Research Statement

Yuzhe Tang, (www.cc.gatech.edu/~ytang36)

I. Introduction

My research is centered on **secure big data management in cloud**. In the last few years, big data has successfully penetrated our daily life in various domains, ranging from social networking, mobile computing, to electronic health-care applications. As big data continues to grow, the processing of it exceeds the average capability of a normal entity, be it a start-up company or an average mobile phone user, and the use of cloud computing has become a popular trend. In cloud computing, the processing of big data is delegated to a powerful platform in a data center, that is, the cloud, which utilizes consolidated resources for data computation with low costs. The new computing paradigm with cloud has recently gained huge popularity in a variety of domains, ranging from people's daily life like storing mobile user data in iCloud, to enterprise supply chains where big cloud service providers (e.g. IBM and Amazon) process outsourced data for small startup companies. With the advent of the new computing paradigm, it poses new challenges and presents new opportunities to design a computer system for cloud computing that meets different application needs.

My research methodology contains a variety of strategies used in different research stages including selective reading, critical thinking, strategic planning and execution. My research usually starts with finding potential research problems through sifting through news, technical blogs and vision papers, and with consultation from peer researchers. It is critical to work on a problem with research value, and I tend to favor the problem that can be modeled mathematically or algorithmically and is challenging to solve (e.g. with conflicting design goals). My methodology for searching for a solution is usually an iterative process, which starts with a naive solution and then iteratively refines it by discovery of new problems of the current solution and solving it. I value the system building and realistic implementation of my research; most of my research is substantiated by a functioning prototype system and realistic experiment results.

II. Dissertation and previous research

My dissertation research addresses the challenges of designing a secure and high performance cloud system for big data serving and processing. On one hand, being able to offer a low price for big data processing is one of the reasons that cloud computing can succeed today. As the size of big data continues to grow, a cost-effective method to utilize computing resources in cloud is a key factor to maintain the low processing expenses. Thus, the first part of my research aims at optimizing the performance of cloud systems under the given hardware constraints. On the other hand, security has become a vital factor to the sustained success of cloud computing with big data. Many types of the big data, be it personal private data (e.g. mobile presence data, digitalized medical records) or corporate confidential information (e.g. sales information), are sensitive in nature. Delegating big data computation from an owner to a third-party cloud tests the owner's trust in cloud which, if handled improperly, could develop into a reluctance of using cloud computing at all. The trust problem compounds in the presence of numerous incidents that big cloud companies are caught of misbehavior. Consider the recently exposed PRISM program, a massive surveillance program led by the US government that inspects client's personal information in big data companies (e.g. Salesforce) and breaches user privacy with a lot of controversies raised on the public trust. As a result, the second part of my research aims at designing a secure and privacy-preserving cloud system that protects data privacy and integrity in the cloud.

A. Performance optimizations of scalable cloud serving system

One important reason contributing to the success of cloud computing is its cost-effective approach to process big data. The ability to squeeze performance from a number of low-end machines is critical for the promise of cloud computing, that is, the high performance data processing with low costs. In this thread of research, I addressed the performance optimization problem on various scalable cloud systems, including key-value stores (and DHT networks in peer-to-peer networks as by my previous research) and scaling up to multi-core machines.

Indexing write-optimized key-value stores: Key-value stores are emerging big data storage and serving systems that attract an increasing amount of attention. Many write-optimized key-value stores, such as HBase and Cassandra, have successfully managed write-intensive workloads for Web 2.0 applications. While most existing key-value stores provide key-based access methods, the value-based access methods are rarely supported due to the challenges in indexing big data at scale. In this project, I addressed the data indexing problem in key-value stores, and proposed HIndexer, a generic index framework adaptable to various key-value stores with optimized data write performance. HIndexer achieves instant index updates in real time, yet with very lightweight maintenance overhead. My approach is to use a performance optimization technique that is designed aware of the unique characteristic of write-optimized key-value stores. Concretely, I discovered that the inefficiency of using conventional indexing approaches on write-optimized workloads comes from an expensive indexing operation, named

index repair. My performance optimization technique [1] defers the index repair operations to a later time, either during an online period when the system is serving the other operations, or in an offline stage when the system is under low workload. The online design of HINDEXER aims at making index repair as lightweight as possible in order to avoid intruding the performance of concurrent online operations, while the offline design optimizes the throughput of batch processing by a tight coupling of the repair operations with a store reorganization operation called compaction. HINDEXER is an adaptive indexing framework that adapts the decision making of deferred index repairs to the current system workload. The proposed HINDEXER technique has been integrated into the IBM BigInsights product [2].

Scaling up stream processing with multi-cores by automatic pipelining: Stream processing, critical to many big data applications such as real-time analytics, is known to be computationally intensive. To fully utilize the multi-core resources, it is desirable to use the system optimization techniques. In this project, I studied the problem of automatic performance optimization of streaming applications by using pipelining parallelism [3]. Through modeling the system CPU bottleneck, I formulated that the design goal is to prevent the single-core bottleneck from happening, in which one or few saturated cores may block the whole system. I proposed a program-to-thread mapping scheme that can dynamically adjust itself based on the changing workloads and automatically avoids the single-core bottleneck. I implemented the proposed scheme in the IBM System S which is an industrial strength big stream processing system. The implementation is based on a lightweight CPU profiler and optimizer under an adaptive control framework.

B. Security and privacy in multi-domain cloud systems

In the age of big data, the digital personal data has prevailed in many domains, including social networks, electronic health, and mobile computing. The processing of big personal data, which is made possible by cloud computing, causes a lot of security and privacy issues. It is challenging to address the security in the face of cloud computing, because not only it is necessary to guarantee the data security under various attacks, but also the computation can be carried out on the cloud data under protection. While the recently proposed fully homomorphic encryption provides a potential method, it is far from being practical due to the extremely high computation overhead. In this thread, my research addressed cloud security problem and approached the solution in a domain-specific fashion. My research addressed the data authenticity in a cloud outsourcing scenario for multi-version key-value stores, and the privacy preservation of possession of personal records in the emerging health-care provider networks.

Freshness authentication in outsourced multi-version key-value stores: This project addresses the data authenticity of an outsourced key-value storage in the cloud. In an outsourced database, the data owner publishes the local data updates to the cloud, which is responsible for serving the big data to a large customer base. The outsourced data and updates are signed to protect the data authenticity. In this work, I focused on the multi-version key-value stores and addressed the authentication of version freshness. That is, it is guaranteed that a latest version of an object is returned from the cloud storage. I proposed an authentication framework [4] to sign the data updates in a streaming fashion and verify the data freshness. To optimize the performance of data signing under an intensive data stream, I further proposed INCBM TREE [5], an authentication structure based on a hierarchy of Bloom Filters digested by a Merkle tree. For efficient verification, the INCBM TREE can effectively summarize the data update stream; yet, it requires a very small memory footprint which lends itself to the owner of limited resources.

Privacy preserving index and search in multi-domain cloud: In the age of cloud computing, losing data control continues to be a major concern and the recent move toward giving data control back to cloud users has given birth to a variety of multi-domain cloud systems for different applications such as distributed social networking, peer-to-peer file sharing and electronic Healthcare It is crucial to support privacy preserving index (or PPI) in the multi-domain cloud for the sake of effective information exchange and sharing between domains. In this project, I proposed a number of PPI frameworks addressing different system designs. First, for efficient secure index construction, I proposed ssPPI [6], based on the novel use of secret sharing in a parallel and distributed computing framework. Second, for effective privacy preservation, I proposed ϵ-PPI [7] that differentiates the protection on different indexed terms with quantitatively controllable privacy. Third, to address the privacy protection under multi-keyword document search, I proposed MPPI [8]. The key challenge in realizing the proposed PPI frameworks comes from the needs for secure index construction in a mutually untrusted network. While the norm to use multi-party computations (or MPC) for secure computation is very expensive when it comes to big data in large network, I proposed several MPC optimization techniques to realize the massive computation in the PPI construction on tens or hundreds of cloud domains.

C. Non-dissertation work

During my Ph.D. program, I have conducted collaborative research on a variety of topics. On privacy and security, we addressed the location privacy on road network by proposing a mix-zone based technique [9]. We further extend the proposed mix-zone technique to the case of continuous queries and delay-tolerant vehicle networks [10]. On the system research, we addressed the problematic design of existing virtualization software which is made agnostic of physical machine locality. We proposed a cross-layer optimization technique [11] to take advantage of the physical-level locality and improve the

performance of inter-virtual machine communications. To process big RDF data, we proposed to partition graph data into a number of networked machines in the cloud with efficiency and flexibility [12]. As performance monitoring in cloud is critical to performance diagnosis and optimization, we proposed a dependable monitoring technique that is resilient to network link failures in large data center networks [13].

Indexing DHT networks with low maintenance overhead: I have previously worked on a research topic on scalable data indexing in peer-to-peer networks. The project aims at providing a generic index on top of distributed hash tables (or DHT), which is a representative peer-to-peer network. Due to the high dynamism in peer-to-peer networks, where peers can freely join and leave the network, it calls for a lightweight index maintenance scheme. I proposed LIGHT, an optimized indexing framework on top of generic DHT networks. The design of LIGHT challenges two seemingly conflicting goals, that is, maximizing query efficiency while minimizing the maintenance overhead. To meet the challenge, I proposed a novel index-to-peer mapping scheme that intelligently minimizes inter-peer communications for both index maintenance and query processing. I applied this idea to different indexing and search scenarios with the devised algorithms, including value-based range query [14], [15], multi-dimensional range queries [16] and k-NN queries [17].

III. Ongoing and Future Research Interests

A. Secure and privacy-protected cloud services for multi-source data analytics

Privacy and security issues continue to be a major concern of cloud users. The cloud service providers, while being able to store, process and even serve sensitive data for their clients, are responsible for protecting data confidentiality and integrity, as required by laws or based on their promises such as terms of services. From another perspective, with a large volume of client data at hand, cloud providers may want to mine valuable information and insights by analyzing multiple datasets across the cloud boundary. To make such multi-source data analytics possible, it is critical to carry out analytics in a privacy preserving manner: Failing in protecting data privacy causes social or even legal issues. Concretely, I would like to investigate the following two scenarios:

When MPC meets big data in cross-domain private data mining: This future project considers multiple private clouds or data providers under different administrative domains who want to collaboratively analyze their data for extracting valuable information insights. To protect information privacy and confidentiality, a straightforward approach is to use the traditional MPC (i.e. multi-party computation) technique that preserves the input data privacy. However, the problem is that generic MPC techniques only scale to a very small network and a very simple computation. In this project, I look forward to utilizing the performance optimization technique at both system and computation levels, to make the MPC-based private big data analytics possible in practice.

Privacy-aware data storage in hybrid cloud: Hybrid cloud is an increasingly popular pattern to process private big data. In a hybrid cloud, an entity manages some resources in an in-house private cloud, while having others provided by an external public cloud. In such systems, it is critical to control the data flow between the private and public domains in order to ensure information confidentiality. In this future project, I would like to explore a domain-aware design of key-value stores that can be deployed on top of hybrid clouds transparently without manual management on the inter-domain data flow. Assuming a public-key infrastructure is used to guarantee the data security, the key challenges are to automatically enforce encryption/decryption on inward and outward data-flow, and to minimize the incurred performance overhead.

B. Scalable systems for big data management in cloud

Supporting big data management and scalable SQL has recently attracted a great deal of attentions in industry and is pioneered by many projects in big companies including Google F1, IBM BigInsights and so on. In the research community, while shared-nothing databases has been researched, it is the sheer volume of big data, the large scale of cloud systems and the unique requirement of modern Web 2.0 applications that make the problem distinct. Given the recent popularity of NoSQL stores and sustained success of relational databases, it presents new system design opportunities and challenges to get the best from the both worlds; while the SQL based relational databases offers strong consistency and optimized query performance, the NoSQL stores excel in schema-less flexibility and scalability. Various interesting design problems can be studied in this new field, including but not limited to the selective query support by adaptive designs of auxiliary structure (e.g. index and materialized views), the analysis of service level agreement and application characteristics for intelligent mapping of applications to correct data infrastructures. In this area, my research plan is to design and develop a middleware layer between storage servers and application servers that extends the functionality of existing open-source NoSQL stores to support interactive analytical and transactional workloads. In this regards, my HINDEXER project is the first step towards Big SQL in cloud with the focus on system designs. I would continue this effort but expand my scope on the big SQL support in cloud. My short-term goal is to work on the following two projects.

Real-time analytics of big data on scalable NoSQL systems: For supporting real-time analytical workloads, materialized views are a promising technique. In filling the gap between what traditional materialized views can support and what is demanded by real-time analytics, the key challenge is on minimizing the overhead of dynamic view maintenance. In this project, I aim at finding a low-maintenance scalable view, and plan to study a series of interesting design problems, such as

partial view design and co-location of view with base data. My previous research experiences on distributed materialized views for top-k aggregation [18] prepared me with necessary insights and know-how on this future project.

Performance optimization of NoSQL systems by on-the-fly compactions: This ongoing project addresses the performance optimization problem of widely used log-structured stores including HBase and Cassandra. The research is motivated by the observed mismatch of the increasing write rate of real cloud workloads and the maximally sustained write rate of log-structured stores. The root cause of the mismatch is the cumbersome reorganization process in NoSQL stores (e.g. compaction in HBase). The project currently proposes a store reorganization mechanism that can be executed on the fly. For lightweight execution with other concurrent online operation, the performance is carefully optimized by making use of idling resources of the computer system.

C. Looking beyond

In the long term, my research goal is to empower average users with secure, efficient and rich access to the big data both inside a cloud and across the cloud boundary. As big data and cloud computing expand into more domains, from bio-informatics, to high-performance computing, to wearable computing, many interesting problems are expected to emerge. In this exciting new age of computing, I would like to embrace these new opportunities and challenges. For the new problems, I would like to challenge them from the perspective of secure and efficient big data management, on which I honed my skills. By studying the trending workload characteristic, modeling modern cloud systems, formulating and solving research problems by mathematical analysis, algorithm design and system prototyping, I am confident that there are many research opportunities in this domain and I can make the impossible possible.

I am a strong believer in team philosophy and collaborative research. I look forward to working with other researchers and creating synergy. My wide range of knowledge and research skills (from distributed system, to information retrieval, to databases) have prepared me for collaborative research at various levels and across different disciplines, both within and beyond computer science. I have collaboration experiences on computer science research on mobile networks [9], virtualization [11] and graph processing [12], and on inter-disciplinary research on Bio-informatics and financial data processing. During these collaborations, I have found that the interaction between researchers of different backgrounds is very informative and highly effective. I am very positive about and looking forward to working with peer researchers on a variety of research topics.

REFERENCES

[1] Yuzhe Tang, Arun Iyengar, Wei Tan, Ling Liu, and Liana Fong. Write-optimized indexing of log-structured key-value stores. In *submission*, 2014.

[2] Wei Tan, Tata, Yuzhe Tang, and Liana Fong. Diff-index: Differentiated index in distributed log-structured data stores. In *EDBT*, 2014.

[3] Yuzhe Tang and Bugra Gedik. Autopipelining for data stream processing. *IEEE Trans. Parallel Distrib. Syst.*, 24(12):2344–2354, 2013.

[4] Yuzhe Tang, Ting Wang, Xin Hu, Jiyong Jang, Ling Liu, and Peter Pietzuch. Authentication of freshness for outsourced multi-version key-value stores. In *submission*, 2014.

[5] Yuzhe Tang, Ting Wang, Xin Hu, Reiner Sailer, Ling Liu, and Peter Pietzuch. Outsourcing key-value stores with verifiable data freshness. In *ICDE*, 2014.

[6] Yuzhe Tang, Ting Wang, Ling Liu, Shicong Meng, and Balaji Palanisamy. Privacy preserving indexing for ehealth information networks. In *CIKM*, pages 905–914, 2011.

[7] Yuzhe Tang, Ling Liu, and Arun Iyengar. e-ppi: Searching information networks with quantitative privacy guarantee. In *submission*, 2014.

[8] Yuzhe Tang and Ling Liu. Multi-keyword privacy-preserving search in distributed personal content networks. In *submission*, 2014.

[9] Balaji Palanisamy, Ling Liu, Kisung Lee, Aameek Singh, and Yuzhe Tang. Location privacy with road network mix-zones. In *MSN*, pages 124–131, 2012.

[10] Balaji Palanisamy, Ling Liu, Kisung Lee, Shicong Meng, Yuzhe Tang, and Yang Zhou. Anonymizing continuous queries with delay-tolerant mix-zones over road networks. *Distributed and Parallel Databases*, pages 1–28, 2013.

[11] Qi Zhang, Ling Liu, Yi Ren, Kisung Lee, Yuzhe Tang, Xu Zhao, and Yang Zhou. Residency aware inter-vm communication in virtualized cloud: Performance measurement and analysis. In *IEEE CLOUD*, pages 204–211, 2013.

[12] Kisung Lee, Ling Liu, Yuzhe Tang, Qi Zhang, and Yang Zhou. Efficient and customizable data partitioning framework for distributed big rdf data processing in the cloud. In *IEEE CLOUD*, pages 327–334, 2013.

[13] Shicong Meng, Arun Iyengar, Isabelle Rouvellou, Ling Liu, Kisung Lee, Balaji Palanisamy, and Yuzhe Tang. Reliable state monitoring in cloud datacenters. In *IEEE CLOUD*, pages 951–958, 2012.

[14] Yuzhe Tang and Shuigeng Zhou. Lht: A low-maintenance indexing scheme over dhts. In *ICDCS*, pages 141–151, 2008.

[15] Yuzhe Tang, Shuigeng Zhou, and Jianliang Xu. Light: A query-efficient yet low-maintenance indexing scheme over dhts. *IEEE Trans. Knowl. Data Eng.*, 22(1):59–75, 2010.

[16] Yuzhe Tang, Jianliang Xu, Shuigeng Zhou, and Wang-Chien Lee. m-light: Indexing multi-dimensional data over dhts. In *ICDCS*, pages 191–198, 2009.

[17] Yuzhe Tang, Jianliang Xu, Shuigeng Zhou, Wang-Chien Lee, Dingxiong Deng, and Yue Wang. A lightweight multidimensional index for complex queries over dhts. *IEEE Trans. Parallel Distrib. Syst.*, 22(12):2046–2054, 2011.

[18] Yuzhe Tang, Junichi Tatemura, Ling Liu, and Hakan Hacigumus. Ktv-tree: Real-time top-k analytics by dynamic view materialization in cloud. In *Tech. Report*, 2010.