  where $i \in [c, 2]$, $\beta_i^{\alpha_l}$ is l number of ciphertexts generated at the block i, $\rho_i^{t_{l(c-1)}}$ is $l(c-1)$ number of trapdoor bits generated at the block i, $\mathcal{E}_1(\mathcal{PK}_j, \mathcal{PK}_{j'}, y_1, \cdots, y_l, G_1)$. Note that any two consecutive PIR encoding functions $\mathcal{E}_\sigma$ and $\mathcal{E}_{\sigma-1}$, $c \geq \sigma \geq 2$ described as $\mathcal{E}_\sigma(G_\sigma, N, \mathcal{MT}(\mathcal{E}_{\sigma-1}), \mathcal{PK}_1, \mathcal{PK}_2)$ in (6.4) is equivalent to the connecting function $C(\mathcal{E}_{\sigma'}, \mathcal{E}_{\sigma'+1})$, $1 \leq \sigma' \leq (c-1)$. The overall response from all the blocks of $\mathcal{DB}_u^{b_v}$ would be R=$R_1||R_2||\cdots||R_u$. The response R is sent back to the user.
- **Reading** ($\mathcal{IE}$): By using the block-specific response R_i=($\beta_i^{\alpha_l}$, $\rho_i^{t_{l(c-1)}}$), the user privately reads the required bits of the interested block $\mathcal{DB}_i$ as follows.

$$\mathcal{D}(p, q, \mathcal{E}_i(\mathcal{PK}_1, \mathcal{PK}_2, \mathcal{MT}^{-1}(\mathcal{D}(p, q, \mathcal{E}_{i+1})))) = (b_1, b_2, \cdots, b_v) = \mathcal{DB}_i$$

  or

$$\mathcal{D}(p, q, \beta_i^{\alpha_l}, \rho_i^{t_{l(c-1)}}) = (b_1, b_2, \cdots, b_v) = \mathcal{DB}_i \tag{6.5}$$

  where $i \in [1, c-1]$, $\mathcal{E}_c(\mathcal{PK}_j, \mathcal{PK}_{j'}, \alpha_1, \cdots, \alpha_l, t_{l(c-1)})$.

Theorem 1 *If any two randomly selected PIR queries are independent to each other, then they exhibit perfect privacy in PIR environment. In other words, for all quadratic residuosity-based perfect privacy PIR protocols, the probability distributions of any two randomly selected queries are always equal and independent to each other, and hence mutual information between those two queries is always zero.*

Proof (Sketch) Consider any PIR query Q=(N, $\mathcal{PK}_1$, $\mathcal{PK}_2$, y_1,$\cdots$, y_l) constructed by the user in $\mathcal{QF}$ algorithm. Note that the domains of each element are $\mathcal{PK}_1 \overset{R}{\leftarrow} \mathcal{S}_{QR}$, $\mathcal{PK}_2 \overset{R}{\leftarrow} \mathcal{S}_{QNR}$ and y_1,$\cdots$, $y_l \overset{R}{\leftarrow} \mathbb{Z}_N^*$. Also consider y_1',$\cdots$, $y_l' \overset{R}{\leftarrow} \mathbb{Z}_N^*$. Since the domain of the query input is always $\mathbb{Z}_N^*$ or the query input is always

independent of the block index, it is intuitive that $Pr[Q_i = (N, \mathcal{PK}_1, \mathcal{PK}_2, y_1, \cdots, y_l) \overset{R}{\leftarrow} Q\mathcal{F}(1^k) : A(n, Q_i, 1^k) = 1]$ is equal to $Pr[Q_j = (N, \mathcal{PK}_1, \mathcal{PK}_2, y_1^{'}, \cdots, y_l^{'}) \overset{R}{\leftarrow} Q\mathcal{F}(1^k) : A(n, Q_i, 1^k) = 1]$. Therefore, the randomly selected queries $X = Q_i$ and $Y = Q_j$ or random variables X and $Y$are independent to each other. Intuitively, $Pr(XY)= Pr(X, Y)= Pr(X) \cdot Pr(Y)$ provided $Pr(Y) >0$. The respective conditional distribution of X and Y and the mutual informations are

$$\boldsymbol{Pr}(\boldsymbol{X} \mid \boldsymbol{Y}) = \frac{Pr(XY)}{Pr(Y)} = \frac{Pr(X) \cdot Pr(Y)}{Pr(Y)} = Pr(X) \tag{6.6}$$

$$\begin{aligned} \boldsymbol{I}(\boldsymbol{X}, \boldsymbol{Y}) = \sum_X \sum_Y Pr(X, Y)\, log\, \frac{Pr(X) \cdot Pr(Y)}{Pr(X) \cdot Pr(Y)} &= \sum_X \sum_Y Pr(X, Y)\, log\, 1 \\ &= \sum_X \sum_Y Pr(X, Y) \cdot 0 = 0 \end{aligned} \tag{6.7}$$

Intuitively, all the PIR queries are mutually exclusive. This implies user privacy is independent of its query input. Therefore, all the PIR queries exhibit perfect privacy, i.e., the server gains no knowledge about the user privacy or block that the user wishes to retrieve by the query analysis using his unlimited computing power.

Theorem 2 *For all perfectPIR protocol, there exists a communication cost $o(n)$ which is always less than the trivial database download cost $O(n)$.*

Proof (Sketch) After each PIR encoding function $\mathcal{E}_i$, $i \in [1, c]$, the modified trapdoor function $\mathcal{MT}$ generates l number of trapdoor bits. Since there are c number of l bit groups present or equivalently lc number of intermediate ciphertexts generated in a block $\mathcal{DB}_j$, $j \in [u]$, there are exactly $l(c-1)$ or $(v-l)$ number of trapdoor bits generated from each block. In total, there are $ul(c-1)$ or $u(v-l)$ number of trapdoor bits generated from the entire database. Clearly, $ul(c-1)$ or $u(v-l)$ is always less than the database size uv. Therefore the communication cost w.r.t the database size is always $o(n)$ which is clearly an acceptable communication cost for "perfect privacy" in PIR environment.

Performance: User generates $k(3+l)$ bit length PIR query Q and sends it to the server. Server generates $l(k+(c-1))$ number of communication bits from each block and hence $u[l(k+(c-1))]$ number of bits from the entire database and sends back this communication bits to the user. Server performs $2ulc$ number of modular multiplications during PIR invocation, and user performs only $2lc$ number of modular multiplications during block retrieval.

6.4 Conclusion with Open Problems

We have successfully constructed a new perfect privacy-preserving information retrieval protocol with $o(n)$ communication cost. The proposed scheme successfully adopts quadratic residuosity-based public key cryptography as the underlying

primitive. In the future, it is essential to reduce the communication cost so that all including the bandwidth-limited applications can adopt the scheme. Therefore, constructing a perfect privacy PIR with the efficient communication cost is still an open problem.

References

1. Beimel, A., Ishai, Y.: Information-theoretic private information retrieval: a unified construction. In: Proceedings of 28th ICALP, pp. 912–926. Springer, Berlin (2001)
2. Beimel, A., Stahl, Y.: Robust information-theoretic private information retrieval. J. Cryptol. **20**(3), 295–321 (2007)
3. Benny, C., Niv, G., Moni, N.: Private information retrieval by keywords. Cryptology ePrint Archive, Report 1998/003 (1998). http://eprint.iacr.org/1998/003
4. Cachin, C., Micali, S., Stadler, M.: Computationally private information retrieval with polylogarithmic communication. In: Proceeedings of 17th Theory and Application of Cryptographic Techniques, EUROCRYPT'99, pp. 402–414. Springer, Berlin (1999)
5. Chang, Y.C.: Single Database Private Information Retrieval with Logarithmic Communication, pp. 50–61. Springer, Berlin (2004)
6. Chor, B., Gilboa, N.: Computationally private information retrieval (extended abstract). In: Proceedings of 29th STOC, pp. 304–313. ACM, New York (1997)
7. Chor, B., Goldreich, O., Kushilevitz, E., Sudan, M.: Private information retrieval. In: Proceedings of the 36th FOCS, pp. 41–50. IEEE Computer Society, Washington (1995)
8. Chor, B., Kushilevitz, E., Goldreich, O., Sudan, M.: Private information retrieval. J. ACM **45**(6), 965–981 (1998)
9. Di Crescenzo, G., Malkin, T., Ostrovsky, R.: Single Database Private Information Retrieval Implies Oblivious Transfer, pp. 122–138. Springer, Berlin (2000)
10. Freeman, D.M., Goldreich, O., Kiltz, E., Rosen, A., Segev, G.: More constructions of lossy and correlation-secure trapdoor functions. Cryptology ePrint Archive, Report 2009/590 (2009). http://eprint.iacr.org/2009/590
11. Gentry, C., Ramzan, Z.: Single-database private information retrieval with constant communication rate. In: Proceedings of 32nd ICALP, pp. 803–815. Springer, Berlin (02005)
12. Gertner, Y., Goldwasser, S., Malkin, T.: A random server model for private information retrieval or how to achieve information theoretic pir avoiding database replication. In: Proceedings of 2nd RANDOM, pp. 200–217. Springer, Berlin (1998)
13. Gertner, Y., Ishai, Y., Kushilevitz, E., Malkin, T.: Protecting data privacy in private information retrieval schemes. In: Proceedings of 13th STOC, pp. 151–160. ACM, New York (1998)
14. Groth, J., Kiayias, A., Lipmaa, H.: Multi-query computationally-private information retrieval with constant communication rate. In: Proceedings of 13th PKC, pp. 107–123. Springer, Berlin (2010)
15. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Cryptography from anonymity. In: Proceedings of 47th FOCS, pp. 239–248. IEEE Computer Society, Washington (2006)
16. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: single database, computationally-private information retrieval. In: Proceedings of 38th FOCS, pp. 364–. IEEE Computer Society, Washington (1997)
17. Kushilevitz, E., Ostrovsky, R.: One-way trapdoor permutations are sufficient for non-trivial single-server private information retrieval. In: Proceedings of 19th Theory and Application of Cryptographic Techniques, EUROCRYPT'00, pp. 104–121. Springer, Berlin (2000)
18. Lipmaa, H.: An oblivious transfer protocol with log-squared communication. In: Proceedings of 8th ISC, pp. 314–328. Springer, Berlin (2005)

19. Melchor, C.A., Gaborit, P.: A lattice-based computationally-efficient private information retrieval protocol (2007)
20. Naor, M., Pinkas, B.: Oblivious transfer and polynomial evaluation. In: Proceedings of 31st STOC, pp. 245–254. ACM, New York (1999)
21. Rabin, M.O.: How to exchange secrets with oblivious transfer. Harvard University Technical Report (2005)
22. Shah, N.B., Rashmi, K.V., Ramchandran, K.: One extra bit of download ensures perfectly private information retrieval. In: IEEE International Symposium on Information Theory, pp. 856–860 (2014)
23. Toledo, R.R., Danezis, G., Goldberg, I.: Lower-cost epsilon-private information retrieval. CoRR abs/1604.00223 (2016)
24. Trostle, J., Parrish, A.: Efficient computationally private information retrieval from anonymity or trapdoor groups. In: Proceedings of 13th ISC, pp. 114–128. Springer, Berlin (2011)

Chapter 7
Intelligent Access Control: A Self-Adaptable Trust-Based Access Control (SATBAC) Framework Using Game Theory Strategy

G. S. Thejas, T. C. Pramod, S. S. Iyengar, and N. R. Sunitha

Abstract Access control mechanisms are widely used to secure the computer and network resources from attacks like leakage of sensitive information and denial of services. It differentiates the honest and dishonest users by limiting their access to the information and resources. Traditional access control models are not efficient in the open network environment where adaptability is required due to the dynamic nature of user, network, and the service provider. In this paper, we propose a modified dynamic framework which is the self-adaptable trust-based access control (SATBAC) with the help of several access request evaluating techniques like opinion and reputation, behavior and history, and credential and location. We then analyze the trust values obtained from the integrated access control evaluation techniques using game theory-based reward-punishment strategy to decide whether to grant access or not. Comparison and implementation of the proposed access control mechanism for e-commerce service are done to highlight the effectiveness of the proposed framework.

Keywords Adaptive access control · Behavior and history · Game theory-based reward-punishment strategy · Opinion and reputation · Security · Trust management

G. S. Thejas (✉) · S. S. Iyengar
School of Computing and Information Science, Florida International University, Miami, FL, USA
e-mail: tgs001@fiu.edu; iyengar@cs.fiu.edu

T. C. Pramod · N. R. Sunitha
Department of Computer Science, Siddaganga Institute of Technology, Tumkur, Karnataka, India
e-mail: pramodtc@sit.ac.in; nrsunitha@sit.ac.in

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_7

7.1 Introduction

Access control mechanism plays an important role in all computing environments. The objective of the access control mechanism is to authorize the user who makes the request of access to information, service, and other resources. It was first introduced in the early 1970s by Lampson as access control matrix model [13]. The matrix defines the access rights among the subject (rows) and object (columns), and the authorization decisions are made based on the matrix entries of row column records. Later, many access control models were proposed by the researchers depending on the different access control requirements, and those models are attribute-based access control (ABAC), discretionary access control (DAC), history-based access control (HBAC), identity-based access control (IBAC), mandatory access control (MAC), organization-based access control (OrBAC), role-based access control (RBAC), rule-based access control (RAC), responsibility-based access control, context-based access control (CBAC), graph-based access control (GBAC), lattice-based access control (LBAC), rule set-based access control (RSBAC), capability-based security, location-based authentication, and risk-based authentication [1, 2, 6, 17].

Now in the era of autonomous computing, the expectation is about managing the tasks by automatically adapting to the environment needs. This enables the security mechanism to be intelligent and adaptive in all the computing environments. To enforce the need of adaptability in access control mechanism, the trust management is one of the key factors by which till today many researchers are proposing adaptable access control models based on trust builders [16, 17]. Nowadays, game theory strategies are gaining more popularity in the area of security. Game theory strategies can be used to analyze and improve the access control decisions at run time for making access decision.

In open network environment like e-commerce, there exists a requirement to have good and efficient access controlling mechanism for the purpose of online transactions which is an important issue. In an online business environment, either user may be a malicious attacker or an honest service requester. In this situation, evaluating user access request is a crucial task where we need self-adaptable access control mechanism which evaluates the user with several factors, builds trust models, and then analyzes the access request efficiently. With this, evaluating access requesting users at run time and making appropriate decision based on several factors of users are an important issue because an experienced user or an attacker can analyze the static authentication schemes in a timely manner and prepare to attack by understanding the model, flaws, and possibilities of faking the entire system.

Our Contribution We address the above said issues by introducing a new modified self-adaptable framework SATBAC. We are using the game theory-based reward-punishment strategy, trust negotiation policy, and access control policy to analyze and make access granting or denying decision at run time. We are considering factors like opinion and reputation, behavior and history, and credential and location to evaluate the user request for obtaining trust rank. In the framework, we suggest appropriate authentication schemes dynamically from the pool of possibilities at

run time based on the analysis which is a novel access control mechanism. Our framework punishes the experienced user (existing user) or an attacker who attempts to fake the entire system.

The rest of the paper is organized as follows: in Sect. 7.2 we review the related work on access control mechanisms, in Sect. 7.3 our proposed SATBAC framework is discussed, in Sect. 7.4 we demonstrate our framework with a case study of e-commerce online shopping example and we present the experimental results, in Sect. 7.5 performance analysis is presented, and in Sect. 7.6 we conclude our research discussion and provide areas of future directions in continuation of this research work.

7.2 Related Work

Since the last decade, many researchers have contributed toward designing good access control models and mechanism to address complex issues related to access control in open systems and networked systems among which trust-based access control models and framework have become very popular. In [21] author Marianne Winslett introduces the concept of building trust in the replacement of paper-based trust building to an electronic-based trust building and theoretically explains the trust management in some real-time environment like hospitals and employees trust management to exchange information. Also, the article lists and describes several issues for the adaptation of trust management approach to establish trust between the participants. Adaptive trust negotiation and access control model was proposed to address malicious attack leading to denial of service or leakage of sensitive information [16]. In [7] authors concentrate on the risk models in the trust-based access control system on the basis of economic theory and demonstrate the implementation on peer-to-peer collaborative spam detection application. In [18], authors attempt to identify access control requirements in the peer-to-peer file sharing environment, and they extend the discretionary access control model with trust-based model to address the requirements and to classify known and unknown users. In [19], authors propose the integration of the trust management component for the existing access control mechanism in the virtual learning application for secure exchange of information with belief of the knowledge. The article [6] outlines the need for providing support to access control in open environment where the identities of the involved parties may be unknown and enforces the use of selective encryption. In [15], authors propose an automated trust negotiation framework supporting the adaptive policies where mutual trust is established among the two parties by gradually exchanging credentials. In [5], authors provide the theoretical analysis for their model, the adaptive automated trust negotiation system, which adjusts the trust negotiation strategies and polices based on the needs. The article [8] explains about the security requirement that must be met and its need for designing access control in web services. In [14], authors propose an adaptive and attribute-based trust model for service level agreement guarantee in a cloud environment and explain the importance of trust attribute model to enforce service level agreement

for evaluating the competence of cloud service based on multiple trust attributes. In [12, 22] authors analyze the traditional access control model using the game theory approach and then propose a proactive access control mechanism by considering constraints and factors like frequency of access, network condition level, and trust of neighboring nodes. With these they provide the probability of permitting access and probability of honesty. In [3], authors explain the importance of trust-based access control models in the future direction of current trending area Internet of Things (IoT) and propose trust-aware access control system for building trust among the wireless devices to exchange information securely ensuring at the same time reliable communication between the trusted devices. In [20], authors propose the game theory-based trust measurement model for social network by considering some factors like service reliability by service nodes, feedback effectiveness by feedback nodes, and recommendation credibility to calculate trust level and provide a solution to the free riding problem.

7.3 Proposed SATBAC Framework Design

Figure 7.1 depicts the proposed SATBAC framework in which access requesting users and service providers are the participants. Access control processing block is where the entire evaluation and analysis of access control take place to make appropriate decisions and to suggest authentication schemes at run time. This block consists of access request trust value calculation unit, history database, access control policy unit, and trust negotiation unit. Sections 7.3.1 and 7.3.2 explain in detail about all the components of the framework.

7.3.1 Access Request Trust Value Calculation (ARTVC) Unit

ARTVC unit contains schemes to calculate access requesting users trust value which is in the range of 0 to 1. The higher the calculated trust value, the higher is the trust.

7.3.1.1 Based on Opinion and Reputation

In this scheme we obtain the opinion and reputation with respect to the access requesting user dynamically in the following ways: known account holder (he/she can be a friend or referrer), from third party, and asking the user to log on to his/her already existing social network account (Facebook, Twitter, etc.). From these sources, the proposed framework adapts itself to obtain opinion and reputation value. Let it be denoted as ORV_{vi} the value of ith party providing opinion and reputation (values must be in the range 0 to 1), C_i is the probability of i being correct, then opinion and reputation trust value is represented by ORC_i. We obtain

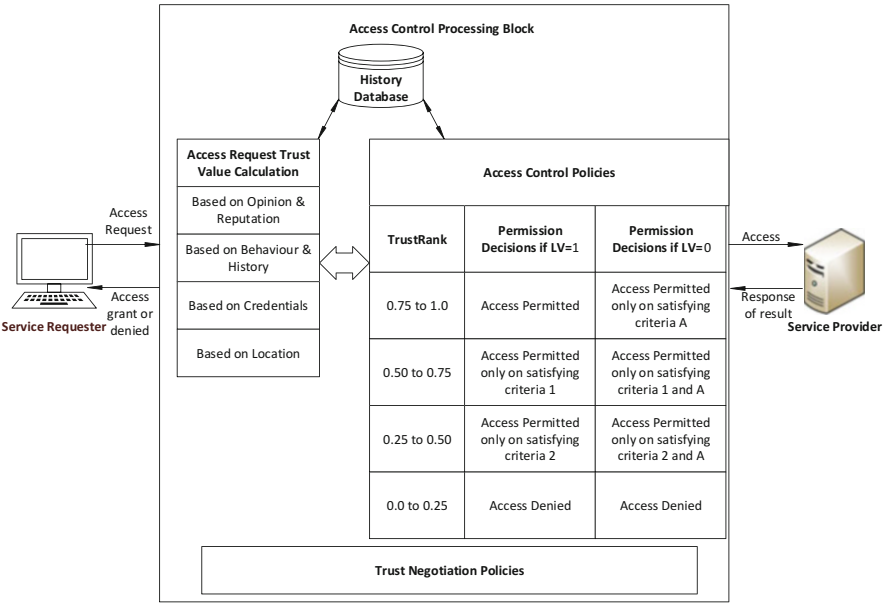


Fig. 7.1 Proposed SATBAC framework design

trust value of opinion and reputation based on Eq. (7.1), where n is the total number of parties contributing opinion and reputation value:

$$ORC_i = \frac{\sum_{i=0}^{n} OR_{vi}}{n} \tag{7.1}$$

7.3.1.2 Based on Behavior and History

In this scheme, access requesting users past interactions and past sessions trust rank (discussed in Sect. 7.3.2) permission decision values are retrieved for calculation of trust value based on behavior and history. All these details will be tracked in our history database. Let i be the number of instances in terms of positive behavior, j be the number of instances in terms of negative behavior, n be the total number of positive behavior, m be the total number of negative behavior, C_i is the probability of i being correct, PB_{vi} be the value of each positive behavior instance, NB_{vj} be the value of each negative behavior instance, the frequency of positive or negative behavior value must range between 0 to 1, then we obtain the present behavior value $PrBC_{ij}$ by Eq. (7.2):

$$PrBC_{ij} = \frac{\sum_{i=0}^{n} PB_{vi}}{n} - \frac{\sum_{j=0}^{m} NB_{vj}}{m} \tag{7.2}$$

In the history database, the existing users past behavior value $PsBC_{ij}$ will be updated by evaluating in the same manner as said above for the previous

interactions. The actual behavior value BC_{ij} is obtained by the following equation (Eq. 7.3):

$$BC_{ij} = \frac{PrBC_{ij} + PsBC_{ij}}{2} \tag{7.3}$$

In case of new user, the following formula is considered for BC_{ij}, Eq. 7.4:

$$BC_{ij} = PrBC_{ij} \tag{7.4}$$

For an existing user, in the history database, there will be a record of past transaction in terms of past interaction with positive transaction *(PT)* and past interaction with negative transaction *(NT)*. These terms are initialized to 0 and incremented by 1 for positive transaction and decremented by 1 for negative transaction. The transaction value, T, is obtained by using Eq. (7.5):

$$T = PT + NT \tag{7.5}$$

The final behavior and transaction values constitute to the final behavior and history *(BH)* value as given in Eq. (7.6):

$$BH = \frac{BC_{ij} from Eq.(7.3) + T}{2} \tag{7.6}$$

In case of new user, the value of T, PT, and NT is considered 0, but once after successful completion of access request and transaction, the PT or NT values will be updated. So in this situation,

$$BH = BC_{ij} from Eq.(7.4) \tag{7.7}$$

7.3.1.3 Based on Credential

In this scheme, minimal credential is collected from the access requesting user at the first stage (like username and password). Later, for the given password credential security strength value is determined. This can be done by prior assignment of values to all possible credentials depending upon the credentials security strength. The possible credentials can be like passwords, onetime password (OTP), last purchased item, last transaction amount, fingerprint, retinal scan, face recognition, and voice recognition. Let each credential types be denoted as i, CD_{vi} be the value of ith credential (values must be in the range 0 to 1), C_i is the probability of i being correct, then credential trust value is represented by CDC_i. We obtain trust value of credential based on Eq. (7.8), where n is the total number of credentials used:

$$CDC_i = \frac{\sum_{i=0}^{n} CD_{vi}}{n} \tag{7.8}$$

7.3.1.4 Based on Location

In this scheme, the access requesting user is evaluated based on the location from where he/she is trying to access with in a timeout period. Based on the current trends (like news regarding terrorism area, most cyberattack area, most sensitive area, and so on), geographical areas are classified into safe zones and unsafe zones. Depending on this classification, there can be change in the access control policies which will be explained in the following section.

7.3.2 Analysis of Access Control Evaluation Based on Game Theory Approach: Reward-Punishment Strategy

Game theory is the process of modeling the strategic interaction between two or more players in a situation with some set of rules and outcomes or payoffs [9],[4],[10]. It is used to make decision scenarios among the players in the game, where a player always try to choose an action based on some strategy or rule or constraint to maximize the player payoff [11]. A game can be described by the following elements: game, players, strategy, sequence, information set, and payoff. In our SATBAC framework, we describe the game as follows:

- *Game:* Self-Adaptable Trust-Based Access Control is the game. Here we analyze the trust values obtained from the collaborative access control evaluation techniques by the use of game theory-based reward-punishment strategy.
- *Players:* Two players are the access requester and access provider/service provider.
- *Strategy:* A complete set of rules that defines possible actions for the players at each stage in the game. Reward-punishment-based strategic decision-making scenarios and set of actions are defined to enable self-adaptability of the framework in a dynamic environment. This is based on (1) TrustRank (2) access control policies which categorize the TrustRank into several categories, considering location information. Access granting decisions are made based on the level of satisfying criteria and (3) trust negotiation policies which dynamically suggest the possible ways to satisfy the criteria.
- *Sequence:* Defines the sequence for players to follow the strategy. In the dynamic environment the framework adapts itself to define the sequences for the players based on ACP (Access Control Policy) and TNP (Trust Negotiation Policy) units to follow the strategy.
- *Information set:* The information that is available at each stage of the game, it can be complete or incomplete information. The service provider may or may not know the access requesting player, and the access requester may or may not know the game strategy and the actions, because of the adaptable behavior of the framework. The known information to the service provider will be the information that is stored in the history database for all existing users. Here the

information set is incomplete, and the required information is exchanged at run time.
- *Payoff:* The reward that will be received by the players on performing some action could be access permitted or access permitted only on satisfying criteria or access permission denied.

Based on the previous analysis, we can calculate the trust rank from Eqs. (7.1), (7.6) or (7.7), and (7.8) as follows using Eq. (7.9) or (7.10), where α, β, and γ are the weights to opinion and reputation, behavior and history, and credential and location. For existing user,

$$TrustRank = \alpha.ORC_i + \beta.BH from Eq.(7.6) + \gamma.CDC_i \tag{7.9}$$

where $\alpha + \beta + \gamma = 1$.
For new user,

$$TrustRank = \alpha.ORC_i + \beta.BH from Eq.(7.7) + \gamma.CDC_i \tag{7.10}$$

where $\alpha + \beta + \gamma = 1$.

Now by using TrustRank from Eqs. 7.9 or 7.10, analysis is made by applying reward-punishment strategy of game theory for making access decisions with the combination of access control policy unit and trust negotiation policy unit. This illustrates the self-adaptability of the framework. TrustRank value thus indicates the potentiality of getting access permissions like access denied/permitted/permitted only on satisfying some criteria. In the reward-punishment strategy, rewarding is done by permitting or denying access to the access requesting user in successful and unsuccessful scenarios, respectively, based on the TrustRank categories CAT-1:(0.75-1.0) and CAT-4:(0.00-0.25), respectively. Punishing is done for those users who fall in the TrustRank categories CAT-2:(0.50-0.75) and CAT-3:(0.25 to 0.50). These rewards and punishments are also based on the location value, i.e., LV= 1 represents safe zone request and LV= 0 represents unsafe zone request.

Algorithm 1 shows the procedure of the proposed self-adaptable trust-based access control framework. In Algorithm 1, criteria A would be collecting some identity proof information like passport, PAN card or Social Security number, bank account proof, and so on from the requesting user and evaluating it with the authorized party. Criteria 1 would be collecting opinion and reputation from third party or other higher credentials that strengthens the security level. Criteria 2 would be providing even more higher credentials that strengthens the security level.

In CAT-3(0.25 to 0.50), both the cases (LV=1 or LV=0) after attaining criteria, if the new TrustRank is within in the range 0.50–0.75, TNP gives one more chance to increase the TrustRank to the range (0.75–1.0) by allowing the user to undertake additional criteria. If the additional criteria are successfully satisfied, access is granted to the user.

The framework adapts itself to make all the evaluations and decisions at run time. All these evaluation and decision-making details will be stored in our history database which we can use in future decisions.

Algorithm 1: SATBAC Mechanism

```
start procedure
Input: Access request from user
Output: Access grant or denied
Step1: ARTVC
TrustRank ⟵ Credentials + Opinion & Reputation + Behaviour & History
Step2: Implication of Reward-Punishment Strategy
                                                                  # CAT-1
if 0.75 < TrustRank <= 1.0 then
    if LV=1 then
        Access permitted
    else if LV=0 then
        User requires to undertake Criteria A
        Calculate New TrustRank
        if 0.75 < New TrustRank <= 1.0 then
            Access permitted
        else
            Access denied
        end if
    end if
                                                                  # CAT-2
else if 0.50 < TrustRank <= 0.75 then
    if LV=1 then
        User requires to undertake Criteria 1
        Calculate New TrustRank
        if 0.75 < New TrustRank <= 1.0 then
            Access permitted
        else
            Access denied
        end if
    end if
else if LV=0 then
    User requires to undertake Criteria 1 + Criteria A
    Calculate New TrustRank
    if 0.75 < New TrustRank <= 1.0 then
        Access permitted
    else
        Access denied
    end if
                                                                  # CAT-3
else if 0.25 < TrustRank <= 0.50 then
    if LV=1 then
        User requires to undertake Criteria 2
        Calculate New TrustRank
        if 0.75 < New TrustRank <= 1.0 then
            Access permitted
```

```
46:         else
47:             Access denied
48:         end if
49:     end if
50: else if LV=0 then
51:     User requires to undertake Criteria 2 + Criteria A
52:     Calculate New TrustRank
53:     if 0.75 < New TrustRank <= 1.0 then
54:         Access permitted
55:     else
56:         Access denied
57:     end if
58:                                                          # CAT-4
59: else if 0.0 < TrustRank <= 0.25 then
60:     if LV=1 then
61:         Access denied
62:     else if LV=0 then
63:         Access denied
64:     end if
65: end if
66: end procedure
```

In trust negotiation policies (TNP) unit, the TNP plays a very important role for the framework self-adaptability at run time. The TNP evaluates and suggests the negotiation criteria to provide more information in case of the user falling in any of the punishment cycle. Dynamically TNP indicates polices for exchanging opinion and reputation values, credentials (that means, what type of credentials between the user and service provider need to be exchanged to build further trust among both), and identity proofs. After this stage, again the user can be evaluated for new trust values at ARTVC unit and analyzed with new TrustRank. Only after satisfying TNP criteria policies the access requesting user can come out of the punishment cycle and get rewarded after reaching the CAT-1, otherwise access is denied. These strategies help to make further permission decision at ACP unit.

7.4 Case Study: E-Commerce Online Shopping Example

To demonstrate our SATBAC framework, we have considered a case study of e-commerce online shopping example where users try to purchase on online shopping websites. Here, the service provider is the online shopping website server. When the online shopping website server gets an access request from the user for placing an order of buying some item, the transaction has to be analyzed before granting permission to proceed with the transaction (Fig. 7.2).

In order to perform transaction, the user has to log in. In our example, we have considered two cases to log in: Case 1, the user has registered with the online shopping website server (using the credentials, the user can log in), and Case 2, log in using one of the existing social networking services. Once the log-in is

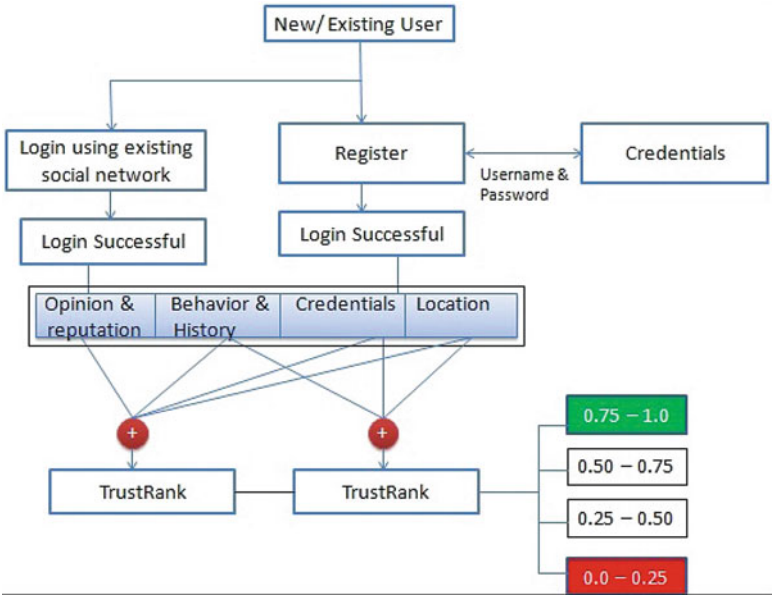


Fig. 7.2 E-commerce application

successful, our ARTVC unit analyzes the user and calculates the TrustRank. The parameters involved in obtaining the TrustRank are opinion and reputation, behavior and history, and credentials and location.

Here, the transaction permissions are granted based on reward-punishment analysis by keeping some user access request evaluation factors, access control policies, and trust negotiation policies as shown in Fig. 7.3, and Table 7.1 shows the experimental results. In Table 7.1 *UT* indicates user type, i.e., new user (*NU*) or existing user (*EU*), *LV* indicates location value, *TR* indicates TrustRank, *CAT* indicates category type as defined in the above section, *D* is obtained by Eq. 7.11, criteria are represented by C1, C2, and CA, *NTR* indicates new TrustRank, *PD* indicates permission decisions, and yes or no is abbreviated as *Y* or *N*, respectively.

$$D = Lowerboundofnext CAT - TrustRank \tag{7.11}$$

where Lower_bound of next CAT can be 0.25,0.50,0.75.

Whenever users go into the punishment cycle, then he/she is allowed to make one or two attempts to come out of the punishment cycle otherwise he/she will be rewarded as access denied. After satisfying criteria 1 and/or 2, the TrustRank value may fall in the ranges like 0.50–0.60 or 0.60–0.75 and 0.25–0.40 or 0.40–0.50, respectively. *D* is the difference between the TrustRank and the next category lower bounds. Users have to satisfy the criteria in such a way that either by providing CD_{vi} or OR_{vi} and ID proofs as specified in the TNP in order to come out of the punishment cycle.

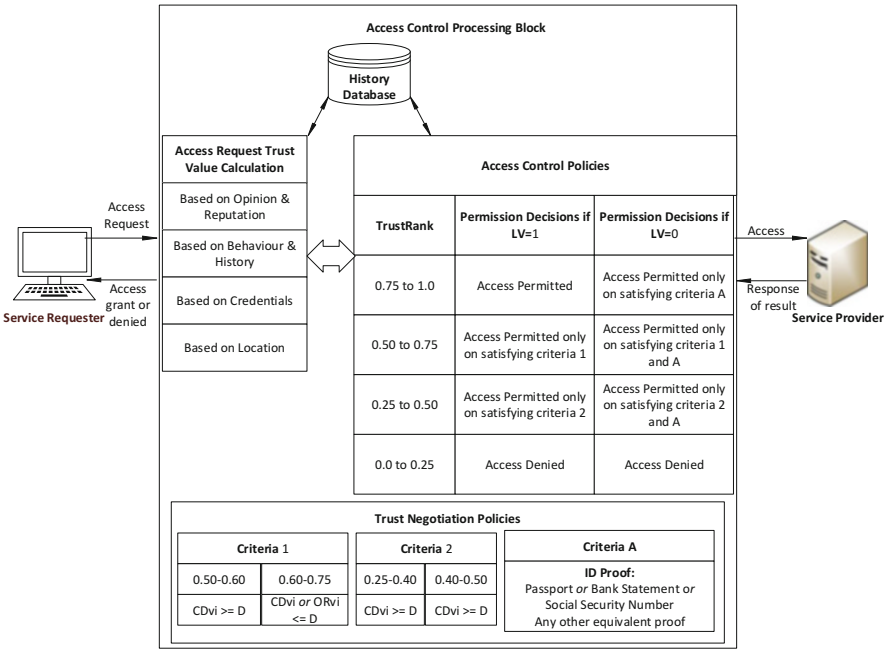


Fig. 7.3 SATBAC framework for e-commerce application

Table 7.1 Online shopping experimental results

UT	LV	TR	CAT	D	C1	C2	CA	NTR	PD
NU1	1	0.55	2	0.20	Y	–	–	0.75	Y
NU2	1	0.33	3	0.17	Y	Y	–	0.79	Y
NU3	1	0.20	4	–	–	–	–	–	N
NU4	0	0.22	4	–	–	–	–	–	N
NU5	0	0.26	3	0.24	N	Y	Y	0.66	N
NU6	0	0.66	2	0.09	Y	–	Y	0.80	Y
EU1	1	0.85	1	–	–	–	–	–	Y
EU2	1	0.73	2	0.02	Y	–	–	0.81	Y
EU2	1	0.59	2	0.16	N	–	–	0.74	N
EU3	0	0.44	3	0.06	Y	Y	Y	0.76	Y
EU4	0	0.70	2	0.05	Y	–	N	0.77	N

7.5 Performance Analysis

Table 7.2 shows the comparison of the proposed scheme with some of the existing access control schemes. *H* indicates *HIGH*, *M* indicates *MODERATE*, and *L* indicates *LOW*. The comparisons are made on three parameters: *dynamic*, *flexible*,

Table 7.2 Comparison of the proposed scheme with existing schemes

Scheme	Dynamic	Flexible	Self-recoverable
Rule based [2, 6, 17]	L	M	L
Behavior based [2, 6, 17]	M	M	M
Credential based [2, 6, 17]	L	M	M
Proposed	H	H	H

and *self-recoverable*. In the proposed scheme, TrustRank relies on multiple factors. If one factor fails to perform, we can rely on other factors to obtain the TrustRank. Also, our framework suggests appropriate authentication schemes and trust building mechanisms dynamically from the pool of possibilities at run time. Hence, the proposed scheme is *dynamic* in nature. Whereas in [2, 6, 17], access control is performed on predefined rules and behaviors. The schemes are static to those rules and behaviors. The proposed scheme supports self-adaptability and real-time access control features. Hence, the scheme is *flexible* in nature. If the TrustRank ranges in CAT-2 (0.50–0.75) or CAT-3 (0.25–0.50), the proposed framework gives a chance for the access requesting user to increase the TrustRank by satisfying some criteria and also punishes experienced users who attempts to fake the system. Hence, the scheme is *self-recoverable*.

7.6 Conclusion and Future Work

In this paper, we have presented a self-adaptable trust-based access control framework which provides dynamic security features to protect resources from the malicious attacks. The access requesting users are evaluated based on factors like opinion and reputation, behavior and history, and credentials and location along with several other factors like feedback and timeout periods of the sessions. With the help of TrustRank, access control polices, and trust negotiation polices, we have analyzed the permission decision using game theory approach and reward-punishment mechanism. These evaluations, analysis, and decisions take place at run time. We have demonstrated our framework on a case study of e-commerce online shopping example with experimental results. Our framework is a generic intelligent self-adaptable access control model which can be used for any security domain and computing environment. This work can be extended by considering many more factors to evaluate the trustworthiness of access requesting users by using intelligent approaches, defining even more tightened access control polices and trust negotiation polices, and simulating the framework for analyzing and tuning the performance.

References

1. Access Control.: Wikipedia, https://en.wikipedia.org/wiki/Access_control (2014). Accessed 03 Jan 2017
2. Ausanka-Crues, R.: Methods for access control: advances and limitations. Harvey Mudd College (2001)
3. Bernabe, J.B., Hernandez Ramos, J.L., Skarmeta Gomez, A.F.: TACIoT: multidimensional trust-aware access control system for the Internet of Things. Soft Comput. **20**(5), 1763–1779 (2015)
4. Bernasco, W., Elffers, H., van Gelder, J.-L., Rauhut, H.: The Oxford Handbook on Offender Decision Making. University of Zurich, Institute of Sociology, Switzerland (2015)
5. Chen, W., Jiang, W.: Analysis and design of an adaptive automated trust negotiation system. In: Proceedings of International Conference on Mechatronic Science, Electric Engineering and Computer, pp. 2320–2325. IEEE, Jilin (2011)
6. De Capitani di Vimercati, S., Foresti, S., Samarati, P.: Recent advances in access control. In: Gertz M., Jajodia S. (eds.). Handbook of Database Security, pp. 1–26. Springer, Berlin (2008)
7. Dimmock, N., Bacon, J., Ingram, D., Moody, K.: Risk models for trust-based access control (TBAC). In: Herrmann P., Issarny V., Shiu S. (eds.) Trust Management. iTrust 2005. Lecture Notes in Computer Science, vol 3477, pp. 364–371. Springer, Berlin (2005)
8. Esfandi, A., Sabbari, M.: Study of access control issue in web services. Int. J. Comput. Appl. **49**, 11–17 (2012)
9. Game Theory, Wikipedia. https://en.wikipedia.org/wiki/Game_theory. Accessed 03 Jan 2017
10. Gintis, H.: Game Theory Evolving: A Problem-Centered Introduction to Modeling Strategic Behavior. Princeton University Press, Princeton (2000)
11. Gintis, H.: A framework for the unification of the behavioral sciences. Behav. Brain Sci. **30**(1), 1–61 (2007)
12. Jingsha, H., Shunan, M., Bin, Z.: Analysis of trust-based access control using game theory. Int. J. Multimed. Ubiquit. Eng. **8**(4), 15–24 (2013)
13. Lampson, B. W.: Protection. In: Proceedings of Fifth Princeton Symposium of Information Science and System, Princeton University, pp. 437–443 (1971). Reprinted in Operating system review, 8, 1, January 1974, pp. 18–24
14. Li, X., Du, J.: Adaptive and attribute-based trust model for service level agreement guarantee in cloud computing. IET Inf. Secur. **7**(1), 39–50 (2013)
15. Liu, B., Lu, H., Zhao, Y.: An efficient automated trust negotiation framework supporting adaptive policies. In: Proceedings of Second International Workshop on Education Technology and Computer Science (ETCS), pp. 96–99. IEEE, Wuhan (2010)
16. Ryutov, T., Zhou, L., Neuman, C., Travis, L., Seamons, K.E.: Adaptive trust negotiation and access control. In: Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies (SACMAT '05), pp. 139–146. ACM, New York (2005)
17. Servos, D., Osborn, S.L.: Current research and open problems in attribute-based access control. ACM Comput. Surv. **49**(4) Article 65, 45 pp. (2017)
18. Tran, H., Hitchens, M., Varadharajan, V., Watters, P.: A trust based access control framework for P2P file-sharing systems. In: Proceedings of the 38th Annual Hawaii International Conference on System Sciences. IEEE, New York (2005)
19. Wang, S., Liu, Q.: Trust-based access control in virtual learning community. In: Wang W., Li Y., Duan Z., Yan L., Li H., Yang X. (eds.) Integration and Innovation Orient to E-Society Volume 2. IFIP International Federation for Information Processing, vol. 252, pp. 514–520. Springer, Boston (2007)

20. Wang, Y., Cai, Z., Yin, G., Gao, Y., Tong, X., Han, Q.: A game theory-based trust measurement model for social networks. In: Computational Social Networks. Springer International Publishing, Cham (2016)
21. Winslett, M.: An introduction to trust negotiation. In: Nixon P., Terzis S. (eds.) Trust Management. iTrust 2003. Lecture Notes in Computer Science, vol 2692, pp. 275–289. Springer, Berlin (2003)
22. Zhang, Y., He, J., Zhao, B., Huang, Z., Liu, R.: Towards more pro-active access control in computer systems and networks. Comput. Secur. **49**, 132–146 (2015)

Chapter 8
Dynamic Firewall Policy Management Framework for Private Cloud

Mahesh Nath Maddumala and Vijay Kumar

Abstract In traditional networking environment, perimeter firewalls restrict the access to resources inside organizations' private network. In cloud computing, every virtual instance has to be accessed by the external parties. Due to its unique computing environment, cloud computing requires tailor-made firewall systems. To provide the desired high-availability and low-latency system, cloud services are replicated and made available in multiple geolocations or availability zones. This often leads to violation of various compliance policies and policy conflicts. The other complex issue is to manage security policies across the distributed data centers. We address these issues and present a dynamic firewall policy management scheme that uses a centralized controller to manage all firewalls in distributed data centers of a private cloud.

Keywords Distributed firewall policies · Cloud policies · Compliance policies

8.1 Introduction to Cloud Data Centers and Their Security Issues

In traditional networking environment, perimeter firewalls restrict the access to the resources inside the private network. In cloud computing, every virtual instance has to be accessed by the external parties. Due to its unique computing environment, cloud computing requires tailor-made firewall systems. NIST [1] defines cloud computing as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." It is no doubt that the cloud computing has transformed the computing discipline into more profitable

M. N. Maddumala (✉) · V. Kumar
University of Missouri–Kansas City, Kansas City, MO, USA
e-mail: mahesh.maddumala@mail.umkc.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_8

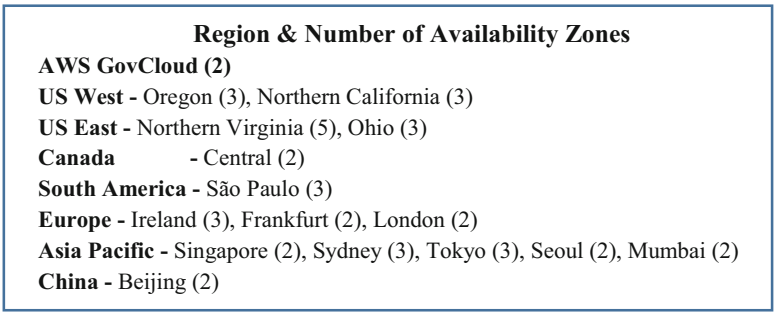


Fig. 8.1 Amazon Web Services Global Infrastructure

and efficient platform, but not without the challenges. One of the major challenges is security and privacy. According to the cloud security 2016 spotlight report by CloudPassage [2], *verifying security policies*, *visibility into infrastructure security*, and *compliance* were named as the top three cloud security challenges that cause the biggest headaches for IT security professionals.

One of the main features of the cloud computing is its high availability and performance. In order to provide low-latency and fault-tolerant services across the globe, customers prefer to replicate the data and applications and make them available at various zones across the globe. The AWS cloud [3] operates data centers in 42 availability zones within 16 geographic regions around the world as shown in Fig. 8.1. Microsoft Azure [4] is available in 34 regions around the world. Although the expansion of the cloud infrastructure to multiple regions significantly improves its availability and performance, it presents unwanted location-based compliance issues, and one such issue is internet censorship. Many countries block certain websites and services due to political and religious reasons, which poses a great challenge to the data centers to offer the services in more stringent countries. For example, Iran [5] has blocked almost 50% of the top 500 visited websites worldwide including YouTube, Facebook, Twitter, and Google Plus. China [6] governmental authorities not only block websites but also monitor the internet access of individuals. Unfortunately such restrictions present compliance challenges.

NIST [1] defines four deployment models of cloud computing—public cloud, private cloud, community cloud, and hybrid cloud. In private cloud, customers are completely responsible for securing their data, applications, platform, and infrastructure. As the cloud data centers are distributed across the globe, managing security policies and ensuring compliance and consistency across the data centers is highly challenging. There are two types of location-based compliance challenges—one is the enforcement of data residency regulations (certain sensitive information must not leave the country) and the other is the enforcement of internet censorship (certain censored information must not enter the country). The first case has been addressed by various governmental compliance policies such as Health Insurance Portability and Accountability Act (HIPAA), Payment Card Industry Data Security

Standard (PCI DSS), etc. The second case is enforced by the individual countries such as China, Iran, Saudi Arabia, etc. using firewalls and other filtering mechanisms. In this chapter, we deal with second case of location-based compliance issues.

In our framework, we address various security issues such as compliance policies, policy anomaly detection, and distribution of policies. We present our policy automation model with a centralized global policy manager which verifies security policies for consistency, compliance, and conflict resolution across all the firewalls.

8.2 Dynamic Firewall Policy Management for Private Cloud

The main objective of our framework is to automate the policy management of distributed firewalls of a private cloud using a centralized controller with a defined set of procedures and policy structure. The policy management includes dynamically updating the policies, ensuring the compliance and consistency of policies, resolving the policy conflicts, and providing the strong security with cooperative firewalls.

In our scheme, we choose the centralized approach over the distributed approach to manage the firewall policies across the data centers. In a distributed topology, firewalls are connected to each other forming a fully connected network so that any firewall can inform its policy changes to all other firewalls in the network. This has two disadvantages: (a) a compromised firewall could compromise all other firewalls leading to a serious security violations and (b) the overhead of information exchange among these firewalls. In a centralized topology, all the firewalls are connected to a central node (a controller). In this setup, unlike distributed topology, only the controller can propagate the policy changes to all firewalls in the network.

Every data center deploys two types of firewalls: one is a perimeter firewall which secures the entire network of the data center and another one is an internal firewall which secures individual virtual machines as shown in Fig. 8.2. The security policies

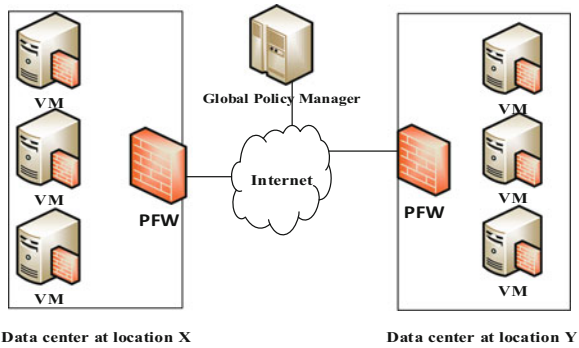


Fig. 8.2 Firewall setup in distributed data centers

configured at the perimeter firewall are common security policies which apply to all virtual machines within the data center whereas virtual machine firewall policies are specific to the individual virtual machines. For example, a virtual machine running webserver may configure webserver-based security policies in its firewall, whereas a virtual machine running email servers may configure only email-related security policies in its firewall.

8.2.1 Policies

We classify the firewall policies of a private cloud into three categories: (a) location-based compliance policies, (b) perimeter policies, and (c) group policies.

8.2.1.1 Location-Based Compliance Policies

During the process of the replicating data and applications to another data center, security policies of virtual machine firewall and data center perimeter firewall also have to be replicated. If the replicated data center is in an internet-restricted country, then that data center has to adhere to the country-specific internet policies. These policies would be configured at the perimeter firewall.

8.2.1.2 Perimeter Policies

These are the general policies configured at the perimeter firewalls to prevent common attacks and unwanted traffic, for example, blocking the access to unused ports, denying unsupported protocols, and rejecting the traffic from prohibited IP addresses.

8.2.1.3 Group Policies

The service-based security policies can be grouped as group policies. For example, VM hosting web services should have web-based security policies in its firewall, and these policies can be applied to all the VMs that are running web services. Also multiple services such as web, email, and databases can be hosted on the same virtual machine. In this case, VM hosting multiple services should have respective service based group in its firewall. These policies are configured at virtual machine firewall.

8.2.2 Dynamic Policies

The predefined security policies can only help to prevent the known attacks. To prevent the emerging advanced attacks, firewalls should be capable of creating

dynamic policies by analyzing the network packets, by creating dynamic IP blacklists, etc. Similar work has been addressed in [7, 8].

8.2.2.1 Dynamic IP Address Blacklist

IP blacklist is a list of blocked IP addresses that are suspected of sending malicious packets. Security administrators maintain a static IP blacklist to block the traffic from certain IP addresses, and the list would be updated by importing latest blacklists from various security sources, often vendors. However there is a problem with the static blacklists. Firewalls could not prevent the attacks coming from the IP addresses which are not listed in the blacklists. We developed an algorithm to generate IP blacklists dynamically by identifying the frequency of the attacks originated by an IP address.

Figure 8.3 illustrates the entire process of generating dynamic IP blacklist. Initially when packet arrives, source IP address will be read and verified in the

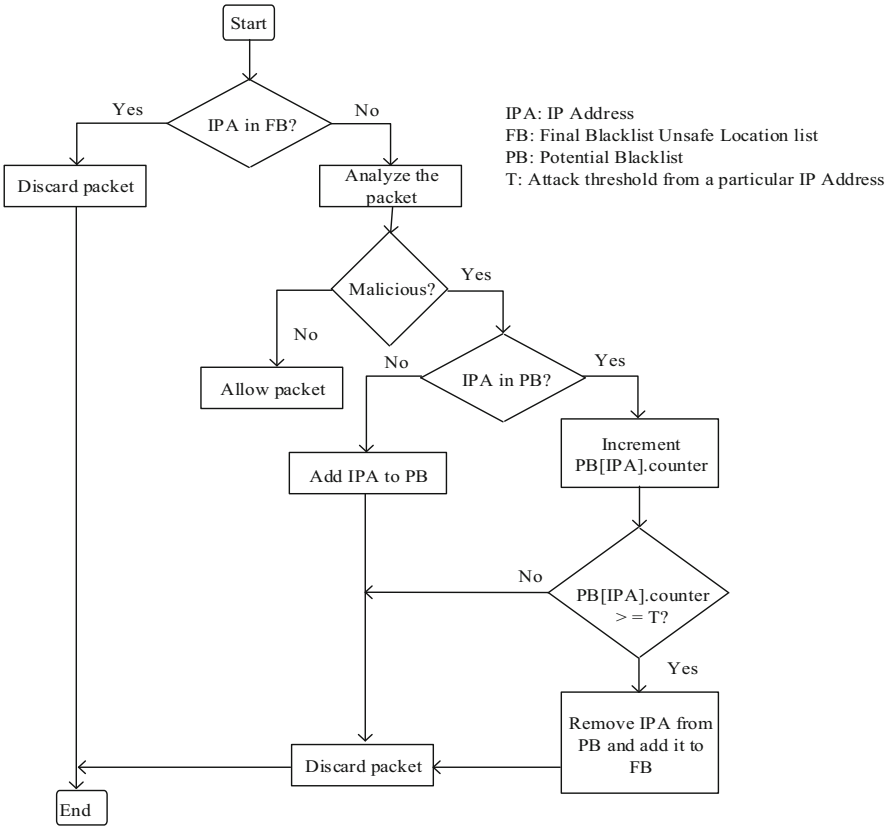


Fig. 8.3 Flowchart of dynamic creation of IP blacklist

final blacklist. If there is a match, then the packet will be discarded; otherwise, the packet will be analyzed by the firewall for any malicious content apart from the firewall policies. If any malicious content is found, then source IP address of the packet is added to the potential blacklist. A separate counter is maintained for each entry of source IP address in potential blacklist and will be incremented whenever a new attack is detected from the same IP address. When the counter value reaches the threshold limit T, IPA will be removed from potential blacklist and moved to final blacklist. Thereafter packets from the IP addresses of final blacklist are completely blocked.

8.3 Anomalies in Policy Configuration

The main and critical task of a security administrator is to configure the firewall policies. Due to the complex nature of policy specifications, it often leads to misconfiguration of policies which results in inconsistent and inaccurate policies. There are mainly two types of anomalies that can occur in policy configuration—redundant and conflict anomalies.

Redundant Anomaly When two policies represent the same set of packets with the same action, then they are said to be in redundant anomaly.

Conflict Anomaly When two policies represent the same set of packets with different actions, then they are said to be in conflict anomaly.

In our framework, we deal with three categories of firewall policies as discussed in Sect. 8.2.1: location-based compliance policies, perimeter policies, and group policies. In this section, we address the anomalies associated with these policies. As the policies in different categories may overlap with each other, they could result in redundant and conflict anomalies. We identified three possible cases that anomalies can occur in the multi-category policies—inter-category anomalies, intra-category anomalies, and intergroup anomalies.

Intra-category Anomalies In this case, the anomalies occur within the same category. For example, policies of perimeter policy category may overlap with each other. This may happen due to the security administrator's inadequate knowledge of policy configuration. The algorithm for the detection of intra-category anomalies is given in Algorithm 1.

ALGORITHM 1. Intra-Category_Policy_Anomaly_Detection
Input: Policy list P.
Output: TRUE or FALSE, and List of conflict and redundant policies.
RedundantPolicies = []; ConflictPolicies = []; status = FALSE
for *i in 1 to n-1* **do**
for *j in i+1 to n* **do**
    if *pi.service = pj.service and pi.value R pj.value*
    then

```
        if pi.action = pj.action then
        RedundantPolicies.append(pi,pj);
        else
        ConflictPolicies.append(pi,pj);
        end
        status = TRUE

    end

end
end
return status
```

Intergroup Anomalies In this case, the anomalies occur between the policies of two or more groups. For example, if a virtual machine firewall requires policies of two or more groups, then GPM has to check for the anomalies that could occur between the policies of two or more groups. The detection of intergroup anomalies is given in Algorithm 2.

ALGORITHM 2. Inter-group_Policy_Anomaly_Detection
Input: Policy list G1, G2,..Gn.
Output: TRUE or FALSE, and List of anomaly groups.

```
anomaly_policies = [ ], status = FALSE
for i in 1 to n-1 do
for j in i+1 to n do
    if Intra-Category_Policy_Anomaly_Detection(Gi+Gj) == TRUE
    then
            anomaly_groups.append(Gi,Gj)
            status = TRUE

    end

end
end
return status
```

Inter-category Anomalies In this case, the anomalies occur between the policies of two or more categories. For example, policies of compliance policy category may overlap with policies of perimeter policies and vice versa. These anomalies are resolved by using precedence of the categories. For example, if there are conflicts between local and group policies, then the conflicts are resolved by retaining group policy and removing local policies.

8.4 Global Policy Manager

The core part of our framework is the centralized controller called Global Policy Manager (GPM). It would be located at the headquarters of a multinational organization.

8.4.1 Dual Mode

GPM operates in two modes simultaneously—manual and automatic. In manual mode, security administrator can update the firewall policies of any firewall or group of firewalls in the global policy base. In automatic mode, GPM receives statistics or/and policy updates from the individual firewalls and updates the global policy base accordingly. In the above two cases, GPM applies the policy changes to the relevant firewalls if necessary.

8.4.2 Global Policy Manager Architecture

The global policy manager consists of four modules: *policy injection interface*, *FW interface*, *policy manager*, and *global policy base* as shown in Fig. 8.4.

8.4.2.1 Policy Injection Interface

In manual mode, security administrators inject the compliance and group policies into the global policy base. This injection occurs during initial configuration of the framework and also whenever there are changes in compliance and group policies. This interface should be made available only to the security administrator who is accountable for security policies of all the branches of an organization.

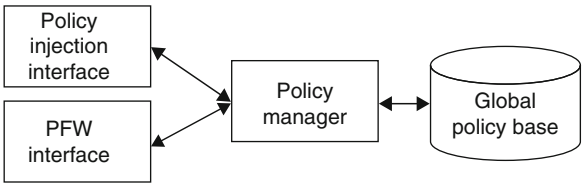


Fig. 8.4 Global policy manager architecture

8.4.2.2 FW Interface

The firewall interface communicates with the perimeter firewall and firewalls of all virtual machines of a data center. This interface is used to collect the statistics and policy changes from the perimeter firewalls.

8.4.2.3 Policy Manager

This is the key module which manages all other modules of the GPM. The activities of the policy manager include (a) analyzing the policies received from the policy injection interface and FW interface; (b) storing the policies in global policy base; (c) fetching the policies from the global policy base; (d) ensuring compliance, consistency, and correctness of the policies; and (e) communicating the policy updates to the applicable perimeter firewalls.

8.4.2.4 Global Policy Base

The policies of all the virtual machine firewalls and perimeter firewall of a data center are stored in global policy base along with the compliance and group policies.

8.4.3 Policy Configuration Procedures

Policy configuration is required to be performed in two scenarios: policies configured during initial setup of firewall and when there is any policy update due to change in existing policies.

8.4.3.1 Initial Policy Configuration

When a perimeter firewall is set up for the first time, policies should be configured as per the security requirements. The procedure of initial policy configuration is illustrated in seven steps (Fig. 8.5). Every PFW has to share location details to the GPM to apply relevant location-based compliance polices.

Virtual machine firewall has to specify the group IDs in order to acquire the group policies. GPM combines the group policies, resolves the conflicts if exists, and sends the final list of policies to the requested virtual machine firewall.

8.4.3.2 Policy Update

This framework supports dynamic policy updates. In other words, policies can be dynamically created, modified, or removed. This will be done in three ways: (a) local update, (b) GPM-initiated update, and (c) peer-initiated update.

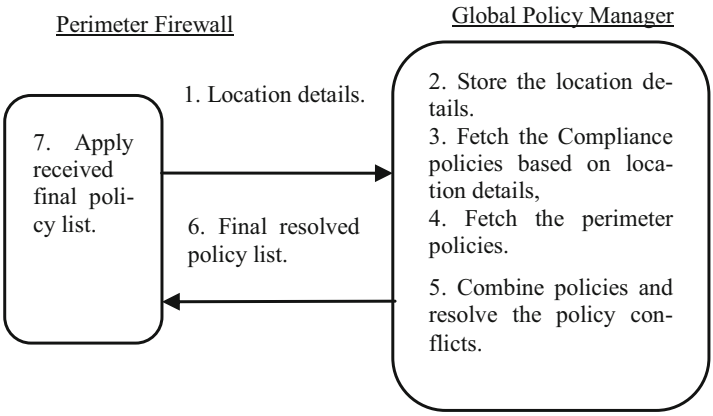


Fig. 8.5 Initial policy configuration

Local Update Whenever a firewall detects malicious content in the packets, it can block that kind of packets with introducing or modifying a policy. This policy is updated locally and will be informed to the GPM.

GPM-Initiated Update This policy update happens when a policy-change decision is made at the organization level and initiated by GPM itself.

Peer-Initiated Update GPM may propagate updated policies to a set of firewalls. These policies are derived from the statistics or other polices received from firewalls.

8.5 Conclusion and Future Work

We have presented dynamic firewall policy management framework which automates the policy management of distributed data center firewalls using a central controller called global policy manager. We have also introduced the location-based compliance policies and dynamic creation of IP blacklists to prevent the attacks from highly unsafe locations. We would like to extend our framework to include the other deployment models of the cloud computing (public and hybrid cloud).

References

1. Mell, P., Grance, T.: The NIST definition of cloud computing. National Institute of Standards and Technology special publication 800-145, 1–3 (2011)
2. CloudPassage: Cloud Security 2016 Spotlight report. https://pages.cloudpassage.com/rs/857-FXQ-213/images/cloud-security-survey-report-2016.pdf (2017). Accessed 28 Feb 2017

3. Amazon web services: AWS Global Infrastructure. https://aws.amazon.com/about-aws/global-infrastructure/ (2017). Accessed 28 Feb 2017
4. Microsoft Azure: Azure regions. https://azure.microsoft.com/en-us/regions/ (2017). Accessed 28 Feb 2017
5. Wikipedia: Internet Censorship in Iran. https://en.wikipedia.org/wiki/Internet_censorship_in_Iran (2017). Accessed 28 Feb 2017
6. Wikipedia: Internet Censorship in China. https://en.wikipedia.org/wiki/Internet_censorship_in_China (2017). Accessed 28 Feb 2017
7. Maddumala, M.N., Kumar, V.: A logic-based security framework for mobile perimeter. In: 16th IEEE international conference on mobile data management (MDM), pp. 30–33 (2015)
8. Jaiswal, C., Maddumala, M.N., Kumar, V.: Location-based security framework for cloud perimeters. IEEE Cloud Comput. **1**(3), 56–64 (2014)

Chapter 9
Evolution of Sensors Leading to Smart Objects and Security Issues in IoT

Sanjeev Kaushik Ramani and S. S. Iyengar

Abstract As human beings started using technology for a better living, came the need for sensing and sensory devices. Sensors brought about a change in the way things are perceived and provided a definitive way forward in the dream of making comfortable living affordable for human beings. The evolution of sensors leading to the emergence of smart objects that are an integral part of the cyber-physical systems (CPS) and the Internet of Things (IoT) have been discussed in this paper. The various challenges that exist in the field of security of the IoT devices and the ways of handling them are outlined. An attempt has been made to discuss the concept of botnets and the various ways in which attackers exploit these especially in the IoT devices and the possible outcomes.

Keywords Sensors · Wireless sensor networks (WSN) · Internet of things (IoT) · Cyber-physical systems · Botnets

9.1 Introduction

It is fascinating to see the number of places in which sensors are being used nowadays, and the number is expected to increase exponentially when the Internet of Things (IoT) devices start replacing normal day-to-day devices. The sensors that we talk of today are existing because of the advancements that we have made in the fields of material sciences and engineering and is being extensively utilized in electronics and computer science. The birth of sensors can be traced back to the early 1800s when one of the first known sensors was built based on the variations in the electrical resistance at different temperatures. This concept gave rise to a copper-based temperature sensor [1]. Since then there have been various new ideas that have been exploited to bring in new sensors to the market. However, drastic

Sanjeev K. R. (✉) · S. S. Iyengar
School of Computing and Information Sciences, Florida International University, Miami, FL, USA
e-mail: skaus004@fiu.edu; iyengar@cs.fiu.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_9

changes in the way sensors were perceived and manufactured came with the advent of semiconductor physics and the various ways in which miniature sensors could be manufactured at low prices.

Sensors today are being used in various applications across many fields. From the concept of having large handheld sensors, we have grown a long way to the current state of the art wherein the sensors are being embedded on to the body or clothing giving rise to wearable computing. Also, the increasing computational capabilities of sensors and the demands of varied applications have led to the advent of smart objects. These smart objects are the basic elements or building blocks of the *cyber-physical systems* (CPS) and *Internet of Things* (IoT).

In this paper, we shall give an exhaustive insight into the way sensors evolved and the advancements that led to the development of smart objects. We shall also discuss about the growth of wireless sensor networks (WSN) and further extend our discussions toward the growth of CPS and IoT. The latter part of the paper highlights the vulnerabilities in IoT devices especially with the growth of botnets and the cyberattacks and their possible impacts on human life. We shall also present the challenges and the ways to mitigate them.

9.2 Sensors and Its Classification

A sensor can broadly be defined as any device that can collect a physical quantity and either process or transmit it in a format suitable for processing at another location. The working of a sensor depends on its active material which is usually called a *transducer*, i.e., something that converts energy from one form to another [1, 2]. All components of a sensor are usually put in a single package with various connectors that would be used for interfacing with the external systems.

One way of classifying the sensors is based on the energy forms they work with as shown in Table 9.1. Another way of classifying sensors is based on the field in which they are used like biological, chemical, electrical, electromagnetic, heat/temperature, magnetic, mechanical motion (displacement, velocity, acceleration, etc.), optical, and radioactivity. Sensors can also be broadly classified based on the applications they are built for like monitoring and surveillance, automation, remote sensing, etc.

The growth in technology that made the miniaturization of the sensors possible opened the doors for sensors to be extensively used in the systems built for process automation. It has also made embedding and mounting of sensors in machines built for specific applications in places where it is impossible for human access. Sensors find applications in various systems such as health care, automobiles, airplanes, communication systems, chemical industries, and many more.

The fundamental components of a sensor (Fig. 9.1) include the sensing component, a power source, and communication or processing components. The sensors could also comprise of additional connectors if they have to interface with other

Table 9.1 Various sensor energy forms

Energy forms	Example measurements
Mechanical	Length, area, volume, all time derivatives such as linear/angular velocity, linear/angular acceleration, mass flow, force, torque, pressure, acoustic wavelength, and acoustic intensity
Thermal	Temperature, specific heat, entropy, heat flow, state of matter
Electrical	Voltage, current, charge, resistance, inductance, capacitance, dielectric constant, polarization, electric field, frequency, dipole moment
Magnetic	Field intensity, flux density, magnetic moment, permeability
Radiant	Intensity, phase, wavelength, polarization, reflectance, transmittance, refractive index
Chemical	Composition, concentration, reaction rate, pH, oxidation/reduction potential

Source: https://www.nap.edu/read/4782/chapter/4

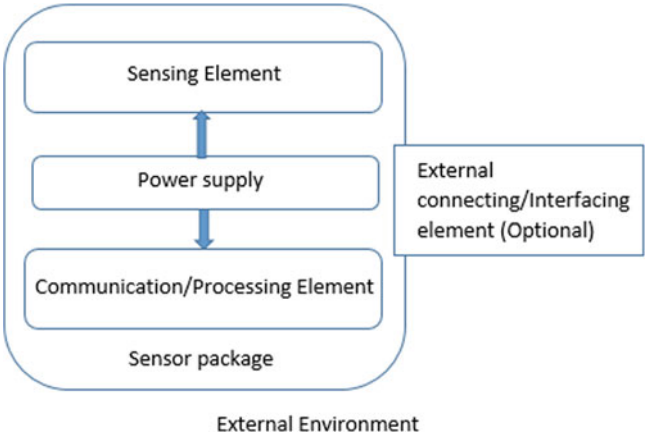


Fig. 9.1 Fundamental components of a sensor

sensors or processors [3]. All these components are carefully packaged into a single unit.

The additional components that a sensor could possess include interconnects for input and output, calibration elements, signal-modifying elements, and actuators. With improved technologies and better packaging techniques, the sizes of sensors have gone down dramatically over the last decade. The miniaturization along with an increase in efficiency led to the development of smart sensors.

9.3 Wireless Sensor Networks

With the advances in wireless communication and digital electronics, there came about a change in the way sensor nodes were designed and the communication

mechanism also saw a complete change [4]. Sensors that were wired till then could be made to communicate wirelessly through existing protocols or formulating new ones. This brought about a radical change in how sensors were used. It opened up new applications and also made processing possible through the concepts of distributed computing. It also promoted the sharing of data and resources through sensor nodes which are configured in the form of networks. This led to the formation of *wireless sensor networks* (WSN).

A wireless sensor network (WSN) can be defined as a wireless network of sensors that are spatially distributed and have a specific function in the environment of deployment. A WSN system usually incorporates a gateway that provides wireless connectivity back to the wired world and distributed nodes [7]. WSNs have many advantages over the traditional sensors in terms of size, the ease of deployment, possibility of collaborative sensing, lower maintenance costs, and self-organizing capabilities to name a few.

Though WSN offers a lot on the plate, there are some issues that have to be addressed. The major issue is security of WSN system and the data that is transmitted by them. The researchers are actively working in this field.

9.4 Smart Sensors

As the industries grew, the need for technology grew and the demand for sensing equipment that are smart also started increasing. The advancements made in semiconductor physics and the *very-large-scale integration* (VLSI) techniques have been used for manufacturing of silicon devices (i.e., efficient and low-cost sensor elements). These sensor elements are multilayered sensors and arrays of sensors. Such sensors could be molded to provide a smart sensing environment. Thus, began the era of smart sensors.

A smart sensor is capable of capturing required parameters from the environment and performing certain predefined computations on the input. This provides a more readable and less erroneous output [1]. Such devices have many applications ranging from monitoring and controlling the environments in a smart grid, reconnaissance in war zones, exploration of new places, etc.

The growth of smart sensors is one of the initial and crucial steps in making the concept of *Internet of Things* (IoT) possible. It has also made it a reality of having a *unique identifier* (UID) for any device in the world and made it competent enough to communicate over the Internet. Smart sensors also find their applications as nodes in a more advanced WSN with built-in actuators termed as the *wireless sensor and actuator network* (WSAN). A smart sensor could have a more complex anatomy as compared to a traditional sensor, and it is based on the function it performs in the environment.

The smart sensors have various advantages over traditional sensors. They are more robust, easy to maintain, and easy to calibrate and have higher availability, better precision, and better collaboration options to name a few. As smart sensors

became widely used in the industry, devices that use multiple such sensors were developed to accomplish complex tasks with ease. Such devices are called smart objects. The following sections talks more about smart objects.

9.5 Process Automation

From when sensors started being used extensively, one of the many applications has been its use in automating the current system. Various industries have employed robots and other smart devices that work based on the data collected by the sensors or network of sensors deployed in the places of work. It is a hard sight nowadays to visualize humans working in places prone to radioactive discharges and in places that are uninhabitable for humans. Specialized or customized sensors have been manufactured by the industries to meet the special requirements.

Most of the industrial automation lines today use sensors starting from modeling and assembling to more complex tasks having catastrophic outcomes. A computer code is usually associated with the sensor and acts as the decision-making module. There are sensors that check for failures of such automated devices too.

Automation is not just applicable only to industries. With the advent of the IoT and CPS, sensors are used in all day-to-day appliances with the motive of automating the world and making it a better place to live in. This also has an impact on the energy consumption. Studies reveal that the usage of appropriate sensors at optimal places reduces the energy consumption by 30–40% which would play a huge impact in preventing global warming.

However, overdependence on the sensors could spell doom to humans if the control falls in the hands of the wrong people. The intelligence that these sensor networks possess could be exploited by adversaries to start off an outbreak of attacks. There have already been evidences of such attacks. Due to the connectivity to internet, the attacks have become more severe and pronounced and would be discussed in the sections below.

9.6 Smart Objects

The discovery of smart sensors led to the manufacturing of new devices that could create a link between the real world and a virtual environment along with a certain degree of intelligence [5]. Such smart objects have become a part of our everyday life and keeps people constantly connected to the digital environment and people. It has also modified the way in which we communicate—be it the content-driven and context-aware computing or the real-time localization using embedded sensors. These objects working in collaboration as a loosely coupled distributed network have been the backbone of the cyber-physical systems and the Internet of Things [6].

The smart objects are built on the vision of the father of ubiquitous computing Mark D. Weiser about every object containing a computer or a tab within itself making information retrieval trivial [8]. The concept of Smart Objects further evolved based on the ideas put forth by Marcelo Kallmann and Daniel Thalmann in their work on a new approach for interaction between the sensors and the virtual environment [9].

Recent smart objects are incorporated with artificial intelligence and are expected to possess the following fundamental properties, namely, *context awareness* and reactive behavior, ability to *abstract and represent* the parameters captured to *interact* with the users and the surroundings, and also work on any kind of feedback [6].

Smart objects are usually broadly categorized as objects that either be *activity aware* if it could record environmental activity and events, *policy aware* if it could check if the activities are as per the set rules, and *process aware* if it can follow a set process and work accordingly [6]. Such smart objects are also expected to have intelligence based on which it could make decisions, notify a problem, or even handle information.

9.7 Cyber-Physical Systems

Cyber-physical systems have been an area of research that is funded by many organizations including the US National Science foundation (NSF) who defines CPS as a system that is engineered and works on the seamless interaction and integration of the physical system with the computational algorithms [10]. It is expected to revolutionize the way people interact with the man-made devices and manage information. This includes its various applications in many sectors of industries. All this progress is only possible because of the presence of smart objects and artificial intelligence. This field is still in its infancy, and we would need greater technological advances before we can have a fully functional CPS in all walks of life. There are various applications of CPS in smart grids and autonomous systems including process automation, aviation, data transfer, and storage to name a few [11].

The advent of CPS would add in more intelligence to the existing embedded systems that could become more adaptable, efficient, robust, safe, and scalable [12]. The challenges that we would have to overcome for the CPS to be extensively deployed include the variations in the design patterns followed by the various engineering disciplines, security of such devices, and the communication medium used by them. The ways to handle and the processing of highly voluminous big data that is generated by these systems and their security are also some of the challenges to be addressed. The popularity of smartphones has also encouraged the inroads that we have made in this field.

9.8 Internet of Things (IoT)

Internet, which changed the way people communicated, is more than 45 years old. There have been many revolutionizing changes with the introduction of the *World Wide Web*, and the mobile internet was introduced in the 2000s [13, 14]. Now we are shifting our focus toward context-aware and content-aware computing. This gave birth to the concept of Internet of Things. The Internet of Things (IoT) is a broad term used to indicate the interconnection and exchange of data among physical devices and other sensor-based devices through the internet. In IoT, there are millions of smart devices that connect to the internet. It is based on the vision of connecting and having access to all real-life physical devices including vehicles, buildings, utility devices, household items, control systems, and other things in real time over the internet. All the above promises are made based on the rapid advancements made in sensor networks and radio frequency identification (RFID) technology making it an integral part of human life today.

The first mention of the term IoT was by the famous British visionary Kevin Ashton in 1999 [17]. Ever since, it has grown leaps and bounds and has various predictions and investments made by many major players in the industry on the outcomes of IoT. The predictions make IoT look complex and also show the enormous benefits that it can provide to people including smart cities, smarter economies, better health facilities, and many more [15]. Technology and market trends have led to the steady growth of the number of devices getting included in IoT.

It is projected that the number of such connected devices have already outnumbered the human population. The revenue that these devices generate is also very huge. A report from Mckincey Global Institute states that the IoT business would deliver approximately $6.2 trillion in revenue by 2025. There were predictions from top officials of leading companies that the world would be flooded with more than 50 billion IoT devices by 2020 (Fig. 9.2). However, we are nowhere close to achieving that number. A recent study by Gartner in 2016 shows that there are only close to 7 billion devices in the market [16]. This slow growth is due to the many challenges that are still to be addressed which include issues related to security, privacy, interoperability, regulatory compliance, and other social and economic issues.

In this paper we would concentrate on the security issues and vulnerabilities that exist in IoT devices with focus on the struggle against botnets.

9.8.1 Security Vulnerabilities of IoT

All IoT sensors and devices are prone to breach of the tenets of information security (confidentiality, integrity, and availability). A study conducted by a technology institute based in France identified about 38 vulnerabilities in terms of poor

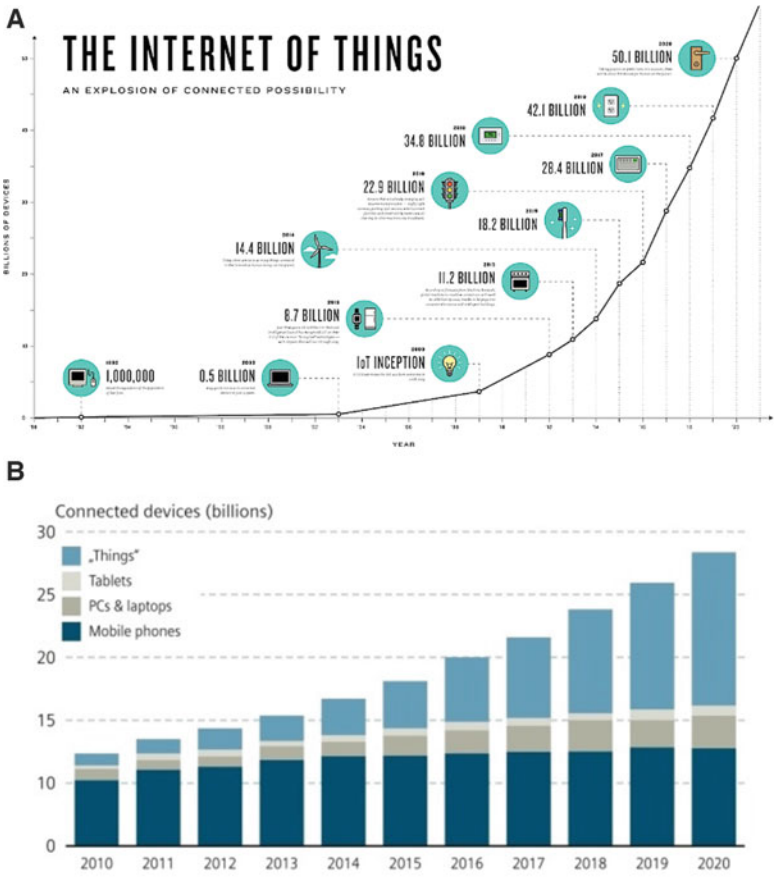


Fig. 9.2 Number of devices connected to the internet by 2020: (**a**) old prediction, (**b**) new prediction (Source: (**a**) https://www.i-scoop.eu/internet-of-things-guide/; (**b**) http://iothought.com/forcast-for-internet-of-things/)

encryption, presence of backdoors, and possible unauthorized access in the 123 products whose firmware images they tested. IoT is one such domain wherein one weak link could spell doom to all devices connected to the network [18].

The introduction of sensors in devices have made it easy for the attackers to track the activities performed by the user of the device. The fact that these devices are highly pervasive into most places puts the privacy of the user at great risk. Connected devices also help companies to collect sensitive data and create profiles of people that could be compromised by an attacker.

In IoT, we talk about devices that are connected over the internet which opens up new doors for the device being hijacked and utilized for illegal activities by the attackers. This gives a new life to the old concept of botnets that became famous after the introduction of distributed computing. A botnet consists of computers that

remotely accesses resources without the knowledge of the owner [19]. IoT being a network of many physical systems that are equipped with IP and MAC addresses and transmit and receive data over the internet makes it an easy breeding ground for attackers to inflict a widespread network attack.

9.8.2 History of Botnets

A network of infected machines that are being used for coordinated attacks and illicit purposes are called zombies or bots [25]. Technically, a bot refers to a device once infected by a malware. This becomes part of a network of infected devices that would be controlled by a single or group of attackers (botmasters) to inflict an attack—a *distributed denial of service* (DDoS) attack. The impact of such an attack could be very huge. The bots try to find vulnerabilities in surrounding networks and spread rapidly as Trojan horses thus infecting a lot of machines [20]. Early records of botnets and studies reveal that they do not consume a lot of network traffic or bandwidth and hence usually go under the radar and are not identified.

There have been various malwares that have been translated as a botnet starting with Zeus, which was first detected in 2007. *Zeus*, also known as *Zbot*, has been used extensively by attackers to extract financial data and banking information. This has also played an active role in the recent ransomware attack. There are many more of such bots in the internet that have been discovered and are closely being monitored.

9.8.3 Botnet Attacks in IoT

One of the early instances of bots being discovered with respect to IoT was reported by a researcher at Proofpoint [19]. It was noticed that there were more than thousands of emails logged in a security gateway maliciously and originated not just from computers but also some household appliances like refrigerators. The discovery of such botnets is becoming very common in recent times [24].

IoT offers cheap and inexpensive devices like webcams, CCTVs, baby monitors, home security systems, wearables, and other devices that connect to the internet but do not usually have a well thought about security system in place.

The systems are usually in motion and communicate through heterogeneous ways. There are some systems that control other systems with portions not protected or may not be intended to be connected to the internet [25]. Some may not even be password protected. These offer an apt breeding ground for the bots to spread and exploit the conditions. The attacks could range from DDoS (Fig. 9.3) to even the encryption of data as in ransomware [21]. Another study by *Forbes Magazine* reveals that in October 2016, a botnet was identified comprising of almost 100,000 IoT devices that were not secured and led to loss in availability of an internet

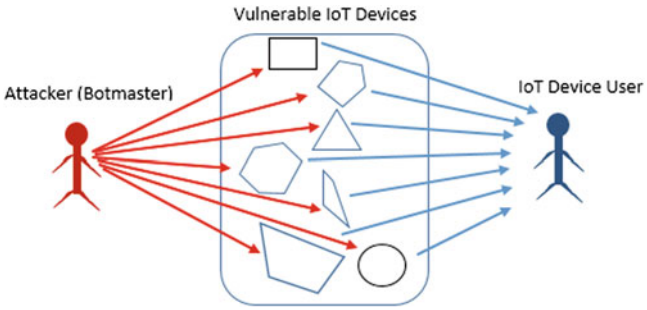


Fig. 9.3 DDoS attacks using IoT devices

infrastructure provider leading to top websites not being available for a short duration [21].

The other worrying factor is that the botnets utilize all the resources it can harness and thus grow rapidly with the multiple IPs in use (since each IoT device has a unique IP of itself) and is highly impossible to track or prevent further damage. Sometimes these bots are said to even morph themselves to prevent being traced (Fig. 9.3).

Recent studies have shown the use of "Hajime," termed as a vigilante IoT worm to clean up the vulnerabilities spread by illicit botnets like "Mirai" (a botnet threatened to take over the internet world [22, 23]) according to its creator. However, the security of such worms is also a concern, and overreliance on it is advisable as it could easily be abused by attackers for their own misleading ideas. According to Kaspersky labs, botnets like Hajime and Mirai exploit the default credential combinations available as a list to induce a trial-and-error way of getting into unsecured networks using the open telnet ports [26–28]. This was first identified by people at Rapidity Networks in October of 2016 and reported to have been responsible for a 600 GBps attack. Currently, the Hajime lacks any attacking code or capability but only has an active propagation module. Another interesting aspect of the Hajime botnet which seems to go by its creator's claim that it is safe is that once it gains access to an unsecure IoT device, it blocks access to four ports, viz., 23, 7547, 5555, and 5358, which are some of the ports that illicit botnets like Mirai exploit to gain access and cause havoc [22].

9.8.4 Techniques to Protect Devices

We could use *proactive* and *reactive* ways to protect devices from infection due to botnets. The proactive ways would include setting up a strong password, updating the security patches, and installing monitoring software that could detect anomalies. *Email filtering* is also a good way to prevent botnets as one of the recent botnets Locky was spread through spam emails [25]. The reactive techniques are to be used

once it is ascertained that the system has been infected by the various signs that it displays. In such cases, it is highly recommended to remove the device from the network to prevent further spreading of the infection. There are certain tools available that can be used for tackling the botnets. The tools utilize the fact that the IoT devices are specialized systems programmed to work with specific inputs and in a definitive manner, thus making it easy to predict the course of action that is required.

9.9 Conclusion

Sensors have changed the way we perceive life. The primary objective of sensors is to acquire data from physical world and also to operate and monitor the systems continuously. Sensor-based devices enable process automation. The growth of sensors over the last few decades is phenomenal. The successful products are smart sensors, smart objects, and cyber-physical systems. Internet has also grown into being integrated to be called the IoT through which it is possible to connect devices also to internet and to exchange information among people, systems, and devices.

The growing popularity of IoT-based devices and the need for them are not going to go down. On the other hand, the users must also be made aware of the security needs of such devices. People should select the devices with at least a minimum level of security. Also, the users need to be informed to change passwords frequently to prevent any kind of brute force attack and from botnets that try to spread through unsecured ports. We should also update the firmware and add firewalls or even change compromised devices. The government should bring in policies to check for a minimum level of security in each of the "smart" devices before they are available for consumption in the market and set benchmarks to determine their safety. Otherwise, the botnets could spread like an inexorable spirit, and we would never know the extent to which they can have an impact.

References

1. Expanding the Vision of Sensor Materials at NAP.edu (n.d.). Retrieved July 1, 2017, from https://www.nap.edu/read/4782/chapter/4
2. Intro to Sensors – NYU Tandon School of Engineering (n.d.). Retrieved July 1, 2017, from http://www.bing.com/cr?IG=D946654F7BDF48F2ADD0EDB472D6BE97&CID=212D33CEA5E168882019390EA4E76944&rd=1&h=k_FqJGUWwaKnW7OmvP1fSxtdmMNYH9ZW3sfnZlZqBy8&v=1&r=http%3a%2f%2fengineeing.nyu.edu%2fgk12%2famps-cbri%2f%2fpdf% 2fIntro%2520to%2520Sensors. pdf&p=DevEx,5062.1
3. Staszewski, A.R.: What is the anatomy of IoT sensor devices? (2016, July 27) Retrieved July 1, 2017, from http://zenseio.com/blog/anatomy-iot-sensor-devices
4. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey, Elsevier Science B.V, pp. 393–422 (2002)

5. Fortino, G., Trunfio, P. (eds.): Internet of things based on smart objects, technology, middleware and applications. Springer, Cham (2014). ISBN 978-3-319-00491-4
6. Kortuem, G., Kawsar, F., Sundramoorthy, V., Fitton, D.: Smart objects as building blocks for the Internet of things. IEEE Internet Comput. **14**(1), 44–51 (2010). https://doi.org/10.1109/MIC.2009.143
7. Buratti, C., Conti, A., Dardari, D., Verdone, R.: An overview on wireless sensor networks technology and evolution. Sensors. **9**, 6869–6896 (2009). https://doi.org/10.3390/s90906869
8. Weiser, M.: The computer for the 21st century. Sci. Am. **265**, 94–104 (1991). https://doi.org/10.1038/scientificamerican0991-94
9. Modeling Objects for Interaction Tasks Marcelo Kallmann and Daniel Thalmann
10. National Science Foundation – Where Discoveries Begin (n.d.). Retrieved July 5, 2017, from https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503286
11. Khaitan, S.K., McCalley, J.D.: Design techniques and applications of cyber physical systems: a survey. IEEE Syst. J. **9**, 350–365 (2014)
12. Alippi, C.: Intelligence for embedded systems, Springer Verlag, 283 pp. ISBN 978-3-319-05278-6 (2014)
13. Sundmaeker, H., Guillemin, P., Friess, P., Woelfflé, S.: Vision and challenges for realising the internet of things, March (2010)
14. Edens, G., Scott, G.: The Packet Protector, IEEE Spectrum, pp. 42–48, April (2017)
15. Ramirez, E.: Privacy and the IoT, Opening remarks of FTC chairwoman: navigating policy issues international consumer electronics show Las Vegas, Nevada, January 6 (2015)
16. Nordrum, A.: Popular internet of things forecast of 50 billion devices by 2020 is outdated (2016, August 18). Retrieved July 1, 2017, from http://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated
17. Internet of things: complete IoT guide – benefits, risks, cases, trends (n.d.). Retrieved July 1, 2017, from https://www.i-scoop.eu/internet-of-things-guide/
18. Samani, R.: 3 key security challenges for the Internet of things (2016, June 06). Retrieved June 26, 2017, from https://securingtomorrow.mcafee.com/business/3-key-security-challenges-internet-things/
19. What is IoT botnet (Internet of Things botnet)? – definition from WhatIs.com (n.d.). Retrieved June 25, 2017, from http://internetofthingsagenda.techtarget.com/definition/IoT-botnet-Internet-of-Things-botnet
20. What is botnet? – definition from WhatIs.com (n.d.). Retrieved June 25, 2017, from http://searchsecurity.techtarget.com/definition/botnet
21. Marr, B.: Botnets: the dangerous side effects of the Internet of things (2017, March 07). Retrieved June 25, 2017, from https://www.forbes.com/sites/bernardmarr/2017/03/07/botnets-the-dangerous-side-effects-of-the-internet-of-things/#2766bb833304
22. Khandelwal, S.: Hajime (2017, April 27). Retrieved June 25, 2017, from http://thehackernews.com/2017/04/vigilante-hacker-iot-botnet_26.html
23. John Leyden 27 Apr 2017 at 16:02 tweet_btn (): Mysterious Hajime botnet has pwned 300,000 IoT devices (n.d.). Retrieved June 25, 2017, from https://www.theregister.co.uk/2017/04/27/hajime_iot_botnet/
24. The IoT's Scramble to Combat Botnets (n.d.). Retrieved June 25, 2017, from http://www.technewsworld.com/story/84523.html
25. Bertino, E., Islam, N.: Botnets and Internet of Things Security, IEEE COMPUTER SOCIETY, February 2017
26. Jerkins, J.A.: Motivating a market or regulatory solution to IoT insecurity with the Mirai Botnet Code, IEEE (2017)
27. Kolias, C., Kambourakis, G., Stavrou, A., Voas, J.: DDoS in the IoT: Mirai and other botnets, IEEE Computer Society, July 2017
28. Angrishi, K.: Turning Internet of Things (IoT) into Internet of Vulnerabilities (IoV): IoT Botnets, arXiv:1702.03681v1 [cs.NI] (2017)

Chapter 10
Multimedia Data Management for Disaster Situation Awareness

Maria E. Presa Reyes, Samira Pouyanfar, Hector Cen Zheng, Hsin-Yu Ha, and Shu-Ching Chen

Abstract To raise awareness in disaster situations, the quality and analysis of disaster-related big data are essential. Recent developments in the collection, analysis, and visualization of multimedia data have led to a significant enhancement in disaster management systems. Crowdsourcing tools, for instance, allow citizens to perform an active role in reporting information relevant to disaster events at a global scale through popular social media sites such as Twitter and Facebook. As multimedia data analysis becomes further advanced, it can augment the disaster situation awareness and provide an efficient and timely response. This paper describes how multimedia data management plays a prominent role in improving the capabilities to readily manage disaster situations. Specifically, visualization provides a more convenient and user-friendly means for individuals who have limited experience in disaster situations. A case study introducing a 3D animation system is presented, which simulates the impacts of storm surge near coastal areas.

Keywords Big data · Multimedia · Disaster management · Data management · Visualization · Simulation

10.1 Introduction

Natural and man-made disasters are unpredictable and difficult to manage and cause considerable devastations to city infrastructures and loss of human lives. According to CNN Money [22], natural disasters caused a global cost of 175 billion dollars in 2016, for which only 30% (about 50 billion) were insured. This loss was recorded as the highest in the past 4 years. Furthermore, it is also important to note that in

M. E. Presa Reyes (✉) · S. Pouyanfar · H. C. Zheng · H.-Y. Ha · S.-C. Chen
School of Computing and Information Sciences, Florida International University, Miami, FL, USA
e-mail: mpres029@cs.fiu.edu; spouy001@cs.fiu.edu; hcen001@cs.fiu.edu; hha001@cs.fiu.edu; chens@cs.fiu.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_10

2016, North America suffered a total of 160 disaster events, which is also reported as the highest number ever since 1980 [22].

Disaster management is often divided into four well-known phases: prevention/mitigation, preparedness, response, and recovery. Prevention or mitigation methods are developed through the different types of approaches, such as disaster risk analysis, vulnerability analysis, resource management, and planning. Most disasters may be impossible to prevent but can be effectively mitigated by analyzing the situation and making the right decisions. Disaster data modeling is helpful to understand the risks and to create an estimate for the losses given a certain scenario. Modeling such a complex system requires the collaboration of professionals in various fields [27]. By analyzing past events, data can be simulated, and possible future scenarios can be assessed.

Multimedia data includes text, image, audio, video, etc. In disaster management, various types of data might come from different sources, and thus effective data fusion and multimodal data analysis approaches are important and challenging to extract useful knowledge for a better understanding of a disaster. Furthermore, visualization is helpful to represent the combination of several multimedia data types in a way that is easier to understand and interpret. Figure 10.1 shows a high-level framework of multimedia data management for disaster situations.

This paper is organized as follows. Section 10.2 discusses various multimedia data types in disaster management. Several well-known data analysis techniques are introduced in Sect. 10.3. A case study introducing the rendering of a storm surge animation in a 3D environment is given in Sect. 10.4. Finally, Sect. 10.5 presents the outlook and future direction.

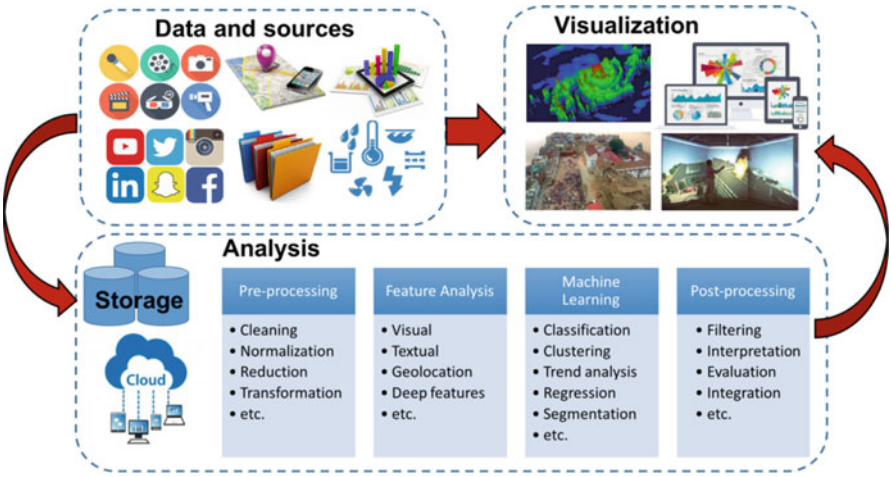


Fig. 10.1 Multimedia data management framework for disaster situations

10.2 Multimedia Data in Disaster Management

The process of data collection and management to provide the aid, relief, and response in disaster situations has always posed a challenge to public emergency services and the disaster management community, due to (1) time constraints; (2) the scattered nature of data sources, owners, and types; and (3) speed of access to data, as pointed by Meissner et al. [15].

Different multimedia data types and sources related to disaster management are listed below.

- **Visual Data** With the advent of low-cost and high-quality imaging devices, at both the consumer and professional levels, the speed and rates of data generation have spiked. The analysis of visual data involves multiple research disciplines, ranging from geospatial analytics to data management and knowledge representation [13]. Recent efforts in enhancing the capture of both image and video, during any phase of the disaster management cycle, have seen a great benefit from the employment of unmanned aerial vehicles (UAVs), specifically from (a) the lower costs of UAVs, (b) its increasing capabilities to capture high-resolution images, and (c) the large amounts of data and metadata generated. In [19], an image dataset is built from the key-frame samples of disaster-related videos on YouTube, which was used as the input of the training and testing of their proposed weighted discretization analysis framework. A very important characteristic of the datasets built from visual data is mentioned in that paper, namely, their imbalanced nature. In [29], Yang et al. take a step further and classify images along with the textual information. Additionally, data from situation reports and user feedback are also taken into account.
- **Sensors Data** The rapid and explosive growth of mobile phones, wearable technology, and Internet of things (IoT) devices has established the collection and processing of data from sensors as an ongoing discipline and research trend. Sheng et al. [23] present one of the first general approaches at leveraging the sensors' capabilities in mobile phones. An important topic in this research is the management of energy efficiency through collaborative sensing as a means to reduce redundancy, thus reducing the energy consumption. A more recent and disaster-management oriented approach is proposed in [5]. By combining the sensing capabilities of Wireless Sensor Networks (WSNs) and the visual capturing capabilities of multi-UAVs systems, applications in disaster management can range from early detection systems to damage assessment and rescue missions, to name the most important ones. In their approach, WSNs act as the detection systems that trigger the launch of diverse types of UAVs in the wake of a disastrous event.
- **Geographic Data** Geographic Information Systems (GIS), like all technology-supported systems, have come a long way from their inception and underutilization days due to technology limitations [30] to today's pervasiveness. The combination of low cost, mobility, and portability of new devices with geographic capabilities, as well as user-generated content, willingly or unwillingly, has built

enormous databases for different purposes. In 2007, Goodchild [7] coined the term *volunteered geographic information* (VGI) to reference all the voluntarily user-generated geographic and geospatial content. Since then, almost all related literature and research in the area have used this term. A broad survey by Granell & Ostermann [8] about the literature and research where VGI is paramount in disaster management found that (a) most of the studies still focus on data and how it should be collected and processed, (b) Twitter is used as a major data source in data-centric research, (c) reliance on Twitter as a unique data source might hinder the validity of the results, and (d) few studies combine VGI with data from official sources.

- **Social Media Data** As for social media data, thanks to the pervasiveness of social media networks, blogs, content generation tools, and photo and video sharing applications, users have become active entities rather than passive participants [6]. Research focus related to disaster-related data originating from social media streams is heavily inclined toward two main trends: (1) the extraction of useful pieces of information and data from the diverse social media networks and (2) the crowdsourcing of knowledge during disaster events. Imran et al. have done extensive research about extraction in [11]. However, as data from social media can sometimes be inaccurate, extemporaneous, malicious, and/or noisy, Planella Conrado et al. [3] have proposed a framework to ensure the quality of data and information at the user and emergency field-responder levels, with the participation of the informed actors and based on near real-time verified data to support the decision-making processes of the non-informed participants.
- **Storing and Sharing Data** There are several advantages and opportunities that the current state of technology can provide: (1) large amounts of data can be generated by the ordinary users, and (2) current technology provides a means to store and process all of these data. Grolinger et al. [9] have proposed a Knowledge as a Service (KaaS) framework as a means to handle a near complete integration of different data sources. By leveraging the advantages of the relational and NoSQL databases, along with diverse cloud technologies, the proposed framework can be capable of (a) satisfying some of the requirements of data for disaster management systems outlined by Meissner et al. [15] and (b) dealing with some of the characteristics noted by Hristidis et al. [10].

10.3 Multimedia Data Analysis in Disaster Management

A key aspect of disaster management systems is how to analyze the data generated in disaster-based situations in an efficient and effective manner. Data mining and machine learning techniques can be utilized to address the challenges in disaster data analysis. These techniques can be divided into several main steps, such as preprocessing, feature or attribute analysis, learning, and post-processing.

- **Preprocessing** Similar to general multimedia data, real-world disaster data can be extracted from multiple heterogeneous sources. Therefore, it is noisy, incomplete, skewed, and inconsistent in nature. In reality, it is impractical to use such complex raw data as the input of the analysis algorithms, and thus careful preprocessing techniques are needed. In general, preprocessing includes, but not limited to, data cleaning, normalization, formatting, reduction, transformation, and missing value interpolation [10]. Jain et al. [12] present a real-time disaster data mining framework based on social networks. Since the extracted tweets include many irrelevant information, data preprocessing is essential to save computational power and to generate more useful data. In particular, language and geographic information are utilized to filter and clean this large and noisy data. Preprocessing is also an important component of the Florida Hurricane Loss Model (FPHLM) [27], a public hurricane model estimating loss costs for personal and commercial residential properties, because this model receives different data formats, often containing noisy and missing values, from the insurance companies. Although preprocessing is a key part of every disaster management system, it was barely discussed in the literature. To the best of our knowledge, there is no general tool or application to automatically clean and format the disaster multimedia data. This can be due to the heavy domain knowledge needed for each disaster-based system which cannot be generalized for other situations.
- **Feature Analysis** Features or attributes are used to discover the knowledge in a dataset and characterize the instances in it. Researchers extract and select useful low-level and mid-level features from the data to find its high-level contents systematically. As disaster data may include different data types, including visual, aural, textual, GIS, etc., it is challenging to extract multimodal discriminative features from the data instances. Low-level visual features such as color, shape, texture, and wavelet, as well as textual features, are extracted and combined for situation report enhancement in [28]. Geospatial information is another important feature widely used in current disaster management systems. A real-time crisis mapping framework is developed to geoparse tweet contents [16]. This framework uses the existing geospatial tools (e.g., OpenStreetMap) to extract street-level, buildings, regions, and local features. Deep learning is a new but powerful method which can be used for feature extraction from raw data. To automatically extract rich features, transfer learning which employs the existing deep learning models (pre-trained on very large datasets) on new data can be leverage. This helps the researchers discover the high-level abstractions (meaning) even with the limited data. Deep neural networks are also used for social media analysis for crisis response and disaster management [18].
- **Machine Learning** Machine learning is the process of discovering, predicting, and learning from data or through experiences. In disaster and emergency management, data comes from a variety of sources and different kinds of knowledge may be needed by different users. Thus, machine learning and data mining algorithms may involve different tasks including clustering, association rule mining, trend analysis, and classification, to name a few. The major challenges in this phase include the following [10]: (1) disaster data is big and heterogeneous

in nature, (2) it includes noisy and uncertain information, (3) its distribution is skewed and imbalanced, and (4) the disaster applications are domain specific. Therefore, conventional machine learning techniques cannot easily handle such complex, multimodal data in an efficient manner.

It has been shown that multimedia data mining can handle the challenges in disaster management applications. In [19], a disaster-based video concept detection approach is proposed using weighted discretization multiple correspondence analysis (MCA) which enhances the correlation between the targets and the feature values. MCA is also utilized in another study [28] where a hierarchical image classification algorithm is proposed for disaster response situations.

Nowadays, deep learning has shown its great potential in different applications such as computer vision, natural language processing, and speech processing. In recent years, the advantages of utilizing neural networks [25] and deep learning [20, 24] in disaster management systems are discussed. For instance, Song et al. [24] present a system which detects the human behavior and mobility in emergency situations using deep learning. Machine learning and deep learning also play important roles in robotics [14]. In these days, trained robots can save lives and prevent disasters, but the main challenge remains in analyzing the data in real-time for disaster recovery situations [2].

- **Post-processing** The results generated by knowledge acquisition algorithms (e.g., neural network, decision tree, etc.) need to be post-processed to be useful and appropriate for the user and customer views. Post-processing, an important component of data mining, includes knowledge filtering and pruning, interpretation, evaluation, and integration. As a disaster happens, a huge amount of data (e.g., social media, videos, mobile data, reports, documents, etc.) is generated and then analyzed by the data mining techniques. Thereafter, efficient techniques are required to filter out and summarize the results. In addition, the results from multimodal sources need to be integrated and fused [28]. Similar to preprocessing, post-processing plays a critical role in FPHLM [27].
- **Tools and Applications** The uses of tools and applications in disaster management have been evolving from time to time. This can result from technological advancements and lifestyle changes, but the objectives remain the same. The goal is to help essential personnel to be quickly aware of the current disaster situation, efficiently respond to different requirements, and make the optimal decision in the shortest time. In Table 10.1, several tools and applications are presented and categorized by different characteristics.

10.4 A Case Study: 3D Storm Surge Impact Animation

Visualization methods serve to assess complex disaster events in an interactive manner. The 3D Storm Surge Impact Animation [21] is a visualization environment which uses GIS data to simulate the impacts of a storm surge near real-world coastal

Table 10.1 Tools and applications in disaster management system

Papers	Applications	Data types
Tran et al. [26]	Mobile GIS Crowdsourcing	Local knowledge Geographic information
Montoya [17]	Mobile GIS	Remote sensing information Geographic information Digital videos
Degrossi et al. [4]	Mobile GIS Crowdsourcing	Volunteered geographic information Text Images Videos
Carley et al. [1]	Mobile GIS Crowdsourcing	Social media data Volunteered geographic information Volunteered geographic information
Song et al. [24]	Mobile Mobile	GPS Disaster situation report
Yang et al. [29]	Mobile Crowdsourcing	Text Images Videos Disaster situation report

areas. More specifically, the 3D model is a visual representation of South Miami Beach. The animation is built using a 3D game engine, Unity,[1] which opens many possibilities for its deployment, namely, its ability for cross-platform support for the popular operating systems (e.g., Windows, Mac, and Linux). The use of the GIS data makes it possible for the model of the city to accurately match a real-life environment. To produce the terrain that depicts an accurate visualization of South Miami Beach, the Light Detection and Ranging (LiDAR) downloaded from The National Oceanic and Atmospheric Administration (NOAA[2]) website is used. LiDAR is a remote sensing technology which produces point clouds, and each point represents the elevation at a specific location. From the LiDAR point cloud data, the bare-earth points can be extracted and a digital elevation model (DEM) can be created, which is a grayscale raster where each pixel contains the height information for each location. Unity makes it easy to connect the animation with the Integrated Computer Augmented Virtual Environment (I-CAVE[3]), a visualization and research facility, ideal for presenting 3D virtual environments. I-CAVE gives users the capability of navigating through the terrain as an immersive experience.

[1] https://unity3d.com/.

[2] https://coast.noaa.gov/dataviewer/.

[3] http://icave.fiu.edu/.

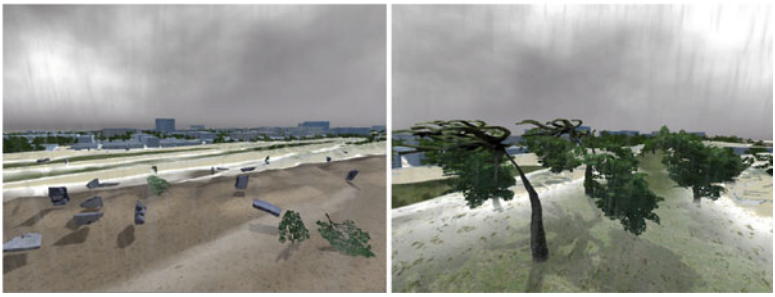


Fig. 10.2 Debris are produced by broken buildings and trees. The model includes different types of debris: pieces of broken wood, building rubble, and torn tree branches, which are affected by the wind and the surge of the waves

The resulting visualization, as shown in Fig. 10.2, can be used in studying the effects of storm surge in a real-world environment.

The following is a list of different parameters that the user can change to create different types of storm surge scenarios.

- Wind Scale: When the user selects the category of the hurricane, the intensity of the wind is set according to the Saffir-Simpson hurricane wind scale.[4] Unity's built-in wind zone component makes it easy to set the parameters for the changes in the wind.
- Rain Intensity: The rain animation was created using the Unity Particle System. A user is able to change the parameters that affect the visual of the rain. Such parameters include the intensity of the rain and the force factor of the wind on the rain.
- Wave Intensity: Waves are able to flood the land, smash the trees and infrastructures, and carry scattered debris to multiple directions.
- Debris Properties: A parameter that a user can set for debris is the average weight which determines how much the scattered pieces can move according to the force of the wind and how much damage it can produce when it hits a building.
- Tree Bend & Break Factor: The tree bends according to the effects of the wind. Under certain conditions, the wind can be strong enough to break a branch from a tree.

10.5 Conclusion and Future Directions

With the advancement in multimedia and data mining techniques, as well as the proliferation of smart technologies including mobile, wearable devices, and big data, disaster management systems have become more intelligent and efficient. This

[4] http://www.nhc.noaa.gov/aboutsshws.php.

paper summarizes the state-of-the-art techniques in multimedia data management for disaster situation awareness. Specifically, it discusses how disaster data are obtained from various sources such as social media, sensors, and videos. Multimedia data analysis for disaster management, including new techniques of preprocessing, feature analysis, machine learning, and post-processing, is presented. In addition, several tools and applications in this area are introduced. Finally, a case study is presented to introduce a 3D storm surge impact animation. Despite the great potential of multimedia data management, there are very few approaches leveraging it in disaster recovery systems.

It is foreseeable that multiple disciplines will intensively work together and be more aggressively engaged in facilitating management in all disaster phases. Tremendous progresses have been made in artificial intelligence (AI), which make robots to be extremely helpful in tasks like searching, rescuing, and inspecting the disaster site without additional casualties. Multi-agent system has also shown promising performance in emergency management tasks. Each agent is independent and versatile and is able to allocate the resource, identify the optimal route, and respond to potential obstacles. All the mentioned characteristics can perfectly fit into any disaster scenario.

Acknowledgements This research is partially supported by NSF CNS-1461926.

References

1. Carley, K.M., Malik, M., Landwehr, P.M., Pfeffer, J., Kowalchuck, M.: Crowd sourcing disaster management: the complex nature of Twitter usage in Padang Indonesia. Saf. Sci. **90**, 48–61 (2016)
2. Christensen, H.I., Okamura, A.M., Mataric, M.J., Kumar, V., Hager, G.D., Choset, H.: Next generation robotics. CoRR. abs/1606.09205 (2016)
3. Conrado, S.P., Neville, K., Woodworth, S., O'Riordan, S.: Managing social media uncertainty to support the decision making process during emergencies. J. Decis. Syst. **25**(sup1), 171–181 (2016)
4. Degrossi, L.C., de Albuquerque, J.P., Fava, M.C., Mendiondo, E.M.: Flood citizen observatory: a crowdsourcing-based approach for flood risk management in Brazil. In: International Conference on Software Engineering and Knowledge Engineering, pp. 570–575 (2014)
5. Erdelj, M., Sekercioglu, A., Natalizio, E.: Wireless sensor networks and multi-UAV systems for natural disaster management. Comput. Netw. **124**, 72–86 (2017)
6. Foresti, G.L., Farinosi, M., Vernier, M.: Situational awareness in smart environments: socio-mobile and sensor data fusion for emergency response to disasters. J. Ambient. Intell. Humaniz. Comput. **6**(2), 239–257 (2015)
7. Goodchild, M.F.: Citizens as sensors: the world of volunteered geography. GeoJournal **69**(4), 211–221 (2007)
8. Granell, C., Ostermann, F.O.: Beyond data collection: objectives and methods of research using VGI and geo-social media for disaster management. Comput. Environ. Urban. Syst. **59**, 231–243 (2016)
9. Grolinger, K., Capretz, M.A.M., Mezghani, E., Exposito, E.: Knowledge as a service framework for disaster data management. In: IEEE Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 313–318 (2013)

10. Hristidis, V., Chen, S.C., Li, T., Luis, S., Deng, Y.: Survey of data management and analysis in disaster situations. J. Syst. Softw. **83**(10), 1701–1714 (2010)
11. Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., Meier, P.: Extracting information nuggets from disaster-related messages in social media. In: International Conference on Information Systems for Crisis Response and Management, pp. 791–800 (2013)
12. Jain, S., Duncan, B.A., Zhang, Y., Zhong, N., Ding, Z.: Real-time social network data mining for predicting the path for a disaster. J. Adv. Inf. Technol. **7**(2), 81–87 (2016)
13. Keim, D., Mansmann, F., Schneidewind, J., Ziegler, H.: Challenges in visual data analysis. In: International Conference on Information Visualisation, pp. 9–16 (2006)
14. Manoochehri, H.E., Jamshidi, K., Monadjemi, A., Shahbazi, H.: Finding curvilinear path features in a layered learning paradigm for humanoid robot using monocular vision. Int. J. Humanoid Rob. **11**(3), 1450023 (2014)
15. Meissner, A., Luckenbach, T., Risse, T., Kirste, T., Kirchner, H.: Design challenges for an integrated disaster management communication and information system. In: IEEE Workshop on Disaster Recovery Networks, vol. 24 (2002)
16. Middleton, S.E., Middleton, L., Modafferi, S.: Real-time crisis mapping of natural disasters using social media. Intell. Syst. **29**(2), 9–17 (2014)
17. Montoya, L.: Geo-data acquisition through mobile GIS and digital video: an urban disaster management perspective. Environ. Model Softw. **18**(10), 869–876 (2003)
18. Nguyen, D.T., Joty, S., Imran, M., Sajjad, H., Mitra, P.: Applications of online deep learning for crisis response using social media information. CoRR. abs/1610.01030, 2–7 (2016)
19. Pouyanfar, S., Chen, S.C.: Semantic concept detection using weighted discretization multiple correspondence analysis for disaster information management. In: IEEE International Conference on Information Reuse and Integration, pp. 556–564 (2016)
20. Pouyanfar, S., Chen, S.C., Shyu, M.L.: An efficient deep residual-inception network for multimedia classification. In: IEEE International Conference on Multimedia and Expo (2017)
21. Presa Reyes, M.E., Chen, S.C.: A 3D virtual environment for storm surge flooding animation. In: IEEE International Conference on Multimedia Big Data, pp. 244–245 (2017)
22. Riley, C.: Natural disasters caused $175 billion in damage in 2016 (2017). http://money.cnn.com/2017/01/04/news/natural-disaster-cost-insurance-2016/index.html. Accessed 10 July 2017
23. Sheng, X., Tang, J., Xiao, X., Xue, G.: Sensing as a service: challenges, solutions and future directions. IEEE Sensors J. **13**(10), 3733–3741 (2013)
24. Song, X., Shibasaki, R., Yuan, N.J., Xie, X., Li, T., Adachi, R.: DeepMob: learning deep knowledge of human emergency behavior and mobility from big and heterogeneous data. ACM Trans. Inf. Syst. **35**(4), 41:1–41:19 (2017)
25. Tian, H., Chen, S.C.: MCA-NN: multiple correspondence analysis based neural network for disaster information detection. In: IEEE International Conference on Multimedia Big Data, pp. 268–275 (2017)
26. Tran, P., Shaw, R., Chantry, G., Norton, J.: GIS and local knowledge in disaster management: a case study of flood risk mapping in Viet Nam. Disasters **33**(1), 152–169 (2009)
27. Yan, Y., Pouyanfar, S., Tian, H., Guan, S., Ha, H.Y., Chen, S.C., Shyu, M.L., Hamid, S.: Domain knowledge assisted data processing for Florida public hurricane loss model. In: IEEE International Conference on Information Reuse and Integration, vol. 3, pp. 441–447 (2016)
28. Yang, Y., Ha, H.Y., Fleites, F., Chen, S.C., Luis, S.: Hierarchical disaster image classification for situation report enhancement. In: IEEE International Conference on Information Reuse and Integration, pp. 181–186 (2011)
29. Yang, Y., Lu, W., Domack, J., Li, T., Chen, S.C., Luis, S., Navlakha, J.K.: MADIS: a multimedia-aided disaster information integration system for emergency management. In: IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 233–241 (2012)
30. Zerger, A., Smith, D.I.: Impediments to using GIS for real-time disaster decision support. Comput. Environ. Urban. Syst. **27**(2), 123–141 (2003)

Chapter 11
Internet of Things: Current Trends and Emerging Prospects

Amartya Sen and Sanjay Madria

Abstract The prevalence of Internet of things (IoT) paradigm has seen the rise of many useful applications in the domains of healthcare, industries, smart city, and so on. However, the paradigm itself is in its dormant stages and still has many open-ended research challenges like interoperability, security and privacy, compliance, and standardization issues which need to be addressed. In this paper, we present a high-level discussion of the overall implications of the IoT paradigm with respect to different application areas and scenarios, domains, and technicalities that needs to be focused for effective incorporation of the IoT concept. Additionally, we also discuss briefly some of the future prospects that can improve the current trends of IoT frameworks.

Keywords Internet of Things · IoT Applications and Challenges · security and risk assessment · overlay networks

11.1 Introduction

The Internet of things (IoT) paradigm and its applications to various domains like industries (Industrial Internet of things or IIoT), healthcare, and smart cities have received considerable attention in the past few years. According to IHS estimates, in 2015, there were 15.4 billion connected devices, and it is projected to rise up to 30.7 billion and 75.4 billion by 2020 and 2025, respectively [19]. In 2016 alone, IDC estimated that there were 28.3 million wearable devices sold and forecasts this value to reach 82.5 million units by 2020 [16]. Given such exponential increase in the number of interconnected devices, the global spending in IoT-related applications was estimated to be around $737 billion in 2016, which will further grow to $1.29 trillion by 2020 [17]. Additionally, the IoT paradigm will contribute about $10 to

A. Sen · S. Madria (✉)
Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, USA
e-mail: asrp6@mst.edu; madrias@mst.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_11

$15 trillion to worldwide GDP growth. Such statistics are not an exaggeration since in 2017 it is estimated that approximately 60% of the global manufacturers will use analytics on data recorded from connected devices to outline actionable insights and develop optimized plans and processes for their workflow [40]. These are some of the staggering information which indicates that the IoT paradigm is growing at an unprecedented rate.

In the recent past, we have seen the maturity of several computing platforms like the cloud (including edge and fog computing) as a way to pool computing resources and offer services in the form of pay-as-you-use models. Further, the developments in hardware efficiency for sensor nodes [33] have also made it easier to own and maintain your wireless sensor networks (WSNs). Additionally, platforms like Sensor Cloud [21] have also made it easier to avail sensing services for users who may not own a WSN. There has also been a growth in other types of sensing devices such as wearable body sensors, RFID tags, and so forth. All these devices and their networks generate a vast amount of data everyday [44]. To make meaningful use of these data streams, we must be able to connect together these isolated islands of devices and their networks and manage them through common interfaces and platforms to perform analytics on the generated data resulting in meaningful and actionable insights applicable across different domains. This is the principal idea behind the IoT paradigm. For example, consider the IIoT services of Michelin's solutions *EFFIFUEL* service whose objective is to reduce the fuel consumptions in truck fleets [18]. They do so by capturing numerous sensing data such as fuel consumption, tire pressure, speed, and geography which are then transmitted to a cloud platform wherein they are analyzed and expert recommendations are made on how to optimize fuel efficiency.

The IoT paradigm is no doubt growing faster than our anticipation, having found roots in application domains like industries, healthcare, and smart cities [41]. Nonetheless, the paradigm and its applications itself are in dormant stages and has to traverse a long way to reach complete maturity. Numerous challenges are to be addressed along with establishment of functional and nonfunctional standards. Thus, our objective here is to discuss on a high level some of the existing trends that have taken place in the domain of IoT related to different application scenarios. Furthermore, we will discuss some of their technical aspects which need to be addressed in order to ensure effective utilization of the IoT paradigm. Finally, we will outline some of the future prospects which can improve the feasibility of IoT-based application scenarios along with introducing the concept of incorporating overlay networks for IoT frameworks which will help in facilitating a user-centric service model, capturing their desired quality of service (QoS) and security preferences.

11.2 IoT Application Scenarios

As mentioned in the previous section, the idea behind IoT and its applications is to interconnect different platforms together and make pragmatic use of the generated data. These networks typically are composed of sensory devices like

sensor nodes, RFID, and wearable body sensors. However, while implementing them for different application scenarios like healthcare or industries, we should not rely on traditional protocols of framework interconnectivity and interoperability. An effective approach requires to evolve implementation models weaving together different aspects of an IoT application scenario along with keeping the users (or consumer) of the services in the center of the service models. These kinds of models will not only require the interconnectivity between networks in the same application scenario but also across different application scenarios. In this regard, we will discuss some of the IoT frameworks belonging to different application scenario.

11.2.1 IoT in Healthcare Applications

IoT enables consistent and remote monitoring for patients in the healthcare domain [15]. Integration of sensing platforms like room sensors, wearable body sensors, and medical equipments like ECG and X-ray machines [37] to Cloud platforms realizes this scenario. Medical professionals can access the data and other analytical services [5] applied on it to take appropriate actions. IoT-based healthcare can provide tailor-made services for individual patients which can be further extended to the premises of their household. Therefore, health monitoring services need not stop once a patient leaves the confines of a hospital. This will be especially useful in activities such as medical therapy and recovery. Additionally, data can also be captured and integrated from the pharmaceutical companies to optimize operations such as logistics, supply chain management, and simulation of drug studies which will keep doctors in the loop and their understanding of patient's conditions to boost the performance of drug development.

11.2.2 IoT in Industries

IoT has widespread application in a lot of industrial sectors ranging from mining operations [42], agricultural improvements [7], to waste water management facilities [26]. Generally these IoT applications are tailor-made based on the needs and organizational policies of the industries where they are incorporated. However, on a higher granularity, they adopt the IoT domains of gathering data from sensory devices, performing analytics on the collected data, and designing intelligent machines and applications that can act upon the results of data analytics. The driving factor in this domain is to make operations cost-efficient but at the same time improve overall productivity. For example, delays and cancellation in the US passenger and cargo aviation cost the industry nearly $11 billion on a yearly basis in terms of maintenance, logistical investments, and so forth. To address this, a joint effort between Accenture and GE aviation was undertaken, called Taleris [13], which was designed using IoT concepts. Taleris is an airline fleet optimization service whose objective is to eliminate avoidable repair costs and minimize delays and disruptions in service availability due to foreseeable conditions. It does so by

collecting data from sensory components accompanied with the aircrafts to monitor the operational conditions and health of different aircraft parts. The collected data is then used to outline optimized predictive maintenance schedules. In doing so, the IoT service can also take into consideration where and when (accounting for the aircraft route) the maintenance operations should be performed in order to minimize disruptions. These kinds of applications can be further improved by integrating it with frameworks that can determine the optimized routes for aerial vehicle trajectories based on different task requests from the ground control [22].

11.2.3 IoT in Disaster Management and Response

Carrying out response and rescue operations in disaster-affected regions is a challenging task since these regions are typically characterized by areas that are devoid of information exchange. Many a times, time-critical rescue operations are to be performed (forest fires) and therefore a need for coordinated efforts exists across a span of geographical region for optimized resource allocation. This is where the concept of IoT-based applications will be beneficial. For example, consider the IoT-based disaster management framework depicted in Fig. 11.1. It consists of physical infrastructure enumerated by IoT devices such as cell phones, wireless sensors, wearable body sensors, RFIDs, gathering and relaying data like CO_2, temperature, user heart rate, and images and video feeds to a management platform like Sensor Cloud. The collected data can be analyzed and queried by regular users to avail information such as k nearest safe zones or evacuation routes. In contrast, rescue workers and first responders can use the Sensor Cloud interface to issue data

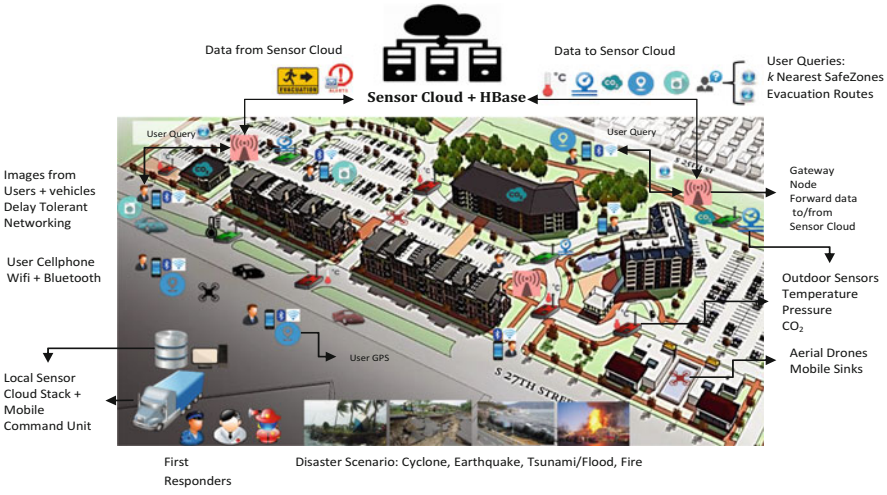


Fig. 11.1 IoT-based disaster management framework

collection tasks across a specified region of interest. The incoming sensory data along with images and video feeds can be used by rescue workers to gauge disaster hit regions they might be entering. Analytics services can also be performed to outline logistical information for the allocation of rescue resources and operations.

These aforementioned IoT application scenarios can also be brought together in order to formulate the foundations of smart cities [43]. Nonetheless, IoT applications and their frameworks should not be just about interconnecting different networks and platforms. It needs to account for the actions that can be performed based on the information obtained from data gathered via the interconnected platforms. In doing so, the users of these frameworks need to be at the center of operations such that user feedbacks and preferences can be immediately tied up to the operational functionalities of the infrastructure for optimal service facilitation.

11.3 Success Factors: Domains and Technical Implications

In the era of IoT advancement, infrastructure providers and manufacturers should not remain disconnected from their consumers. As the IoT paradigm provides a connected ecosystem of users, infrastructure, their behavioral data on which one can perform real-time analytics to suggest and outline actionable insights, there is a need to develop new operational models based on a hybrid concept of developing the products along with providing services and support associated to those products. These kinds of product-service models will essentially need to be user-centric, for example, integrating consumer usage data to product development life cycle management, or performing analytics on the sensory data generated by different devices to schedule predictive maintenance operations to prevent permanent breakdown of devices. The effective realization of the IoT paradigm through the product-service hybrid model depends on successful implementation in some few key domains. At a higher level of granularity, these domains can be summarized as:

- Sensor-driven computing
- Analytics on collected data
- Self-aware applications
- Seamless user integration

Sensor-driven computing is the core building block of IoT implementation across any application scenario. An apprehension of the environment is required to develop and perform any task related to it. This sort of desired outcome can be achieved through the embedded sensing capabilities of the interconnected sensory devices which are growing at a remarkable rate. The sensing capabilities can provide data on temperature, pressure, and CO_2 levels of the surroundings. Whereas new generation wearable body sensors can relay attributes such as heartbeat rate, blood pressure, and so forth. Furthermore, improvements to the hardware implementations and cost of sensory devices [10, 33] are enabling it with possibilities which could not have

been feasible with the traditional sensor nodes running on AA batteries and having limited processing capabilities. We also need to pay attention to efficient techniques of data collection algorithm which will save energy and operational cost in IoT environments [20] and incorporate novel techniques of data compression [9] that will improve the bandwidth consumption and be able to convey more information in a packet size as compared to the traditional scenario.

Data analytics is the key feature that helps in converting the perception provided by the sensory devices about its users and the environment into actionable insight. It is the primary component of the *service* part in the product-service hybrid model. Data analytics can generate actionable tasks like processing the sensory data from the equipment of manufacturing machinery and forecasting predictive maintenance scheduling. These kinds of analytics on sensor-driven computing have been put to practice, for example, *Caterpillar* is using industrial analytics on their dealer's sensory information originating from machines and engines, to give them feedbacks which will help in proactive engagement of any likelihood of operational failure [31]. Similarly, companies like Virtual Radiologic Corp. (vRad) is providing analytics services in the healthcare scenario by collecting and interpreting data from X-ray and MRI [37]. The role of analytics in this ecosystem is of primary importance as it cannot only suggest actionable tasks but eventually shape the public opinion about the products and devices that make up the infrastructure of an IoT framework. Although, as required in some time critical application scenarios, generating relevant actionable tasks depend upon analytics to be performed in (hard or soft) real-time. Furthermore, with the exponential rise in the number of interconnected devices, the amount of data being generated every minute can be overwhelming [35]. Therefore, traditional models of data analytics need to be revised and made more optimized to be incorporated in the IoT scenario [44].

As we traverse toward the end of the maturity spectrum for the IoT paradigm and its related applications, the future lies in the prospect of developing self-aware intelligent applications. In this regard, applications and machines in the future should be able to integrate the analytics outcome and user preferences to their product life cycle in order to perform autonomous improvements and upgrades. Applications across different platforms will also be able to interact with each other to make intelligent operational decisions further consolidating the IoT paradigm of global interconnectivity. A dormant example of such a product can be found in the Nest Thermostat which can interact with its users to comprehend and then manage their energy consumptions [23]. Products like these can be made to interact with networks such as smart electric grids in order to optimize the energy consumption of smart cities. Current practices of serving as a medium for machine-to-machine interaction can also be found in applications like Volvo's CareTrack [38] wherein it can generate reports aiding users to optimally manage their truck fleets. Looking ahead, incorporation of intelligent machines and applications will enable scenarios such as entire factories operating based on interaction between different machines with little to no human oversight thereby boosting productivity. These kinds of autonomous self-aware intelligent machines and applications will be able to launch tasks by cooperating and organizing among themselves. They will be able to self-

incorporate the feedback gained from analytical services and users to improve their interfaces and operational aspects. This in turn will help to reduce their operating costs and improve overall output. Being self-aware will also enable them to prevent accidents and failures during their operations. However, the most challenging aspect among the desired features of intelligent machines and applications will rest in their capability to operate under uncertain and adverse conditions.

Users need to be at the crux of any application scenario that we might develop. Consumer behavior and preferences help shaping and improving the operational standards of products and applications. Hence, the product-service hybrid models that will help in the effective incorporation of the IoT paradigm in different application scenarios need to be user-centric. In this regard, one must be able to take into consideration a user's preferences in terms of their quality of service and experience (QoS and QoE) [12] as well as security requirements [24, 34]. Currently, works mostly address the inclusion of QoS requirements in their IoT application scenarios to improve the overall performance. Some introductory approaches also discuss the decomposition of security requirements and address it across different layers of their IoT applications [29]. However, these incorporations need to be performed from the user's perspective. Optimization algorithms should be developed to address different aspects like operational costs, performance, security requirements, and implementations via the computed (and requested) feedbacks which must be dynamically incorporated by machines and applications in near real-time. These kinds of approaches will further consolidate the concepts of intelligent machines and applications to tailor themselves autonomously as per user requirements.

Nonetheless, addressing the success of these aforementioned domains will be challenging as some of the technical aspects of traditional computing platforms should be changed such that it can be implemented in the IoT ecosystem. We outline two such areas—networking and interoperability, cybersecurity and risk assessment in the following subsections.

11.3.1 Networking and Interoperability

A typical IoT-based application is composed of numerous devices, each having their own unique architecture, software, and hardware specifications. The interconnectivity between these devices and their networks in such a heterogeneous environment is not a trivial task [32]. Networking and interoperability is one of the primary challenges that need to be addressed in order to make strides of progress toward the maturity of IoT-based applications. Dynamic discovery of participating devices, providing resilient services, autonomous service negotiation, and facilitation are some of the key IoT features that are desired and challenged by the lack of standardized networking and interoperability techniques [6]. In a practical scenario, it is safe to assume that in the near future, the hardware and architecture of the participating devices like sensor nodes, RFIDs, and others will not change drastically. As such, the task of networking and facilitating interoperability lies in developing efficient software for these devices, designing frameworks and networking protocols that

will help bridge the gap. An instantiation of ongoing efforts in this regard can be found in AllSeen Alliance's AllJoyn framework [1] which aims to provide developers achieve interoperability between their devices in an IoT ecosystem. The framework facilitates connectivity between device-to-device and the cloud platform which helps to bypass the hassles of transport layer protocol and other heterogeneity challenges that may arise due to device brands, platforms, and operating systems. Another beneficial feature of the AllJoyn framework is in being an open source framework which currently has more than 180 contributing technology partners like LG, Microsoft, and Qualcomm. Other notable efforts lies in the proposed (yet dormant) information-centric networking (ICN) [2] protocol which is about retrieving information and content based on specific naming conventions (instead of IP addresses) which ignores data origination servers and distributed channels. In doing so, the ICN protocol supports in-network caching and replication which will benefit the plethora of resource-constrained devices that partake in a typical IoT application and will allow for the incorporation of content-based security policies which are essentially energy efficient in contrast to their traditional counter parts. This kind of protocol will also benefit data dissemination and networking tasks as it will disregard the heterogeneity of different interconnected platform and their individual traditional networking protocols.

Another effort in addressing IoT-based networking challenges can be found in software-defined networking (SDN) protocol [3, 25] which decouples the network control from packet forwarding and is directly programmable to adjust dynamically based on rate of flow and services. This sort of feature will be useful given the rate at which data is sensed and transmitted in an IoT environment. However, these solutions are in their dormant stages, and a lot of research issues still need to be addressed. We have to consider network management policies (centralized vs. decentralized) with respect to different IoT frameworks used in smart cities, healthcare, or industries. Additionally, issues such as scheduling and link selection also needs to be addressed [11] to optimize attributes like throughput and performance along with reducing the operational costs.

11.3.2 Cybersecurity and Risk Assessment

Cybersecurity and risk assessment is another challenging aspect which needs much attention in the domain of an IoT framework. The challenges essentially arise due to the interconnectivity between vast number of heterogeneous devices, all of which may have different hardware and software specifications. For example, security threats and countermeasures that are applicable to handheld devices are not the same for sensor nodes or wearable body sensors. Additionally, the strength of security requirements varies based on the nature of services provided by these devices and their networks. Therefore, it is not feasible to design universally applicable security measures and policies across an entire IoT framework. Furthermore, certain IoT framework's infrastructures are also composed of devices which may have physical impact like smart electric grid, conveyor belts, and programmable units in the

industry. Being able to exploit or cause malfunction of these devices can have much more dire consequences than traditional cybersecurity outcomes like loss or leakage of data [39].

The IoT frameworks encompass several different layers like infrastructure, services, sensing, and communicates data over wireless and wired mediums. Therefore, the security requirements also span across multiple fields like ensuring confidentiality, integrity, and availability of data. Security requirements also extend to data (user) privacy and anonymity, trust, non-repudiation, authentication, and access control [14]. One way to address all of these security requirements is to individually set up countermeasures across different layers of an IoT framework's protocol stack, for example, physical/MAC, networking, routing, and application layer. Currently, each of these layers follows a set of protocols to realize the overall framework architecture. For example, the constrained application protocol (CoAP) [4] of the application layer, used for interoperability purposes which is coherent with the representation state transfer architecture of the web. The CoAP protocol enables IoT sensing devices to interact with current Internet applications without the need of a specialized translation methodology. Securing CoAP will ensure the security requirements associated with functionalities that are realized by this protocol. Nonetheless, the protocol stack way for addressing security in IoT frameworks is not yet foolproof. There still exist several open-ended research challenges that need to be addressed. Revisiting the CoAP security instantiation which is bounded by the Datagram Transport Layer Security (DTLS) [27], this scheme has its own limitations since it requires to perform a handshake for authentication and key agreement (ECC public key cryptography) purposes with the sensing devices, much of which are resource constrained.

Additionally, security policies and protocols need not remain static throughout the lifetime of an IoT framework. In other words, service requirements may require to give more emphasis on quality of service rather than security for a given instance [24]. In such cases, tradeoffs are required between QoS and security protocols to enable the framework's or user's service demands. This becomes more challenging because to accommodate, for example, the increased QoS factors, we need to reduce the strength of the security protocols, like switch to a lightweight encryption scheme. This sort of action digresses from a framework's initial security requirements estimated by performing risk assessment. Furthermore, different components and users of an IoT framework may have different security requirements. Security requirements may also change with context, evolving threat models and use-case scenarios. This necessitates the concepts of variable and adaptive security requirements, and its implementation is something that also needs to be addressed in the IoT paradigm. Currently in this regard, initial outlines for the concept of adaptive security management have been demonstrated for E-health applications utilizing IoT paradigm [29].

However, before optimal security policies can be designed and applied, one should be able to assess the risks and threats that an IoT framework's infrastructure may be vulnerable to. As such, risk assessment is an imperative step that needs to be performed. However, the task becomes challenging considering the heterogeneity

of the involved infrastructure and the likelihood of scenarios where devices may join and leave the framework from time to time. In this regard, one must be able to account for the logical relationship between the cross-platform devices and how exploitations in one platform may affect the other platforms [30].

11.4 Future Prospects of IoT

IoT-based frameworks are the future of computing platforms that will revolutionize the way we communicate and how services will be carried out. In addition to optimizing the way things are implemented in the traditional domains like industries, healthcare, and smart cities, a lot of scope also lies in IoT's application to promising upcoming sectors like management of renewable sources of energy and its infrastructure. Furthermore, as the analytics on data generated from all the interconnected devices will be used to design more efficient and actionable insights, it can also be incorporated to weave realities and better assist users in this ecosystem of automation. To elaborate further, inclusion of the new technologies like artificial reality (AR) [8] and virtual reality (VR) [36] to IoT-based frameworks can contribute manifolds to supplant the user-centric model which is a necessity in the product-service hybrid framework for IoT applications. For example, in the domain of IoT-based healthcare framework, data from patients' wearable body sensors can be used to monitor their conditions remotely. However, this can be taken a step further and data can be integrated with AR devices to aid users perform basic first aid operations like giving CPR or using devices such as the defibrillator. In an event of emergency and lack of immediate attention from a medical professional, such actions can save a life along with ensuring that the actions are still guided without any risks. Similarly, if we consider the domain of IoT-based applications for disaster management, incoming raw sensory data along with images and video feeds can be used by rescue workers with a virtual reality (AR) platform to gauge disaster hit regions they might be entering. These kinds of incorporations will reduce the uncertainties of entering inside adverse conditions such as buildings on fire and so forth.

11.4.1 Overlay Networks Connecting IoT Devices and User Experience

With the exponential projected increase of interconnected devices, IoT frameworks will be characterized by having devices with redundant sensory ranges. To elaborate this, consider a data collection task from a region consisting of N number of devices. In a traditional approach, such a task will activate all N devices in the region. Although, if n ($n \subset N$) devices are sufficient to encompass the region of interest for

the task, this will make activating all the devices inefficient. A better approach will be to selectively activate certain number of devices such that it covers the region of interest and is able to facilitate the task. This will leave remaining devices to service other tasks in the same region. This kind of practice will also be beneficial in terms of conserving energy of these IoT devices which may be resource constrained as a result of nature of the device (wireless sensor nodes) or circumstances (lack of power in disaster hit regions to recharge or replace batteries). It will also boost the overall throughput and performance of the applications since multiple sensory-driven tasks can be carried out concurrently.

Furthermore, selection of IoT devices should not solely be guided by service parameters (region of interest or duration), but it also needs to account for metrics such as user's Quality of Service (QoS), Quality of Experience (QoE), and security preferences. This kind of outlook will also contribute to the user-centric phenomena for product-service hybrid models for IoT framework. QoS metrics can be measured in terms of network parameters like response time, throughput, packet loss, and so on. However, these kinds of QoS measure do not resonate equivocally with general users. As such, QoS measurement can be estimated by allowing users to express their Quality of Experience. QoE is more subjective in contrast to QoS measurement involving parameters like—given a time window, how does a user feel about a service (emotion), what is the end goal of the user by using this service (objective), and why does a user avail this type of service (incentive). By having users express these QoE metrics, one can measure the desired QoS metrics given the context and time window of the service. Such estimations can be done by either translating the qualitative user QoE responses to quantitative scales or using quantitative measures such as heart rate relayed from body sensors or a hybrid method that integrates both. Another piece in the puzzle for user-centric frameworks should also be able to encompass a user's security preferences as the accumulated data in IoT environments associates directly to a user's personal use and immediate surroundings. For example, consider the case of Nest thermostats or smart grid meters being able to transmit sensor information about the user's energy consumption behavior in their household. Furthermore, in the healthcare domain, majority of the sensed data is coming from wearable body sensors and so forth. If acted upon maliciously, these data can be used to monitor a user's behavior encroaching their privacy or tampering it to provide misleading information which can lead to dire consequences. Hence, service facilitation also requires users to be able to specify their desired security and privacy levels along with service parameters and QoE or QoS requirements. Overlay networks can also help in this regard because it provides the capabilities to selectively choose IoT devices that can satisfy the user requirements and also meet the framework's objectives to carry out the task efficiently.

In the formulation of overlay networks within IoTs, different sensing nodes will form an integral part of the overlays by providing data related to different sensing activities within IoT ecosystem to facilitate multiple tasks in a given time window across a geographical region. The participating IoT devices in multiple networks might be resource constrained (energy or processing power), and therefore, one

cannot incorporate all the desired QoE/QoS, security requirements and dispense the actual services (sensing). Hence, a good balance needs to be achieved between services dispensed and the security availed, adhering to the requirements of user QoE/QoS (in accordance to the user-centric model). In dense deployment of IoT networks, one can use redundant nodes to operate during the same time period to service multiple users, for example, during busy morning walks along a trail route. To generalize this concept, individual IoT nodes belonging to the same or different networks can be selected to form overlay networks on the fly to facilitate service requests [28] from different users spanning different geographical regions, thereby facilitating dedicated virtual infrastructure for users without the need to wait for a task to complete.

In the formulation of overlays in an IoT environment, we can divide service provisioning tasks into two different but correlated modules: *Performance* and *Security*. The variable security policies if addressed across multiple IoT platforms necessitate the need for a layer that will be able to translate the heterogeneous security policies. Although, applying security measures comes at the cost of network's QoS parameters (bandwidth, performance, accuracy, and precision). Users may express security requirements in terms of a lower bound and a security range [34]. The security ranges will be influenced by the nature of the IoT service (e.g., real-time vs. offline data requests) and its effect on network QoS parameters. The minimum security requirements could be in the directions of having accurate data, available with some level of encryption. However, for real-time data access, there will be a higher priority on aspects such as data stream rate as a result of which users may opt for a lightweight encryption scheme with alternate data packet authentication. If users are more concerned about their privacy, it will make them opt for heavyweight encryption schemes and authenticating every data packet. Therefore, a user's desired tradeoffs between network QoS parameters and security requirements can be specified using a security range with a minimum bound. We need to take these user requirements and map them into the available physical infrastructure and security policy encapsulated in the network. Thus, we envision that overlay networks will be formed using a two-tiered decision-making framework: (1) Tier 1, dynamically form overlays that provide optimal services to a user in terms of their QoS requirements, and (2) Tier 2, identify and output the overlays that satisfy a user's security preferences, and if not, users may want to renegotiate the prespecified security preferences.

The formulation of such overlay networks in Tier 1 can be achieved using concepts of Markov decision process, a finite state automaton having four tuples:

$$\text{MDP} = \{S, a, P_a(s_1, s_2), R_a(s_1, s_2)\} \tag{11.1}$$

where, S is a finite set of states of IoT devices belonging to the same or different networks present in the selected region of interest; a is the finite set of actions such as add a device, delete a device, put a device to sleep (active) mode, or change its functionality from one to another. Execution of certain action results in a transition of state which is denoted with a probability $P_a(s_1, s_2)$. Some factors

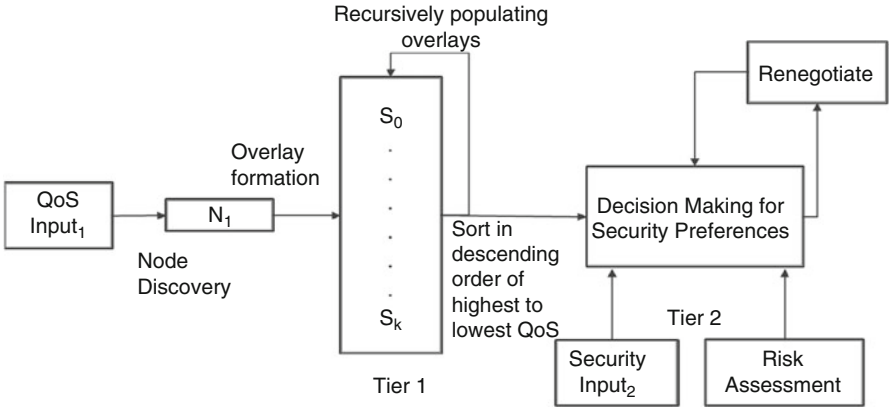


Fig. 11.2 Depicts the framework for formation of overlays for wireless network adhering to user's QoS and security requirements

in the overlay scenario that may affect P_a are remaining energy of the devices, their sensing capabilities and available processing power. The reward or expected outcome of a transition is denoted by $R_a(s_1, s_2)$. In the overlay scenario, rewards will be instantiated (in terms of QoS) to minimize the number of devices required to meet the sensing coverage, to extend the lifetime and continuity of the service, and so forth. The reward criteria for formulated overlays can have a qualitative scoring such as good, bad, and worse. This qualitative scoring will be converted to quantitative scores based on the organizational policies, and the overlays yielding maximum rewards will be shortlisted in descending order by Tier 1 of the decision-making framework (a high-level abstraction is shown in Fig. 11.2) and passed to Tier 2 for security assessment.

Tier 1 uses the QoS specification as input and performs its operation. First, the framework will start with a node discovery process in which it randomly chooses a sensor node available within the region of interest of the user application/task/query. Starting with this node (let us call it the initiator node), it will check whether the application's service region is satisfied or not. If it is unsatisfied, it will reinitiate the node discovery looking for other nodes that will help increase the coverage of the application's region of interest. In this regard, the decision to whether or not add the new nodes to the overlay of the initiator node is aided by an inference engine and Markov decision processes. The inference engine is composed of the following factors: (1) node's sensing capabilities required for the application to run, (2) remaining energy of the node being considered given the running frequency of the application, (3) node's memory and processing power required by the application, and (4) sensing coverage provided by the node being considered. It is desired to form the overlays with the most optimal number of sensor nodes. The aforementioned factors will correspond to rewards of Markov decision processes. The output of Tier

1 will be the feasible sets of overlays that can serve a user's application, sorted in decreasing order of satisfiability in terms of QoS parameters.

Tier 2 will use the output of Tier 1 along with the user's security preferences provided in the initial input. Additionally, it will use the RA (risk assessment) module [30] to compute the security policies of the formed overlays. This module will assess the feasibility of different known attacks in IoT networks and compute the net threat level to the network in terms of a percentage of confidentiality, integrity, and availability. Although we could estimate the threat level on the overlay's security parameters by estimating the impact of different feasible security attacks on a network in isolation, it would not be sufficient since attacks can be used in conjunction to execute more degenerate attacks. For example, a malware attack can subvert a node which can then legitimately participate in network activities causing attacks such a Sinkhole. Attack graphs in this context will help to depict the logical correlation between different feasible attacks. In case the user's security preferences are not met by any of the formulated overlays, then there is a need to renegotiate the user preferences, either in terms QoS parameters or security requirements or both. Along these lines, preferences on security can be higher than that for QoS requirement. In such cases, the decision-making framework can apply risk assessment before applying QoS assessment. Furthermore, this can be taken one step further by performing tradeoff analysis between performance and security and measuring the output by using utility functions to measure multiple options users can select.

11.5 Conclusion and Looking Ahead

This paper provides a summary of different applications and technical issues in the IoT domain and discusses some open research issues. There is lots of research that need to be addressed in this promising paradigm of IoT. To begin with, compliance and regulatory information needs to be established for different application scenarios. There is also a need for standardization techniques related to the fields of IoT networking, analytical services, security, and risk assessment policies. Traditional protocols for varying tasks such as access control or routing need to be modified appropriately so that they are less resource hungry and more effective. Further, the disruption that will be brought about by the adoption of IoT paradigm to legacy systems and their framework also requires much attention so as to ensure that the transition is graceful and does not result in a breakdown. This will also ensure prolonged life cycle of the IoT devices along with continuous operational guarantees. As the IoT-based applications and their framework mature, one can expect to gain dependable interconnected infrastructures that are resilient to foreseeable failures and can communicate with each other to provide smart tailor-made services to the users by utilizing the capabilities of sensing and control.

References

1. AllJoyn: Alljoyn framework. allseenalliance.org/
2. Amadeo, M., Campolo, C., Quevedo, J., Corujo, D., Molinaro, A., Iera, A., Aguiar, R.L., Vasilakos, A.V.: Information-centric networking for the internet of things: challenges and opportunities. IEEE Netw. **30**(2), 92–100 (2016)
3. Bedhief, I., Kassar, M., Aguili, T.: SDN-based architecture challenging the IoT heterogeneity. In: 2016 3rd Smart Cloud Networks Systems (SCNS), pp. 1–3 (2016)
4. Bormann, C., Castellani, A.P., Shelby, Z.: CoAP: an application protocol for billions of tiny internet nodes. IEEE Internet Comput. **16**(2), 62–67 (2012)
5. Boulton, C.: Apple's new health focus comes at propitious time. Wall Street J. (2014). https://blogs.wsj.com/cio/2014/06/10/apples-new-health-focus-comes-at-propitious-time/
6. Bröring, A., Schmid, S., Schindhelm, C.K., Khelil, A., Käbisch, S., Kramer, D., Phuoc, D.L., Mitic, J., Anicic, D., Teniente, E.: Enabling IoT ecosystems through platform interoperability. IEEE Softw. **34**(1), 54–61 (2017)
7. Bunge, J.: Big data comes to the farm, sowing mistrust: seed makers barrel into technology business. Wall Street J. (2014). https://www.wsj.com/articles/no-headline-available-1393372266
8. Buntz, B.: 10 killer applications of the IoT and augmented reality. Internet of Things Inst. News Anal. (2016). http://www.ioti.com/iot-trends-and-analysis/10-killer-applications-iot-and-augmented-reality
9. Cao, X., Madria, S., Hara, T.: A WSN testbed for Z-order encoding based multi-modal sensor data compression. In: 14th IEEE International Conference on Sensing, Communication, and Networking, SECON 2017, San Diego, CA, pp. 1–2 (2017)
10. Carbone, J.: Expect sensor prices to fall. Digikey (2013)
11. Dhondge, K., Shorey, R., Tew, J.: HOLA: heuristic and opportunistic link selection algorithm for energy efficiency in industrial internet of things (IIoT) systems. In: 2016 8th International Conference on Communication Systems and Networks (COMSNETS), pp. 1–6 (2016)
12. Dong, M., Kimata, T., Sugiura, K., Zettsu, K.: Quality-of-Experience (QoE) in emerging mobile social networks. IEICE Trans. Inf. Syst. **E97.D**(10), 2606–2612 (2014)
13. Etihad airways and taleris implement new technology to predict aircraft maintenance faults, reduce flight delays. BusinessWire (2013)
14. Granjal, J., Monteiro, E., Silva, J.S.: Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun. Surv. Tutorials **17**(3), 1294–1312 (2015)
15. Hossain, M.S., Muhammad, G.: Cloud-assisted industrial internet of things IIoT - enabled framework for health monitoring. Comput. Netw. **101**(C), 192–202 (2016)
16. Idc press release (2016). www.idc.com/getdoc.jsp?containerId=prUS41100116
17. Idc press release (2017). www.idc.com/getdoc.jsp?containerId=prUS42209117
18. Kumar, D.: Step on the pedal of cloud services. Michelin Solutions Press release (2013). CruxialCIO.com
19. Lucero, S.: Complimentary whitepaper: IoT platforms - enabling the internet of things (2016). cdn.ihs.com/www/pdf/enabling-IOT.pdf
20. Luong, N.C., Hoang, D.T., Wang, P., Niyato, D., Kim, D.I., Han, Z.: Data collection and wireless communication in internet of things (IoT) using economic analysis and pricing models: a survey. IEEE Commun. Surv. Tutorials **18**(4), 2546–2590 (2016)
21. Madria, S., Kumar, V., Dalvi, R.: Sensor cloud: a cloud of virtual sensors. IEEE Softw. **31**(2), 70–77 (2014)
22. Mekala, A.R., Madria, S., Linderman, M.: Aerial vehicle trajectory design for task aggregation. In: 16th IEEE International Conference on Mobile Data Management, MDM 2015, Pittsburgh, PA, pp. 319–322 (2015)
23. NEST: Demand response programs will reach nearly $10 billion in annual revenue by 2023. Navigant Research (2014)

24. Nieto, A., Lopez, J.: Security and QoS relationships in mobile platforms. In: 4th FTRA International Conference on Computer Science and Its Applications (CSA 2012), vol. 203, pp. 13–21 (2012)
25. Ogrodowczyk, T., Belter, B., LeClerc, M.: IoT ecosystem over programmable SDN infrastructure for smart city applications. In: 2016 5th European Workshop on Software-Defined Networks (EWSDN), pp. 49–51 (2016)
26. Press release, Accenture to help thames water prove the benefits of smart monitoring capabilities (2014). https://newsroom.accenture.com/industries/utilities/accenture-to-help-thames-water-prove-the-benefits-of-smart-monitoring-capabilities.htm
27. Rescorla, E., Modadugu, N.: DTLS: datagram transport layer security. RFC 4347 (2006)
28. Sarakis, L., Zahariadis, T., Leligou, H.C., Dohler, M.: A framework for service provisioning in virtual sensor networks. EURASIP J. Wirel. Commun. Netw. **2012**(1), 135 (2012)
29. Savola, R.M., Abie, H.: Metrics-driven security objective decomposition for an e-health application with adaptive security management. In: Proceedings of the International Workshop on Adaptive Security, ASPI '13, vol. 6, pp. 6:1–6:8 (2013)
30. Sen, A., Madria, S.: Risk assessment in a sensor cloud framework using attack graphs. IEEE Trans. Serv. Comput. **10**(6), 942–955 (2017)
31. Skipper, G.C.: Predictive maintenance and condition based monitoring (2013). ConstructionEquipment.com
32. Soursos, S., Zarko, I.P., Zwickl, P., Gojmerac, I., Bianchi, G., Carrozzo, G.: Towards the cross-domain interoperability of IoT platforms. In: 2016 European Conference on Networks and Communications (EuCNC), pp. 398–402 (2016)
33. Takahashi, D.: Spansion goes battery-less with tiny 'internet of things' chips (2014). Venturebeat.com
34. Taleb, T., Hadjadj-Aoul, Y.: QoS2: a framework for integrating quality of security with quality of service. Wiley J. Secur. Commun. Netw. **5**(12), 1462–1470 (2012)
35. Terdiman, D.: How GE got on track toward the smartest locomotives ever (2014). cnet.com
36. Vaccari, A.: How virtual reality meets the industrial IoT (2016). wiki.aalto.fi/download/attachments/109392027/How-VR-meets-IIoT.pdf
37. Virtual-Radiologic: future of radiology microsite (2011). vRad.com
38. Volvo: volvo construction equipment website. www.volvoce.com
39. Wagstaff, J.: All at sea: global shipping fleet exposed to hacking threat. The World Economic Forum report: Global Risks 2014 (2014)
40. Whitepaper: IoT and digital transformation: a tale of four industries. digitalistmag.wpengine.netdna-cdn.com/files/2016/03/IDC_IoT_white_paper_Mar2016.pdf
41. Whitmore, A., Agarwal, A., Xu, L.: The internet of things–a survey of topics and trends. Inf. Syst. Front. **17**(2), 261–274 (2015)
42. Wilson, J.: Miners tap into rich seam of internet of things. Financial Times (2014)
43. Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M.: Internet of things for smart cities. IEEE Internet Things J. **1**(1), 22–32 (2014)
44. Zhang, Y., Li, W., Zhou, P., Yang, J., Shi, X.: Big sensor data: a survey. In: 9th IEEE International Conference on Internet and Distributed Computing Systems IDCS, Wuhan, pp. 155–166 (2016)

Chapter 12
Public Key Cryptosystem for Privacy Sensitive Location-Based Services

K. M. Mahesh Kumar and N. R. Sunitha

Abstract Almost every smartphone and wireless devices are equipped with GPS and other location-enabling technologies, which has enabled users to access location-based services, a popular service offered based on the user's geographical location. In order to get a wide range of location-based services like locating nearby friends and locating nearby places/venues or public places (point of interest), users are forced to reveal their actual location; users are left with no option other than compromise location information causing privacy risk. In this paper, we revisited a protocol proposed by Muhammad N. Sakib and Chin-Tser Huang based on ECC concepts for proximity testing to preserve users location privacy. We made suitable modifications to the existing solution to overcome the false negatives in proximity testing and to reduce the unnecessary communication and computation cost. We have suggested an improvement to enable symmetric key exchange between communicating parties which can be used to securely share the location coordinates to calculate the actual distance between communicating parties. Our scheme withstands triangulation attacks and reveals no information about user's exact location to either service providers or communicating parties or attackers, unless it is revealed by the user himself/herself.

Keywords Elliptic curve cryptography · Location based services · Location privacy · Public key cryptosystem

12.1 Introduction

The smartphones and other wireless devices like tablets and PDA have tremendously grown in the last decade in terms of computation capability and variety of application and services they can support. Location-based services (LBS) is one such application which has gained huge popularity over the recent years; LBS

K. M. Mahesh Kumar (✉) · N. R. Sunitha
Department of CSE, Siddaganga Institute of Technology, Tumakuru, Karnataka, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_12

has been powered by the advances in location-enabling technologies like Global Positioning System (GPS), cell tower-based identification, Internet Protocol (IP) address approximation, and Wi-Fi triangulation. Latest survey reveals that LBS is most popular among the users of social networking applications [6] such as geotagging of photos and videos, check-ins, directory services for nearby places, friend finder, etc. LBS application can be categorized into three main types:

1. Point-of-Interest based (PoI)
2. Friend-Finding (FF)
3. People-Discovery (PD)

PoI applications are used by users to locate nearest places like ATMs, bus station, restaurants, etc. Family and friends can be tracked or located using friend-finder applications; people-discovery applications are useful in case of locating and interacting with new people who are total strangers.

Users using LBS applications are forced to provide access to their location information to the application service provider in order to access the service; this compromises the user's location privacy. Major challenge in LBS application is to preserve the user's location privacy.

12.2 Related Work

The main focus of our paper is to address the problem of location proximity, i.e., dealing with the problem of computing whether user $'A'$ is at a certain distance from user $'B'$ or not; the major challenge here is to preserve the user location information of both user $'A'$ and $'B'$ private to each other and the service provider and just reveal only the proximity and not the actual distance between user $'A'$ and $'B'$.

Disclosing only the proximity rather than the distance between user $'A'$ and $'B'$ helps in preventing external attacks like triangulation effectively. Class of solutions which uses proximity-based approach are referred to as privacy-preserving location-proximity (PPLP) [1, 2, 5, 6, 8–10, 12] protocols. There are several ways in which we can achieve location privacy. Several researchers have used k-anonymity [4, 11], where there exist a set of users and the location of the user is indistinguishable. These solutions focus mainly on hiding the identity of the user rather than location coordinates.

In this paper we try to readdress the issue of location privacy in proximity-based services proposed by Muhammad N. Sakib and Chin-Tser Huang in [7]. We retain the elliptic curve-based proximity test solution provided in [7] and try to make it more efficient.

Contributions Our contributions in this paper are as follows:

- We propose an algorithm and steps to sanitize the GPS coordinates to eliminate false negatives for location proximity.

- We reduce communication and computation cost by eliminating the unnecessary message exchanges suggested by authors in [7].
- We suggest steps to share private key among the communicating parties within the proximity range without incurring overhead.

12.3 Background

12.3.1 Testing Proximity of Users by GPS Coordinates Matching

GPS coordinates are a pair of signed floating-point numbers *(±x, ±y)* which represents latitude and longitude values of the location on the surface of the earth. Say we take two real location values, say L_A(13.3268, 77.126) and L_B(13.3267, 77.1180), by looking at the values we can clearly see that there is partial match among the location coordinates, indicating proximity among the coordinates. Refer to Table 12.1 for details about precision values and proximity range.

12.3.2 Distance Calculation Using GPS Coordinates

We can make use of the following equation to compute the distance between two location coordinates in kilometers.

$$\begin{aligned} \text{Distance} = \text{acos}(&\cos(\text{radians}(90 - \text{lat1})) * \cos(\text{radians}(90 - \text{lat2})) \\ &+ \sin(\text{radians}(90 - \text{lat1})) * \sin(\text{radians}(90 - \text{lon2})) \\ &* \cos(\text{radians}(\text{lat1} - \text{lat2}))) * 6371 \end{aligned} \tag{12.1}$$

Example: distance between say L_A(13.3268, 77.126) and L_B(13.3267, 77.1180) using Eq. (12.1) is 876 m.

Table 12.1 Various precision values and corresponding distance ranges

Decimal places	Decimal degrees	N/S or E/W distance at equator	E/W distance at 45 N/S
5	0.00001	1.1132 m	787.1 mm
4	0.0001	11.132 m	7.871 m
3	0.001	111.32 m	78.71 m
2	0.01	1.1132 km	787.1 m
1	0.1	11.132 km	7.871 km
0	1.0	111.32 km	78.71 km

12.3.3 Elliptic Curve Basics

Elliptic curve cryptography is a public key cryptosystem. Generally, an elliptic curve is defined over a finite field consisting of finite points satisfying the below equation:

$$y^2 = x^3 + ax + b \tag{12.2}$$

The equation will be defined over a large finite field denoted by prime number P. Elliptic curve contains numerous points satisfying the elliptic curve along with a special point called point at infinity (Θ).

The following operations are possible on an elliptic curve:

- *Point addition:* adding two points on the curve results in a third point which satisfies the curve.
- *Point multiplication:* multiplying a point on the curve with a scalar (integer) value results in a point which satisfies the curve (i.e., repeated addition of given point, also referred to as point doubling).

12.4 Proposed Work

In this section we revisit the proposed work of Muhammad N. Sakib and Chin-Tser Huang in [7] and propose an algorithm to minimize the false negative results of the existing solution and to reduce the communication and computation overhead for proximity test plus suggest an improvement of private key exchange. The proposed work is as shown in Fig. 12.1 and is divided into two parts:

1. Proximity test
2. ECDHE private key exchange

Algorithm-1	Sanitization of GPS coordinates
Purpose:	Avoids false negatives by sanitizing GPS coordinates
Input:	Latitude or longitude value in floating-point format $\pm abc.xyz$
Output:	Latitude or longitude value in floating-point format $\pm abc.000$ or $\pm abd.000$ or $\pm abc.xyz$
Steps:	
if	$'xyz' == '99*'$ (* indicate any digit between 0 and 9)
return:	$\pm abd.000$ (where $\|abd = abc + 1\|$)
else if	$'xyz' == '00*'$
return:	$\pm abc.000$
else	
return:	$\pm abc.xyz$

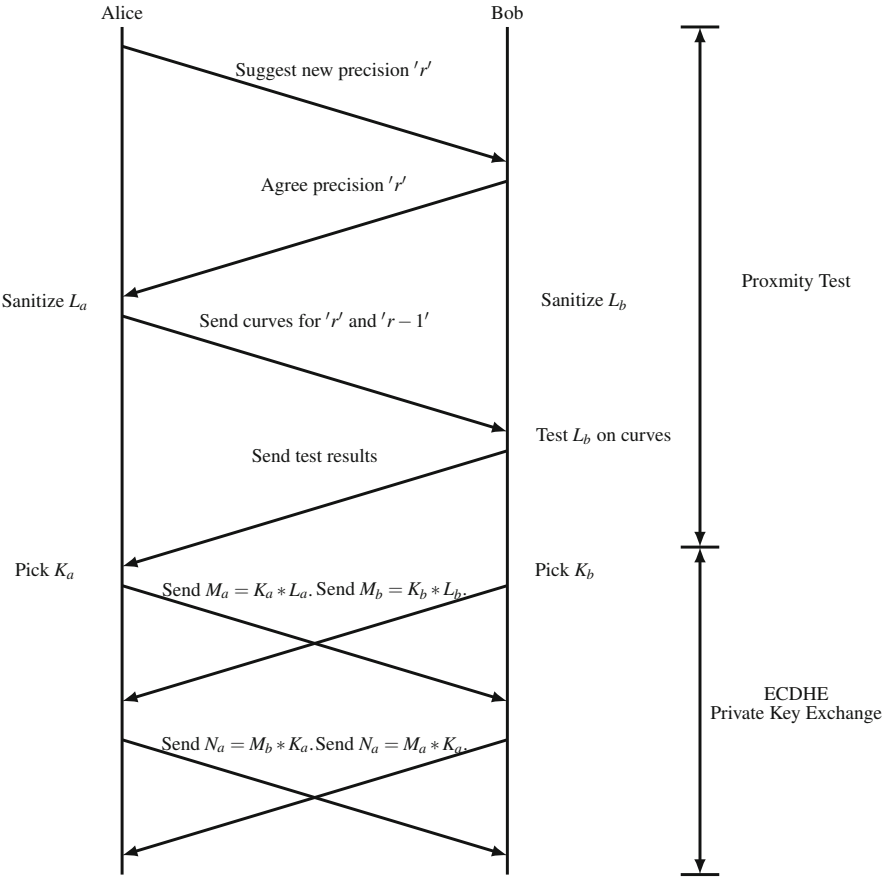


Fig. 12.1 Proximity test and ECDHE private key exchange between Alice and Bob

12.4.1 Proximity Test

12.4.1.1 Decimal Precision Agreement (Optional)

To begin with, Alice and Bob can agree upon a decimal precision value based upon their proximity search requirement; this is an optional step; we can avoid this if the default value for decimal precision value r is set (generally we use $r = 2$).

12.4.1.2 Sanitization of GPS Coordinates

Sanitization of GPS coordinates is done at both ends of communication, i.e., Alice and Bob sanitize their coordinates independently using the agreed precision value

r and $r-1$ using Algorithm-1, e.g., if the agreed precision value $r = 2$, then the coordinate value $L_A(13.3268, 77.126)$ becomes $L_A(13.32, 77.12)$, since elliptic curves do not deal with floating points, we convert it to whole number by removing decimal points. Final coordinates after sanitization will be $L_A(1332, 7712)$ for $r = 2$ and $L_A(133, 771)$ for precision $r - 1 = 1$.

At Bob's end we introduce additional sanitization steps for precision $r - 1 = 1$ as follows:

- case 1: $L_B(x, y-1)$ i.e., (original,low)
- case 2: $L_B(x, y+1)$ i.e., (original,high)
- case 3: $L_B(x-1, y)$ i.e., (low,original)
- case 4: $L_B(x+1, y)$ i.e., (high,original)
- case 5: $L_B(x-1, y-1)$ i.e., (low,low)
- case 6: $L_B(x-1, y+1)$ i.e., (low,high)
- case 7: $L_B(x+1, y-1)$ i.e., (high,low)
- case 8: $L_B(x+1, y+1)$ i.e., (high,high)

For example, $L_B(133, 771)$ for precision $r - 1 = 1$ becomes as follows:

- case 1: $L_B(133, 770)$
- case 2: $L_B(133, 772)$
- case 3: $L_B(132, 771)$
- case 4: $L_B(134, 771)$
- case 5: $L_B(132, 770)$
- case 6: $L_B(132, 772)$
- case 7: $L_B(134, 770)$
- case 8: $L_B(134, 772)$

12.4.1.3 Elliptic Curve Generation

Using Eq. (12.2), coefficient a, large prime number p, and sanitized values, Alice generates elliptic curves for precision value r and $r-1$ as follows:

$$b_i = (y_i^2 - x_i^3 - ax_i) \bmod p \tag{12.3}$$

Alice then forwards the curve parameters p, a, b_r, and b_{r-1} to Bob.

12.4.1.4 Elliptic Curve Evaluation

Bob upon receiving the elliptic curve parameters p, a, b_r, and b_{r-1} verifies his sanitized value $L_b(x_b, y_b)$ with precision r by substituting values into Eq. (12.2). Upon successful verification, he replies back with a positive result, else he continues with verification with precision $r - 1$ by substituting values into Eq. (12.2). If the

test is successful, he replies back with a positive result, else he continues with verification of $L_b(x_b, y_b - 1)$, $L_b(x_b, y_b + 1)$, $L_b(x_b - 1, y_b)$, $L_b(x_b + 1, y_b)$, $L_b(x_b - 1, y_b - 1)$, $L_b(x_b - 1, y_b + 1)$, $L_b(x_b + 1, y_b - 1)$, and $L_b(x_b + 1, y_b + 1)$ by substituting values into Eq. (12.2), one set of coordinates at a time in case of success, he replies back with a positive result and discontinues the test. If no match found, he replies back with negative result. These test results are sufficient for proximity testing.

We demonstrate our technique using sample values as follows:

12.4.2 ECDHE Private Key (Symmetric Key) Exchange

In [7] they have proposed additional steps of ECDHE for verification of location proximity which we feel is an unnecessary burden; instead the same can be used for symmetric key exchange between Alice and Bob. ECDHE symmetric key exchange steps are as follows:

Alice:

- Selects a secret value K_a randomly
- Computes $M_a = K_a * L_a$ using the verified curve and sanitized coordinate L_a
- Sends M_a to Bob

Bob:

- Selects a secret value K_b randomly
- Computes $M_b = K_b * L_b$ using the verified curve and sanitized coordinate L_b
- Sends M_b to Alice (similar to Alice)

Alice:

- Receives M_b from Bob
- Computes $N_a = K_a * M_b = K_a * K_b * L_b$
- Uses N_a as symmetric key

Bob:

- Receives M_a from Alice
- Computes $N_b = K_b * M_a = K_b * K_a * L_a$
- Uses N_b as symmetric key ($N_a = N_b$ if all steps are correct)

Repeat the above steps with different random secrets to arrive at a different secret key when required.

Alice	Bob
$L_a(37.295, 28.135)$	$L_b(37.321, 28.138)$
Sanitize L_a using default precision $r = 2$ and $r - 1 = 1$	Sanitize L_a using default precision $r = 2$ and $r - 1 = 1$
$L_a(3729, 2813)$ for $r = 2$ and $L_a(372, 281)$ for $r - 1 = 1$	$L_b(3732, 2813)$ for $r = 2$ and $L_b(373, 281)$ for $r - 1 = 1$
$a = 5$ and $P = 1{,}000{,}003$ in Eq. (12.3)	
for $r = 2$ $b_2 = 660{,}373$	
for $r = 1$ $b_1 = 598{,}409$	
send $a = 5$, $P = 1{,}000{,}003$, $b_2 = 660{,}373$ *and* $b_1 = 598{,}409$	
	Substitute $a = 5$, $P = 1{,}000{,}003$, $b_2 = 660{,}373$, $L_b(3732, 2813)$ in Eq. (12.2) and evaluate the curve
	$912{,}948 \not\equiv 162{,}264$ (indicates curve does not satisfy)
	Substitute $a = 5$, $P = 1{,}000{,}003$, $b_1 = 598{,}409$, $L_b(373, 281)$ in Eq. (12.2) and evaluate the curve
	$78{,}961 \not\equiv 495{,}235$ (indicates curve does not satisfy)
	Our proposed extension
	Case 1: substitute $a = 5$, $P = 1{,}000{,}003$, $b_1 = 598{,}409$, $L_b(373, 280)$ in Eq. (12.2) and evaluate the curve
	$78{,}400 \not\equiv 495{,}235$ (indicates curve does not satisfy)
	Case 2: substitute $a = 5$, $P = 1{,}000{,}003$, $b_1 = 598{,}409$, $L_b(373, 282)$ in Eq. (12.2) and evaluate the curve
	$79{,}524 \not\equiv 495{,}235$ (indicates curve does not satisfy)
	Case 3: substitute $a = 5$, $P = 1{,}000{,}003$, $b_1 = 598{,}409$, $L_b(372, 281)$ in Eq. (12.2) and evaluate the curve
	$78{,}961 \equiv 78{,}961$ (indicates curve values satisfied for precision 1 for case 3)
$a = 5$, $P = 1{,}000{,}003$, $b_1 = 598{,}409$, $L_a(372, 281)$ in Eq. (12.2)	Send positive result for precision value 1
$78{,}961 \equiv 78{,}961$	Same value at both ends indicate proximity for precision value 1

12.5 Security Analysis

Security of our work relies on the hardness of elliptic curve discrete log problem (ECDLP). Let M is point on the curve and k be a secret integer $N = [k] * M$, given

M and N it is hard to reveal k if the finite field value P is sufficiently large. Please refer [3] for more details about ECDLP.

Our paper has achieved the following four security goals which are listed below (interested readers can refer [7] for proofs and additional information):

1. Alice and Bob can perform proximity verification, in order to know if they are located in a certain distance range.
2. Either Alice or Bob cannot narrow down on each other's single specific location information (i.e., region smaller than the Earth) if they are not in the proximity range. But they get to know they are not within the specified distance.
3. Either Alice or Bob cannot narrow down on each other's single specific location information (i.e., region smaller than the Earth) if they are within the proximity range, except if they are in the eyesight distance.
4. None of the third party (intruder/communication server) can narrow down on single specific region smaller than the Earth where Alice or Bob is located.

Our desired goals are achieved using ECC and the hardness of elliptic curve discrete logarithm problem. Our protocol assumes that both Alice and Bob use the protocol to communicate only about their proximity information. If at all they want to share their location information for calculating distance, they can do so by first exchanging symmetric key using ECDHE private key exchange and exchanging the encrypted coordinates using the symmetric key.

12.6 Conclusion

In our paper we revisited the proposed work of Muhammad N. Sakib and Chin-Tser Huang in [7] and propose an algorithm to minimize the false negative results of the existing solution and to reduce the communication and computation overhead for proximity test plus suggest an improvement of symmetric key exchange for secure communication. Our contributions are as follows:

- We proposed an algorithm plus additional steps to sanitize the GPS coordinates to eliminate false negatives for location proximity.
- We reduced communication and computation cost by eliminating the unnecessary message exchanges suggested by the authors in [7].
- We suggested an improvement to share symmetric key among the communicating parties within the proximity range without adding any overhead to the scheme.

References

1. Freni, D., Vicente, C.R., Mascetti, S., Bettini, C., Jensen, C.S.: Preserving location and absence privacy in geo-social networks. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management - CIKM '10. ACM Press, New York (2010). https://doi.org/10.1145/1871437.1871480

2. Hallgren, P., Ochoa, M., Sabelfeld, A.: InnerCircle: a parallelizable decentralized privacy-preserving location proximity protocol. In: 2015 13th Annual Conference on Privacy, Security and Trust (PST). IEEE, New York (2015). https://doi.org/10.1109/pst.2015.7232947
3. Hankerson, D., Menezes, A.: Elliptic Curve Discrete Logarithm Problem, pp. 397–400. Springer, Boston (2011). https://doi.org/10.1007/978-1-4419-5906-5_246
4. Magkos, E.: Cryptographic approaches for privacy preservation in location-based services: a survey. Int. J. Inf. Technol. Syst. Approach **4**(2), 48–69 (2011). http://dx.doi.org/10.4018/jitsa.2011070104
5. Mascetti, S., Freni, D., Bettini, C., Wang, X.S., Jajodia, S.: Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies. VLDB J. **20**(4), 541–566 (2010). https://doi.org/10.1007/s00778-010-0213-7
6. Narayanan, A., Thiagarajan, N., Lakhani, M., Hamburg, M., Boneh, D.: Location privacy via private proximity testing. In: Proceedings of the Network and Distributed System Security Symposium, NDSS 2011, San Diego, CA, 6th February - 9th February 2011 (2011). http://www.isoc.org/isoc/conferences/ndss/11/pdf/1_3.pdf
7. Sakib, M.N., Huang, C.T.: Privacy preserving proximity testing using elliptic curves. In: 2016 26th International Telecommunication Networks and Applications Conference (ITNAC). IEEE, New York (2016). https://doi.org/10.1109/atnac.2016.7878794
8. Šeděnka, J., Gasti, P.: Privacy-preserving distance computation and proximity testing on earth, done right. In: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '14, pp. 99–110. ACM, New York (2014). http://doi.acm.org/10.1145/2590296.2590307
9. Šikšnys, L., Thomsen, J.R., Šaltenis, S., Yiu, M.L., Andersen, O.: A Location Privacy Aware Friend Locator, pp. 405–410. Springer, Berlin (2009). http://dx.doi.org/10.1007/978-3-642-02982-0_29
10. Šikšnys, L., Thomsen, J.R., Šaltenis, S., Yiu, M.L.: Private and flexible proximity detection in mobile social networks. In: 2010 Eleventh International Conference on Mobile Data Management. IEEE, New York (2010). https://doi.org/10.1109/mdm.2010.43
11. Talukder, N., Ahamed, S.I.: Preventing multi-query attack in location-based services. In: Proceedings of the Third ACM Conference on Wireless Network Security, WiSec '10, pp. 25–36. ACM, New York (2010). http://doi.acm.org/10.1145/1741866.1741873
12. Zhong, G., Goldberg, I., Hengartner, U.: Louis, Lester and Pierre: Three Protocols for Location Privacy, pp. 62–76. Springer, Berlin (2007). http://dx.doi.org/10.1007/978-3-540-75551-7_5

Chapter 13
A Study of Contact Durations for Vehicle to Vehicle Communications

Quynh Nguyen and Bhaskar Krishnamachari

Abstract Emerging vehicular networks will take advantage of vehicle to vehicle short-range communications. In such networks the V2V link is active only so long as the two nodes stay within communication range of each other. This contact duration which is a statistical phenomenon affected by the mobility of vehicles on the road network will have a significant impact on the network performance, and thus merits deeper understanding. We undertake an intensive study of contact duration using real trace data from taxis in Shanghai assuming a fixed-range radio. We quantify the aggregate contact duration statistics, as well as the contact duration statistics conditioned on factors such as time, location, and vehicle directions.

Keywords Vehicular networks · Contact duration statistics

13.1 Introduction

Vehicular network (VANETs) has become an emerging area attracting extensive research effort in the past few years. In VANETs, it is challenging to create an end-to-end path between any pair of nodes due to either extremely dynamic and node behavior-dependent mobility or very sparse network architecture [6].

In these opportunistic VANETs, a contact (an encounter) between any pair of nodes is generated when the pair is within communication range of each other. However, nodes in the networks keep moving in their own ways so communication links among mobile nodes are on and off continuously; the pair will no longer be in contact when one node moves out of the other node's vicinity area or its link quality fluctuates. Therefore, there are hardly any existing complete path from sources to destinations, and even discovered complete paths are very unstable and of short-term duration. In those VANETs whose links among nodes are generated highly

Q. Nguyen (✉) · B. Krishnamachari
Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA
e-mail: quynhngu@usc.edu; bkrishna@usc.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_13

dynamically and intermittently, store-carry-forward techniques and opportunistic data dissemination framework are introduced as a way to overcome intermittent connectivity, although relative high delay is sometimes the cost to get information from one source to another destination.

Understanding the contact process and knowing related encounter statistics can improve data distribution demand in terms of either reducing the overall delay, saving relaying bandwidth, local storage buffer and energy cost at intermediate nodes, or increasing data throughput and transfer reliability, etc. A pair of nodes having higher meeting frequency and shorter inter-contact time [1, 9, 10] compared to the others will have better opportunity to transmit data to each other. Any nodes having higher encounter frequency with larger group of neighboring nodes within a shorter time duration can become better candidates as intermediate nodes for relaying data in routing algorithms. Any pair that has contact duration longer than those of other pairs, which is a sign that contact link between them is more stable and data transferred between them, is also a more reliable candidate to reach a given destination successfully.

Among multiple crucial parameters of the contact process, contact duration is a critical parameter [5, 11]. We consider in this work an intensive study about vehicular contact duration based on the data obtained by Shanghai Jiao Tong University (http://wirelesslab.sjtu.edu.cn/taxi_trace_data.html). We make two sets of contributions:

- We analyze the aggregated contact duration distribution based on the real taxi data trace.
- We quantify the contact duration conditioned on different parameters including time, location, and vehicle directions.

The rest of this paper is organized as follows: Sect. 13.2 lists related work; Sect. 13.3 gives an introduction about the dataset and the methodology for the vehicular contact duration study; and Sect. 13.4 shows observed facts and draws conclusions about vehicle contact duration distribution and statistics and impact of time, location, and direction on vehicle contact. And finally, we present a concluding discussion in Sect. 13.5.

13.2 Related Work

Contact duration has become an important parameter applied by multiple researchers in different protocols and techniques to enhance data access, data transfer and network connectivity in general delay tolerant networks as well as vehicular networks. The authors in [3] presented link contact duration-based routing protocol to deliver as many messages as possible within a short time. In [4], PRoPHET, a probabilistic routing protocol for delay-tolerant networks, both inter-meeting time and contact duration are taken into account to compute the delivery probability to improve performance of the proposed routing protocol. Another work applying contact duration and meeting frequency so as to estimate message delivery

probability and present a novel routing algorithm is [8]. X. Zhuo *et al.* also used contact duration to enhance the traditional cooperative caching protocol to improve the performance of data access in DTNs. In [11], the same technique is used to improve data replication for data sharing in DTNs.

Furthermore, there are also other works trying to characterize contact duration patterns in a vehicular network. Y. Li *et al.* [5] has carried out experiments using Beijing and Shanghai traces to study the contact duration characteristics. They concluded that the contact duration obeys an exponential distribution, while beyond a characteristic time point it decays as a power law one. While many works such as this have been utilizing contact duration as a crucial factor to help enhance the performance of their protocols in DTNs and VANETs, our work mainly brings into focus the impact of multiple factors (time, location, and direction) on contact duration distribution after intensively analyzing and studying contact information obtained from the Shanghai taxi trace.

13.3 Dataset Introduction and Approach Methodology

13.3.1 Shanghai Dataset

SUVnet-Trace Data, obtained from SJTU Wireless Sensor Networks Lab (Shanghai Jiao Tong University), includes GPS information of roughly 2400 taxis in Shanghai city from January 31 to February 27. The coverage of the area is 6340 km square. Figure 13.1 shows the total region covered by the Shanghai dataset.

We have studied contact duration for 10 weekdays from Feb 5 to Feb 16 at two different time slots 7 am–9 am and 11 am–1 pm. The first slot is the rush period, while the second one represents the normal hours. After dividing the whole region into 1 km × 1 km grid, we compute densities of all cells and have picked some cells for further investigation. Figure 13.2 shows corresponding maps for three types of areas. We chose Region 1 as a representative for a densely populated residence area including the center of residential district, Shanghai Concert Hall, and Jin Jiang Tower. Region 2 is mainly part of the Shanghai Hongqiao International Airport (A1, A2, A4, A5, A7, and A8); A6 covers the Shanghai Zoo; and A3 and A9 are civil areas. Finally, Region 3 is a sparser residence area (Pengpuxincun and Linfen Road Residential district).

Table 13.1 shows the average density of both time slots (7 am–9 am and 11 am–1 pm) of all areas in corresponding regions over 10 days. The density unit is [number of vehicles/km^2/minute]. We have made the standard assumption that the effective V2V communication radio range is 300 m using 802.11p [7].

From the table, we could see clearly that Region 1 has very high density (ranging between 9.75 and 22.9), including many residential districts, Shanghai Concert Hall, and Jin Jiang Tower. The traffic in Region 3 is much less dense (less than 5). In Region 2, there are almost no traffic in A1, A4, and A7 (less than 1); and A5 is the densest area compared to all other areas (almost 21) because it is the entry of

Fig. 13.1 Shanghai covered region

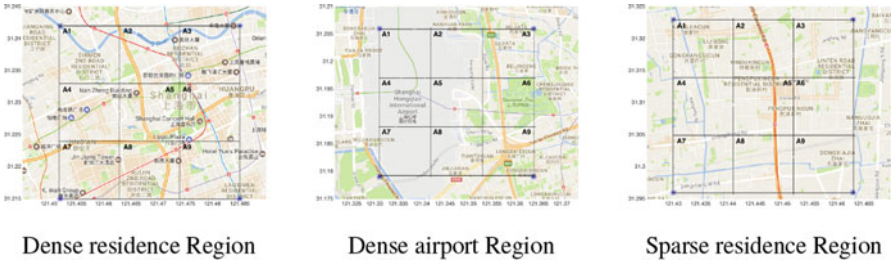


Fig. 13.2 Shanghai data coverage description

Table 13.1 Average density of all areas in each region

Area	Region 1	Region 2	Region 3
A1	10.9238	0.1388	1.2625
A2	12.7233	3.6996	2.6158
A3	9.7842	5.2100	4.9737
A4	19.2858	0.5513	2.5329
A5	22.8887	21.0396	4.5896
A6	17.3758	2.3792	2.1292
A7	16.7371	0.0558	2.9500
A8	15.5487	0.9517	3.9387
A9	15.7575	1.8817	2.4517

Shanghai Hongqiao International Airport Terminal 1 and has all the major airport shuttle stations, airport shopping malls... which is the major places to pick up and drop off for taxis services.

13.3.2 Methodology

There are a few parameters which impact contact duration that we want to analyze, including time, location, and direction. Analyzing the selected 3×3 cells area described in Sect. 13.3.1, we can collect all empirical contact duration giving GPS and timestamp information for each cell within the interesting period. In the study, we used two-sample Kolmogorov-Smirnov test and two-sample Anderson-Darling test which are nonparametric tests for validation if the two samples are generated from the same distribution. Both of them are statistic tools used to test whether two underlying one-dimensional probability distributions differ. However, according to [2] (https://stat.ethz.ch/R-manual/R-patched/library/stats/html/ks.test.html), Anderson-Darling test is used more reliably for a smaller number of samples, while Kolmogorov-Smirnov test is suitable for a larger number of samples. Based on the number of samples (larger or smaller than 10,000), we performed correspondingly suitable validation test to decide if any two empirical contact duration set at different time, location, or direction are generated from the same distribution meaning that they share the same contact duration characteristics. Based on those results, we quantified time, location, and direction impact on contact duration by estimating portion of which they have the same distribution given corresponding conditions.

We will make the contact duration statistics dataset publicly available at http://anrg.usc.edu/www/Downloads/.

13.4 Factual Observations

This section shows the observation of encounter duration statistics varying time, location, and direction conditions and examines their impact.

13.4.1 Time

Figure 13.3 describes the CCDF (complementary cumulative density function) result of aggregated contact duration over 10 days at 7 am and 11 am correspondingly for area 5 of 3 regions. Different colors represent different days. For both periods of week days, we can see that even though there is little variation for larger duration, the encounter duration information at the same area on different days tend

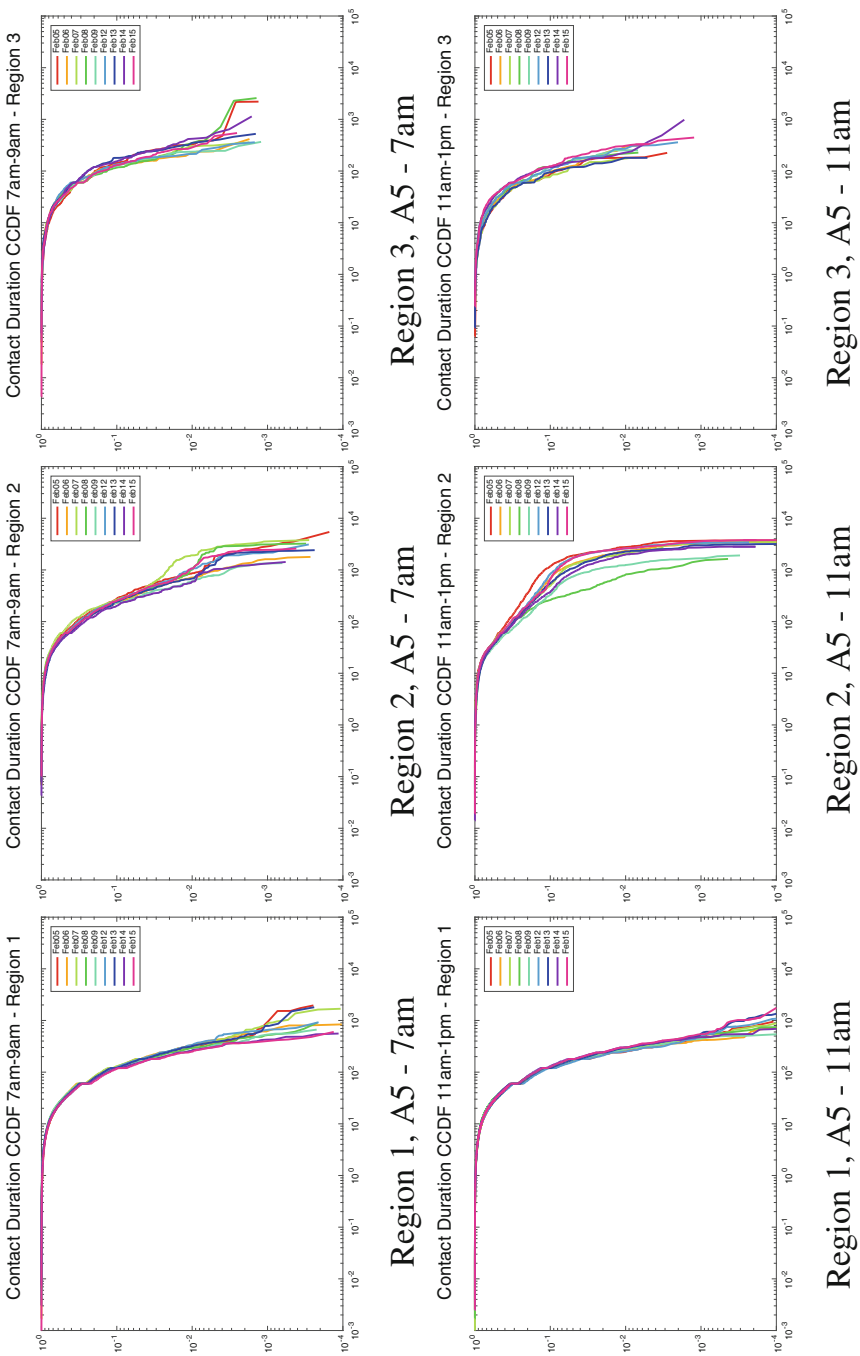


Fig. 13.3 Contact duration CCDF—area 5 of 3 regions over 2 time slots—time comparison

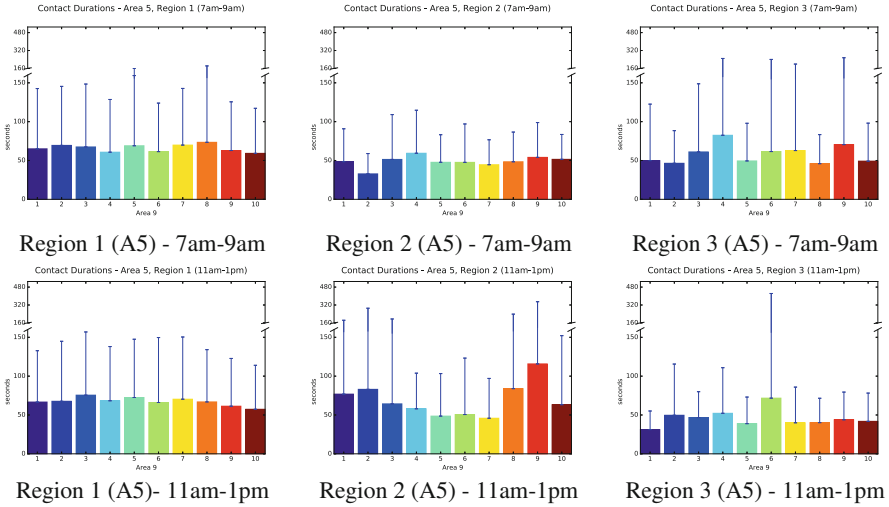


Fig. 13.4 Mean and variance—time comparison

Table 13.2 KS and AD test result: portion of pair of days over varying location

Location	Region1-I1	Region1-I2	Region2-I1	Region2-I2	Region3-I1	Region3-I2
A1	0.6000	0.2222	0.75	0.9556	1	0.7556
A2	0.2667	0.0667	0.5333	0.5556	0.4889	0.6000
A3	0.1778	0.2444	0.2444	0.2667	0.3556	0.2667
A4	0.1333	0.0222	0.3778	0.4222	0.4667	0.9556
A5	0.0889	0.0667	0.0667	0	0.3111	0.3778
A6	0.2667	0.0444	0.5333	0.1778	0.6222	0.7333
A7	0.1778	0.0222	0.5	0.3929	0.6667	0.9333
A8	0.1778	0.0667	0.5333	0.8000	0.2889	0.3556
A9	0.2000	0.1556	0.7111	0.6667	0.6222	0.8000

to match each other quite well, especially for smaller values. It appears that there are stable traffic flow characteristics within a specific area even for different weekdays, resulting in a stable contact duration pattern for each area.

Furthermore, the bar charts in Fig. 13.4 illustrate the changing of encounter duration through 10 days for Area 5. We can still see the steady mean and very little variation across different days within the same area even for both periods.

Finally, Table 13.2 shows an average of results from applying 95%-confidence validation test (Kolmogorov-Smirnov and Anderson-Darling, depending on the number of samples available in each case) which indicate whether the contact duration distributions are similar, over different pairs of days. Higher numbers indicate more similar distributions for a given area across different pairs of days.

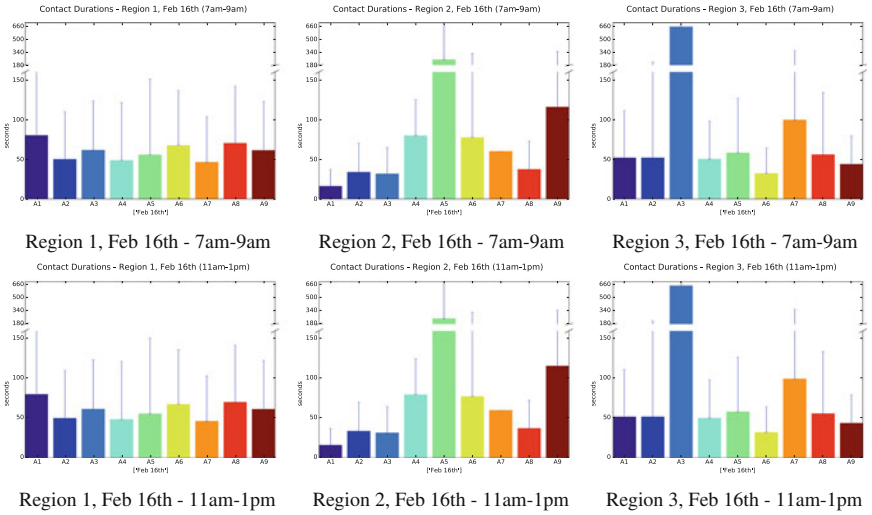


Fig. 13.5 Mean and variance—location comparison

In summary, we can give some conclusions:

- Each area has its own contact distribution pattern on different days. Different days do not have large impact on contact distribution within the same area.
- Some areas show a greater variation in the contact distribution over time than others.
- The variation in contact distribution can also depend on particular time of the day

13.4.2 Location

Similarly, the bar charts in Fig. 13.5 illustrate the changing of encounter duration through 9 areas for one specific day Feb 16. In these figures, we can see clear variation across different areas within the same day for both periods. In Region 2, the mean encounter duration for Area 5 (the densest area having entry of Shanghai Hongqiao International Airport Terminal 1) and Area 9 (the major intersection Longbai'ercun) is much higher compared to all other areas. There is also a peak in Area 3 in Region 3.

Table 13.3 shows the proportion of pairs of areas where a 95%-confidence KS and AD test indicates that the contact duration distributions are identical. Higher numbers indicate more similar distributions across areas on a given day. There is little difference in portions of similarity between the two periods. However, portions of similarity among different locations are quite low compared to the calculated portions among days. Therefore, we can see a weak impact of location on contact duration. There is low coherence in terms of contact duration at different locations.

Table 13.3 KS and AD test result: portion of 9 areas over 10 days

Day	Region1-I1	Region1-I2	Region2-I1	Region2-I2	Region3-I1	Region3-I2
Feb 5th	0.0556	0	0.3214	0.1944	0.1944	0.4167
Feb 6th	0.0278	0.0278	0.5	0.2500	0.3611	0.2500
Feb 7th	0.0278	0.0278	0.2143	0.3056	0.3611	0.3889
Feb 8th	0.0556	0.0278	0.1071	0.3571	0.4167	0.3889
Feb 9th	0.0278	0	0.3571	0.4722	0.3611	0.5556
Feb 12th	0.0278	0	0.4643	0.2778	0.4167	0.4722
Feb 13th	0.0278	0	0.1389	0.4444	0.3333	0.4444
Feb 14th	0.0278	0.0278	0.4167	0.2222	0.1389	0.5278
Feb 15th	0.0278	0.0278	0.2500	0.3611	0.4444	0.3056
Feb 16th	0.0278	0	0.2857	0.4167	0.2500	0.1944

In summary, we can give some conclusions:

- Each area has its own contact distribution pattern. Different locations do have large impact on contact distribution within the same day.
- The variation of the mean contact duration across areas depends on the particular day.

13.4.3 Direction

Figure 13.6 describes the CCDF (complementary cumulative density function) result of aggregated contact duration as well as contact distribution generated for same direction, opposite direction, and perpendicular direction at Area 5 over 10 days at 7 am and 11 am correspondingly. Vehicles traveling on different directions, east, west, south, and north, are identified, and contact duration is estimated for vehicles going on the same, opposite, or perpendicular direction. For both time intervals, CCDF for different types of relative directions are quite different. The CCDF generated from the opposite direction flow of vehicles seems most close to the aggregated CCDF compared to the other two types.

Table 13.4 shows the proportion of pairs of areas and days having the majority direction (same, opposite, perpendicular direction) where a 95%-confidence KS and AD test indicates that the contact duration distributions are identical. Higher numbers indicate more similar distributions across areas on a given day.

In summary, we can give some conclusions:

- The contact duration distributions are sensitive to the pairwise direction of the two vehicles contacting each other. The shortest contact duration is for vehicles going in perpendicular direction to each other.
- The contact durations corresponding to opposite direction contacts have the least variation across days. The contact durations for same direction contact have the most variation across days.

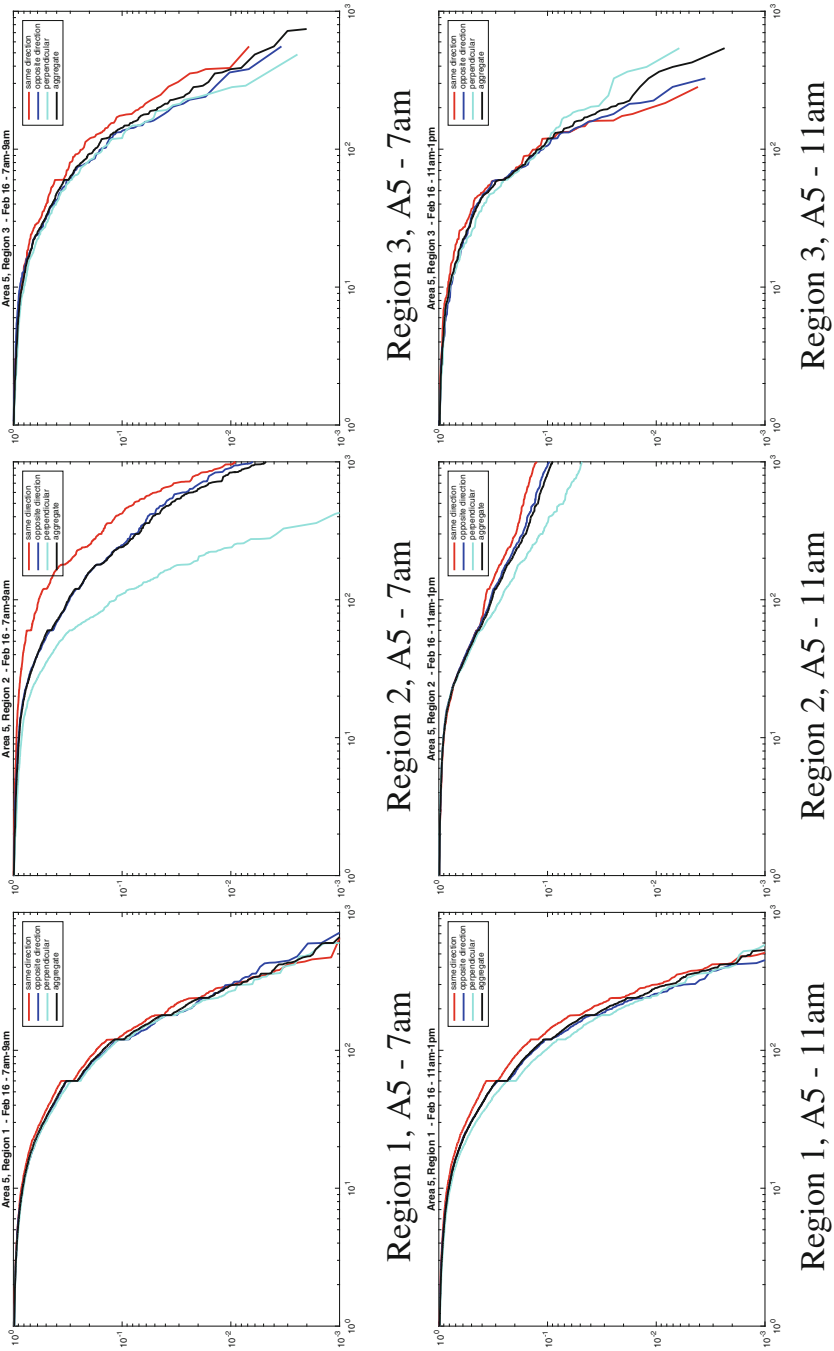


Fig. 13.6 Contact duration CCDF—area 5 of 3 regions over 2 time slots—direction comparison

Table 13.4 KS and AD test result: portion of 9 areas over 10 days

Direction	Region1-I1	Region1-I2	Region2-I1	Region2-I2	Region3-I1	Region3-I2
Same	0.0667	0.0111	0.4444	0.4103	0.5955	0.7000
Opposite	0.9778	0.8889	0.9333	0.8767	0.9775	0.9773
Perpendicular	0.0667	0.0222	0.5200	0.4247	0.6854	0.7386

13.5 Conclusion

This is one of the first studies in the literature to undertake a fine-grained analysis of contact duration from a real set of vehicular traces. Besides presenting the aggregate contact duration, we have examined and presented how contact durations vary with area, time of day, different days, as well as relative pairwise vehicular directions. We find that the contact durations tend to be somewhat stable over different days but vary over areas. The pairwise direction of vehicles is shown to have a significant impact on contact duration: vehicles going in the same direction tend to have the longest contact duration, while vehicles going in a perpendicular direction to each other were found to have the shortest contact duration.

References

1. Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R., Scott, J.: Impact of human mobility on the design of opportunistic forwarding algorithms. In: IEEE Infocom (2006)
2. Engmann, S., Cousineau, D.: Comparing distributions: the two sample Anderson-Darling Test as an alternative to the Kolmogorov-Smirnoff Test. J. Appl. Quant. Methods. **6**(3), 1–17 (2011)
3. Jung, K.-H., Lim, W.-S., Jeong, J.-P., Suh, Y.-J.: A link contact duration-based routing protocol in delay-tolerant networks. Springer Wirel. J. **19**(6), 1299–1316 (2013)
4. Lee, H.J., Nam, J.C., Seo, W.K., Cho, Y.Z., Lee, S.H.: Enhanced PRoPHET routing protocol that considers contact duration in DTNs. In: 2015 International Conference on Information Networking (ICOIN), Cambodia (2015)
5. Li, Y., Jin, D., Zen, L., Chen, S.: Revealing Patterns of opportunistic contact durations and intervals for large scale urban vehicular mobility. In: ICC (2013)
6. Scholz, F.W., Stephens, M.A.: K-sample anderson-darling tests. J. Am. Stat. Assoc. **82**(399), 918–924 (1987)
7. Yin, J., ElBatt, T., Yeung, G., Ryu, B., Habermas, S., Krishnan, H., Talty, T.: Performance Evaluation of Safety Applications Over DSRC Vehicular Ad Hoc Networks. ACM, New York (2004)
8. Yu, C., Tu, Z., Yao, D., Lu, F., Jin, H.: Probabilistic routing algorithm based on contact duration and message redundancy in delay tolerant network. Int. J. Commun. Syst. **29**, 2416–2426 (2016)
9. Zhang, X., Kurose, J., Levine, B.N., Towsley, D., Zhang, H.: Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing. In: IEEE Infocom (2007)
10. Zhu, H., Li, M., Luoyifu, Xue, G., Zhu, Y.: Impact of traffic influxues: revealing exponential inter-contact time in urban VANETs. IEEE Trans. Parallel Distrib. Syst. **22**(8), 1258–1266 (2011)
11. Zhuo, X., Li, Q., Gao, W., Cao, G., Dai, Y.: Contact duration aware data replication in Delay Tolerants Network. In: 19th IEEE International Conference on Network Protocols (2011)

Chapter 14
Flexible Bandwidth Scheduling for Streaming Data Movement Over Dedicated Networks

Liudong Zuo, Michelle Zhu, Chase Wu, Nageswara S. V. Rao, Min Han, and Anyi Wang

Abstract A wide range of scientific disciplines are generating large amounts of data at a high speed, which must be transferred to remote sites for real-time processing. Reserving bandwidths over dedicated channels in high-performance networks (HPNs) within a specified time interval has proved to be an effective solution to such high-demanding data transfer. Given a bandwidth reservation request, if the desired bandwidth within the specified time interval cannot be satisfied, most of the existing scheduling algorithms simply reject the request, which would immediately terminate the application. One reasonable approach to mitigate this issue is to provide an alternative bandwidth reservation option to schedule the desired bandwidth within the time interval closest to the specified one. We propose a flexible bandwidth reservation algorithm that considers both the best and alternative bandwidth reservation options for a given request. Extensive

L. Zuo (✉) · A. Wang
Computer Science Department, California State University, Dominguez Hills, Carson, CA, USA
e-mail: lzuo@csudh.edu; awang26@toromail.csudh.edu

M. Zhu
Department of Computer Science, Montclair State University, Montclair, NJ, USA
e-mail: zhumi@montclair.edu

C. Wu
Department of Computer Science, New Jersey Institute of Technology, Newark, NJ, USA
e-mail: chase.wu@njit.edu

N. S. V. Rao
Computer Science and Mathematics Division, Oak Ridge National Laboratory,
Oak Ridge, TN, USA
e-mail: raons@ornl.gov

M. Han
School of Software Engineering, Chengdu University of Information Technology, Chengdu,
Sichuan, China
e-mail: hanmin@cuit.edu.cn

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_14

simulations are conducted to show the superior performance of the proposed scheduling algorithm compared with a heuristic approach adapted from existing scheduling algorithms.

Keywords Bandwidth reservation · Dynamic provisioning · High-performance networks · Quality of service

14.1 Introduction

A wide range of scientific disciplines such as earthquake simulation and high energy physics are generating large amounts of real-time simulation, observational, and experimental data at a high speed [8]. For example, the large-scale data exploration process at the Korea Superconducting Tokamak Advanced Research (KSTAR) generates 3.9 TB of data within only 10 s [3]. Handling such extremely large volumes of data in a timely manner goes far beyond the data processing ability of the current KSTAR. A promising solution for timely data analysis is to quickly move the data to remote collaborating sites from memory to memory, such as Oak Ridge National Laboratory (ORNL) and National Energy Research Scientific Computing Center (NERSC), for near real-time data analysis. How to transfer data at such scales in a fast and reliable way with guaranteed performance is critical for data storage and analysis [9]. Reserving bandwidths over dedicated channels provisioned by high-performance networks (HPNs) within a specified time interval has emerged as a promising solution [1, 4, 6, 12]. For example, the On-Demand Secure Circuits and Advance Reservation System (OSCARS) deployed in ESnet is one of the most extensively used bandwidth reservation services, and ESnet and KSTAR are currently working together to improve the data transfer performance.

In general, a user submits a bandwidth reservation request (BRR) specifying the desired bandwidth to be reserved and the bandwidth reservation start and end time. Upon the receival of a BRR from the user, the bandwidth reservation service provider searches for and allocates corresponding network resources. If there are sufficient available network resources, the desired bandwidth can be successfully scheduled within the specified time interval; otherwise, to the best of our knowledge, most of the existing scheduling algorithms would simply reject the BRR, resulting in an immediate termination of the application. To address this issue, one reasonable approach is to perform flexible scheduling to schedule the desired bandwidth within the time interval closest to the user-specified one and return such option to the user for selection. Accordingly, we propose a bandwidth reservation algorithm, referred to as Flexible Streaming Bandwidth Scheduling (FSBS), which considers both the best and alternative bandwidth reservation options for a given BRR. For performance comparison, we design a heuristic scheduling algorithm adapted from existing scheduling algorithms, referred to as Basic Streaming Bandwidth

Scheduling (BSBS). Extensive simulations are conducted to show the superior performance of FSBS compared with BSBS. To the best of our knowledge, our work is among the first to study bandwidth scheduling with alternative reservation options in HPNs.

The rest of this paper is organized as follows. The related work is described in Sect. 14.2. The mathematical models are presented in Sect. 14.3. The algorithm design and illustration are detailed in Sect. 14.4. The performance evaluations are conducted in Sect. 14.5. We conclude our work in Sect. 14.6.

14.2 Related Work

Big data transfer through bandwidth reservation in HPNs has been widely used in various scientific domains, and many bandwidth reservation problems have been investigated in the past. We conduct a brief survey of related work as follows.

Balman *et al.* proposed one bandwidth reservation algorithm [1] to schedule a user request that specifies the total volume of data to be transferred, a maximum bandwidth that can be used on the client sites, and a desired time interval, within which the transfer must be completed. Upon the receival of such a request, the proposed algorithm finds the bandwidth reservation option with the earliest data transfer completion time or with the shortest data transfer duration. For a similar data transfer request, Lin and Wu considered the following four advance bandwidth scheduling problems: (1) fixed path with fixed bandwidth (FPFB), (2) fixed path with variable bandwidth (FPVB), (3) variable path with fixed bandwidth (VPFB), and (4) variable path with variable bandwidth (VPVB) [6]. The objective is to minimize the data transfer end time for a given transfer request with a pre-specified data size. A detailed problem complexity analysis was conducted, and corresponding algorithms were proposed for each of these problems.

Zuo and Zhu studied the problem of scheduling all BRRs concurrently along different paths in an HPN while achieving their best average transfer performance [14]. These problems were proved to be NP-complete, and heuristic algorithms were proposed. Similar problems on one fixed path were also studied [11, 13]. Zuo *et al.* further studied the scheduling of two generic types of BRRs concerning data transfer reliability: (1) to achieve the highest data transfer reliability under a given data transfer deadline and (2) to achieve the earliest data transfer completion time while satisfying a given data transfer reliability requirement [10, 15]. Two periodic bandwidth reservation algorithms were proposed to optimize the scheduling of individual BRRs within BRR batches.

All of the above work studied the scheduling of bandwidth reservation requests specifying the size of data to be transferred. The problem of bandwidth reservation for streaming data movement still remains open and is the focus of our work, where if the given BRR could not be achieved, we attempt to provide an alternative bandwidth reservation option for the user to choose.

14.3 Mathematical Models

We model an HPN as a graph $G = (V, E)$, where V and E represent the set of nodes and links, respectively [6, 12, 14]. For illustration purposes, we present an example HPN G in Fig. 14.1, where $V = \{v_s, v_0, v_d\}$ and $E = \{v_s - v_d, v_s - v_0, v_0 - v_d\}$. The dynamic bandwidth reservation and release on the links for data movements lead to the variation of G, namely, the available bandwidth of an link $l \in E$ may vary from time to time as shown by the available bandwidth table of links of G on the right side of Fig. 14.1. For convenience, we suppose that link l maintains a list of available bandwidths specified as a segmented constant function of time [6]. We further represent a link's available bandwidth using a 3-tuple of time-bandwidth (TB): $(t_l[i], t_l[i+1], b_l[i])$, which denotes the available bandwidth $b_l[i]$ of link l within time interval $[t_l[i], t_l[i+1]]$, $i = 0, 1, 2, \ldots, T_l - 1$, where T_l is the total number of time slots of link l. We set $t_l[T_l] = +\infty$ as there is no bandwidth reserved on any link of G after time point $t_l[T_l - 1]$. For example, link $v_0 - v_d$ in Fig. 14.1 has three TBs, $(0, 3s, 12Gb/s)$, $(3s, 8s, 10Gb/s)$, and $(8s, +\infty, 12Gb/s)$, while link $v_s - v_d$ only has one TB, $(0, +\infty, 6Gb/s)$.

Bandwidth reservation for data movement is generally made on one network path, which is defined as an ordered set of nodes from the source to the destination by concatenating one or multiple links [6]. Before computing the path for bandwidth reservation, we combine the TB lists of all links together to build an aggregated TB (ATB) list to store the available bandwidths of all links of G in each intersected time slot. The ATB list is denoted as $\{(t[0], t[1], b_0[0], \ldots, b_{|E|-1}[0]), \ldots, (t[T-1], t[T], b_0[T-1], \ldots, b_{|E|-1}[T-1])\}$, where T is the total number of new time slots after the aggregation of the TB lists of all links. For example, after aggregating the TB list of all links of G in Fig. 14.1, we have four time slots: $[0, 3s]$, $[3s, 6s]$, $[6s, 8s]$, and $[8s, +\infty)$, and the ATB list is $\{(0, 3s, 6Gb/s, 9Gb/s, 12Gb/s), (3s, 6s, 6Gb/s, 8Gb/s, 10Gb/s), (6s, 8s, 6Gb/s, 11Gb/s, 10Gb/s)$, and $(8s, +\infty, 6Gb/s, 11Gb/s, 12Gb/s)\}$. For convenience, we put the two endpoints of all time slots in a TreeSet to

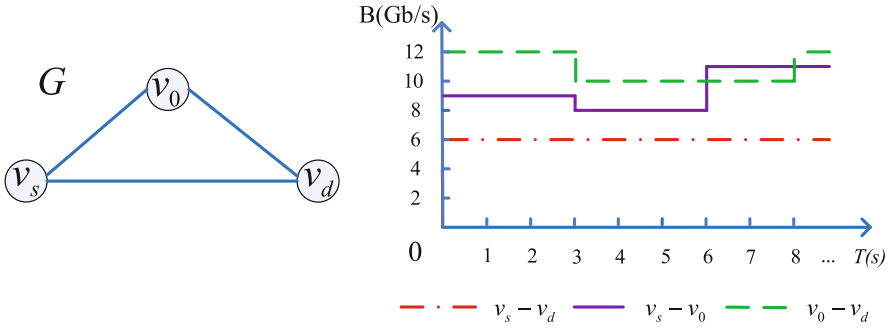


Fig. 14.1 The topology of an example HPN (left) and the available bandwidth table of the links of the example HPN (right)

facilitate the design of scheduling algorithms in Sect. 14.4. For example, the TreeSet after the ATB list aggregation of the links of G in Fig. 14.1 is $\{0, 3s, 6s, 8s, +\infty\}$.

We denote a BRR as a 5-tuple (v_s, v_d, B, t_S, t_E), where the user requests to reserve bandwidth B within time interval $[t_S, t_E]$ from source v_s to destination v_d. For example, suppose that G in Fig. 14.1 receives a BRR at time point 0 requesting to reserve bandwidth of $9Gb/s$ within the range $[4s, 7s]$ from v_s to v_d. The corresponding BRR is denoted as $(v_s, v_d, 9Gb/s, 4s, 7s)$.

We denote a bandwidth reservation option as a 4-tuple (p, B, t_s, t_e), where the HPN schedules bandwidth B on path p within time interval $[t_s, t_e]$. For example, for BRR $(v_s, v_d, 9Gb/s, 4s, 7s)$, we are not able to successfully schedule $9Gb/s$ within $[4s, 7s]$. The bandwidth reservation option that is closest to the required time interval $[4s, 7s]$ is $(v_s - v_0 - v_d, 9Gb/s, 6s, 9s)$, as shown in next section.

14.4 Algorithm Design and Analysis

This section focuses on the algorithm design of FSBS and BSBS. We first present the pseudocode of FSBS and BSBS, followed by the algorithm analysis and a brief illustration using one example BRR.

14.4.1 Algorithm Design and Illustration of FSBS

The pseudocode of FSBS is shown in Algorithms 1–3. We provide a brief illustration of FSBS using the example BRR received by G at time point 0: $(v_s, v_d, 9Gb/s, 4s, 7s)$.

Following Line 1 of Algorithm 1, we build the ATB list and create the TreeSet TS: $\{0, 3s, 6s, 8s, +\infty\}$. After Lines 2–7 of Algorithm 1, the available bandwidths of the links of G are: $b_{v_s-v_d} = 6Gb/s$, $b_{v_s-v_0} = 8Gb/s$, and $b_{v_0-v_d} = 10Gb/s$. The path with the largest available bandwidth returned by the modified Dijkstra's algorithm is $v_s - v_0 - v_d$, and its available bandwidth is $8Gb/s < 9Gb/s$. Hence, the requested bandwidth could not be scheduled within the specified time interval. We create bandwidth reservation list LBR and call Algorithm 2.

Currently, $m' = 1$ and the "while" loop begins. After the iteration, we could not identify any path with available bandwidth of at least $9Gb/s$. The "while" loop continues and $m' = 0$. When $i == 0$, after Lines 4–8 of Algorithm 2, the available bandwidths of the links of G are: $b_{v_s-v_d} = 6Gb/s$, $b_{v_s-v_0} = 9Gb/s$, and $b_{v_0-v_d} = 12Gb/s$. The path with the largest available bandwidth returned by the modified Dijkstra's algorithm is $v_s - v_0 - v_d$, and its available bandwidth is $9Gb/s$. We create bandwidth reservation option $(v_s - v_0 - v_d, 9Gb/s, 0, 3s)$ and add it to LBR. We have $t' = 4s - 3s = 1s$. The "while" loop ends here, and we then call Algorithm 3.

Currently, $n = 3$ and the "while" loop begins. After the iteration, we could not identify any path with available bandwidth of at least $9Gb/s$. The "while" loop

Algorithm 1: FSBS (Flexible Streaming Bandwidth Scheduling)

INPUT: $G = (V, E)$, a BRR (v_s, v_d, B, t_S, t_E).
OUTPUT: One bandwidth reservation option.

Combine the TB lists of all links together to build the ATB list. Create a TreeSet TS containing the two endpoints of all time slots. Identify the index of the largest element in TS that is no larger than t_S and the index of the smallest element that is no less than t_E, denoted by m and n, respectively;
for each $l \in E$ **do**
  $b_l = +\infty$;
  for $m \leq i \leq n-1$ **do**
    $b_l = min\,(b_l, b_l[i])$;
  end for
end for
Run modified Dijkstra's algorithm to identify the path with the largest available bandwidth from v_s to v_d within time interval $[TS[m], TS[n]]$. Suppose that the returned path is p and its available bandwidth is b;
if $b \geq B$ **then**
  Create bandwidth reservation option (p, B, t_S, t_E) and return it to the user.
else
  Create bandwidth reservation option list LBR and call Algorithm 2;
  Return the bandwidth reservation option in LBR to the user.
end if

Algorithm 2: Left Closest Bandwidth Reservation Option Computation

INPUT: $G = (V, E)$, TS, BRR (v_s, v_d, B, t_S, t_E), and bandwidth reservation option list LBR.
OUTPUT: NULL.

Initialize variables $t' = +\infty$ and $m' = m$;
while $m' \geq 0$ **do**
  for $n-1 \geq i \geq m'$ **do**
    for each $l \in E$ **do**
      $b_l = +\infty$;
      for $m' \leq j \leq i$ **do**
        $b_l = min\,(b_l, b_l[j])$;
      end for
    end for
    Run modified Dijkstra's algorithm to identify the path with the largest available bandwidth from v_s to v_d within time interval $[TS[m'], TS[i+1]]$. Suppose that the returned path is p and its available bandwidth is b;
    if $(TS[i+1] - TS[m']) \geq (t_E - t_S)$ && $b \geq B$ **then**
      Add bandwidth reservation option $(p, B, TS[i+1] - (t_E - t_S), TS[i+1])$ to LBR;
      $t' = t_S - TS[i+1]$ and break the "while" loop;
    end if
  end for
  $m' - -$;
end while
Call Algorithm 3.

Algorithm 3: Right Closest Bandwidth Reservation Option Computation

INPUT: $G = (V, E)$, TS, the BRR (v_s, v_d, B, t_S, t_E), t', and bandwidth reservation option list LBR.
OUTPUT: NULL.

```
1: while n ≤ |TS| − 1 do
2:   for m ≤ i ≤ n − 1 do
3:     if TS[i] − t_E ≥ t' then
4:       Continue;
5:     end if
6:     for each l ∈ E do
7:       b_l = +∞;
8:       for i ≤ j ≤ n − 1 do
9:         b_l = min(b_l, b_l[j]);
10:      end for
11:    end for
12:    Run modified Dijkstra's algorithm to identify the path with the largest available
       bandwidth from v_s to v_d within time interval [TS[i], TS[n]]. Suppose that the returned
       path is p and its available bandwidth is b;
13:    if (TS[n] − TS[i]) ≥ (t_E − t_S) && b ≥ B then
14:      if t' < +∞ then
15:        Remove the element from LBR;
16:      end if
17:      Add bandwidth reservation option (p, B, TS[i], TS[i] + (t_E − t_S)) to LBR;
18:      Break the "while" loop.
19:    end if
20:  end for
21:  n++;
22: end while
```

continues and $n = 4$. When $i == 2$, after computation, the path with the largest available bandwidth returned by the modified Dijkstra's algorithm is $v_s - v_0 - v_d$, and its available bandwidth is $10Gb/s > 9Gb/s$. Currently, $t' = 1s < +\infty$, we remove the current bandwidth reservation option $(v_s - v_0 - v_d, 9Gb/s, 0, 3s)$ in LBR from LBR and add the newly created bandwidth reservation option $(v_s - v_0 - v_d, 9Gb/s, 6s, 9s)$ to it. The "while" loop ends here.

At this point, LBR contains the following bandwidth reservation option: $(v_s - v_0 - v_d, 9Gb/s, 6s, 9s)$, which is returned to the user. The user makes a decision based on the current situation to either choose or reject the returned bandwidth reservation option. Figure 14.2 shows the topology of G if the user chooses the returned bandwidth reservation option; otherwise, G keeps the same topology as shown in Fig. 14.1.

In the worst case, the time complexity of FSBS is $O(T^3 \cdot |E| + T^2 \cdot (|E| + |V|log|V|))$.

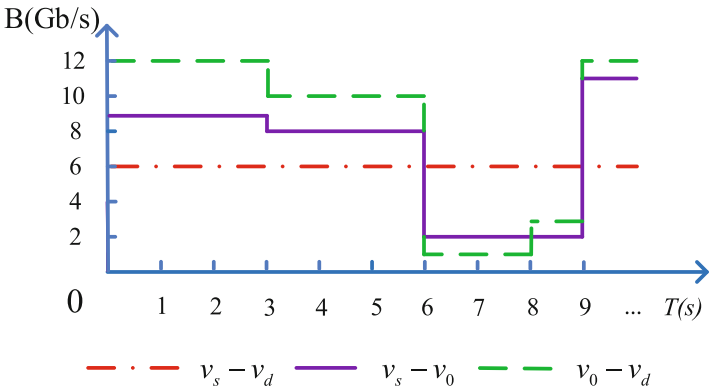


Fig. 14.2 The topology of the example HPN after the user accepts the returned bandwidth reservation option $(v_s - v_0 - v_d, 9Gb/s, 6s, 9s)$

14.4.2 Algorithm Design and Illustration of BSBS

As mentioned in Sect. 14.1, if there is no sufficient resource to satisfy the required bandwidth within the specified time interval, most of the existing bandwidth scheduling algorithms would simply reject the request. BSBS follows this scheduling strategy. Algorithm 4 shows the pseudocode of BSBS. In the worst case, its complexity is $O(T \cdot |E| + (|E| + |V|log|V|))$.

From the illustration of FSBS, we know that for $(v_s, v_d, 9Gb/s, 4s, 7s)$, we are not able to schedule bandwidth of $9Gb/s$ within $[4s, 7s]$. In this case, BSBS directly sends a reject message to the user.

Algorithm 4: BSBS (Basic Streaming Bandwidth Scheduling)

INPUT: $G = (V, E)$, a BRR (v_s, v_d, B, t_S, t_E).
OUTPUT: The best bandwidth reservation option if the given BRR can be successfully scheduled or a reject message, otherwise.

1: The same as Lines 1–7 of Algorithm 1;
2: Run modified Dijkstra's algorithm to identify the path with the shortest distance from v_s to v_d within time interval $[TS[m], TS[n]]$. Suppose that the returned path is p with available bandwidth b;
3: **if** $b \geq B$ **then**
4: Create bandwidth reservation option (p, B, t_S, t_E) and return it to the user.
5: **else**
6: Send a reject message to the user.
7: **end if**

14.5 Performance Evaluation

The OSCARS of ESnet is one of the most widely used bandwidth reservation services [2, 5, 7] in broad science communities. To mimic the real-life ESnet scenarios, we construct its topology using the data gathered from ESnet and conduct extensive simulations on this real-life network topology [12].

For a random generated BRR (v_s, v_d, B, t_S, t_E), v_s and v_d are randomly selected among the collected nodes, B is set to be a random integer between $1Gb/s$ and $8Gb/s$, t_S is randomly selected from $[0, 19s]$, and t_E is randomly selected from $(t_S, 20s]$. We run 10 sets of simulations. In the i-th set of simulations, $1 \leq i \leq 10$, there are 10 different batches of $i \times 200$ randomly generated BRRs. We implement and execute both FSBS and BSBS to process the same batch of randomly generated BRRs and measure the following two performance metrics in each simulation: (1) BRR scheduling success ratio, defined as the percentage of BRRs that have been successfully scheduled within a BRR batch, and (2) average data transfer completion time of the scheduled BRRs within a BRR batch. We plot in Figs. 14.3 and 14.4 the average performance metrics and the corresponding variances with the 95% confidence level across all the 10 BRR batches in each set of simulations.

The curves of FSBS_Best and BSBS in Fig. 14.3 represent the scheduling ratios of the BRRs that have been successfully scheduled by FSBS and BSBS, respectively. The average values of the above two ratios are 84.26% and 74.14%, and the average data transfer completion times of these two groups of BRRs are 10.41 s and 10.03 s (the curves of FSBS_Best and BSBS in Fig. 14.4), respectively.

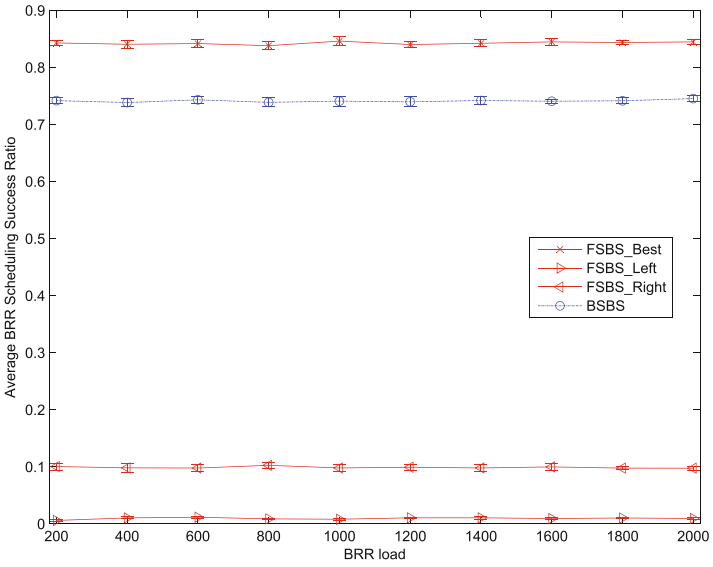


Fig. 14.3 Comparison of the BRR scheduling success ratio

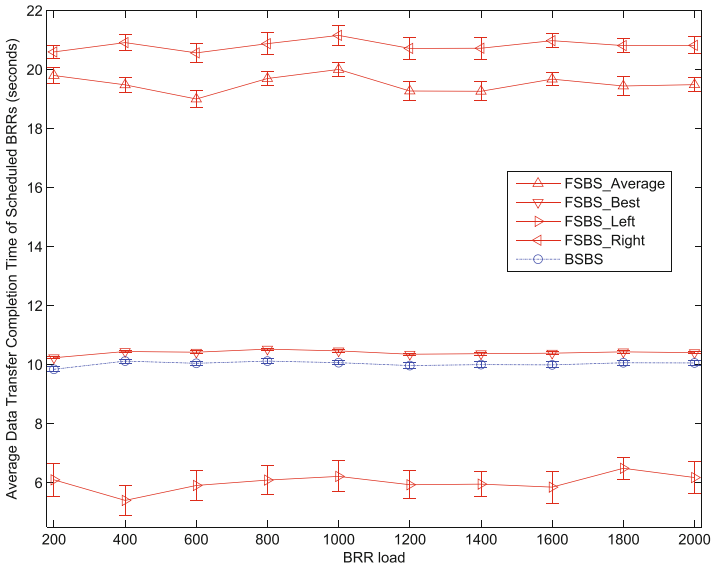


Fig. 14.4 Comparison of the average data transfer completion time of the scheduled BRRs

The curves of FSBS_Left and FSBS_Right in Fig. 14.3 represent the percentages of the BRRs within the entire batch whose closest bandwidth reservation options fall before and after the user-specified time interval, respectively. The average of the above two ratios are 0.96% and 9.90%, and the average data transfer completion times of these two groups of BRRs are 6.01 s and 20.82 s (the curves of FSBS_Left and FSBS_Right in Fig. 14.4), respectively. The average data transfer completion time of the closest bandwidth reservation options is 19.52 s as shown by the curves of FSBS_Average in Fig. 14.4.

The above performance measurements illustrate the flexibility of FSBS with an improved overall scheduling performance in comparison with BSBS.

14.6 Conclusion

In this paper, we studied flexible bandwidth scheduling for streaming data movement over dedicated networks. For a user request specifying the bandwidth reservation time interval and the desired bandwidth, if there is a lack of sufficient resources to satisfy the request, we considered providing a bandwidth reservation option to schedule the desired bandwidth within a time interval closest to the user-specified one and proposed a flexible scheduling algorithm, Flexible Streaming Bandwidth Scheduling (FSBS). For performance comparison, we also designed one basic scheduling algorithm, Basic Streaming Bandwidth Scheduling (BSBS). Extensive simulations were conducted to show the superior performance of FSBS.

The proposed scheduling algorithm has great potential to improve the stability of scientific applications running in network environments with limited resources.

It is of our future interest to develop more flexible service models and scheduling algorithms to improve network-based application performance and consider the scheduling of BRRs with different priority values in more complex environments shared by different groups of users.

References

1. Balman, M., Chaniotakisy, E., Shoshani, A., Sim, A.: A flexible reservation algorithm for advance network provisioning. In: Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 1–11. Washington, DC, USA (2010)
2. Charbonneau, N., Vokkarane, V.M., Guok, C., Monga, I.: Advance reservation frameworks in hybrid ip-wdm networks. IEEE Commun. Mag. **49**(5), 132–139 (2011)
3. Dart, E., Hester, M., Zurawski, J.: Fusion energy sciences network requirements review - final report 2014. In: ESnet Network Requirements Workshop (2014)
4. Guok, C., Robertson, D., Thompson, M., Lee, J., Tierney, B., Johnston, W.: Intra and interdomain circuit provisioning using the oscars reservation system. In: 3rd International Conference on Broadband Communications, Networks and Systems, pp. 1–8, San Jose, CA, USA (2006)
5. Lehman, T., Yang, X., Ghani, N., Gu, F., Guok, C., Monga, I., Tierney, B.: Multilayer networks: an architecture framework. IEEE Commun. Mag. **49**(5), 122–130 (2011)
6. Lin, Y., Wu, Q.: Complexity analysis and algorithm design for advance bandwidth scheduling in dedicated networks. IEEE/ACM Trans. Netw. **21**(1), 14–27 (2013)
7. Monga, I., Guok, C., Johnston, W.E., Tierney, B.: Hybrid networks: lessons learned and future challenges based on esnet4 experience. IEEE Commun. Mag. **49**(5), 114–121 (2011)
8. Stergiou, C., Psannis, K.E.: Recent advances delivered by mobile cloud computing and internet of things for big data applications: a survey. Int. J. Netw. Manag. **27**(3), e1930 (2016). https://doi.org/10.1002/nem.1930
9. Tanwir, S., Battestilli, L., Perros, H., Karmous-Edwards, G.: Dynamic scheduling of network resources with advance reservations in optical grids. Int. J. Netw. Manag. **18**(2), 79–105 (2008)
10. Zuo, L., Zhu, M.M.: Bandwidth provision strategies for reliable data movements in dedicated networks. In: 2016 IEEE International Conference on Big Data (Big Data), pp. 3069–3078 (2016)
11. Zuo, L., Zhu, M.M.: Improved scheduling algorithms for single-path multiple bandwidth reservation requests. In: The 10th IEEE International Conference on Big Data Science and Engineering (BigDataSE-16), pp. 1692–1699 (2016)
12. Zuo, L., Zhu, M., Wu, C.: Fast and efficient bandwidth reservation algorithms for dynamic network provisioning. J. Netw. Syst. Manag., **23**(3), 420–444 (2015)
13. Zuo, L., Zhu, M.M., Wu, C.Q.: Concurrent bandwidth scheduling for big data transfer over a dedicated channel. Int. J. Commun. Netw. Distrib. Syst. **15**(2/3), 169–190 (2015)
14. Zuo, L., Zhu, M.M., Wu, Q.Q.: Concurrent bandwidth reservation strategies for big data transfers in high-performance networks. IEEE Trans. Netw. Serv. Manage. **12**(2), 232–247 (2015)
15. Zuo, L., Zhu, M.M., Wu, C.Q., Zurawski, J.: Fault-tolerant bandwidth reservation strategies for data transfers in high-performance networks. Comput. Netw. **113**, 1–16 (2017)

Chapter 15
Stochastic Tools for Network Intrusion Detection

Lu Yu and Richard R. Brooks

Abstract With the rapid development of Internet and the sharp increase of network crime, network security has become very important and received a lot of attention. We model security issues as stochastic systems. This allows us to find weaknesses in existing security systems and propose new solutions. Exploring the vulnerabilities of existing security tools can prevent cyber-attacks from taking advantages of the system weaknesses. We propose a hybrid network security scheme including intrusion detection systems (IDSs) and honeypots scattered throughout the network. This combines the advantages of two security technologies. A honeypot is an activity-based network security system, which could be the logical supplement of the passive detection policies used by IDSs. This integration forces us to balance security performance versus cost by scheduling device activities for the proposed system. By formulating the scheduling problem as a decentralized partially observable Markov decision process (DEC-POMDP), decisions are made in a distributed manner at each device without requiring centralized control. The partially observable Markov decision process (POMDP) is a useful choice for controlling stochastic systems. As a combination of two Markov models, POMDPs combine the strength of hidden Markov Model (HMM) (capturing dynamics that depend on unobserved states) and that of Markov decision process (MDP) (taking the decision aspect into account). Decision-making under uncertainty is used in many parts of business and science. We use here for security tools. We adopt a high-quality approximation solution for finite-space POMDPs with the average cost criterion and their extension to DEC-POMDPs. We show how this tool could be used to design a network security framework.

Keywords POMDP · DEC-POMDP · IDS

L. Yu (✉) · R. R. Brooks
Clemson University, Clemson, SC, USA
e-mail: lyu@g.clemson.edu; rrb@g.clemson.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_15

15.1 Intrusion Detection System

Intrusion detection systems continuously monitor the computer system or network and generate alarms to inform the system administrator of suspicious events. IDSs are now considered a necessary addition to the security infrastructure of an organization [9]. The objective of intrusion detection is to detect malicious activities and accurately differentiating them from benign activities. According to the Common Intrusion Detection Framework (CIDF) [11], a general IDS architecture has four modules:

- Event-box (E-box) sensors monitor and collect information about the target system.
- Database-box (D-box) stores information from the E-box.
- Analysis-box (A-box) analyzes data stored in D-box and generates alarms if necessary.
- Response-box (R-box) implements countermeasures to thwart malicious intrusions.

The primary classes of detection methodologies include *signature-based detection*, *anomaly-based detection*, and *stateful protocol analysis* [9]. IDSs that employ signature-based detection identify attacks by comparing existing signatures of known attacks with the stored network traffic. When a match is found, IDSs will trigger the corresponding countermeasure to counteract the detected intrusion. Signature-based detection provides accurate detection results for well-specified attacks and effective known countermeasures can be taken. The major drawback of signature-based detection is its inability to detect new, unknown attacks. With new attacks appearing continuously, signature-based detection techniques suffer from high false negative (FN) rates. The anomaly-based detection has the potential to detect new types of attacks by estimating the deviation of observed information from the predefined baseline of "normal." However, there still exist several significant issues regarding anomaly-based detection, including high FP rates, low throughput but high cost, absence of appropriate metrics, etc. [5]. Stateful protocol analysis may provide more accurate detection results than the anomaly-based detection but is much more resource intensive due to the complex analysis and overhead generated from state tracking [9]. Typically, the more an IDS's detection accuracy can be improved from the default configuration, the less efficient it is. As a result, continuous monitoring may cause excessive computation bandwidth, which is undesirable for any computer system and network.

In addition, various intrusion prevention technologies have been implemented in IDSs, such as logging off the unauthorized user, shutting down the system, or reconfiguring the network if possible [16], etc. Despite strengthening the security of the information and communication systems, the intrusion prevention capabilities entail high cost in terms of energy and host resources.

It is not hard to see from the above introduction that, as a popular and effective tool against cyber-attacks by guarding the system's critical information, the resource cost of IDSs must be taken into account. IDS scheduling is needed to balance between security performance and resource consumption.

There are three primary types of IDS, namely, network-based IDS (NIDS), host-based IDS (HIDS), and stack-based IDS (SIDS). We choose HIDS since HIDS can be selectively deployed on critical machines, such as management servers, data servers and administrator consoles, etc.

15.2 Honeypot

Honeypots are needed to supplement IDSs in the proposed security scheme because they complement most other security technologies by taking a proactive stance. A honeypot is a closely monitored computing resource used as a trap to ensnare attackers. As defined by Spitzner in [10], "A honeypot is a security resource whose value lies in being probed, attacked, or compromised." The principal objectives of honeypots are to divert attackers away from the critical resources and study attacker exploits to create signatures for intrusion detection. The attraction of honeypots to attackers mitigates the threat of malicious attacks and thus helps secure the valuable information and important services located on the real targets.

Based on the level of interaction between the honeypots and the attackers, honeypots can generally be divided into the *high-interaction honeypots* and the *low-interaction honeypots*. Typical examples are honeyd [8] and honeynet [10]. Honeyd allows users to set up multiple virtual honeypots with different characteristics and services on a single machine. Honeynet monitors a larger and more diverse network when one honeypot may not be sufficient. It is very complex and expensive to deploy and maintain a high-interaction honeypot because it emulates almost all the activities found in a normal operating system. Deployment and configuration of a low-interaction honeypot is much easier and cheaper since it only simulates some system services. "BitSaucer" [1] proposed by Adachi et al. is a hybrid honeypot composed of both low-interaction and high-interaction capabilities.

Honeypots can also be divided into: *research* and *production honeypots* [7]. The primary function of a research honeypot is to extract the signatures of emerging attacks, which can be used to improve the detection accuracy of IDSs. A thorough understanding of the observed traffic data is time-intensive and requires analysts with comprehensive expertise in almost all network-related fields. Moreover, the deployment of research honeypots provides little benefit in strengthening the system security. Production honeypots are placed within the production network to mitigate risk. Most production honeypots are low-interaction honeypots and capture limited information. Example of production honeypot is Nepenthes [2]. Since our scheme is meant to improve security, we use production honeypots in combination with IDSs.

A honeypot mostly does not deal with false positives like IDS since all the services simulated by a honeypot have no production value. All the traffic that enters and leaves a honeypot is suspicious and should be monitored and analyzed [4]. However, not all attempts to access a honeypot are malicious. For example, a person might mistype the address of a computer and accidentally connect to a deployed honeypot. As a result, uncertainty is also involved in honeypots scheduling.

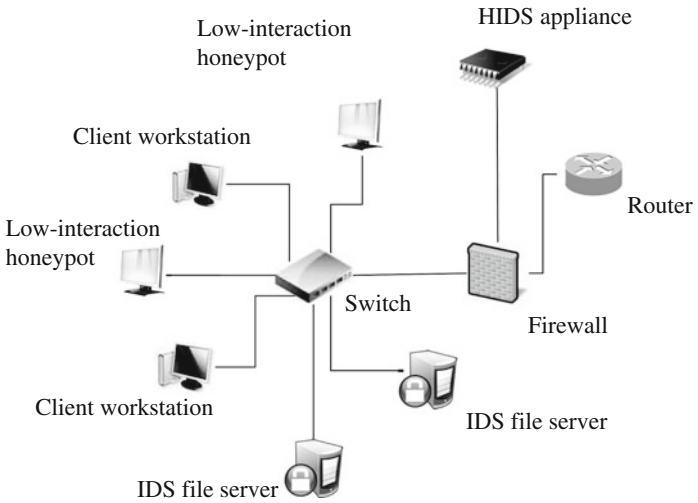


Fig. 15.1 Example distributed hybrid security scheme combining HIDSs with honeypots

15.3 System Model

Both HIDSs and honeypots detect intrusions and can operate at all times. However, intrusion detection and system emulation consume a large amount of energy and other resources, including memory, processor usage, and disk storage. We need to balance security performance and the resource cost by scheduling IDS and honeypot activities. The scheduling problem is modeled as a discrete-time process. In the proposed scheme, more than one HIDS or honeypot can be active during each time period.

We now consider an example of the hybrid system. Assume a local area network (LAN) is equipped with $K - H$ HIDSs and H honeypots. This brings the total number of the security devices to K. Without loss of generality, we also assume the network topology is static and there are more machines than available HIDSs. Inevitably, only some machines have HIDSs installed. An example network is shown in Fig. 15.1.

Suppose each HIDS can operate in three modes: *monitor*, *prevention*, and *sleep*, which are the action spaces for a HIDS. The HIDS is set to *monitor* for intrusion detection and can sleep for energy saving. A preventive action might be taken by switching to *prevention* mode if an unauthorized or malicious activity is identified and consumes more resources. Similarly, the action space of a deployed honeypot can be specified as *monitor*, *analysis*, and *sleep*. Further analysis will be carried out if traffic anomalies are detected. Each HIDS makes decision based on local information, which can be modeled as a partially observable Markov decision process (POMDP), which combines the strength of hidden Markov model (HMM) [6, 15] (capturing dynamics that depend on unobserved states) and that of Markov decision process (MDP) (taking the decision aspect into account).

Let $S^{(k)}(t)$ denote the state of an arbitrary device k (HIDS or honeypot) at time t, we assume $S^{(k)}(t) = \langle X^{(i)}(t), Y^{(k)}(t)\rangle$, where $X^{(k)}(t)$ represents the security condition and $Y^{(k)}(t)$ represents the resource consumption level. For instance, the security state can be simply divided into: "$secure$" and "$compromised$." If we use the notation $\mathscr{X}$ to denote the state space of $X^{(k)}(t)$, then $\mathscr{X} = \{secure, compromised\}$. The state space of $Y^{(k)}(t)$, denoted by $\mathscr{Y}$, includes three consumption levels: $\{low, medium, high\}$. The correspondence between different consumption levels and operating states are:

- *Low*: The device is not chosen, i.e., in the sleep mode.
- *Medium*: The device is working in the monitor mode.
- *High*: The HIDS/honeypot is working in prevention/analysis mode.

Since the IDSs used are HIDSs, each HIDS only monitors the machine it resides on, ignoring the rest of the network. As a decentralized control scheme, the decision to activate a certain security device is based on local observations. To complete the problem, we assume the observation space is identical to the space of security conditions, i.e., $\mathscr{O} = \mathscr{X} = \{$"$secure$", "$compromised$"$\}$. Note that an intrusion alarm does not necessarily mean there is an attack, and vice versa. Intrusion detection can make two types of errors: false positive (FP) and false negative (FN). A large volume of FPs result in lots of time wasted on determining whether an alert is an attack when it is actually benign [3]. FNs result in security holes due to failing to raise alarms when intrusions occur. Since the goal of intrusion detection is to precisely differentiate the intrusions from legitimate behaviors, both errors are significant performance indexes of IDSs and have been embodied in the observation probabilities. For instance, the following observation probability

$$\begin{aligned} Pr\{O^{(k)}(t+1) = \text{"}compromised\text{"} \mid A^{(k)}(t) \\ = \text{"}monitor\text{"}, S^{(k)}(t+1) = \langle secure, medium\rangle\} \end{aligned} \tag{15.1}$$

is equal to the FP rate of device k.

The local state of each host in the network is closely related to the security posture of the entire network. The analysis is divided into two cases:

- Switching between different modes of an HIDS will change the power consumption level of the local machine but have no influence on the attack activities in the LAN.
- The activation of a honeypot will positively impact the network's operation and security by distracting adversaries away from the valuable resources in the LAN and accordingly will mitigate the threat posed to the rest of the network.

It is obvious from the preceding analysis that the choice of each agent may affect the state of the entire network. This makes the decentralized partially observable Markov decision process (DEC-POMDP) a more suitable tool to model the scheduling for the distributed system. Some might argue that a centralized controller can also be adopted in this case. However, a common drawback exists in most centralized controls: it may consume a prohibitive amount of bandwidth and instantaneous

communication between the agents and the controller. In addition, possible security breaches are brought in since the transmitted information may be intercepted by the adversaries. Therefore, the DEC-POMDP is generally more preferable.

15.4 DEC-POMDP Formulation for the Distributed Hybrid Security System Combining IDS and Honeypot

We define the state of the DEC-POMDP formulation at time t, denoted by $S(t)$, as the combination of $S^{(k)}(t)$, $i = 1, 2, \cdots, K$. Notationally,

$$S(t) = [S^{(1)}(t), S^{(2)}(t), \cdots, S^{(K)}(t)]$$

Similarly, we can write the action of time t as

$$A(t) = [A^{(1)}(t), A^{(2)}(t), \cdots, A^{(K)}(t)]$$

We now consider the state transition law. The state $S(t)$ evolves based on a $(|\mathscr{X}| \times |\mathscr{Y}|)$-state Markov decision process. Let W denote the state transition probability function, then

$$W(\overline{s}, \overline{a}, \overline{s}') = Pr\{S(t+1) = \overline{s}' | S(t) = \overline{s}, A(t) = \overline{a}\}$$

where $\overline{s}, \overline{s}' \in \mathscr{S}^K$ and $\overline{a} \in \mathscr{A}^K$.

The observation probabilities of the DEC-POMDP formulation are slightly different from the standard POMDP. As indicated in (15.1), the quantities of the observations probabilities are assigned according to the FP and FN rates of the devices. Consequently, the observation of each agent only depends on the local information. It follows that $V^{(k)}(a, s', o')$, the observation probability of device i, is given by

$$V^{(k)}(a, s', o') = Pr\{O^{(k)}(t+1) = o' \mid A^{(k)}(t) = a, S^{(k)}(t+1) = s'\}$$

Finally, the immediate reward at time t is defined as the sum of each local immediate reward

$$r(S(t), A(t)) = \sum_{k=1}^{K} r(S^{(k)}(t), A^{(k)}(t)) \tag{15.2}$$

The values of the immediate rewards are assigned according to the following rules: A successful detection of an attack results in a large reward; on the contrary, an unnecessary further analysis staged due to a misjudgment will cause a large penalty, so will the misdetection of an attack; furthermore, monitoring is not free and monitoring a secure machine comes at a small penalty.

15.5 NLP-Based Solution of the DEC-POMDP

The scheduling model for the hybrid system is a DEC-POMDP. Thus, we need to augment the POMDP solution method in [13, 14] to situations of multiple controllers. As was mentioned in [12], the solution of a DEC-POMDP consists of a set of policy graphs, one for each agent. Accordingly, the goal is to optimize a set of finite state machines (FSM). We will show in the followings that the extension of the NLP-based solution in [13, 14] to DEC-POMDP is very straightforward. In order to present the algorithm for DED-POMDPs, we make the following assumptions:

- There are K agents in the DEC-POMDP.
- The state space of the DEC-POMDP is denoted by $\mathscr{S}$. Each agent has the same action space $\mathscr{A}$ ("*prevention*" of an IDS corresponds to "*analysis*" of a honeypot) and observation space $\mathscr{O}$.
- Each agent chooses the actions according to a fixed-size FSC. The set of the nodes in the FSC of agent k is denoted by $\mathscr{N}^{(k)}$.
- We use the notation $\overline{n}$ to denote a vector of length K, where $\overline{n}(k) \in \mathscr{N}^{(k)}$. The observation vector $\overline{o}$ and the action vector $\overline{a}$ are defined likewise.
- $x_k(n, a)$ and $y_k(n, o', n')$ are the control variables of the FSC of agent k.

The formal representation of the NLP-based solution of DEC-POMDPs satisfying the above assumptions is:

For variables : $\pi_{\overline{n}s\overline{a}}$ and $g^{(k)}(n_k, {o_k}', {n_k}', {a_k}')$,

$$\text{where} g^{(k)}(n_k, {o_k}', {n_k}', {a_k}') = x_k(n', a')y_k(n, o', n')$$

$$\text{maximize} \sum_{\overline{n}} \sum_{s \in \mathscr{S}} \sum_{\overline{a} \in \mathscr{A}} \pi_{\overline{n}s\overline{a}} \cdot r(s, \overline{a}^K)$$

Subject to :

$$\text{For } \forall s' \in \mathscr{S}, \forall \overline{n} \in \bigwedge, \forall \overline{a}' \in \mathscr{A}^K,$$

$$\pi_{\overline{n}'s'\overline{a}'} = \sum_{\overline{o} \in \mathscr{O}^K} \sum_{s \in \mathscr{S}} \sum_{\overline{a} \in \mathscr{A}^K} \{\pi_{\overline{n}s\overline{a}} \sum_{\overline{o}' \in \mathscr{O}^K} P(s, \overline{a}, s') Q(\overline{a}, s', \overline{o}')$$

$$\prod_k g^{(i)}(n_k, {o_k}', {n_k}', {a_k}')\},$$

$$\forall n_k \in \mathscr{N}^{(k)}, \forall {o_k}' \in \mathscr{O}, \sum_{n_k'} \sum_{{a_k}' \in \mathscr{A}} g^{(k)}(n_k, {o_k}', {n_k}', {a_k}') = 1,$$

$$\text{for } k = 1, 2, \cdots, K$$

The optimal solution to the NLP in (15.3) provides an optimal set of FSCs of the given size. The solution representation owes its availability to one critical factor: each agent behaves independently. That is, all the policy graphs are independent from each other.

15.6 Summary

This chapter starts with the introduction of the two security technologies adopted in the proposed security scheme. We choose HIDS, in combination with honeypot, with the purpose to integrate the advantages of both tools in our system. We formulate the decentralized control of the system as a DEC-POMDP. In the end of the chapter, we show how to extend the FSC-based POMDP algorithm described in [13, 14] to solving DEC-POMDP.

References

1. Adachi, Y., Oyama, Y.: Malware analysis system using process-level virtualization. In: Proceedings of IEEE Symposium on Computers and Communications, pp. 550–556 (2009)
2. Baecher, P., Koetter, M., Dornseif, M., Freiling, F.: The nepenthes platform: An efficient approach to collect malware. In: Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection, pp. 165–184. Springer, Berlin (2006)
3. Bakar, N., Belaton, B., Samsudin, A.: False positives reduction via intrusion alert quality framework. In: Joint IEEE Malaysia International Conference on Communications and IEEE International Conference on Networks, pp. 547–552 (2005)
4. Baumann, R.: http://security.rbaumann.net/download/honeyd.pdf. Originally published as part of the GCIA practical
5. Garcia-Teodoroa, P., Diaz-Verdejoa, J., Macia-Fernandeza, G., Vazquezb, E.: Anomaly-based network intrusion detection: Techniques, systems and challenges. Comput. Secur. **28**(1–2), 18–28 (2009)
6. Lu, C., Schwier, J.M., Craven, R.M., Yu, L., Brooks, R.R., Griffin, C.: A normalized statistical metric space for hidden Markov models. IEEE Trans. Cybern. **43**(3), 806–819 (2013)
7. Mokube, I., Adams, M.: Honeypots: Concepts, approaches, and challenges. In: ACMSE 2007, Winston-Salem, NC, pp. 321–325 (2007)
8. Provos, N.: In: Proceedings of the 12th USENIX Security Symposium, pp. 1–14 (2004)
9. Scarfone, K., Mell, P.: Guide to Intrusion Detection and Prevention Systems (IDPS). Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD (2007). NIST special publication 800-94
10. Spitzner, L.: Honeypots: Tracking Hackers. 1st edn. Addison-Wesley, Boston, MA (2002)
11. Tung, B.: The common intrusion detection framework (1999). http://gost.isi.edu/cidf/
12. Yu, L.: Stochastic tools for network security: Anonymity protocol analysis and network intrusion detection. Ph.D. thesis, Clemson University (2012). http://tigerprints.clemson.edu/all_dissertations/1061/
13. Yu, L., Brooks, R.: Observable subspace solution for irreducible pomdps with infinite horizon. In: Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research, p. 83. ACM (2011)

14. Yu, L., Brooks, R.R.: Applying pomdp to moving target optimization. In: Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop, p. 49. ACM (2013)
15. Yu, L., Schwier, J.M., Craven, R.M., Brooks, R.R., Griffin, C.: Inferring statistically significant hidden Markov models. IEEE Trans. Knowl. Data Eng. **25**(7), 1548–1558 (2013)
16. Zheng, J., Jamalipour, A.: Wireless Sensor Networks: A Networking Perspective. John Wiley & Sons, Hoboken (2009)

Chapter 16
Techniques to Certify Integrity and Proof of Existence for a Periodical Re-encryption-Based Long-Term Archival Systems

A. H. Shanthakumara and N. R. Sunitha

Abstract The periodical re-encryption-based archival systems have many specific characteristics such as actively re-encrypting the stored data objects periodically with or without conscious to the owner. For such a system, traditional techniques cannot be applied to check the integrity and proof of existence of a data object. For example, most traditional systems check integrity and proof of existence by comparing hash value of a data object stored in the archival system to the corresponding hash value with the owner. It may not be realistic solution due to continuous change in the bit patterns of the archivals due to periodical re-encryption. Therefore, we present a solution that is not only suitable to a specific periodical re-encryption-based archival system but also to any existing storage systems from long-term point of view.

Keywords Certify Integrity · Proof of Existence · Archival Systems

16.1 Introduction

The impact of digital information environments has been remarkably universal, with the generation of vast quantities of information. Sometimes these data are stored in a special infrastructure called digital preservation or archival system. Along with newly produced data, many countries have digitalized and stored the documents which are on papers to save physical space. The land registers [4, 18] in Europe and the digitization of records of high courts in India [20] are to name a few. The main goal of such a system is to secure the long-term persistence of information in digital form.

There are many reasons to keep on changing the bit patterns of the stored data objects periodically in the archival systems [1], with or without conscious to the

A. H. Shanthakumara (✉) · N. R. Sunitha
Siddaganga Institute of Technology, Tumakuru, Karnataka, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_16

owner of the data object. Today, we have several protection solutions based on cryptography. However, such solutions are not guaranteed to be secure in future as computer power and cryptanalysis evolve [24]. For example, we can use quantum computer techniques to attack applications currently using RSA signatures. Also, there is a possibility of strengthening the cryptographic security with advanced computation power by increasing the size of the key or block size. In order to protect the data for long term, there can be a change in key or key size or block size or cryptographic algorithm itself or format transformations resulting in change of bit pattern of data.

This work is based on our previous work, which is the periodical re-encryption-based archival system [23]. The archived data object may transform into several versions in a short period of time. In order to preserve the assurance of integrity all the way back to that of the original data object is a real challenge. Traditionally, integrity is checked by comparing the stored hash value with a newly computed hash from a data object being archived. In periodical re-encryption-based system, the bit patterns of the stored data object will keep on changing periodically. Hence it is not a feasible solution to compare newly computed hash value to the stored hash value with the owner in order to ensure integrity. We have many systems [8, 10, 12, 16, 22, 25] which address this problem through reregistration process of a data object to the archival system when it is re-encrypted with new credentials. It also involves the verification of the integrity and authenticity before the data object is transformed into a new format or version. This process is not practical, because the owner or organization who attached the data objects may not be online always or may no longer be available over a long period in order to authenticate during reregistration process. The data stored in archival systems which provides periodical re-encryption is useful, if their integrity (data objects are unaltered) is protected over a long term and also a proof of existence (a time reference when the data object is witnessed) is provided.

16.2 Related Work

In this section, we describe some common integrity checking and proof of existence techniques traditionally used in a digital archival systems.

16.2.1 Cryptographic Hash Function

Cryptographic hash function, as defined by Wenbo Mao [14], is a deterministic function which maps a bit string of an arbitrary length to a bit string of fixed length called a hash value or digest or simply hash and satisfies important properties: (a) hash value $h(x)$ for an input x should be computationally indistinguishable from a uniform binary string in the interval $(0, 2^{|h|})$ (Mixing transformation); (b)

it should be computationally infeasible to find two inputs x and y with $x \neq y$ such that $h(x) = h(y)$ (collision resistance); and (c) given a hashed value h, it should be computationally infeasible to find an input string x such that $h = h(x)$ (pre-image resistance). The security of traditional archival systems relies on the hardness of defeating one of the hash function properties. Therefore, today's secured hash function becomes insecure in future as cryptanalysis evolves, and thus, no single hash function can be secure from long-term point of view[15].

16.2.2 Digital Signature

A digital signature scheme consists of three algorithms [15, 16, 25]. An efficient key generation algorithm generates private and public key such that the message m is encrypted using private key that can be decrypted by using public key or vice versa. An efficient signing algorithm generates a signature on a given message m and by using private key. An efficient verification algorithm verifies and decides the signature is valid or not. RSA [21] and variants of ElGamal [6, 13] are popular digital signature schemes.

16.2.3 Merkle Tree

A Merkel tree [15, 17] is one of the most widely used hash linking schemes. The leaves of the Merkle binary tree are the hash values of the data objects being processed. The value stored at each internal node is the concatenated hash values of the children. The value computed at the root of the tree is called Merkle root, which represents the compressed value of all the data objects to be processed.

For example [8, 22], let us consider four data objects D_1, D_2, D_3, and D_4 with a corresponding hash values $h1$, $h2$, $h3$, and $h4$ which are to be processed at a time T_0. The corresponding Merkle tree is shown in Fig. 16.1.

The values of the internal nodes are obtained by $h12 = h(h1||h2)$, $h34 = h(h3||h4)$, and Merkle root $MR = h(h12||h34)$. To check the proof of data object D_2 whose hash value is $h2$, the required path is $D2 \rightarrow h2 \rightarrow h12 \rightarrow MR$. We can compute MR mathematically $h(h(h1||h2)||h34)$ with the information $h1$, $h2$, and $h34$. This information is called authentication path to the data object D_2. The

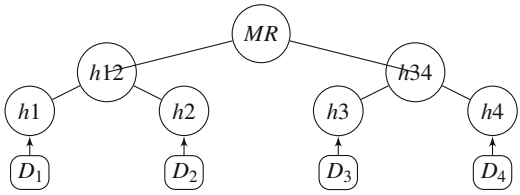


Fig. 16.1 Merkle tree with time stamp

compressed value MR is used as a proof of existence of a data object. Change to any of the data objects will result in a different MR value.

16.2.4 Time Stamp

Whenever an archivist, who manages the data objects, has a copy to be time-stamped, he or she transmits the hash value of a data object to a trusted time stamping authority (TSA). An authority records the date and time the hash value was received and retains a copy for safekeeping. Any challenger can check the integrity by comparing the archivist hash value with the TSA record [2, 7, 8, 11, 15, 22].

16.2.5 Evidence Record Syntax (ERS)

The Evidence Record Syntax (ERS), proposed by Gondrom et al. in RFC 4998 [8, 15], holds an evidence for each data object. The evidence record contains a sequence of certificates issued by archivist, and each certificate contains time stamp issued by a trusted time stamping authority (TSA) based on the archivist request on a Merkle root. Each time stamp covers a group of data objects by using Merkle tree.

16.2.6 One-Way Accumulators

One-way accumulator [3] is a function which takes two arguments from comparably sized domains and produces a result of similar size. In other words, it is a quasi-commutative hash function with two input arguments and produces fixed-size digest. The combination of quasi-commutative and one-way properties is used to develop one-way accumulator. The desired property of one-way accumulator is obtained by considering function $Acc : X \times Y \rightarrow X$ and asserting that for all $x \in X$ and for all $y1$,$y2 \in Y$, $Acc(Acc(x, y1), y2) = Acc(Acc(x, y2), y1)$.

16.2.7 Patricia Trees

It is a space-optimized data structure in which each node with a single child is merged with its parent [9]. It supports comparably high-speed search and insertion of a new node to the data structure. Unlike regular trees, the key at each node is compared considering a group of bits, where the quantity of bits in that group at that node is an r-ary tree. It is binary when r is 2. The example as shown in Fig. 16.2 represents a binary tree containing the strings 000, 011, 100, and 101. A new string is easily inserted into a data structure as each node represents a unique string and

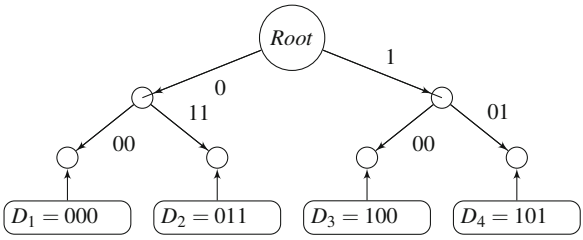


Fig. 16.2 Patricia tree containing the strings 000, 011, 100, and 101

its position is uniquely determined by its value. The root node represents an empty string, and leaf node represents the actual string in a data structure. The searching is as easy as tree traversal starting from a root and following left path if the bit of a string is 0 and, otherwise, the right path. The process is repeated until string is found or all bits of a string are exhausted.

16.3 Proposed Scheme

We propose a scheme by considering three actors in an archival system: data owner, who generates data object that needs to be archived for long-term usage; an archivist, who manages the archived data object in a secured manner; and a trusted middle layer between the owner and an archivist, which is defined as below.

16.3.1 Lightweight Integrity Management Layer (IML)

It computes necessary operation to provide proof of integrity and existence of data object. It uses an XOR operation to compute proof of integrity. In order to provide a proof of existence of a data object, it computes Merkle root on a special data structure called Patricia tree. The special feature of this data structure is to support time stamping scheme [19]. The value for proof of existence of a data object is computed based on the Patricia tree over a group of hash value of a data object.

16.3.2 Scheme Description

The basic computations required are defined in Fig. 16.3. The data owner computes $h \leftarrow Hash(D_i)$ before archiving the data object D_i. The notations B_i and A_i indicate the hash values of data object D_i before and after the encryption/re-encryption, respectively. The record $E_i \leftarrow (F_i, S_i)$ provides proof of integrity, and

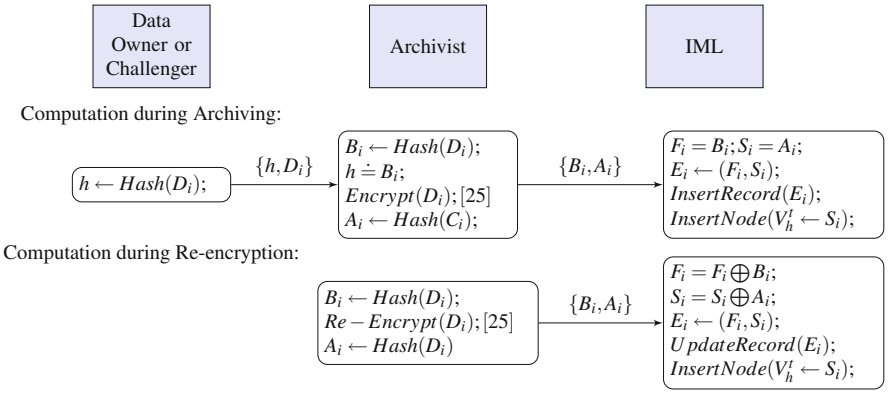


Fig. 16.3 The basic computations

the node V_h^t computed and inserted to Patricia tree at the time interval t will be used to confirm the existence of data object D_i.

Any challenger or the owner, who wishes to confirm the integrity of the data object D_i, can collect the record E_i from IML and current hash value A_i from the archivist. He/she computes $F_i \bigoplus A_i \doteq S_i \bigoplus h$ to confirm the integrity of the data object. If it does not match, there is a loss of integrity somewhere in the re-encryption stage. In order to find the exact interval where the integrity was lost, IML must store the record E_i for all re-encryption stages.

We organize our data structure based on Patricia tree. It is implemented as binary tree indexed by original hash value, h, computed at the time of archiving to enable efficient search. The internal nodes of the tree store small metadata that mainly contains a set of computed hash values by using hash values of its children at different stages. This set of hash values at each node is ordered by time intervals t. It also stores two additional values to represent left and right child records used at the time of computing hash value. Specifically, we represent it with the notations $V_n^t = \{h(V_l^t || V_r^t), l, r\}$. The record V is stored at the node n during the time interval t. The symbols l and r indicate the left and right child records used to calculate hash value of the node. The leaf node of the tree holds one additional record E to store the values for proof of integrity.

Let us consider the two data objects D_1 and D_2 that are to be archived at a time T_1. Let us assume that 00 and 01 are hash values of data objects D_1 and D_2, respectively. The IML computes the values for proof of integrity and updates the records to the corresponding internal nodes of the tree. At each leaf node, top record V holds the current hash value of the data object and, bottom record E holds the values for proof of integrity. Figure 16.4 shows a data structure at time T_1. The value of the record at root node $V^1 = \{h(V_0^1 || null), 1, null\}$ will be used as Merkle root. Authentication path is computed with the records V_{00}^1, V_{01}^1, and V^1.

Similarly, in a time interval T_2, let us assume that the data object D_2 is re-encrypted and data object D_3 with hash value 10 is inserted. In a time interval T_3,

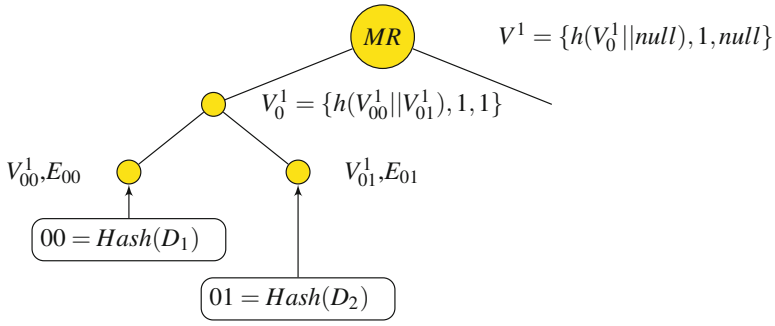


Fig. 16.4 Data objects D_1 and D_2 are archived at time interval T_1

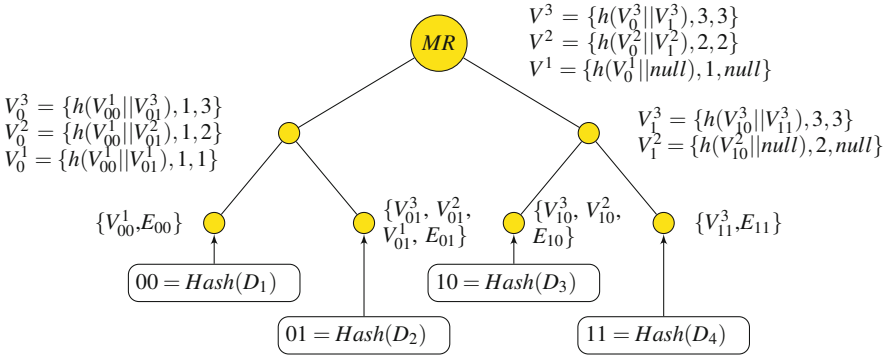


Fig. 16.5 Data structure at the time interval T_3

the data objects D_2 and D_3 are re-encrypted, and the data object D_4 with a hash value 11 is inserted. Figure 16.5 shows the corresponding data structure.

The records at the root of the tree can be used as Merkle root and, index of the record can be considered as time stamp at which the Merkle root is constructed. The owner or any challenger can construct Merkle root with the help of authentication path and specific time interval as record index and confirms the existence of their data object. For example, in order to confirm the existence of D_2, the records V_{01}^3, V_{00}^1, V_1^3, and V^3 are required. Similarly, to confirm D_1, the records V_{00}^1, V_{01}^1, and V^1 are required. The IML periodically removes the records which are out of scope in order to reduce the space overhead. For example, with reference to Fig. 16.5, at the time interval T_3, all records with index 2 (time interval 2) except V_{10}^2 are irrelevant and can be removed from the tree. The record V_{10}^2 is required to prove that the data object D_3 with hash value 10 is archived at a time interval T_2.

16.4 Implementation

We implemented IML layer using Java code on a desktop machine and performed some experiments using Enron's employees email data set [5]. These experiments run on an algorithm called a re-encryption [23]. We implemented Patricia tree-based data structure to hold the records in IML layer. The following functions describe the overview of IML layer:

1. The **SearchNode** enables to search the data object given by hash value. It is basically traversing from root node to corresponding leaf node in the tree.
2. The **InsertNode** enables to insert a new node to the tree during new archival.
3. The **InsertRecord** enables to insert a new record to a node. It could be used when there is a change in the metadata of the node.
4. The **UpdateRecord** enables to update the values for proof of integrity after re-encryption of a data object occurred.
5. The **DeleteRecord** enables to delete a record of a node. It is used when IML wants to remove irrelevant records from the tree.

The IML maintains a special list to make a note on the data objects, which are undergone in the process of re-encryption. Periodically, IML calculates Merkle root on hash values of all those data objects and accordingly inserts new records to the corresponding nodes in the data structure.

16.5 Results and Discussion

In the long-term archival system, it may not be possible for the owner or client to be online always in order to receive and store the certificates issued by the archivist. Therefore, client server-based systems may not be strongly advisable solutions for long-term and re-encryption-based archival systems.

The major problem of the ERS schemes [8] is the size of linking information. The size of the metadata is linearly increasing with the number of data objects of an archival system. In addition, it may not be a practical solution due to frequent re-encryption events of the data object.

The major problem of one-way accumulator-based scheme is the high complexity involved in computing a one-way accumulation on every re-encryption of a data object. If the archival node takes one microsecond per data object to generate a certificate with a new accumulator value, it could process only around 8% data objects compared to ERS schemes.

The major complexity is eliminated by using simple XOR operation in our scheme. The simulation of IML shows that it could process 2–3% more data objects than ERS scheme in the time period of 30 ms. We selected 90,000 emails randomly of Enron's employees with an average size of 1.9 KB. We treat each email with all its attachment as a single data object. The simulation started with ten initial data objects

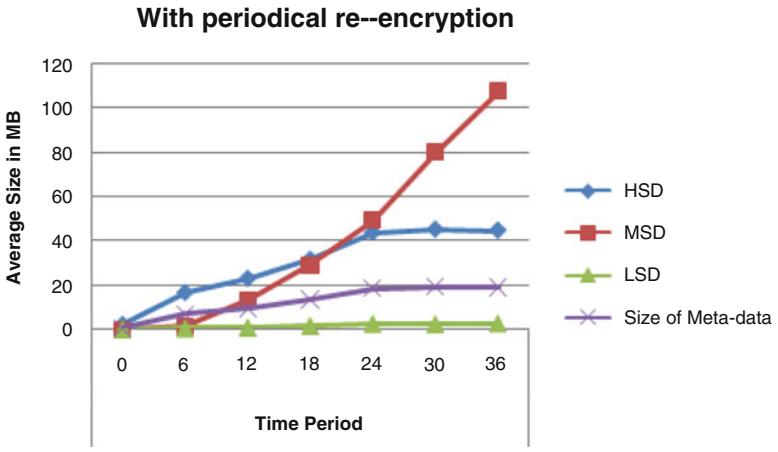


Fig. 16.6 Storage requirements for data objects and metadata when considering the periodical re-encryption

and uniformly added as defined in [23]. On every millisecond (a time interval), IML calls to calculate the proof of existence among the hashes of the data objects which are re-encrypted during that time interval.

Over 5 runs of simulation of periodical encryption algorithm [23] could process up to 81,400 emails in 30 time intervals, out of which nearly 23,600 data objects are highly sensitive data objects (HSD). It also shows that there is a linear increment in medium sensitive data objects (MSD) over a period of time. Figure 16.6 shows the storage requirements for data objects and metadata. It shows that the required size of the metadata for our scheme is stabilized to around 20 MB as stabilization is observed with respect to HSD objects. Figure 16.7 shows the storage requirement of metadata that is linearly increasing with respect to number of data objects in the absence of periodical re-encryption. It shows that our scheme could also be used with traditional system.

The IML scheme is lightweight and easy to implement on any periodical re-encryption-based archival systems. Storage space required for storing the information with respect to integrity checking and proof of existence is comparatively reduced.

16.6 Conclusion and Future Work

The obsolescence of hash functions or cryptographic security algorithms affects the solutions of archival systems from long-term point of view. The traditional, reregistration process of data object to an archival system whenever the version or format changed may not be a realistic solution. Ensuring the integrity and proof of

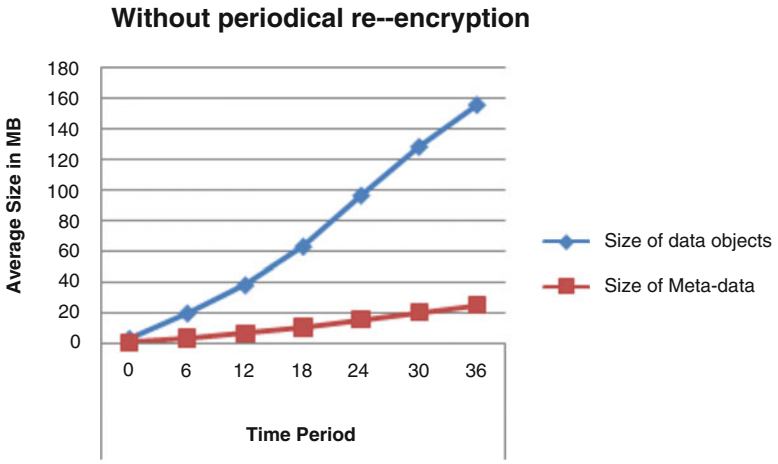


Fig. 16.7 Storage requirements for data objects and metadata when without considering the periodical re-encryption

existence of a data object whenever the owner needs confirmation is a real challenge of any archival systems.

In this paper we have shown the challenge and need of a specific technique to certify long-term integrity and proof of existence for a periodical re-encryption-based archival system. Our scheme uses most suitable time stamping technique and focus more on realistic implementation in the archival systems. The experiments demonstrate that metadata required to provide proof of integrity and proof of existence are compact in size, which is reaching 20 MB for 81,000 emails. The simulation results show that the storage requirement for metadata is stabilized over a period of time instead of linearity. This scheme is very lightweight and easy to implement. The owner need not to store all the certificates issued by the archivist, and data object integrity and proof of existence are easily verifiable by any challenger at any point of time.

The unfocused issue in our scheme is that if the data object is altered, it is unable to generate a caution alarm message to authorized person or archivist.

References

1. Baker, M., Shah, M., Rosenthal, D.S.H., Roussopoulos, M., Maniatis, P., Giuli, T., Bungale, P.: A fresh look at the reliability of long-term digital storage. In: Proceedings of EuroSys 2006, pp. 221–234, April 2006
2. Bayer, D., Haber, S., Stornetta, W.: Improving the efficiency and reliability of digital time-stamping. In: Sequences II: Methods in Communication, Security, and Computer Science, pp. 329–334. Springer, Berlin (1993)

3. Benaloh, J., de Mare, M.: One way accumulators, a decentralized alternative to digital signatures. In: Advances in Cryptology - EUROCRYPT '93, LNCS 765, pp. 274–285. Springer, Berlin (1994)
4. Centre of Registers and Information Systems: e-Land Register. http://www.egov-estonia.eu/e-land-register. Accessed 20 Jan 2015
5. Cohen, W.: Enron email dataset. http://www.cs.cmu.edu/~enron. Accessed 20 Jan 2015
6. Gamal, T.E.: On computing logarithms over finite fields. In: Williams, H.C. (ed.) CRYPTO. Lecture Notes in Computer Science, vol. 218, pp. 396–402. Springer, Berlin (1985)
7. Giuli, P., Maniatis, P., Giuli, T., Baker, M.: Enabling the long-term archival of signed documents through time stamping. In: Proceedings of the 1st USENIX Conference on File and Storage Technologies. FAST '02, USENIX Association, pp. 31–46 (2002)
8. Gondrom, T., Brandner, R., Pordesch, U.: Evidence Record Syntax (ERS) Securing Electronic Business Processes, in RFC 4998, pp 367–375 (2007)
9. Goodrich, M., Tamassia, R., Hasic, J.: An efficient dynamic and distributed cryptographic accumulator. In: Proceedings of Information Security Conference (ISC). Lecture Notes in Computer Science, vol. 2243 , pp. 372–388. Springer, Brelin (2002)
10. Haber, S.: Content integrity service for long-term digital archives. In: Archiving 2006, pp. 159–164. IS&T, Ottawa (2006)
11. Haber, S., Stornetta,W.S.: How to time-stamp a digital document. J. Cryptol. **3**(2), 99–111 (1991)
12. Haber, S., Kamat, P., Kamineni, K.: A content integrity service for digital repositories. In: 3rd international conference on open Repositories, Southamton, UK (2008).
13. Johnson, D., Menezes, A., Vanstone, S.A.: The elliptic curve digital signature algorithm (ecdsa). Int. J. Inf. Sec. **1**(1), 36–63 (2001)
14. Mao, W.: Modern Cryptography Theory and Practice. Hewlett Packard Professional Books, 1st edn. Prentice Hall, Upper Saddle River (2006)
15. Martin, A., Gagliotti, V., Daniel, C., Alexander, W., Johannes, B.: Authenticity, integrity and proof-of-existence for long-term archiving: a survey. IACR Cryptology ePrint Archive, p. 499 (2012)
16. Martin, A.G. Vigil, Moecke, C.T., Custdio, R.F., Volkamer, M.: The Notary Based PKI A Lightweight PKI for Long-Term Signatures on Documents. Public Key Infrastructures, Services and Applications. Lecture Notes in Computer Science, vol. 7868, pp. 85–97. Springer, Berlin (2013)
17. Merkle, R.C.: A certified digital signature. In: Brassard, G. (ed.) Advances in Cryptology - CRYPTO '89. Lecture Notes in Computer Science, vol. 435, pp. 218–238. Springer, Berlin (1989)
18. Ministere de la Justice: Livre Foncier. https://www.livrefoncier.fr. Accessed 20 Jan 2015
19. Oprea, A., Bowers, K.D.: Authentic time-stamps for archival storage. IACR Cryptology ePrint Archive 2009, p. 306 (2009)
20. RFP for Scanning and Digitization of Records of High Court of Himachal Pradesh. http://hphighcourt.nic.in/pdf/TenderDigitiz21112014.pdf. Accessed 20 Jan 2015
21. Rivest, R., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978)
22. Sangchul, S., Joseph, J.: Techniques to audit and certify the long-term integrity of digital archives. Int. J. Digit. Libr. **10**(2–3), 123–131 (2009)
23. Shanthakumara, A.H., Sunitha, N.R.: A novel archival system with dynamically balanced security, reliability and availability. In: 2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, pp. 27–33. IEEE (2016). https://doi.org/10.1109/CSITSS.2016.7779435
24. Storer, M.W., Greenan, K.M., Miller, E.L.: Long term threats to secure archives. In: Proceedings of the 2006 ACM Workshop on Storage Security and Survivability (2006)
25. Troncoso, C., De Cock, D., Preneel, B.: Improving secure long-term archival of digitally signed documents. In: Proceedings of the 4th ACM International Workshop on Storage Security and Survivability. StorageSS '08, pp. 27–36. ACM, New York (2008)

Chapter 17
SCADA: Analysis of Attacks on Communication Protocols

T. C. Pramod and N. R. Sunitha

Abstract SCADA (supervisory control and data acquisition) systems are used to monitor and control the processes of industrial facilities remotely. The use of standard technologies and interconnections between the systems lead to variety of security attacks. SCADA systems are being the part of many critical applications of the society. Any minute deviation in the normal operation of the system results in serious consequences. Hence, securing the industrial control systems is a high priority issue. One can provide security and safety to the system by identifying possible sources of threats and objectives of attackers and continuous monitoring of the operations of the system. In this paper, attack incidents occurred on command and control systems are presented (from the year 1982 to 2017), the general attacker goals on SCADA systems are discussed, SCADA communication protocols and its normal/abnormal behaviors are analyzed using the Wireshark tool.

Keywords Critical infrastructures · DNP3 · Industrial control systems · ModbusTCP · SCADA attacks · SCADA security · Stuxnet · Ransomware · Wireshark

17.1 Introduction

Critical infrastructures are the primary needs of the society. As the need for such infrastructures is increasing along with technological improvements, the global usage, interconnections, and sophistication in the operations of these infrastructures are also increasing. In order to reduce the complexity, there is a need for simple and efficient process supporting remote control of various activities, which can be achieved through automation. Thus, these infrastructures heavily depend on industrial control system (ICS) such as SCADA systems, distributed control systems

T. C. Pramod (✉) · N. R. Sunitha
Department of Computer Science, Siddaganga Institute of Technology, Tumakuru, Karnataka, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_17

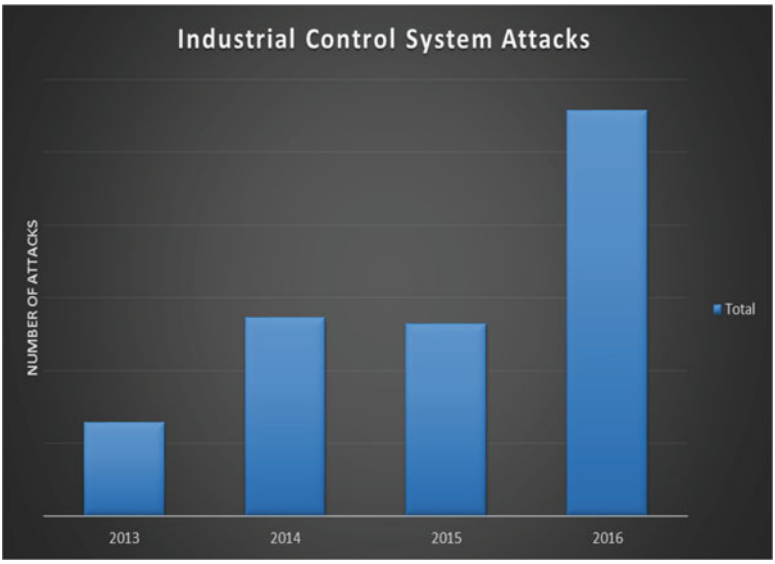


Fig. 17.1 Industrial control system attacks [2]

(DCS), and Remote Terminal Units (RTU). Presently, the critical infrastructures such as electric power grids, water distribution plants, paper and pulp industry, oil refineries, chemical production and processing, and manufacturing plants are the major examples in which the SCADA systems are playing a critical role [1].

In the past, considering security for the industrial automation domain had less attention. From the year 2010, due to massive security attacks, the implication of security features for command and control systems is proliferating. Figure 17.1 shows the total attacks on ICS on different years [2]. The following are the reasons for the proliferation of cyberattacks on the ICS:

- The use of standard technologies in the automation systems is increasing. The use of COTS (commercial off-the-shelf) hardware and software products, common internet protocols and solutions, and Windows- and Unix-like operating systems is quite common in the industrial automation systems.
- To achieve cost-effectiveness, fast decision-making and to optimize the production and manufacturing processes, the industrial networks are increasingly interconnected.
- Initially, the industrial communication protocols (Modbus, DNP3, etc.) were designed as serial communication protocols (to run over a serial connection). Over time, these protocols are used as application layer protocols on top of the TCP/IP stack. But, the lack of adoption of security measures and cryptographic mechanisms provides a way for the hackers to interrupt the normal communications.

- Incorporation of wireless technologies in the industrial automation systems. Many security attacks are possible in wireless solutions [3].
- Insecure communication links are also a security concern. The communication between the control center and remote locations may take place using the Internet/radio or microwave/leased lines. Compromising these communication links is easy for the hackers [4].
- Connection of industrial automation network with third-party vendors, contractors, alliance partners, and outsourcing also leads to cyberattack incidents.
- Significant information about automation and control systems is freely available to the public sector. Search engines like Google dorks [5], Shodan [6], and Pastebin [7] provide significant information about the industrial control systems online.

To secure the infrastructures, solutions like secure communications (encryption and decryption) and intrusion detection systems (IDS) can be used. Since the lifetime of ICS is high, i.e., in decades, many industries contain legacy hardware and software systems. Also, in some real-time industrial applications, the latency involved in performing cryptographic operations may not be tolerable. This introduces the difficulty to incorporate encryption/decryption operations. In such cases, IDS can be used for monitoring the malicious activities. Also, some of the proprietary or legacy protocols used in ICS may not be supported by current IT security tools such as firewalls or IDS. An alternate solution is the use of forensic techniques where details like where the attack originated, the processed involved, and the responsible identity for the attacks can be determined.

The rest of this paper is organized as follows: In Sect. 17.2, an overview of SCADA systems is presented. In Sect. 17.3, attack incidents (year 1982–2017) occurred on SCADA systems are discussed. In Sect. 17.4, attacker goals on SCADA systems are discussed in general. In Sect. 17.5, possible attacks on Modbus and DNP3 protocols are listed. In Sect. 17.6, using the Wireshark tool, Modbus packets are analyzed, and Sect. 17.7 gives the conclusion.

17.2 SCADA Systems

SCADA systems are designed to monitor and control the industrial processes remotely. Figure 17.2 shows the architecture of the SCADA systems. It consists of the following devices:

- HMI (human-machine interface): HMI provides an interface for the operator to interact with the system and to view and react to the process status and historical events [1].
- MTU (Master Terminal Unit): MTU is the higher-level device in the SCADA system, which collects the data from the distributed field-level equipments by issuing commands, stores and processes the data, and displays the information in the form of graphs, curves, and tables to HMI.
- SUBMTU: To alleviate the burden of the primary MTU, SUBMTUs are used.

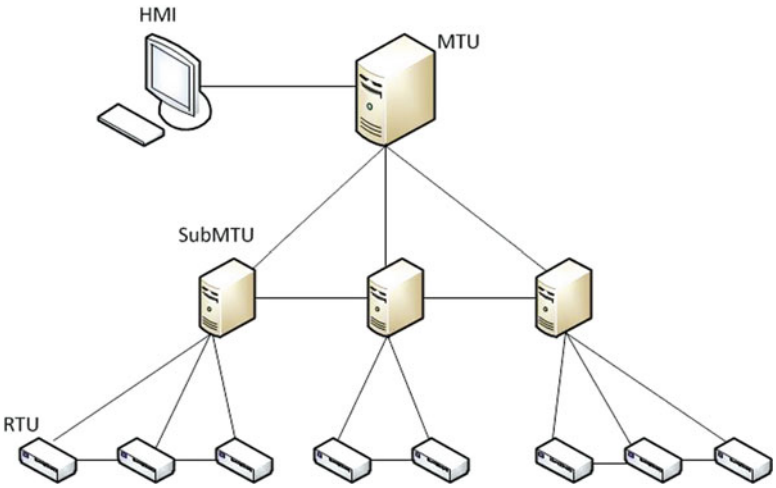


Fig. 17.2 SCADA architecture [8]

- RTU (remote terminal unit): RTUs are used for data acquisition from sensor devices and actuators. They send the collected data to master terminal unit (MTU) in digital format. They are located remotely from the control center.

17.2.1 Behavior of the SCADA Systems

- *Traffic periodicity:* In SCADA systems, the packet transmission rate is periodic in nature. The stability is due to the automated process. Communications occur based on polling mechanism (HMI polls PLC at a fixed frequency).
- *Fixed number of devices:* The network consists of fixed number of devices.
- *Continuous operation:* The systems are intended to operate ceaselessly for a long time.
- *Limited number of protocols:* The number of protocols used in SCADA network is less.
- *Limited number of packets:* The network has low throughput. The communications are regular in nature.
- *Limited human-initiated actions:* The HMI-to-PLC communication is extremely regimented device-to-device communication, with minimal human-initiated actions.

17.3 Attack Incidents on Command and Control Systems

A view on attack incidents occurred on command and control systems is portrayed in Table 17.1 (from the year 1982–2017).

17.4 Attacker's Goals on SCADA Systems

By observing the key characteristics of the SCADA systems, we have identified some of the general attacker objectives:

- **Gaining access into the SCADA network:** In order to perform malicious activities and to disrupt the normal processes, the primary goal of the attacker is to enter the SCADA network. The attackers may enter the system via vendors/contractors (third party), disgruntled employees, unsecured remote field sites, IT network, SCADA transmission media, wireless interface, Internet, corporate network, remote access, infected USB or laptops, physical attack, or poorly configured firewalls.
- **Identifying SCADA devices and available ports in the network:** Once the attacker gets access into the network, their intention is to identify the devices, ports used/opened, and communication paths. Recognizing who is master, who are slaves, and protocols used for communication is their curiosity.
- **Switching on/switching off the devices:** The operations of SCADA systems are remote based. The attacker may send a command to switch off the device in place of controller's command to switch on the device and vice versa.
- **Reading data from the devices:** The communication protocols were designed without considering cryptographic features. Reading the conversation between master and slave devices or reading the data directly from the slave or master device is the attacker interests.
- **Writing data into the devices:** The lack of authentication helps the attackers to write the data into the devices. The attacker likes to overwrite the existing programs or modify the configurations of the devices.
- **Disrupting the communications:** Sending invalid commands, delaying the response or reply between the devices, performing packet loss, overflow in the communication path, modifying the transmitted readings, and sending misleading values to the system operator are the attacker interests.
- **Compromising the devices:** Virus, Worms, and Trojans are used to compromise the devices. Through the compromised devices, the attacker controls the operations of the industrial plant.
- **System-related threats:** Exploit software/configuration vulnerabilities of the SCADA system (buffer overflow due to illegal packet size, exploiting bug, stack overflow, misconfigured radio network).

Table 17.1 Command and Control Systems attack incidents

Year	Place	Attacks
1982	Russian-Siberian pipeline	Trojan was used to take control of the SCADA system [9]
1992	Chevron refinery in Richmond, California	Former employee crippled the functionality of the alert system by hacking the computers [10]
1994	Salt River Project in Arizona	Gained access into the network through dial-up modem. Controlling the water channels and collecting employee's log-in and password and customer information were the attacks observed [11]
1997	Worcester Airport, Massachusetts	Accessed and crippled the service of computer in the telephone company. The telephone service was disconnected for airport departments [12]
1999	Gazprom in Russia	Hacker was associated with the employee of the gas company to take control of gas flow [13]
1999	Bellingham, Washington	Leakage in the gas flow; disabling the monitoring and controlling functionality of control systems [14]
2000	Queensland, Australia	A former employee has taken control of Maroochy Shire water control system. The huge amount of sewage water was spilled into rivers, hotels, and parks [15]
2001	California	The attack was to gain access into the computer network of the command and control system [12]
2003	Ohio, USA	Worm (SQL Slammer) was used to mask the display and plant processes of Davis-Besse nuclear power plant [16]
2003	Florida, USA	Virus "Sobig" was used to turn off the train signaling systems of the CSX Corporation (transportation service provider) [12]
2004	Gulf of Mexico	Virus "Sasser" was used to perform a buffer overflow attack and propagate the vulnerability to other systems [17]
2007	Willows, USA	The dismissed employee installed an illicit software on Tehama-Colusa Canal Authority SCADA system [18]
2010	Natanz, Iran	Virus "Stuxnet" was used to exploit zero-day vulnerabilities. The target was Siemens PLCs. Changing the computer code and altering the speed of centrifuges were the attacker objectives [19]
2011	America, Europe, and Asia	"Night Dragon" Trojan social engineering was used to gather the operational blueprints and project financial information of control systems [20]

(continued)

Table 17.1 (continued)

Year	Place	Attacks
2012	America, Europe, and Asia	Virus "Duqu" was used to gather the control system data [21]
2012	Iran, Lebanon, Syria, and Sudan	Virus "Flame" was used to steal sensitive information, sniff the traffic, and capture the screenshots [22]
2012	Canada	Chinese hackers stole the project files of the product (OASyS SCADA) belonging to the SCADA maker "Telvent" [23]
2014	USA and Europe	Virus "Havex" was used to steal the information from the infected computers by installing remote access tool. The virus was distributed using vendor websites and email attachments [24]
2014	USA	Malware "BlackEnergy" was used to compromise the HMI of control systems [25]
2014	Germany	Social engineering and spear phishing were used to damage the German steel plant (blocked the shutdown functionality of blast furnace) [26]
2015	Finland, the UK, and the USA	The report entitled "SCADA attacks are on the rise" by Dell security team mentioned that attacks on SCADA systems are doubled compared to the previous year [27]
2015	Ukraine	Trojan BlackEnergy is used to cut off the power in several regions of Ukraine [28]. The BlackEnergy backdoor was used to plant a KillDisk component onto the targeted computers
2016	–	Irongate malware, similar to Stuxnet, was used to hack the Siemens industrial control systems [29]
2016	European energy company	Furtim malware is used to create a backdoor on targeted ICS. The intention was to extract data or potentially shut down the energy grid [30]
2016	USA	Using a cellular modem, the attackers compromised the dam's command and control system [2]
2016	San Francisco	Public transport system was infected with ransomware, allowing passengers to ride free during the busy Thanksgiving holiday weekend [31]
2017	Austria	Hacking attack that took over the hotel's entire electronic key management system. Hotel's computer system and payment systems were also compromised [32]
2017	–	Researchers simulated a piece of ransomware taking control of a water treatment plant and poisoning a city's water supply [33]

- **Insider attacks:** Gaining access rights, getting the credentials of engineer/operator, stealing the passwords, altering the employee data, and manipulating the access list are the attacker interests.
- **Attack on acquisition data:** The attacker may try to control the collection of valid entries in the logs or may alter/delete the recorded entries.
- **Crashing the devices for a period of time:** Access the devices, keeping the device in busy mode, and disrupting the normal traffic flow are attacker interests.
- **Attacks on SCADA systems:** The primary targets include the master, field devices, and communication paths. The following are the possible attacks:
  Master level: Violating authorization, data modification, DOS attack, bypass control, information leakage, illegitimate use, physical attack, resource exhaustion, theft, tunneling, introducing virus, worms and Trojan horse, and unauthorized access.
  Communication link: Data modification, eavesdropping, replay attack, man in the middle attack, rerouting the messages, sniffing, and traffic analysis.
  Field level: Violating authorization, DOS attack, data modification, sniffing, spoofing, and physical attack.

17.5 Possible Attacks on Modbus TCP and DNP3 Protocols

Modbus [34] and DNP3 [35] are the commonly used communication protocols to connect industrial devices. Modbus is a master-slave protocol. Communications are polling based. The master device sends a request message to the slave device. Upon receiving the request, the slave device sends either a normal response or an exception. It is predominantly used in the gas and oil sectors [36]. Recently, Modbus has been extended to support the TCP/IP stack (Modbus TCP) [34]. The protocol is simple and reliable, but it does not provide any security feature (authentication and confidentiality). Messages are exchanged in plain text. This leads to the possibility of security attacks in the industrial networks [36].

DNP3 consists of four layers (application, pseudo-transport, data link, and physical). DNP3 is commonly used in North America for power grids and oil refiners [35].

- **Possible Attacks on Modbus TCP protocol:** Table 17.2 shows the Modbus TCP attacks with respect to TCP, and Table 17.3 shows the Modbus TCP attacks with respect to Modbus TCP.
- **Possible Attacks on DNP3 Protocol:**
  Table 17.4 shows the DNP3 attacks with respect to data link layer. Table 17.5 shows the DNP3 attacks with respect to pseudo-transport layer. Table 17.6 shows the DNP3 attacks with respect to application layer.

Table 17.2 Modbus TCP attacks with respect to TCP [37]

Modbus TCP attacks	With respect to TCP
Irregular TCP framing	The intention is to jeopardize by creating improperly framed messages
TCP FIN flood	Intentionally terminate the TCP connection by setting a FIN flag
TCP RST flood	Intentionally reset the TCP connection by setting a RST flag
TCP pool exhaustion	Prevent the Modbus device to accept new connections. The intention is to exhaust the connections to achieve DOS attack

Table 17.3 Modbus TCP attacks with respect to Modbus TCP

Modbus TCP attacks	With respect to Modbus TCP
Broadcasted message spoofing	Broadcasting false messages to slave devices
Replaying	Attacker fools the devices by resending the stored data
Controlling the devices	Behave as master and directly control the slave devices
Perform malicious activities with function code (FC) and subfunction code (SFC)	FC: 08 and SFC:0A—clears all counters and diagnostic registers of the addressed field device. FC: 08 SFC: 01—enables remote restart FC: 17—return the status information of the addressed field device
Network scanning	Acquiring information about the field devices by sending favorable message to all addresses
Delay in response	Introducing delay in response, in slave to master communication. FC:08-04—enforce listen mode
Retrofitting the device	Man-in-the-middle attack by introducing the device in the unprotected communication path
DOS attack	Flood with large number of Modbus packets with invalid CRC A large number of packets with invalid CRC may crash or make the system go to idle state

17.6 Modbus Packet Analysis Using Wireshark Tool

The following section gives the analysis made on the captured packets (Fig. 17.3).
Number of packets: 21,159
Protocol: Modbus

- To check the request and reply packets of the master and slave devices, the following filters can be used:
  Filter the specific source IP: ip.src==10.0.0.57
  Filter the specific destination IP: ip.dst==10.0.0.3
- What is the time taken (delay) for the request and response packets (response time)?
  In Fig. 17.4, the time difference between 7th (query) and 8th (response) packets is 0.000792 s.

Table 17.4 DNP3 attacks with respect to data link layer [38]

DNP3 attacks	With respect to data link layer
Modifying the destination address	Send the request message to other devices (reroute the request)
False data broadcasting	Send erroneous data to all devices
Masking the available function	Function code: 14/15 is used to send a message to the master. It indicates functionality is not implemented or not functioning in the outstation device
Length overflow attack	Specifying invalid length in the length field of the message
Reset the device	The attack uses the function code: 1 to make the targeted device restart
Masking the device	The attack sets the DFC flag to 1 to intimate the master that the slave is busy and cannot handle the request

Table 17.5 DNP3 attacks with respect to pseudo-transport layer

DNP3 attacks	With respect to pseudo-transport layer
Fragmented message interruption	Interrupt the reassembly process of fragmented messages by setting the FIR and FIN flag with invalid time
Alter the transport sequence	Inserting the series of frames to cause processing errors

Table 17.6 DNP3 attacks with respect to application layer

DNP3 attacks	With respect to application layer
Perform malicious activities with function codes	DNP3 message with function code (FC) to outstations FC:02 $\Longrightarrow$ writes data objects to an outstation FC:09/10(No ack) $\Longrightarrow$ freeze and clear the data objects FC: 17 $\Longrightarrow$ reinitialize data objects FC: 18 $\Longrightarrow$ terminate the running applications FC: 21 $\Longrightarrow$ damage the unsolicited messages from slave to master
Send invalid internal indications (IIN) (16 bit)	Attacker sets the bit, which is reserved to intimate the master that the configuration file of the outstation is corrupted

- In SCADA system, packet transmission rate is periodic in nature (Sect. 17.2.1). By observing the response time, we can classify the communication patterns as normal (N)/retransmission (R)/miss (M)/abnormal (A) as follows (Fig. 17.5):
  - **Normal (N):** This state indicates that the communication pattern between the devices is normal. If the timing difference between the request and response

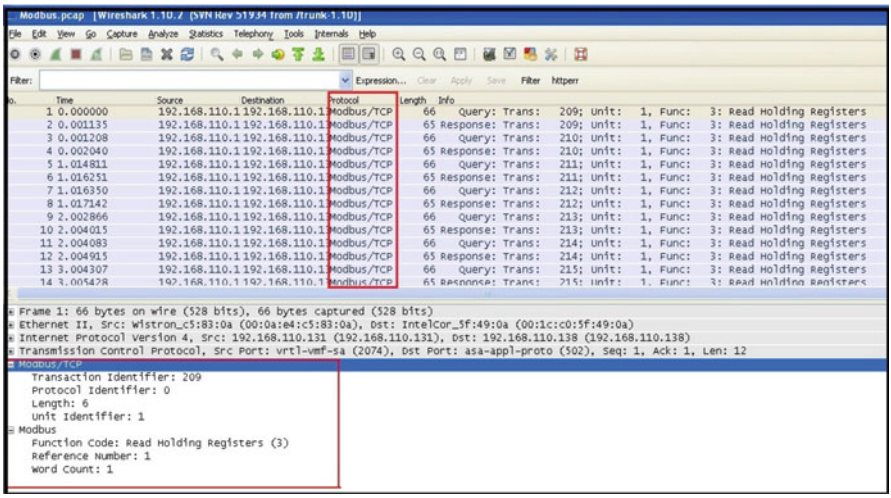


Fig. 17.3 Modbus packets

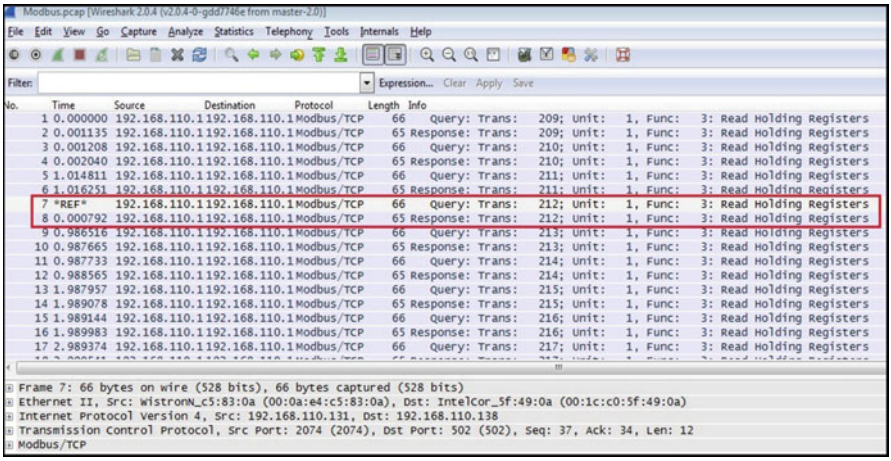


Fig. 17.4 Response time between the request-reply packets

packets is within the normal threshold time (NT), then it is classified as normal packet (Case 1), i.e., WaitingTime (WT) = (T2-T1)<=NT.

SCADA systems are static in nature (Sect. 17.2.1). The IP addresses assigned to the systems are not changed frequently. These features help to consider some more parameters to classify the packet as normal or not. Additional parameters that can be considered along with the timing parameter are IP address, packet size, protocol used, and transaction ID.

Cases	Communication between the two devices [Source IP – Destination IP]	Request/Query Message (Rq)	Reply Message (Rp)	Time Difference	State Estimation			
					N	R	M	A
1	D1-D2	Rq @ time T1	Rp @ time T2	T2-T1<=NT				
2	D1-D2	Rq @ time T1 Rq @ time T2	- Rp @ time T3	NT<T3-T1<=RT				
3	D1-D2	Rq @ time T1 Rq @ time T2	- -	WT>RT				
4	D1-D2	Rq @ time T1	Rp @ time T2 Rp @ time T3 Rp @ time T4 ⋮	T2-T1<=NT				

Fig. 17.5 State estimation

The packet can be classified as normal packet, if the packet contains valid IP addresses, valid protocol, valid packet size, and same transaction ID between the request and reply packets.

- **Retransmission (R):** If the timing difference between the request and response packets is greater than the normal threshold time but less than the retransmission threshold time (RT), it is classified as retransmission packet, i.e., NT<(WT=(T3-T1))<=RT. Reaching this state does not mean that there is malicious activity, but normal communication patterns are missing.
  The reasons for retransmission states are as follows: device was not ready to handle the request due to congestion or poor communication link or due to security attack (Case 2).
- **Miss (M):** The sender sends the request, but does not receive any reply from the receiver, leading to missing state. In this case, because of no reply, it is not possible to calculate the timing difference. Instead, if the waiting time is greater than retransmission threshold time, i.e., WT > RT, the packet is classified as missing state.
  The reasons for missing state are congestion or poor communication link or packet drop (Case 3).
- **Abnormal (A):**The sender sends the request, but receiver tries to send number of duplicate response messages in short span of time. In this case, the timing difference between the request and the first response is within the normal threshold time, but without request messages the receiver has sent multiple replies, this leads to the abnormal state. Flooding number of packets in a short span of time leads to DoS attack (Case 4).

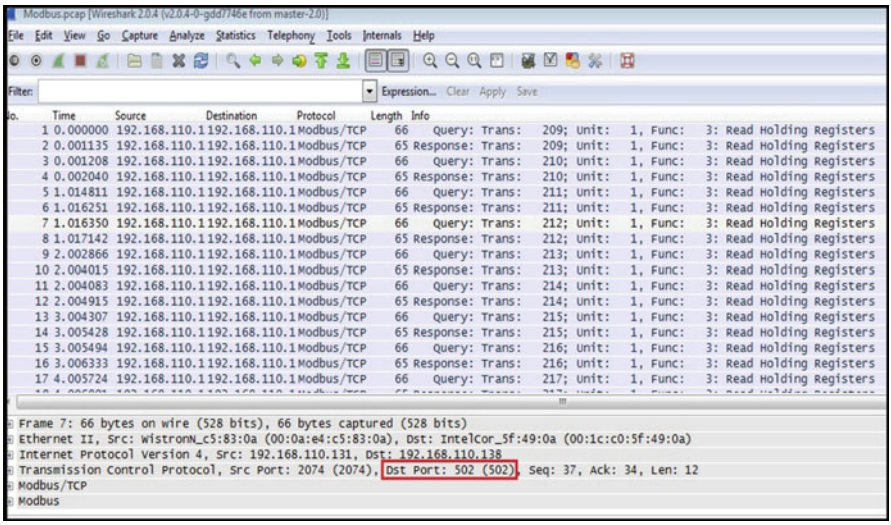


Fig. 17.6 Modbus port number and payload size

- The port used for communication (Modbus uses port 502) and the payload that is limited to at most 253 bytes in Modbus communications can be observed using Wireshark (Fig. 17.6).
- The function codes between the request and response message can be observed by using the Filter: modbus.functioncode==“function code number.” The function code number could be 3, 5, 18, etc. If any packet contains invalid function code, it shall be considered as invalid/malicious packets.
- IO graph: Wireshark IO graphs show the overall traffic seen in a capture file which is usually measured in bytes per second (Fig. 17.7).

17.7 Conclusion

SCADA systems are being the part of critical infrastructures. The proliferation of security attacks and cybercrime incidents on SCADA systems enforcing the industries to consider security is a critical issue. In this paper, attack incidents occurred on SCADA systems (from the year 1982 to 2017) is listed. The attacker goals on SCADA systems are discussed in general. The possible attacks on Modbus TCP protocol are analyzed using the Wireshark tool.

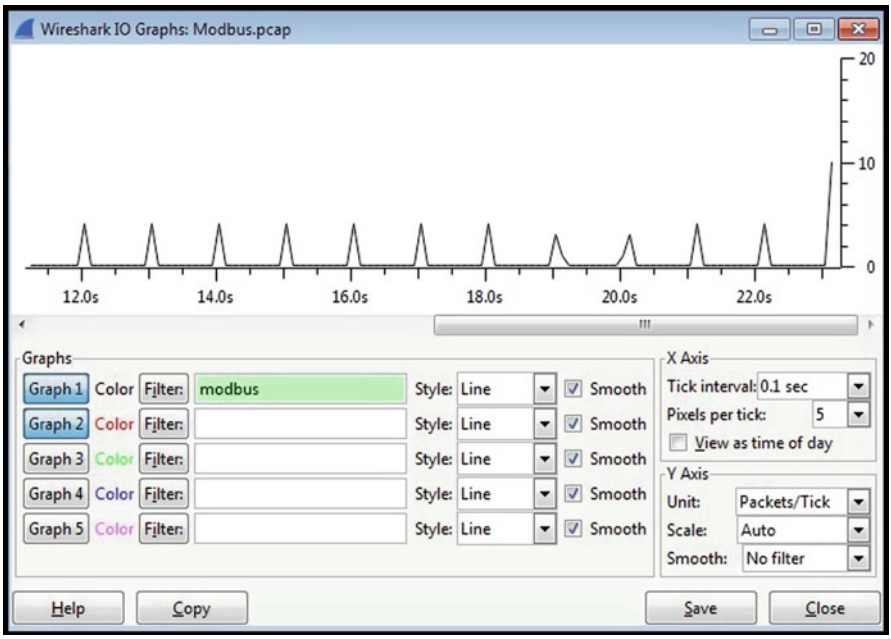


Fig. 17.7 IO graph

References

1. Spellman, F.R.: Energy Infrastructure Protection and Homeland Security. Bernan Press, Lanham (2016)
2. McMillen, D.: Attacks targeting industrial control systems (ics) up 110 percent. https://securityintelligence.com/attacks-targeting-industrial-control-systems-ics-up-110-percent/. Accessed July 2017
3. Sastry, S., Cardenas, A.A., Roosta, T.: Rethinking security properties, threat models, and the design space in sensor networks: a case study in scada systems. Ad Hoc Netw. **7**, 1434–1447 (2009)
4. Dacey, R.F.: Critical infrastructure protection: challenges and efforts to secure control systems: gao-04-628t. GAO Reports **1**, 29–30 (2004)
5. Google dorks. http://www.exploit-db.com/google-dorks/. Accessed 1 Feb 2017
6. Shodan. https://www.shodan.io/. Accessed 1 Feb 2017
7. Wilhoit, K.: Who is really attacking your ics equipment. Trend Micro Incorporated (2013)
8. Sunitha, N.R. et al.: Kmi for scada and wirelesshart in iacs. In: 2015 IEEE 20th Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1–4. IEEE, New York (2015)
9. Daniela, T.: Communication security in scada pipeline monitoring systems. In: Roedunet International Conference (RoEduNet), 2011 10th, pp.1–5. IEEE, New York (2011)
10. Denning, D.E.: Cyberterrorism: the logic bomb versus the truck bomb. Glob. Dialogue **2**(4), 29 (2000)
11. Turk, R.J., et al.: Cyber Incidents Involving Control Systems. Idaho National Engineering and Environmental Laboratory, Idaho Falls (2005)
12. Miller, B., Rowe, D.: A survey scada of and critical infrastructure incidents. In: Proceedings of the 1st Annual Conference on Research in Information Technology, pp. 51–56. ACM, New York (2012)

13. Tsang, R.: Cyberthreats, vulnerabilities and attacks on scada networks. University of California, Berkeley, Working Paper (2010). http://gspp.berkeley.edu/iths/Tsang_SCADA%20Attacks.pdf (as of 28 Dec 2011)
14. Mustard, S.: Security of distributed control systems: the concern increases. Comput. Control Eng. J. **16**(6), 19–25 (2005)
15. Stamp, J., Dillinger, J., Young, W., DePoy, J.: Common vulnerabilities in critical infrastructure control systems. SAND2003-1772C. Sandia National Laboratories (2003)
16. Nicholson, A., Webber, S., Dyer, S., Patel, T., Janicke, H.: Scada security in the light of cyber-warfare. Comput. Secur. **31**(4), 418–436 (2012)
17. Canavan J.: The evolution of malicious irc bots. In: Virus Bulletin Conference, pp. 104–114 (2005)
18. Cardenas, A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., Sastry S.: Challenges for securing cyber physical systems. In: Workshop on Future Directions in Cyber-Physical Systems Security, p. 5 (2009)
19. Hernández Jiménez, J., Chen, Q., Nichols, J., Calhoun, C., Sykes, S.: Towards a cyber defense framework for scada systems based on power consumption monitoring. In: Proceedings of the 50th Hawaii International Conference on System Sciences (2017)
20. Night dragon. http://www.pcworld.com/article/219251/article.html. Accessed 17 Jan 2017
21. Misra, S., Maheswaran, M., Hashmi, S.: Case studies of selected iot deployments. In: Security Challenges and Approaches in Internet of Things, pp. 77–94. Springer, Berlin (2017)
22. Cyberwars. https://www.rt.com/news/flame-stuxnet-kaspersky-iran-607/. Accessed 17 December 2017
23. Maker of smart-grid control software hacked. https://www.wired.com/2012/09/scada-vendor-telvent-hacked/. Accessed 17 January 2017
24. Meshram, A., Haas, C.: Anomaly detection in industrial networks using machine learning: a roadmap. In: Machine Learning for Cyber Physical Systems, pp. 65–72. Springer, Berlin (2017)
25. Meara, K.O., Shick, D., Spring, J., Stoner, E.: Malware capability development patterns respond to defenses: two case studies (2016)
26. Green, B., Prince, D., Busby, J., Hutchison, D.: The impact of social engineering on industrial control system security. In: Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy, pp. 23–29. ACM, New York (2015)
27. Flowers, A.S., Smith, S.C., Oltramari, A.: Security taxonomies of industrial control systems. In: Cyber-security of SCADA and Other Industrial Control Systems, pp. 111–132. Springer, Berlin (2016)
28. Khan, R., Maynard, P., McLaughlin, K., Laverty, D., Sezer, S.: Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid. In: 4th Int'l Symposium ICS & SCADA Cyber Security Research. BCS, pp. 53–63 (2016)
29. https://thehackernews.com/2016/06/irongate-stuxnet-malware.html/. Accessed 17 January 2017
30. Leyden, J.: Scada malware caught infecting european energy company. https://www.theregister.co.uk/2016/07/12/scada-malware/. Accessed July 2017
31. Ransomware attack on san francisco public transit gives everyone a free ride. https://www.theguardian.com/technology/2016/nov/28/passengers-free-ride-san-francisco-muni-ransomeware. Accessed July 2017
32. Hackers take over a hotels computer system, lock guests in rooms and hold hotel to ransom. goo.gl/9JRsOA. Accessed July 2017
33. Rsa 2017: Researchers create ransomware for industrial control systems. goo.gl/eYPxxY. Accessed July 2017
34. Goldenberg, N., Wool, A.: Accurate modeling of modbus/tcp for intrusion detection in scada systems. Int. J. Crit. Infrastruct. Prot. **6**(2), 63–75 (2013)
35. Amoah, R., Camtepe, S., Foo, E. Formal modelling and analysis of dnp3 secure authentication. J. Netw. Comput. Appl. **59**, 345–360 (2016)

36. Ramos, R., Barbosa, R.: Anomaly detection in SCADA systems-a network based approach. PhD thesis, Centre for Telematics and Information Technology, University of Twente (2014)
37. Huitsing, P., Chandia, R., Papa, M., Shenoi, S.: Attack taxonomies for the modbus protocols. Int. J. Crit. Infrastruct. Prot. **1**, 37–44 (2008)
38. East, S., Butts, J., Papa, M., Shenoi, S.: A taxonomy of attacks on the dnp3 protocol. In: International Conference on Critical Infrastructure Protection, pp. 67–81. Springer, Berlin (2009)

Chapter 18
Security Threats and Solutions for Virtualization and Migration in Virtual Machines

N. Ravi and N. R. Sunitha

Abstract Cloud has been a dominant player in information technology in the recent years. The main composing factor of cloud is virtualization among many others. Due to the widespread acceptance of cloud, the researchers have identified that the area is a favorable habitat for the attacks. The attacks and threats always have poured gloomy clouds on the user raising the concern about their privacy and security. The virtualization technology dates back to the advent of mainframe computers and has been the area of refinement over the years. Since the cloud is built on obsolete virtualization technology, it eases the combination of attack pattern for the malicious user. Also, the migration of virtual machine (VM) holds key status in the effective management of data centers. As the virtualization technology imbibed on cloud is obsolete, the migration technique is also prone to attack and ineffective management. This paper discusses the threats and associated attacks pertaining to virtualization and migration in VM and proposes a new framework which will enhance the security of virtualized environment. The proposed framework handles the migration of VM in an effective manner.

Keywords Virtualization · Virtual Machine (VM) · DASDVM · Security threats · Migration · Cloud computing threats · SLA

18.1 Introduction

Cloud computing or simply cloud refers to the technical setup of hardware resources and software implementation which makes realization of several crucial business requirements such as availability, performance, security, scalability, accessibility, etc., to run a business effectively. The kind of freedom of operation was facilitated for the cloud by the advent of new technologies such as virtualization, service-

N. Ravi (✉) · N. R. Sunitha
Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, Karnataka, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_18

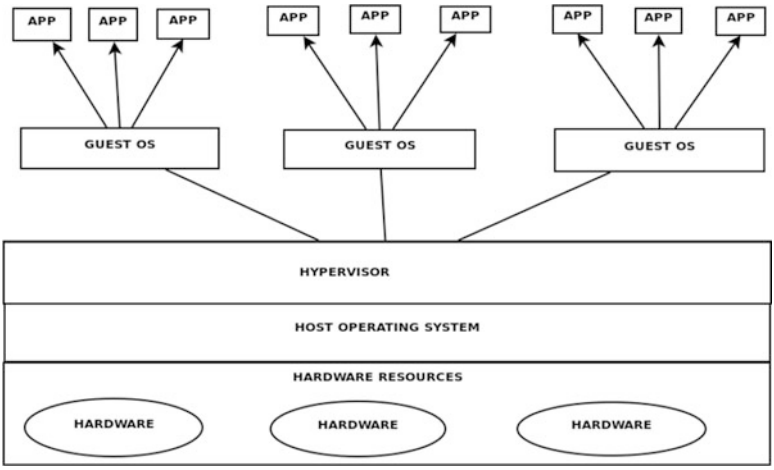


Fig. 18.1 Hosted virtualization

oriented architecture, automatic computing, and web services, among others. Virtualization is a remarkable and noteworthy technology that separates physical resources and creates an illusion of dedicated resources of the hardware/software which will cater to the needs of the user. This is done with the help of hardware, software, or combination of both. A virtual machine monitor(VMM)/hypervisor is a piece of software or hardware that instantiates the virtual instances that the VM emulates a guest operating system (OS) and presents a distinct instance of the host operating system to mimic the fully functional OS. The VMM is an intermediate component and a lightweight module implemented either by software or hardware [4]. This is illustrated in Fig. 18.1. Based on the type of implementation, virtualization can be broadly of the following breed, namely, hardware virtualization and software virtualization. Virtualization can also be done on several layers such as hardware, software, memory, data, network, and storage.

18.2 Security Issues and Categorization

Although, the virtualization has paved the way for the tremendous ascendancy of cloud computing, industry experts and researchers put forth the view of concerns for the security [8]. The security definition has evolved over the years considering the scale, usage, context, and other several parameters. The transition has been from an amoebic fringe to layered multidimensional scale. Traditionally security was coined around the terms integrity, confidentiality, and availability. These revolved around and referred to data in general. Today's Internet has grown beyond data without losing the centrality to data. Concerns that are originating and prevailing in recent times lie in layers such as architectural issues, network issues, and legal

issues. Among the security issues, we concentrate on virtualization mainly because of the minimal work done on the particular domain of virtualization and as it is one of the building blocks and major contributing factor which is binding the cloud architecture today. We can identify the security issues such as VM image splitting, VM isolation, VM sprawl, VM rollback, VM escape, VM migration, hypervisor issues, cross-VM side-channel attack, VM creation attack, and VM scheduler attack [1, 7, 9] and more on the virtualization front. The acquaintance of each of the concerns is as follows.

VM image splitting: A VM image repository is used to fire up the process of instantiation. A VM image repository could be used by malicious user to access the VM image and do changes which could potentially harm the repository base and also can compromise the multi-tenant environment.

VM isolation: A VM needs to be isolated from one another in order to create a safe multi-tenant environment. Although the VMs are logically differentiated, they access the same underlying hardware resource.

VM sprawl: It is a condition wherein a number of VMs are instantiated on the host which are growing exponentially, and most of them are either underutilized or idle which does not serve any purpose. This leads to poor resource utilization.

VM rollback: Rollback is a technique where the VM can be reverted back to the last known good state. This can be done with the help of log files. This helps the VM in case the migration process fails. The rollback provides the user the freedom and resilience. On the other hand, this process can revert back the privacy and configuration settings that were previously disabled.

VM escape: It is a condition where the VM escapes or gets abducted from the control of hypervisor or VMM. This condition affects the underlying hardware resource and makes the hardware resources available to the malicious user; the user can do demonic acts with the access to the resource. The IaaS model is the most affected model with this condition.

Hypervisor issues: The influential piece of software that manages the VM is the hypervisor or VMM. It beholds the duty of orchestrating the process of virtual resource allocation and management. If this module is compromised, then there are endless possibilities for the intruder to carry out. The intruder always looks for the loopholes of the VMM or the weak entry point to gain access to VMM.

Cross-VM side-channel attack: This condition can occur in the VMs dwelling on the same physical host as that of attacker where in the information such as keys, resource spending is being exposed and stolen by the attacker.

VM creation attack: A malevolent code could be placed inside the VM image before the instantiation process; thereby when the VM creation process outsets, the resultant VM is an outcast.

VM scheduler attack: Loopholes in scheduler can result in theft of service or resource abduction. For example, the VM's scheduler is scheduled to run at a particular instance preserving the credit balance of the time slice.

VM migration: This is a mechanism of transferring the VM from one physical host to another without shutting the VM down. Summarizing the security threats

pertaining to the virtualization layer, all the abovementioned pitfalls raise the concern in a substantial way.

18.2.1 Migration of VM

This is the process of transferring the VM from one host machine to another host machine so that the process of execution does not halt. There are two types of migration: live migration and non-live migration. In non-live migration, the VM from one machine is shutdown, copied to physical machine, and then rebooted at the host machine. This kind of migration has downtime. On the other hand, live migration is a process carried out without the downtime. Live migration is carried out in phases, namely, push phase, stop and copy phase, and pull phase. Push phase: the memory pages are copied which are unmodified/fresh from host to destination machine, while the VM is still running. Stop and copy phase: as the name indicates, the hypervisor stops the execution of VM and copies the remaining pages to the destination machine, and then the VM is instantiated. Pull phase: the new VM is instantiated in destination machine; when a page is being reached out and it is not present, then a page fault occurs, and it is been copied from host machine. VM migration is the core process to maintain the health of the data center. During migration, disk state, network state, and memory states are copied from host to destination machine. VM placement is a process of placement of VM for effective utilization of the data center. VM placements take into account intra- and inter-data center traffic, energy consumption of the machine, locality, and SLA compliance factor in order to provide high availability, load balancing, high reliability, and security.

18.3 Security Repercussions of the Compromise on Cloud Infrastructure

The security attack on cloud could give rise to security breaches such as data integrity violation, denial of service, manipulation of data, data ransom, data duplication, service-level agreement (SLA) violation, availability of data, confidentiality of data, trust issues, legal issues, accountability of resources, transparency, theft of service, and many more new concerns that are alarmingly increasing periodically.

Data integrity: A malevolent user who holds the access to data repository and manipulates the data in any form. In this case, data is said to lose its integrity. Data integrity refers to the consistency and accuracy of data in its original form. The attack on virtualized environment on cloud could lead to data integrity issues.

Manipulation of data: Certain protocols which are used for programming interface such as REST, SOAP, and HTTP are vulnerable to communication threats, and

the user can make use of these loopholes to gain access to the environment and utilize this environment to manipulate the data.

Availability of data: Data is the necessary unit for any transaction or process to sail smoothly in any corporation. Data must be available all the time for handling of the processes. If an attacker has the possession of data repository, they can hamper the availability of data.

Denial of service: Cloud services are hampered by the possession of resources by the attacker, and when the requests for a resource come in from the user who needs it, the service request could be denied because of the illegal possession of the resources by the attacker.

Theft of service: The malignant user may get hold of the resources through a weak communication channel or port and steal the resources such as network, VM, computing power, etc. This affects the transparency of the services and creates trust issues between user and the cloud service provider (CSP).

SLA violation: Service-level agreement is a document which states the terms and conditions to avail the service and also states countermeasures which are to be taken when they are violated. SLA violation refers to the breaching of the contract and not abiding by it. This happens when the malignant user creates a user-specific attack and frames the CSP for the shortage of resources.

Resource accountability: As virtualization creates multi-tenant environment, isolation is a primary requirement to meet the accountability of the resources. The attacks such as DoS (denial of service) and DDoS (distributed denial of service) could assess the resource metering significantly wrong at both CSP and user side.

18.4 Literature

The literature provides us with the different works on virtualization and migration in VM [1, 7]. It discusses the kinds of attack that enables the malicious user to gain access to hypervisor. The author also sheds light on the threats pertaining to the virtual environment. The authors in [1] have done a thorough survey on the security issues and the available IDS for mitigating the attacks and the threats. The authors have given the bird's eye taxonomy of the security issues for the different service models [1]. They also discuss the proposed solution in literature in detail. Some of the existing proposals are Mirage, EVDIC, ImageElves, and OPS-offline. Mirage is an image management system where the authors have proposed secure mechanism to share the images with the help of access control framework along with filters to drain out unwanted information [11]. The advantage of Mirage is that it provides secure retrieval and storing of VM images and greatly helps in auditing. The drawback of Mirage is that the image handled is dormant type meaning, while execution, the state of the image might be compromised, but the attack would not be noticed till the image gets updated in the repository. Also, the framework does not comply with privacy and integrity. In [6], authors have discussed about another image monitoring system named EVDIC which uses encryption to secure the VM

image. It uses AES encryption with $k = 256$ bit. EVDIC also stores integrity info of the VM image which is an advantage. However, EVDIC does not have the support for outdated software removal and owner's leftover data removal which are crucial to secure VM image. Authors in [13] have proposed a framework to provide VM security called VNSS. It contains components such as controller and multiple agents to secure the VM. Although the framework is implemented on Xen hypervisor, the scalability of the framework is still questionable. In [12], the authors have proposed a completely radical approach of trusting only the processor chip and the remaining components as untrusted. This framework gives good support for scalability, data privacy, and integration of the system. It takes help of both software and hardware to protect the system. Authors in [5] have used the VM introspection technique to propose Exterior. It is a dual VM architecture that pushes secure virtual machine (SVM) to host the guest VM (GVM). SVM executes kernel which is the same as GVM and redirects the memory state at hypervisor from SVM to GVM. This change gives the impression that the program is being run on GVM. This approach provides concrete security but is compromising on integrity of the hypervisor. The authors in [2] have proposed a technique for migration which is based on trusted computing. If the trust credentials are not favorable during launch process of VM, then the migration will not occur. The advantage of this approach is remote auditing facility. Authors in [10] propose a new architecture to overcome the drawbacks of old virtualization technology. This architecture introduces new blocks on the existing virtualization stack. This introduction provides security to some level but fails if the initiating hypervisor itself is affected putting all the VMs prone to attack. Taking the inspiration from the same architecture [10], authors in [3] have proposed a framework which introduces cryptographic encryption between VMM and the hardware and also make use of honeypots between VM and VMM. They bring in the extra layer of security between the hardware and hypervisor but also introduce delay for the key exchange while accessing the resource. All the above work in literature where in the solutions have been proposed keeping in mind only one parameter (e.g., hypervisor security or migration or VM image security, guest instance security). Nonetheless, the works did provide realistic solution toward security. We try to break the one-factor approach and intend to provide a framework which would provide solution to both virtualization and migration in VM.

18.5 Proposed Solution for Virtualization and Migration

We propose an alternate solution for the existing solution in literature for virtualization environment [10]. We have introduced the new modules upon the existing architecture such as Virtual Machine Integrity Monitor (VIM), Virtual Machine Resource Manager (VRM), and Dynamic Analyzer and Smart Decider for Virtual Machine (DASDVM).

18.5.1 Architecture

The proposed architecture is built on several assumptions and privileges. We assume that the host OS's kernel space can be held through special privilege access and can be accessed without interruption in the proposed architecture. We have introduced security monitors to overlook the protocols and processes in the VM and also VMM. Figure 18.2 illustrates the framework. The hypervisor communicates with the host OS which in turn communicates with the hardware through the kernel. The user could experience a delay, every time the hypervisor is listening to the hardware to nurse the needs. Hence we created a pool of resources that holds hardware-specific instantiation modules. This module helps to nurse hypervisor in case of hardware requirement. This module is placed in the kernel space of the host OS to efficiently carry out the process. Every time the VM needs a hardware resource, instead of giving interrupt signals to the hardware, the hypervisor could make use of the hardware device repository (HDR) to nourish the needs. The HDR is implemented as lightweight add-on module which can fire up the process of resource instantiation. We have proposed additional modules for providing additional security and smart management of the VM and VMM, namely, DASDVM, VIR, and VRM. The VRM handles the resource management of the VM and feeds the information to DASDVM, so that it can take intelligent decisions. The decisions will be based on the parameters such as current execution state, number of resources (clock cycles, cache, page hits, etc.), network congestion, and many more. The VRM listens to the status of the VM and records all the information. On the other hand, VIM is a security supervisor who checks the status of VMs, and if there are any possible attacks on any VM running on the same hardware, then it flags the VM and sends the information to DASDVM to not use the VM for further use and suspend the VM.

18.5.2 Components/Modules

DASDVM: The Dynamic Analyzer and Smart Decider for Virtual Machine is a module housed in the hypervisor to analyze the traffic, CPU cycles, workload, health of VM, and resource utilization of the data center. This is a logical analyzer and decision-making module that run when the hypervisor executes. This is a two-tier behavior analyzer and decision-maker. The functionality of the DASDVM is varied based on the parameters and the state of the machine. HIM: The Hypervisor Integrity Manager is a security monitor for the hypervisor that evaluates the security attributes. It maintains the log of hypervisors in the network to efficiently instantiate or shutdown the hypervisor based on the health of the VMM. VIM: The Virtual Machine Integrity Monitor is placed on the each virtual instance/guest OS. This is a lightweight module placed to monitor the cleanliness of the VM. VRM: The Virtual Machine Resource Manager is placed on each guest OS. It monitors the resource usage and the management of the VM.

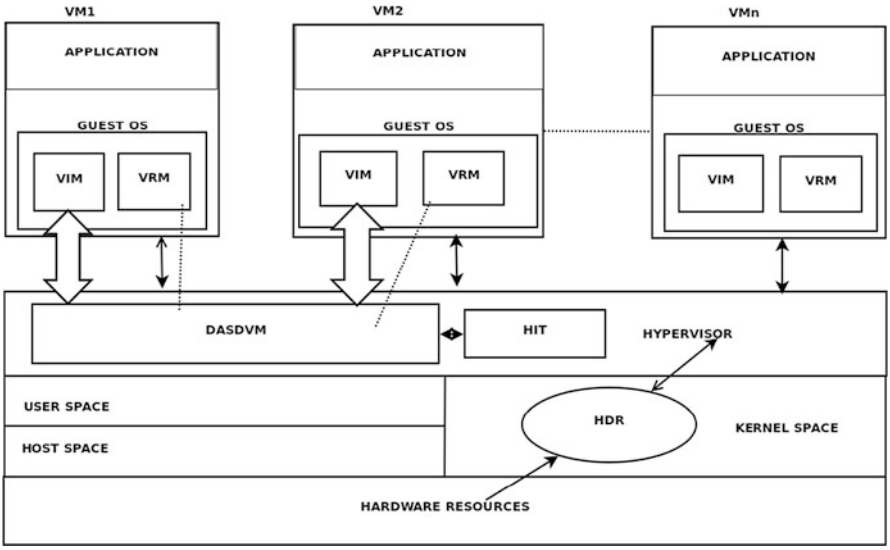


Fig. 18.2 Secure virtualized framework

18.5.3 Implications of Proposed Architecture on the Threats

The proposed architecture provides solution for the issues such as cross-VM side-channel attack, denial-of-service attack, and migration issues. The Cross-VM Side-Channel Attack: Cross-channel attack could be the implication of co-residency of VMs on the same hardware unit. The cross-VM side-channel attack could expose the information of cryptographic keys and network info to the attack user residing on the same hardware. Our approach consists of VIT module which indicates the worthiness of VM (VIT has integrity value which is stored as hash value and stored in the table), and it is fed to the DASDVM at periodic interval. This interval depends on the load of the DASDVM. Usually the time interval to feed the information is 7 s. If the value is low beyond the threshold (below 3, and the flag for high utilization from VRM is set), then sharing of information and resources between the VMs is prohibited by DASDVM, as this could indicate a possible compromise of the VM. Thus the VM is blocked from participation in the network. Denial of Service: DoS attack implies the resource wastage. The resources are being held by attackers who just pull out the resource anonymously. The resource, on one hand, is being wasted and, on the other, cannot be used when there is an actual need. Our approach takes care of this situation with the involvement of VRM module where it constantly monitors the usage of VM (network, bandwidth, data space). If the consumption is more compared to the previous needs (info log is maintained at DASDVM), then the DASDVM raises a ticket, and the user for which the VM is allocated should reply with the reason for the exceeding usage. The usage limit is set keeping in mind the previous instances (if the user is first timer, then the quota limit is set by

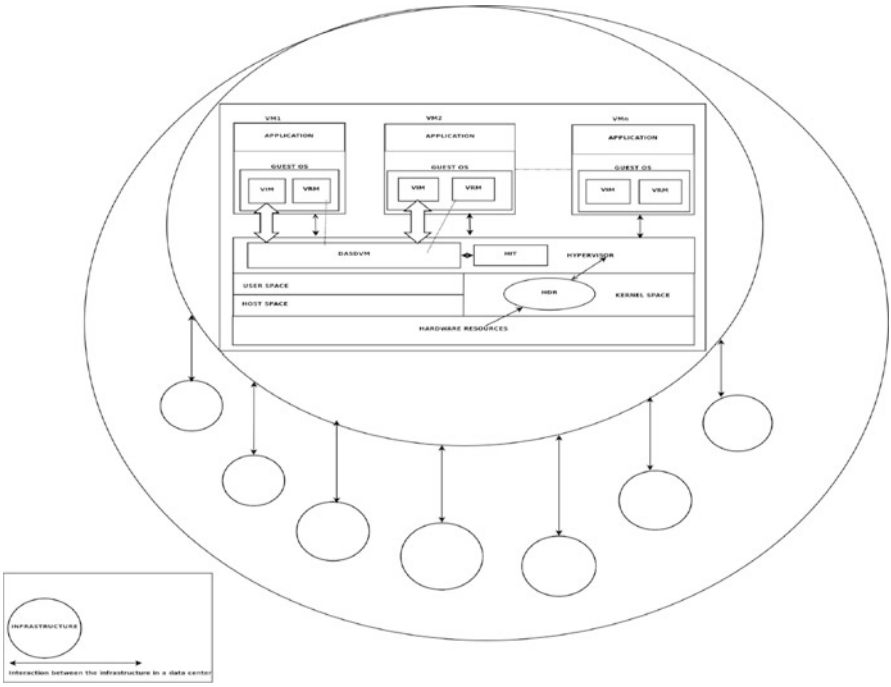


Fig. 18.3 Secure migration framework

the user). Likewise we can also solve resource accountability problem. Migration Issues: Migration of VM involves intricate parameters. Migration is carried based on network usage, legal issues, and architectural issues. Our approach, with the help of modules such as DASDVM and VRM, mitigates the migration challenges so as to effectively manage the data center. If the malicious attacker gains the access to VM and chooses to instantiate the VMM, then the running VM will be malignant. In order to avoid this situation during migration of VM from one network to another, HIM and VIM will be of significant use. The HIM indicates the purity of the hypervisor based on the values in the table (stored as hash value and contains the network ID, hypervisor ID, and the value). Likewise VIM indicates the purity of VM. The DASDVM will take the decision about the instantiation or management of hypervisor based on this information. While migrating, if the HIM/VIM value does not comply with the threshold value, then the migration will not occur, and the ID value of that particular hypervisor in the particular network will be blacklisted. This decision is taken care by DASDVM. This is illustrated in Fig. 18.3. The VRM gets the resource usage of the individual VM, and this info is fed to DASDVM. The smart decision-maker analyzes all these info and communicates with other network hypervisors (particularly DASDVM) and makes decision which will place the VM so that less overhead is incurred and effective resource utilization is also achieved.

18.6 Conclusion

In this paper, we have presented the security threats pertaining to virtualization and migration and the associated repercussions with the security issues of the respective layers. We have discussed the solutions in literature for virtualization and migration. We have proposed our own framework to handle the virtualization and migration. The introduction of add-on modules such as VIM, VRM, HIM, HDR, and DASDVM have improved the virtualized environment as well as support good live migration process. There is a scope of improvisation for this framework and is a work for future enhancement.

References

1. Ali, M., Khan, S.U., Vasilakos, A.V.: Security in cloud computing: opportunities and challenges. Inf. Sci. **305**, 357–383 (2015). https://doi.org/10.1016/j.ins.2015.01.025
2. Aslam, M., Gehrmann, C., Bjorkman, M.: Security and trust preserving VM migrations in public clouds. In: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications. IEEE (2012). https://doi.org/10.1109/trustcom.2012.256
3. Bahulikar, S.: Security measures for the big data, virtualization and the cloud infrastructure. In: 2016 1st India International Conference on Information Processing (IICIP). IEEE (2016). https://doi.org/10.1109/iicip.2016.7975336
4. Bauman, E., Ayoade, G., Lin, Z.: A survey on hypervisor-based monitoring. ACM Comput. Surv. **48**(1), 1–33 (2015). https://doi.org/10.1145/2775111
5. Fu, Y., Lin, Z.: EXTERIOR. ACM SIGPLAN Notices **48**(7), 97 (2013). https://doi.org/10.1145/2517326.2451534
6. Kazim, M., Masood, R., Shibli, M.A.: Securing the virtual machine images in cloud computing. In: Proceedings of the 6th International Conference on Security of Information and Networks - SIN '13. ACM Press (2013). https://doi.org/10.1145/2523514.2523576
7. Khan, M.A.: A survey of security issues for cloud computing. J. Netw. Comput. Appl. **71**, 11–29 (2016). https://doi.org/10.1016/j.jnca.2016.05.010
8. Latif, R., Abbas, H., Assar, S., Ali, Q.: Cloud Computing Risk Assessment: A Systematic Literature Review, pp. 285–295. Springer, Berlin (2014). https://doi.org/10.1007/978-3-642-40861-8_42
9. Rawat, S., Tyagi, R., Kumar, P.: An investigative study on challenges of live migration. In: 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO). IEEE (2016). https://doi.org/10.1109/icrito.2016.7784939
10. Sabahi, F.: Secure virtualization for cloud environment using hypervisor-based technology. Int. J. Mach. Learn. Comput. **2**, 39–45 (2012). https://doi.org/10.7763/ijmlc.2012.v2.87
11. Wei, J., Zhang, X., Ammons, G., Bala, V., Ning, P.: Managing security of virtual machine images in a cloud environment. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security - CCSW '09. ACM Press (2009). https://doi.org/10.1145/1655008.1655021
12. Xia, Y., Liu, Y., Chen, H.: Architecture support for guest-transparent VM protection from untrusted hypervisor and physical attacks. In: 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA). IEEE (2013). https://doi.org/10.1109/hpca.2013.6522323
13. Xiaopeng, G., Sumei, W., Xianqin, C.: Vnss: A network security sandbox for virtual computing environment. In: 2010 IEEE Youth Conference on Information, Computing and Telecommunications, pp. 395–398 (2010)

Chapter 19
TASB-AC: Term Annotated Sliding-Window-Based Boosting Associative Classifier for DNA Repair Gene Categorization

A. Vidya, Santosh Pattar, M. S. Roopa, K. R. Venugopal, and L. M. Patnaik

Abstract Damage to DNA affects the biochemical pathways of the cell and leads to aging, if not repaired. Several genes in the genome of an organism are responsible for DNA repair activities, however, not all of them are related to the biological aging process. In this paper, we develop a data mining technique to relate association of DNA repair genes with the aging process of the organism. Nucleotide sequence of the DNA repair genes is annotated with their respective biochemical properties and is then converted to a transactional dataset. Further, biological features are extracted from the dataset by constructing an associative classifier. To select significant gene features, we employ sliding-window technique to divide the gene sequence into subsequences and thus increase their count. An extensive evaluation is performed of the proposed technique by taking human DNA repair genes along with their biochemical properties like gene ontology terms and protein–protein interactions. We also provide biological interpretation of the features extracted from the classification technique.

Keywords Associative classifier · DNA repairs genes · Gene-document · Rule pruning · Sliding window · Subsequence

A. Vidya (✉)
Department of Information Science and Engineering, Vivekananda Institute of Technology, Bangalore, India

S. Pattar · M. S. Roopa · K. R. Venugopal
Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore, India

L. M. Patnaik
Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_19

19.1 Introduction

With aging, there is a significant decline in functional activities of an organism and thus increases the mortality rate in all organisms, from single celled to multicellular age, but they differ in the rate at which aging progresses and the lifespan [1]. Several diseases like neurodegenerative and cancer are offset with the aging process. Thus a better understanding of the biology of the aging process leads us to more insights of these diseases [2]. Several theories have been proposed in the past to elucidate the biological aging process. DNA damage theory states that accumulation of damages to DNA, caused by natural phenomenon, leads to aging. Although DNA has a stable structure, several factors like change in pH, temperature, nucleotides, and exposure to ultraviolet radiation can destabilize it and lead to its damage.

A repair system can try to undo the damage and constitutes of DNA repair pathways where repair proteins, translated by specific DNA repair genes, interact with each other to stabilize the structure of the DNA. The significance of DNA repair system in aging process has been established with the study of progeroid syndromes that are a class of genetic disorders that resemble physiological aging. Diseases that belong to this class are Hutchinson-Gilford Progeria Syndrome (HGPS), Cockayne Syndrome (CS), Werner Syndrome (WS), Rothmund-Thomson Syndrome (RTS), Trichothiodystrophy (TTD), Xeroderma Pigmentosum (XP), Bloom syndrome (BS), and Restrictive Dermopathy (RD). Damages to DNA repair genes are the major factors that lead to a defective repair system and thus the disease itself [3].

Damage to DNA can be caused by sources within the cell (called as endogenous damage), like by-products of certain metabolic pathways that attack DNA at specific sites or an error in replication process of DNA, and sources that are outside the cell (called as exogenous damage), like exposure to harmful radiations, contaminations, and viruses. A cell has a natural evolutionary defense mechanism to repair these damages that are classified into two types, viz., (i) direct reversal mechanism, in which a damage to DNA is modified chemically to regain the original configuration, enzymes like glycosylases and repair alkylation damage to DNA, where a change of base in thymine (T) is replaced with cytosine (C). (ii) Excision repair mechanism, in contrast to above method, the damaged part of DNA is broken down from the parent molecule and replaced with the correct version. This subclass is further divided into three repair mechanisms as nucleotide excision repair (NER), base excision repair (BER), and mismatch repair (MMR) that take help of various identification enzymes to find and replace a specific type of damage [4].

Study of DNA damage gives us a greater understanding of DNA repair pathways and their role in the biological aging process [5]. Although various studies have been carried out in the past to associate DNA damage and aging, it is quite tedious and time-consuming due to the involvement of in vitro and in vivo experiments. In this chapter, a data mining-based method has been proposed to establish the relationship between DNA damage, its repair process, and aging with respect to humans.

Developments in high-throughput sequencing techniques that are fast, low cost, and reliable have resulted in an accumulation of a large amount of genomic and

proteomic data. There is also an increase in the number of aging genes that are identified in recent years. However, it is a time-consuming and tedious task to derive insights from this huge data. Thus, there is a need for computational techniques that can analyze and interpret them at high speed.

Data mining involves such methods that discover some unseen knowledge in a huge database. There are two specific tasks, predictive and descriptive, that form the broad categories into which data mining tasks can be divided. While descriptive tasks present a consolidated view of the complete dataset by presenting unknown patterns, predictive tasks try to fill in an unseen/future data [6]. These two approaches can be combined in the form of associative classifier that makes use of association rules to predict a predefined class of unknown data item. Agrawal et al. [7] introduced association rule mining as a method to extract relationship patterns among the items from the market basket data that contains data pertaining to the transactions made by a customer at a grocery store. Further, these association rules can be used to construct a classifier that can be used to predict the class of an unknown data item [8].

19.1.1 Motivation

Several techniques have been developed in the past that employ classification algorithms to associate DNA repair genes with the aging process; these methods have used classifiers like J48, support vector machines (SVM), and Naive Bayesian classifier [9–12]. However, none of them use gene sequences as an attribute to construct the classifier. Existence of structural and functional information in a gene sequence motivates us to develop a classifier that extracts patterns from these sequences along with other attributes [13].

19.1.2 Contribution

The present work is the extension of our earlier work [14]; here we construct a transactional dataset consisting of gene sequences of DNA repair genes and some of their biochemical properties in terms of gene ontology (GO) terms and interaction of the repair genes in their respective metabolic pathways. Frequently appearing items in the dataset are mined through a well-known approach, FP-growth, and then class association rules (CARs) are constructed from them. A large number of rules are generated in this phase so as to account for biologically important data items that appear less frequently in the dataset. CARs are later subjected to a pruning step, where we make use of boosting technique to eliminate all those CARs that have very low coverage in the dataset. Final rules are then used to construct the classifier that is used to predict the relatedness of DNA repair genes to the aging process. In

addition, we analyze the CARs for biological significance and presented the role of DNA repair system in aging.

19.1.3 Organization

The rest of this chapter is organized as follows: Sect. 19.2 provides the literature survey on the application of data mining techniques to aging. Section 19.3 states the problem definition and gives an overview of the system employed for solving the problem. Section 19.4 discusses of the four-phase algorithm used to construct the classifier. Section 19.5 discusses the dataset used and experiments performed along with the results. Concluding remarks are presented in Sect. 19.6.

19.2 Literature Survey

Several classification algorithms in the domain of data mining have been proposed in the past to extract interesting patterns about DNA damage theory of aging. We provide a brief description of them in the following section.

Fang et al. [11] devised a binary classification model using random forest to categorize aging genes into DNA repair and non-DNA repair class. Protein–protein interactions were considered as features in the classification step. Several attributes of aging genes related to DNA repair were identified as DNA replication and recombination process like cell cycle, death, and functional maintenance. However, gene sequences were not considered as a feature in the classification step.

Salim et al. [15] considered biological sequences as attributes in the classification technique. Modified Apriori algorithm is used to find fixed length subsequence patterns that appear frequently in a sequence of very long length. Through the use of bit wise operations, there is a drastic reduction in search space and thus gives a good classification performance. However, the biological significance of the mined subsequences is not provided.

Becerra et al. [16] also used biological sequences to construct association rules and then constructed classifier from them. Association rules are mined from protein sequences, and a profile is built from them that is then fed to artificial neural network and support vector machines for feature extraction and prediction of protein secondary structure. Parameters of the classifier are difficult to adjust and thus are impractical for sequences of very large length.

Yu and Wild [17] constructed an associative classifier based on a weighting scheme. Each item in the dataset is given a weight through a scoring scheme, and then frequent items are mined to construct association rules. Combinations of HITS and PageRank algorithms are used to assign weights to the data items. The weights considered are generic in nature and do not provide any biological insights.

Yoon et al. [18] proposed a binary classifier through boosting technique for multistage classification problem. Amino acid sequences of proteins are divided into

k-word subsequences and subjected to association rule construction. These rules are then pruned by boosting technique to increase the accuracy of the classifier. No other features apart from protein sequences were used for the construction of the classifier, which suffers from lower biological significance.

Freitas et al. [10] used J48 and SVM classifier to classify DNA repair genes into aging and not-aging class. Different sets of gene ontology terms and protein–protein interaction (PPI) features of the DNA repair genes are extracted and used as attributes in the classification approach. Biological interpretations of the selected features are discussed. However, the classification algorithms used do not consider any biological importance of the features used for classification and thus fail to provide greater insights.

Our work differs from the above works, as we have used the combination of sequence information and biological significant information viz., biochemical properties of DNA repair genes in form of GO terms and pathway information in the form of PPIs. Frequent gene subsequences are identified as motifs along with GO terms and PPIs that are associated with DNA repair genes.

19.3 System Model for DNA Repair Gene Categorization

19.3.1 Objectives

Our objective is to use the binary associative classifier to investigate the relationship between DNA repair genes and aging. More specifically, we have two goals; the first goal is to design and implement a classification algorithm and apply it to the DNA repair gene dataset to find gene properties that effectively discriminate between aging DNA repair genes and not-aging DNA repair genes. The second goal is to interpret the meaning of the discovered patterns in the light of biological knowledge about DNA repair genes and aging.

19.3.2 Problem Definition

The problem is modeled on the document classification problem in text mining domain. The objective is to classify accurately and efficiently a set of given DNA repair genes and their properties into a set of predefined classes. There are two predefined classes, aging related and not-aging related, and thus is a binary classification problem. For a given DNA repair gene, its nucleotide sequences and other biological properties are combined to form a document-like structure called *gene-document*. The gene sequence is divided into fixed length *k*-gene-words, where *k* is the length of the gene subsequence and annotated with respective GO terms and PPIs to form words of gene-documents that are analogs to words in the document.

Each gene-document is then combined to form transactional dataset, where gene-words represent the transactions. CARs are then mined from the dataset to extract features and then analyzed for their biological interpretations. We assume that the gene sequences and their properties are free from biological noise.

Let $G = \{g_1, g_2, g_3, \ldots, g_n\}$ be the set of n DNA repair genes. These genes are modeled as gene-documents, represented through set D, where $D = \{d_1, d_2, d_3, \ldots, d_n\}$ and d_i represents an individual gene-document. A gene-document is made up of two kinds of gene-words, GW:

1. Fixed length gene subsequences, $S = \{s_{1_k}, s_{2_k}, \ldots s_{m_k}\}$, where k is the subsequence length and s_{i_k} is the ith addition feature subsequence obtained from sliding-window, and
2. Term annotations, made up of GO terms GO, and PPIs P, thus $TA = \{(ta_l) \mid ta_l \in GO \; or \; ta_l \in P\}$

The predefined classes in the classification problem are represented through the set of class labels, $CL = \{A, \; NA\}$, where A is set of DNA repair genes that are related to aging and NA is set of DNA repair genes that are not related to aging. Set of training gene-documents are represented as:

$$T = \{(d_i, cl_j) \mid d_i \in D, cl_j \in CL\}$$

Using T, CARs are constructed and these rules are in implication form as:

$$gw_1, gw_2 \ldots gw_m \Rightarrow cl_j(R_s, R_c)$$

where left-hand side of the CARs, called as rule antecedent, consist of gene-words and right-hand side of the CARs, called as rule consequent, consists of class label, A or NA, i.e., $gw_1, gw_2 \ldots gw_m \in GW$ and $cl_j \in CL$. Two measures are defined on the above CARs as (i) rule support (R_s), the number of training gene-documents in which both gene-word and class label occur ($gw_m \cup cl_j$), and (ii) rule confidence (R_c), the conditional probability of class label given the gene-word ($cl_j \mid gw_m$). The number of gene-words on left-hand side of the CAR is defined as rule order of the CAR. Notations used in this chapter are listed in Table 19.1. An example of CAR is given below:

$$AATAGAC, XPC \Rightarrow A(19, 1.0000)$$

where *XPC* is a term annotation that belongs to PPI class and "AATAGAC" is the gene subsequence of a DNA repair gene in the training document set. R_s and R_c of the rule are 19 and 1.0000, respectively. This rule can be interpreted as in the training set gene-documents with XPC and "AATAGAC" as gene-words and A as class label appear totally 19 times and 100% of the gene-documents with *XPC* and "AATAGAC" as gene-words belong to the A class.

Table 19.1 List of notations used

Symbol	Description
G	Set of DNA repair genes
n	Total number of genes in the dataset
D	Set of gene-document
d_i	An individual gene-document
GW	Set of gene-words
S	Set of gene subsequences of a particular gene
k	Gene subsequence length
s_{i_k}	ith gene subsequence of sth gene
GO	Set of GO term annotations
P	Set of PPI term annotations
TA	Set of term annotation
CL	Set of class labels
A	Aging class
NA	Not-aging class
T	Set of training gene-documents
R_s	Rule support
R_c	Rule confidence

19.3.3 System Model

Overall execution flow of the proposed model is depicted in the Fig. 19.1. DNA repair gene sequences are collected from the DNA sequence database and are subjected to subsequence division through Algorithm 3 and sliding-window Algorithm 4, described in Phase 1 of Sect. 19.4, to increase the count of the feature in the dataset. GO terms and PPIs of respective DNA repair genes are extracted and annotated in the last stage of gene-document creation (Algorithm 2). CARs are constructed by mining frequent gene-words that are further pruned through boosting technique to construct the final set of rules in the classifier. The performance of the classifier is evaluated by subjecting the unseen test gene-document to it and noting the output.

19.4 Methodology for Gene-Document Creation and Classification

19.4.1 Overall Flow of the System

Overall flow of the proposed method is described in Term Annotated Sliding-Window-Based Boosting Associative Classifier (TASB-AC) algorithm, as shown in Algorithm 1. Classification task is divided into training and testing phase (Fig. 19.1).

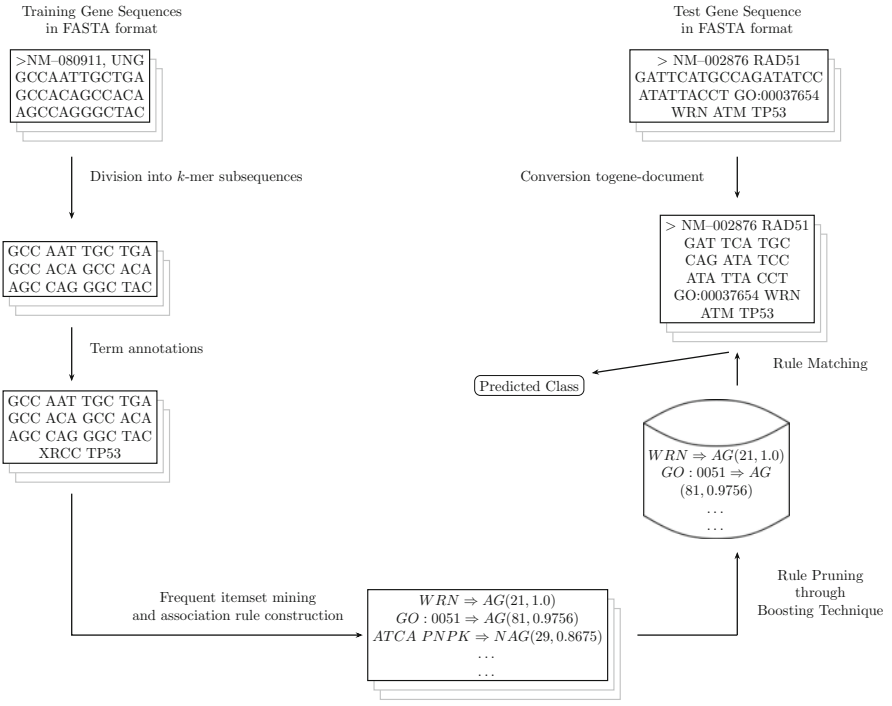


Fig. 19.1 Execution flow of the overall process

To build a binary associative classifier that classifies DNA repair genes into aging and not-aging classes, we employ the following four-phase procedure:

Phase 1: Transactional dataset construction (i.e., gene-documents)
Phase 2: Rule mining
Phase 3: Rule puring, and
Phase 4: Testing

Algorithm 1: TASB-AC

Input: A pool of *n* DNA Repair Genes, *G*

(i) Gene-document construction.
(ii) Frequent terms mining of gene-words through FP-growth algorithm and construction of CARs.
(iii) Rule pruning through boosting technique to construct the associative classifier.
(iv) Evaluating the performance of the classifier using *k*-fold cross validation technique.

Gene-documents are constructed first by collecting gene sequences and various properties of genes from database such as GenBank, HPRD, and GO database. Gene sequences are divided into subsequences, and their respective GO terms and

PPI information are annotated; these features form the gene-words. Frequently occurring gene-words are then mined through FP-growth algorithm, which are used to construct CARs. These rules are further pruned through boosting technique to get the final set of rules that make up the classifier. In testing phase, test genes are also converted to gene-documents, and then CARs are applied to get the class label. Prediction accuracy of the classification model is measured through k-fold cross validation technique. Most classification algorithms require training data to be present in fixed record length, i.e., all data instances should have fixed number of attributes. Proposed approach does not have such constraint as gene-documents are modeled as text and can have any number of gene-words.

19.4.1.1 Phase 1: Transactional Dataset Construction

In this phase, transactional dataset is constructed as detailed in Algorithm 2. Gene-documents are mapped to transactions, and words within the gene-documents (i.e., gene-words) are thought of as items of the transactions. BioPerl[1] is used to extract nucleotide sequences of genes in FASTA format from GenBank DNA database [19]. Each gene sequence is retrieved as a separate file and preprocessed to remove the descriptor line.

Algorithm 2: Gene-document construction

Input: A pool of n DNA repair genes, G
Output: Term-document matrix (TDM) of gene-documents
foreach $d_i \in D$ **do**
  Step 1: Retrieve DNA repair gene sequences from DNA database.
  Step 2: Divide the gene sequence into fixed length subsequences of length k using Algorithm 3.
  Step 3: Apply sliding-window to gene subsequences, using Algorithm 4.
  Step 4: Retrieve term annotations (i.e., GO & PPI terms) from biological databases.
  Step 5: Annotate terms to gene subsequences to construct gene-document.
  Step 6: Perform indexing of gene-words in gene-documents to obtain TDM of the gene-document set.
end

To map the gene sequence to words of a document in transactional dataset, we divide the gene sequence into fixed length subsequences. Algorithm 3 is used for subsequence division. The function getSubSequence (g_i,k,j) in Algorithm 3 returns subsequence of the gene g_i with length k starting from the jth position, and s_{i_j} is the returned gene subsequence. For a given gene sequence, if $k = 4$ then subsequence division would have created subsequences as:

[1] http://bioperl.org/.

GTAT TTAG TAGA CGGG TTCA CTAT GTAG GGTC ATGG AGCT

Algorithm 3: Subsequence division

Input: G: Set of DNA repair genes
L: Gene sequence length, and
k: Subsequence length
Output: S: Set of gene subsequences, each of length k
foreach $g_i \in G$ **do**
  $j = 0$;
  while $j < L$ **do**
    $s_{i_j} \leftarrow$ getSubSequence(g_i,k,j) ;
    $j \leftarrow j + k$;
  end
  $i \leftarrow i + 1$;
end

The number of gene subsequences are increased, so as to extract more features, with the help of Algorithm 4. Here, we employ sliding-window concept with fixed window size of k to extract more subsequence feature set. w is the loop control variable within each gene-document, and g_{w_j} represents the gene subsequence of wth sliding-window iteration with j as is its starting position. For a given gene sequence with k as 4, Algorithm 4 creates three additional features as:

GTAT|TTAG|TAGA|CGGG|TTCA|CTAT|GTAG|GGTC|ATGG|AGCT
GTA|TTTA|GTAG|ACGG|GTTC|ACTA|TGTA|GGGT|CATG|GAGC|T
GT|ATTT|AGTA|GACG|GGTT|CACT|ATGT|AGGG|TCAT|GGAG|CT
G|TATT|TAGT|AGAC|GGGT|TCAC|TATG|TAGG| GTCA|TGGA|GCT

Term Annotations

Apart from gene sequences, we have used biochemical properties of DNA repair genes as additional features in construction of the dataset, specifically GO terms and PPIs information are used. These are treated as term annotations and appended to the respective gene subsequences after the application of Algorithm 4.

Algorithm 4: Sliding-Window

Input: G: Set of DNA repair genes
L: Gene sequence length, and
k: Subsequence length
Output: S: Additional set of gene subsequences
foreach $g_i \in G$ **do**
  for $w = 1$ *to* k **do**
    $j \leftarrow 1$;
    if $w \neq 1$ **then**
      $s_{w_0} \leftarrow$ getSubSequence(g_i,w,0) ;
    end
    while $j < L$ **do**
      $s_{w_j} \leftarrow$ getSubSequence(g_i,k,j) ;
      $j \leftarrow j + k$;
    end
  end
end

Gene Ontology (GO) Terms

Gene Ontology[2] is a database that contains descriptions of gene and its product in form of well-defined vocabulary. Description of the genes is segregated into three classes as cellular component (CC), molecular function (MF), and biological process (BP). We have retrieved the GO terms of DNA repair genes using Biomart library in R[3] using RefSeq IDs of the genes and annotated them to the respective gene-documents.

Protein–Protein Interaction (PPI)

For PPI information we used Human Protein Reference Database (HPRD).[4] It is a well-curated and highly reliable database for human proteins. Binary PPI information of HPRD (release 9) dataset is used for term annotations. For all the DNA repair genes, their binary interaction partners are found out and annotated to respective gene-documents. Term annotations are then coded to alphabet words, so as to make gene-document free from numbers. For a given gene, after subsequence division and term annotations, the gene-document looks as follows:

[2] http://www.geneontology.org/.

[3] https://bioconductor.org/packages/release/bioc/html/biomaRt.html.

[4] http://www.hprd.org/.

GTAT TTAG TAGA CGGG TTCA CTAT GTAG GGTC ATGG AGCT GO:0004119 GO:0012425 GO:0035478 GO:0057896 GO:00312536 ATM TP53 XPC PCNA NHEJ1

Gene-words are indexed in the final step of dataset creation to obtain term-document matrix (TDM). Bow toolkit [20] is used to index the gene-words.

19.4.1.2 Phase 2: Rule Mining

Frequent gene-words are mined from the gene-documents to construct CARs; FP-growth algorithm [21] is used for the purpose. In initial stage, it constructs FP-tree structure by logging in the occurrence count of gene-words. Rule support (R_s) is used as threshold to discard all those gene-words that fall below it; here R_s is set to considerably low value so as to obtain huge set of gene-words. Once the set of frequently occurring gene-words are obtained, CARs are constructed in the final stage. These steps are detailed in the Algorithm 5.

Algorithm 5: FP-growth

Input: Term-document matrix(TDM).
Output: Class association rules (CARs).
Step 1 : Construction of FP-tree.
(i) Scan TDM of gene-document set to find support count of each gene-word.
(ii) Discard those gene-words that fall below threshold support.
(iii) Sort frequent gene-words in descending order based on their support count.
(iv) Construct FP-tree by scanning each gene-document at a time and include a path corresponding to it.
Step 2 : Frequent gene-word set construction.
(i) Mine FP-tree to construct conditional Pattern base for each suffix pattern.
(ii) Construct conditional pattern FP-tree.
(iii) Recursively mine on such FP-tree.
(iv) Concatenate suffix-pattern with the frequent patterns generated from a conditional FP-tree.
Step 3 : Construct rules out of these generated frequent gene-words.

19.4.1.3 Phase 3: Rule Pruning

In this phase only those CARs that have high coverage threshold are retained, and a final classifier is built out of them. We use boosting algorithm proposed by Yoon

et al. [18] to eliminate the rules that have low coverage threshold. Steps used in this phase are shown in the Algorithm 6.

Algorithm 6: Boosting Algorithm to Prune CARs

Input: Class association rules (CARs), with their support and confidence.
Output: Final Associative Classifier, AC.
Step 1 : Sort CARs according to their confidence and support.
Step 2 : Initialize weights of training data instances to 1.
Step 3 : Apply CAR to training data
(i) If it classifies correctly then add it to final rule set AC.
(ii) Update the weight of training data instance.
(iii) If the weight is less than the coverage threshold then delete the training data instance from the training dataset.
Step 4 : Repeat step 3 for all rules.

19.4.1.4 Phase 4: Testing

In this final phase, the gene-documents are divided into training and testing set. Training set documents are used in construction of the CARs are detailed in the three phases above. Testing set documents are used to validate the accuracy of the associative classifier built.

19.5 Experimental Setup

19.5.1 Dataset

Human DNA repair genes are considered to test the proposed classification method. A list of DNA repair genes is taken from Wood's list [22] that contains a total of 178 genes. We have collected their RefSeq IDs. These genes are then classified into two classes, genes which are related to aging process and genes which are not related to aging process as aging class and not-aging class. Build 18 of GenAge [23] database is used to establish the classification of genes. All those DNA repair genes that are listed in GenAge database are taken in aging class, and those DNA repair genes which are not present are taken into the not-aging class. Thirty-seven DNA repair genes are thus listed in the aging class while remaining 141 genes in the not-aging class. Nucleotide sequences of these genes are then collected from the GenBank databank to construct the respective gene-documents.

Term annotations are obtained from the respective databases and annotated to the gene-documents; 925 GO terms and 913 binary PPIs are obtained in total for the 178 DNA repair genes. In Sect. 19.5.3 it is established that for the subsequence length of 7, accuracy of classifier is greater than all other subsequence lengths. At this length

the dataset contains 1246 gene-documents after feature extension and annotation. Among them 259 documents are of aging class, and remaining 987 are of not-aging class; also the number of unique gene-words is 18,995.

19.5.2 Performance Evaluation

Term Annotated Sliding-Window-Based Boosting Associative Classifier (TASB-AC) is a generic associative classification method. The codes for the prediction system are written with Perl, R, and Shell scripts. The programs are run on a Linux machine (Ubuntu, 12.04) with 3 GB memory and 2.4 GHz CPU speed. To measure the performance of the binary associative classifier, we use tenfold cross validation method. Gene-documents are divided into ten non-overlapping subsets, which contains equal number of documents and then given to classifier for training and testing. Out of ten sets, nine are used for training the classifier (i.e., construction of CARs), and the remaining one set is used for validating the classifier (i.e., testing). We repeat the procedure until all the sets are used in the testing phase. Performance of each run is noted, and overall performance is taken as average of all the runs. We considered three performance measures to evaluate the binary classifier; they are Precision (P), F-score (F_s), and Recall (R). A confusion matrix is constructed in each run of the classifier to determine these measures that consists the following actual and predicted counts:

1. **True Aging Count** (T_A): Number of actual aging DNA repair genes that are predicted correctly by the classifier in the aging class.
2. **False Aging Count** (F_A): Number of actual aging DNA repair genes that are predicted incorrectly by the classifier in the aging class.
3. **True Not-aging Count** (T_{NA}): Number of actual not-aging DNA repair genes that are predicted correctly by the classifier in the not-aging class.
4. **False Not-aging Count** (F_{NA}): Number of actual not-aging DNA repair genes that are predicted incorrectly by the classifier in the not-aging class.

Precision gives the percentage of Aging DNA repair gene-documents that are classified correctly into aging class, while Recall gives the percentage of aging DNA repair gene-documents that are classified incorrectly into not-aging class. Equations (19.1) and (19.2) are used for calculation of Precision and Recall, respectively.

$$P = \frac{T_A}{T_A + F_{NA}} \tag{19.1}$$

$$R = \frac{T_A}{T_A + F_A} \tag{19.2}$$

F-score gives the weighted average of Precision and Recall. To calculate F-score, we use Eq. (19.3).

$$F_S = 2 \cdot \frac{P \times R}{P + R} \tag{19.3}$$

Accuracy of the binary classifier is given by the Eq. (19.4). It gives the portion of gene-documents in the testing dataset that are classified correctly.

$$Accuracy = \frac{T_A + T_{NA}}{T_A + F_A + F_{NA} + T_{NA}} \tag{19.4}$$

19.5.3 Results and Discussions

Two sets of experiments were performed with various subsequence lengths of DNA repair genes, one by considering only gene sequences as gene-words of documents (i.e., without annotations) and another with both gene sequences and term annotations as gene-words of the documents. R_s and R_c are set to 20% and 50%, respectively, so as to mine large number of gene-words.

Gene subsequence length is taken along X-axis while accuracy of the classifier is taken along Y-axis as shown in Fig. 19.2. With respect to gene-documents that contain term annotations, accuracy of the classifier increases linearly with the subsequence length and then attains a constant value, while with respect to gene-

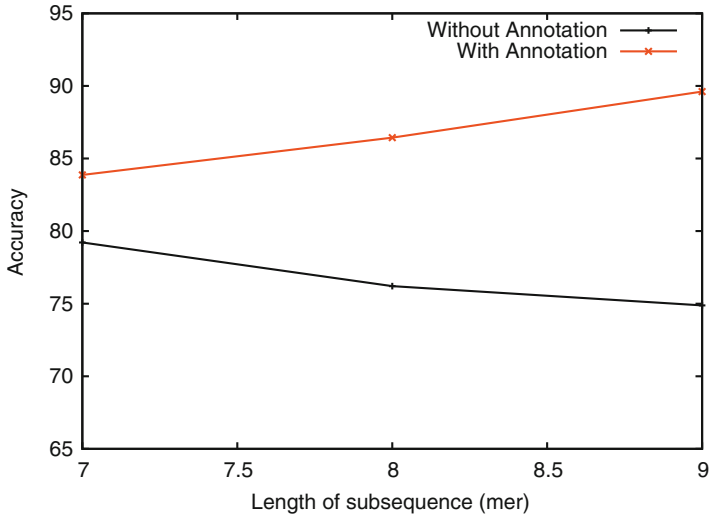


Fig. 19.2 Accuracy vs. gene subsequence length

documents that do not contain term annotations, accuracy of the classifier drops considerably with increase in the subsequence length. It is thus established that term annotations provide significant contribution to the classification accuracy.

To determine the optimal length of gene subsequence, same set of experiments are performed, and the number of CARs generated in the training set is noted. Figure 19.3 shows the number of CARs generated with respect to the gene subsequence length. It is seen from the graph that number of CARs generated drops substantially in both the experiments with the increase in the subsequence length. We analyze CARs generated for each subsequence length and noted that for higher subsequence lengths the left-hand side of the CARs contained considerably less gene terms. Considerable number of subsequence terms were noted for subsequence length of 7 thus we choose it to be the optimal value. Some of the CARs generated at this length are shown below:

Some of the CARs Generated by the Associative Classifier
GO:0001578 GO:0005678 → NA (94,0.7589)
ATTTAAC CCATGAT → NA (47, 0.9875)
GO:0001578 CCATGAT → NA (78, 0.7894)
GO:0006512 TTACTAA → NA (92, 0.9456)
GO:0005354 GO:0007524 → NA (82, 0.8715)
GO:0007814 GO:0007829 GO:0008927 → A (89, 1.0000)
GO:0007125 GO:0002541 TP53 → A(69, 1.0000)
WRN → A (82, 1.0000)
GO:0001874 WRN → A(64, 1.0000)
GO:0003314 GO:0005789 BRCA1 → A (81, 0.9076)

CARs are analyzed to know how DNA repair genes are related to aging. Rules that belong to aging class have significant term annotations in the left-hand side and thus contribute more than gene subsequences in predicting the class of DNA repair genes. For not-aging class along with GO terms, gene subsequences also appear in the left-hand side of the CARs and thus have greater prediction power. Ten most frequently appearing GO terms in the left-hand side of the CARs are listed in Table 19.2. For analysis we consider only those GO terms that belong to biological process (BP) class, it can be seen from the table that the term "GO:0009411 (response to UV)" appears most number of times in the left-hand side of the aging class, i.e., 322 times, while for CARs belonging to not-aging class this term appears only 24 times. All CARs with this term have a confidence score of 100%, indicating this term has highest prediction power in the classifier. Ten DNA repair genes out of 37 that belong to aging class are annotated with this GO term; it can thus be considered that most of the aging DNA repair genes try to repair damages to DNA caused by external factors, i.e., exposure to UV radiation (exogenous damage).

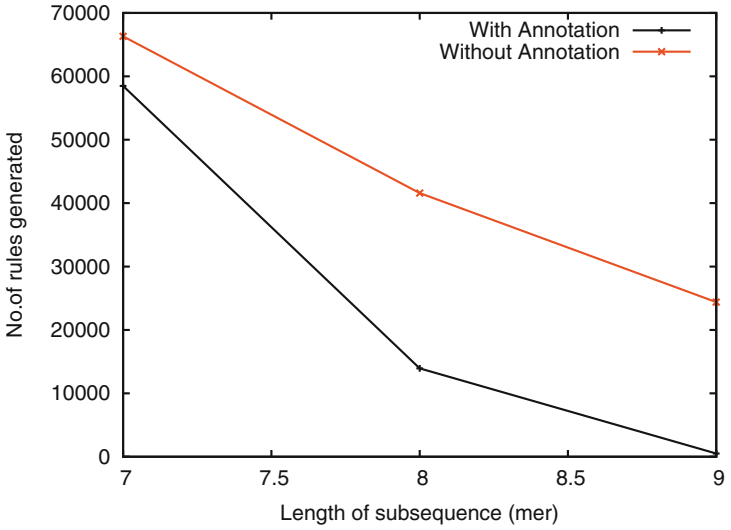


Fig. 19.3 Number of rules generated v/s gene subsequence length

Table 19.2 GO terms count for aging and not-aging classes

Not-aging class		Aging class	
GO terms	Occurrence count	GO terms	Occurrence count
GO:0005634	9623	GO:0005515	1427
GO:0006281	9428	GO:0003677	954
GO:0005515	7323	GO:0005654	795
GO:0003677	2179	GO:0005634	609
GO:0046872	1618	GO:0006281	515
GO:0006260	1529	GO:0006302	369
GO:0005654	832	GO:0005524	350
GO:0005524	678	GO:0009411	322
GO:0005737	631	GO:0006974	299
GO:0003684	528	GO:0003684	255

In not-aging class, most frequently appearing GO term in BP class is "GO:0006260 (DNA Replication)"; it appears 1529 times in the left-hand side of the CARs that belong to not-aging class with confidence score varying between 85% and 100%. While in aging class, this term appears only 24 times; it can thus be concluded that DNA repair genes that rectify errors during DNA replication process belong to not-aging class (i.e., endogenous damage). We thus conclude that DNA repair genes that repair endogenous damages belong to not-aging class, while those that repair exogenous damages belong to not-aging class.

Ten most frequently appearing binary PPI partners for aging and not-aging class are listed in Table 19.3. From the table it can be noted that PPI terms have greater

Table 19.3 Occurrence count of binary PPI partners for aging and not-aging class

Not-aging class		Aging class	
Genes	Number of interacting partners	Genes	Number of interacting partners
RAD9B	173	TP53	619
REV1	170	WRN	459
FANCA	120	BRCA1	380
XPC	117	APEX1	225
FANCC	107	PCNA	218
MLH1	105	PARP1	214
HES1	100	XRCC5	203
NHEJ1	86	PRKDC	183
PCNA	59	RAD51	141
ESR1	58	ATM	137

Table 19.4 Most frequently appearing *k*-word gene subsequences

Aging	Not-aging
GAAGAAG	TTTATTT
AAGAAGA	TTTTAAA
AGAAGAA	ATATTTT
ATGGAAA	CCTGGAA
TGGCCAT	TTGAAAC
AAAGAAG	CTCCTCC
TTTTCCT	GGAGGAG
GGAGATG	AAAAGAA
CCTGGAG	GCTGCTG
AAAAGAA	GAGTTCA

prediction power in aging class than in not-aging class, as their frequency count is larger in the former class. It is found that if a DNA repair gene is annotated with WRN, TP53, and XRCC5 as protein interacting partner then that gene most likely belongs to aging class, as these proteins are found to be in aging pathways. Table 19.4 shows the most frequently appearing gene subsequences that can be thought of as sequence motifs in the respective classes.

19.6 Conclusions

In this work, we have proposed, designed, implemented, and analyzed a binary associative classifier called Term Annotated Sliding-Window-Based Boosting Associative Classifier (TASB-AC) that classifies DNA repair genes into aging and not-aging classes. Along with biochemical properties of the DNA repair genes, nucleotide sequences are also considered as features to train the classifier. To increase the number of sequence feature set, a sliding-window approach is employed

that boosts the performance of the classifier and also identifies the most frequently appearing sequence motifs. CARs that are generated during the training phase of the classifier are analyzed for biological interpretations. The DNA repair genes that repair damages caused by external factors belong to aging class, while those that repair damages caused by internal factors belong to not-aging class. In future, we plan to include other gene properties like structure of the gene and its orientation as features for training the classifier and to apply a weighing scheme to the CARs so as to prioritize them during pruning stage.

References

1. Moffitt, T.E., Belsky, D.W., Danese, A., Poulton, R., Caspi, A.: The longitudinal study of aging in human young adults: knowledge gaps and research agenda. J. Gerontol. A **72**(2), 210–215 (2017)
2. Lombard, D.B., Chua, K.F., Mostoslavsky, R., Franco, S., Gostissa, M., Alt, F.W.: DNA repair, genome stability, and aging. Cell **120**(4), 497–512 (2005)
3. Kirschner, K., Chandra, T., Kiselev, V., Flores-Santa Cruz, D., Macaulay, I.C., Park, H.J., Li, J., Kent, D.G., Kumar, R., Pask, D.C., et al.: Proliferation drives aging-related functional decline in a subpopulation of the hematopoietic stem cell compartment. Cell Rep. **19**(8), 1503–1511 (2017)
4. Cadet, J., Davies, K.J.: Oxidative DNA damage & repair: an introduction. Free Radic. Biol. Med. **107**, 2–12 (2017)
5. Li, Y.-H., Zhang, G.-G., Guo, Z.: Computational prediction of aging genes in human. In: Proceedings of 2010 International Conference on Biomedical Engineering and Computer Science (ICBECS), pp. 1–4 (2010)
6. Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier, Amsterdam (2011)
7. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. ACM SIGMOD Rec. **22**(2), 207–216 (1993)
8. Song, K., Lee, K.: Predictability-based collective class association rule mining. Expert Syst. Appl. **79**, 1–7 (2017)
9. Jiang, H., Ching, W.-K.: Classifying DNA repair genes by Kernel-based support vector machines. Bioinformation **7**(5), 257–263 (2011)
10. Freitas, A.A., Vasieva, O., de Magalhães, J.P.: A data mining approach for classifying DNA repair genes into ageing-related or non-ageing-related. BMC Genomics **12**(1), 27 (2011)
11. Fang, Y., Wang, X., Michaelis, E.K., Fang, J.: Classifying aging genes into DNA repair or non-DNA repair-related categories. In: Proceedings of the International Conference on Intelligent Computing, pp. 20–29 (2013)
12. Wan, C., Freitas, A.A.: Two methods for constructing a gene ontology-based feature network for a Bayesian network classifier and applications to datasets of aging-related genes. In: Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics, pp. 27–36 ACM (2015)
13. Pevsner, J.: Bioinformatics and Functional Genomics. Wiley, New York (2015)
14. Vidya, A., Pattar, S., Tejaswi, V., Venugopal, K.R., Patnaik, L.M.: DNA repair gene catergorization through associative classification. In: 7th International Conference on Advanced Computer Theory and Engineering (ICACTE-2014), vol. 7, pp. 1–5 (2014)
15. Salim, A., Chandra, S.V.: Association rule based frequent pattern mining in biological sequences. In: Proceedings of the 2013 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1–5 (2013)

16. Becerra, D., Vanegas, D., Cantor, G., Niño, L.: An association rule based approach for biological sequence feature classification. In: IEEE Congress on Evolutionary Computation, 2009. CEC'09., pp. 3111–3118 (2009)
17. Yu, P., Wild, D.J.: Discovering associations in biomedical datasets by link-based associative classifier (LAC). PloS One **7**(12), e51018 (2012)
18. Yoon, Y., Lee, G.G.: Subcellular localization prediction through boosting association rules. IEEE/ACM Trans. Comput. Biol. Bioinform. **9**(2), 609–618 (2012)
19. Benson, D.A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Sayers, E.W.: GenBank. Nucleic Acids Res. **41**(D1), D36–D42 (2012)
20. McCallum, A.K.: Bow: a toolkit for statistical language modeling, text retrieval, classification and clustering (1996). [Online]. Available: http://www.cs.cmu.edu/~mccallum/bow
21. Borgelt, C.: An implementation of the FP-growth algorithm. In: Proceedings of the 1st International Workshop on Open Source Data Mining: Frequent Pattern Mining Implementations, pp. 1–5 (2005)
22. Wood, R.D., Mitchell, M., Lindahl, T.: Human DNA repair genes, 2005. Mutat. Res. Fundam. Mol. Mech. Mutagen. **577**(1), 275–283 (2005)
23. Tacutu, R., Craig, T., Budovsky, A., Wuttke, D., Lehmann, G., Taranukha, D., Costa, J., Fraifeld, V.E., De Magalhães, J.P.: Human ageing genomic resources: integrated databases and tools for the biology and genetics of ageing. Nucleic Acids Res. **41**(D1), D1027–D1033 (2012)

Chapter 20
Using Markov Models and Statistics to Learn, Extract, Fuse, and Detect Patterns in Raw Data

R. R. Brooks, Lu Yu, Yu Fu, Guthrie Cordone, Jon Oakley, and Xingsi Zhong

Abstract Many systems are partially stochastic in nature. We have derived data-driven approaches for extracting stochastic state machines (Markov models) directly from observed data. This chapter provides an overview of our approach with numerous practical applications. We have used this approach for inferring shipping patterns, exploiting computer system side-channel information, and detecting botnet activities. For contrast, we include a related data-driven statistical inferencing approach that detects and localizes radiation sources.

Keywords Hidden Markov model (HMM) · Tracking system · Smart grid security · Phasor measurement unit (PMU) · Bitcoin transaction analysis · Radiation detection and localization · Linear regression · Maximum likelihood estimation (MLE)

20.1 Introduction

Markov models have been widely used for detecting patterns [4, 7, 15, 17, 23, 24, 32, 36]. The premise behind Markov models is that the current state only depends on the previous state and that transition probabilities are stationary. This makes Markov models versatile, as this is a direct result of the causal world we live in. Often, these models can only be partially observed. In that case, we refer to the collective (observable and non-observable) model as a hidden Markov model (HMM).

Stochastic processes can successfully model many system signals. Some of these systems cannot be accurately represented using a Markov model or an HMM due to the uncertainty of input data. One task that benefits from stochastic signal processing is the detection and localization of radioactive sources. Since radioactive decay follows a Poisson distribution [21], radiation measurements must be treated

R. R. Brooks (✉) · L. Yu · Y. Fu · G. Cordone · J. Oakley · X. Zhong
Holcombe Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA
e-mail: rrb@clemson.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_20

as stochastic variables. A prevalent method for localizing radioactive sources is maximum likelihood estimation (MLE) [11, 13, 19]; we consider bootstrapping the MLE using estimates from a linear regression model [10].

In Sect. 20.2, background information about Markov models is presented, along with references to our previous work developing the methods presented here. Section 20.3 discusses new developments in the inference of HMMs, detection using HMMs, methods for determining the significance of HMMs, HMM metric spaces, and applications that use HMM-based detection. Section 20.4 analyzes radiation processes, maximum likelihood localization, and the use of linear regression to estimate source and background intensities. In Sect. 20.5, closing remarks are presented.

20.2 Background Information and Previous Work

A Markov model is a tuple $G = (S, T, P)$, where S is a set of states, T is a set of directed transitions between the states, and $P = \{p(s_i, s_j)\}$ is a probability matrix associated with transitions from state s_i to s_j such that

$$\sum_{s_j \in S} p(s_i, s_j) = 1, \quad \forall s_i \in S \tag{20.1}$$

In a Markov model, the next state depends only on the current state. An HMM is a Markov model with unobservable states. A standard HMM [14, 27] has two sets of random processes: one for state transition and the other for symbol outputs. Our model is a deterministic HMM [23, 25, 28], which only has one random process for state transitions. Therefore, given the current state and the symbol associated with the outgoing transition, the next state is deterministic. For each deterministic HMM, there is an equivalent standard HMM and vice versa [37]. This deterministic property helps us infer HMMs from observations.

Schwier et al. [30] developed a method for determining the optimal window size to use when inferring a Markov model from serialized data. Brooks et al. [7] found that HMM confidence intervals performed better than the maximum likelihood estimate. This was illustrated through an analysis of consumer behavior. Building on this work, Lu et al. [40] devised a method for determining the statistical significance of a model, as well as determining the number of samples required to obtain a model with a desired level of statistical significance.

20.3 Markov Modeling

20.3.1 Inferring Hidden Markov Models

HMM inference discovers the HMM structure and state transition probabilities from a sequence of output observations. Traditionally, the Baum-Welch algorithm [27] is used to infer the state transition matrix of a Markov chain and symbol output

probabilities associated with the states of the chain, given an initial model and a sequence of symbols. As a result, this algorithm requires the a priori structural knowledge of the Markov process that produced the outputs.

Schwier et al. [29] developed the zero-knowledge HMM inference algorithm. The assumption of HMM inference is (1) the transition probabilities are constant, (2) the distribution of the underlying HMM is the stationary, and (3) the Markov process is ergodic. In essence, this means that the system is Markovian and nonperiodic and has a single strongly connected component. The input of the algorithm is a string of symbols, and the output is the HMM. If the input data are not provided as a set of strings but rather as a set of continuous trajectories, then [18] explains how to change trajectories into symbolic strings.

The original algorithm only requires one parameter, L, as the input, which refers to the maximum number of history symbols used to infer the HMM state space and estimate associated probabilities. Take "abc" as an example input string. For $L = 1$, it calculates transition probabilities $P(a|a)$, $P(a|b)$, $P(a|c)$, and so on, by creating a state for each symbol and counting how many times the substrings "aa," "ab," and "ac" occur in the training data and dividing them by the number of occurrences of the symbol "a." The chi-squared test is used to compare the outgoing probability distributions for all pairs of states. If two states are equivalent, they are merged.

For $L = 2$, the number of states used for representing conditional probabilities is squared. For example, in a system that has $L = 2$, it could be the case that $P(c|a) \neq P(c|aa)$. In which case "a" and "aa" will be different states. The state space increases exponentially with L, as does the time needed to infer the HMM. We showed that L can be determined automatically as part of HMM inference. The idea is to keep increasing L until the HMM model stabilizes, i.e., $\text{HMM}(L) == \text{HMM}(L + 1)$. This data-driven approach finds the system HMM with no a priori information. This process is described in detail in [28, 29].

If an insufficient amount of observation data is used to generate the HMM, the model will not be representative of the actual underlying process. A model confidence test is used to determine if the observation data and constructed model fully express the underlying process with a given level of statistical significance [40]. This approach calculates a lower bound on the number of samples required. If the number of input samples is less than the bound, more data is required. New models should be inferred with more data and still need to be checked for confidence. This approach allows us to remove the effect of noise in the HMM inference.

20.3.2 Detection with HMM

Once the HMM is inferred from symbolized data and passes the model confidence test, it can be used to detect whether or not a data sequence was generated by the same process. The traditional Viterbi algorithm [27] finds the HMM that was most likely to generate the data sequence by comparing probabilities generated by the HMMs. For data streams, it is unclear what sample size to use with the Viterbi algorithm. Also as data volume increases, the probability produced by the Viterbi

algorithm decreases exponentially and may suffer floating-point underflow [7]. To remedy this, confidence intervals (CIs) are used [7]. With this approach, the certainty of detection increases with the number of samples, and the floating-point underflow issues of the Viterbi algorithm are eliminated [7]. The CI approach can determine, for example, whether or not observed network traffic was generated by a botnet HMM.

Given a sequence of symbolized traffic data and a HMM, the CI method in [7] traces the data through the HMM and estimates the transition probabilities and confidence intervals. This process maps the observation data into the HMM structure. It then determines the proportion of original transition probabilities that fall into their respective estimated CIs. If this percentage is greater than a threshold value, it accepts that the traffic data adequately matches the HMM.

Lu et al. [23] used an alternative approach. Instead of estimating transition probabilities and corresponding CIs, they calculated the state probabilities, which are the proportion of time the system stays in a specific state, and their corresponding CIs. Once all state probabilities and CIs were estimated for the observation sequence, they determined whether each state is within a certain confidence interval. For the whole HMM, the proportion of states was obtained whose estimated state probability matched the corresponding confidence intervals. If an observation sequence was generated from the HMM, it would follow the state transitions of the underlying stochastic process that the HMM represents, and its state probabilities would converge to the asymptotic state probabilities if the sequence length was large enough. Therefore, more states would match their estimated CIs. Generally, a sequence that is generated by the HMM would have a high proportion of matching probabilities, while a sequence that is not an occurrence of the HMM will have a low proportion of matching probabilities. Similar to the detection approach in [7], a threshold value can be set for this proportion of matching states, to determine whether the observation sequence matches with the HMM.

In [7], receiver operating characteristic (ROC) curves were used to find optimal threshold values. By varying the threshold from 0 to 100% (0% threshold means we accept everything and we will have a high false-positive rate. 100% threshold means we reject everything and we will have a low true-positive rate), they progressively increased the criteria for acceptance. Using the ROC curve drawn from detection statistics with different thresholds, the closest point to $(0, 1)$ was found, which represents 0% false-positive rate and 100% true-positive rate. This considered the trade-off between true-positive and false-positive rates. Therefore, the corresponding threshold of that point was the optimal threshold value.

20.3.3 Statistically Significant HMMs

In [40], we address the problem of how to ensure the inferred HMM accurately represents the underlying process that generates the observation data. The measure of the accuracy is known as *model confidence*, which means the degree to which

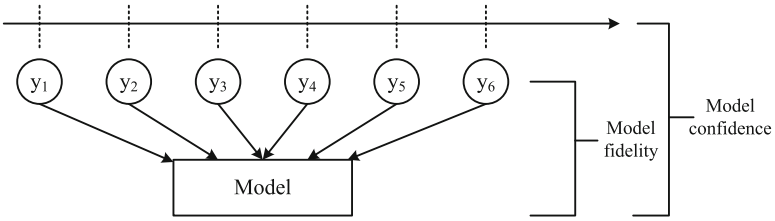


Fig. 20.1 Hierarchy of the process, observations, and model showing the relationship between model fidelity and model confidence (adopted from [40])

a model represents the underlying process that generated the training data. As illustrated in Fig. 20.1, model confidence is different from *model fidelity*, where the latter refers to the agreement between the inferred model and the training data.

Mathematically, we calculate an upper bound on the number of samples required to guarantee that the constructed model has a given level of significance. In other words, we have shown how to determine within a given level of statistical confidence (α) if a "known unknown" transition does not occur, given two user-defined thresholds ϵ and α. The parameter ϵ determines the minimum probability that transitions with probabilities no less than ϵ should be included in the constructed model. The parameter α is the confidence level that shows the accuracy of the model result.

For each state, s, we use the z-test [5] to determine if the inferred model includes all transitions with probabilities no smaller than $\gamma_s^\epsilon = \epsilon/\pi_s$ with desired significance α, where π_s is the asymptotic probability of s as the constructed Markov model is irreducible. Concretely, we test the null hypothesis H_0: $\overline{X_s^U} = \gamma_s^\epsilon$ against the alternative hypothesis H_1: $\overline{X_s^U} < \gamma_s^\epsilon$, where $\overline{X_s^U}$ denote the sample average of the probability of an unobserved transaction of state s. We use $\gamma_s^\epsilon - \overline{X_s^U}$ as a test statistic, rejecting the null hypothesis if $\gamma_s^\epsilon - \overline{X_s^U}$ is too large. We have not observed the transition, thereby $\overline{X_s^U} = 0$. We fail to reject H_0 if we need to collect more data. Otherwise, we say that sufficient data has been collected. If we fail to reject the null hypothesis for all states, the z-test also finds the minimum amount of training samples needed for transaction transitions with probabilities no smaller than γ_s^ϵ to be detected with desired level of significance α.

To use the z-test in this manner, we propose a simple algorithm to perform online testing of the observation sequence. The algorithm determines if a constructed model statistically represents a data stream in the process of being collected. We first collect a sequence of observation data $\mathbf{y}$ of some length D and construct a model from the collected data. With the constructed model, we determine the z-statistics and find if the experimental statistic provides $100 \cdot (1 - \alpha)\%$ confidence that a transition with probability ϵ does not occur. If $\mathbf{y}$ is not sufficiently long, we will be unable to construct a model from the data; additional data should be gathered. The algorithm is provided in [40].

20.3.4 HMM Metric Space

A metric is a mathematical construct that describes the similarity (or difference) between two models. It is useful to know if two processes are the same, except for rare events (e.g., events occurring once in a century), since we would typically consider them functionally equivalent. Eliminating duplicate models can reduce system complexity by decreasing the number of models used to analyze using observation data. Grouping similar models can increase the number of samples available for model inference, leading to higher fidelity system representations. To determine if two deterministic HMMs G_1 and G_2 are equivalent, we first let G_1 generate an observation sequence $\mathbf{y}$ of length D and then run $\mathbf{y}$ through G_2. Using frequency counting, we obtain a set of sample transition probabilities for each transition of G_2, denoted by T_2'. The probabilities of all outgoing transitions conditioning on each state follow multinomial distribution, which are then approximated by a set of binomial distributions in this context. This allows us to use the standard χ^2-test to compare if the two sets of transition distributions $\{T_2', T_2\}$ are equivalent with significance α [22], where T_2 is the set of transition probabilities of G_2. We define equivalence ($G_1 \equiv G_2$) as G_1 and G_2 accepting the same symbol sequences with a user-defined statistical significance α. The corresponding algorithm is provided in [25]. Note that, if there does not exist a path across G_2 that yields observation sequence $\mathbf{y}$, we can immediately conclude that G_1 and G_2 are not equivalent.

If G_1 and G_2 are not equivalent, we quantify the distance between them as how much their statistics differ with significance α. Given a probability $0 \leq P_{th} \leq 1$, we prune all transactions with probability no greater than P_{th} from both models and then renormalize the probabilities of remaining transitions for each state to obtain two sub-models. The pruning process is illustrated in Fig. 20.2 using a simple model with threshold value $P_{th} = 0.002$. The distance between the two models is defined as the minimum P_{th} that yields two equivalent sub-models. To determine this distance, we progressively remove the least likely events from both models

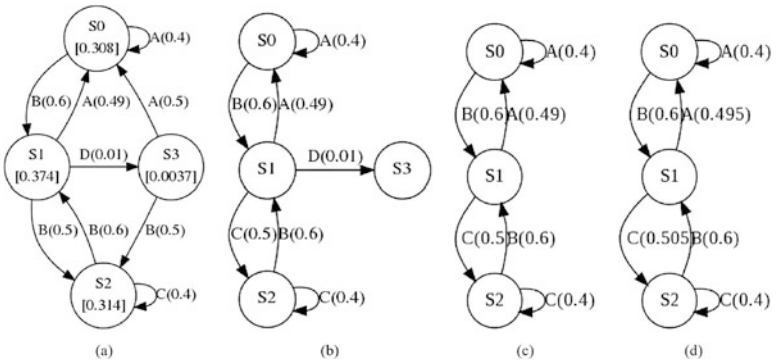


Fig. 20.2 Process to remove transitions: (**a**) original model (asymptotic state probabilities are in square brackets); (**b**) initial step with $P_{th} = 0.002$; (**c**) removal of absorbing states; and (**d**) resulting model with rescaled probabilities (Adopted from [25])

until the remaining sub-models are considered equivalent. Specifically, starting from $P_{th} = 0$, we prune all transactions with probability no greater than P_{th} and then renormalize the probabilities of remaining transactions for each state. We gradually increase P_{th} and repeat the pruning operation for each P_{th} until the remaining sub-models are considered equivalent with the predefined confidence level α. Let $d(G_1, G_2)$ denote the statistical distance between G_1 and G_2, and then $d(G_1, G_2)$ is equal to the stopping value of P_{th}. We show in [25] that $d(G_1, G_2)$ is a metric as it fulfills all the necessary conditions as a metric.

Since the χ^2-test is used, we need to ensure that enough samples (D) are used for transition probabilities to adequately approximate normal distributions. In order to approximate the binomial distribution as normal distribution, the central limit theorem is used to calculate the required sample size.

This is similar in spirit to Kullback-Leibler divergence (KLD); however, unlike KLD, our approach is a true metric. It does not have KLD's limitations and provides a statistical confidence value for the distance. We illustrated the performance of the metric by calculating the distance between different, similar, and equivalent models. Compared with KLD measurement, our approach is more practical and provides a true metric space.

20.3.5 HMM Detection Applications

We have used this approach to:

- Track shipping patterns in the North Atlantic;
- Identify protocol use within encrypted VPN traffic;
- Identify the language being typed in an SSH session;
- Identify botnet traffic;
- Identify smart grid traffic within encrypted tunnels and disrupt data transmission;
- Create transducers that transform the syntax of one network protocol into another one;
- De-anonymize Bitcoin currency transfers.

20.3.5.1 Track Shipping Patterns in the North Atlantic

Symbolic transfer functions (STFs) were developed for modeling stochastic input/output systems whose inputs and outputs are both purely symbolic. Griffin et al. [18] applied STFs to track shipping patterns in the North Atlantic by assuming the input symbols represent regions of space through which a track is passing, while the output represents specific linear functions that more precisely model the behavior of the track. A target's behavior is modeled at two levels of precision: the symbolic model provides a probability distribution on the next region of space and behavior (linear function) that a vehicle will execute, while the continuous model predicts the position of the vehicle using classical statistical methods. They

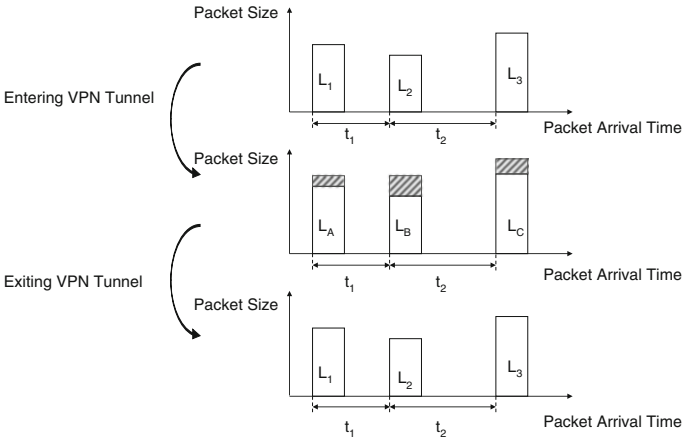


Fig. 20.3 Timing and size side-channels in a VPN tunnel

collected over 8 months' worth of data for 13 distinct ships, representative of a variety of vessel classes (including cruise ships, Great Lakes trading vessels, and private craft). The STF algorithm was used to produce a collection of vessel track models. The models were tested for their prediction power over the course of 3 days.

20.3.5.2 Identify Protocol Use Within Encrypted VPN Traffic

A VPN is an encrypted connection established between two private networks through the public network. Packets transferred through a VPN tunnel have the source and destination IP of the private networks, which are not always the final destination of the packet. The destination IPs for each packet are encrypted and cannot be seen by any third party. A typical VPN implementation encrypts the packets with little overhead and pads all packets in a given size range to the same size. Thus, the timing side-channel and packet size side-channel can be used to identify the underlying protocol even after encryption. These side-channels are shown in Fig. 20.3. In [42], a synchrophasor network protocol is identified in an encrypted VPN tunnel.

20.3.5.3 Identify the Language Being Typed in an SSH Session

SSH encryption is mathematically difficult to break, but the implementation is vulnerable to side-channel analysis. In an interactive SSH session, users' keystroke-timing data are associated with inter-packet delays. In [20], the inter-packet delays are used to determine the language used by the victim. This is achieved by comparing the observed data with the HMM of known languages. In [33], the timing side-channel is used to extract the system password from interactive SSH sessions.

20.3.5.4 Identify Botnet Traffic

Botnets are becoming a major source of spam, distributed denial-of-service attacks (DDoS), and other cybercrimes. Chen et al. [24] used traffic timing information to detect the centralized Zeus botnet. The reasons to use timing information are (1) inter-packet timings relate to the command and control processing time of the botnet; (2) the bots periodically communicate with the command and control (C&C) server for new commands or new data; (3) the inter-packet delays filter out constant communication latencies; and (4) it does not require reverse engineering the malware binaries or encrypted traffic data. An HMM is inferred from inter-packet delays of Zeus botnet C&C communication traffic. Using the behavior detection method with confidence intervals (CIs) of HMMs [7], they were able to detect whether or not a sequence of traffic data is botnet traffic. Experimental results showed this approach can properly distinguish wild botnet traffic from normal traffic.

20.3.5.5 Identify Smart Grid Traffic Within Encrypted Tunnels and Disrupt Data Transmission

In a power grid, phasor measurement units (PMUs) send measurement data (or synchrophasor data) over the Internet to a control center for closed-loop control. Any unexpected change in the variance of the packet delay can cause instability in the power grid and even blackouts. PMUs are deployed at critical locations and are usually protected by security gateways.

The use of security gateways and VPN tunnels to encrypt PMU traffic eliminates many possible vulnerabilities [3, 45], but these devices are still vulnerable to denial-of-service (DoS) attacks that exploit a side-channel vulnerability. In [44], encrypted PMU measurement traffic is identified and dropped through an exploitation of side-channel vulnerabilities. The underlying protocol is detected as described in Sect. 20.3.5.2, and the target protocol's packets are dropped. The attack leads to unstable power grid operations. Figure 20.4 illustrates such an attack. This attack is tough to detect because only the measurement traffic is dropped and all other traffic is untouched (e.g., ping or SSH traffic). From the victim's point of view, all systems work fine, but no measurement traffic is received.

20.3.5.6 Botnet Domain Generation Algorithm (DGA) Modeling and Detection

Botnets have been problematic for over a decade. In response to takedown campaigns, botmasters have developed techniques to evade detection. One widely used evasion technique is DNS domain fluxing. Domain names generated with domain generation algorithms (DGAs) are used as the *rendezvous points* between botmasters and bots. In this way, botmasters hide the real location of the C&C servers

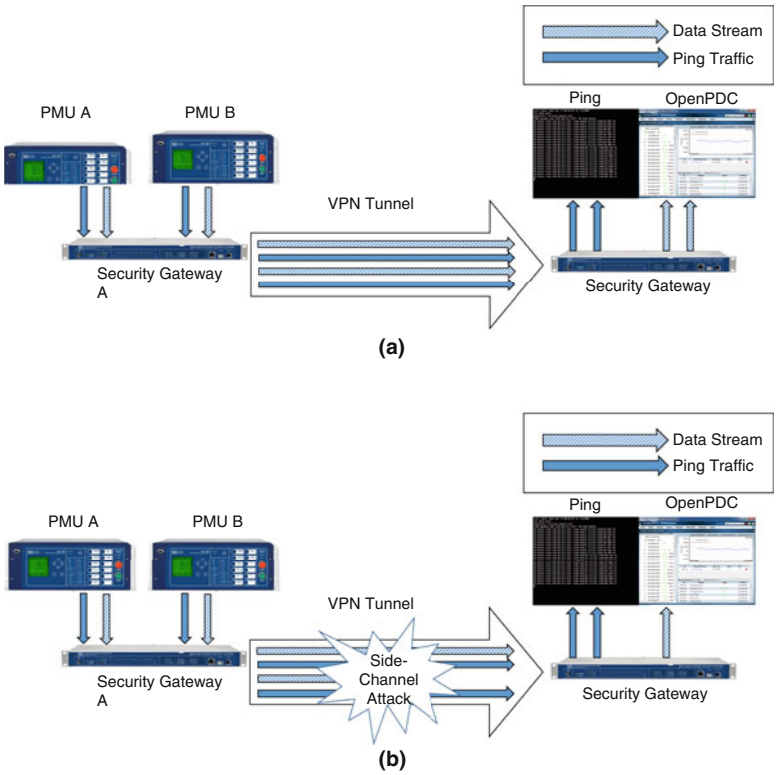


Fig. 20.4 An example of a DoS attack in a VPN tunnel: (**a**) a VPN network carries ping traffic and data streams from two PMUs to control center; (**b**) during a DoS attack, a PMU data stream within the VPN tunnel is identified and dropped

by changing the mapping between IP addresses and domain names frequently. Fu et al. [17] developed a new DGA using HMMs, which can evade current DGA detection methods (Kullback-Leibler distance, edit distance, and Jaccard index) [39] and systems (Botdigger [41] and Pleiades [1]). The idea is to infer an HMM from the entire space of IPv4 domain names. Domain names generated by the HMM are guaranteed to have the same lexical features as the legitimate domain names. With the opposite idea, two HMM-based DGA detection methods were proposed [15]. Since the HMM expresses the statistical features of the legitimate domain names, the corresponding Viterbi path of a given domain name can be found, which indicates the likelihood that the domain name is generated by the HMM. The probability returned by the HMM is a measure of how consistent the domain name is with the set of legitimate domain names.

20.3.5.7 A Covert Data Transport Protocol

Protocol obfuscation is widely used for evading censorship and surveillance and hiding criminal activity. Most firewalls use DPI to analyze network packets and filter out sensitive information. However, if the source protocol is obfuscated or transformed into a different protocol, detection techniques won't work [43]. Fu et al. [16] developed a covert data transport protocol that transforms arbitrary network traffic into legitimate DNS traffic in a server-client communication model. The server encodes the message into a list of domain names and registers them to a randomly chosen IP address. The idea of the encoding is to find a unique path in the HMM inferred from legitimate domain names, which is associated with the message. The client does a reverse-DNS lookup on the IP address and decodes the domain names to retrieve the message. Compared to DNS tunneling, this method doesn't use uncommon record types (TXT records) or carry suspiciously large volume of traffic as DNS payloads. On the contrary, the resulting traffic will be normal DNS lookup/reverse-lookup traffic, which will not attract attention. The data transmission is not vulnerable to DPI.

20.3.5.8 Bitcoin Transaction Analysis

Bitcoin has been the most successful digital currency to date. One of the main factors contributing to Bitcoin's success is the role it has found in criminal activity. According to Christin [9], in 2012, the Silk Road (a popular dark Web marketplace) was handling 1.2 million USD worth of Bitcoin transactions each month. This is largely attributed to Bitcoin's appeal—pseudonymity, which disassociates users with the publicly available transactions.

It is natural to consider financial transactions as a Markovian process since each transaction is governed solely by previous states. All Bitcoin transactions can be publicly viewed on the blockchain, as shown in Fig. 20.5. This motivates the notion that transactions can be represented by a Markov model. However, since Bitcoin is pseudonymous, there are hidden states that correspond to the Bitcoin users. The process of grouping the observable transactions with their respective users infers the underlying HMM.

Theoretically, this HMM would render traditional Bitcoin money laundering useless since existing Bitcoin laundering techniques are reminiscent of a shell game.[1] Current research is focused on inferring the HMM from the Bitcoin blockchain, leveraging existing research that identifies transient transactions used for change functionality [2, 26].

[1] https://en.wikipedia.org/wiki/Shell_game.

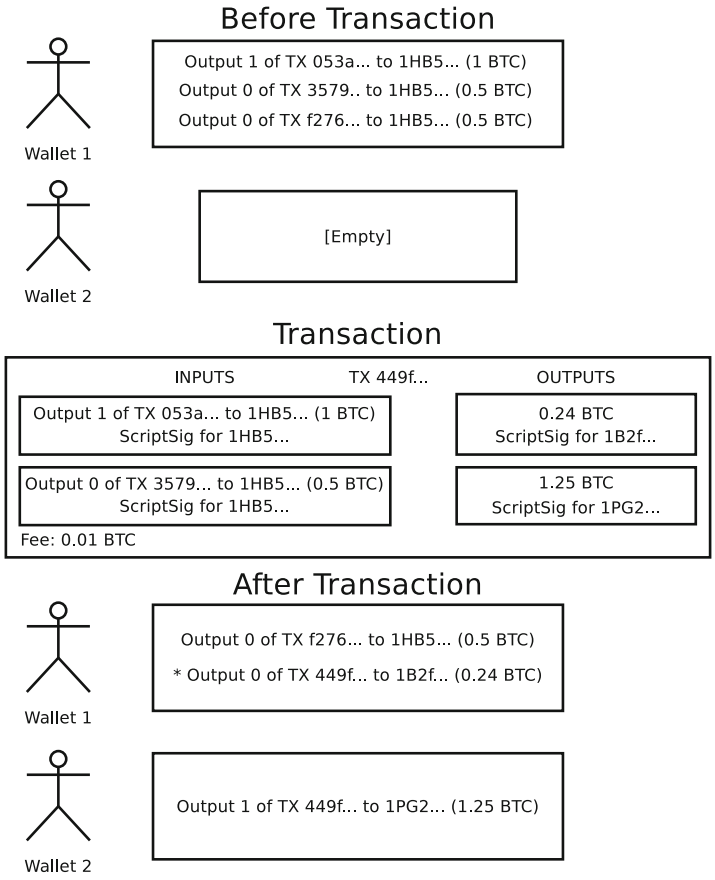


Fig. 20.5 An example of a Bitcoin transaction found in the public blockchain

20.4 Stochastic Signal Processing for Radiation Detection and Localization

20.4.1 Radiation Processes

The detection and localization of radioactive sources, especially those that emit ionizing gamma radiation, are a problem that is of great interest for national security [35]. Such a task is not simple due to the physics of radiation signal propagation, stochastic nature of radiation measurements, and interference from background noise. Assuming a uniform propagation medium between the source and location of measurement, as well as a negligible attenuation due to the medium, a simple radiation propagation model is given by

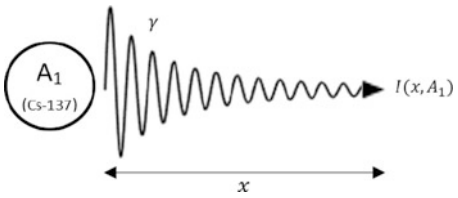


Fig. 20.6 The intensity of a radiation signal decreases exponentially with distance when moving through a uniform medium

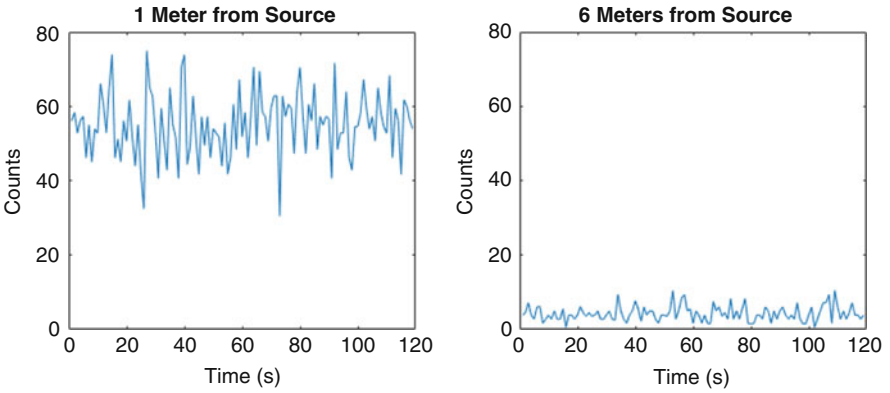


Fig. 20.7 Real measurements of a 35 µCi Cs-137 source at various distances (IRSS datasets [34])

$$I = \frac{A}{x^2} + B, \tag{20.2}$$

where I is the total radiation intensity at the measurement location, A is the intensity of the radioactive source, x is the distance from the source to the measurement location, and B is the intensity of the background radiation at the measurement location.

It is evident from (20.2) that propagation of the signal from a radioactive point source is governed by the inverse-square law, which states that the intensity of the signal is inversely proportional to the distance from the source. Figure 20.6 provides a visual example of the effect the inverse-square law has on the intensity of a radiation signal.

Ionizing radiation is commonly measured using scintillation counters, which integrate the number of times that an incident gamma particle illuminates a scintillation material over a given time period. The number of incidents integrated over each time period is referred to as "counts." Due to the physics of radioactive decay, the measurements of scintillation counters are Poisson random variables [21]. As a result, single radiation measurements are unreliable since the measurement of a high-intensity source will have a proportionally high variance. An example of the difference in variance between a high-intensity and low-intensity signal is given in Fig. 20.7. Observe that the high-intensity signal recorded 1 m from the source has a much higher range of values than the low-intensity signal recorded 6 m from the source.

20.4.2 Maximum Likelihood Estimation

Basic radiation detection methods for national security involve the deployment of one or two sensors in the form of portal monitors located at choke points along the road. While simple and reliable for detection, the use of portal monitors is not practical for several locations, such as a widespread urban environment with complex road networks [6]. Recently, the major focus of research has been on the use of distributed networks of detectors, which are much more suitable for the detection and localization of radiation sources over a widespread area. With a distributed detector network, the detection and localization of radiation sources becomes a complex problem requiring the fusion of large amounts of stochastic data. One of the most common fusion methods used for this purpose is maximum likelihood estimation [8, 19, 38].

In short, maximum likelihood estimation (MLE) is a search over all possible parameters of the target radiation source. While the number of parameters depends on the radiation model, they commonly include the horizontal and vertical coordinates of the source and the radioactive intensity of the source. Each combination of possible parameters is plugged into a likelihood function, which gives the probability that a source with the given parameters causes each detector in the field to have their current measurements. The parameters that generate the highest likelihood are selected as the maximum likelihood estimate of the source. An example likelihood function based on the radiation model in (20.2) is given by

$$L(\theta) = \sum_{i=1}^{N_d} \left(\ln I_i \sum_{j=1}^{w} \left(c_{ij} - 1 \right) \right), \tag{20.3}$$

where N_d is the number of detectors being used for the localization, w is the length of the time window, c_{ij} is the measurement for the ith detector at timestep j, and θ is the vector of input source parameters, which are used to generate the intensity at the ith detector, I_i, using (20.2). The maximization of the likelihood function is then given as

$$\hat{\theta}_{ml} = \arg\max L(\theta), \tag{20.4}$$

where $\hat{\theta}_{ml}$ is the maximum likelihood estimate of the source parameter vector. A visual example of an MLE localization is shown in Fig. 20.8. Note that the likelihood function in (20.3) is a simplification of the logarithm of the joint Poisson probability for the likelihood of the measurements at each individual detector [11].

The primary criticism of MLE localization is that it is computationally intensive, requiring a brute force search over all parameters. A common method for reducing the total computational load of the MLE search is the use of an iterative search [8, 13], which allows the large ranges of parameter space to be excluded from the overall localization. While iterative MLE localization is much faster than standard

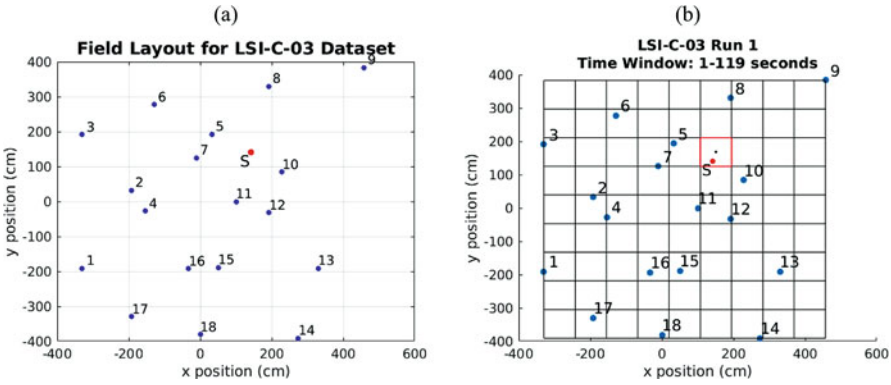


Fig. 20.8 An example of a low-resolution MLE localization: (**a**) layout of a detector field from IRSS datasets [34]. Detectors are blue dots indexed by numbers, and the radiation source is a red dot tagged with an "S." (**b**) A coarse MLE localization has been performed over the detector field. The center of each grid region was used as input parameters for (20.4). The grid region with a red border and a black dot at the center is the area selected by the search

MLE search, throwing out large areas of parameter space makes the search liable to be caught within local maxima [31]. In [11], we proposed the use of grid expansion to mitigate these types of errors. With grid expansion, the search area is expanded by a given percent between each iteration, allowing the MLE search to span into areas that would have been thrown out by a standard iterative algorithm. We found that despite a requiring small increase in computation time, the use of grid expansion corrected errors in the situations where the search was getting caught within local maxima while maintaining good localization performance in cases where the error was not occurring.

20.4.3 Linear Regression

Observe that the radiation propagation model given in (20.2) is linear with respect to $1/x^2$. Furthermore, by the law of large numbers, the average of the Poisson distributed detector measurements over a large time window is a representative of the radiation intensity at their respective locations. Given these two observations, a linear regression model built using the average detector measurements over a large enough time window and their distances to the source location may be used to estimate both the source and background intensities, where the slope of the regression line is the source intensity estimate and the intercept of the regression line is the background intensity estimate. Figure 20.9 shows a linear regression model built using detector data and their inverse squared distances to a source location estimate over a 10 s time window. In [12], we use the source estimate from a linear regression model to successfully detect the presence of a moving source within a distributed detector field.

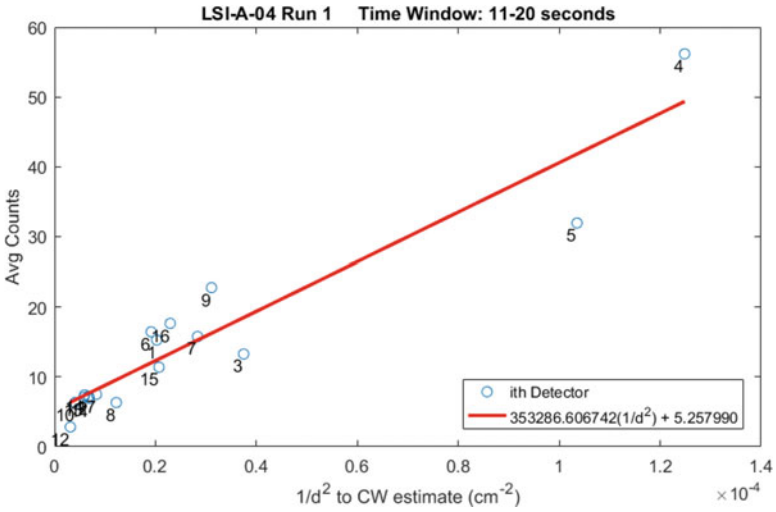


Fig. 20.9 A linear regression model built on average detector measurements vs. their inverse squared distance to an estimate of the source location. The source and background intensity estimates are provided by the slope and intercept values of the line, which are around 3.533×10^5 counts and 5.258 counts, respectively

20.4.3.1 Linear Regression and MLE

One of the difficulties of maximum likelihood estimation is the initialization of the search over state space. In a real-time scenario, it is ideal to use small search ranges as a means to conserve computational power and keep the localization up to date with the influx of detector data. While the search area for the position parameters is typically well defined, the search range over the source intensity parameter is not nearly as obvious and can span a large range of possible values. Furthermore, an MLE localization requires prior knowledge of the background intensity, B, since computation of the likelihood function (20.3) requires the solution of the propagation model (20.2). In [10], we show that the source intensity and background estimates provided by the linear regression model can be used to speed up an MLE localization by reducing the search range over the intensity parameter and allowing the removal of detectors from the localization whose measurements are most likely to only include background noise.

20.5 Conclusions

In this chapter, we highlighted the use of Markov models and stochastic signal processing to learn, extract, fuse, and detect patterns in raw data. In Sect. 20.3, we introduced the deterministic hidden Markov model (HMM), which is a useful tool

to draw information out of a large amount of data. We described the properties of HMMs and highlighted the usefulness of HMMs for several detection applications. In Sect. 20.4, we described the uses of stochastic signal processing for the detection and localization of radiation sources. We highlighted the advantages and drawbacks of localization with maximum likelihood estimation (MLE) and described the estimation of source and background intensities using a linear regression model based on detector counts.

References

1. Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou, N., Abu-Nimeh, S., Lee, W., Dagon, D.: From throw-away traffic to bots: detecting the rise of DGA-based malware. In: USENIX Security Symposium, vol. 12 (2012)
2. Barber, S., Boyen, X., Shi, E., Uzun, E.: Bitter to better—how to make bitcoin a better currency. In: International Conference on Financial Cryptography and Data Security, pp. 399–414. Springer, New York (2012)
3. Beasley, C., Zhong, X., Deng, J., Brooks, R., Kumar Venayagamoorthy, G.: A survey of electric power synchrophasor network cyber security. In: Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES, pp. 1–5 (2014). https://doi.org/10.1109/ISGTEurope.2014.7028738
4. Bhanu, H., Schwier, J., Craven, R., Brooks, R.R., Hempstalk, K., Gunetti, D., Griffin, C.: Side-channel analysis for detecting protocol tunneling. Adv. Internet Things **1**(02), 13 (2011)
5. Bowerman, B.L., O'connell, R.T.: Linear Statistical Models: An Applied Approach. Brooks/Cole, Belmont (1990)
6. Brennan, S.M., Mielke, A.M., Torney, D.C., Maccabe, A.B.: Radiation detection with distributed sensor networks. Computer **37**(8), 57–59 (2004). https//doi.org/10.1109/MC.2004.103
7. Brooks, R.R., Schwier, J.M., Griffin, C.: Behavior detection using confidence intervals of hidden markov models. IEEE Trans. Syst. Man Cybern. B **39**(6), 1484–1492 (2009)
8. Chin, J.C., Yau, D.K., Rao, N.S., Yang, Y., Ma, C.Y., Shankar, M.: Accurate localization of low-level radioactive source under noise and measurement errors. In: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, pp. 183–196. ACM, New York (2008)
9. Christin, N.: Traveling the silk road: a measurement analysis of a large anonymous online marketplace. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 213–224. ACM, New York (2013)
10. Cordone, G., Brooks, R.R., Sen, S., Rao, N.S., Wu, C.Q.: Linear regression for the initialization of radioactive source localization via maximum likelihood estimation (Under preparation)
11. Cordone, G., Brooks, R.R., Sen, S., Rao, N.S., Wu, C.Q., Berry, M.L., Grieme, K.M.: Improved multi-resolution method for MLE-based localization of radiation sources. In: Proceedings of the 20th International Conference on Information Fusion, FUSION 2017 (2017)
12. Cordone, G., Brooks, R.R., Sen, S., Rao, N.S., Wu, C.Q., Berry, M.L., Grieme, K.M.: Linear regression for radioactive source detection. Presented at 2017 IEEE Nuclear Science Symposium, Medical Imaging Conference and Room-Temperature Semiconductor Detector Workshop (NSS/MIC/RTSD) (2017)
13. Deb, B.: Iterative estimation of location and trajectory of radioactive sources with a networked system of detectors. IEEE Trans. Nucl. Sci. **60**(2), 1315–1326 (2013)
14. Eddy, S.R.: Hidden Markov models. Curr. Opin. Struct. Biol. **6**(3), 361–365 (1996)
15. Fu, Y.: Using botnet technologies to counteract network traffic analysis. Ph.D. thesis, Clemson University (2017)

16. Fu, Y., Jiay, Z., Yu, L., Zhong, X., Brooks, R.: A covert data transport protocol. In: 2016 11th International Conference on Malicious and Unwanted Software (MALWARE), pp. 1–8. IEEE, New York (2016)
17. Fu, Y., Yu, L., Hambolu, O., Ozcelik, I., Husain, B., Sun, J., Sapra, K., Du, D., Beasley, C.T., Brooks, R.R.: Stealthy domain generation algorithms. IEEE Trans. Inf. Forensics Secur. **12**(6), 1430–1443 (2017)
18. Griffin, C., Brooks, R.R., Schwier, J.: A hybrid statistical technique for modeling recurrent tracks in a compact set. IEEE Trans. Autom. Control **56**(8), 1926–1931 (2011)
19. Gunatilaka, A., Ristic, B., Gailis, R.: On localisation of a radiological point source. In: Information, Decision and Control, 2007. IDC'07, pp. 236–241. IEEE, New York (2007)
20. Harakrishnan, B., Jason, S., Ryan, C., Richard, R.B., Kathryn, H., Daniele, G., Christopher, G.: Side-channel analysis for detecting protocol tunneling. Adv. Internet Things **1**(2), 13–26 (2011). https://doi.org/10.4236/ait.2011.12003
21. Knoll, G.F.: Radiation Detection and Measurement. Wiley, New York (2000)
22. Kutner, M.H., Nachtsheim, C., Neter, J.: Applied Linear Regression Models. McGraw-Hill/Irwin, Chicago (2004)
23. Lu, C.: Network traffic analysis using stochastic grammars. Ph.D. thesis, Clemson University (2012)
24. Lu, C., Brooks, R.: Botnet traffic detection using hidden Markov models. In: Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research, p. 31. ACM, New York (2011)
25. Lu, C., Schwier, J.M., Craven, R.M., Yu, L., Brooks, R.R., Griffin, C.: A normalized statistical metric space for hidden Markov models. IEEE Trans. Cybern. **43**(3), 806–819 (2013)
26. Ober, M., Katzenbeisser, S., Hamacher, K.: Structure and anonymity of the bitcoin transaction graph. Future Internet **5**(2), 237–250 (2013)
27. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE **77**(2), 257–286 (1989)
28. Schwier, J.: Pattern recognition for command and control data systems. PhD dissertation, Clemson University (2009)
29. Schwier, J.M., Brooks, R.R., Griffin, C., Bukkapatnam, S.: Zero knowledge hidden Markov model inference. Pattern Recogn. Lett. **30**(14), 1273–1280 (2009)
30. Schwier, J.M., Brooks, R.R., Griffin, C.: Methods to window data to differentiate between Markov models. IEEE Trans. Syst. Man Cybern. B **41**(3), 650–663 (2011)
31. Sheng, X., Hu, Y.H.: Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. IEEE Trans. Signal Process. **53**(1), 44–53 (2005)
32. Sin, B.K., Ha, J.Y., Oh, S.C., Kim, J.H.: Network-based approach to online cursive script recognition. IEEE Trans. Syst. Man Cybern. B **29**(2), 321–328 (1999)
33. Song, D.X., Wagner, D., Tian, X.: Timing analysis of keystrokes and timing attacks on SSH. In: Proceedings of the 10th Conference on USENIX Security Symposium - Volume 10, SSYM'01. USENIX Association, Berkeley, CA, USA (2001). URL http://dl.acm.org/citation.cfm?id=1251327.1251352
34. U.S. Department of Homeland Security: Intelligent radiation sensing system. URL https://www.dhs.gov/intelligent-radiation-sensing-system
35. U.S. Government Publishing Office: Nuclear terrorism: strengthening our domestic defenses. Senate Hearing 111–1096, Hearings before the Committee on Homeland Security and Governmental Affairs U.S. Senate, 111th Congress, 2nd Session (2010)
36. Van, B.L., Garcia-Salicetti, S., Dorizzi, B.: On using the Viterbi path along with HMM likelihood information for online signature verification. IEEE Trans. Syst. Man Cybern. B **37**(5), 1237–1247 (2007)
37. Vanluyten, B., Willems, J.C., De Moor, B.: Equivalence of state representations for hidden Markov models. Syst. Control Lett. **57**(5), 410–419 (2008)
38. Vilim, R., Klann, R.: Radtrac: a system for detecting, localizing, and tracking radioactive sources in real time. Nucl. Technol. **168**(1), 61–73 (2009)

39. Yadav, S., Reddy, A.K.K., Reddy, A., Ranjan, S.: Detecting algorithmically generated malicious domain names. In: Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement, pp. 48–61. ACM, New York (2010)
40. Yu, L., Schwier, J.M., Craven, R.M., Brooks, R.R., Griffin, C.: Inferring statistically significant hidden markov models. IEEE Trans. Knowl. Data Eng. **25**(7), 1548–1558 (2013)
41. Zhang, H., Gharaibeh, M., Thanasoulas, S., Papadopoulos, C.: Botdigger: Detecting DGA bots in a single network. In: Proceedings of the IEEE International Workshop on Traffic Monitoring and Analysis (2016)
42. Zhong, X., Arunagirinathan, P., Ahmadi, A., Brooks, R., Venayagamoorthy, G.K., Yu, L., Fu, Y.: Side channel analysis of multiple PMU data in electric power systems. In: Power System Conference (PSC), 2015 Clemson University, pp. 1–6 (2015)
43. Zhong, X., Fu, Y., Yu, L., Brooks, R., Venayagamoorthy, G.K.: Stealthy malware traffic-not as innocent as it looks. In: 2015 10th International Conference on Malicious and Unwanted Software (MALWARE), pp. 110–116. IEEE, New York (2015)
44. Zhong, X., Jayawardene, I., Venayagamoorthy, G.K., Brooks, R.R.: Denial of service attack on tie-line bias control in a power system with PV plant. IEEE Trans. Emerg. Top. Comput. Intell. **1**(5), 375–390 (2017)
45. Zhong, X., Yu, L., Brooks, R., Venayagamoorthy, G.K.: Cyber security in smart DC microgrid operations. In: 2015 IEEE First International Conference on DC Microgrids (ICDCM), pp. 86–91. IEEE, New York (2015)

Chapter 21
A Control-Based Modeling Approach for Simulating Reaction to Stress Interventions

Juan M. Calderon and Luis G. Jaimes

Abstract A recent study showed that around 77% of the US population experiences regularly physiological symptoms of stress such as sleep deprivation, fatigue, and migraines and 73% psychological symptoms such anxiety, angry, and lack of focus. Thus, psychological stress remains an important issue of public health. In this paper, we propose a dynamical system based on control theory for simulating and testing the human reaction to stress interventions. We use the well-known analogy of a fluid reservoir or tank to model the user's accumulation of stress. In this model, inputs correspond to stressors which increase the level of fluid, and the outputs correspond to interventions which release stress and drain the tank. Thus, we tackle the problem keeping a user's healthy stress level by using a fuzzy logic approach. Through several examples and extensive simulations, we evaluate the performance of our proposed mechanism.

Keywords Health interventions · Control theory · Fuzzy logic

21.1 Introduction

Data streams coming from these sensors are used as the input of inference models. These models usually have the form of machine learning algorithms, which are trained using ground-truth data to recognize the condition of a patient [6, 7, 12]. For instance, by using respiration rate and accelerometer data, it is possible to infer

J. M. Calderon (✉)
Bethune-Cookman University, Daytona Beach, FL, USA

Santo Tomas University, Bogotá, Colombia
e-mail: juancalderon@usantotomas.edu.co

L. G. Jaimes
Florida Polytechnic University, Lakeland, FL, USA
e-mail: ljaimes@floridapoly.edu

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7_21

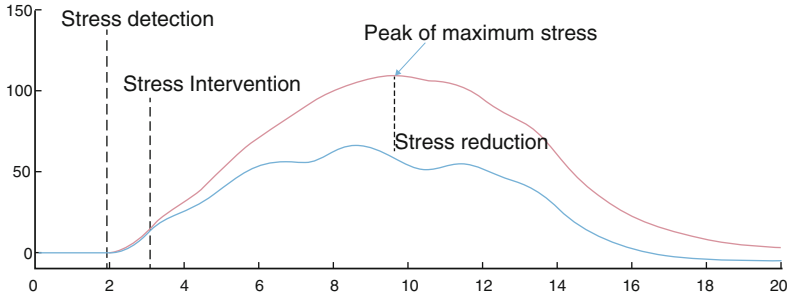


Fig. 21.1 Stress model

chest expansion. Similarly, by monitoring arm movement, we can infer smoking behaviors [4, 15]. Lastly, a stress model could use the arousal of HRV, RR, and ST signals to infer stress [2].

Based on the information provided by the first layer, a recommender system in the second layer takes the decision of whether or not to intervene [8]. Additionally, if the decision is to intervene, the system has to decide in an autonomous way when to intervene (i.e., the timing), the intervention doses, and chooses the intervention that maximizes the effectiveness of the treatment [5].

However, before experimenting with real users, it seems reasonable to tune the system by simulation before using it with real humans. Some research in this area corresponds to the work of Jaimes [5, 8] who proposed to use a probability sampling mechanism from beta distribution in order to model the human reaction to a set of intervention. However, the use of beta distribution seems very simplistic for simulating human reaction to intervention. Another is the work of Murray [10] who formalizes a set of computational model also based on control theory. A well-known body of work in this field corresponds to the work of Rivera [13].

In this paper, we model the human reaction to an intervention by proposing a modified version of the analogy of the fluid level in a tank [13]. Here, the inputs represent a set of stressors, and the main tank represents a human who is able to cope with them. However, once a threshold is reached, an alarm is trigged which indicates that it is time to deliver an intervention, in order to control the stress' levels. In our model, which is represented in Fig. 21.1, an intervention is represented by the action of opening the output in order to dray the tank [1] and regulate the stress level.

21.2 Related Work

Delivery of supportive therapies and treatments often takes place in medical facilities such as hospitals and rehabilitation institutions. The disadvantage of this traditional approach is the limited access that the average population has to public or private health systems. In addition, in some cases, the capacity of health institutions is not enough to follow up with treatments for the entire patient population.

Therefore, institutions often have to prioritize based on the severity of the conditions and the availability of medical personal and budget.

A first step to address this problem was the development of ecological momentary assessment (EMA) [16]. EMA was developed by the social behavioral community to gather user's feedback in real time. This feedback often has the form of descriptions about feelings or circumstances that took place when a patient experienced a crisis (e.g., smoking cravings, stress). This feedback, which gathers a natural environment and records in real time, is used later for treatment support. The logic about the use of EMA is that patients in follow-up sessions usually forget the details about the exact circumstances about what happened when the crisis event took place.

A second milestone toward treatment support in natural environments was the development of ecological momentary intervention (EMI) [3]. The goal of EMI is to complement and reinforce the treatments that take place in the health institutions by the use of interventions in natural environments. EMI-EMA is used to gather the user's feedback (i.e.,written descriptions, non-sensor data), whereas EMI is used to provide treatment in semi-real time. This support has the form of a phone call or a text message sent by the physician or caregiver. This combination has been used as a treatment mechanism for a variety of disorders such as smoking cessation [15], weight loss [9], anxiety reduction [11], as well as eating disorder reduction [14]. In all the cases, the intervention works in an asynchrony way (i.e., they are not just in time interventions); they respond when the user requires help, and not by the activation of an automatic sensor-based system.

21.3 Stress Model

We model stress as positive continuous signal whose amplitude corresponds to the level/intensity/severity of stress. Thus, waves in the signal (a period of increasing intensity followed by decreasing intensity) are considered stress episodes; see Fig. 21.2. From this, we say that an intervention is effective if it modifies the shape of the original stress episode wave, such that the stress episode has a shorter duration and/or is less intense. The level of effectiveness of the intervention is thus measured as the difference in width and height between the original stress wave (i.e., no intervention) and the intervention-modified stress wave.

21.3.1 Stress Episodes vs Stress as a Continuous Stress Signal

There are two dominant approaches for simulating the evolution of an stress signal over a long period of time. A first approach considers stress as episodic phenomena. The stress episode is defined as an interval between the time when stress is detected and the time when the patient shows no more signs of stress.

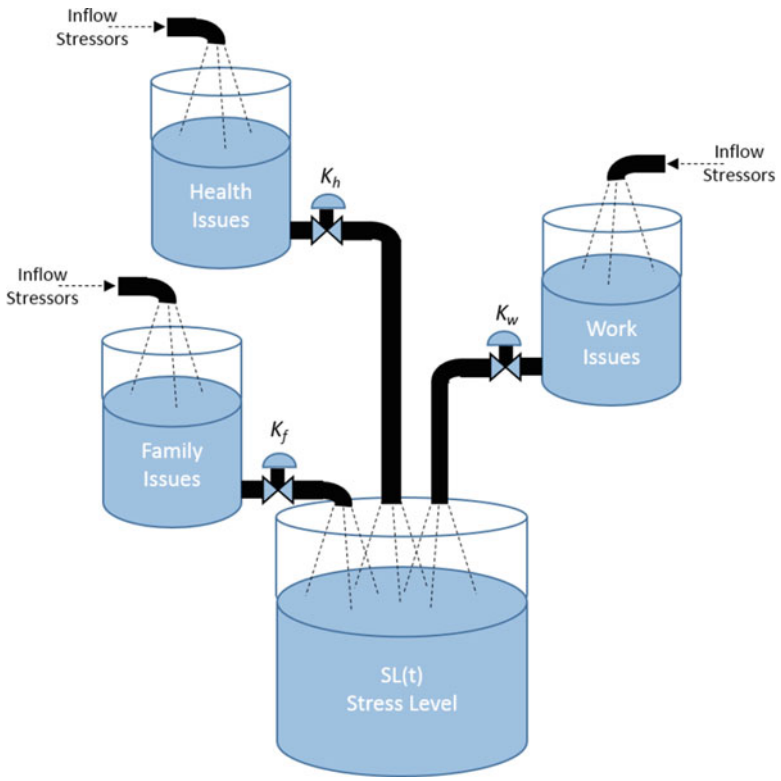


Fig. 21.2 Stress wave

The second approach models stress as a fluctuation of a continuous signal that represents stress disturbances. In these cases, a system continuously monitors the patient and tries to keep a smooth signal by providing different types of interventions, with different levels of intensity, and sometime providing a null intervention, namely, when the patient does not show any manifestation of stress. In this paper, we assume the second approach.

21.3.2 Patient Model

The patient model is one of the most important stages of this work, because the proposed model allows to focus on different and important characteristics of the patient behavior. The patient model tries to have in account the different kinds of situations that can affect the stress level.

The patient is modeled as a dynamic system, who is affected by different external situations such as labor problems, health issues such as illnesses, and family and personal matters among any more regular life complications. We take inspiration from control theory to model the patient's well-being in terms of stress as a water

reservoir. Here, the level of water is assumed as patient's stress level. Different daily life situations might contribute to the stress level on different ways depending on patient's ability to cope with those situation. According to the proposed model, the patient stress level is modeling as a group of reservoirs as show by Fig. 21.2.

$$\mathrm{SL}(t) = \sum_{i=1}^{n} K_i f_i(t) \tag{21.1}$$

where SL is level stress, K_i is the general gain of different kinds of stress generator, and $f_i(t)$ is the way stress generator affects the stress level. It is worthy to clarify the conceptual difference between K_i and f_i. K_i is the level of stress generator that can affect the stress level. It is directly related with how much important is the problem for the patient, and this gain is represented by a constant value. The f_i represents the affectation way of the stress generator over the stress level. It allows to model different characteristics of stress generators such as how fast the issue can affect the stress level or how long is the affectation of the issue over the healthiness of the patient. The f_i is represented as a dynamic system of the 1st or 2nd order according to the affectation way. The f_i is usually represented by a differential equation in the time (t) domain or a transfer function in the frequency (s) domain.

Figure 21.3 shows the stress level effect of the stress generator using different values of K_i. Different values of K_i generate different stress levels, but it does not change the affectation way such as time to reach the maximum stress level.

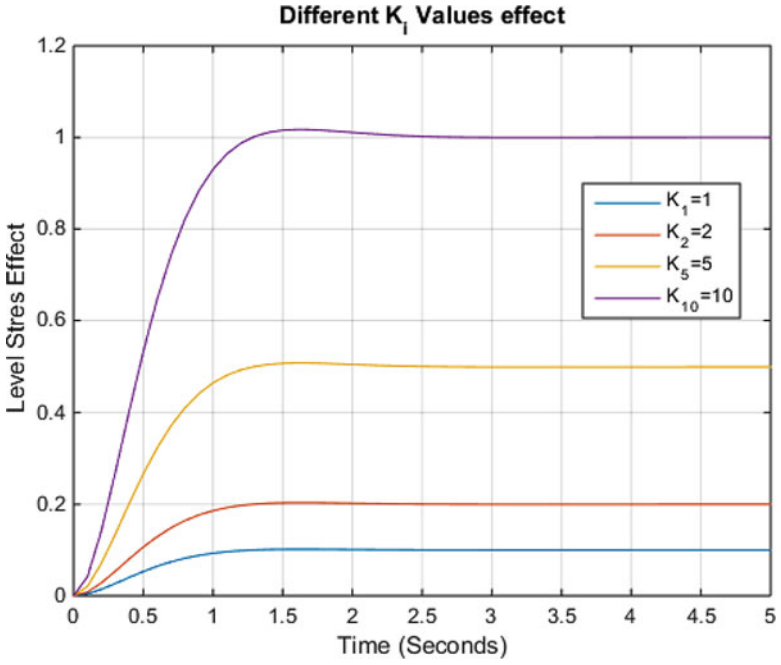


Fig. 21.3 K_i value effect

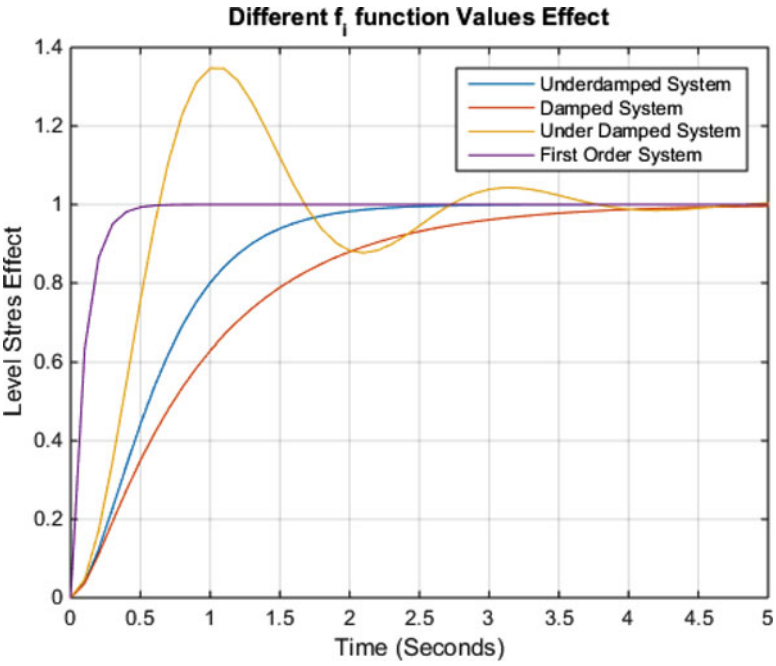


Fig. 21.4 f_i function value effect

Figure 21.4 shows the time response of the stress generator. It is modeled through different transfer function versions. Figure 21.4 shows the time response of several f_i models such as overdamped, damped, underdamped, and first order. The settling time changes according to any model and the way to reach the steady-state value. Those are just a simple example about how f_i can change the model of the stress generator according to any kind of life issues.

21.3.3 Experiments and Results

This section presents a set of four experiments. For the different experiments, we vary three inputs, namely, the severity of a stressful event, the duration of the stress episode caused by that event, and the patient's resilience or ability to cope with stressful situations. Each experiment includes the following information. (1) Time at which the patient reaches his/her maximum level of stress. (2) Period of time the patient remains in his/her maximum stress level. (3) Time for recovering from a stressful situation, namely time that takes to stabilize the stress signal.

21.3.3.1 Experiment 1

Here, we simulate the effects of external stressors and factors on the stress levels. Here, an external event causes a slowly increase in the patient's stress which reaches

its maximum peak or climax after 30 days. Examples of these types of stressors include adverse financial situations such as lack of money to pay the bills. Thus, at the end of the month, the stress reaches its maximum level. The case is depicted in Fig. 21.5.

21.3.3.2 Experiment 2

In this experiment, we vary the time from 0 to 8 (x-axis) in which the patient is exposed to a stressful situation. As Fig. 21.6 shows, the maximum peak of stress is

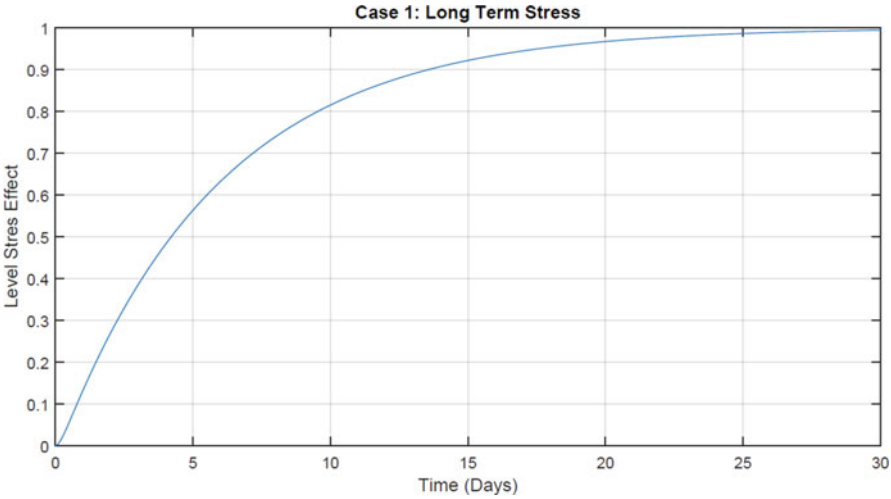


Fig. 21.5 Case 1: Long-term stress

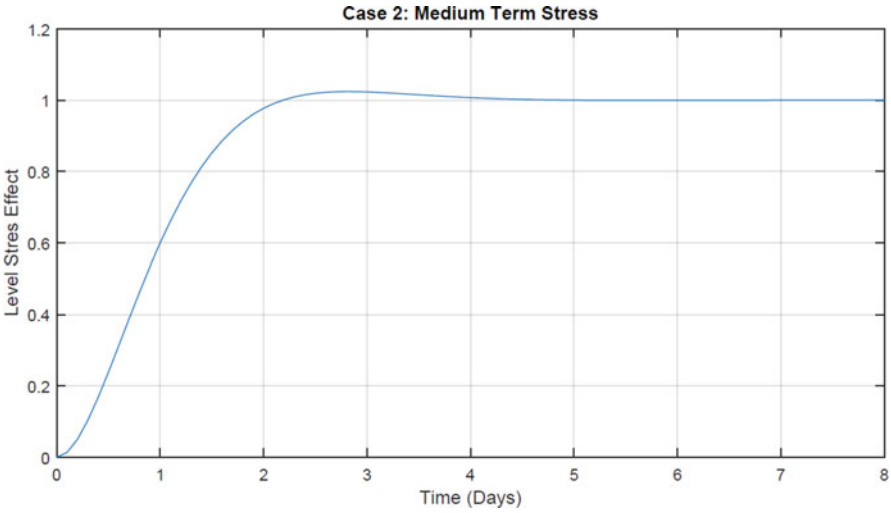


Fig. 21.6 Case 2: Medium-term stress

reached very fast. This type of stressors may correspond to a close project deadline. This situation will produce an amount of stress that will remain until the end of the project.

21.3.3.3 Experiment 3

This experiment shows how the stress reaches its maximum value in a short period of time. This increment could be generated by unexpected discussions or brief incident while driving home. Here, we observe a rapid increment of short duration followed by rapid decrement of stress shown in Fig. 21.7.

21.3.3.4 Experiment 4

This experiment shows a variety of situations. The first case shows the stress levels for a period of 30 days. A longtime issue is present as a permanent concern such as the payment of a mortgage at the end of the month. A medium-term issue spans for 10 days starting at 5th day of the month. And two short-term stressors happen at days 15th and 25th, respectively. These could be a result of short and unexpected family discussion that generates a fast increase of the stress levels but with a short-term duration. Thus, the total stress level is represented as sum of the different kinds of concerns presented through the month. Thus, each type of situation may affect each person at a different level which also depends on the person level of resiliency (Fig. 21.8).

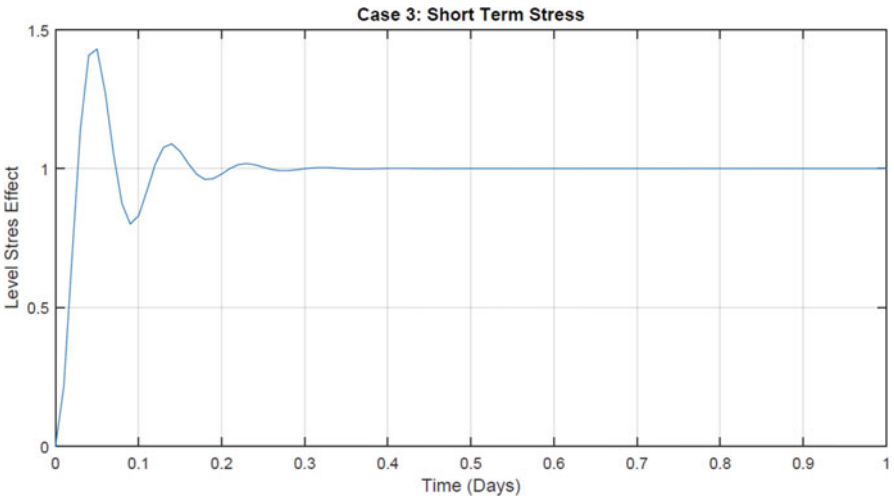


Fig. 21.7 Case 3: Short-term stress

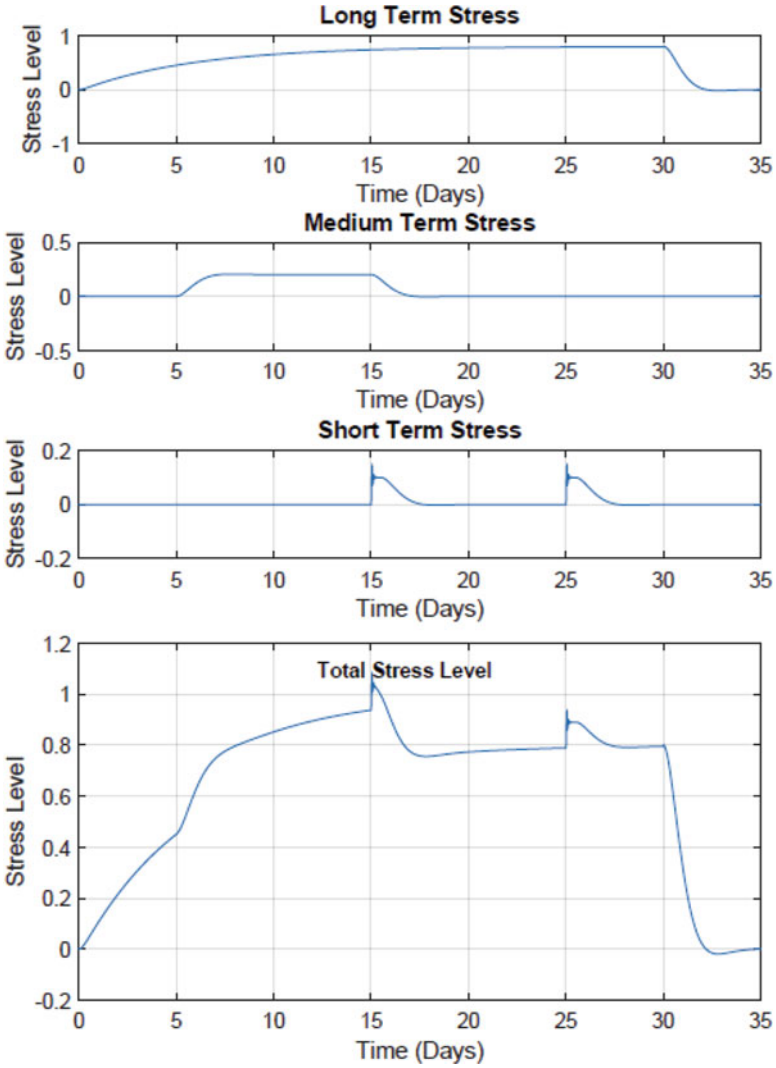


Fig. 21.8 Case 4: Total stress level

These cases try to show how it is possible to model the different types of problems that a person may encounter in their daily lives.

Problems can vary in duration, intensity, and type of affectation. The different combination of these parameters represent a different type of persons, and they should be adjusted as you get more knowledge of patient context in order to correctly model the patient's reaction to the problems of daily life.

21.3.4 Conclusions

This work presents a system to model the behavior of the level of stress for a patient according to the type of stressful situation that he/she faces in everyday life.

The model is based on assuming the behavior of stress levels as a dynamic system over time, assigning different levels of affectation on the patient and different duration times. The model is based on a system of differential equations (most of them are second order), which is used to describe the type of problem and create a general model based on the reservoir model.

The idea of modeling the behavior of a patient's stress levels is centered on having a mathematical model of the patient that will allow the future application of control theory techniques for the selection of personalize stress interventions.

References

1. Calderon, J., Jaimes, L.G.: A fuzzy control-based approach for the selection of health interventions. In: 15th IEEE Annual Consumer Communications & Networking Conference 2018, pp. 1–6. IEEE, Las Vegas (2018)
2. Ertin, E., Stohs, N., Kumar, S., Raij, A., Al'Absi, M., Shah, S.: Autosense: unobtrusively wearable sensor suite for inferring the onset, causality, and consequences of stress in the field. In: SenSys, pp. 274–287 (2011)
3. Heron, K.E., Smyth, J.M.: Ecological momentary interventions: incorporating mobile technology into psychosocial and health behaviour treatments. Br. J. Health Psychol. **15**(1), 1–39 (2010)
4. Jaimes, L.G., Raij, A.: Forecasting biorhythms for preventative stress interventions. In: Proceedings of the ACM Workshop of Biological Rhythms and Technology, at CHI 2014. ACM, New York (2014)
5. Jaimes, L.G., Llofriu, M., Raij, A.: A stress-free life: Just-in-time interventions for stress via real-time forecasting and intervention adaptation. In: 9th International Conference on Body Area Networks, BODYNETS, London (2014)
6. Jaimes, L.G., Llofriu, M., Raij, A.: Calma, an algorithm framework for mobile just in time interventions. In: IEEE-SoutheastCon 2015, pp. 1–5, April 2015
7. Jaimes, L.G., Calderon, J., Lopez, J., Raij, A.: Trends in mobile cyber-physical systems for health just-in time interventions. In: SoutheastCon 2015, pp. 1–6. IEEE, New York (2015)
8. Jaimes, L.G., Llofriu, M., Raij, A.: Preventer, a selection mechanism for just-in-time preventive interventions. IEEE Trans. Affect. Comput. **7**(3), 243–257 (2016)
9. Joo, N.-S., Kim, B.-T.: Mobile phone short message service messaging for behaviour modification in a community-based weight control programme in korea. J. Telemed. Telecare **13**(8), 416–420 (2007)
10. Murray, T., Hekler, E., Spruijt-Metz, D., Rivera, D.E., Raij, A.: Formalization of computational human behavior models for contextual persuasive technology. In: International Conference on Persuasive Technology, pp. 150–161. Springer, Berlin (2016)
11. Newman, M.G., Consoli, A., Taylor, C.B.: Computers in assessment and cognitive behavioral treatment of clinical disorders: anxiety as a case in point. Behav. Ther. **28**(2), 211–235 (1997)
12. Plarre, K., Raij, A., Hossain, S.M., Ali, A.A., Nakajima, M., al'Absi, M., Ertin, E., Kamarck, T., Kumar, S., Scott, M., Siewiorek, D.P., Smailagic, A., Wittmers, L.E.: Continuous inference of psychological stress from sensory measurements collected in the natural environment. In: IPSN, pp. 97–108 (2011)

13. Riley, W.T., Rivera, D.E., Atienza, A.A., Nilsen, W., Allison, S.M., Mermelstein, R.: Health behavior models in the age of mobile interventions: are our theories up to the task? Transl. Behav. Med. **1**(1), 53–71 (2011)
14. Robinson, S., Perkins, S., Bauer, S., Hammond, N., Treasure, J., Schmidt, U.: Aftercare intervention through text messaging in the treatment of bulimia nervosa feasibility pilot. Int. J. Eat. Disord. **39**(8), 633–638 (2006)
15. Rodgers, A., Corbett, T., Bramley, D., Riddell, T., Wills, M., Lin, R.-B., Jones, M.: Do u smoke after txt? results of a randomised trial of smoking cessation using mobile phone text messaging. Tob. Control. **14**(4), 255–261 (2005)
16. Stone, A.A, Shiffman, S.: Ecological momentary assessment (ema) in behavorial medicine. Ann. Behav. Med. **16**, 199–202 (1994)

Author Index

B
Babu, B.S., 63–82
Bhargavi, K., 63–82
Bhat, R., 85–94
Bichindaritz, I., 35–43
Brooks, R.R., 197–204, 265–281

C
Calderon, J.M., 285–294
Chen, S.-C., 137–145
Cordone, G., 265–281

E
Elish, K.O., 1–11

F
Fu, Y., 265–281

H
Ha, H.-Y., 137–145
Han, M., 185–195
Harrison, C., 1–11

I
Iyengar, S.S., 97–109, 125–135

J
Jaimes, L.G., 285–294

K
Keshan, N., 35–43
Krishnamachari, B., 173–183
Kumar, V., 113–122

L
Liu, Q., 13–31
Livingston, J., 1–11

M
Maddumala, M.N., 113–122
Madria, S., 147–160
Mahesh Kumar, K.M., 163–171

N
Nguyen, Q., 173–183

O
Oakley, J., 265–281

P
Parimi, P.V., 35–43
Parvez, I., 45–59
Patnaik, L.M., 245–263
Pattar, S., 245–263
Phoha, V.V., 35–43
Pramod, T.C., 97–109, 219–232
Presa Reyes, M.E., 137–145
Presa Reyes, S., 137–145

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7

Q
Quinn, M., 1–11

R
Rao, N.S.V., 13–31, 185–195
Ravi, N., 235–244
Roopa, M.S., 245–263

S
Sanjeev, K.R., 125–135
Sarwat, A.I., 45–59
Sen, A., 147–160
Shanthakumara, A.H., 207–216
Sundararajan, A., 45–59
Sunitha, N.R., 85–94, 97–109, 163–171, 207–216, 219–232, 235–244

T
Thejas, G.S., 97–109

V
Venugopal, K.R., 245–263
Vidya, A., 245–263

W
Wang, A., 185–195
Wu, C., 185–195

Y
Yu, L., 197–204, 265–81

Z
Zheng, H.C., 137–145
Zhong, X., 265–281
Zhu, M., 185–195
Zuo, L., 185–195

Subject Index

A
Accelerated AdaBoost, 73–74
Access control mechanism
  access control matrix model, 98
  access control policy, 98
  adaptability, 98
  adaptive trust negotiation, 99
  IoT, 100
  objective, 98
  peer-to-peer collaborative spam detection application, 99
  performance analysis, 108–109
  SATBAC framework, 98
    ARTVC unit (*see* Access request trust value calculation unit)
    e-commerce online shopping example, 106–108
    proposed framework, 100, 101
    reward-punishment strategy (*see* Game theory-based reward-punishment strategy)
  security mechanism, 98
  trust attribute model, 99–100
  trust-based access control, 99
  trust management, 98, 99
  trust negotiation policy, 98
Access request trust value calculation (ARTVC) unit
  behavior and history, 101–102
  credential, 102
  location, 103
  opinion and reputation, 100–101
Accuracy percentage (ACC), 39–42
AdaBoost, 73–74
Advanced metering infrastructure (AMI) smart meters, 46, 56–57
  See also Local area sensor network
Affordable Care Act, 4
Aggregate Poincaré maps, 23
Aging
  binary associative classifier, 249
  data mining techniques, 248–249
  dataset, 257–258
  document classification problem, 249–251
  execution flow, 251–252
  gene-document
    classification task, 251–252
    document classification problem, 249–251
    gene sequences, 252–253
    prediction accuracy, 253
    rule mining, 256
    rule pruning, 256–257
    TASB-AC algorithm, 251–252
    testing, 257
    transactional dataset, 253–256
  gene subsequence length
    vs. accuracy of classifier, 259–260
    GO terms, 260–261
    not-aging class, 261–262
    vs. number of CARs generated, 260–261
  performance evaluation, 258–259
Apriori algorithm, 80–81
Artificial reality (AR), 156
ARTVC unit, *see* Access request trust value calculation unit
Association rules, 248

N. S. V. Rao et al. (eds.), *Proceedings of International Symposium on Sensor Networks, Systems and Security*, https://doi.org/10.1007/978-3-319-75683-7

Associative classifier, 248
Authentication path, 209–210

B
Bandwidth reservation request (BRR)
  aggregated TB (ATB), 188–189
  average transfer performance, 187
  BSBS, 192–194
  data transfer request, 187
  example, 188
  FSBS
    modified Dijkstra's algorithm, 189–191
    performance evaluation, 193–194
    returned bandwidth reservation option, 191–192
    TreeSet, 189–190
    "while" loop, 189
  5-tuple, 189
  3-tuple of time-bandwidth (TB), 188
Basic Streaming Bandwidth Scheduling (BSBS), 192–194
Baum-Welch algorithm, 266–267
BBR, 19, 20
BIDMachRF, 68
Big Data
  advanced data analysis techniques, 64
  application, 2
  data breaches, 5–6, 10
  definition, 1
  disasters (*see* Disaster management)
  ethics, 2–3
  in finances, 4
  in fitness, 5
  GPUs, 64
  in healthcare
    Affordable Care Act, 4
    health informatics, 4
    patient care, 4
    security and privacy, 3
  implementation level and processing level challenges, 64
  information flow
    data collector, 6, 7, 9
    data miner, 6–9
    data provider, 7, 8
    decision maker, 6, 8, 9
    security standards, 8–9
    stages, 6, 10
  machine learning algorithms
    analytic applications, 81–82
    GPU empowering (*see* GPU empowering machine learning)
    taxonomy, 64, 65
  in retail, 5
  security and privacy, 3–4
  three Vs, 63
Binary SVM, 75
Bitcoin, 275–276
Botnets
  attacks, 133–134
  history, 133
  traffic, 273
  vulnerabilities in IoT devices, 126

C
CCDF, *see* Complementary cumulative density function
Cloud computing
  deployment models, 114
  features, 114
  NIST definition, 113
  tailor-made firewall systems, 113
Cloud data centers and security, 114–115
Cloud infrastructure, 238–239
Cloud policies, *see* Dynamic firewall policy management, private cloud
Cloud service provider (CSP), 238–239
Command and control (C&C) server, 273
Common Intrusion Detection Framework (CIDF), 198
Community cloud, 114
Complementary cumulative density function (CCDF)
  direction comparison, 181–183
  time comparison, 177, 178
Computationally bounded PIR (cPIR), 87–88
Compute Unified Device Architecture (CUDA), 67, 71, 72, 79
Confidence intervals (CIs), 268
Counts, 277
Cross validation (CV), 36, 39, 40, 41, 42
CUDA based Decision Tree (CUDT), 66
CUDAT, 66
CudaTree, 68
Cyber-physical systems (CPS), 126, 130
Cybersecurity, 154–156

D
Database management system (DBMS), 3
Data breaches, 5–6, 10
Data collector/concentrator, 6, 7, 9, 57
Datagram Transport Layer Security (DTLS), 155
Data miner, 6–9
Data privacy, 89

Data provider, 7, 8
Data transfers
  dedicated connections *vs.* shared connections, 14
  disk-to-disk transfers, 14
  file transfer measurements
    filesystems, 24
    Lustre over wide-area networks, 28–30
    over long-haul connections, 24
    wide-area disk-to-disk file transfer, 24
    XDD, 25–28
  memory-to-memory transfers, 14
  network testbed with FileSystems, 16–17
  research advances, 31
  throughput profiles (*see* Throughput profiles)
DDoS attack, *see* Distributed denial of service attack
Decentralized partially observable Markov decision process (DEC-POMDP)
  distributed system, 201–202
  NLP-based solution, 203–204
  observation probabilities, 202
Decision maker, 6, 8, 9
Decision tree (J48), 39
Deep belief network, 74–75
Deep Q-learning, 70–71
Deep SARSA learning, 69–70
Denial of service (DoS), 238–239
Disaster management
  disaster data modeling, 138
  IoT, 150–151
  multimedia data
    analysis, 140–143
    data management framework, 138
    future directions, 144–145
    geographic data, 139–140
    sensors data, 139
    social media data, 140
    storing and sharing data, 140
    3D storm surge impact animation, 142–144
    visual data, 139
  preparedness, 138
  prevention/mitigation, 138
  recovery, 138
  response, 138
Disaster situation awareness, *see* Disaster management
Distress, 36
Distributed denial-of-service attacks (DDoS), 133, 134, 273
Distribution management system (DMS), 46
DNA repairs genes
  aging
    binary associative classifier, 249
    data mining techniques, 248–249
    dataset, 257–258
    document classification problem, 249–251
    execution flow, 251–252
    gene-document, 251–257
    gene subsequence length, 259–262
    performance evaluation, 258–259
  CARs, 247–248
  diseases, 246
  DNA damage, 246–247
  evolutionary defense mechanism, 246
  motivation, 247
  organization, 248
  sliding-window algorithm, 251–252, 255
DNP3
  application layer, 226, 228
  data link layer, 226, 228
  pseudo-transport layer, 226, 228
  Wireshark tool, 227–232
DNS traffic, 275
Domain generation algorithms (DGAs), 273–274
Dynamic Analyzer and Smart Decider for Virtual Machine (DASDVM), 241–243
Dynamic firewall policy management, private cloud
  anomalies in policy configuration, 118–119
  centralized topology, 115
  distributed topology, 115
  dynamic policies, 116–118
  GPM
    architecture, 120–121
    dual mode, 120
    policy configuration, 121–122
  group policies, 116
  internal firewall, 115
  location-based compliance policies, 116
  perimeter firewall, 115–116
  perimeter policies, 116
  webserver-based security policies, 116
Dynamic IP address blacklist, 117–118

E

ECDHE private key (symmetric key) exchange, 166, 167, 169, 170
Ecological momentary assessment (EMA), 287
Ecological momentary intervention (EMI), 287

Electrocardiogram (ECG) signals
classifiers, 39
extracted features, 39
fiducial points, 36
instantaneous heart rate, 37
multisensor stress analysis, 36
non-fiducial points, 36
and QRS wave complex, 38
ROC area and ACC, 39–42
R-R interval, 39
significance of work, 37
Encounter duration statistics
data distribution, 174
direction
CCDF, 181, 182
KS and AD test result, 181, 183
location, 180–181
time
CCDF, 177, 178
contact distribution variation, 180
KS and AD test result, 179
mean and variance, 179
Energy management system (EMS), 46
Enterprise information system (EIS), 46, 47
Eustress, 36
EVDIC, 239–240
Evidence Record Syntax (ERS), 210, 214

F
False negative (FN) rates, 198, 201
False positive (FP) rates, 198, 201
Fiducial points, 36
File transfer measurements
filesystems, 24
Lustre over wide-area networks
Ethernet clients, 29
features, 28
throughput profiles, 29, 30
using LNet routers, 29
over long-haul connections, 24
wide-area disk-to-disk file transfer, 24
XDD
GridFTP with direct I/O, 28
I/O threads, 25
mean default I/O Lustre file write rates, 25–27
mean XFS file write rates, 25, 26
OSTs, 25
throughput profiles, 28
XFS-to-XFS transfers, 25
Finite state machines (FSM), 203–204
Fixed path with fixed bandwidth (FPFB), 187
Fixed path with variable bandwidth (FPVB), 187
Flexible Streaming Bandwidth Scheduling (FSBS)
modified Dijkstra's algorithm, 189–191
performance evaluation, 193–194
returned bandwidth reservation option, 191–192
TreeSet, 189–190
"while" loop, 189
Frequency disturbance recorders (FDRs), 48
Friend-Finding (FF), 164
Functional entity (FE), 46, 47
Fuzzy logic approach, *see* Stress interventions

G
Game theory-based reward-punishment strategy, 98
game elements, 103–104
SATBAC mechanism, 104–106
TNP unit, 106
trust rank, 104
Gene-document
classification task, 251–252
document classification problem, 249–251
gene sequences, 252–253
prediction accuracy, 253
rule mining, 256
rule pruning, 256–257
TASB-AC algorithm, 251–252
testing, 257
transactional dataset, 253–256
Gene Ontology (GO) terms, 255
Geographic Information Systems (GIS), 139
Global policy manager (GPM)
architecture, 120–121
dual mode, 120
policy configuration, 121–122
GotoBLAS linear algebra library function, 67
GPApriori, 80
GPM-initiated update, 121, 122
GPS coordinates
distance calculation using, 165
sanitization of, 167–168
testing proximity by, 165
GPU empowering machine learning
accelerated AdaBoost, 73–74
deep belief network, 74–75
deep Q-learning, 70–71

deep SARSA learning, 69–70
efficiency of matrix operations, 65
GPU-NB, 72–73
large scale linear regression, 76–77
large scale logistic regression, 78
massively parallel HMM, 79
optimized random forest, 68–69
parallel Apriori, 80–81
parallel decision tree, 66–67
parallel K-means, 71–72
parallel neural network, 67, 68
parallel SVM, 75–76
GPU-NB, 72–73
Guest VM (GVM), 240

H
Hardware device repository (HDR), 241
Hidden Markov model (HMM), 200
detection
Bitcoin, 275–276
botnets traffic, 273
CIs, 268
DGAs, 273–274
protocol obfuscation, 275
ROC curves, 268
smart grid traffic, 273–274
SSH session, 272
track shipping patterns, 271–272
Viterbi algorithm, 267–268
VPN, 272
directed transitions, 266
inference, 266–267
massively parallel HMM, 79
metric space, 270–271
model confidence, 268–269
model fidelity, 269
radiation detection and localization
gamma radiation, 276
ionizing radiation, 277
linear regression, 279–280
MLE, 278–279
radiation propagation model, 276–277
random processes, 266
Highly sensitive data objects (HSD), 215
High-performance networks, *see* Bandwidth reservation request (BRR)
HMM, *see* Hidden Markov model
Home area network (HAN), 56
Host-based IDS (HIDS), 199–201
Human-machine interface (HMI), 221
Human Protein Reference Database (HPRD), 255–256
Hybrid cloud, 114

I
IDMS, *see* Integrated distribution management system
Industrial control system (ICS), 220–221
Information-centric networking (ICN) protocol, 154
Information flow
data collector, 6, 7, 9
data miner, 6–9
data provider, 7, 8
decision maker, 6, 8, 9
security standards, 8–9
stages, 6, 10
Information-theoretic PIR (itPIR), 87
Integrated distribution management system (IDMS), 46, 48
Integrity management layer (IML)
basic computations, 211–212
data object, 211–213
data structure, 213
ERS schemes, 210, 214
implementation, 214
periodical encryption algorithm, 215
re-encryption stages, 212
storage requirement, 215–216
Intelligent electronic devices (IEDs), 46
Internet of Things (IoT), 100
application scenarios
in disaster management and response, 150–151
in healthcare applications, 149
in industries, 149–150
AR, 156
botnets
attacks, 133–134
history, 133
data interconnection and exchange, 131
device protection techniques, 134–135
domains and technical implications
cybersecurity and risk assessment, 154–156
data analytics, 152
networking and interoperability, 153–154
optimization algorithms, 153
product-service hybrid model, 151
seamless user integration, 151
self-aware intelligent applications, 152–153
sensor-driven computing, 151–152
EFFIFUEL service, 148
growth, 147–148
implementation models, 149
number of devices connected, 131, 132

Internet of Things (IoT) (*cont.*)
overlay networks and user experience
decision-making framework, 159, 160
inference engine, 159
Markov decision process, 158, 159
overlays formation, 159–160
performance and security, 158
QoE metrics, 157–158
QoS metrics, 157–158
redundant sensory devices, 156
risk assessment, 160
security preferences, 157
pay-as-you-use models, 148
predictions, 131
real-life physical devices, access to, 131
security vulnerabilities, 131–133
sensors, 125, 126, 128
sensory devices, 148–149
VR, 156
Intrusion detection systems (IDS), 221
anomaly-based detection, 198
CIDF, 198
DEC-POMDP
distributed system, 201–202
NLP-based solution, 203–204
observation probabilities, 202
HIDSs, 200–201
honeypots, 199–200
POMDP, 200, 202, 203
signature-based detection, 198
stateful protocol, 198
types, 199
IoT, *see* Internet of Things
IoT entity (IE)
modules (*see* IoT sensor networks)
smart grid layout, 46, 47
IoT sensor networks
LASN
applications, 57–58
architecture for AMI smart meters, 56–57
challenges and future research directions, 58–59
for customer, 46, 47
MASN
applications, 54
architecture for transactive energy, 52–54
challenges and future research directions, 54–55
subtransmission and distribution, 46
transactive energy, 47
WASN
architecture for synchrophasors, 49–50
challenges and future research directions, 51–52
generation and transmission, 46
layout, 48, 49
offline applications, 51
PDCs, 48–49
PMUs, 48
real-time applications, 51
WAMS, 47

J
Jacobi and Legendre symbols, 88

K
K-means algorithms, 71–72
Knowledge as a Service (KaaS) framework, 139
Kullback-Leibler divergence (KLD), 271

L
Large scale linear regression, 76–77
Large scale logistic regression, 78
LASN, *see* Local area sensor network
LBS, *see* Location-based services
Lightweight integrity management layer (IML), 211
Linear regression, 279–280
Local area network (LAN), 200, 201
Local area sensor network (LASN)
applications, 57–58
architecture for AMI smart meters, 56–57
challenges and future research directions
heterogeneous network, 58
interference, 59
interoperability, 59
QoS, 59
security, 58
topology design, 58
for customer, 46, 47
Local update, 121, 122
Location-based services (LBS), 163
applications categories, 164
contributions, 164–165
ECDHE private key (symmetric key) exchange, 166, 167, 169, 170
elliptic curve cryptography, 166
GPS coordinates, 165
k-anonymity, 164
location privacy, 164
PPPLP protocols, 164
proximity test, 166, 167

decimal precision agreement, 167
elliptic curve evaluation, 168–169
elliptic curve generation, 168
GPS coordinates, sanitization of, 167–168
security analysis, 170–171
Logistic (LO) classifier, 39
Lyapunov exponent methods, 15, 23, 24

M
Machine learning
analytic applications
computer vision, 82
driverless cars, 81
E-commerce, 82
modern Facebook faces recognition with deep learning, 81
natural language processing, 82
precision medicine, 82
smartphones, 81
Twitter hashtag prediction, 81
Twitter users latent attributes, 81
GPU empowering (*see* GPU empowering machine learning)
multimedia data analysis, 140
taxonomy, 64, 65
Markov decision process (MDP), 158, 159, 200
Markov model, 265–266
MASN, *see* Medium area sensor network
Massively parallel HMM, 79
Master terminal unit (MTU), 221
Maximum likelihood estimation (MLE), 278–280
MDM, *see* Meter data management
MDP, *see* Markov decision process
Medium area sensor network (MASN)
applications, 54
architecture for transactive energy, 52–54
challenges and future research directions, 54–55
subtransmission and distribution, 46
transactive energy, 47
Medium sensitive data objects (MSD), 215
Merkle root, 209–210
Meter data management (MDM), 46, 57
Modbus
application layer, 226, 228
data link layer, 226, 228
pseudo-transport layer, 226, 228
security attacks, 226
TCP, 226–227
Modified Apriori algorithm, 248
Multi-class SVM, 75
Multilayer perceptron (MP), 39
Multimedia data, disaster management
analysis
data mining, 140
feature analysis, 141
machine learning, 140–142
post-processing, 142
preprocessing, 141
tools and applications, 142, 143
data management framework, 138
future directions, 144–145
geographic data, 139–140
sensors data, 139
social media data, 140
storing and sharing data, 140
3D Storm Surge Impact Animation
GIS data simulation, 142, 143
I-CAVE, 143
LiDAR, 143
storm surge scenarios, 144
visual representation, South Miami Beach., 143
visual data, 139
Multistage classification problem, 248–249

N
Naïve Bayes (NB) classifier, 39
Nearest neighbor (IB1), 39
Neighborhood area network (NAN), 56–57
Net metering, 57
Network-based IDS (NIDS), 199
Neural network, 39
NLP, 203–204
Non-fiducial points, 36
Normal threshold time (NT), 229

O
Oblivious transfer (OT) schemes, 88
Open Compute Language (OpenCL), 79, 80
Operational entity (OE), 46, 47
Optimized random forest, 68–69
Outage management system (OMS), 46
Overlay networks, IoT
decision-making framework, 159, 160
inference engine, 159
Markov decision process, 158, 159
overlays formation, 159–160
performance and security, 158
QoE metrics, 157–158
QoS metrics, 157–158
redundant sensory devices, 156

Overlay networks, IoT (*cont.*)
risk assessment, 160
security preferences, 157

P
Parallel Apriori, 80–81
Parallel decision tree, 66–67
Parallel K-means, 71–72
Parallel neural network, 67, 68
Parallel SVM, 75–76
Partially observable Markov decision process (POMDP), 200, 202, 203
Patient model, 288–290
Peer-initiated update, 121, 122
People-Discovery (PD), 164
Perfect privacy
information-theoretic single-database PIR scheme, 88
PIR query, 89
PIR scheme
bit group encoding, 91
connecting two encoding functions, 91
generic l-bit perfect privacy PIR scheme, 92–93
l-bit input *vs.* l-output property combination, 90–91
public key combinations, 89, 90
public key cryptography, 89–90
random variables, 88
single database PIR, 89
solution, 86–87
Periodical re-encryption-based archival systems
IML
basic computations, 211–212
data object, 211–213
data structure, 213
ERS schemes, 210, 214
implementation, 214
periodical encryption algorithm, 215
re-encryption stages, 212
storage requirement, 215–216
integrity checking and proof of existence
cryptographic hash function, 208–209
digital signature scheme, 209
Merkel tree, 209–210
one-way accumulator, 210
Patricia tree, 210–211
TSA, 210
Phasor data concentrators (PDCs), 46
Phasor measurement units (PMUs), 46, 273–274
PIR, *see* Private information retrieval
Poincaré map, 15, 23
Point-of-Interest based (PoI), 164
Poisson random variables, 277
Policy injection interface, 120
Privacy-enabled retrieval techniques, 85
Privacy-preserving location-proximity (PPLP) protocols, 164
Privacy sensitive location-based services, *see* Location-based services (LBS)
Private block retrieval (PBR), 86
Private cloud, 114
See also Dynamic firewall policy management
Private information retrieval (PIR)
client-server privacy critical applications, 87
cPIR, 87–88
itPIR, 87
notations and preliminaries, 88–89
organization, 88
peer-to-peer privacy critical applications, 87
perfect privacy (*see* Perfect privacy)
scenarios, 86
PRoPHET, 174
Protein interaction (PPI) features, 249
Protein–Protein Interaction (PPI), 255–256
Protocol-driven privacy architecture, 86
Public cloud, 114
Public key cryptography, 89–90
Public key cryptosystem, 166

Q
QoS, *see* Quality of Service
Quadratic residuosity, 87, 88
Quadratic residuosity assumption (QRA), 89
Quality of Experience (QoE) metrics, 157–158
Quality of Service (QoS)
communication, 52
metrics, 157–158
parallel K-means, 71

R
Random forest (RF), 39
Random guessing, 87
Ransomware, 225
Receiver operating characteristic (ROC) curves, 39–42, 268
Reinforcement learning 4J (RL4J), 70
Remote terminal unit (RTU), 222
Renewable energy sources (RESs), 46
Residential energy management, 58

Retransmission threshold time (RT), 230
Risk assessment, 154–156, 160
Round-trip times (RTTs), 25–26
  concave and convex regions, TCP, 14, 15
  CUBIC throughput measurements, 22
  Poincaré map of UDT, 23–24
  throughput box plots with variable RTTs, 19, 20
  throughput with variable RTTs, 18, 19

S
Seamless user integration, 151
Self-adaptable trust-based access control (SATBAC) framework, 98
  ARTVC unit (*see* Access Request Trust Value Calculation Unit)
  e-commerce online shopping example, 106–108
  proposed framework, 100, 101
  reward-punishment strategy (*see* Game theory-based reward-punishment strategy)
Sensor-driven computing, 151–152
Sensors
  additional components, 127
  copper-based temperature sensor, 125
  CPS, 126, 130
  energy forms, 126, 127
  fundamental components, 126–127
  IoT (*see* Internet of things)
  process automation, 129
  smart objects, 126, 129–130
  smart sensors, 128–129
  working, 126
  WSN, 127–128
Sequential minimal optimization (SMO)
  algorithms, 75
  stress classifiers, 39
Service-level agreement (SLA), 238–239
Single instruction multiple data (SIMD), 67
Smart grid
  classification, 46
  distributed sensor networks, 45
  EIS, 46, 47
  entities
    FE, 16, 47
    IE (*see* IoT sensor networks)
    layout, 46, 47
    OE, 46, 47
  IEDs, 46
  prosumers, 46
  real-time situation awareness, 46
Smart meter, 56
Smart sensors, 128–129
SMO, *see* Sequential minimal optimization
Software-defined networking (SDN) protocol, 154
SPRINT, 66
SSH session, 272
Stack-based IDS (SIDS), 199
Stress classification
  chronic job stress, 36
  distress, 36
  ECG signals
    classifiers, 39
    extracted features, 39
    fiducial points, 36
    instantaneous heart rate, 37
    multisensor stress analysis, 36
    non-fiducial points, 36
    and QRS wave complex, 38
    ROC area and ACC, 39–42
    R-R interval, 39
    significance of work, 37
  eustress, 36
  principal component analysis, 36
Stress interventions
  continuous signal, 288
  control theory, 286
  disadvantage, 286
  EMA, 287
  EMI, 287
  health institutions, 286–287
  patient model, 288–290
  patient's resilience/ability
    long-term stress, 290–291
    medium-term stress, 291–292
    short-term stress, 292
    total stress level, 292–293
    stress episode, 287
Stuxnet, 224, 225
SUBMTUs, 221
Supervisory control and data acquisition (SCADA) systems
  architecture of, 221–222
  attacker objectives, 223, 226
  behavior of, 222
  command and control systems, 223–225
  critical infrastructures, 220
  DNP3
    application layer, 226, 228
    data link layer, 226, 228
    pseudo-transport layer, 226, 228
    Wireshark tool, 227–232
  ICS, 220–221
  TCP, 226–227

Support vector machine (SVM)
binary, 75
multi-class, 75
parallel, 75–76
SUVnet-Trace Data, 175
Symbolic transfer functions (STFs), 271–272
Synchrophasors, 49–50
See also Wide area sensor network

T
TCP, 226–227
concave and convex regions, 14, 15, 18
monotonicity, 18
qualitative properties, 17
scalable TCP, 14, 15
throughput measurement profiles, 18–20
Temporal analysis, *see* Electrocardiogram (ECG) signals
Throughput profiles
definition, 14
dynamics
CUBIC throughput traces with large buffers, 22
Lyapunov exponents, 23, 24
Poincaré map, 23
RTT cluster, 24
sample send rate and NACK traces, 22
transfer rates, 21
TCP and UDT
concave and convex regions, 14, 15, 18
monotonicity, 18
qualitative properties, 17
scalable TCP, 14, 15
throughput measurement profiles, 18–20
throughput model, 20–21
time traces, 15
wide-area Lustre, 29, 30
XDD and GridFTP with direct I/O, 28
TNP, *see* Trust negotiation policy
Transactive energy (TE), 52–54
See also Medium area sensor network
Transducer, 126
Trapdoor function, 88–89
Trust attribute model, 99–100
Trust-based access control, 99
Trusted third party (TTP), 57
Trusted time stamping authority (TSA), 210
Trust management, 98, 99
Trust negotiation policy (TNP), 98, 106
TrustRank, 104, 106, 109

U
UDP-based Data Transfer Protocol (UDT)
concave and convex regions, 14, 15, 18
monotonicity, 18
qualitative properties, 17
scalable TCP, 14, 15
throughput measurement profiles, 18–20
Unique identifier (UID), 128
Unmanned aerial vehicles (UAVs), 139

V
VANETs, *see* Vehicular network
Variable path with fixed bandwidth (VPFB), 187
Variable path with variable bandwidth (VPVB), 187
Vehicle to vehicle communications, *see* Vehicular network
Vehicular network (VANETs)
contact duration
characteristics, 175
contributions, 174
cooperative caching protocol, 175
methodology, 177
PRoPHET, 174
routing protocol, 174
Shanghai dataset, 175–177
statistics (*see* Encounter duration statistics)
contact process, 174
mobile nodes, communication links, 173
Very-large-scale integration (VLSI), 128
Virtual machine (VM)
cloud computing threats, 238–239
hosted virtualization, 236
security threats
architectural issues, 236
creation process, 237
cross-VM side-channel attack, 237, 242–243
escapes, 237
hypervisor, 237
image splitting, 237
isolation, 237
legal issues, 236–237
migration, 237–238
network issues, 236
rollback, 237
scheduler, 237
sprawl, 237
virtualization and migration
architecture, 241–242
components/modules, 241

EVDIC, 239–240
GVM, 240
Mirage, 239
SVM, 240
Virtual Machine Integrity Monitor (VIM), 241, 243
Virtual machine monitor (VMM), 241
Virtual Machine Resource Manager (VRM), 241–242
Virtual reality (VR), 156
Visualization, 138, 142–144
Viterbi algorithm, 267–268
VNSS, 240
Volunteered geographic information (VGI), 140
VPN, 272

W
WaitingTime (WT), 229
WASN, *see* Wide area sensor network
Weighting scheme, 248
Wide area measurement system (WAMS), 47
Wide area sensor network (WASN)
architecture for synchrophasors, 49–50
challenges and future research directions
analytics, 51
communication, 52
cybersecurity, 52
data quality, 51
placement, 51
generation and transmission, 46
layout, 48, 49
offline applications, 51
PDCs, 48–49
PMUs, 48
real-time applications, 51
WAMS, 47
Wireless sensor and actuator network (WSAN), 128
Wireless sensor networks (WSNs), 127–128, 139
Working set selection (WSS) logic, 75