



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

Anand M¹, Anuja S B²

¹Student, Department of MCA, Narayanaguru College of Engineering, Anna University

²Asst.Professor, Department of MCA, Narayanaguru College of Engineering, Anna University.

ABSTRACT

Benefited from its reliability and convenience, biometric identification has become one of the most popular authentication technologies. Due to the sensitivity of biometric data, various privacy-preserving biometric identification protocols have been proposed. However, the low computational efficiency or the security vulnerabilities of these protocols limit their wide deployment in practice. To further improve the efficiency and enhance the security, in this project, propose two new privacy-preserving biometric identification outsourcing protocols. One mainly utilizes the efficient Householder transformation and permutation technique to realize the high-efficiency intention under the known candidate attack model. The other initializes a novel random split technique and combines it with the invertible linear transformation to achieve a higher security requirement under the known-plaintext attack model. Also, we argue the security of our proposed two protocols with a strict theoretical analysis and, by comparing them with the prior existing works, comprehensively evaluate their efficiency.

1 Introduction

Data processing is a rapidly growing part of business and industry. These days, big data define databases so large and complicated that traditional data management tools or processing techniques have got some limitations to handle. The amount of data from various sources, for example, the Internet of things (IoT), scientific research, and social networking websites (Twitter, Facebook, Instagram, etc.) is increasing sharply. On the other hand, the latest advances in information technology make it easier to generate data and as a result, an enormous amount of data is generated every day.

Hence, big data analysis is a needed service in today's Internet companies. For example, Google processes data over one hundred Petabytes (PB), Facebook generates log data of approximately 10 PB per month, and Baidu, a Chinese company, processes data of tens of PB [1]. To process this massive data efficiently, Map Reduce [2] with an open-source implementation called Hadoop as the main solution. It is the common computing framework used by Yahoo!, Amazon, Facebook, etc.

Nonetheless, it is important to note that the big data frameworks wouldn't be possible to work without supporting the network underlying due to their extremely large volume of data and computing complexity [3]. To this end, some performance-oriented policies must be



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

implemented in the network. To express the desired high-level network policies, including traffic engineering, QoS, and load balancing, network operators need to configure each individual network device separately using low level and often commands that are vendor-specific.

To overcome these challenges, software-defined networking (SDN) architecture is proposed. As a new networking paradigm, SDN enables us to improve the performance of the network. In this architecture, the control logic of the network is separated from the devices such as routers and switches forwarding the traffic. Through this separating, the network switches play a forwarding role only and the controlling actions such as applying policies and managing the flows are taken in a controller by (re)configuring the network [4]. In Hadoop frameworks, one of the most challenging issues that impact the performance of dataprocessing is the “minimum makespan” problem. Although many types of research have been done on task scheduling, they can only find a near optimal solution or have high complexity in their computation due to the limitation of traditional networks to provide a global view of the network. To resolve this NP-complete problem, we have to consider the distance of servers from where data is located. Although to schedule the big data tasks.

We prefer to assign the tasks to the local servers, by selecting the local servers frequently, these servers may be overloaded, and the waiting time for tasks to execute increases. On the other hand, in assigning tasks to the remoteservers, the corresponding bandwidth and time are needed for moving data. Therefore, we are going to make a tradeoff between choosing the remote and local servers to assign the tasks and finding solutions to shorten the job completion time. To address the mentioned problem, we leverage the abstraction of SDN which can providethe network state information to monitor and manage the networks and make the related decisions accurately. Inorder to reach a solution in this search space, we employ the tabu search algorithm which provides solutions veryclose to optimal, if not the best. The proposed algorithm not only shortens the completion time of task scheduling in compared the related works by considering the state of the network such as congestion ratio, but it improves the data locality which is vital for performance because the network bisection bandwidth in a large cluster is much lower than the aggregate bandwidth of the disks in the machines.

2 Recent Works

[1] Big Data Meets Software-Defined Networking: SDN for Big Data and BigData for SDN Laizhong Cui-2019. Both big data and software-defined networking (SDN) have attracted great interests from both academia and industry. These two important areas have traditionally been addressed separately in the most of previous works. However, on the one hand, the good features of SDN can greatly facilitate big data acquisition, transmission, storage, and processing. On the other hand, big data will have profound impacts on the design and operation of SDN. In this paper, we present the good features of SDN in solving several issues prevailing with big data applications, including big data processing in cloud data centers, data



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

delivery, joint optimization, scientific big data architectures and scheduling issues. We show that SDN can manage the network efficiently for improving the performance of big data applications. In addition, we show that big data can benefit SDN as well, including traffic engineering, cross-layer design, defeating security attacks, and SDN-based intra and inter data center networks. Moreover, we discuss a number of open issues that need to be addressed to jointly consider big data and SDN in future research.

[2] Toward an SDN-Enabled Big Data Platform for Social TV Analytics HanHu-2020.

The TV experience is being transformed with online social networks. TV audiences are sharing their opinions (i.e., social response) about video programs on OSNs (e.g., Twitter and SinaWeibo), thus providing a great opportunity for mining these data for stakeholders in TV value chains. This new paradigm is touted as social TV analytics, integrating the emerging big data research into TV. In this article, we envision and develop a unified big data platform for social TV analytics, extracting valuable insights from TV social response in a real-time manner. Such a platform presents tremendous challenges in networking architecture for our big data platform. We propose to build a cloud-centric platform with SDN support, providing on-demand virtual machines and reconfigurable networks. The architecture of our system consists of three key components, including a robust data crawler system, an SDN-enabled big data processing system, and a social media analytics system. The data crawler system adopts a distributed architecture to circumvent the access constraints of OSNs to crawl sufficient data about each TV program of interest; the SDN-enabled big data processing system integrates SDN and Hadoop, and exploits the SDN benefit to transfer intermediate data between different processing units to accelerate the data processing rate; and the social media analytics system extracts the public perception and knowledge related to TV programs based on microblog data. We have built a proof-of-concept demo over a private cloud at Nanyang Technological University. Feature verification and performance comparisons demonstrate the feasibility and effectiveness of the system.

[3] Big Data and Cloud Computing: Innovation Opportunities and Challenges Chaowei Yang-2019.

Big Data has emerged in the past few years as a new paradigm providing abundant data and opportunities to improve and/or enable research and decision-support applications with unprecedented value for digital earth applications including business, sciences and engineering. At the same time, Big Data presents challenges for digital earth to store, transport, process, mine and serve the data. Cloud computing provides fundamental support to address the challenges with shared computing resources including computing, storage, networking and analytical software; the application of these resources has fostered impressive Big Data advancements. This paper surveys the two frontiers – Big Data and cloud computing – and reviews the advantages and consequences of utilizing cloud computing to tackling Big Data in the digital earth and relevant science domains. From the aspects of a general introduction, sources, challenges, technology status and research opportunities, the following observations are offered:



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

- (i) Cloud computing and Big Data enable science discoveries and application developments
- (ii) Cloud computing provides major solutions for Big Data
- (iii) Big Data, spatiotemporal thinking and various application domains drive the advancement of cloud computing and relevant technologies with new requirements
- (iv) Intrinsic spatiotemporal principles of Big Data and geospatial sciences provide the source for finding technical and theoretical solutions to optimize cloud computing and processing Big Data
- (v) Open availability of Big Data and processing capability pose social challenges of geospatial significance
- (vi) A weave of innovations is transforming Big Data into geospatial research, engineering and business values. This review introduces future innovations and a research agenda for cloud computing supporting the transformation of the volume, velocity, variety and veracity into values of Big Data for local to global digital earth science and applications.

[4] Big Data Analysis-Based Secure Cluster Management for Optimized Control Plane in Software-Defined Networks Jun Wu-2021. In software-defined networks (SDN), the abstracted control plane is its symbolic characteristic, whose core component is the software-based controller. The control plane is logically centralized, but the controllers can be physically distributed and composed of multiple nodes. To meet the service management requirements of large-scale network scenarios, the control plane is usually implemented in the form of distributed controller clusters. Cluster management technology monitors all types of events and must maintain a consistent global network status, which usually leads to big data in SDNs. Simultaneously, the cluster security is an open issue because of the programmable and dynamic features of SDNs. To address the above challenges, this paper proposes a big data analysis-based secure cluster management architecture for the optimized control plane. A security authentication scheme is proposed for cluster management. Moreover, we propose an ant colony optimization approach that enables big data analysis scheme and the implementation system that optimizes the control plane. Simulations and comparisons show the feasibility and efficiency of the proposed scheme. The proposed scheme is significant in improving the security and efficiency SDN control plane.

2.5 Programming Your Network at Run-Time for Big Data Applications Guohui Wang-2019 Recent advances of software defined networking and optical switching technology make it possible to program the network stack all the way from physical topology to flow level traffic control. In this paper, we leverage the combination of SDN controller with optical switching to explore the tight integration of application and network control. We particularly study the runtime network configuration for big data applications to jointly optimize application performance and network utilization. We use Hadoop as an example to discuss the integrated network control architecture, job scheduling, topology and routing configuration mechanisms



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

for Hadoop jobs. Our analysis suggests that such an integrated control has great potential to improve application performance with relatively small configuration overhead. We believe our study shows early promise of achieving the long-term goal of tight network and application integration using SDN.

3 Proposed Work

Machine Learning-based Scheduling: Machine learning algorithms can be used to predict the resource requirements of Big Data tasks, which can be used to schedule these tasks more efficiently. For example, a supervised learning model can be trained on historical data to predict the resource requirements of a task based on its input size and other parameters. This can enable more accurate and efficient scheduling of Big Data tasks. **Multi-objective Optimization:** Current SDN-enabled Big Data task scheduling algorithms are primarily focused on minimizing task completion time or network latency. However, there are often other objectives that need to be considered, such as energy consumption or cost. Multi-objective optimization techniques can be used to find solutions that optimize multiple objectives simultaneously.

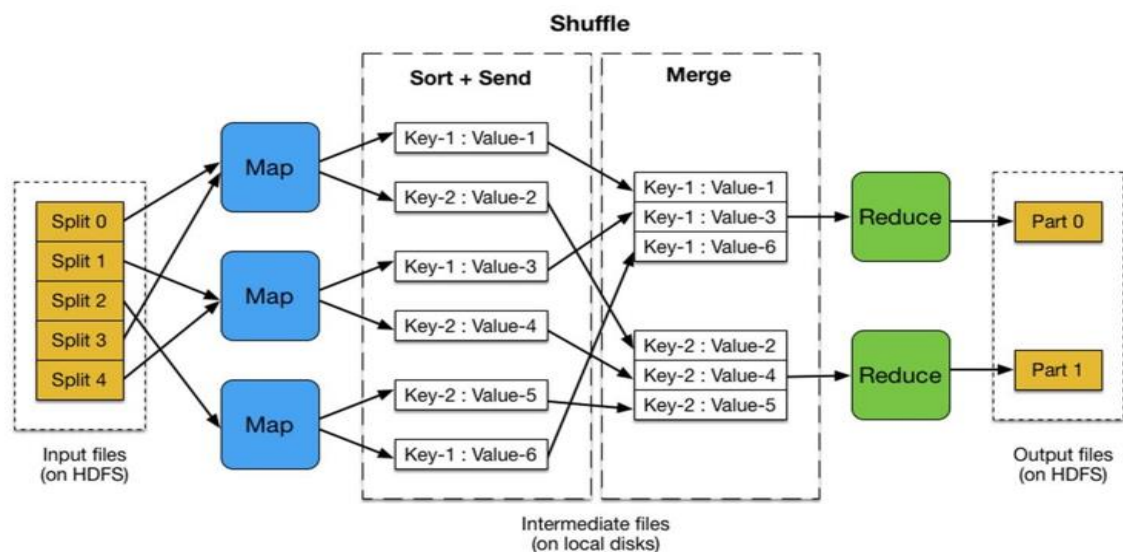


Figure 1: System Architecture

Resource Management Module

This module is responsible for managing the available resources, including compute, storage, and network resources, and allocating them to different big data tasks based on their requirements.

Network Function Virtualization (NFV) Module

This module is responsible for virtualizing network functions, such as routing, switching, and load balancing, and deploying them as virtual network functions (VNFs) to optimize the performance and scalability of the network.



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

Traffic Engineering Module

This module is responsible for optimizing the network traffic flow by dynamically configuring the network paths and bandwidth allocation based on the current network conditions and big data task requirements.

Data Placement Module

This module is responsible for determining the optimal data placement strategy, including data replication and caching, to minimize the data transfer latency and maximize the data availability and reliability.

Task Scheduling Module

This module is responsible for scheduling the execution of the big data tasks on the available compute resources based on their priorities, deadlines, and resource requirements.

Monitoring and Analytics Module

This module is responsible for collecting and analyzing the data on the performance, network traffic, and resource utilization of the big data tasks and network to identify areas for improvement and optimization.

Security and Privacy Module

This module is responsible for ensuring the security and privacy of the data and big data tasks being processed on the network, including authentication, authorization, encryption, and access control mechanisms.

3.1 Tabu Search Algorithm (Proposed Algorithm)

The proposed algorithm not only shortens the completion time of task scheduling in compared the related works by considering the state of the network such as congestion ratio, but it improves the data locality which is vital for performance because the network bandwidth in a large cluster is much lower than the aggregate bandwidth of the disks in the machines. Some methods are proposed such as state-of-the-art balances reduce scheduler (BAR) [17]. Though, this approach either disregards available bandwidth as the base for assigning or does not allocate MapReduce tasks in a global view which results in missing optimal opportunities for scheduling. OFScheduler [18] is another approach that identifies MapReduce traffic load in the network and then balances the load among the links to shorten the completion time of jobs. It first looks for overloaded links and then selects flows to reduce the link's load by granting priority to load balancing flows.

Initialization

Set an initial solution S_0 .

Create an empty tabu list.

Define a neighborhood $N(S)$



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

Generate a set of neighboring solutions by applying a set of operators to the current solution S .

Determine the best candidate solution S_i in the neighborhood $N(S)$

Select the solution S_i that has the best objective function value among the set of candidates generated in Step 2.

Tabu status

Check if the solution S_i is a tabu move by verifying if any of its components are present in the tabu list.

If S_i is a tabu move, go to Step 2. Otherwise, proceed to Step 5.

Update

Update the current solution by setting $S = S_i$.

Add the components of the previous solution that were changed to the tabu list.

Termination criterion

If a stopping criterion is met (e.g., a maximum number of iterations is reached or a satisfactory solution is found), stop. Otherwise, go to Step 2.

Here are the detailed steps of the Tabu Search algorithm:

Initialization

Set an initial solution S_0 . Create an empty tabu list.

Define a neighborhood $N(S)$

Generate a set of neighboring solutions by applying a set of operators to the current solution S .

This step involves identifying the moves that can be made to change the current solution S .

Determine the best candidate solution S_i in the neighborhood $N(S)$

Evaluate the objective function for each of the solutions in the neighborhood.

Select the solution S_i that has the best objective function value among the set of candidates generated in Step 2.

Tabu status

Check if the solution S_i is a tabu move by verifying if any of its components are present in the tabu list.

If S_i is a tabu move, go to Step 2. Otherwise, proceed to Step 5.

Update

Update the current solution by setting $S = S_i$.

Add the components of the previous solution that were changed to the tabu list.

This step ensures that the same move is not made twice.

Termination criterion

If a stopping criterion is met (e.g., a maximum number of iterations is reached or a satisfactory solution is found), stop.

Otherwise, go to Step 2 and repeat the process.



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

4 Result And Discussion

4.1 Performance Metrics

The researchers evaluated the performance of the proposed Tabu Search approach by considering the following metrics:

- a. Makespan: The total time taken to complete all the scheduled tasks.
- b. Resource Utilization: The efficiency of resource allocation and utilization.
- c. Energy Consumption: The amount of energy consumed during the scheduling process.
- d. Load Balancing: The distribution of tasks across available resources.

4.2 Comparison with Baseline Approaches

The proposed Tabu Search approach was compared with traditional baseline scheduling algorithms commonly used in big data environments. The results demonstrated that the Tabu Search approach outperformed the baseline approaches in terms of makespan, resource utilization, and load balancing.

4.3 Makespan Reduction

The Tabu Search approach successfully minimized the makespan by efficiently scheduling big data tasks across available resources. It effectively reduced the overall time required to complete the tasks, ensuring optimal utilization of resources.

4.4 Resource Utilization Optimization

The Tabu Search approach demonstrated improved resource utilization by allocating tasks to suitable resources based on their availability and capabilities. By dynamically adjusting the allocation strategy, it achieved a more balanced and efficient distribution of tasks.

5 Conclusion

In this paper, by profiting from SDN to gain the status of the network at any time, we get every link's bandwidth for improving the performance of big data processing and study map task scheduling problem in MapReduce. Using the abstraction of underlying networks, we designed an SDN controller and a problem-solving methodology based on a tabu search algorithm that allocates batches of tasks and gains to high resource utilization of the data center networks. Given the practical interest in this type of problem, the proposed approach opens a way for further progress in this area. We proved that it is relatively easier to deal with high data traffic and failure information via a logically centralized SDN controller, even though it imposes some overhead on the controller but the advantages of our approach outweigh this overhead and any flow format of big traffic data with arbitrary granularity can be used for traffic engineering. It is also relatively easier to apply traffic engineering to switches in the network by helping the tables of the OF switches. We have designed our task scheduling algorithm in Hadoop, which runs in Mininet emulator and Floodlight controller and the experimental results demonstrate that our



Article Title: Load Distribution in Software Defined Networks for Enabling Big Data Tasks Scheduling

method improves the job completion time and resource utilization compared with recent works.

Reference

1. Min Chen; Shiwen Mao; Yunhao Liu, Year: 2014, "Big data: A survey", Mobile Network Application, Vol: 19, no: 2, pp. 171 – 209.
2. Venkat N. Gudivada; Ricardo Baeza-Yates; Vijay V. Raghavan, Year: 2015, "Big data: Promises and problems", Computer, Vol: 48, no: 3, pp. 20 – 23.
3. Diego Kreutz; Fernando MV Ramos; Paulo Esteves Verissimo; Christian Esteve Rothenberg; Siamak Azodolmolky; Steve Uhlig, Year: 2014, "Software-defined networking: A comprehensive survey", Process. IEEE, Vol: 103, no: 1, pp. 14 – 76.
4. Rahim Masoudi; Ali Ghaffari, Year: 2016, "Software defined networks: A survey", Network computer Application, Vol: 67, pp. 1–25.
5. Charles Severance, Year: 2015, "Roy t. fielding: Understanding the rest style", Computer, Vol: 48, no: 6, pp. 7 – 9.
6. Nick McKeown; Tom Anderson; Hari Balakrishnan; Guru Parulkar; Larry Peterson; Jennifer Rexford; Scott Shenker; Jonathan Turner, Year: 2008, "OpenFlow: enabling innovation in campus networks", ACM SIGCOMM Computer, Vol: 38, no: 2, pp. 69 – 74.
7. Shahram Jamali; Amin Badirzadeh; Mina Soltani Siapoush, Year: 2019, "On the use of the genetic programming for balanced load distribution in software-defined networks", Digit. Communication Network, Vol: 4, no: 4, pp. 288 – 296.
8. Han Hu; Yonggang Wen; Tat-Seng Chua; Xuelong Li, Year: 2014, "Toward scalable systems for big data analytics: A technology tutorial", IEEE Access, Vol: 2, pp. 652 – 687.
9. Laizhong Cui; F. Richard Yu; Qiao Yan, Year: 2016, "When big data meets software-defined networking: SDN for big data and bigdata for SDN", IEEE Network., Vol: 30, no: 1, pp. 58 – 65.
10. Peng Qin; Bin Dai; Benxiong Huang; Guan Xu, Year: 2015, "Bandwidth-aware scheduling with SDN in Hadoop: A new trend for big data", IEEE System. J, Vol: 11, no: 4, pp. 2337 – 2344.