

# A Hadoop Job Scheduling Algorithm Based on Pagerank

**Weijian Huang, Hailong Yang**

*The Information and Electrical Engineering Academy of Engineering of Hebei University, Handan 056038, Hebei Province, China*

## Abstract

Aiming at the problem that the job scheduling algorithm based on the classical model of cloud computing in Hadoop is not high, the new job scheduling algorithm based on PageRank algorithm is proposed. Under the premise of ensuring the user experience, we propose a new job scheduling algorithm named ValidRank, which is based on the combination of hierarchical weight and waiting time. Then for the typical K Top problem and the PI problem of N, compared with the traditional job scheduling algorithm of Fair scheduler Fair and Capacity scheduler. Experimental results show that the improved ValidRank scheduling algorithm can improve the performance of the algorithm in multi-layer iterative environment.

Key words: MAP REDUCE, JOB SCHEDULER, PAGE RANK, VALID RANK

## 1. Introduction

With the rapid development of Internet companies such as Google, Amazon and Alibaba, cloud computing and big data have turned into two hot research fields. Since Google's distributed computing system was implemented by Apache foundation, Hadoop has gradually developed into a data management and processing system completely, however the core of this system is still the MapReduce system. Although the model of MapReduce distributed model is not complicated, there are still many problems need to be solved, so the optimization and improvement of MapReduce model for Hadoop is a hot research topic in recent years.

In order to improve operating efficiency, many scholars have done a lot of research on task scheduling in Hadoop, put forward many effective algorithms and application of measures to adapt to different environments. Such as in order to improve the performance of Hadoop scheduler, a possible approach to a research evaluation by calculating the renown of authors of scientific papers is proposed[1]. Some scholars have presented a fast random walk-based distributed algorithm for computing PageRank[2]. The innovation of joint consideration of MapReduce

jobs and the results is also proposed to improve reliability[3]. A resource prefetching method was proposed to improve the performance of scheduling jobs[4]. To save the time of executing jobs, a new parallel method of PageRank in Hadoop is proposed[5]. But for the type of job that has a dependency relationship, Such as DAG (Acyclic Graph Directed) iterative calculation, In mathematics, the optimal solution is obtained by the results of the obtained results. The optimization and improvement of the job scheduling algorithm above are not necessarily applicable, Moreover, the traditional job scheduling algorithm based on Hadoop is not very efficient in scheduling the job with a dependency relationship, So we need to further optimize the scheduling algorithm of Hadoop, so that it can maintain a high efficiency of scheduling.

In this paper, we propose a ValidRank algorithm. Firstly, the resource is pre-processed based on PageRank algorithm, Then we propose the hierarchical weights based on dependency relation which is similar to a child node in a tree. At last, the waiting time weight is calculated based on the waiting time. The VR algorithm based on PageRank not only considers the hierarchical weights of the tasks that have a dependency relationship, but also considers the waiting

time weights for the job with independent operations.

## 2. Job scheduling in Hadoop

In simple terms, Hadoop is composed of HDFS (Hadoop Distributed File System) to store data and MapReduce to process data. It is capable of batch processing of large amounts of data offline. There is a pluggable component called the scheduler in Hadoop, its function is to allocate the resources in the system according to a certain way to work. Before scheduling job, first of all, we should initialize the submitted job, encapsulating the information of tasks and records of the job waiting to be scheduled, it is used to monitor status of tasks in a job. Job scheduler first obtain task fragmentations which has been divided into splits from the Shared file system and create a task list, then assign each task an idle Map slot to run the task, setting the number of reduce's tasks by `setNumReduceTasks()`.

The job scheduling of Hadoop is dynamic, and it will bring some cost when scheduling a task. So the users can design and implement the scheduler of their own needs by inheriting `JobControl` class. There are three classic schedulers:

### (1) FIFO Scheduler

It is Hadoop default startup scheduler, The scheduler first sorted according to according to the priority of job waiting to be executed. And then according to the job to arrive at the time of the order to decide which job to perform.

### (2) Capacity Scheduler

The scheduler first allocating resources to multiple queues, the FIFO scheduling method is then used to schedule the queue. At the same time in order to avoid the high priority users to submit the job to monopolize cluster resources, it will limit the amount of resources per user.

### (3) Fair Scheduler

Being similar to capacity scheduler, It can not only have a number of queues, but also be able to allocate idle resources to each queue, and each queue in the job can be fair share of the queue of all resources.

In addition to above three common schedulers in Hadoop, in recent years many Hadoop schedulers for special scenarios have emerged. Such as the scheduler of Longest Approximate Time to End, based on the existing problem of detecting the presence of Hadoop in the homogeneous clusters, the efficiency is not high, a heterogeneous cluster scheduler is proposed[6]. The Deadline Scheduler can improve the real-time Hadoop operations is not high, as well as the the Constraint-based Schedule for the hard real-time requirements, this scheduler is used to optimize scheduler with complete time constraints[7],

such as to finish the job in a given deadline. According to the research of the Hadoop default scheduler and the predecessor of the scheduler algorithm, in the following paper, we hope to further research on the problem of the low efficiency of the scheduling problem with the improvement of the job scheduling algorithm.

## 3. Job scheduling algorithm based on PageRank

### 3.1. Resources preprocessing based on PageRank

PageRank (PR) is a kind of effective grade score algorithm. It is the application of Google in search engine and calculates the value of web pages, which is irrelevant to subject. The PageRank value in the algorithm is is on behalf of the importance of a web ,which is only related to the link structure of network[8]. The center of the algorithm is according to the number of positive links in the PageRank algorithm to determine the weight of a web page, first of all, pre-processing the cluster resource. In the process of submitting the job, different users may submit the same job, and different users or users of the job may be dependent, as shown in Figure 1, user A and user B both submit job2, While the implementation of the job 2 and job3 requires the results of job1. Job 4,5,6 also depends on the results of 2 jobs, it is similar to a DAG data structure, which constitutes a non return directed graph. At this point, there is a direct dependence between job2, job3 and job1. job 4, job5, job6 and job 1 have an indirect dependence. Usually the case, in the process of multi-user shared cluster resources, When the scale of the individual user submits a job is not very large, users exclusive cluster resources or user average share of cluster resources may affect the execution efficiency of other users to submit jobs, so the size of the user's weight is determined by the dependence of the user's job. To the users who are strong relied on by others give a greater weight, we priority to allocate more of the cluster resources, so as to help the rational use of cluster resources.

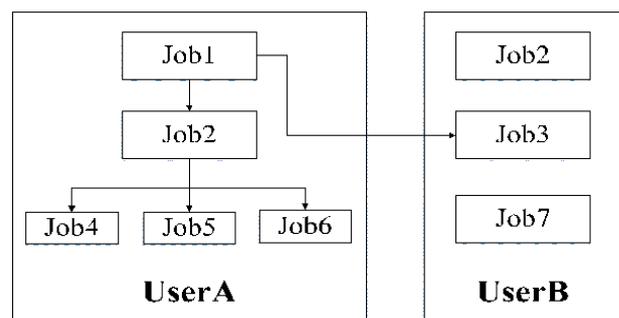


Figure1. The dependency relationship between different users

According to the idea of PageRank algorithm, through the idea of "The more users who are dependent on the job of other users, the more important users must be" to judge the importance of the user. An iterative calculation of the weight of the user A is shown in the formula (1):

$$PR(A) = PR(A) + d \left( \sum_{i=1}^n \frac{PR(mi)}{C(mi)} D(mi) \right) - \frac{PR(A)}{C(A)} D(A) \quad (1)$$

PR in formula is on behalf of the weight of user A, initial value is 1,  $m_1 \sim m_n$  represent n users all rely on the results of user A's jobs, C represents the total number of jobs submitted by a user  $m_i$  in a certain period of time, the PR with the ratio of C is the weight of a job of user  $m_i$ , D indicates that the number of jobs that need to be dependent on the results of the user A. In the process of scheduling operations, the user's weight may change with the increase of new users, d is a probability value that will not generate random scheduling with the increase of the job, it usually takes 0.85. The second part of formula (1) calculate the weight of user A which is dependent on by other users, that is, if the user A in the job is more dependent on other users, then the weight PR is greater, the last part of formula (1) represents the user A rely on the weight of the other users, that is, if the jobs of user A rely more on other users, the less weight of PR. User weights are dynamically changed along with the user's submitted job. When the available computing resources in the cluster are insufficient, the user's weight in the formula (1) will converge to PR (A).

### 3.2. The implementation of the ValidRank job scheduling algorithm

Suppose we have a number of users in the cluster, each user has a number of jobs. If the user's PageRank weight is assigned to an average of the respective job is clearly unwise and if only depend on the PR value calculated the weight of a single job is not complete. Although a job may be linked to a lot of jobs, but the weight is not necessarily the highest, because it may rely on the results of other jobs to be scheduled. The pre-optimization algorithm based on PageRank, higher weights are assigned to users who are linked to more users, but the calculation of the user's job weight also need to take into account the level of the job in the indirect dependency relationship, that is, the priority of the job execution. Such as the calculation of N after  $P_i$  need to calculate the values of the N-M ( $N > M$ ), if the request for the first N is in the job ahead of the first N-M, if the calculation period is too long, you can first calculate the job of the first N-M bit and recursively compute the job on the basis of the results of the N. Because the two users through the for-

mula (1) calculated the weights may be the same, so the job scheduling in addition to considering the efficiency of the implementation, but also related to if a user's job for a long time is not scheduled, the user experience problems. So the algorithm based on PageRank algorithm is improved, and the VR (ValidRank) algorithm is proposed. That in the process of scheduling job, in addition to rely on the user's PR value to calculate weight, we also rely on the hierarchy and waiting time of independent job.

(1) The establishment of the scheduling model under Hadoop

The improved job scheduling model is shown in Figure 2. Firstly, according to the idea of PageRank algorithm, the weight of the user is determined, that is, a user is selected, then the job is selected according to the VR algorithm, and the resource is allocated to the job [9].

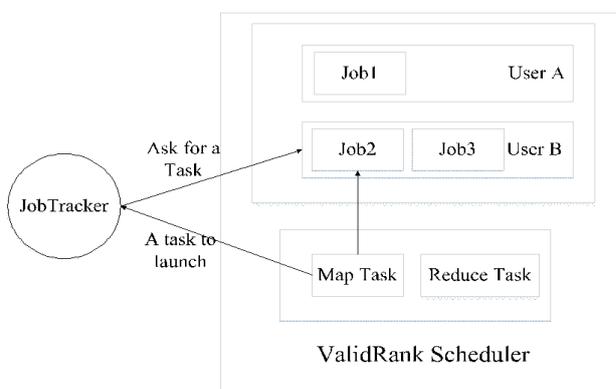


Figure 2. The job scheduling mode of improvement

(2) The calculating of the weight of the users' job  
 Firstly, the weight of the user PR (A) is calculated according to the formula (1), then according to the level of the user's job, the job i is calculated from the position of the network. The location of job is similar to the hierarchy of the nodes in the tree structure. Which directly depends on the relationship between the job is the father and son, indirect dependency assignment is the relationship between grandparents and grandchildren. The hierarchy of job i can be approximated by using the method of computing nodes in the tree and taking the reciprocal as the HR (I), the initial value is 1. As shown in the formula (2), m is the maximum number of jobs in the reliant network which have directly reliant relationship, it is similar to the degree of the nodes in the tree, N is the total number of jobs that are dependent on other jobs [10].

$$HR(i) = \frac{1}{\lfloor \log_m N \rfloor + 1} \quad (2)$$

For the job without reliance, that is, If a user's job is independent of other jobs, the weight is improved as the job waiting time is extended. As shown in the formula (3), the initial weight is 0, WR is the weight of the job, and the Run-Percent is the percentage of the task completed in the job. The number of completed tasks can be counted by the task of the heartbeat information statistics. Wait-Time is a time when an independence job has already waited. That the lower the completion of a job, the longer the waiting time, the higher the weight.[11].

$$WR(i) = \frac{1}{RunPercent(i)} * WaitTime(i) \quad (3)$$

Therefore, the improved job scheduling algorithm ValidRank is shown in the formula (4), VR is the abbreviation of it in rest of paper. In view of the traditional job scheduling algorithm, VR algorithm redefined the weight of the job that has a dependency relationship. When scheduling jobs in a multi-user environment, Firstly, based on PageRank, the user with high priority is selected, and then the user with high priority is given to scheduling, which not only takes into account the user's overall weight, but also takes into account the jobs' hierarchy weights of a user[12].

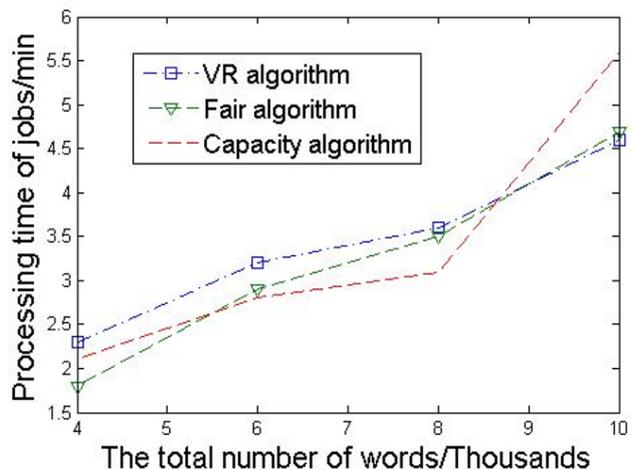
$$VR(i) = PR(A) + HR(i) + WR(i) \quad (4)$$

**4. Experimental verification and conclusion**

The job that has a dependency relationship is more common in mathematics, as in the geometric proof of the problem often encountered in the following. In solving the three-dimensional geometry problems due to the different needs, Some jobs require volume, and some jobs require for the area, and some other jobs need to find the circumference or based on the weight of the capital cost, but they may depend on the job of demanding side[13]. The dependency relationship among the users is the most obvious in the job with the iterative property, because the back of the results need to rely on the previous value, so by directly using the results in front of the results can improve the operation speed of the back jobs. Such as the calculation of the N after PI, we can directly calculate the rest N-M by other users may have been calculated to complete the calculation of the M (M<=N). In this experiment, we use the existing four idle computers to configure the Hadoop cluster to simulate the server cluster, one high performance computer as the Master node, configured as CPU clocked at 3.1GHz, memory 8GB, solid state hard drive 256GB. Three ordinary computers as Slave nodes, configured as 2.3GHz CPU, memory 2GB, Mechanical hard disk 320GB. Node names are Master, Node1, Node3, Node2. Each

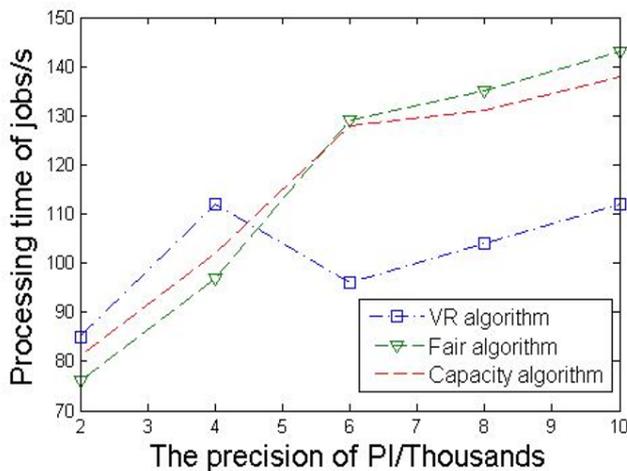
node uses the ubuntu-14.04 system, the cloud computing platform uses the open source Hadoop-1.0.0[14], the Java operating environment is jdk1.6.0\_27, the experiment data uses FactoryData5.5 (data factory) to generate. For a job with a dependency, experiments are carried out to verify the multi-layer iterative model respectively using Top K model with with two layer iterative relations and compute N after after PI with a multi iteration model. Aiming at the problem of top k, because it can be split into two MapReduce job, first of all, user A submit a job to count the number of words.

Then user B submit the job to the to find the highest frequency words before K, so the split of the two jobs have a certain dependence, namely the execution of the second tasks need to rely on the results of the first job[15]. Finally, user B submit the job to count the number of letters of the highest frequency K words. That is, the number of letters of the alphabet is required to rely on the results of the previous two jobs. For the problem of after N PI, experiments were conducted by different users with different values less than N. Contrast improved VR scheduling algorithm and improved VR scheduling algorithm's job processing time[16], as shown in Figure 3, 4.



**Figure3.** The processing time of Top K

As can be seen from Figure 3, for the Top K problem, in a time when the relationship is not deep, with the increase of data processing, the advantages of VR algorithm are not obvious. Job execution efficiency is similar to the fair scheduling algorithm. And for the sort of problem, Capacity algorithm consumes more time.



**Figure 4.** The processing time of calculate the N after PI

For a job with a multi dependency relationship, from Figure 4, we can see that with the deepening of the level, the results of the completed jobs can be used to improve the performance of the behind jobs, the calculation time is greatly saved. The improved VR algorithm has obvious advantages, compared with the traditional scheduling algorithm, it can significantly improve the performance of the iterative type jobs with multi-layer dependency relations.

**5. Conclusions**

Through the research of PageRank algorithm and MapReduce, the experimental results show that compared with other algorithms, the improved VR algorithm not only takes into account the short job priority and high load job priority, but also compared to the traditional job scheduling algorithm, the overall performance gap is not big. However, it is high to perform the efficiency of the iterative job with dependency relationship, it can effectively improve the efficiency of the operation of the Hadoop cluster. The next step we will carry out further research on the scheduling algorithm of the JobTracker node model in the heterogeneous environment.

**Acknowledgements**

This work was supported by Natural Science Foundation of Hebei Province of China in 2015.

**References**

1. Michal Nykl, Karel Jezek, Dalibor Fiala et al. (2014) PageRank variants in the evaluation of citation networks. *Journal of Informetrics*, 8(11), p.p. 683-692.
2. Atish Das Sarma, Anisur Rahaman Molla, Gopal Pandurangan, et al. (2015) Fast distributed PageRank computation. *Theoretical Computer Science*, 7(561), p.p. 113-121.
3. Chi-Yi Lin, Jih-Kai Liao (2014) A job-oriented load-distribution schem for cost-effective

- Name Node service in HDFS. *Int.J.Web and Grid Service*, 10(4), p.p. 319-337.
4. Wan Bing, Huang Mengxing, Duan Qian (2014) Hadoop job scheduling algorithm based on Prefetching. *Application Research og Computing*, 31(6), p.p. 1639-1643.
5. Ping Yu, Xiang Yang, Zhang Bo et al. (2014) Implementation of Paralle PageRank Algorithm Based on MapReduce. *Computer Engineering*, 40(2), p.p. 33-38.
6. Jiong Xie, Yun Tian, Shu Yin , et al. (2013) Adaptive Preshuffle in Hadoop clusters. *Procedia Computer Science*, 18(3), p.p. 2458-2467.
7. Bezerra A, Hernández P, Espinosa A, et al. (2013) Job scheduling for optimizing data locality in Hadoop clusters. *Proc of the 20th European MPI Users' Group Meeting*, New York, USA, p.p. 271-276.
8. B. Bahmani, K. Chakrabarti, D. Xin (2011) Fast personalized pagerank on mapreduce. *Proc.of ACM SIGMOD Conference*, Utah, USA, p.p. 973-984.
9. Fard H. M. , Prodan R. , Fahringer T. (2013) A truthful workflow schedeling mechanism for commercial multicloud environment. *IEEE transaction on parallel and distributed system*, 24(6), p.p. 1203-1212.
10. Yan, E., Ding, Y., & Sugimoto, C. R. (2010) P-Rank: An indicator measuring prestige in heterogeneous scholarly networks. *Journal of the American Society for Information Science and Technology*, 62(3), p.p. 467-477.
11. Su Sen, Li Jian,Huang Qinjia et al. (2013) Cost-efficient task scheduling for executing large programs in the cloud. *Parallel computing*, 39(45), p.p. 177-188.
12. Aysan Rasooli, Douglas G. Down (2014) Guidelines for Selecting Hadoop Schedulers Based on System Heterogeneity. *J Grid Computing*, 12(2), p.p. 499-519.
13. Yu, K., Chen, X., & Chen, J. (2012) A multidimensional PageRank algorithm of literatures. *Journal of Theoretical and Applied Information Technology*, 44(2), p.p. 308-315.
- HadoopReleases[EB/OL].[2014-09-12].<http://hadoop.apache.org/releases.html>.
14. Beloglazov A, Abawajy J, Buyya R. (2012) Energy-aware resource allocation heuristic for efficient management of data centers for cloud computing. *Future Generation Computer. Systems*, 28(5), p.p. 755-768.

15. Junho Choi, Chang Choi, Kangbin Yim et al. (2013) Intelligent Reconfigurable Method of

Cloud Computing Resources for Multimedia Data Delivery. *Informatica*, 24(3), p.p. 381-394.



# A Multi-Path DSDV-Based Routing Protocol for WIA-PA Network

**Chuan Gu, Min Li**

*School of Software Engineering, Anyang Normal University, Anyang 455000, Henan, China*

### Abstract

WIA-PA is a kind of wireless network for industrial process automation, which has a high requirement for the real-time and reliability of the data transmission. Based on the deficiency of the existing routing protocols, this paper puts forward a new kind of WIA-PA routing protocol - MWD. The protocol chooses the stablest link in the routing table as the optimal route, and then based on the node disjoint strategy, determines a route as the backup route, which disjoints with the optimal routing node. When the node needs to transmit data, it can search the local routing table and choose the optimal route to transmit data. If the optimal route fails sometime, it can transmit data through backup route. The simulation results show that MWD protocol can enhance the real-time and reliability of the WIA-PA significantly.

Key words: WIA-PA NETWORK, MULTI-PATH ROUTING PROTOCOL, HIGH RELIABILITY, HARD REAL-TIME

### 1. Introduction

Industrial wireless network technology is a new emerging kind of special wireless sensor network technology which is facing the information interaction between devices, and is suitable for harsh industrial field, and has a strong anti-interference, low power consumption, high real-time performance and high reliability. WIA-PA (Wireless Networks for Industrial Automation-Process Automation), which is used for the industrial automation process, is the specification for industrial wireless networks researched

and developed independently by China and became the IEC international standard in October 2008. It adopts the two layers of network topology combined with star and MESH [1]. The network routing protocol is the important support and guarantee for the high quality node network communication. However, up to now, this kind of network doesn't have an efficient, stable and mature routing protocol.

There are a lot of routing protocols concerning wireless networks. However different specific applications have different requirements. For the WIA - PA